

UNIVERZITET U SARAJEVU
ELEKTROTEHNIČKI FAKULTET
ODSJEK ZA RAČUNARSTVO I INFORMATIKU

PRAKTIKUM: POSLOVNI INFORMACIONI SISTEMI

Service Requirements Specification Dokument

Studenti:

BOJADŽIĆ *Hanna*, 1421/17311

GRAĐANIN *Ehvan*, 1438/17335

HALILBEGOVIĆ *Kemal*, 1469/16682

KRUPALIJA *Ehlimana*, 1431/17461

KUBAT *Ismar*, 1422/17148

KULOVIĆ *Nejra*, 1519/17484

MUFTIĆ *Belma*, 1423/17260

PRIBIŠIĆ *Tarik*, 1536/2018

Predmetni nastavnik:

doc. dr. ANEL TANOVIĆ, dipl. ing. el.

April, 2019

Sadržaj

1	ITIL procesi	1
1.1	<i>Change Management</i>	1
1.2	<i>Event Management</i>	2
2	Opis i namjena sistema	4
2.1	Osnovne informacije o kompaniji	4
2.2	Poslovni procesi	4
2.2.1	Upravljanje promjenama	5
2.2.2	Upravljanje događajima	6
2.3	Pregled slučajeva upotrebe	7
2.4	Metodologija razvoja sistema	8
2.5	Prototip sistema	10

1 ITIL procesi

1.1 Change Management

Upravljanje promjenama je ITIL (*Information Technology Infrastructure Library*) proces koji pripada skupu procesa u okviru tranzicije usluga. Njegova svrha je upravljanje promjenama koje se dešavaju nad nekim IT servisom, te olakšavanje prilagođavanja promjenama, kao i njihovom uvođenju i što boljem procesiranju zahtjeva za promjenama. U prvim verzijama ITIL-a promjene su bile generičke, no u kasnijim verzijama (od verzije 3 nadalje) prepoznata je potreba za uvođenjem tzv. modela promjena, koji razdvajaju promjene u različite kategorije prema značaju i hitnosti.

Pri upravljanju promjenama potrebno je definisati sljedeće pojmove:

1. **Zahtjev za promjenom (Request for Change - RFC)** - trebaju postojati jasno definisani slučajevi u kojima se prepozna potreba za vršenjem promjena. Promjene mogu inicirati osobe od interesa (*stakeholderi*, ukoliko je u pitanju IT usluga koju koristi neka kompanija i sl.), ili mogu biti vodene odzivom sistema, odnosno inicirane zbog grešaka, pogrešnog ili slabog korištenja nekih dijelova sistema. Zbog toga je važno omogućiti različitim osobama od interesa iniciranje promjena, kao i osigurati da se vrši praćenje korištenja sistema te analiza uspješnosti i frekvencije korištenja istog.
2. **Promjena** - moguće je vršiti različite akcije nad sistemom koje se smatraju promjenom: dodavanje novih komponenti, izmjenu ili potpuno uklanjanje starih, spajanje različitih funkcionalnosti i mnoge druge. Različite vrste promjena trebaju se tretirati na različite načine. Potrebno je jasno definisati šta je moguća promjena, koja je njen kategorija, koliki je njen prioritet te ko je zadužen za implementaciju iste.
3. **Osoblje za upravljanje promjenama** - treba postojati jasno definisana uloga za osobe koje će moći nadzirati promjene, odobravati vršenje istih ili odbacivati prijedloge. Od ovog osoblja očekuje se da je u stanju brzo i adekvatno reagovati na hitne promjene, kao i da izvrši ispravno rasuđivanje da li će neka promjena pozitivno ili negativno utjecati na sam sistem (potrebno je izvršiti analizu koja se dokumentuje u vidu *Projected Service Outage - PSO* dokumenta, u kojem su izlistane sve predvidljive devijacije od plana promjena).

Upravljanje promjenama sastoji se od četiri osnovna podprocesa:

1. **Filtriranje promjena**, u okviru kojeg se pregledaju svi primljeni zahtjevi za vršenje promjena, analiziraju se i vrši se odluka koji zahtjevi će se prihvati, a koji zahtjevi će se odbiti, uz što se prilaže PSO dokument.
2. **Savjetovanje sa CAB (Change Advisory Board i komitetom za hitne promjene**, što se odnosi na argumentovanje prethodno utvrđenih zahtjeva koji su prepoznati kao neophodni, te predstavljanje tih argumenata savjetima i komitetima koji jedini imaju autoritet za odobravanje promjena. Tek nakon što se promjene odobre, počinje se sa njihovom implementacijom. U svakom drugom slučaju, promjene se odbacuju.
3. **Upravljanje promjenama**, što se odnosi na dodjeljivanje posla zaposlenicima ovisno o vrsti zahtjeva koji je potrebno ispuniti te praćenje rada na dodavanju novih, zamjenom ili uklanjanjem postojećih funkcionalnosti u skladu s postojećim zahtjevima. Upravljanje promjenama vrši se stalno, od početka do kraja procesa njihovog uvođenja u sistem.

4. **Izvještavanje o promjenama**, odnosno kreiranje izvještaja o svakoj izvršenoj promjeni, što uključuje argumente za i protiv vršenja promjena, PSO dokument, dozvolu od CAB i drugih komiteta, kao i izvještaj o samom procesu implementacije i uvođenja promjena te kako je uvođenje istih u sistem promijenilo poslovanje, tj. da li je uvođenje promjena imalo pozitivan efekat ili ne.

