

Oblikovanje programske potpore

Ak. God. 2017./2018.

Upravljanje stambenom zgradom

Dokumentacija, Rev. 2

Grupa: *Eureka*

Voditelj: *Toni Martinčić*

Datum predaje:

18.01.2018.

Mentor: *Katarina Potkonjak*

Popis članova grupe i zadatci:

Toni Martinčić – voditelj projekta, koordinacija članova projekta, kontrola kvalitete rada, dizajn UML dijagrama, arhitektura sustava, implementacija web aplikacije

Ivana Mršić – arhitektura sustava, UML dijagrami, baza podataka, implementacija web aplikacije

Marko Ćurlin – arhitektura sustava, UML dijagrami, baza podataka, implementacija web aplikacije

Marko Cavalli – arhitektura sustava, UML dijagrami, baza podataka, implementacija web aplikacije

Luka Hrgović – arhitektura sustava, UML dijagrami, baza podataka, testiranje

Ivan Biškup – arhitektura sustava, UML dijagrami, baza podataka, testiranje

Sadržaj:

- 1. Dnevnik promjena dokumentacije4**
 - 2. Opis projektnog zadatka6**
 - 3. Pojmovnik8**
 - 4. Funkcionalni zahtjevi9**
 - 5. Ostali zahtjevi34**
 - 6. Arhitektura i dizajn sustava35**
 - 6.1. Svrha, opći prioriteti i skica sustava35**
 - 6.2. Dijagram razreda s opisom38**
 - 6.3. Dijagram objekata47**
 - 6.4. Ostali UML dijagrami48**
 - 7. Implementacija i korisničko sučelje55**
 - 7.1. Dijagram razmještaja55**
 - 7.2. Korištene tehnologije i alati56**
 - 7.3. Isječak programskog koda vezan za temeljnu funkcionalnost57**
 - 7.4. Ispitivanje programskog rješenja61**
 - 7.5. Upute za instalaciju81**
 - 7.6. Korisničke upute82**
 - 7.6.1. Upute stanaru, predstavniku i upravitelju82**
 - 7.6.2. Upute administratoru85**
 - 8. Zaključak i budući rad89**
 - 9. Popis literature91**
- Dodatak A: Indeks (slika, dijagrami, tablica, ispis koda)92**
- Dodatak B: Dnevnik sastajanja96**
- Dodatak C: Prikaz aktivnosti grupe98**
- Dodatak D: Pregled rada i stanje ostvarenja100**

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autor(i)	Datum
0.1	Napravljen predložak. Dodan opis projektnog zadatka, popis članova i zadatci.	Martinčić, Mršić	30.10.2017.
0.2	Dodan pojmovnik.	Cavalli	30.10.2017.
0.3	Dodani funkcionalni zahtjevi.	Mršić, Hrgović	30.10.2017.
0.31	Dodani ostali zahtjevi.	Biškup, Ćurlin	30.10.2017.
0.4	Dodan popis literature.	Martinčić	31.10.2017.
0.5	Opis obrazaca uporabe.	Biškup, Ćurlin	4.11.2017.
0.51	Dodan obrazac uporabe.	Ćurlin	11.11.2017.
0.6	Dodani sekvencijski dijagrami.	Biškup, Hrgović, Mršić	11.11.2017.
0.7	Svrha, opći prioriteti i skica sustava.	Biškup, Martinčić	11.11.2017.
0.8	Dijagram razreda s opisom.	Cavalli	11.11.2017.
0.9	Dijagram objekata.	Hrgović	12.11.2017.
1.0	Prva verzija dokumentacije.		12.11.2017.
1.1	Ostali dijagrami.	Hrgović, Cavalli	3.1.2017.
1.2	Dijagram razmještaja.	Ćurlin	3.1.2017.
1.3	Korištene tehnologije i alati.	Mršić	3.1.2017.
1.4	Isječak programskog koda vezan za temeljnu funkcionalnost	Biškup	4.1.2017.
1.5	Ispitivanje programskog rješenja.	Biškup, Ćurlin	4.1.2017.
1.51	Dodani slučajevi u ispitivanje programskog rješenja.	Ćurlin, Hrgović	5.1.2017.

1.6	Korisničke upute.	Mršić, Cavalli	10.1.2017.
1.7	Upute za instalaciju.	Martinčić	17.1.2017.
1.8	Dnevnik sastajanja.	Biškup	17.1.2017.
1.9	Indeks slika.	Mršić	18.1.2017.
1.91	Dnevnik aktivnosti.	Martinčić	18.1.2017.
2.0	Druga verzija dokumentacije.		18.1.2017.

Tablica 1 Dnevnik promjena dokumentacije

2. Opis projektnog zadatka

Cilj projekta je praktično primijeniti postupke oblikovanja programske potpore na rješavanje konkretnih problema upravljanja stambenim zgradama. Također izraditi projektnu dokumentaciju i implementaciju svih zadataka, te omogućiti preglednost i dostupnost svih funkcionalnosti i sadržaja.

Funkcionalnosti koje se žele postići su jednostavnija i lakša komunikacija između samih stanara i svih ostalih zainteresiranih stranaka. Također osigurati mogućnost objavljivanja svih stranaka na oglasnoj ploči. Same objave bi imale vrijeme nastanka i mogući istek objave, odnosno vrijeme kada bi se objava maknula sa oglasne ploče, ako korisnik tako naloži kod unosa. Isto tako korisnik bi mogao maknuti vlastitu objavu u bilo kojem trenutku. Druga funkcionalnost bi bila raspored čišćenja snijega gdje bi korisnik mogao vidjeti svoj termin čišćenja, i ako mu možda isti ne odgovara, mogao bi predložiti zamjenu stanaru čiji mu termin odgovara. Isto tako treba omogućiti generiranje naloga, te svim stanarima omogućiti pregled tijeka zajedničkog novca u bilo kojem trenutku, te pregled vlastitih uplata i svih plaćenih i ne plaćenih pričuva. Omogućiti svim stanarima prijavljivanje određene štete, gdje bi imali podjelu na hitne štete koje bi se automatski provodile i manje hitne gdje bi za to trebala suglasnost predstavnika suvlasnika. Još neke od funkcionalnosti bi bile dogovaranje sastanaka putem aplikacije i nekih zajedničkih druženja i sastanaka. Potrebno je napomenuti da će svaki korisnik moći obavljati samo one akcije unutar aplikacije koje će mu biti unaprijed predodređene privilegijama.

Predviđa se izrada web aplikacije koja bi se sastojala od početne stranice gdje bi se korisnici prijavljivali pomoću svojih korisničkih imena i lozinki. Nakon prijave bi se korisniku otvorio ekran sa oglasnom pločom gdje bi mogao vidjeti sve obavijesti koje su navedene u gornjem odlomku, te isto tako dodavati nove. Korisnik bi također mogao otvoriti stranicu sa nalozima gdje bi imao uvid u sve naloge zgrade i stanara. Treća stranica bi predstavljala troškove gdje bi svaki korisnik mogao pratiti tijek novca i troškove zgrade. Te bi zadnji ekran predstavljao mogućnost preglada osobnih podataka korisnika koje bi mogao promijeniti, podatke zgrade u kojoj stanuje ili posjeduje stan, te bi imao mogućnost uključiti ili isključiti notifikacije.

Sve akcije korisnika, koje su nepredviđene i ne ispravne su pokrivenе unutar sustava.

Sustav će osim stanara kao korisnika, imati još nekoliko različitih tipova korisnika:

1. *Predstavnik stanara* – osoba koja ima sve ovlasti kao i obični stanar, ali ima mogućnost određivanja prioriteta obavljanja određenih poslova u zgradi, kao i potvrđivanje naloga koji su unešeni u sustav, te bi imao mogućnost brisanja svih objava i naloga po potrebi.

2. *Upravitelj zgrade* - ovlaštena firma koja vodi brigu o financijama zgrade i slanju osoba za određene popravke na samoj zgradi, te o izvršavanju naloga, odnosno provođenju i plaćanju obavljenih naloga, te potvrđivanju uplaćenih pričuva.

3. *Administrator* – osoba koja se brine o samoj stranici i promjenama na njoj. Isto tako ima mogućnost dodavanja i brisanja novih korisnika i dodjeljivanje određenih privilegija određenim osobama, što je u ovom slučaju predstavnik stana i upravitelj zgrade.

3. Pojmovnik

Apache Tomcat: open source aplikacijski web poslužitelj za pokretanje i prikaz web stranica.

CSS (Cascading Style Sheets): stilski jezik koji se koristi za opis prezentacije dokumenta napisanog pomoću HTML jezika.

Git: distribuirani sustav za upravljanje izvornim kodom.

Hibernate: objektno/relacijska tehnologija koja povezuje objekte iz Java s relacijskim tablicama baze podataka.

HTML (HyperText Markup Language): prezentacijski jezik za izradu web stranica.

IntelliJ IDEA: Java integrirano razvojno okruženje za razvoj računalnog softvera.

Java: objektno orijentirani programski jezik koji se može koristiti za izradu web aplikacija.

JavaScript: skriptni programski jezik, koji se izvršava u web pregledniku na strani korisnika.

React: biblioteka JavaScript programskog jezika koji služi za izradu korisničkog sučelja.

Redux: spremnik koji sadrži stanje JavaScript aplikacije.

Spring Boot: olakšava stvaranje samostalnih aplikacija koje se temelje na Spring proizvodima koje možete „samo pokrenuti“.

SQL: standardni programski jezik za manipuliranje, pohranu i preuzimanje podataka u bazama podataka.

UML (Unified Modeling Language): standardizirani jezik za slikovno predstavljanje i modeliranje objekata.

4. Funkcionalni zahtjevi

Dionici:

- Stanar
- Predstavnik stanara
- Upravitelj zgrade
- Administrator

Aktori i njihovi funkcionalni zahtjevi:

- Stanar – inicijator
 - Prijavljuje se lozinkom i korisničkim imenom
 - Prijavljuje štetu, kvar i izgubljene/pronađene stvari
 - Zatražuje promjenu termina čišćenja snijega po potrebi
 - Uvid u raspored čišćenja snijega, naloge, troškove i opće informacije o zgradama, kao i vlastite opće informacije
 - Objavljuje na oglasnoj ploči
 - Izmjenjuje vlastite obavijesti na oglasnoj ploči
 - Briše vlastite objave
- Predstavnik stanara – inicijator
 - Prijavljuje se lozinkom i korisničkim imenom
 - Prijavljuje štetu, kvar i izgubljene/pronađene stvari
 - Određuje važnost i redoslijed popravaka i radova
 - Ažurira i potvrđuje naloge
 - Zatražuje promjenu termina čišćenja snijega po potrebi
 - Objavljuje raspored čišćenja snijega
 - Uvid u raspored čišćenja snijega, naloge, troškove i opće informacije o zgradama, kao i vlastite opće informacije
 - Objavljuje na oglasnoj ploči
 - Uređuje vlastite objave na oglasnoj ploči
 - Briše objave po potrebi
- Upravitelj zgrade – inicijator
 - Prijavljuje se lozinkom i korisničkim imenom
 - Obrađuje i ažurira naloge
 - Dogovara posao sa izvođačima radova
- Administrator – inicijator
 - Upisuje i briše korisnike iz sustava
 - Upravlja ovlastima
 - Održava šifrarnike

- Baza podataka – sudionik
 - Čuva podatke o svim stanarima, troškovima i nalozima
 - Čuva aktivnosti na oglasnoj ploči i raspored čišćenja snijega
 - Čuva podatke o svim zgradama i stanovima

Opis obrazaca uporabe:

- **UC1 – PrijavljanjeKorisnika**
 - **Glavni sudionik:** Stanar, Predstavnik, Upravitelj, Administrator.
 - **Cilj:** Prijaviti se u sustav kako bi mogao koristiti aplikaciju.
 - **Sudionici:** Baza podataka
 - **Preduvjeti:** Da ima korisnički račun.
 - **Rezultat:** Korisnik je prijavljen i ima pristup sadržaju i funkcijama stranice.
 - **Željeni scenarij:**
 - 1) Korisnik unosi svoje korisničko ime i lozinku.
 - 2) Sustav provodi provjeru podataka, te upućuje korisnika na početnu stranicu.
 - **Mogući drugi scenarij:**
 - 1) Korisnika nema u bazi podataka, te ga sustav vraća na stranicu za prijavu.
 - 2) Korisnik je unio krivu lozinku i korisničko ime, te ga sustav vraća na stranicu za prijavu.
- **UC2 – PogledajOglasnuPloču**
 - **Glavni sudionik:** Stanar, predstavnik, upravitelj.
 - **Cilj:** Pogledati trenutno stanje na oglasnoj ploči.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik je prijavljen.
 - **Rezultat:** Prikaz svih objava na oglasnoj ploči.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju za pregled oglasne ploče.
 - 2) Sustav prebacuje korisnika na oglasnu ploču, te mu prikazuje oglasnu ploču sa objavama.
 - **Mogući drugi scenarij:** -

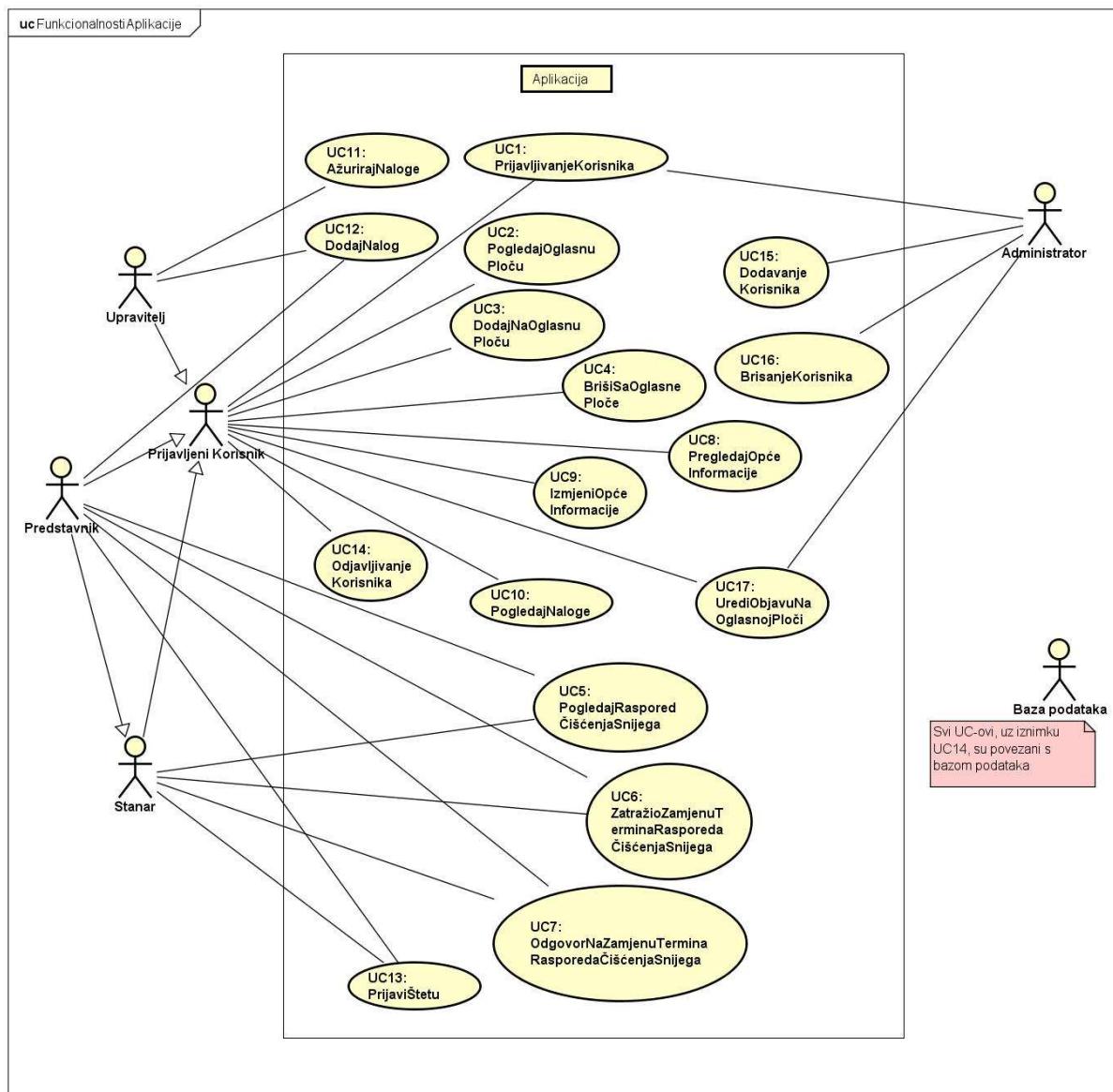
- **UC3 – DodajNaOglasnuPloču**
 - **Glavni sudionik:** Stanar, predstavnik, upravitelj.
 - **Cilj:** Dodati nove objave na oglasnoj ploči.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik je ulogiran, te se nalazi na oglasnoj ploči.
 - **Rezultat:** Korisnik dodaje podatke na oglasnu ploču.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju dodavanja objave.
 - 2) Sustav prikazuje formu za unos podataka objave.
 - 3) Korisnik unosi podatke i potvrđuje unos.
 - 4) Sustav provjerava unesene podatke, bilježi dodavanje te osvježava oglasnu ploču.
 - **Mogući drugi scenarij:**
 - 1) Korisnik odustaje od objavljivanja. Scenarij završava.
 - 2) Pogrešno su uneseni podaci, sustav odbija unos korisniku, te mu to javlja porukom.
- **UC4 – BrišiSaOglasnePloče**
 - **Glavni sudionik:** Stanar, predstavnik, upravitelj.
 - **Cilj:** Obrisati objave na oglasnoj ploči.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik je ulogiran, te se nalazi na oglasnoj ploči.
 - **Rezultat:** Korisnik briše podatke na oglasnoj ploči.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju brisanja objave.
 - 2) Sustav bilježi brisanje te osvježava oglasnu ploču.
 - **Mogući drugi scenarij:** -
- **UC5 – PogledajRasporedČišćenjaSnijega**
 - **Glavni sudionik:** Stanar i predstavnik.
 - **Cilj:** Pogledati trenutno stanje na rasporedu čišćenja snijega.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik je prijavljen.
 - **Rezultat:** Otvara se stranica sa rasporedom čišćenja snijega.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju za pregled rasporeda čišćenja snijega.
 - 2) Sustav prikazuje korisniku trenutno stanje rasporeda čišćenja snijega.
 - **Mogući drugi scenarij:** -

- **UC6 – ZatražioZamjenuTerminaRasporedaČišćenjaSnijega**
 - **Glavni sudionik:** Stanar i predstavnik.
 - **Cilj:** Zatražiti zamjenu termina u rasporedu čišćenja snijega.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik je prijavljen, te se nalazi na toj stranici.
 - **Rezultat:** Korisnik traži zamjenu termina.
 - **Željeni scenarij:**
 - 1) Korisnik odabire termin koji mu odgovara.
 - 2) Sustav bilježi zahtjev korisnika u bazi, te šalje poruku korisniku čiji je datum odabran.
 - **Mogući drugi scenarij:** -
- **UC7 – OdgovorNaZamjenuTerminaRasporedaČišćenjaSnijega**
 - **Glavni sudionik:** Stanar i predstavnik.
 - **Cilj:** Korisnik prihvata ili odbija zamjenu u rasporedu čišćenja snijega.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik je prijavljen, te se nalazi na toj stranici.
 - **Rezultat:** Korisnik odgovara na zahtjev za zamjenu termina.
 - **Željeni scenarij:**
 - 1) Korisnik prihvata zamjenu termina.
 - 2) Sustav šalje poruku o prihvaćanju zamjene korisniku koji je zatražio zamjenu.
 - 3) Sustav osvježava promjene u rasporedu čišćenja snijega.
 - **Mogući drugi scenarij:**
 - 1) Zamjena odbijena, te sustav javlja korisniku koji je zatražio izmjenu porukom o odbijanju zamjene.
- **UC8 – PregledajOpćeInformacije**
 - **Glavni sudionik:** Stanar, predstavnik, upravitelj.
 - **Cilj:** Dobiti uvid u opće informacije korisnika i zgrade.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik mora bit prijavljen.
 - **Rezultat:** Otvara se stranica sa općim informacijama korisnika i zgrade.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju za pregled općih informacija.
 - 2) Sustav prikazuje informacije o korisniku i zgradu.
 - **Mogući drugi scenarij:** -

- **UC9 – IzmjeniOpćeInformacije**
 - **Glavni sudionik:** Stanar, predstavnik, upravitelj.
 - **Cilj:** Izmijeniti opće informacije korisnika ili zgrade.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik mora bit prijavljen, te se nalaziti na toj stranici.
 - **Rezultat:** Korisnik izmjenjuje informacije.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju izmjene vlastitih podataka ili podataka o zgradi.
 - 2) Sustav prikazuje formu za izmjenu podataka.
 - 3) Korisnik unosi nove podatke.
 - 4) Sustav provodi provjeru podataka i ažurira unesene podatke.
 - **Mogući drugi scenarij:**
 - 1) Uneseni podatci su neispravni, te sustav odbija akciju uz poruku.
 - 2) Korisnik nema privilegiju za izmjenu odabranih podataka.
- **UC10 – PogledajNaloge**
 - **Glavni sudionik:** Stanar, predstavnik, upravitelj.
 - **Cilj:** Dobiti uvide u naloge.
 - **Sudionici:** Baza podataka
 - **Preduvjeti:** Korisnik mora biti prijavljen.
 - **Rezultat:** Otvara mu se stranica sa nalozima na njegov zahtjev.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju za pregled naloga.
 - 2) Sustav prikazuje korisniku sve naloge.
 - **Mogući drugi scenarij:** -
- **UC11 – AžurirajNaloge**
 - **Glavni sudionik:** Upravitelj.
 - **Cilj:** Ažurirati postojeće naloge.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik mora bit ulogiran i nalaziti se na toj stranici.
 - **Rezultat:** Korisnik ažurira odabrane naloge.
 - **Željeni scenarij:**
 - 1) Korisnik označava plaćene pričuve i naloge.
 - 2) Sustav osvježava i sprema promjene.
 - **Mogući drugi scenarij:** -

- **UC12 – DodajNaloge**
 - **Glavni sudionik:** Upravitelj, predstavnik.
 - **Cilj:** Dodati nove naloge.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik mora bit ulogiran i nalaziti se na toj stranici.
 - **Rezultat:** Korisnik dodaje nove naloge.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju za dodavanje novog naloga.
 - 2) Sustav prikazuje formu za unos naloga.
 - 3) Korisnik unosi podatke novog naloga i potvrđuje ih.
 - 4) Sustav provjerava i ažurira unesene podatke.
 - **Mogući drugi scenarij:**
 - 1) Uneseni podatci su neispravni, te sustav odbija akciju uz poruku.
- **UC13 – PrijaviŠtetu**
 - **Glavni sudionik:** Stanar, predstavnik.
 - **Cilj:** Prijaviti nastalu štetu i kvar.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik je ulogiran.
 - **Rezultat:** Šteta je prijavljena.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju za unos štete.
 - 2) Sustav prikazuje formu za unos štete.
 - 3) Korisnik unosi sve potrebne podatke za opis štete.
 - 4) Sustav provjerava ispravnost podataka, te ažurira unesene podatke.
 - **Mogući drugi scenarij:**
 - 1) Uneseni podatci su neispravni, te sustav odbija akciju uz poruku.
- **UC14 – OdjavljivanjeKorisnika**
 - **Glavni sudionik:** Stanar, predstavnik, upravitelj.
 - **Cilj:** Odjaviti korisnika iz aplikacije.
 - **Sudionici:** -
 - **Preduvjeti:** Korisnik je prijavljen.
 - **Rezultat:** Korisnik se odjavljuje, te se otvara stranica za prijavu korisnika.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju za odjavu.
 - 2) Sustav odjavljuje korisnika, te ga vraća na stranicu za prijavu.
 - **Mogući drugi scenarij:** -

- **UC15 – DodavanjeKorisnika**
 - **Glavni sudionik:** Administrator.
 - **Cilj:** Dodati novog korisnika.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik nije dodan, administrator je prijavljen.
 - **Rezultat:** U bazu podataka se dodaje novi korisnik.
 - **Željeni scenarij:**
 - 1) Administrator odabire opciju za dodavanje korisnika.
 - 2) Sustav prikazuje formu za unos novog korisnika.
 - 3) Administrator unosi podatke i potvrđuje ih.
 - 4) Sustav provjerava podatke, te dodaje korisnika .
 - **Mogući drugi scenarij:**
 - 1) Uneseni su krivi podatci, te sustav javlja pogrešku.
 - 2) Postoji korisnik s istim privilegijama(predstavnik, upravitelj), te sustav javlja grešku.
- **UC16 – BrisanjeKorisnika**
 - **Glavni sudionik:** Administrator
 - **Cilj:** Obrisati postojećeg korisnika.
 - **Sudionici:** Baza podataka.
 - **Preduvjeti:** Korisnik već postoji, administrator je prijavljen.
 - **Rezultat:** Korisnik ne postoji više u bazi.
 - **Željeni scenarij:**
 - 1) Administrator odabire korisnika koji se briše.
 - 2) Sustav osvježava bazu podataka.
 - **Mogući drugi scenarij:** -
- **UC17 – UrediObjavuNaOglasnojPloči**
 - **Glavni sudionik:** Stanar, predstavnik, upravitelj, administrator.
 - **Cilj:** Urediti objave na oglasnoj ploči.
 - **Sudionici:** Baze podataka.
 - **Preduvjeti:** Korisnik je ulogiran.
 - **Rezultat:** Korisnik uređuje podatke na oglasnoj ploči.
 - **Željeni scenarij:**
 - 1) Korisnik odabire opciju uređivanja objave.
 - 2) Sustav prikazuje formu za uređivanje objave.
 - 3) Sustav provjerava unesene promjene, te ih ažurira.
 - **Mogući drugi scenarij:**
 - 1) Uneseni su krivi podatci tokom uređivanja, te sustav odbija uređivanje porukom.