1.2 Event Management

Upravljanje događajima je ITIL (*Information Technology Infrastructure Library*) proces koji pripada skupu procesa u okviru upravljanja uslugama. Njegova svrha je da nadzire sve značajne događaje u sistemu, detektuje nove i procesira ih na adekvatan način. Ovaj proces u *ITIL*-u 2011 povezan je sa upravljanjem pristupom preko interfejsa, budući da su ta dva procesa u bliskoj korelaciji: u sistemu koji nadzire važne, često sigurnosno osjetljive događaje, onemogućavanje neautorizovanog pristupa jedan je od najvažnijih zahtjeva.

Pri upravljanju događajima potrebno je definisati sljedeće pojmove:

1. **Događaj** - u kompleksnim sistemima postoje različite vrste događaja i notifikacija, te je potrebno napraviti razgraničenje kojim vrstama događaja se bavi sistem upravljanjem događajima (događaji se najčešće dijele na informacije, upozorenja i greške).
2. **Prioritet** - različiti događaji imaju različit značaj, pri čemu je neke događaje potrebno hitno procesirati, dok drugi događaji mogu čekati određeno vrijeme. Iz tog razloga neophodno je svim vrstama događaja dodijeliti određene prioritete, koji se najčešće označavaju brojevima.
3. **Odziv** - sistem najčešće treba reagovati na događaje, posebno ukoliko je njihov prioritet veliki. Iz tog razloga potrebno je definisati vrstu i vrijeme odziva na različite događaje, te način njihovog procesiranja.

Upravljanje događajima sastoji se od četiri osnovna podprocesa:

1. **Korištenje mehanizama za nadzor događaja**, koji se odnosi na stalnu mogućnost sistema da detektuje događaje, kao i da nadzire postojeće događaje i vrši njihovo procesiranje. Budući da je broj događaja koji konstantno ulaze u sistem veoma veliki, vrlo je važno ispravno izvršiti diferenciranje na različite vrste događaja, kako se ne bi detektovao veliki broj nevažnih ili nekritičnih događaja. S druge strane, sistem mora biti u mogućnosti reagovati na događaje na adekvatan način, posebno u slučaju kad se više različitih događaja treba procesirati u isto vrijeme.
2. **Prvi nivo korelacije: Filtriranje događaja**, odnosno analiza događaja kako bi se odredilo da li je događaj relevantan za sistem ili ne. Ukoliko je događaj samo informacija, najčešće nije potrebno vršiti nikakve akcije te se takvi događaji mogu ignorisati. No, upozorenja i greške se moraju procesirati na adekvatan način, te je vrlo važno imati adekvatne mehanizme kako takvi događaji ne bi bili ignorisani (ali i kako informacije ne bi bile procesirane, ukoliko to nije neophodno).
3. **Drugi nivo korelacije: Izbor odziva**, odnosno dalja analiza događaja koji nisu filtrirani na prethodnom nivou. Budući da je za upozorenja i greške najčešće potrebno imati adekvatan odziv, na ovom nivou potrebno je odrediti da li odziv može biti automatski (u pitanju je neki od poznatih događaja za koje postoji predefinisani koraci odziva), ili je potrebno eskalirati situaciju na viši nivo.

4. **Pregled događaja i njegovo zatvaranje**, koji se odnosi na dodatnu provjeru aktivnih događaja koji se trenutno nadziru. Ukoliko je automatska akcija bila dovoljna da se događaj procesira na adekvatan način i stanje vrati u normalno, te ukoliko je sistem proslijedio informacije o događaju individuama sistema koji su ga procesirali na adekvatan način, takav događaj moguće je zatvoriti, kako bi se novi događaji mogli procesirati bez čekanja.

2 Opis i namjena sistema

2.1 Osnovne informacije o kompaniji

Kompanija **Idea d.o.o. Sarajevo** je mlada kompanija koja se bavi razvojem i održavanjem informacionih sistema za državne institucije, poput Zavoda za izgradnju Kantona Sarajevo, Zavoda za informatiku i statistiku Kantona Sarajevo, Direkcije za robne rezerve Kantona Sarajevo i mnogih drugih. Ovi informacioni sistemi omogućavaju pregled svih zaposlenika, njihovih personalnih dosjeva (uključujući lične podatke, razne vrste dokumenata, rješenja i ugovora) te iniciranje različitih akcija, poput slanja na seminare, edukacije i stručna usavršavanja, vršenje ocjenjivanja radnika, donošenje različitih vrsta rješenja te vršenje sistematizacije.

Softver koji je razvijen potrebno je svakodnevno održavati. Korisnici sistema su najčešće srednje životne dobi, bez tehničkog znanja na visokom nivou, te se svakodnevno susreću s različitim poteškoćama u korištenju sistema. Ponekad dolazi do pojave grešaka, na koje je potrebno što prije reagovati, kako se ne bi ugrozilo ispravno i kontinualno funkcionisanje sistema. Same institucije često šalju zahtjeve za promjenama sistema, zbog stalnog donošenja novih odredbi te izmjene starih, zbog čega je sistem potrebno stalno ažurirati kako bi odgovarao trenutnom stanju i olakšavao poslovanje ovih institucija.

Iz svih ovih razloga, Idea d.o.o. Sarajevo ima potrebu za razvojem aplikacije koja će im omogućiti upravljanje događajima i promjenama u sistemu. Ovakva aplikacija olakšati će detekciju grešaka i zahtjeva za promjenama, ubrzati reagovanje na hitne promjene te osigurati da sistem kontinualno funkcioniše bez problema, te da njegovi korisnici mogu imati pouzdano i ažurirano okruženje u kojem će im biti jednostavno raditi.