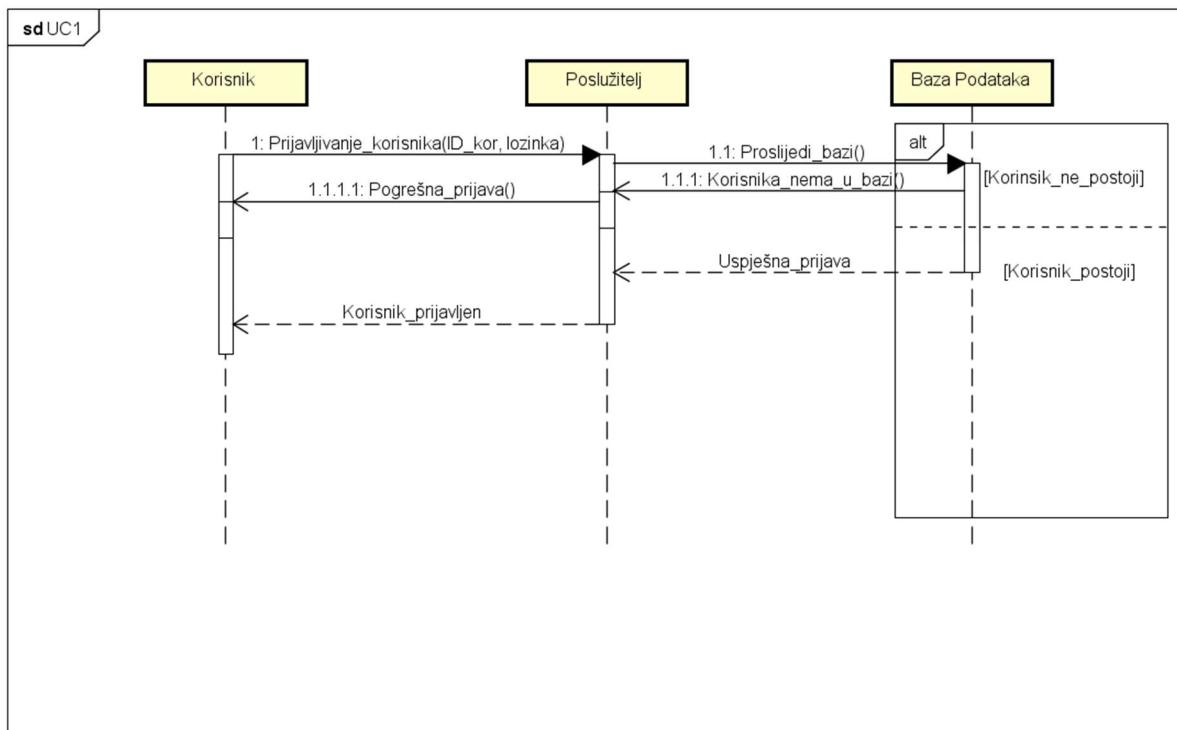


Slika 1 Funkcionalnosti aplikacije

Sekvencijski dijagrami :

Obrazac uporabe UC1 (PrijavljivanjeKorisnika)

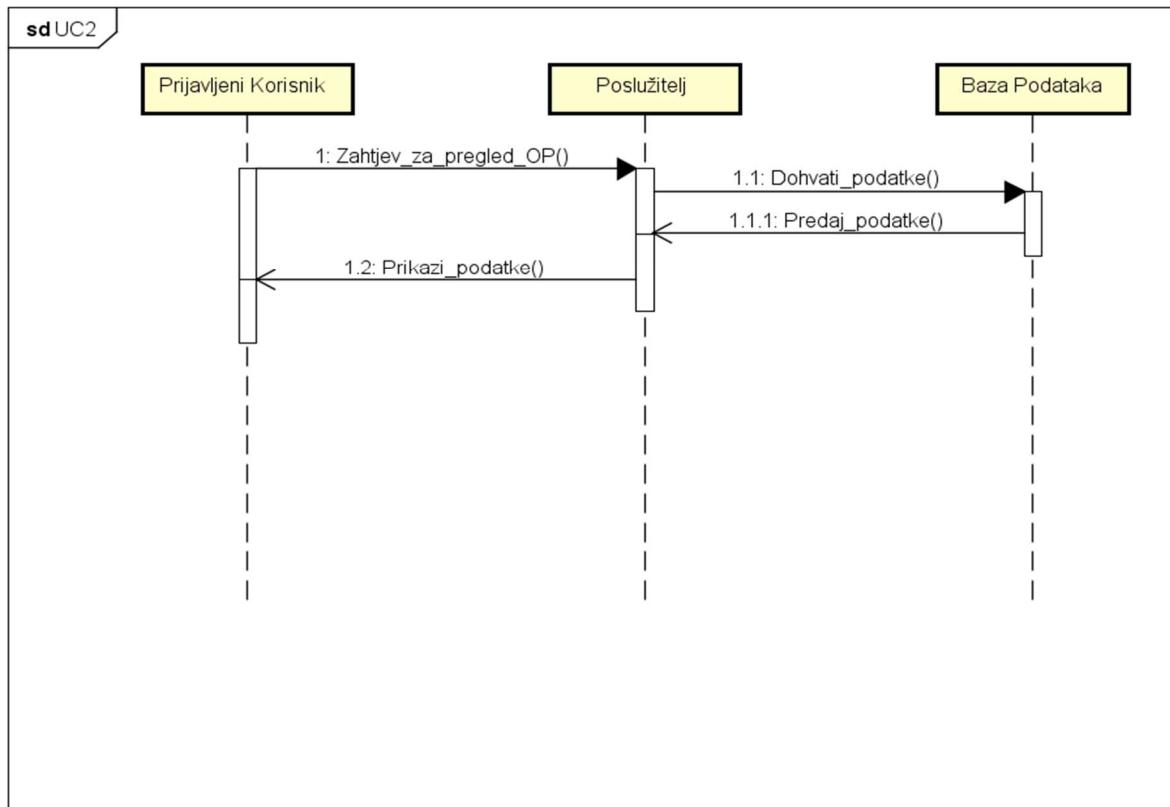
Korisnik se prijavljuje na sustav sa svojim ID-em i lozinkom. Poslužitelj prosljeđuje ID i lozinku bazi koja provjerava da li korisnik postoji u bazi. Ukoliko korisnik ne postoji, baza obavještava poslužitelja koji javlja korisniku da je pogrešna prijava. U suprotnom korisnik se uspješno prijavljuje u sustav.



Slika 2 Sekvencijski dijagram obrasca PrijavljivanjeKorisnika

Obrazac uporabe UC2 (PogledajOglasnuPloču)

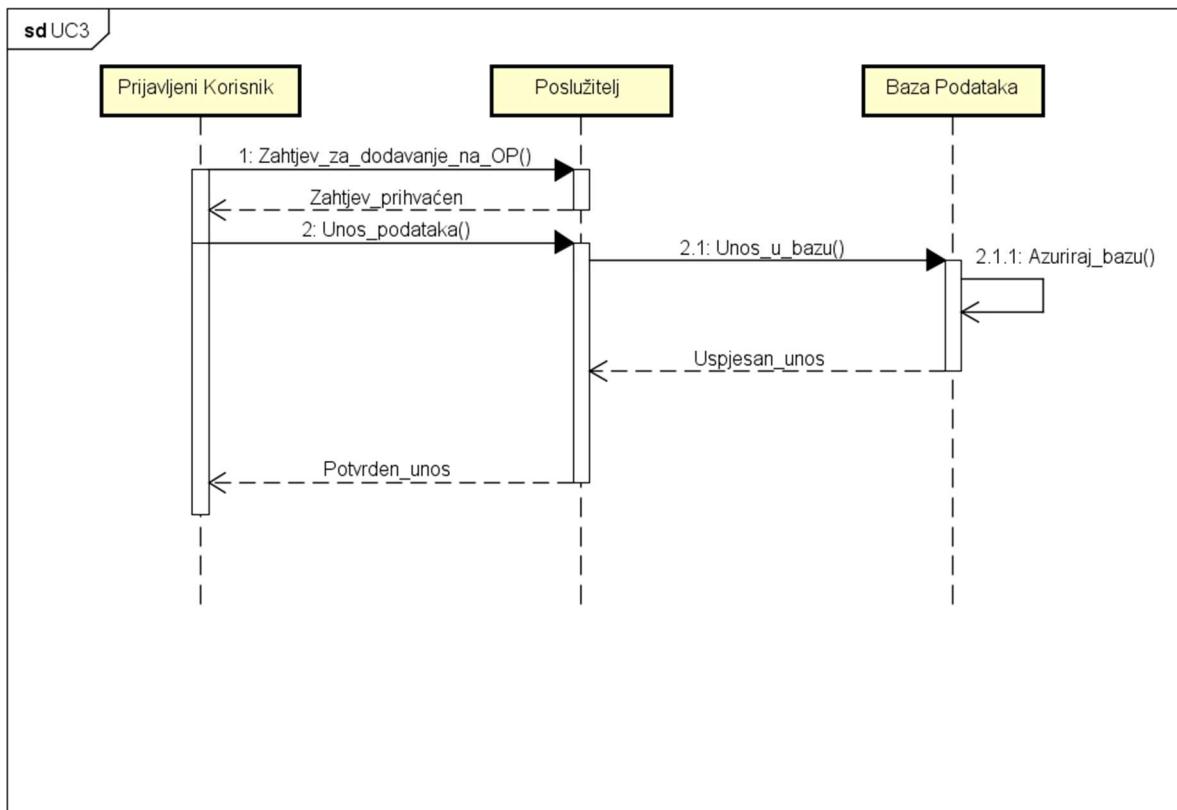
Prijavljeni korisnik zahtjeva pregled oglasne ploče. Poslužitelj dohvaća sve podatke iz baze podataka i daje ih na uvid korisniku.



Slika 3 Sekvencijski dijagram obrasca PogledajOglasnuPloču

Obrazac uporabe UC3 (DodajNaOglasnuPloču)

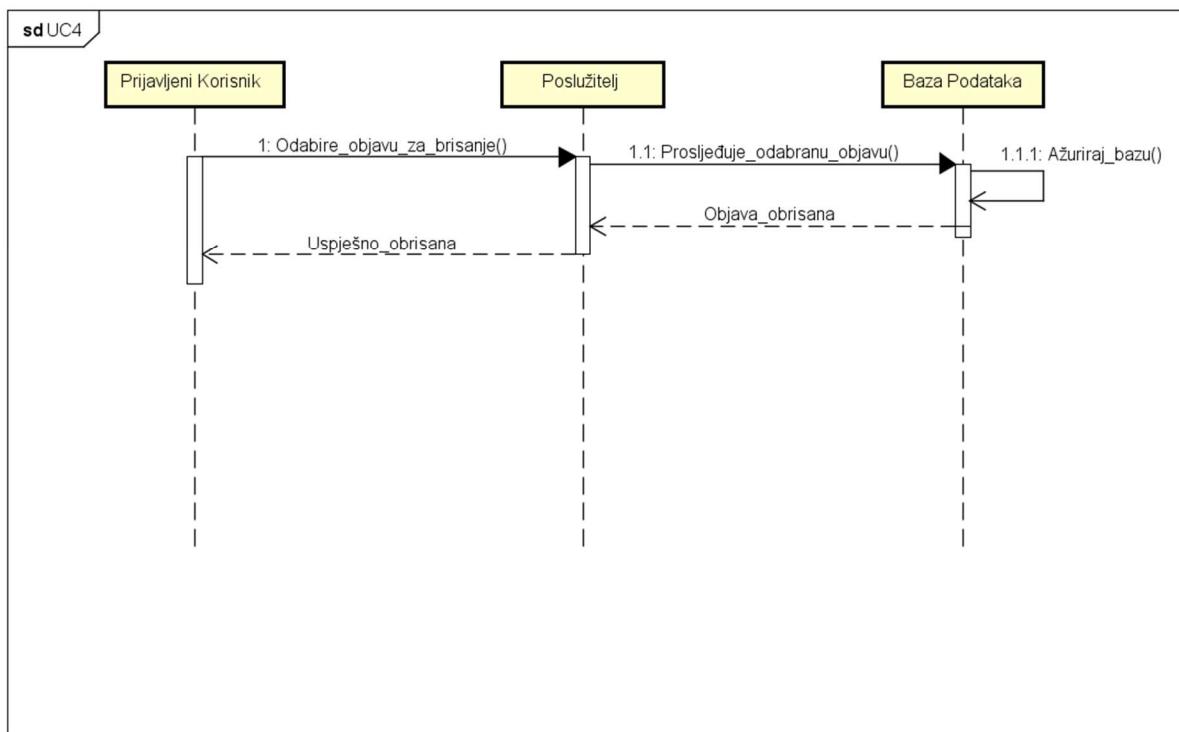
Korisnik zahtjeva dodavanje obavijesti na oglasnu ploču, što mu poslužitelj dozvoljava. Zatim korisnik unosi sve potrebne podatke koje želi dodati, poslužitelj ih prosljeđuje bazi podataka koja ih unosi i sprema. I na kraju korisnik dobiva obavijest da je njegov unos uspješno napravljen.



Slika 4 Sekvencijski dijagram obrasca DodajNaOglasnuPloču

Obrazac uporabe UC4 (BrišiSaOglasnePloče)

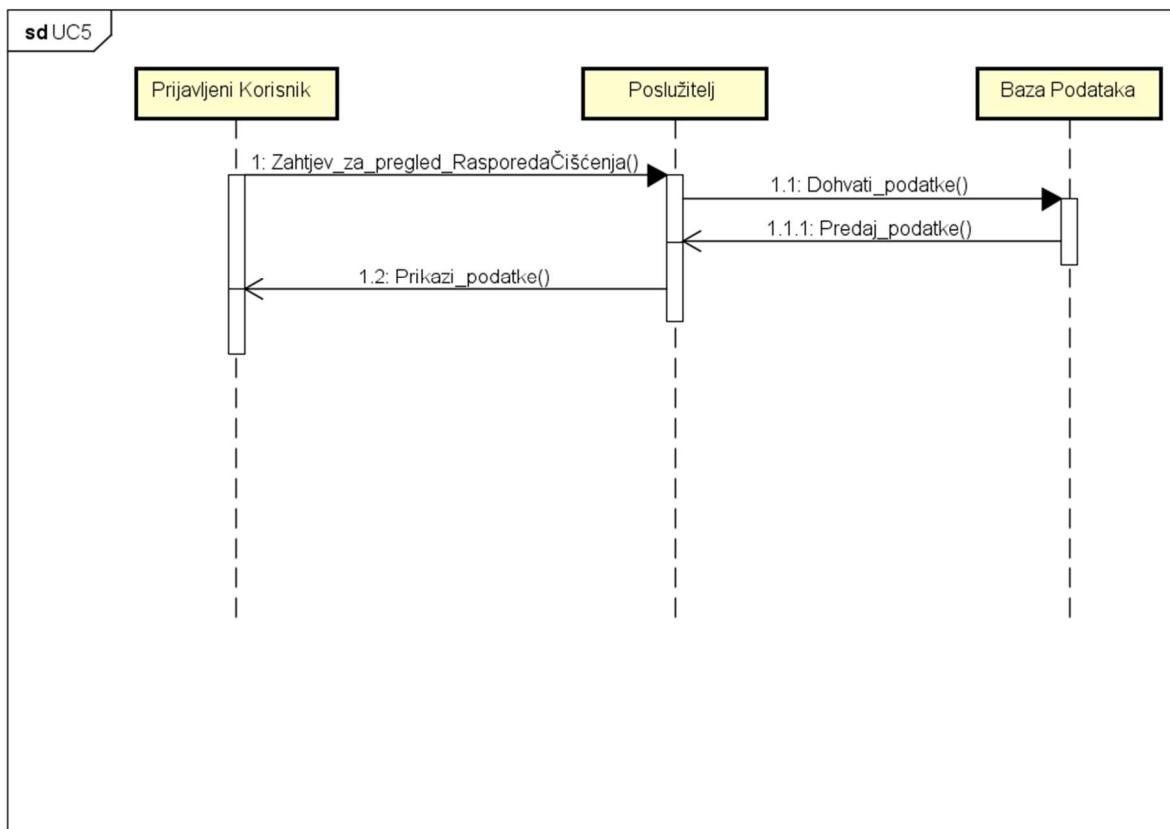
Korisnik odabire objavu koju želi obrisati. Poslužitelj prosljeđuje taj zahtjev bazi podataka koja obavlja brisanje i javlja poslužitelju da je objava obrisana. Na kraju poslužitelj vraća korisniku poruku o uspješno obavljenom brisanju objave.



Slika 5 Sekvencijski dijagram obrasca BrišiSaOglasnePloče

Obrazac uporabe UC5 (PogledajRasporedČišćenjaSnijeg)

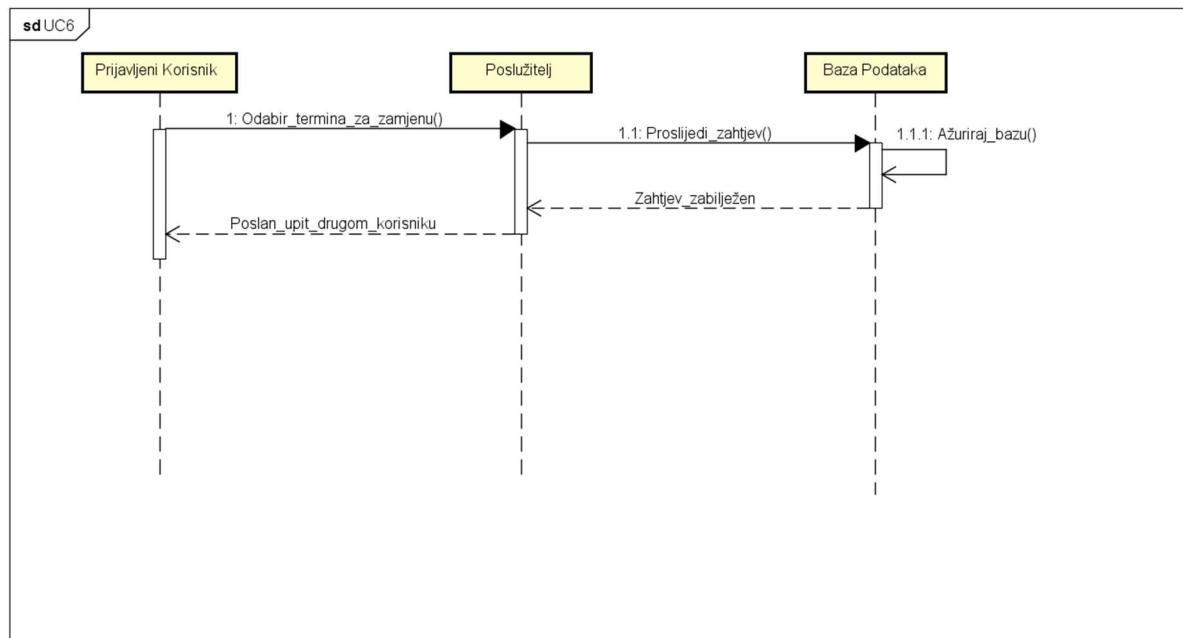
Prijavljeni korisnik zahtjeva pregled rasporeda čišćenja snijega. Poslužitelj dohvaća sve podatke iz baze podataka te ih daje na uvid korisniku.