2.2 Poslovni procesi

Kompanija želi informacioni sistem koji omogućava sljedeće funkcionalnosti:

- Pregled svih zahtjeva za promjenama od strane tima za upravljanje promjenama;
- Pisanje izvještaja (PSO dokumenta) o predloženoj promjeni te slanje istog komitetu za promjene na pregled;
- Konačno odobravanje ili odbijanje promjena od strane komiteta za promjene;
- Prihvatanje i odbijanje promjena, te njihovo označavanje kompletiranim nakon implementacije;
- Slanje promjena timu za produkciju;
- Prijavljivanje grešaka i problema pri implementaciji od strane tima za produkciju;
- Pregled statističke analize korištenja sistema, uz informacije o slabo ili nikako korištenim dijelovima;
- Prijava problema pri korištenju od strane korisnika sistema koji firma održava.

Sve ove funkcionalnosti trebaju biti objedinjene u jednoj aplikaciji, te mora biti osigurana sigurnost i konzistentnost podataka. U nastavku će biti prikazane osnovne informacije vezane za ITIL procese pomoću kojih je moguće modelirati poslovne procese sistema za koji je potrebno napraviti aplikaciju.

2.2.1 Upravljanje promjenama

Sama svrha sistema je praćenje korištenja softvera koji se održava te prijem i procesiranje zahtjeva za promjene. U skladu s tim, definišu se dvije kategorije promjena:

- *Obične promjene*, odnosno promjene koje inicijalizira tim za upravljanje promjenama na osnovu zahtjeva koje šalju državne institucije formalnim putem (van samog sistema za upravljanje promjenama), ili koje tim za upravljanje promjenama ocijeni neophodnim na osnovu statističke analize korištenja sistema (slabo ili nikako korišteni dijelovi sistema);
- *Hitne promjene*, koje se detektuju na osnovu prijave problema pri korištenju od strane korisnika sistema koje firma održava. Prijava problema korisnicima je omogućena unutar same aplikacije.

Osim kategorija, promjene imaju i svoje vrste, kako slijedi:

- Dodavanje novih komponenti;
- Izmjena postojećih komponenti;
- Brisanje postojećih komponenti.

Svaka promjena mora imati definisanu kategoriju i vrstu. Osoblje za upravljanje promjenama vrši njihov pregled i filtriranje - zahtjev se analizira, te se ili odbija uz obrazloženje, ili se šalje komitetu za promjene na dodatnu analizu i odobravanje. Komitet za promjene ima mogućnost pregleda svih promjena koje su im poslane od strane tima za upravljanje promjenama, te odbijanje ili odobravanje promjena. Te informacije prosljeđuju se timu za upravljanje promjenama, te ukoliko se odobri vršenje promjena, iste se dodjeljuju jednom od dostupnih timova za razvoj softvera.

Na osnovu ovih informacija, moguće je dodijeliti određene funkcionalnosti sistema različitim vrstama korisnika, na način kako slijedi:

- Pregled svih zahtjeva za promjenama - **tim za upravljanje promjenama**;
- Pisanje izvještaja (PSO dokumenta) o predloženoj promjeni te slanje istog komitetu za promjene na pregled - **tim za upravljanje promjenama**;
- Prihvatanje i odbijanje promjena, te njihovo označavanje kompletiranim nakon implementacije - **tim za upravljanje promjenama**;
- Konačno odobravanje ili odbijanje promjena - **komitet za promjene**;
- Slanje promjena timu za produkciju - **tim za upravljanje promjenama**;
- Pregled statističke analize korištenja sistema, uz informacije o slabo ili nikako korištenim dijelovima - **tim za upravljanje promjenama**.

2.2.2 Upravljanje događajima

Događaje u sistemu predstavljaju prijave problema od strane korisnika sistema koji se održavaju, ili od strane razvojnog tima koji vrši implementaciju izmjena u sistemu. Pregled i nadziranje događaja vrši tim za upravljanje promjenama, dok korisnici sistema i razvojni tim mogu vršiti iniciranje novih događaja.

U skladu s ovim, potrebno je izvršiti definisanje svih različitih događaja u sistemu, njihovih tipova i prioriteta, što je prikazano u Tabeli 1.

Vrsta događaja	Inicijator	Prioritet
Greška u sistemu	Korisnik	3
Pogrešno korištenje sistema	Korisnik	1
Nemogućnost implementacije promjene	Razvojni tim	2
Greška nakon uvođenja promjena	Korisnik	3
Greška pri testiranju sistema	Razvojni tim	2

Tabela 1: Svi događaji sistema

Svi događaji prikazuju se timu za upravljanje promjenama, koji vrše odziv na događaje, na jedan od tri načina:

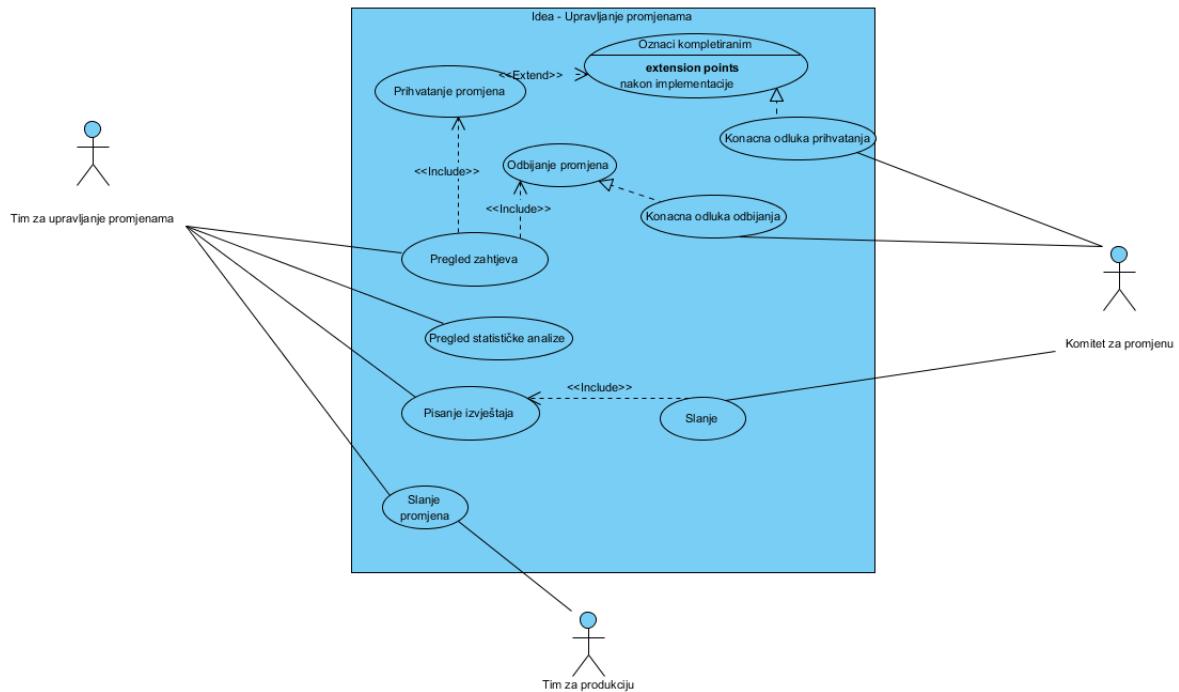
- *Otkazivanje promjena*, koje se vrši ukoliko razvojni tim prijavi da nije u mogućnosti da implementira promjene iz nekog razloga (poput samog načina na koji sistem funkcioniše, nedovoljnog broja informacija o samoj promjeni koju je potrebno izvršiti, nemogućnosti povezivanja sa postojećim sistemom i sl.).
- *Slanje razvojnom timu na popravljanje*, koje se vrši ukoliko korisnici prijave grešku u sistemu prije ili nakon uvođenja promjena ili ukoliko dođe do greške pri testiranju sistema.
- *Organizacija edukacije korisnika ili razgovor s korisnikom*, koji se vrše ukoliko korisnik prijavi da ne zna kako ispravno koristiti sistem.

Na osnovu ovih informacija, moguće je dodijeliti određene funkcionalnosti sistema različitim vrstama korisnika, na način kako slijedi:

- Prijavljanje grešaka i problema pri implementaciji promjena - **razvojni tim**;
- Prijava problema pri korištenju sistema koji firma održava - **korisnici sistema koji se održava**.

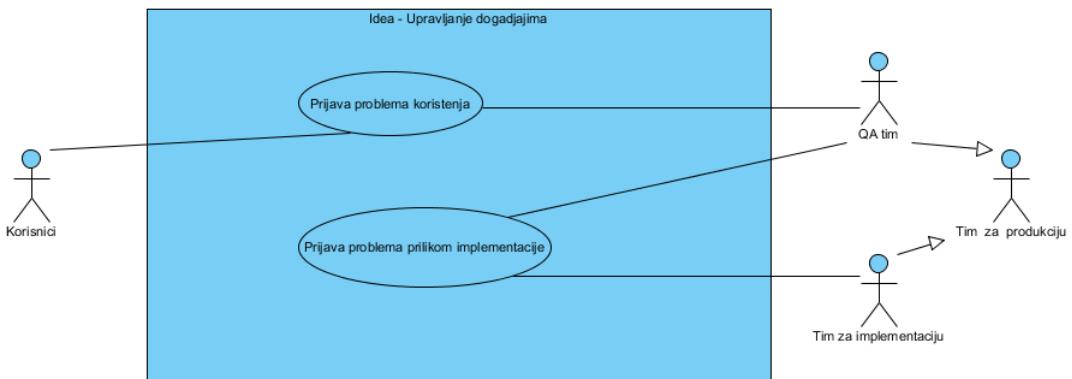
2.3 Pregled slučajeva upotrebe

Slučajevi upotrebe sistema koji odgovaraju funkcionalnostima vezanim za upravljanje promjenama prikazani su na Slici 1. Akteri sistema su tim za upravljanje promjenama, komitet za promjene i tim za produkciju, koji učestvuju u realizaciji slučajeva upotrebe.



Slika 1: Dijagram slučajeva upotrebe za upravljanje promjenama u sistemu

Slučajevi upotrebe sistema koji odgovaraju funkcionalnostima vezanim za upravljanje događajima prikazani su na Slici 2. Akteri sistema su korisnici sistema i tim za produkciju, koji učestvuju u realizaciji slučajeva upotrebe.