Slika 6 Sekvencijski dijagram obrasca PregledajRasporedČišćenjaSnijeg

Obrazac uporabe UC6 (ZatražioZamjenuTerminaRasporedaČišćenjaSnijega)

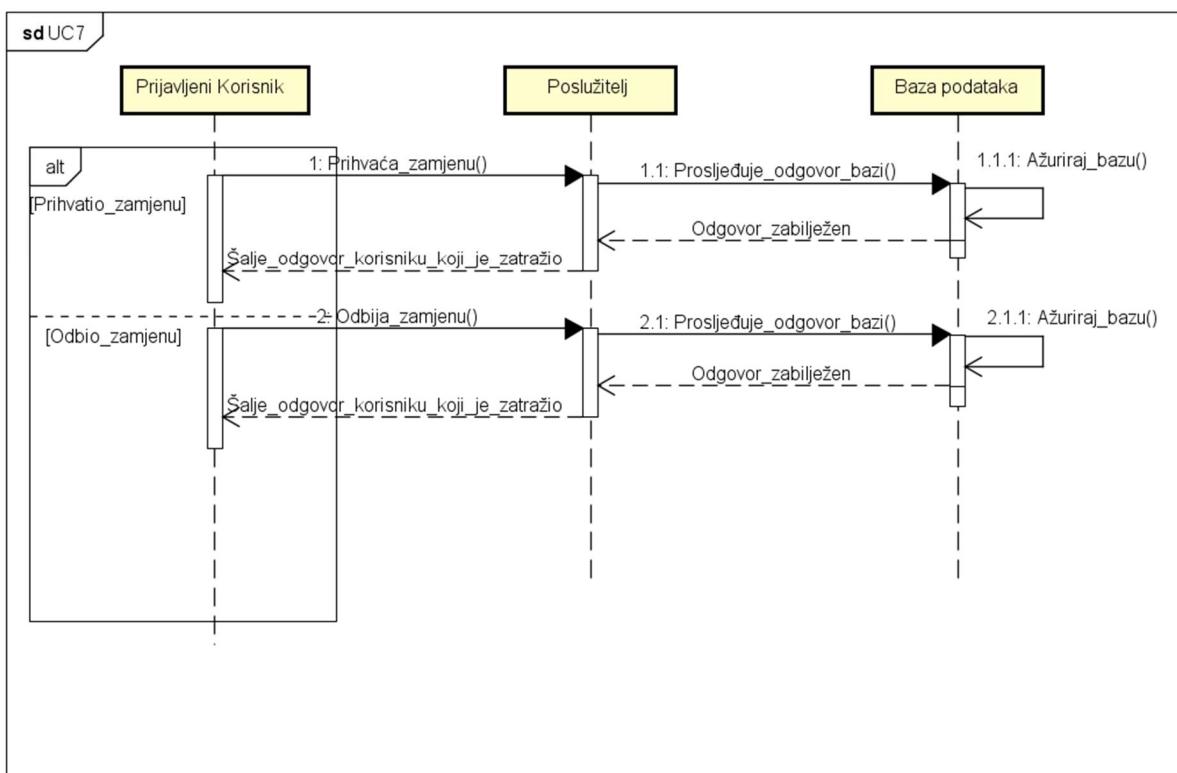
Korisnik odabire termin koji mu odgovara za zamjenu, te poslužitelj to proslijeđuje bazi koja to bilježi. Nakon toga poslužitelj obavještava korisnika čiji je datum za zamjenu zatražen.



Slika 7 Sekvencijski dijagram obrasca ZatražioZamjenuTerminaRasporedaČišćenjaSnijega

Obrazac uporabe UC7 (OdgovorNaZamjenuTerminaRasporedaČišćenjaSnijega)

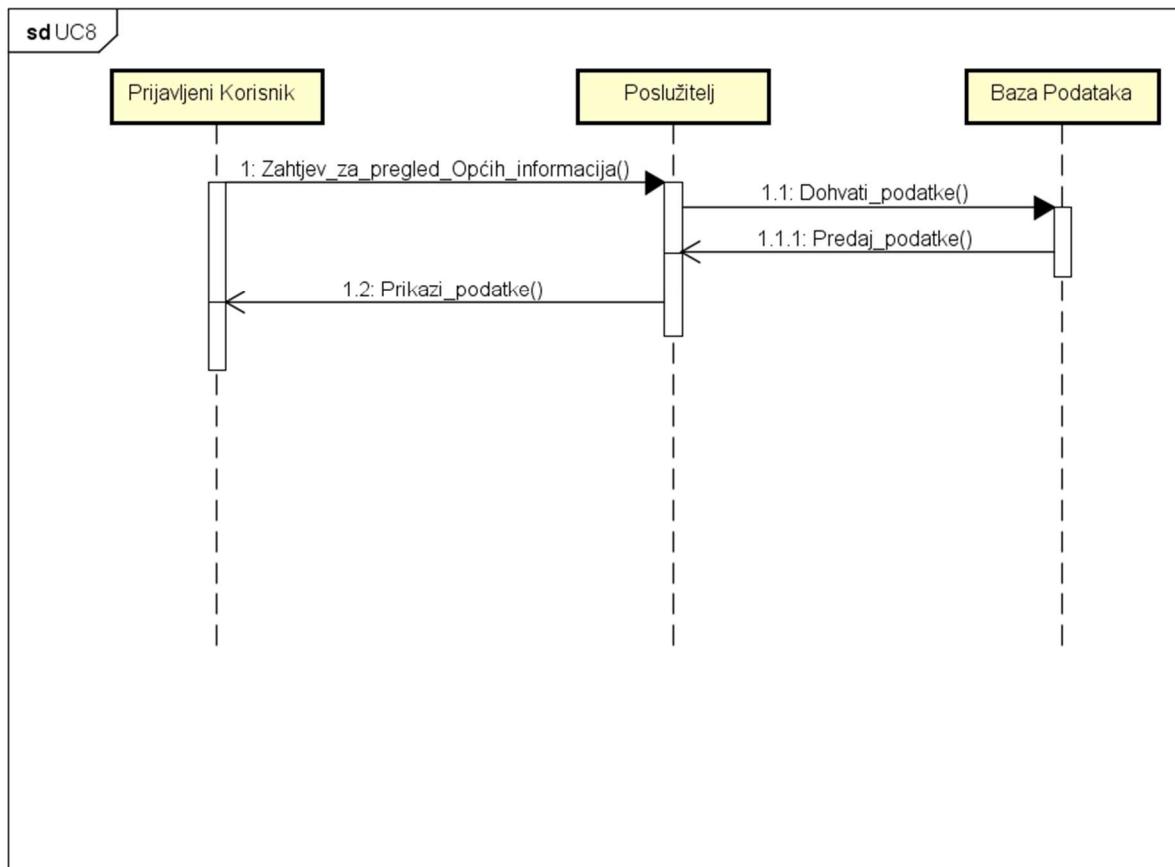
U ovom slučaju prijavljeni korisnik ima dvije opcije. Ako odabere prihvatanje zamjene, poslužitelj to prosljeđuje bazi koja to ažurira u svojim podatcima, te nakon toga poslužitelj šalje obavijest korisniku da je poslana potvrđena poruka korisniku koji je zatražio zamjenu. U drugom slučaju, ako je korisnik odbio zamjenu, događa se isti scenarij osim što je poruka posljana korisniku koji je zatražio zamjenu u ovom slučaju negativna.



Slika 8 Sekvencijski dijagram obrasca OdgovorNaZamjenuTerminaRasporedaČišćenjaSnijega

Obrazac uporabe UC8 (PregledajOpćeInformacije)

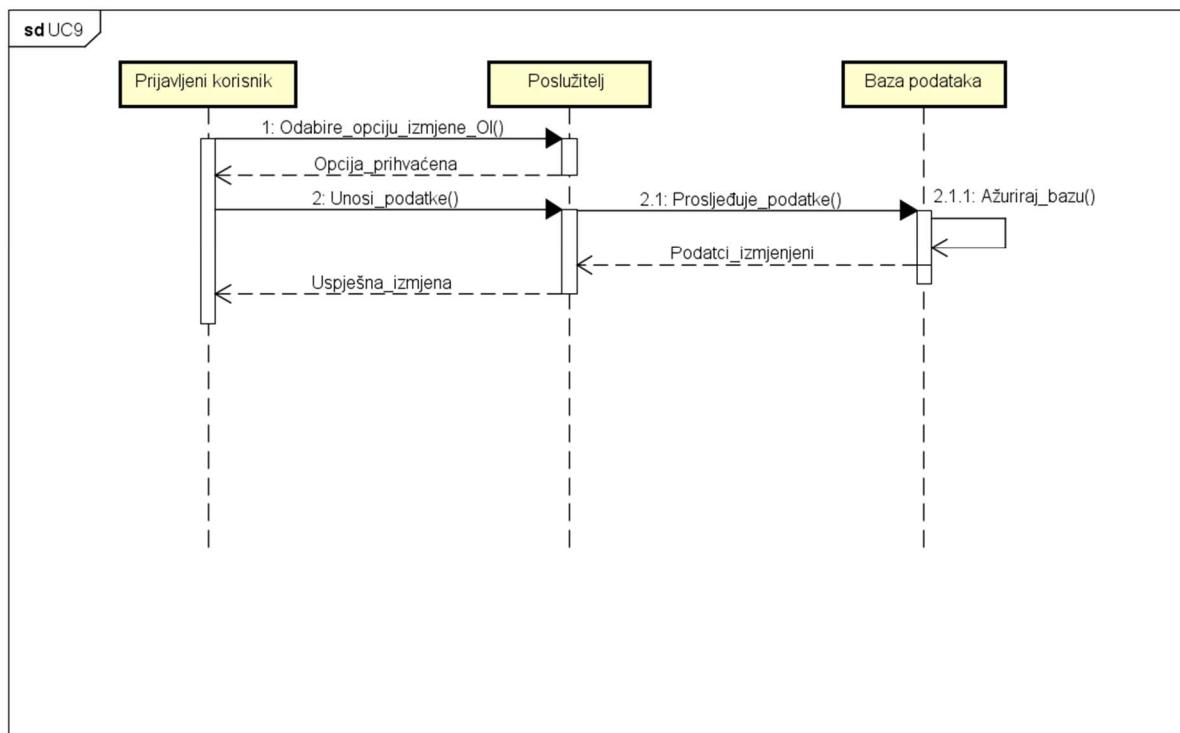
Prijavljeni korisnik zahtjeva pregled općih informacija. Poslužitelj dohvaća sve podatke iz baze podataka te ih daje na uvid korisniku.



Slika 9 Sekvencijski dijagram obrasca PregledajOpćeInformacije

Obrazac uporabe UC9 (IzmjeniOpćeInformacije)

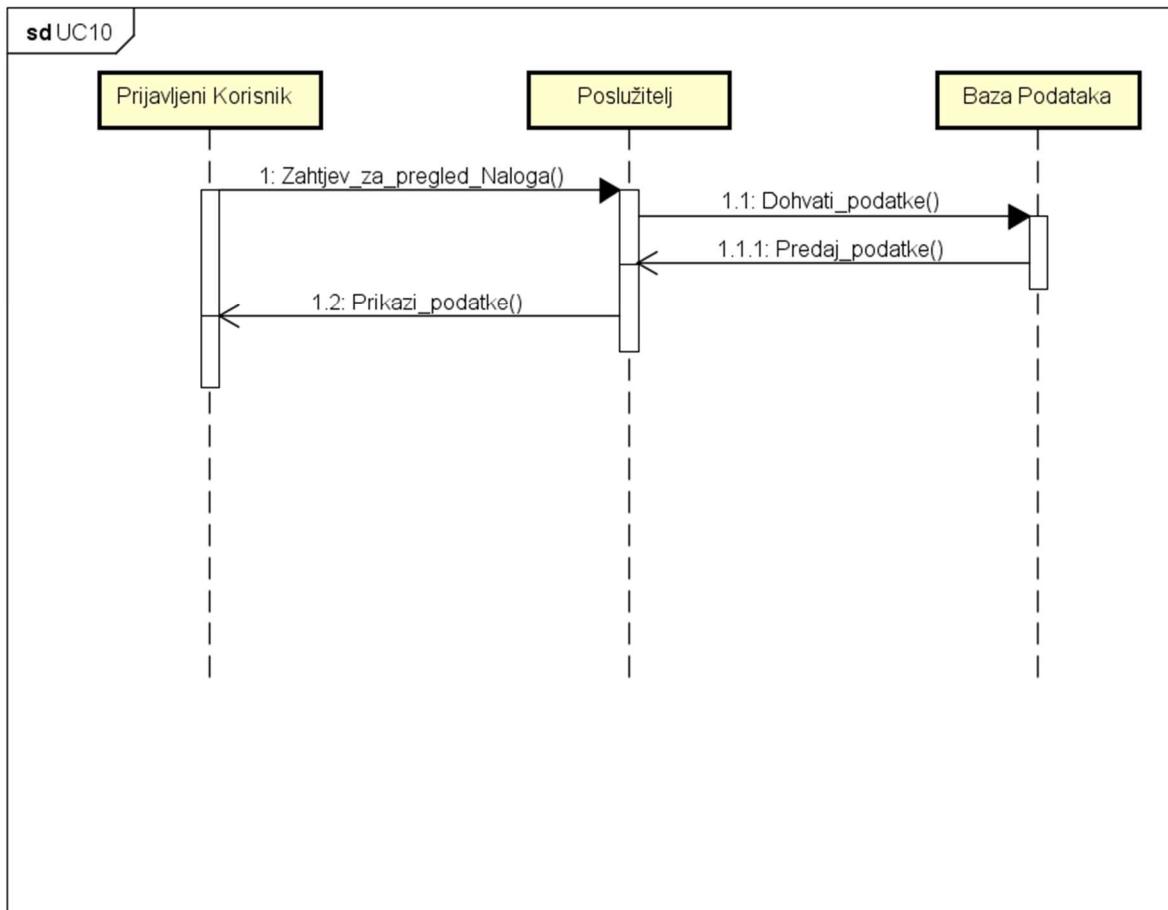
Prijavljeni korisnik odabire opciju izmjenu podataka u općim informacijama. Poslužitelj mu odobrava radnju, te korisnik unosi nove podatke. Poslužitelj te podatke prosljeđuje bazi koja ažurira iste, te potvrđuje izmjenu poslužitelju koji obavještava korisnika da je uspješna izmjena.



Slika 10 Sekvencijski dijagram obrasca IzmjeniOpćeInforamcije

Obrazac uporabe UC10 (PogledajNaloge)

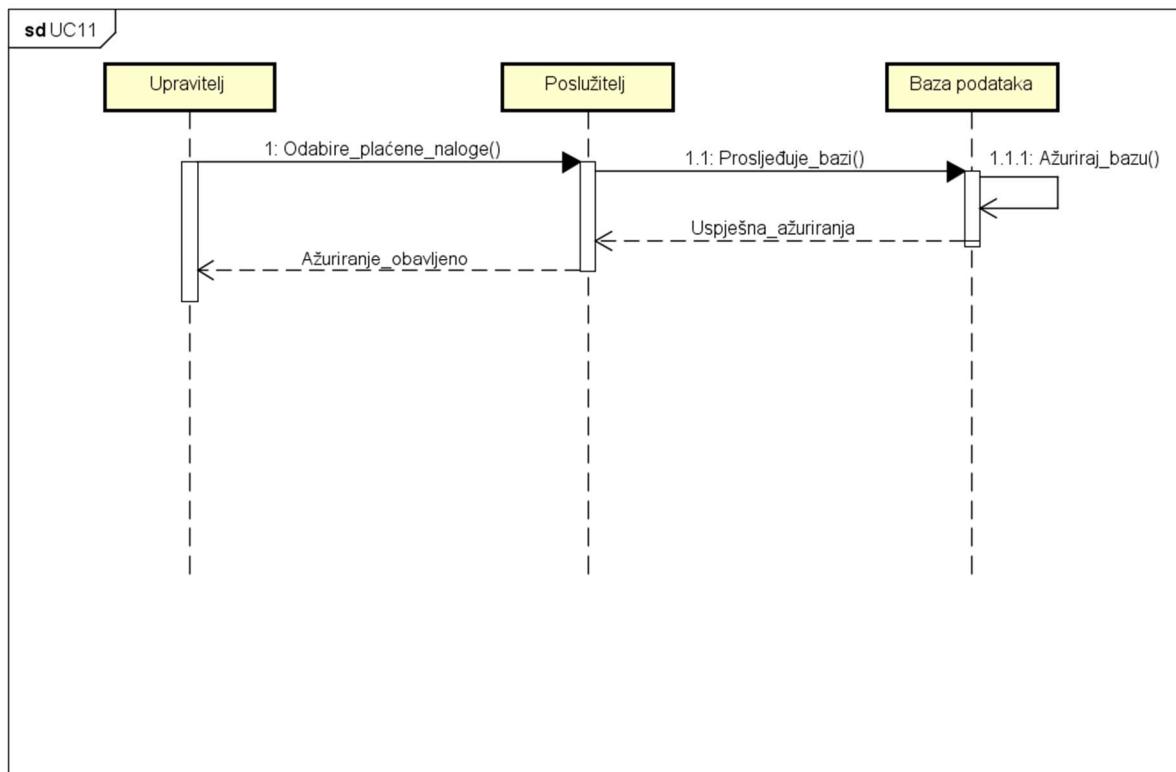
Prijavljeni korisnik zahtjeva pregled naloga. Poslužitelj dohvaća sve podatke iz baze podataka te ih daje na uvid korisniku.



Slika 11 Sekvencijski dijagram obrasca PogledajNaloge

Obrazac uporabe UC11 (AžurirajNaloge)

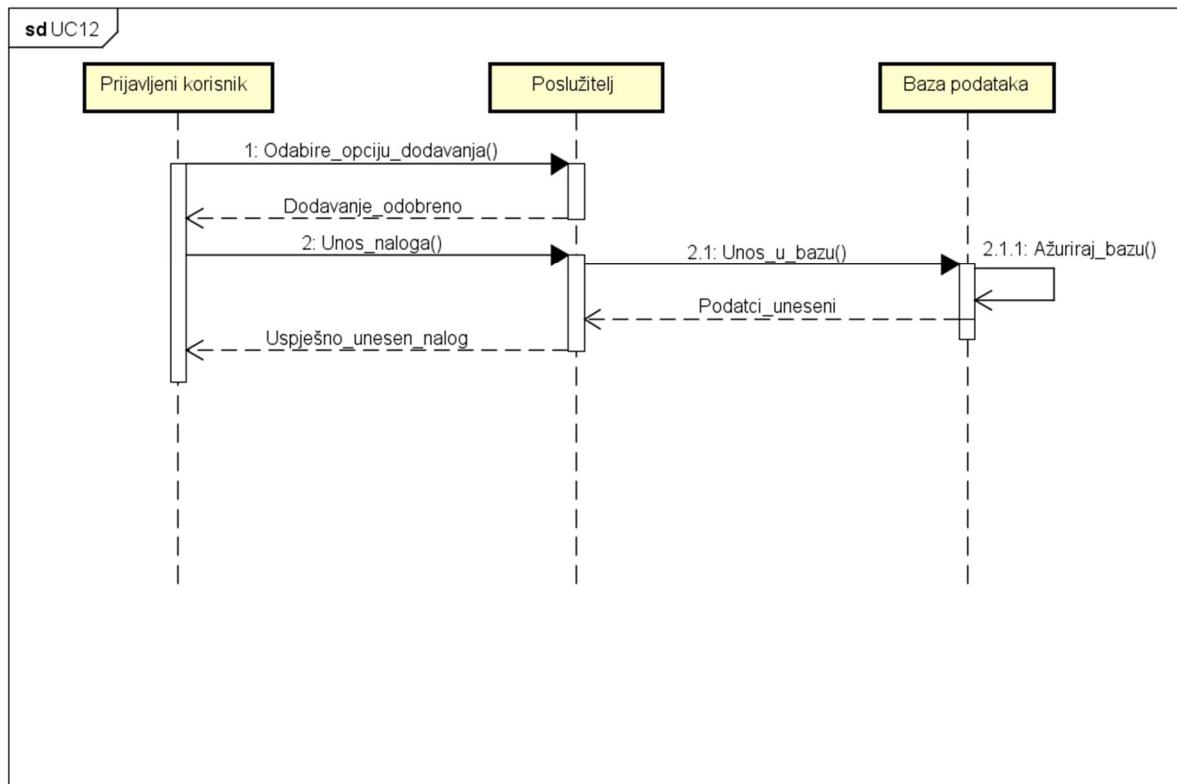
Upravitelj odabire naloge i pričuve koji su plaćeni, te ih poslužitelj prosljeđuje bazi koja ažurira označene naloge u svojoj bazi, te šalje povratnu poruku o uspješnosti, koju onda poslužitelj prosljeđuje upravitelju.



Slika 12 Sekvencijski dijagram obrasca AžurirajNaloge

Obrazac uporabe UC12 (DodajNaloge)

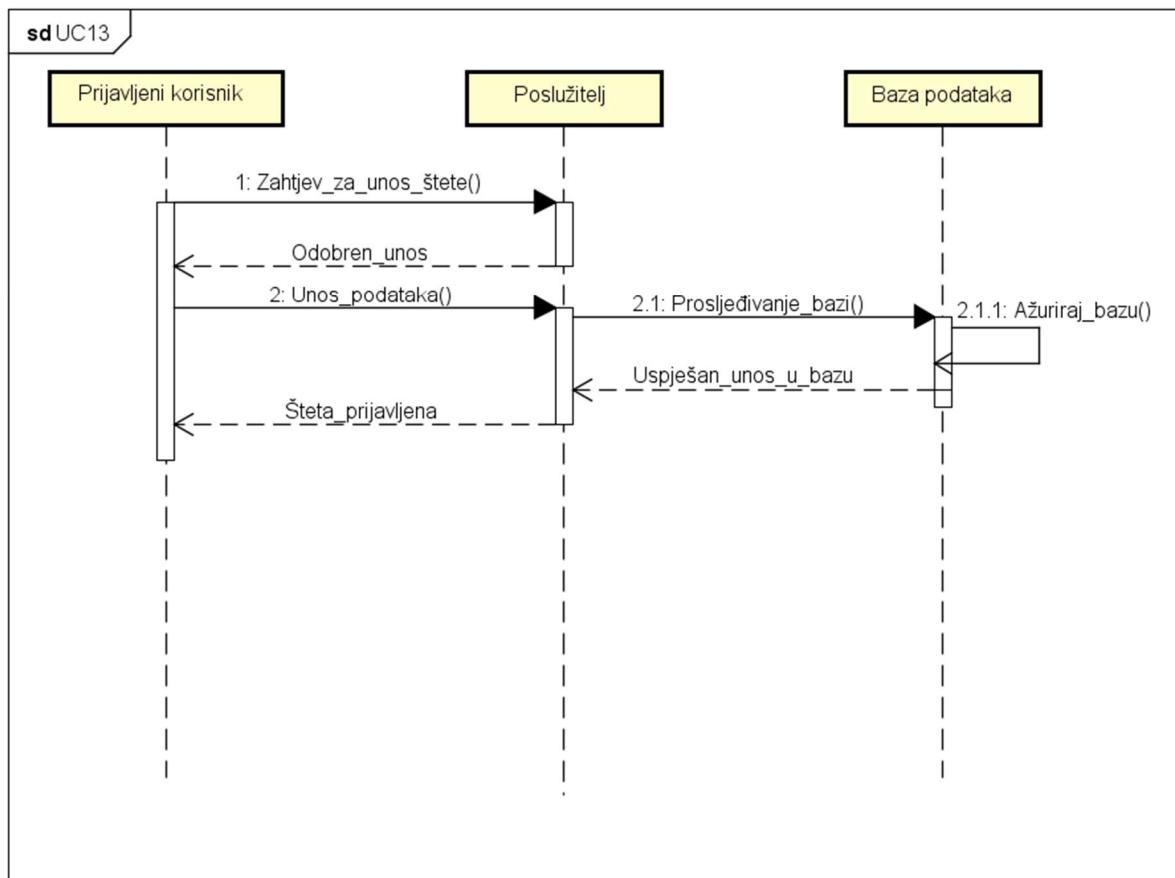
Korisnik, u ovom slučaju, upravitelj ili predstavnik, odabiru opciju dodavanja novih naloga, koje im poslužitelj odobrava, nakon čega oni unose sve potrebne podatke. Te podatke poslužitelj prosljeđuje bazi koja ih pohranjuje i vraća poruku o uspješnosti pohrane krajnjem korisniku.