Slika 2: Dijagram slučajeva upotrebe za upravljanje događajima u sistemu

2.4 Metodologija razvoja sistema

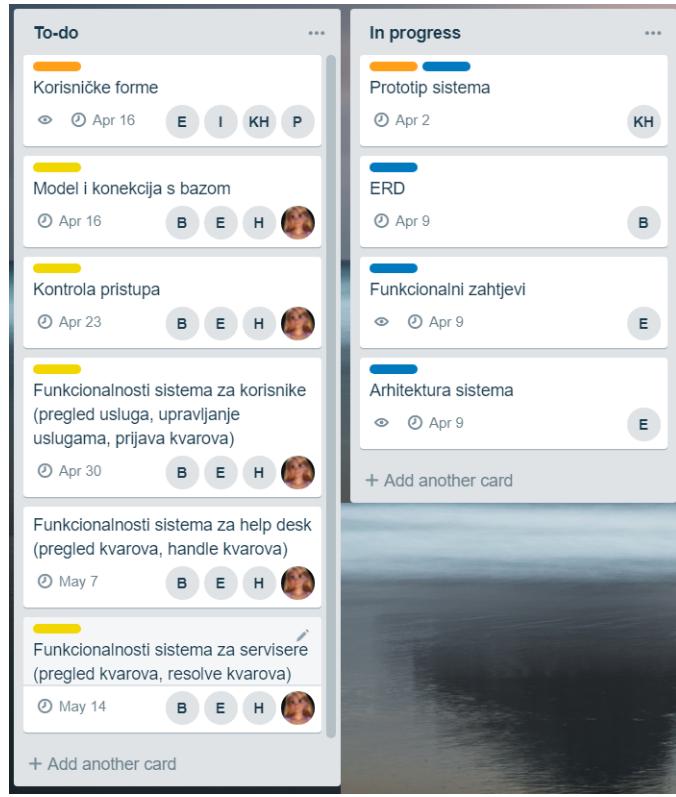
Za razvoj sistema biti će korištena SCRUM metodologija, koja predstavlja jednu od najpoznatijih agilnih metodologija za razvoj softvera. Razvoj sistema obavljati će se u sprintovima, sa trajanjem svakog sprinta od po 7 dana, pri čemu će se funkcionalnosti sistema razdvojiti prema korisničkim ulogama. Dodatno, *task-ovi* će biti razdvojeni na **frontend** i **backend** taskove, a novi zadaci biti će dodavani po potrebi (ovisno o mogućnostima raspodjele većih zadataka na manje podzadatke).

Inicijalni *Trello Board* prikazan je na Slici 3 i 4. Na slikama je vidljivo da su zadaci podijeljeni u nekoliko različitih grupa:

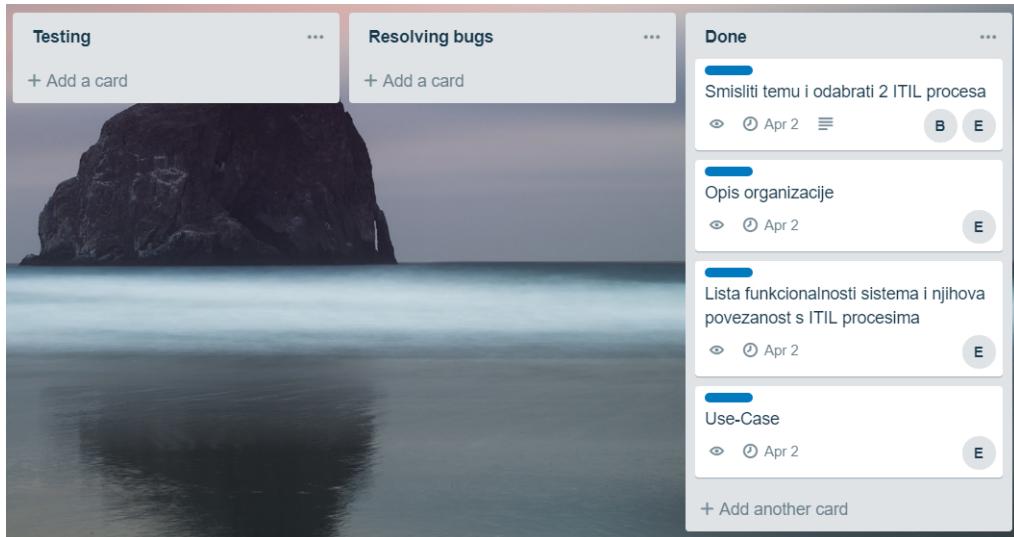
- *To-do* - skup svih zadataka koje je potrebno izvršiti;
- *In progress* - skup zadataka koji su ocijenjeni najvažnijim po određenom kriteriju (vrijeme isporuke ili kritičnost za sistem), te koji se trenutno izvršavaju;
- *Testing* - skup zadataka koji su implementirani no potrebno je izvršiti njihovu verifikaciju;
- *Resolving bugs* - skup zadataka iz prethodnih inkremenata u kojima su pronađene greške, te koje se trenutno ispravljaju;
- *Done* - skup svih zadataka koji su uspješno završeni.

Zadaci se dijele u tri skupine:

1. *Documentation*, što se odnosi na pisanje dokumentacije vezane za projekat;
2. *Backend*, što se odnosi na implementaciju funkcionalnosti te persistencije podataka;
3. *Frontend*, što se odnosi na implementaciju korisničkog interfejsa.