Slika 13 Sekvencijski dijagram obrasca DodajNaloge

Obrazac uporabe UC13 (PrijaviŠtetu)

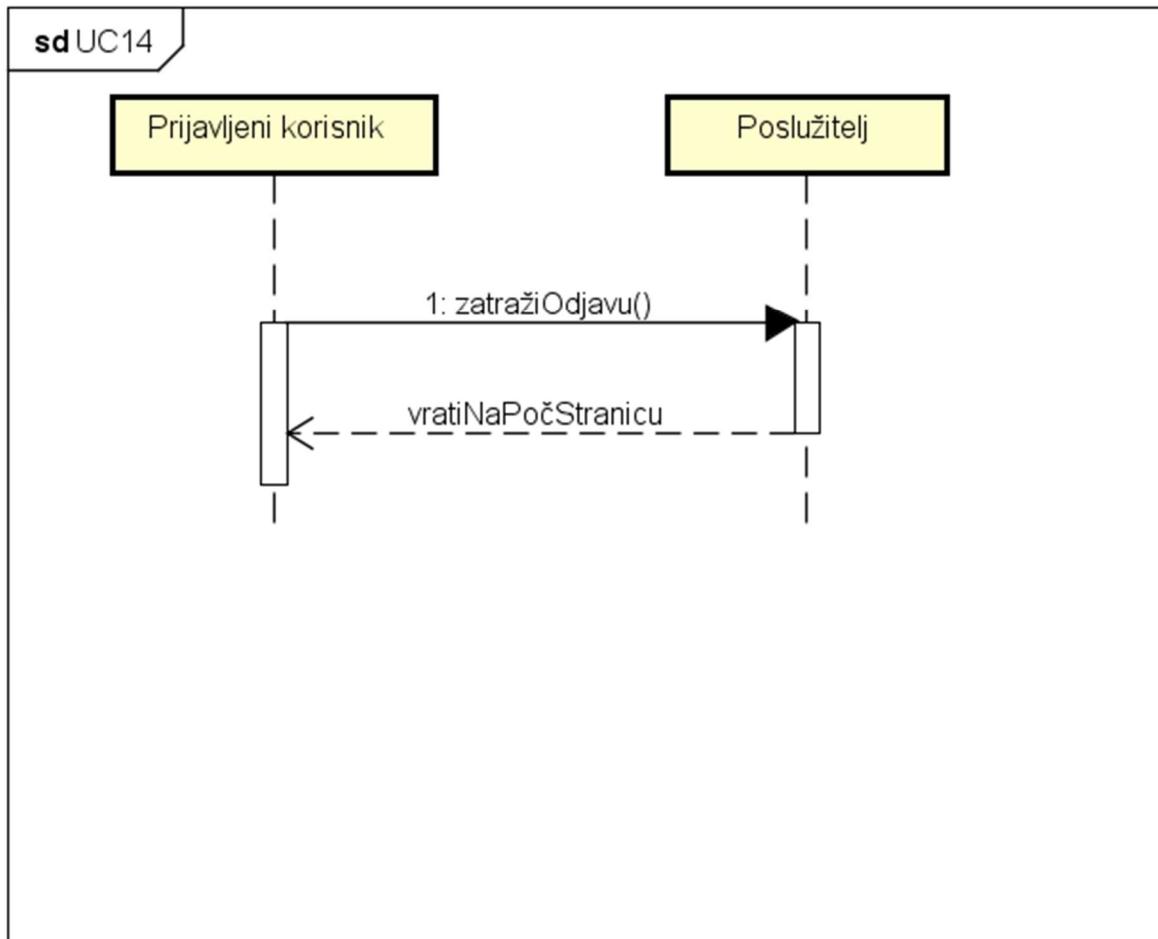
Prijavljeni korisnik zatražuje unos štete, što mu poslužitelj odobrava. Nakon toga korisnik unosi podatke koje poslužitelj prosljeđuje bazi koja ih pohranjuje i dojavljuje porukom uspješan unos, koji poslužitelj nakon toga prosljeđuje korisniku.



Slika 14 Sekvencijski dijagram obrasca PrijaviŠtetu

Obrazac uporabe UC14 (OdjavljivanjeKorisnika)

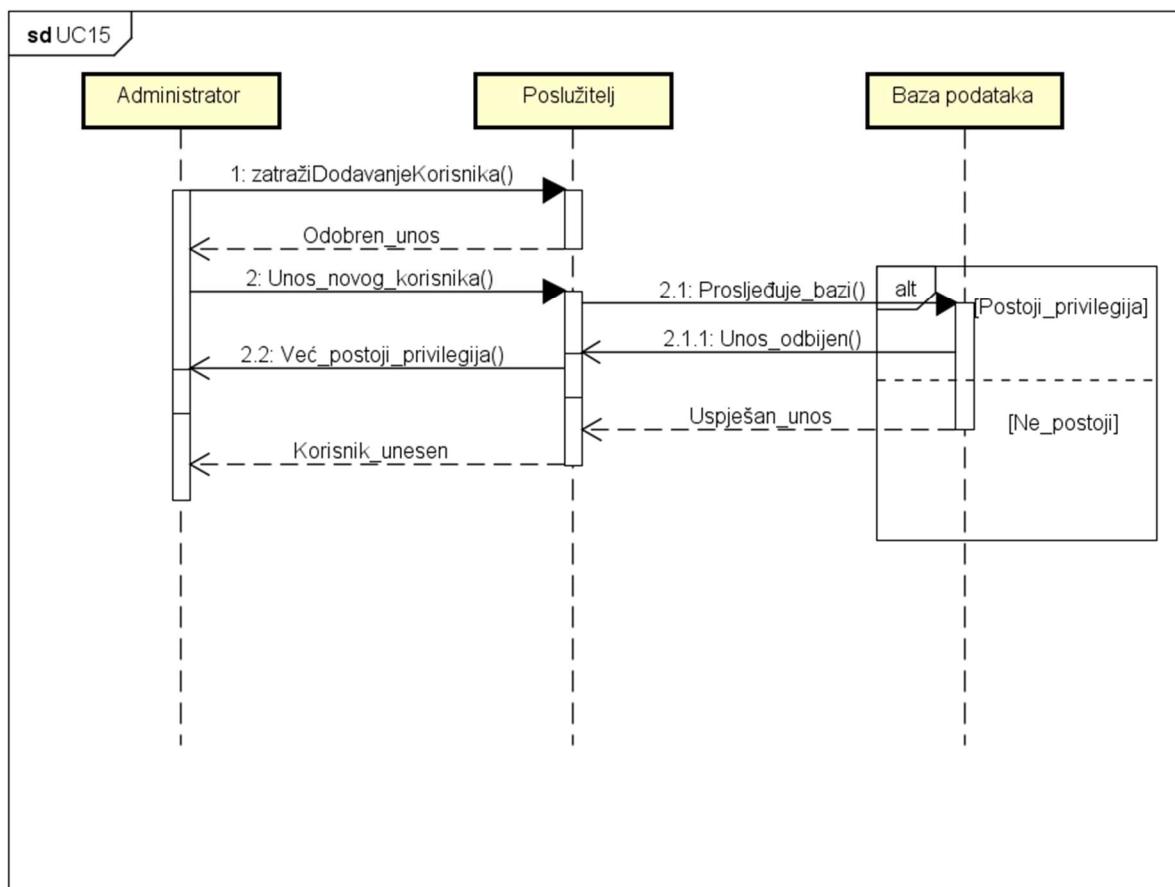
Prijavljeni korisnik zatražuje odjavu od poslužitelja, koji ga na zahtjev odjavljuje i vraća na početnu stranicu.



Slika 15 Sekvencijski dijagram obrasca OdjavljivanjeKorisnika

Obrazac uporabe UC15 (DodavanjeKorisnika)

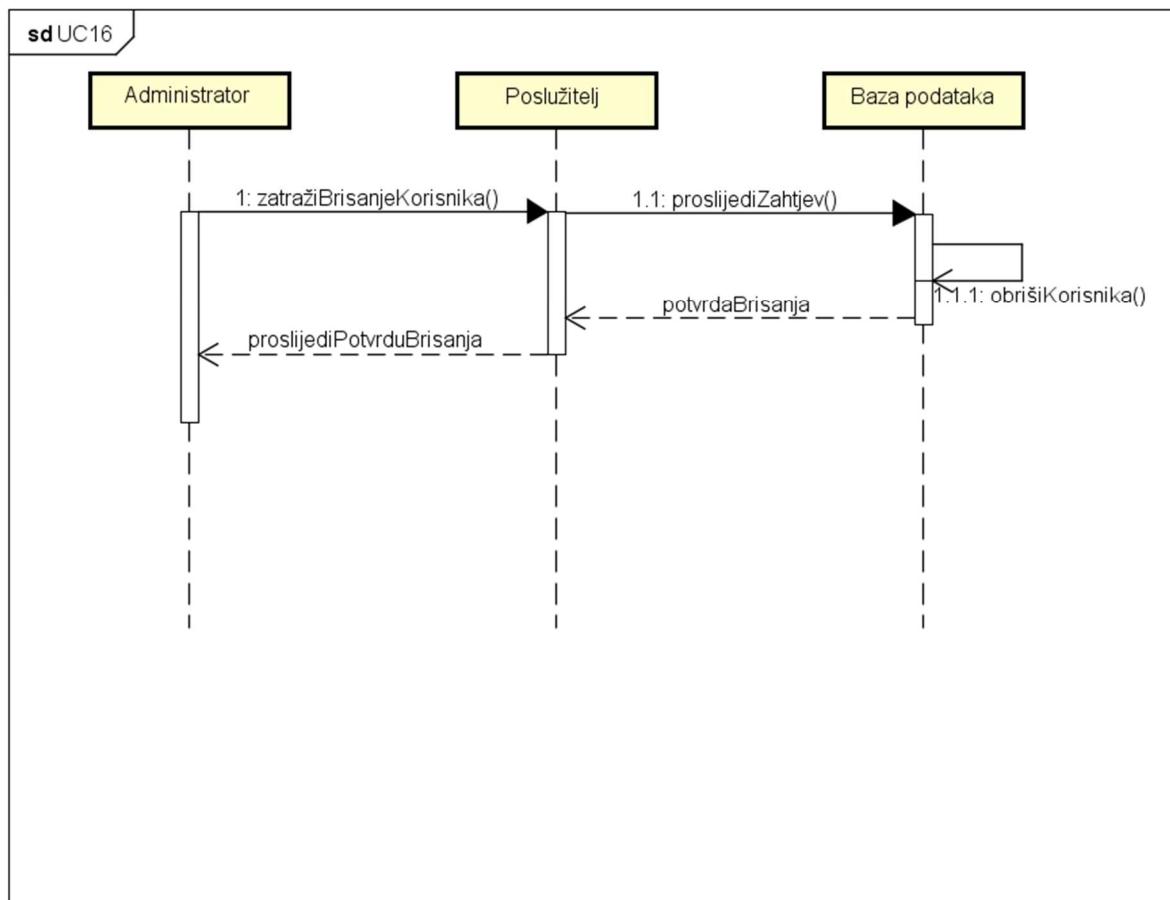
Administrator zatraži dodavanje korisnika koje mu poslužitelj odobrava. Zatim administrator unosi podatke koji se prosljeđuju bazi, te ako je unijet korisnik sa privilenjima koje već postoje onda mu se unos odbija sa odgovarajućom porukom. U suprotnom, baza pohranjuje unos korisnika, te unos potvrđuje porukom koja u konačnici dolazi do administratora.



Slika 16 Sekvencijski dijagram obrasca DodavanjeKorisnika

Obrazac uporabe UC16 (BrisanjeKorisnika)

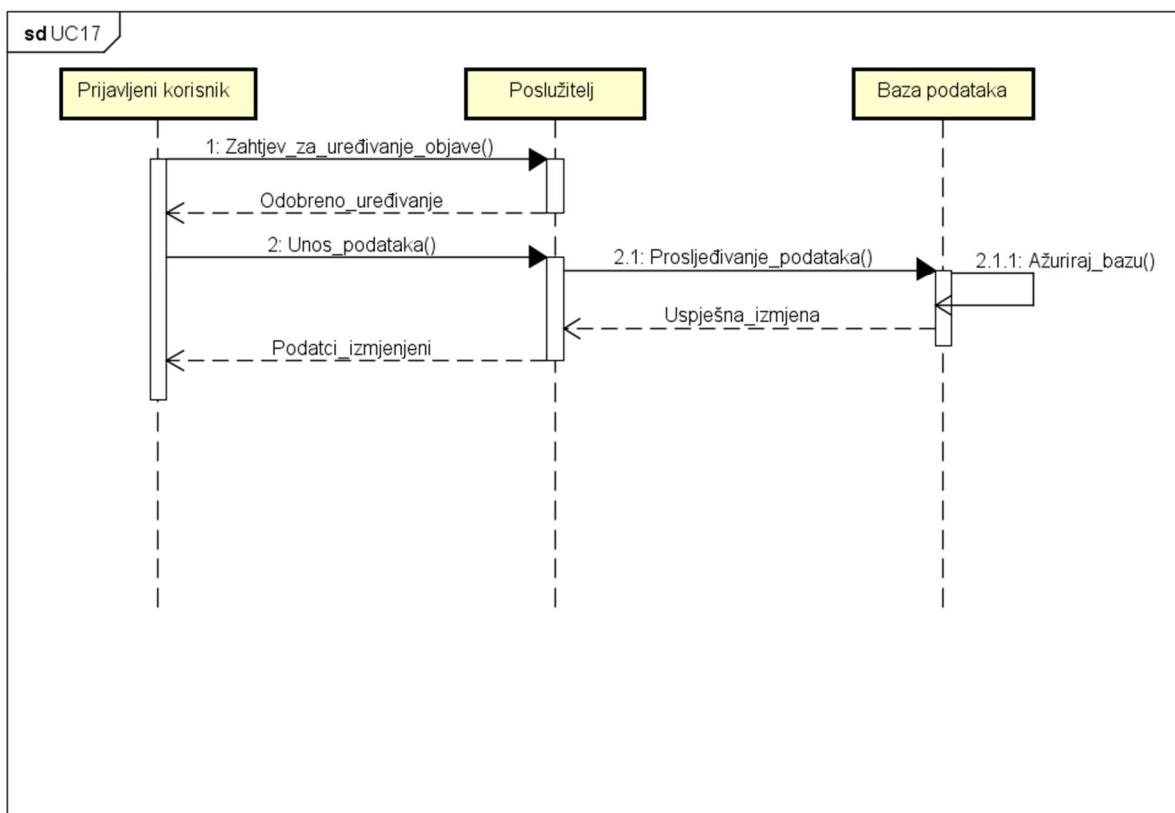
Administrator zatražuje brisanje korisnika. Poslužitelj zahtjev prosljeđuje bazi podataka, koja briše korisnika iz baze i to javlja porukom o potvrdi brisanja koju poslužitelj u konačnici prosljeđuje administratoru.



Slika 17 Sekvencijski dijagram obrasca BrisanjeKorisnika

Obrazac uporabe UC17 (UrediObjavuNaOglasnojPloči)

Prijavljeni korisnik zahtjeva uređivanje objave na oglasnoj ploči, što mu poslužitelj odobrava. Nakon toga korisnik unosi nove podatke, odnosno izmjenjuje već unesene podatke, koje onda poslužitelj proslijeđuje bazi, koja ih pohranjuje te dojavljuje uspješnu izmjenu poslužitelju koji u konačnici javlja to korisniku.



Slika 18 Sekvencijski dijagram obrasca UrediObjavuNaOglasnojPloči

5. Ostali zahtjevi

- Sustav mora podržavati paralelan rad više korisnika
- Sustav i korisničko sučelje moraju podržavati sve znakove hrvatske abecede, kao i interpunkcijske znakove
- Odgovor aplikacije ne smije trajati duže od 3 sekunde
- Otpornost na SQL injection i XSS
- Aplikacija mora bit napravljena u Javi, te se mora koristit Spring Boot

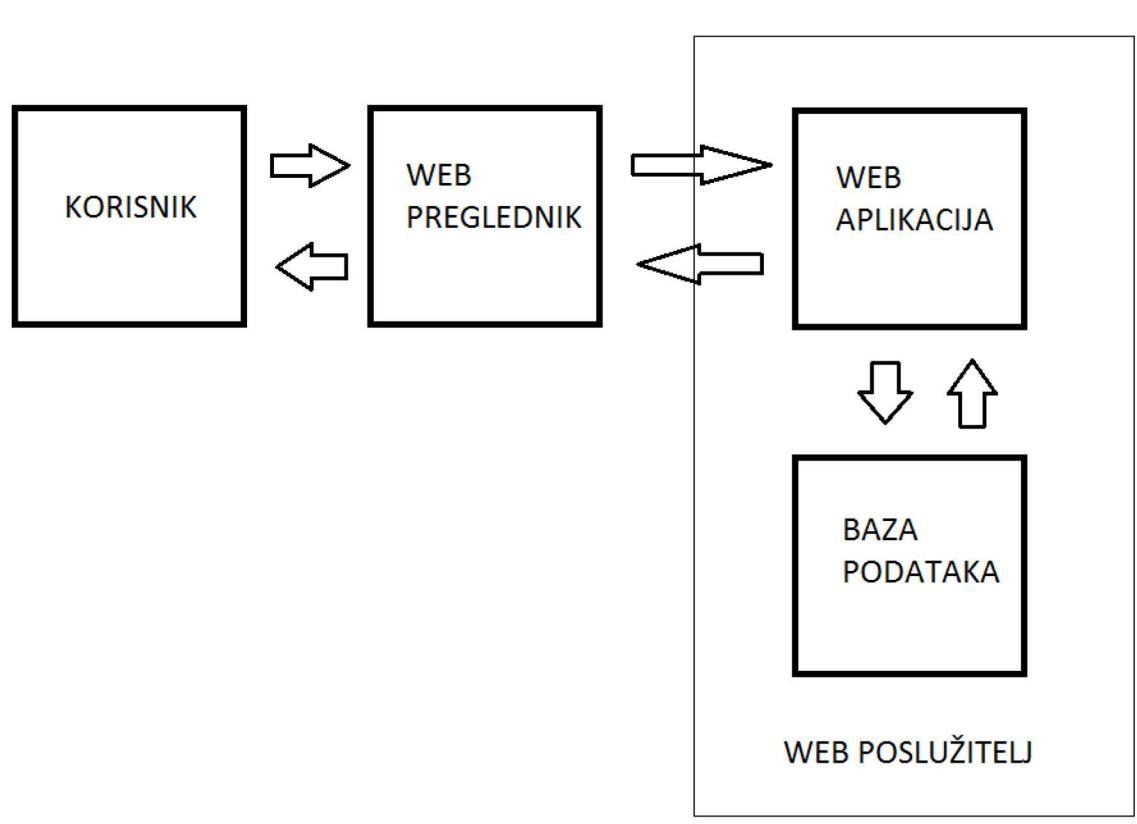
6. Arhitektura i dizajn sustava

6.1. Svrha, opći prioriteti i skica sustava

Arhitektura našeg sustava je web aplikacija iz razloga da korisnici ovog sustava mogu lakše komunicirati i lakše koristiti sam sustav.

Podsistavi arhitekture:

- Web poslužitelj
- Web aplikacija
- Baza podataka



Slika 19 Skica sustava

Web poslužitelj

Web poslužitelj obrađuje sve zahtjeve koji dolaze sa web aplikacije HTTP protokolom, koji nakon obrade šalje odgovarajući odgovor web aplikaciji, također HTTP protokolom.

Naša web aplikacija će se izvoditi *Apache Tomcat* web poslužitelju koji je opisan u pojmovniku ove dokumentacije.

Web aplikacija

Putem web aplikacije korisnik ima pristup svim funkcionalnostima sustava ovisno o pridjeljenim ovlastima. Preko web aplikacije šaljemo HTTP zahtjev na poslužitelj kojim dohvaćamo i šalje podatke bazi podataka.

Zadaća web aplikacije je isto tako da prikazuje vizualne informacije u obliku HTML dokumenta, dok će preglednik te dokumente prikazivati u obliku web stranica.

Baza podataka

Baza podataka sadrži pohranu svih podataka o korisnicima i svim potrebnim podatcima za određene funkcionalnosti. Zadaća same baze je da brzo i po potrebi dohvaća podatke koje je korisnik zatražio, osvježava promjene koje su unešene od strane korisnika i sprječava mogući unos podataka koji se već nalaze u samoj bazi.

Entiteti baze podataka:

Zgrada – Ovaj entitet sadrži sve bitne informacije o zgradama. Atributi u ovom entitetu su: ID, Adresa, Fond, IDPredstavnika, IDUpravitelja. Entitet ima One-to-Many vezu na entitet **Stan** preko atributa ID prema atributu IDZgrade i dvije Many-to-One veze preko atributa IDPredstavnika i IDUpravitelj prema atributu ID u entitetu **Korisnik**.

Stan – Ovaj entitet sadrži sve podatke o pojedinom stanu. Atributi ovog entiteta su: ID, Površina, IDZgrade, IDVlasnika. Entitet ima vezu Many-to-One preko atributa IDZgrade na atribut ID u entitetu **Zgrada** i Many-to-One vezu prema entitetu **Korisnik** preko atributa IDVlasnika, prema atributu ID.