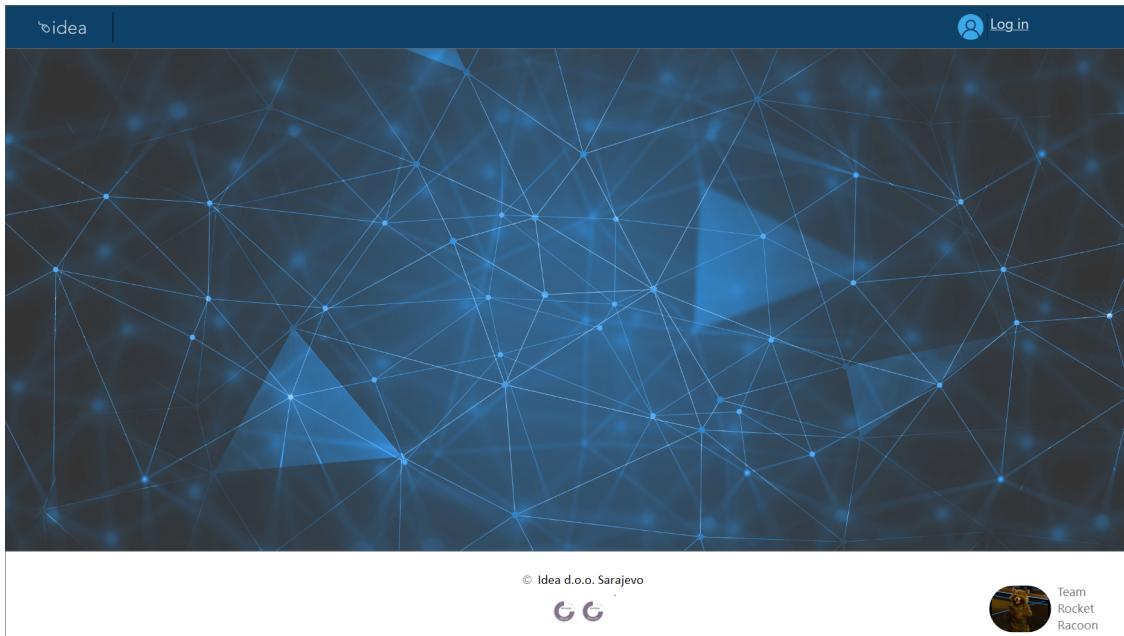
Slika 3: *Trello board* projekta, zadaci koji se trebaju izvršiti i koji se trenutno izvršavaju



Slika 4: *Trello board* projekta, zadaci koji su završeni te dijelovi sistema koji se testiraju ili popravljaju

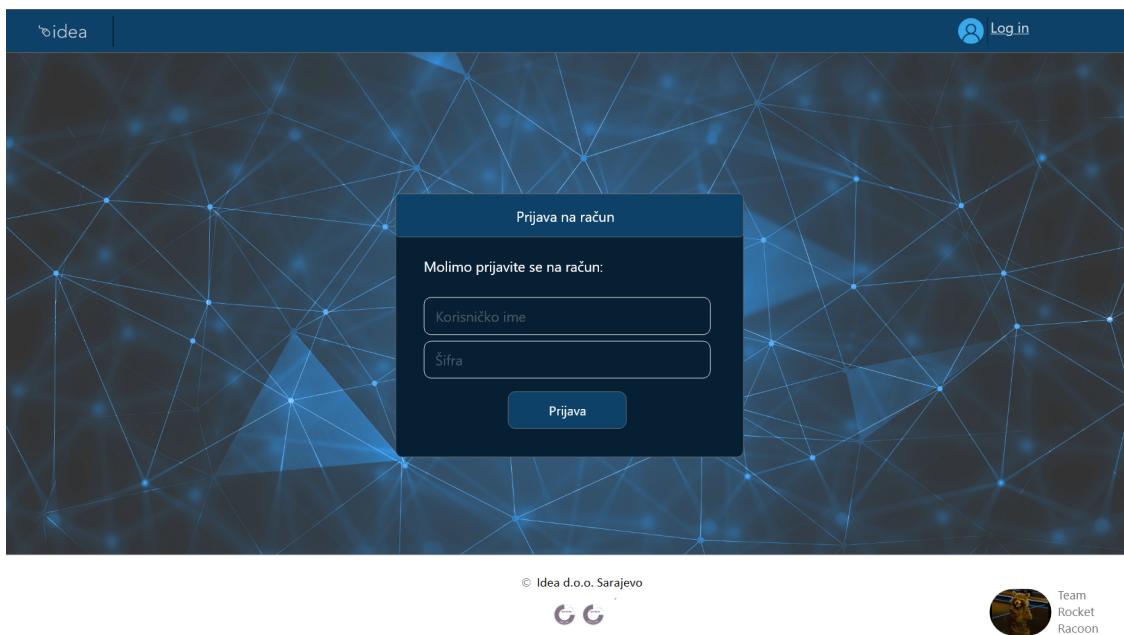
2.5 Prototip sistema

Na Slici 5 prikazan je osnovni izgled ekrana aplikacije, koja ima korisnički interfejs kreiran poštujući pravila dobrog dizajna.