Korisnik – Entitet koji sadrži sve podatke o pojedinom korisniku, kao što su atributi: ID, Ime, Prezime, Mail, Lozinka, Privilegija, Podsjetnik. Entitet ima dvije vezu One-to-Many preko atributa ID prema atributima IDPrimatelja i IDPlatitelja u entitetu **Nalog**, jednu One-to-Many vezu preko atributa ID prema atributu IDVlasnika u entitetu **Stan**, dvije One-to-Many veze preko atributa ID prema atributima IDPredstavnika i IDUpravitelja u entitetu **Zgrada**, jednu Many-to-One vezu preko atributa ID prema atributu IDOsobe u entitetu **TerminČišćenjaSnijega**, jednu One-to-Many vezu preko atributa ID prema atributu IDKorisnika u entitetu **Notifikacija**, jednu One-to-Many vezu preko atributa ID prema atributu IDKreatora u entitetu **Odjava**, te One-to-Many vezu preko atributa ID prema atributu IDKreatora u entitetu **Trošak**.

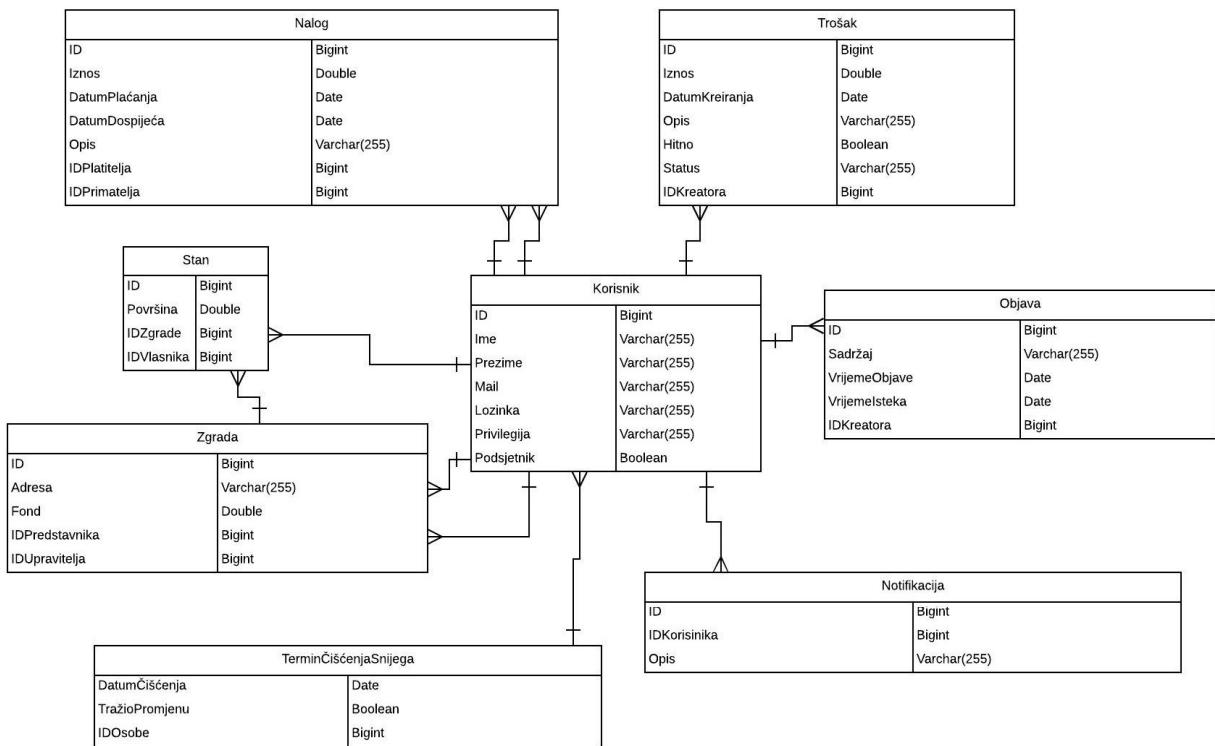
Trošak – Entitet koji sadrži sve podatke o troškovima zgrade sa atributima: ID, Iznos, DatumKreiranja, Opis, Hitno, Status, IDKreatora. Entitet sadrži vezu Many-to-One preko atributa IDKreatora prema atributu ID u entitetu **Korisnik**.

Nalog – Entitet koji sadrži podatke potrebne o nalozima zgrade i attribute kao što su: ID, Iznos, DatumPlaćanja, DatumDospijeća, Opis, IDPlatitelja i IDPrimatelja. Entitet sadrži dvije Many-to-One veze preko atributa IDPlatitelja i IDPrimatelja prema atributu ID u entitetu **Korisnik**.

TerminČišćenjaSnijega – Entitet koji pregled termina čišćenja za određenog korisnika i mogućnost da isti zatraži promjenu termina. Entitet sadrži attribute: DatumČišćenja, TražioPromjenu, IDOsobe. Sadrži jednu One-to-Many vezu preko atributa IDOsobe prema atributu ID u entitetu **Korisnik**.

Notifikacija – Entitet koji prikazuje obavijest i korisnika kome je obavijest namjenjena. Sadrži attribute: ID, IDKorisnika, Opis. Entitet sadrži jednu Many-to-One vezu preko IDKorisnika prema atributu ID u entitetu **Korisnik**.

Odjava – Entitet koji sadrži objave, njihovo vrijeme objave, vrijeme isteka i sadržaj objave, koje je stvorio određeni korisnik. Sadrži attribute: ID, Sadržaj, VrijemeObjave, Vrijemelsteka i IDKreatora. Entitet sadrži jednu Many-to-One vezu preko atributa IDKreatora prema atributu ID u entitetu **Korisnik**.



Slika 20 ER model baze podataka

6.2. Dijagram razreda s opisom

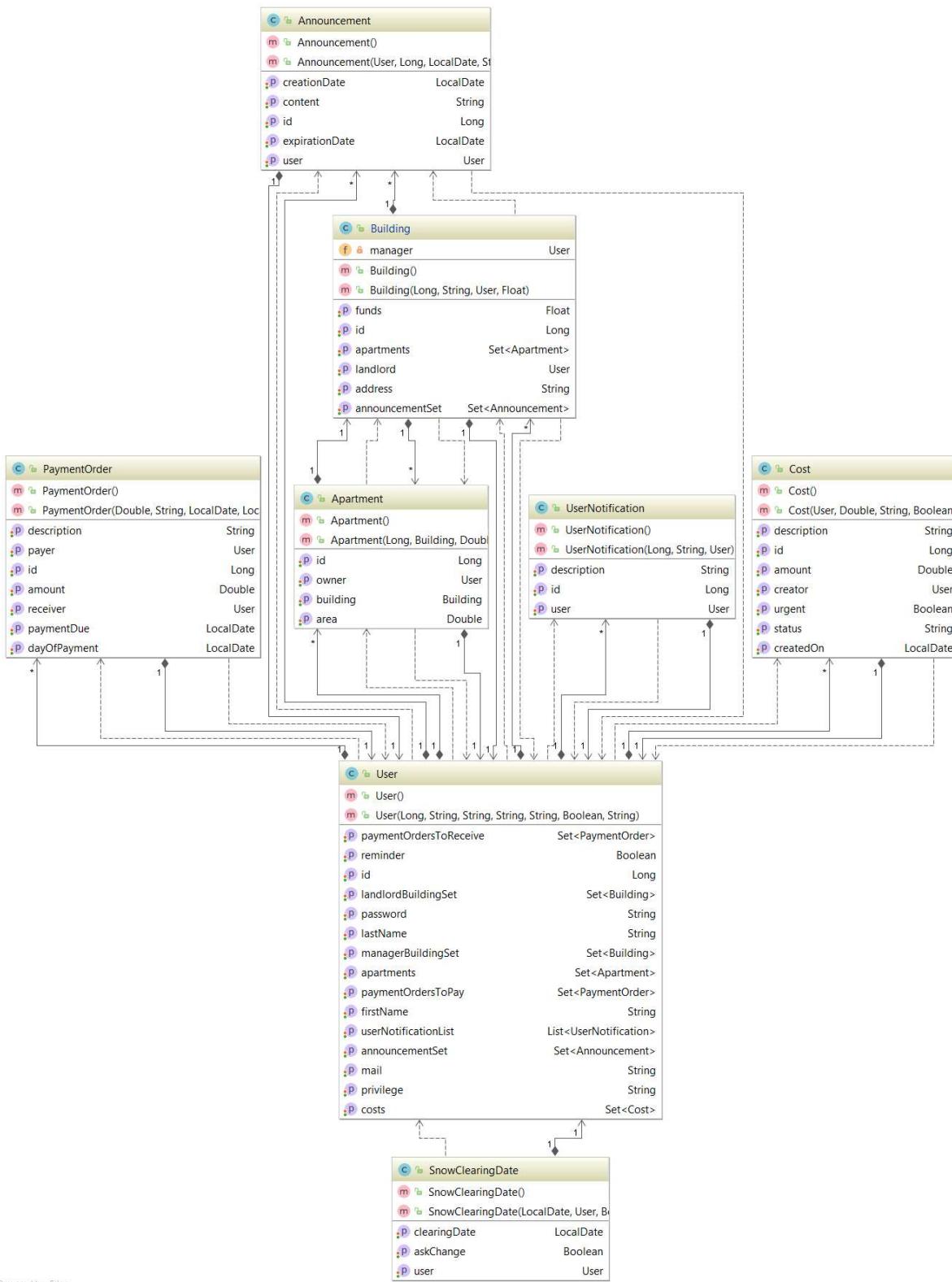
U ovom odlomku opisani su svi razredi te priložen pripadajući dijagrami. Također su navedene sve pripadajuće metode te vrste pristupa tim metodama. Iz obrazaca uporabe vidimo koje operacije trebamo implementirati za svaki razred. Razredi su podijeljeni u tri paketa: *domain*, *service* i *controller*. U paketu *domain* nalaze se svi potrebni modeli podataka koji se koriste u izradi aplikacije, u paketu *service* nalaze razredi zaduženi za komunikaciju s bazom podataka, a u paketu *controller* nalaze se svi razredi koji upravljaju korisničkim zahtjevima.

Razredi koji su u paketu *domain*:

- Obavijest korisniku
 - Razred koji oblikuje obavijesti koje se prikazuju korisniku u aplikaciji, njegovi atributi su: **identifikator**, **tekst obavijesti** i objekt **Korisnik** kojem se prikazuje obavijest
 - Postoje javne metode za postavljanje i dohvaćanje svih svojih atributa
- Korisnik
 - Razred koji predstavlja korisnika web-aplikacije, sadrži atrbute: **identifikator**, **ime**, **prezime**, **adresu elektroničke pošte**, **privilegiju**, zastavicu **Podsjetnik** koja označava želi li korisnik primati obavijest ako nisu izvršili uplatu na vrijeme, **skup obavijesti**, **listu notifikacija**, **skup troškova**, **skup zgrada** u kojima ima stanove i **skup stanova** kojih je vlasnik
 - Sadrži javne metode za postavljanje i dohvaćanje svih svojih atributa
- Termin čišćenja snijega
 - Razred koji modelira termin čišćenja snijega sa atributima: **datum čišćenja snijega**, objektom **Korisnik** koji je zadužen za čišćenje i zastavicom **zatražiZamjenu** koja je aktivna ako je korisnik zatražio zamjenu
 - Metode su javne i omogućavaju dohvaćanje i postavljanje svih svojih atributa
- Nalog
 - Razred koji predstavlja naloge za plaćanje, sadrži atrbute: **identifikator** koji jednoznačno određuje korisnika, **iznos** uplate, **opis** sadrži tekst kojim se pobliže objašnjava svrha plaćanja, **dospijeće** kao datum do kada je potrebno platiti nalog, **datumPlaćanja** označava datum kada je korisnik izvršio uplatu, **platitelj** predstavlja korisnika(dužnika) zaduženog za plaćanje, **primatelj** je korisnik kojem se sredstva uplaćuju
 - Metode su javne i omogućavaju dohvaćanje i postavljanje svih svojih atributa
- Trošak
 - Razred koji modelira troškove zgrade, razred sadrži atrbute: **identifikator** troška, objekt **Korisnik** koji je stvorio trošak, **cijenu** troška, **opis** troška, **datumKreiranja**, zastavicu **hitnost** koja je aktivna ako je trošak hitan, **status** pobliže pojašnjava u kojoj je fazi realizacije taj trošak
 - Sadrži javne metode za postavljanje i dohvaćanje svih svojih atributa
- Zgrada
 - Razred koji oblikuje zgradu sa atributima: **identifikator** koji jednoznačno određuje zgradu, **adresa** zgrade, **predstavnikStanara** kao objekt razreda

Korisnik koji predstavlja sve stanare jedne zgrade , **sredstva** označavaju koliko je trenutno raspoložovo stanje na računu zgrade, **skup obavijesti** koje su trenutno na oglasnoj ploči, **skup stanova** koji pripadaju toj zgradi

- Sadrži javne metode za postavljanje i dohvaćanje svih svojih atributa
- Stan
 - Razred koji predstavlja stan u zgradu sa sljedećim atributima: **identifikator** koji jednoznačno određuje stan, **vlasnik** stana koji je objekt razreda Korisnik, **zgrada** objekt razreda Zgrada označava kojoj zgradi pripada, **površina** broj koji predstavlja površinu stana u metrima kvadratnim
 - Sadrži javne metode za postavljanje i dohvaćanje svih svojih atributa
- Obavijest
 - Razred koji modelira obavijesti koje se nalaze na oglasnoj ploči sa atributima: **identifikator** koji jednoznačno određuje obavijest, **datumKreiranja** je datum kada je korisnik kreirao obavijest, **datumIsteka** je datum kada obavijest ističe te se ona miče sa oglasne ploče, **sadržaj** ima tekst obavijesti i objekt **Korisnik** koji je stvorio obavijest
 - Sadrži javne metode za postavljanje i dohvaćanje svih svojih atributa

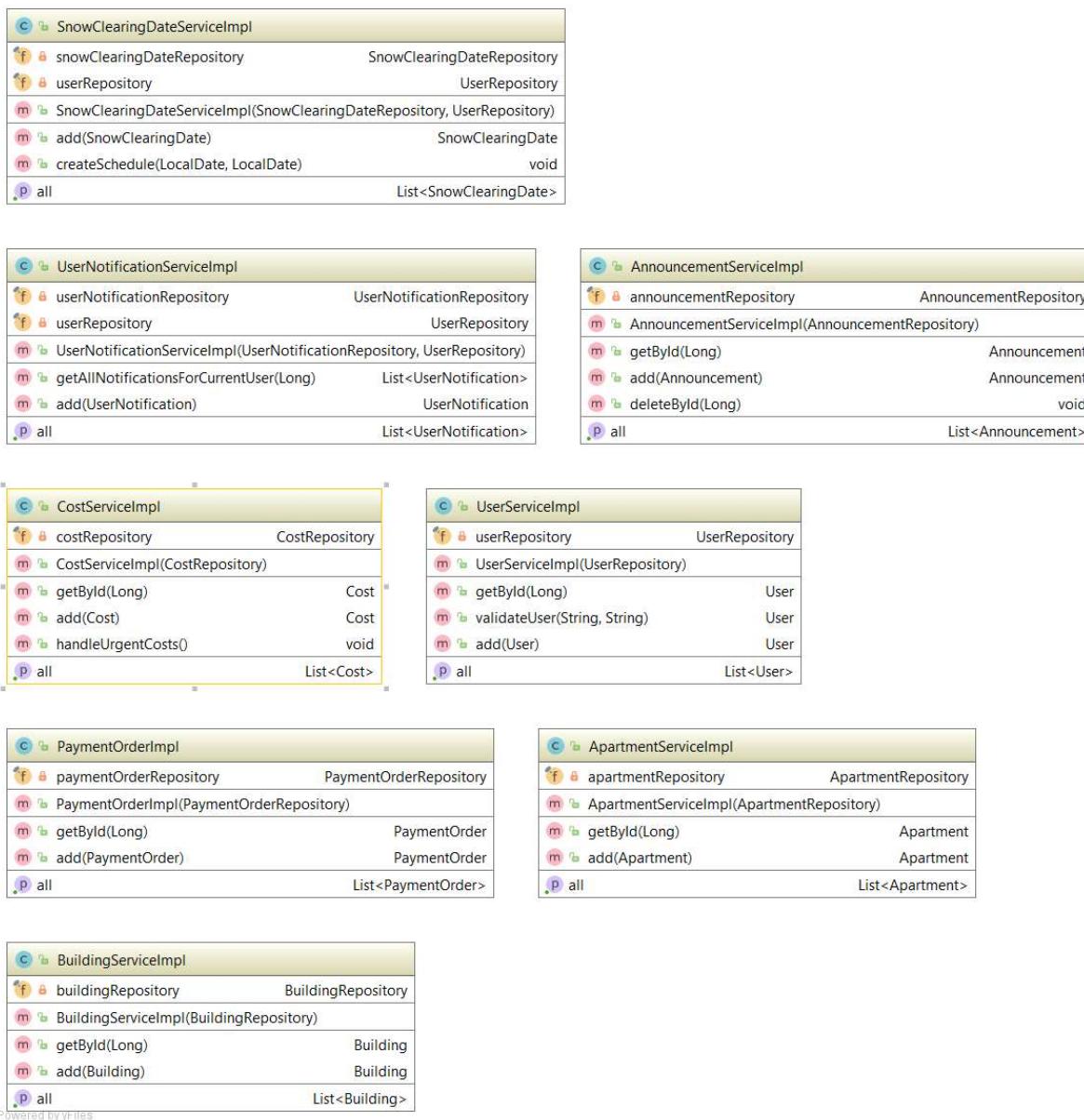


Slika 21 Dijagram razreda paketa domain

Razredi koji su u paketu *service*:

- Servis obavijesti korisniku
 - Razred je zadužen za komunikaciju s bazom podataka i upravljanje obavijestima korisnika koje su prisutne u bazi. Njegovi atributi su repozitorij korisnika te repozitorij svih obavijesti.
 - Sadrži javne metode za dohvaćanje obavijesti iz baze, pojedinačne i sve obavijesti i sve obavijesti pojedinog stanara. Postoje i javne metode za dodavanje novih te brisanje starih obavijesti.
- Servis termina čišćenja snijega
 - Razred je zadužen za komunikaciju s bazom podataka i upravljanje terminima čišćenja koji su prisutni u bazi. Njegovi atributi su repozitorij korisnika te repozitorij svih termina čišćenja snijega.
 - Sadrži javne metode za dohvaćanje termina čišćenja iz baze, pojedinačnih i svih termina i svih termina pojedinog stanara. Postoje i javne metode za dodavanje novih i brisanje starih termina te stvaranje rasporeda čišćenja snijega za sve stanare.
- Servis korisnika
 - Razred je zadužen za komunikaciju s bazom podataka i upravljanje korisnicima koji su prisutni u bazi. Njegov atribut je repozitorij korisnika.
 - Sadrži javne metode za dohvaćanje korisnika iz baze, pojedinačnih i svih korisnika. Postoje i javne metode za dodavanje i brisanje korisnika te metoda za provjeru postoji li korisnik u bazi podataka.
- Servis naloga
 - Razred je zadužen za komunikaciju s bazom podataka i upravljanje nalozima koji su prisutni u bazi. Njegov atribut je repozitorij naloga.
 - Sadrži javne metode za dohvaćanje naloga iz baze, pojedinačnih i svih naloga. Postoje i javne metode za dodavanje i brisanje naloga.
- Servis troška
 - Razred je zadužen za komunikaciju s bazom podataka i upravljanje troškovima koji su prisutni u bazi. Njegov atribut je repozitorij troškova.
 - Sadrži javne metode za dohvaćanje troškova iz baze, pojedinačnih i svih troškova. Postoje i javne metode za dodavanje i brisanje troškova.
- Servis zgrade
 - Razred je zadužen za komunikaciju s bazom podataka i upravljanje zgradama koje su prisutne u bazi. Njegov atribut je repozitorij zgrada.
 - Sadrži javne metode za dohvaćanje zgrada iz baze, pojedinačnih i svih zgrada. Postoje i javne metode za dodavanje i brisanje zgrada.

- Servis stana
 - Razred je zadužen za komunikaciju s bazom podataka i upravljanje stanovima koji su prisutni u bazi. Njegov atribut je repozitorij zgrada.
 - Sadrži javne metode za dohvaćanje zgrada iz baze, pojedinačnih i svih zgrada. Postoje i javne metode za dodavanje i brisanje zgrada.
- Servis obavijesti
 - Razred je zadužen za komunikaciju s bazom podataka i upravljanje obavijestima koji su prisutni u bazi. Njegov atribut je repozitorij obavijesti.
 - Sadrži javne metode za dohvaćanje obavijesti iz baze, pojedinačnih i svih obavijesti. Postoje i javne metode za dodavanje i brisanje obavijesti.

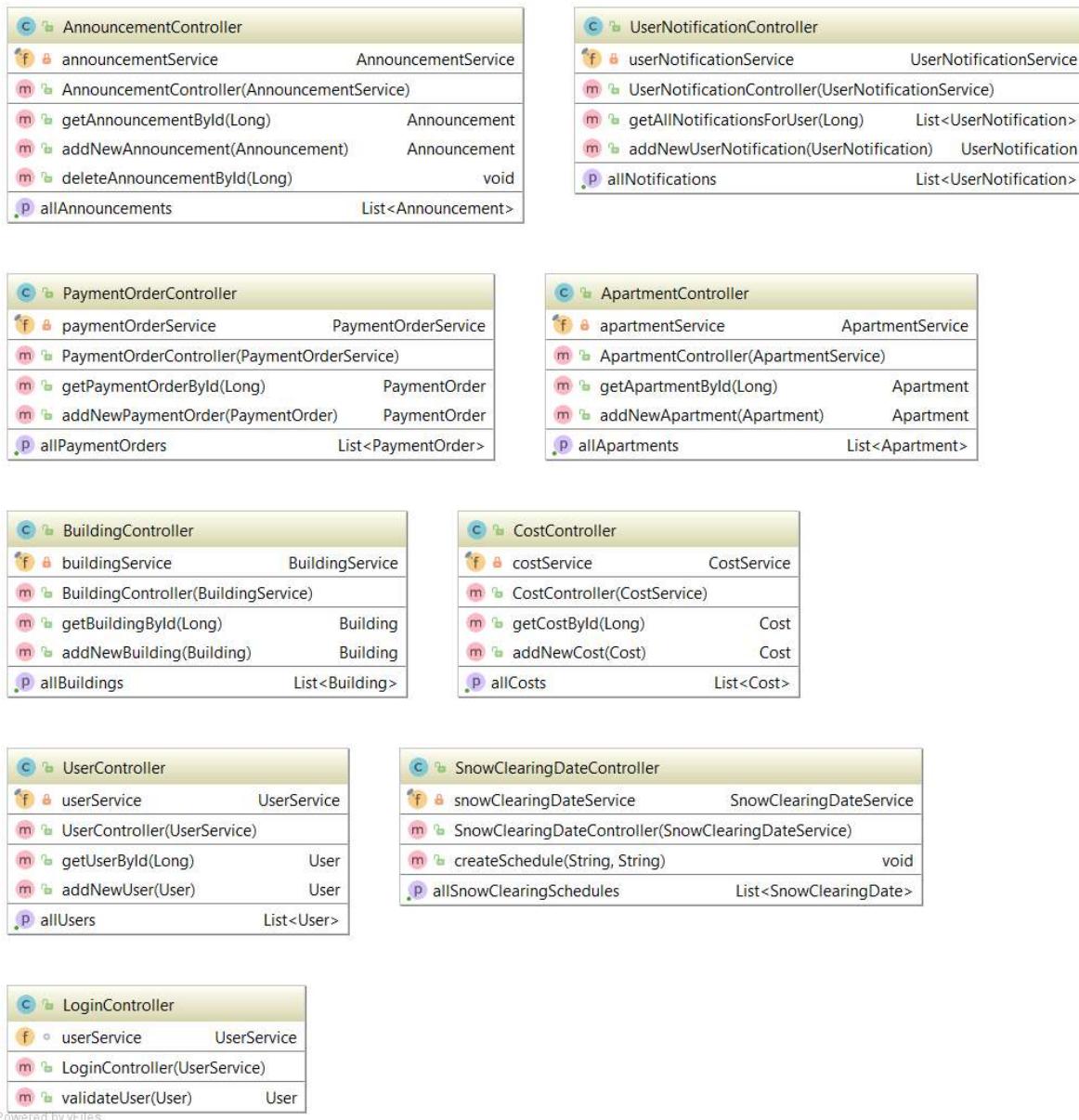


Slika 22 Dijagram razreda paketa service

Razredi koji su u paketu *controller*:

- Upravitelj obavijesti
 - Razred koji upravlja korisničkim zahtjevima za obradu podataka obavijesti. Njegov atribut je servis obavijesti.
 - Sadrži javne metode koje komuniciraju sa servisom obavijesti za dohvaćanje pojedinačnih i svih obavijesti. Postoje i javne metode za dodavanje i brisanje obavijesti.
- Upravitelj stanja
 - Razred koji upravlja korisničkim zahtjevima za obradu podataka stanova. Njegov atribut je servis stanova.
 - Sadrži javne metode koje komuniciraju sa servisom stanova za dohvaćanje pojedinačnih i svih podataka stanova. Postoje i javne metode za dodavanje i brisanje stanova.
- Upravitelj zgrade
 - Razred koji upravlja korisničkim zahtjevima za obradu podataka zgrada. Njegov atribut je servis zgrada.
 - Sadrži javne metode koje komuniciraju sa servisom zgrada za dohvaćanje pojedinačnih i svih podataka zgrada. Postoje i javne metode za dodavanje i brisanje zgrada.
- Upravitelj troškova
 - Razred koji upravlja korisničkim zahtjevima za obradu podataka troškova. Njegov atribut je servis troškova.
 - Sadrži javne metode koje komuniciraju sa servisom troškova za dohvaćanje pojedinačnih i svih podataka troškova. Postoje i javne metode za dodavanje i brisanje troškova.
- Upravitelj prijave korisnika
 - Razred koji upravlja prijavom korisnika na aplikaciju. Njegov atribut je servis korisnika.
 - Sadrži javne metode koje komuniciraju sa servisom korisnika za provjeru ispravnosti podataka koje je korisnik unio pri prijavi na sustav.
- Upravitelj naloga
 - Razred koji upravlja korisničkim zahtjevima za obradu podataka naloga. Njegov atribut je servis naloga.
 - Sadrži javne metode koje komuniciraju sa servisom naloga za dohvaćanje pojedinačnih i svih podataka zgrada. Postoje i javne metode za dodavanje i brisanje naloga.

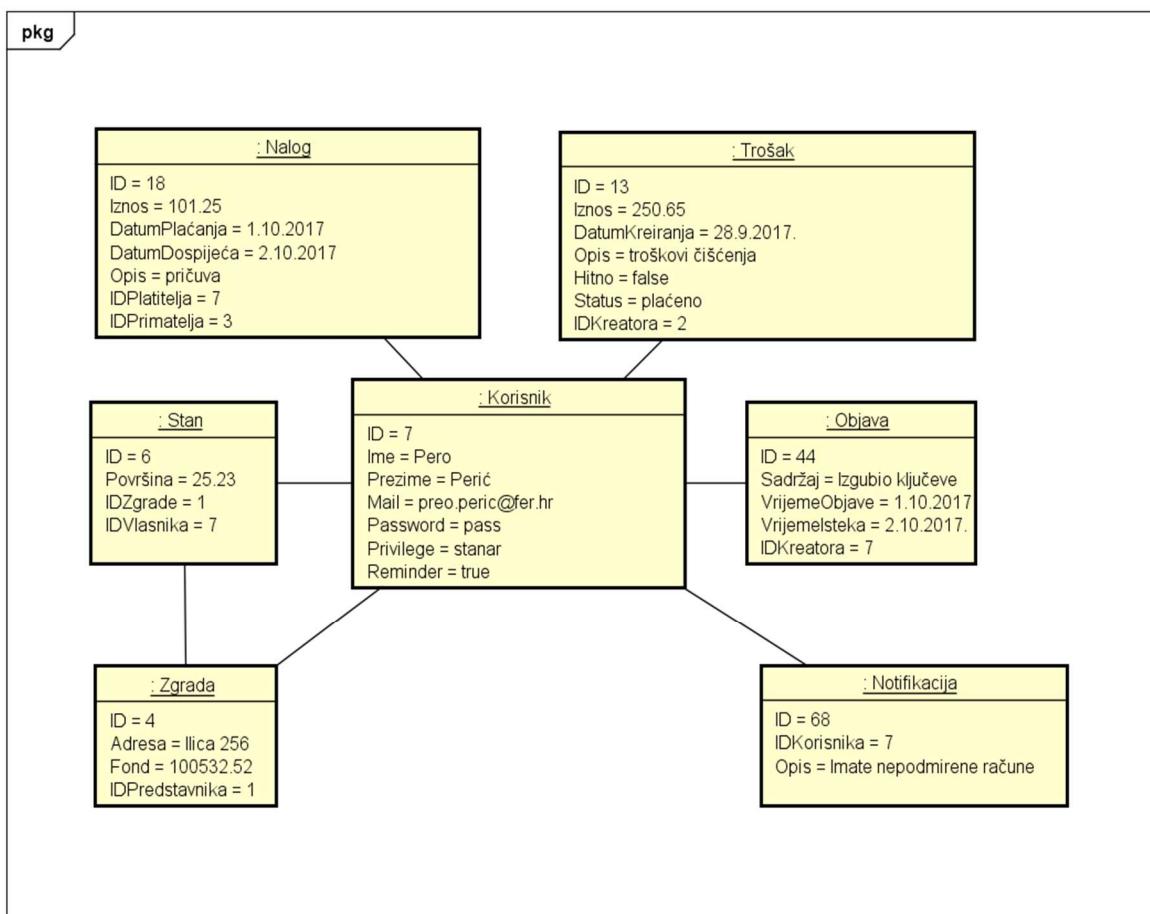
- Upravitelj termina čišćenja snijega
 - Razred koji upravlja korisničkim zahtjevima za obradu podataka termina čišćenja snijega. Njegov atribut je servis termina čišćenja snijega.
 - Sadrži javne metode koje komuniciraju sa servisom termina čišćenja snijega za dohvaćanje pojedinačnih i svih podataka termina čišćenja snijega. Postoje i javne metode za dodavanje i brisanje termina čišćenja snijega te kreiranje rasporeda čišćenja snijega.
- Upravitelj korisnika
 - Razred koji upravlja korisničkim zahtjevima za obradu podataka korisnika. Njegov atribut je servis korisnika.
 - Sadrži javne metode koje komuniciraju sa servisom korisnika za dohvaćanje pojedinačnih i svih podataka korisnika. Postoje i javne metode za dodavanje i brisanje korisnika.
- Upravitelj obavijesti korisnicima
 - Razred koji upravlja korisničkim zahtjevima za obradu podataka obavijesti korisnicima. Njegov atribut je servis obavijesti korisnicima.
 - Sadrži javne metode koje komuniciraju sa servisom obavijesti korisnicima za dohvaćanje pojedinačnih i svih podataka obavijesti korisnicima. Postoje i javne metode za dodavanje i brisanje obavijesti "korisnicima".



Slika 23 Dijagram razreda paketa controller

6.3. Dijagram objekata

Pomoću dijagrama objekata prikazat ćemo stanje sustava u nekom trenutku. Odabrali smo trenutak kada je na web aplikaciji prijavljen korisnik Pero Perić. Pero je vlasnik stana površine 25.23 m² u llici 256. Također vidi svoj plaćeni nalog za 'pričuvu', te također i plaćeni trošak čišćenja koji iznosi 250.65. Stanar Pero je objavio objavu 'Izgubio ključeve', koja ističe dan nakon objavlјivanja. Pero ima uključene podsjetnike, a primio je i notifikaciju o nepodmirenenim računima.



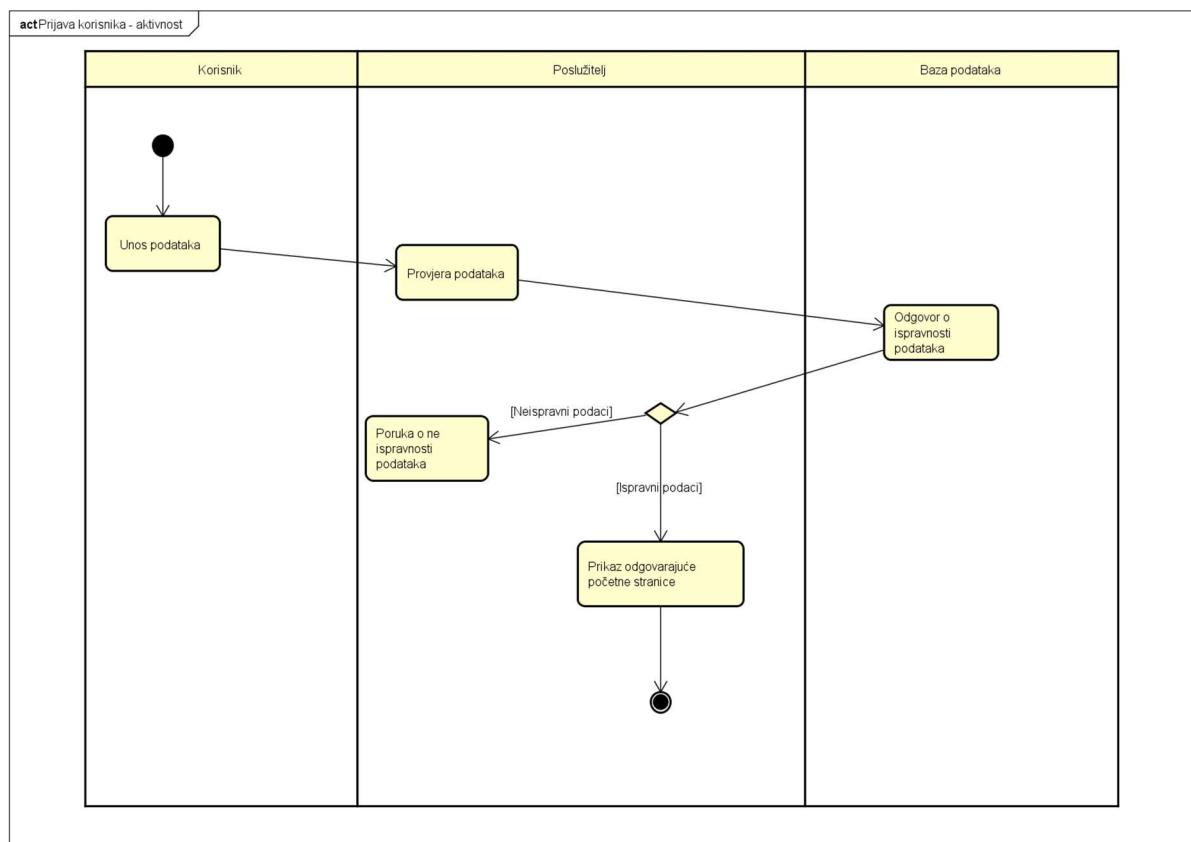
Slika 24 Dijagram objekata stvarnog stanja sustava

6.4. Ostali UML dijagrami

Dijagrami aktivnosti

Prijava korisnika

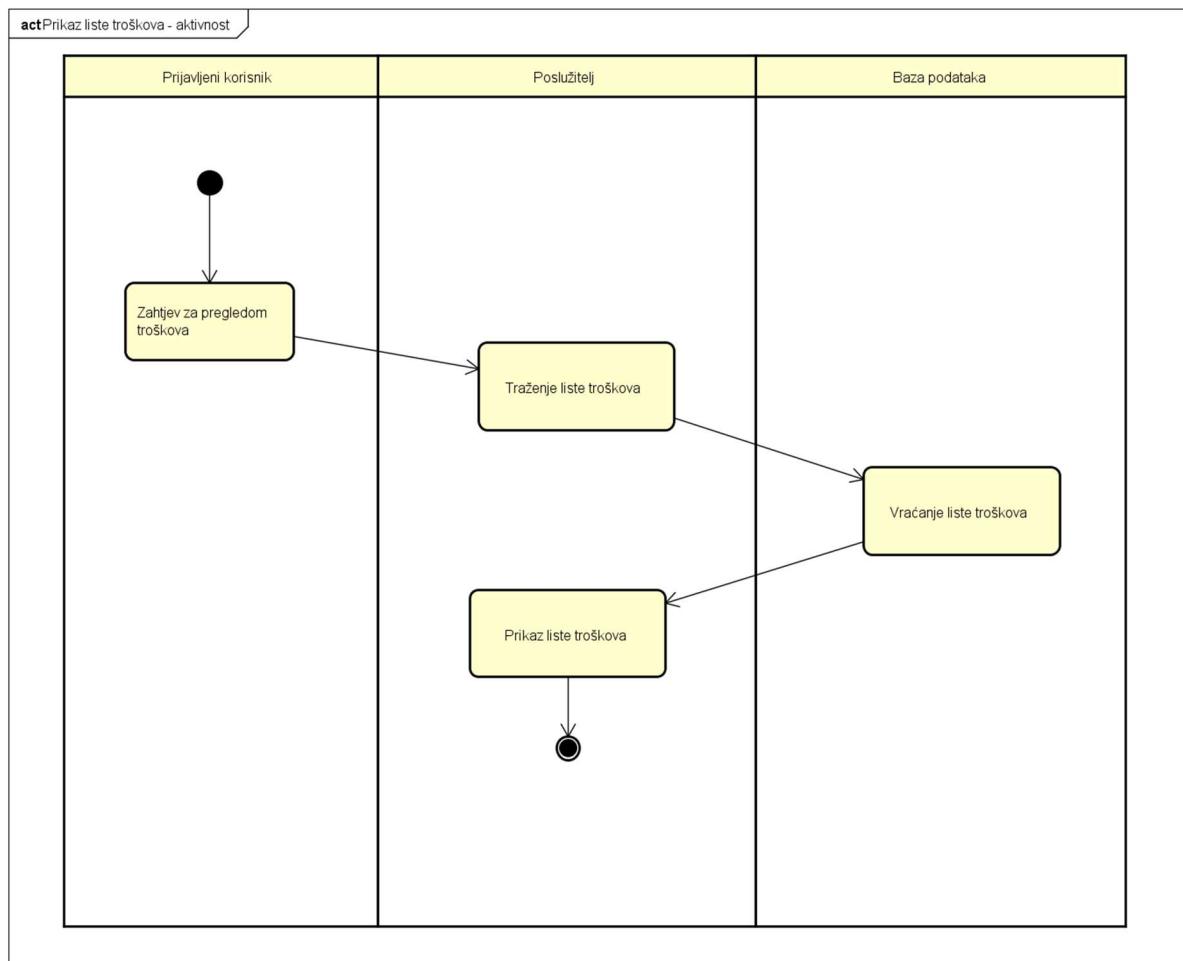
Dijagram aktivnosti „Prijava korisnika“ prikazuje komunikaciju između korisnika, poslužitelja i baze podataka. Korisnik unosi podatke o prijavi, te poslužitelj, prosljeđujući podatke bazi podataka, provjerava ispravnost istih. Baza odgovorom potvrđuje ispravnost ili neispravnost unesenih podataka. Ako uneseni podaci nisu ispravni, korisnik ostaje na stranici za prijavu, dok u slučaju ispravnih podataka, korisnika se prosljeđuje na odgovarajuću početnu stranicu, ovisno o vrsti korisnika (administrator/stanar).



Slika 25 Dijagram Aktivnosti-Prijava Korisnika

Prikaz liste troškova

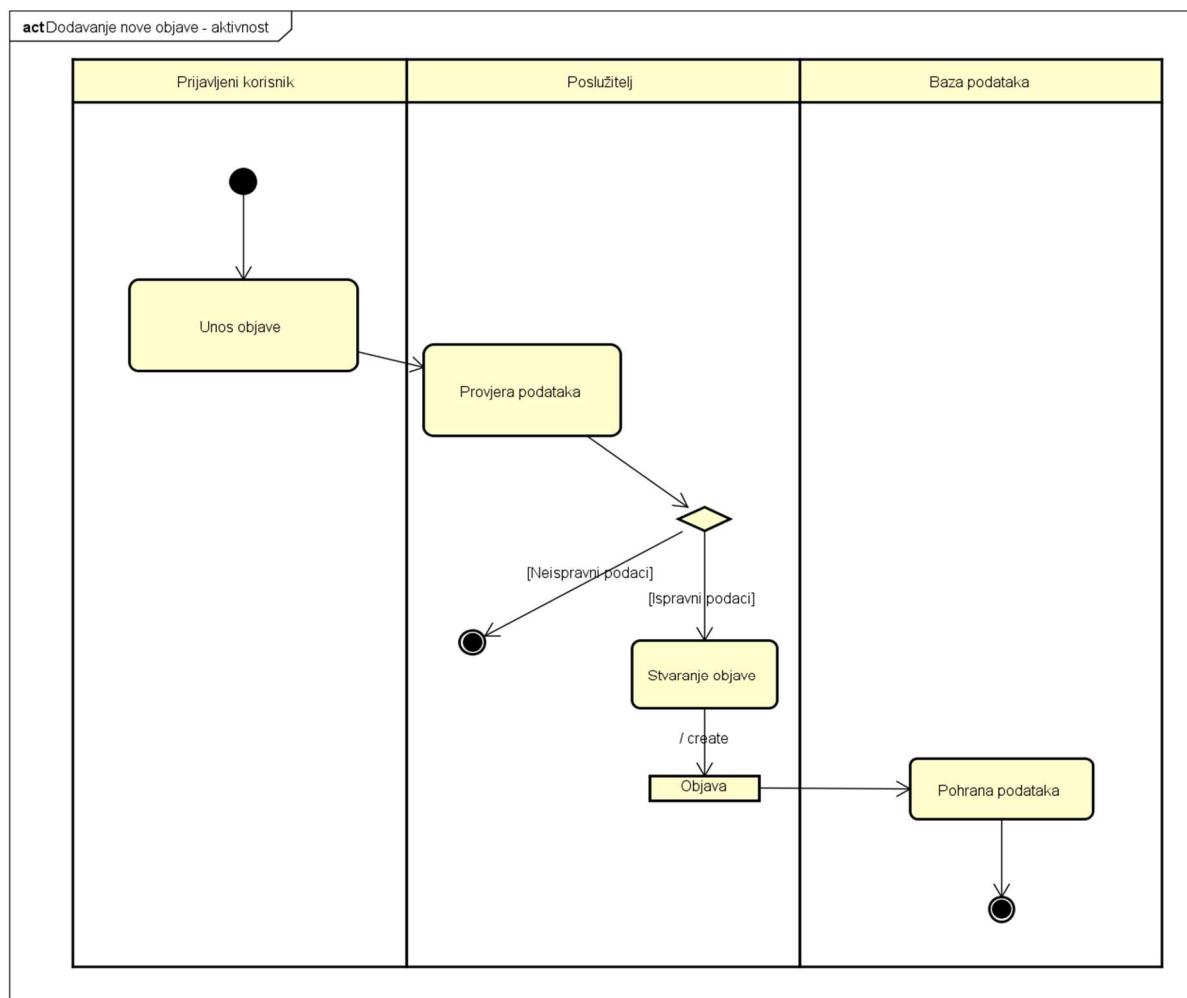
Dijagram aktivnosti „Prikaz liste troškova“ prikazuje komunikaciju između prijavljenog korisnika, poslužitelja i baze podataka. Prijavljeni korisnik šalje zahtjev za pregledom troškova. Poslužitelj taj zahtjev prosljeđuje bazi podataka, koja mu vraća listu troškova.



Slika 26 Dijagram Aktivnosti - Prikaz liste troškova

Dodavanje nove objave

Dijagram aktivnosti „Dodavanje nove objave“ prikazuje komunikaciju između prijavljenog korisnika, poslužitelja i baze podataka. Prijavljeni korisnik unosi podatke o novoj objavi. Poslužitelj provjerava ispravnost tih podataka i ako su neispravni/nepotpuni, korisnika se pušta da ponovno unese podatke. Ako su uneseni podaci ispravni, stvara se nova objava koja se pohranjuje u bazu podataka.