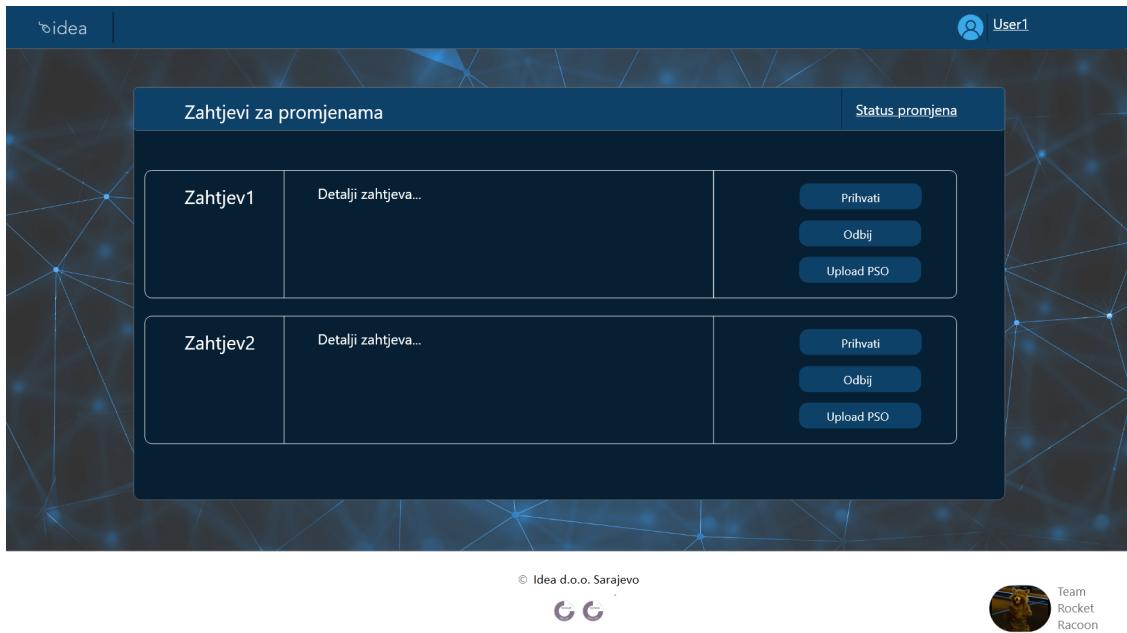
Slika 5: Osnovni izgled ekrana aplikacije

Na Slici 6 prikazan je izgled interfejsa za unos korisničkih podataka, nakon čega se omogućava pristup aplikaciji za različite vrste korisnika.



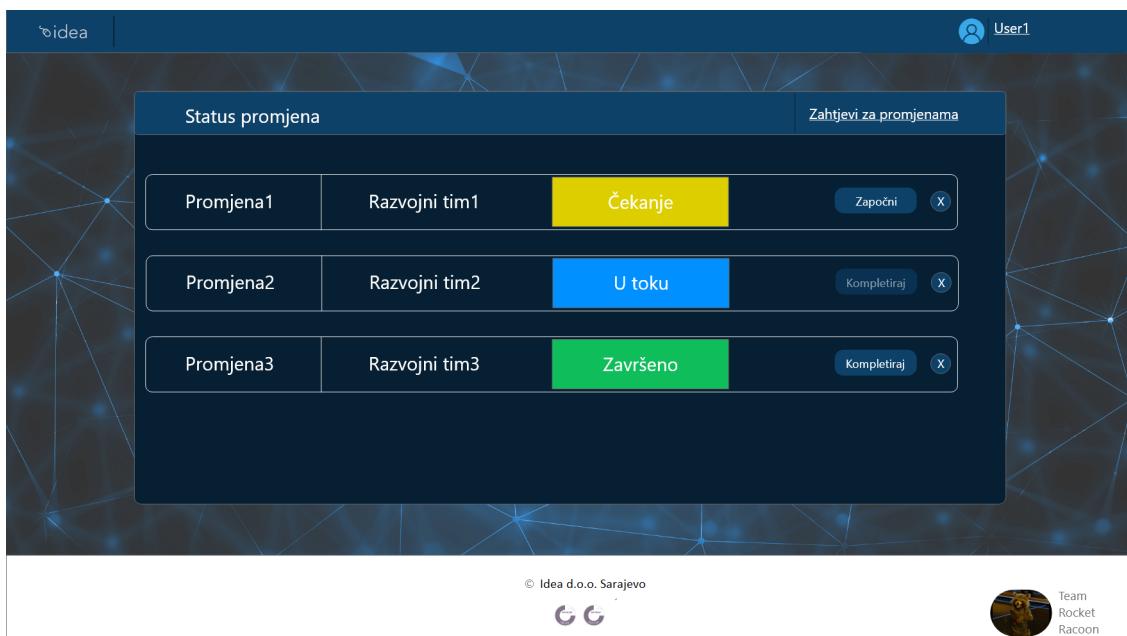
Slika 6: Izgled interfejsa za unos korisničkih podataka

Na Slici 7 prikazan je izgled ekrana za pregled svih zahtjeva za promjenama, koje može vršiti tim za upravljanje promjenama. Nakon pregleda promjene, vrši se njena analiza i *upload-uje* se PSO dokument, te se pritiskom na dugme **Prihvati** promjena automatski šalje komitetu za promjene. Alternativno, promjena se može odmah odbiti.