Slika 27 Dijagram Aktivnosti - Dodavanje nove objave

Dijagram stanja

Dijagrami stanja su ponašajni (dinamički) dijagrami koji opisuju diskretna stanja sustava i prijelaze između njih. Dijagram stanja „Korisnički prikaz“ prikazuje mogućnosti korisnika počevši od prijave. Početna stranica je „Stranica prijave“. U njoj korisnik unosi korisničke podatke za prijavu, te ako su uneseni neispravni, korisniku se šalje odgovarajuća poruka o grešci. No ako su podatci ispravni, korisnika se preusmjerava na „Stranicu oglasne ploče“. Korisnik nakon prijave ima više mogućnosti između kojih može odabratи, ovisno o podacima koje želi pregledati. Na raspolaganju su mu:

Nalozi- do kojih dolazi klikom na „Nalozi“, što ga preusmjerava na „Stranicu naloga“, gdje iste može pregledati,

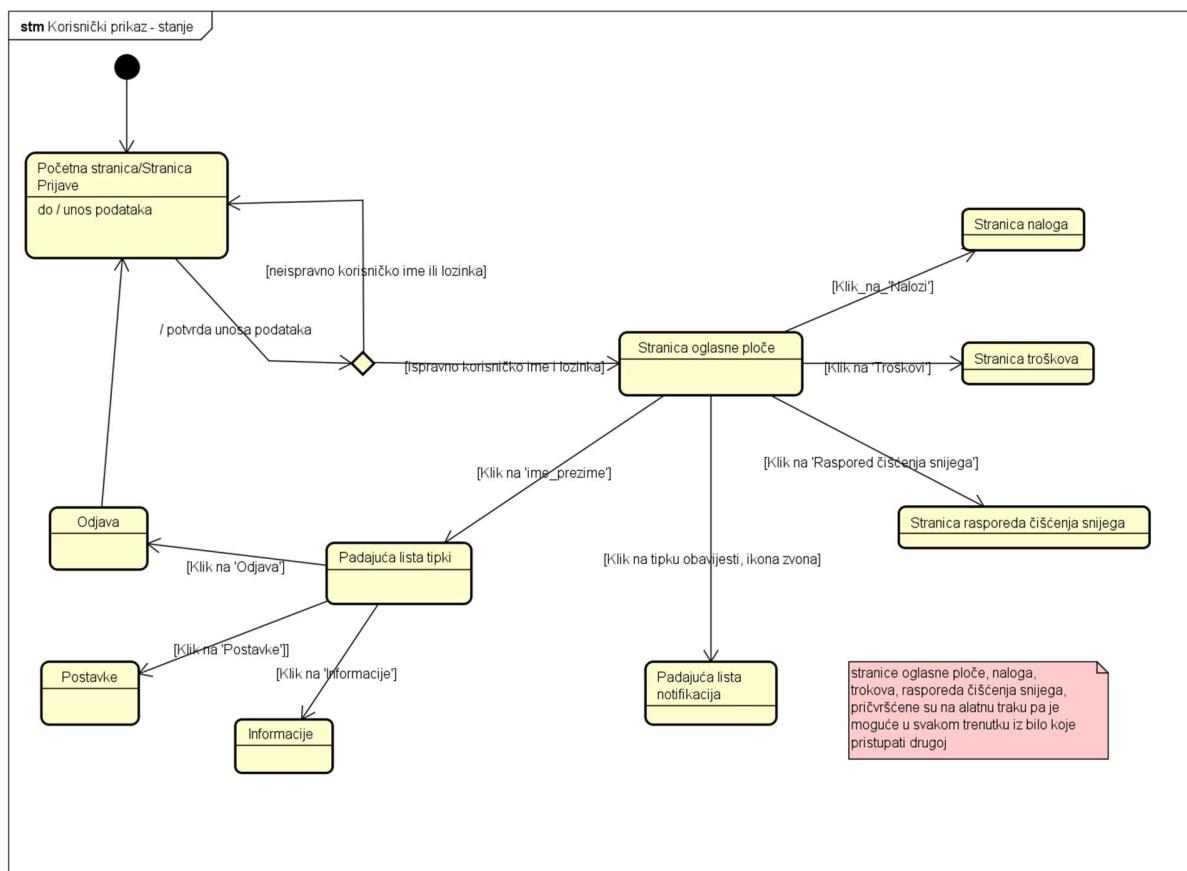
Troškovi- do kojih dolazi klikom na „Troškovi“, što ga preusmjerava na „Stranicu troškova“, gdje iste može pregledati,

Raspored čišćenja snijega- do kojih dolazi klikom na „Raspored čišćenja snijega“, što ga preusmjerava na „Stranicu rasporeda čišćenja snijega“, gdje iste može pregledati,

Padajuća lista notifikacija

Padajuća lista tipki- koja sadrži „Odjavu“, „Postavke“ i „Informacije“. Klikom na „Odjavu“ korisnik se odjavljuje i vraća na „Stranicu za Prijavu“. Klikom na „Postavke“, korisniku se otvara prozor gdje može urediti svoje korisničke podatke. Klikom na „Informacije“, korisniku se otvara prozor koji pokazuje korisničke informacije, zajedno sa informacijama o stanu i zgradiji kojoj pripada korisnik.

Do svih navedenih podataka, može se jednostavno doći, jer su za alatnu traku pričvršćeni odgovarajući gumbi koji omogućuju brzi prijenos s jedne na bilo koju drugu stranicu.

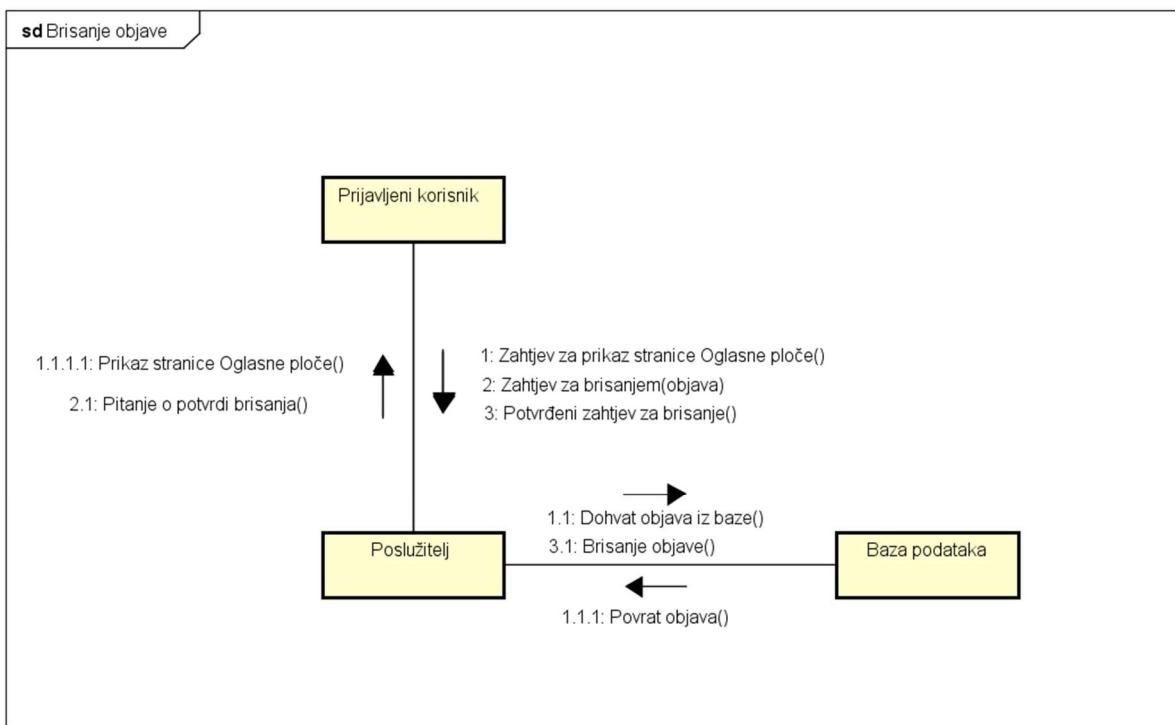


Slika 28 Dijagram Stanja - Korisnički prikaz

Komunikacijski dijagrami

Brisanje objave

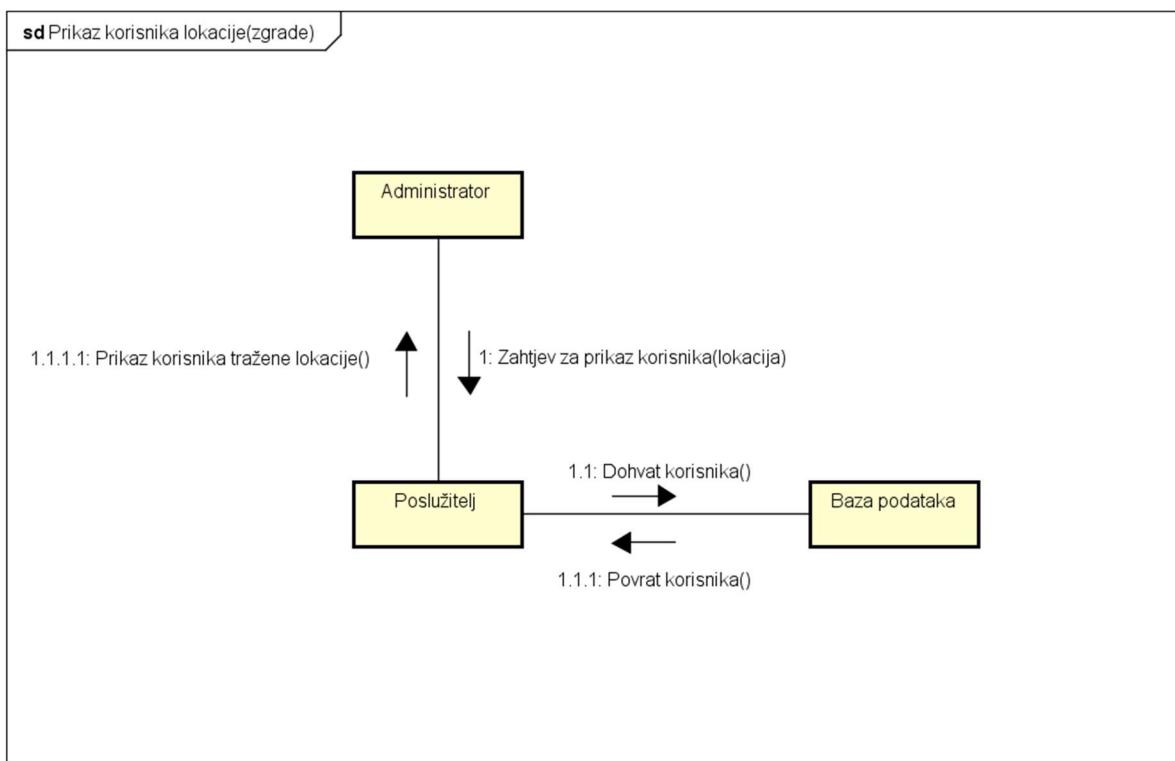
Komunikacijski dijagram „Brisanje objave“ prikazuje komunikaciju između prijavljenog korisnika, poslužitelja i baze podataka u ovisnosti o vremenu. Korisnik prvo zatražuje prikaz „Stranice oglasne ploče“, prepostavljajući da se na istoj ne nalazi. Baza puni podatke koje poslužitelj dalje proslijeđuje korisniku. Korisnik odabire koju objavu želi obrisati, na što ga poslužitelj traži dodatnu potvrdu za brisanje. Pristankom na dodatnu potvrdu u bazi podataka briše se odabrana objava.



Slika 29 Komunikacijski dijagram - Brisanje objave

Prikaz korisnika lokacije

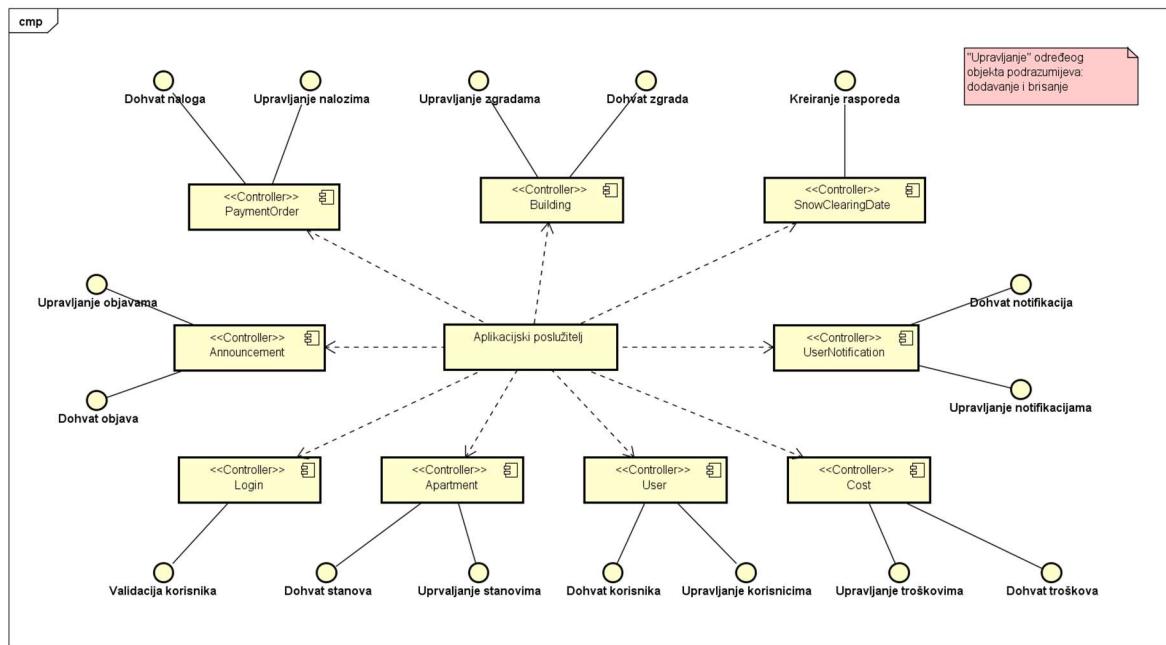
Komunikacijski dijagram „Prikaz korisnika lokacije“ opisuje komunikaciju između administratora, poslužitelja i baze podataka. Administrator zatražuje prikaz korisnika(stanara) koji žive na određenoj adresi, tj. u određenoj zgradi. Poslužitelj podatke o traženim korisnicima dohvaća iz baze podataka, te ih proslijeđuje nazad administratoru.



Slika 30 Komunikacijski dijagram - Prikaz korisnika lokacije(zgrade)

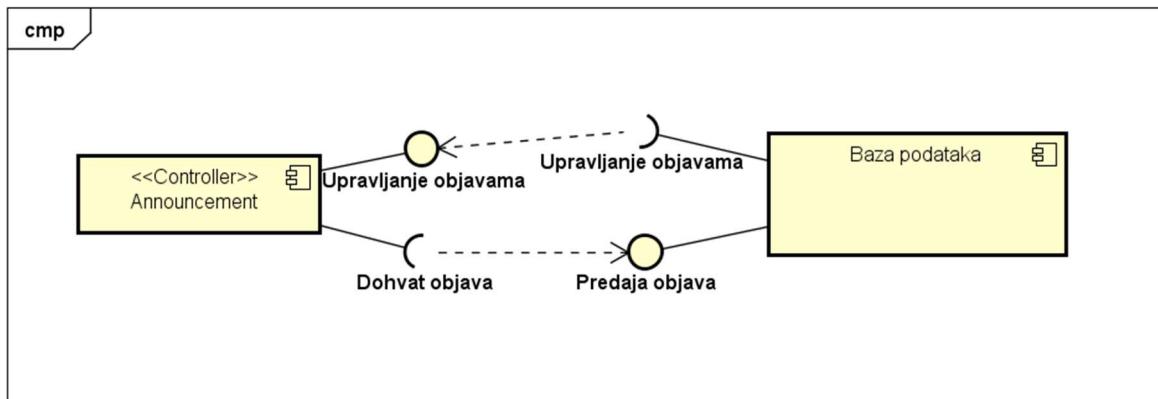
Dijagrami komponenti

Dijagram komponenti prikazuje strukturu sustava pomoću komponenti koje ga sačinjavaju. Naš sustav ima oblik web aplikacije, napravljene korištenjem Java. Korisnik raznim akcijama (unosom, klikom miša) prema korisničkom sučelju ukazuje na promjene koje, određeni kontroler, na određeni način, treba izvršiti. Te promjene također se primjenjuju i u bazi podataka. Tako je npr. jedan od naših kontrolera "PaymentOrder", čija je zadaća upravljanje i dohvata naloga iz baze podataka.



Slika 31 Dijagram komponenti

Na ovom dijagramu komponenti prikazana je komunikacija i rad komponente "Announcement" kontrolira sa bazom podataka. U dijagramu se jasno vidi kada se iz baze podataka uzimaju podaci (Dohvatom objava), a kada se u bazu spremaju ili uređuju podaci (Upravljanje objavama). Analogni prikaz vrijedio bi i za ostale kontrolere, samo prema zadaćama prikazanim kao na prethodnoj slici.

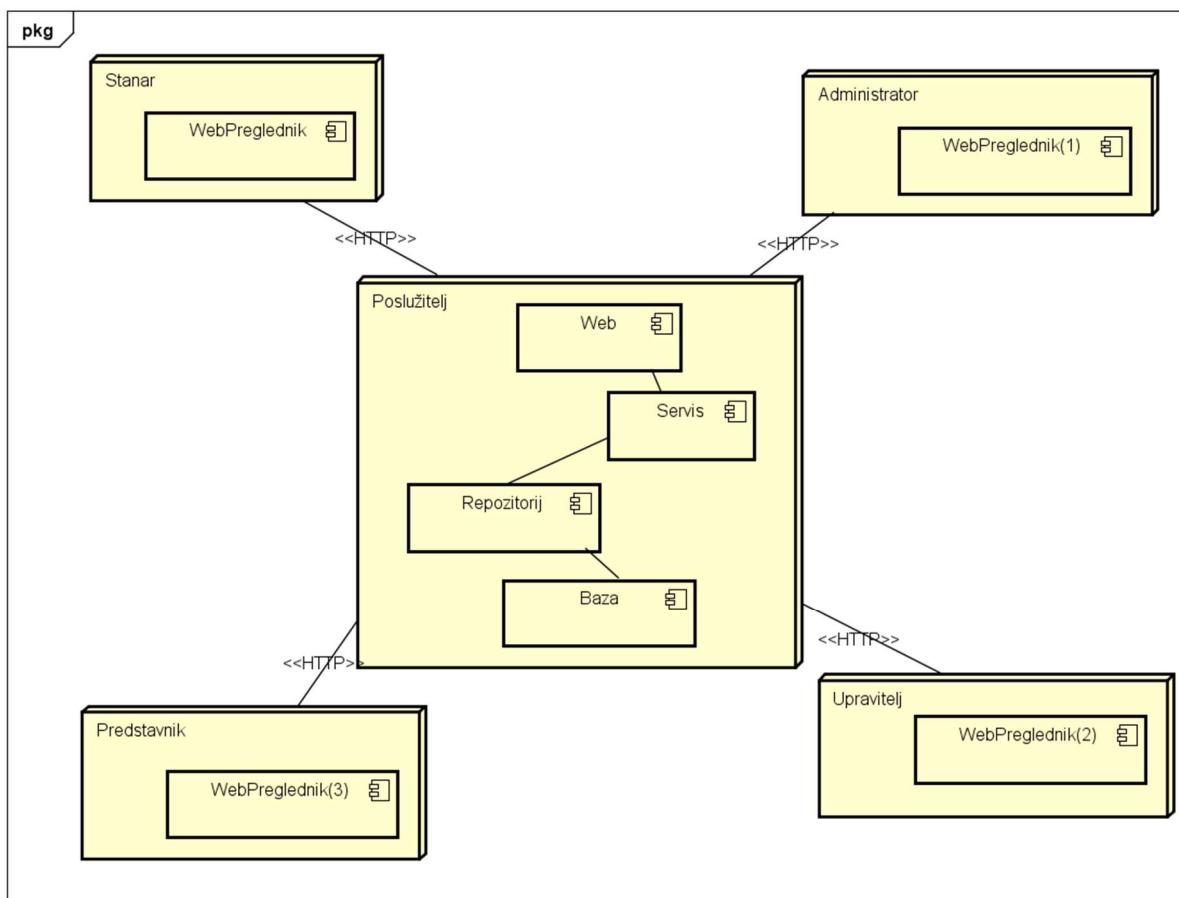


Slika 32 Dijagram komponenti - Objave

7. Implementacija i korisničko sučelje

7.1. Dijagram razmještaja

Pomoću dijagrama razmještaja smo prikazali, slikom 25, generalnu topologiju sustava. Sustav je baziran na arhitekturi „klijent-poslužitelj“, a komunikacija između računala klijenata i poslužitelja odvija se preko HTTP veze. Klijenti u našem slučaju su stanar, administrator, upravitelj i predstavnik koji se preko web preglednika spajaju na web poslužitelj. Unutar poslužitelja se nalazi aplikacija koja se sastoji od web sloja, servisnog sloja i sloja za pristup bazi podataka.



Slika 33 Dijagram Razmještaja

7.2. Korištene tehnologije i alati

Pri izradi web aplikacije korištena je razvojna okolina *IntelliJ IDEA*. Također je prilikom izrade dokumentacije i aplikacije korišten *Git* sustav pomoću kojeg su se kontrolirale verzije na repozitoriju.

Za izradu UML dijagrama korišten je Astah Professional.

React:

React je biblioteka JavaScript programskog jezika koja služi za izradu korisničkog sučelja. Omogućuje programerima stvaranje velikih web-aplikacija koje koriste podatke i mogu se mjenjati kroz vrijeme bez učitavanja stranice. Prvenstveno nastoji osigurati brzinu, jednostavnost i skalabilnost.

Redux:

Redux je spremnik koji sadrži stanje JavaScript aplikacije. Primarno (ali ne i nužno) se koristi zajedno sa React-om za izgradnju korisničkih sučelja.

Spring Boot:

Spring Boot olakšava stvaranje samostalnih aplikacija koje se temelje na Spring proizvodima koje možete „samo pokrenuti“.

CSS:

CSS je stilski jezik koji se koristi za opis prezentacije dokumenta napisanog pomoću HTML jezika. CSS-om se uređuje sam izgled i raspored stranice.

HTML:

To je prezentacijski jezik za izradu web stranica. Hipertekst dokument stvara se pomoću HTML jezika. HTML jezikom oblikuje se sadržaj i stvaraju se hiperveze hipertext dokumenta. Prikaz hipertext dokumenta omogućuje web preglednik. Temeljna zadaća HTML jezika jest uputiti web preglednik kako prikazati hipertext dokument.