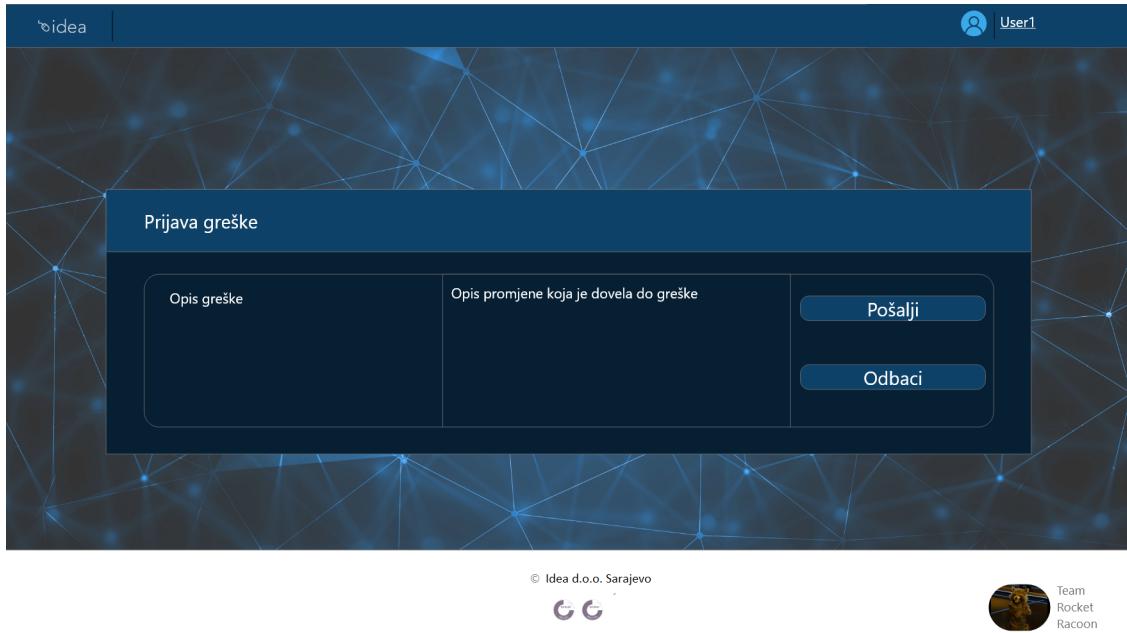
Slika 7: Izgled interfejsa za pregled i procesiranje promjena

Na Slici 8 prikazan je izgled ekrana za pregled svih odobrenih promjena, koje može vršiti tim za upravljanje promjenama. Promjene se mogu poslati razvojnog timu, te se vršiti njihov kontinualni nadzor i otkazivanje, ukoliko se pokaže da je promjenu nemoguće, vrlo teško ili neisplativo ugraditi u sistem.



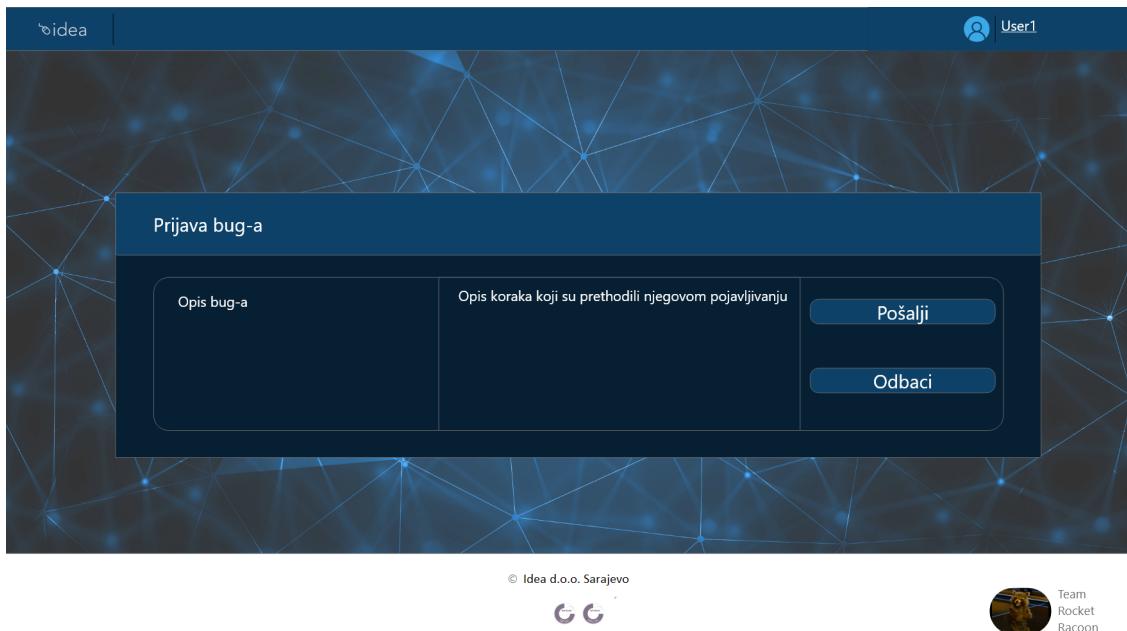
Slika 8: Izgled interfejsa za pregled statusa odobrenih promjena

Na Slici 9 prikazan je izgled ekrana za prijavu greške od strane korisnika sistema koji se održava. Nakon unosa neophodnih informacija, greške se evidentiraju u sistemu kao događaji koje zatim pregleda tim za upravljanje promjenama te na koje adekvatno reaguje.



Slika 9: Izgled interfejsa za prijavu greške u sistemu

Na Slici 10 prikazan je izgled ekrana za prijavu *bug-a* od strane razvojnog tima. Nakon unosa neophodnih informacija, *bug-ovi* se evidentiraju u sistemu kao događaji koje zatim pregleda tim za upravljanje promjenama te na koje adekvatno reaguje.



Slika 10: Izgled interfejsa za prijavu *bug-a* u sistemu