Hibernate ORM:

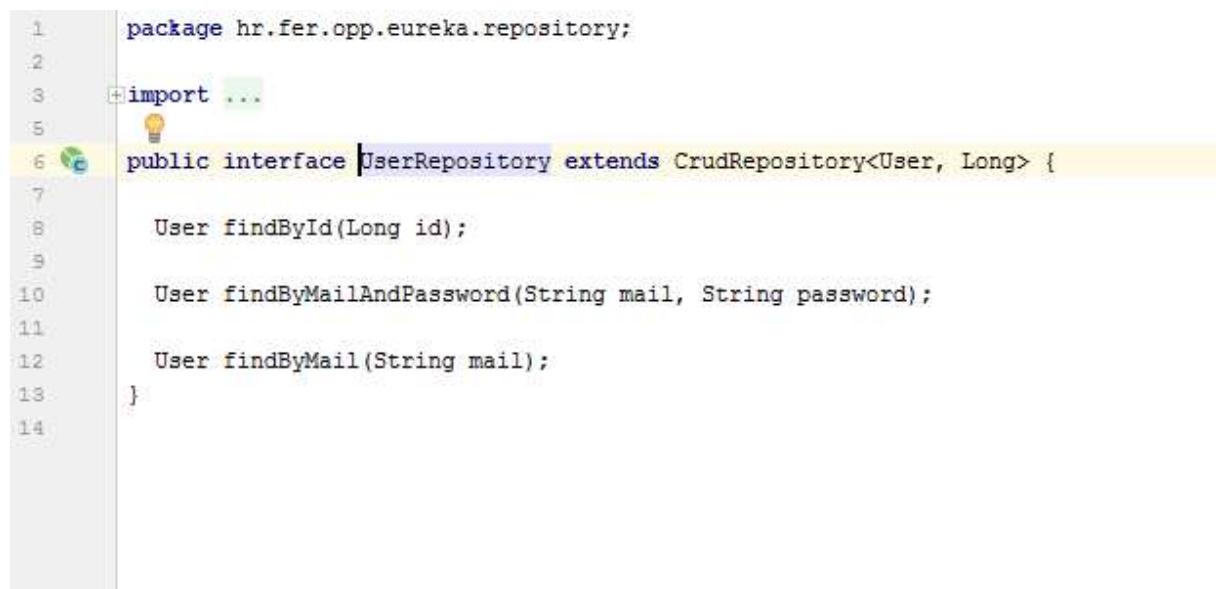
Hibernate ORM je radni okvir koji mapira objektno-relacijski model baze podataka u programskom jeziku Java. Pruža okvir za mapiranje objektno orientiranog modela u relacijsku bazu podataka. Njegova glavna značajka je mapiranje iz Java klase u tablice baze podataka. Također nudi sučelje za rad sa SQL upitima i obrađuje konverziju rezultata SQL upita u Java objekte.

JSON:

Java Script Object Notation je minimalan format paketa za prijenos podataka koji ima prednost u lakoći čitanja i pisanja za ljude, te se odlikuje lakoćom strojne obrade generiranjem i parsiranjem što ga čini pogodnim kao sredstvo za prijenos podataka neovisno o tome koja se platforma koristi.

7.3. Isječak programskog koda vezan za temeljnu funkcionalnost

Prikazano je sučelje *UserRepository.java* koje sadrži metode koje omogućuju komuniciranje sa bazom podataka, odnosno mogućnost pretraživanja korisnika po ID-u, mail-u ili mail-u i lozinci zajedno. Također *CrudRepository.java* sadrži metode kao što su spremanje i brisanje zapisa.



```
1 package hr.fer opp.eureka.repository;
2
3 import ...
4
5 public interface UserRepository extends CrudRepository<User, Long> {
6
7     User findById(Long id);
8
9     User findByMailAndPassword(String mail, String password);
10
11    User findByMail(String mail);
12
13 }
```

Slika 34 UserRepository

Slika 27 prikazuje *UserService.java*, odnosno sučelja zadužena za komunikaciju s bazom podataka. Sadrži javne metode za dohvaćanje korisnika iz baze, za dodavanje i brisanje korisnika, za provjeru postoji li korisnik, te omogućuje editiranje korisničkih podataka. Sve metode su ostvarene u *UserController.java*. *UserService.java* je servisni sloj koji sadrži poslovnu logiku kod upravljanja korisnicima. Servisni sloj povezuje web sloj i rezitorij.

The screenshot shows a Java code editor with the following code:

```
1 package hr.fer opp.eureka.service;
2
3 import ...
4
5 public interface UserService {
6     List<UserResponse> getAll();
7     UserResponse getById(Long id);
8     UserResponse validateUser(String mail, String password);
9     UserResponse add(UserRequest userRequest, Long apartmentId);
10    UserResponse add(UserRequest userRequest);
11    UserResponse getByMail(String mail);
12    UserResponse edit(UserRequest userRequest);
13    UserResponse editPassword(UserRequest userRequest);
14    UserResponse toggleReminderValue(Long id);
15    void deleteById(Long id);
16}
```

The code defines a public interface named `UserService` with various methods for managing users. The interface is annotated with `hr.fer opp.eureka.service` and imports `...`. It includes methods for getting all users, getting a user by ID, validating a user's credentials, adding a user (with and without apartment ID), getting a user by email, editing a user, changing a password, toggling a reminder value, and deleting a user by ID.

Slika 35 UserService

Prikazan je isječak razreda koji upravlja korisničkim zahtjevima i implementacija je metoda iz razreda *UserService.java*. Razred *UserController.java* sadrži javne metode koje komuniciraju sa *UserService.java* za dohvaćanje pojedinačnih i svih podataka korisnika. *UserController.java* je web sloj koji upravlja korisnicima i zaprima http zahtjeve, te poziva metode servisnog sloja.

```
1 package hr.fer opp.eureka.controller;
2
3 import ...
4
5
6 @RestController
7 public class UserController {
8
9     private final UserService userService;
10
11     @Autowired
12     public UserController(UserService userService) { this.userService = userService; }
13
14     @GetMapping("/api/users")
15     public List<UserResponse> getAllUsers() { return userService.getAll(); }
16
17     @GetMapping("/api/users/{id}")
18     public UserResponse getUserId(@PathVariable Long id) { return this.userService.getById(id); }
19
20     @PostMapping("/api/users/{apartmentId}")
21     public UserResponse addNewUser(@RequestBody final UserRequest userRequest, @PathVariable Long apartmentId) {
22         return userService.add(userRequest, apartmentId);
23     }
24
25     @PutMapping("/api/users/edit")
26     public UserResponse editUser(@RequestBody final UserRequest userRequest) {
27         return this.userService.edit(userRequest);
28     }
29
30     @PutMapping("/api/users/edit-password")
31     public UserResponse editUserPassword(@RequestBody final UserRequest userRequest) {
32         return this.userService.editPassword(userRequest);
33     }
34 }
```

Slika 36 UserController

User.java je entitet koji predstavlja korisnika, te sadrži sve atribute i povezanosti s ostalim entitetima.

```
1 package hr.fer.opp.eureka.domain.user;
2
3 import ...
4
5 @Entity
6 @Table (name = "app_user")
7 public class User {
8
9     @Id
10    @GeneratedValue(strategy = GenerationType.IDENTITY)
11    private Long id;
12
13    private String firstName;
14
15    private String lastName;
16
17    private String mail;
18
19    @Enumerated(EnumType.STRING)
20    private UserPrivilege privilege;
21
22    private Boolean reminder;
23
24    private String password;
25
26    @JsonIgnore
27    @OneToMany(mappedBy = "user")
28    private Set<Announcement> announcementSet;
29
30    @JsonIgnore
31    @OneToMany (mappedBy = "user", fetch = FetchType.LAZY)
32    private List<UserNotification> userNotificationList;
33
34
35    @JsonIgnore
36    @OneToMany (mappedBy = "creator", fetch = FetchType.LAZY)
37    private Set<Cost> costs;
38
39    @JsonIgnore
40    @OneToMany (mappedBy = "landlord", fetch = FetchType.LAZY)
41    private Set<Building> landlordBuildingSet;
42
43    @JsonIgnore
44    @OneToMany (mappedBy = "manager", fetch = FetchType.LAZY)
45    private Set<Building> managerBuildingSet;
46
47    @JsonIgnore
48    @OneToMany (mappedBy = "owner", fetch = FetchType.LAZY)
49    private Set<Apartment> apartments;
50
51    @JsonIgnore
52    @OneToMany (mappedBy = "payer", fetch = FetchType.LAZY)
53    private Set<PaymentOrder> paymentOrdersToPay;
54
55    @JsonIgnore
56    @OneToMany (mappedBy = "receiver", fetch = FetchType.LAZY)
57    private Set<PaymentOrder> paymentOrdersToReceive;
58
59    @JsonIgnore
60    @OneToMany (mappedBy = "user", fetch = FetchType.LAZY)
61    private Set<SnowClearingDate> snowClearingDates;
62
63
64    public User() {
65    }
```

Slika 37 UserDomain

7.4. Ispitivanje programskog rješenja

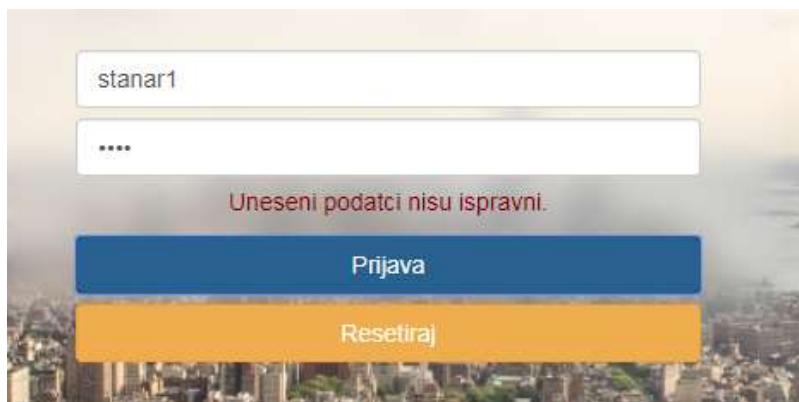
Ispituju se temeljne funkcionalnosti sustava i sve radnje svih korisnika po slučajevima gdje će svaki slučaj imati opis slučaja, te očekivani i dobiveni rezultat.

Slučaj 1:

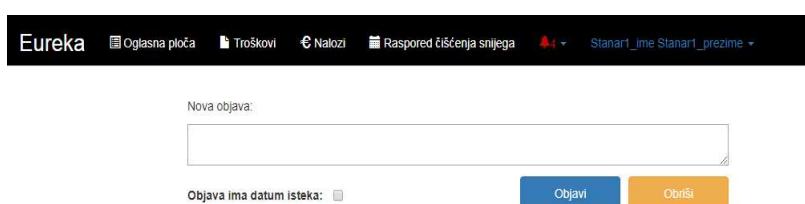
Opis: Korisnik se prijavljuje na sustav.

Očekivani rezultat: Korisnik se uspješno prijavio.

Dobiveni rezultat: Ako je korisnik unio dobru lozinku i korisničko ime, korisnik se prijavljuje u sustav, dok ako je unio neke pogrešne podatke, ispisati će mu se poruka da uneseni podaci nisu ispravni.



Slika 38 Pogrešna prijava



Slika 39 Uspješna prijava

Slučaj 2:

Opis: Unos nove objave na oglasnoj ploči

Očekivani rezultat: Nova objava sa datumom isteka je uspješno unesena i nalazi se na oglasnoj ploči.

Dobiveni rezultat: Ukoliko pod datum isteka stavimo datum koji je u prošlosti, sustav će korisniku odbiti objavu i naznačiti datum kao grešku. U suprotnom objava će biti prikazana na oglasnoj ploči.

Nova objava:
Izgubljeni ključevi

Objava ima datum isteka: 10-01-2018 X

Objavi **Obriši**

Slika 40 Neuspješan unos nove objave

Nova objava:

Objava ima datum isteka: 14-01-2018

Objavi **Obriši**

Izgubljeni ključevi
 Stanar1_ime Stanar1_prezime X

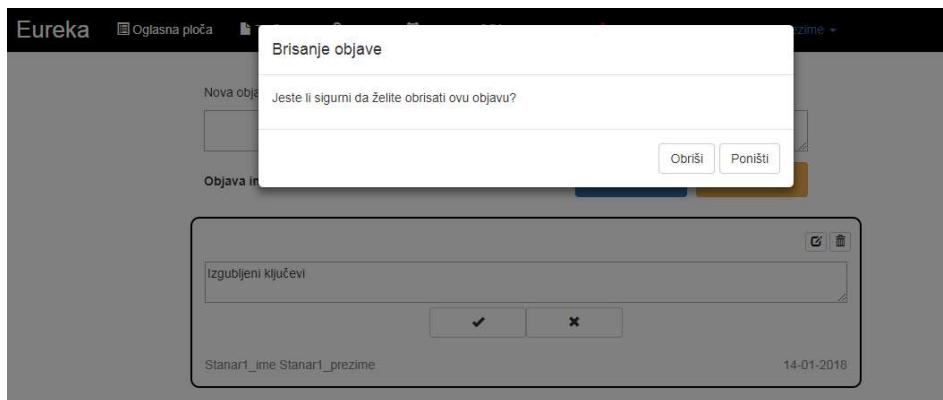
Slika 41 Uspješan unos objave

Slučaj 3:

Opis: Brisanje objave sa oglasne ploče.

Očekivani rezultat: Objava će biti uklonjena sa oglasne ploče.

Dobiveni rezultat: Korisnik je prije konačnog brisanja objave upitan je li siguran u brisanje objave i ako je siguran klikom na obriši, objava će biti obrisana.



Slika 42 Brisanje objave



Slika 43 Objava obrisana

Slučaj 4:

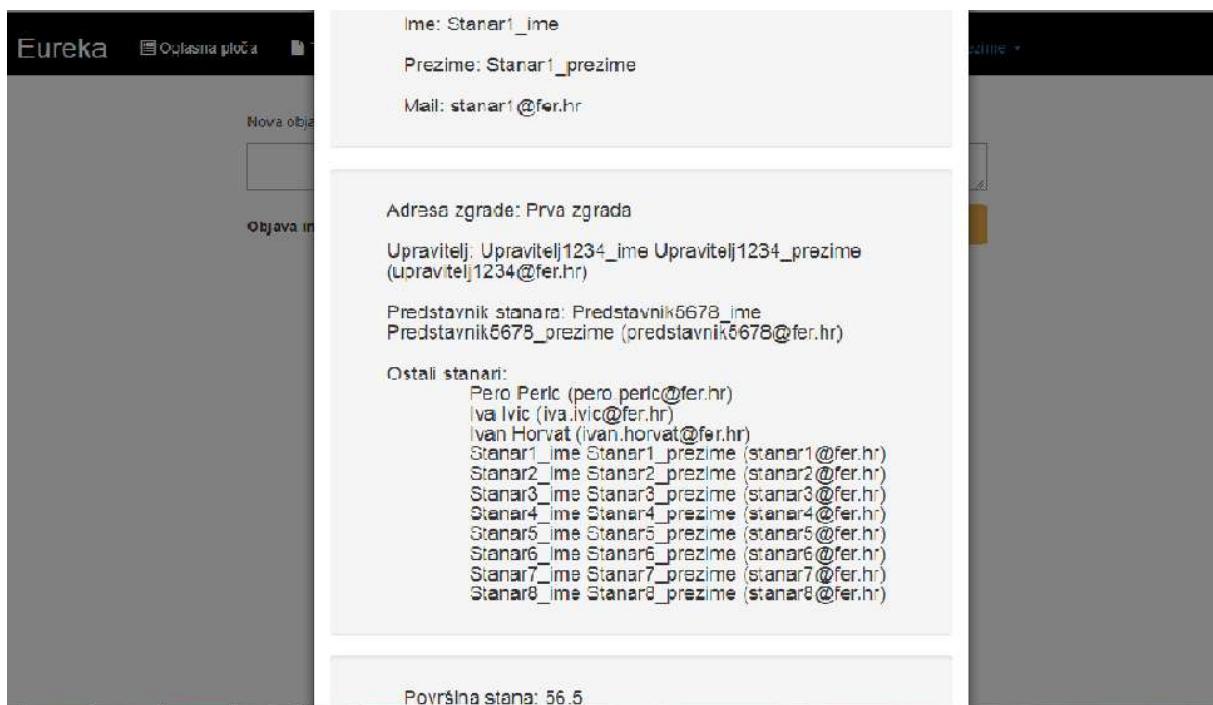
Opis: Pregled informacija korisnika.

Očekivani rezultat: Korisniku se prikazuju sve informacije o zgradi u kojem se njegovom stan nalazi, kao što su adresa zgrade, tko je upravitelj, i ostali stanari. Isto tako dobiva prikaz kvadrature svoga stana i prikaz vlastitih podataka.

Dobiveni rezultat: Korisniku se otvaraju svi podaci.



Slika 44 Pregled informacija-prvi korak



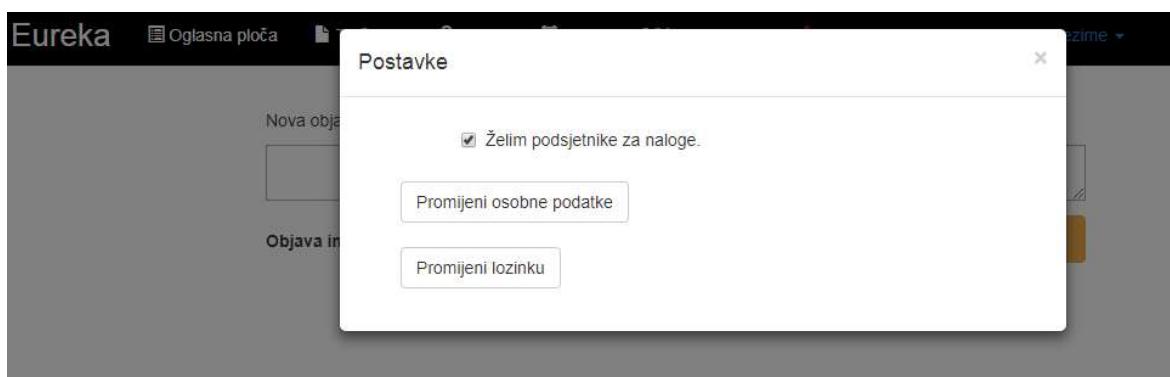
Slika 44 Pregled informacija

Slučaj 5:

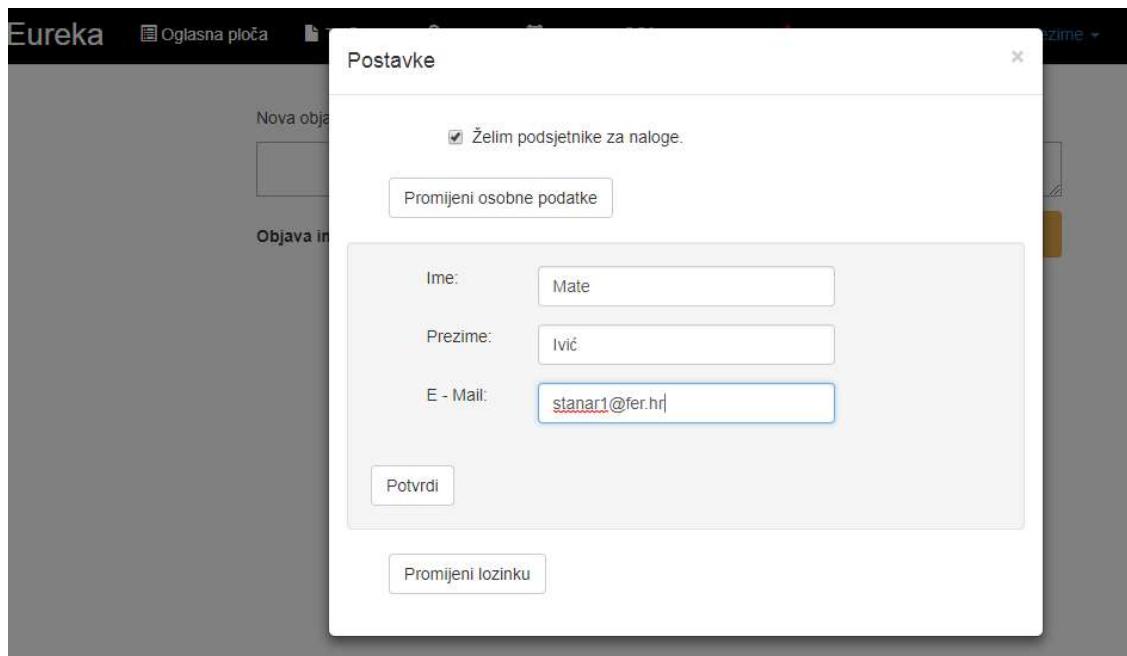
Opis: Uređivanje postavki, odnosno osobnih podataka te lozinke, te mogućnost odabira primanja podsjetnika.

Očekivani rezultat: Korisnik odabire koje podatke želi promijeniti te ih mijenja i potvrđuje te su podatci uspješno izmjenjeni, te ako želi odabire da želi primati podsjetnike.

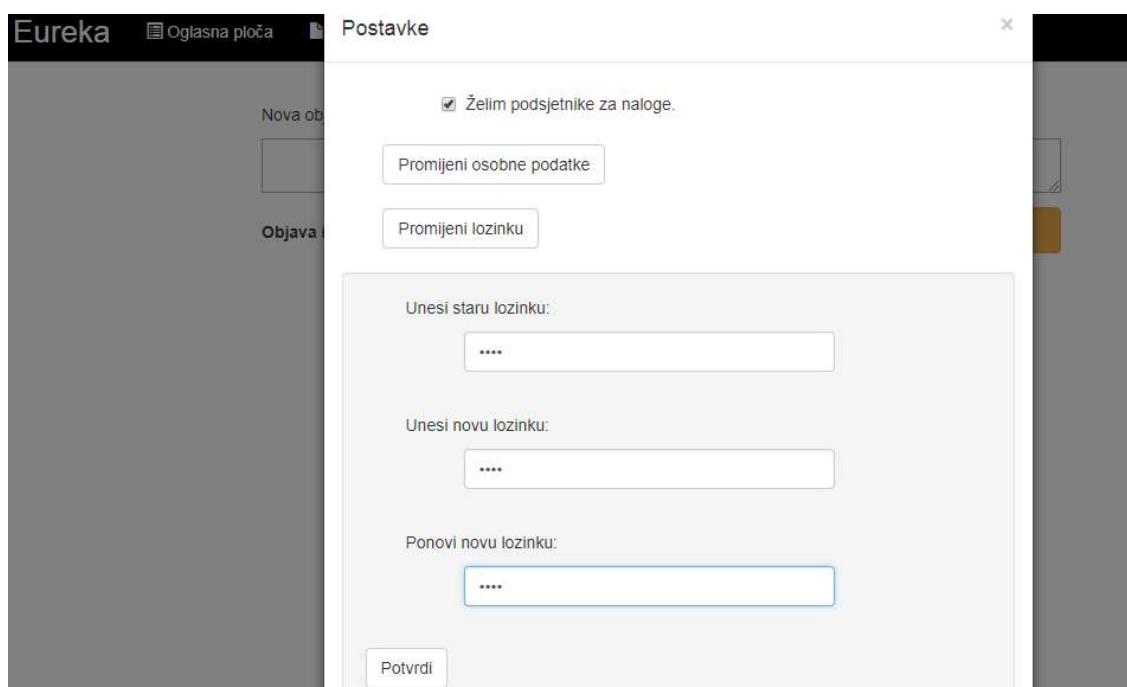
Dobiveni rezultat: Korisnik je uspješno promijenio osobne podatke te lozinku, te označio da želi primati obavijesti.



Slika 45 Uređivanje postavki



Slika 46 Promjena osobnih podataka



Slika 47 Promjena lozinke



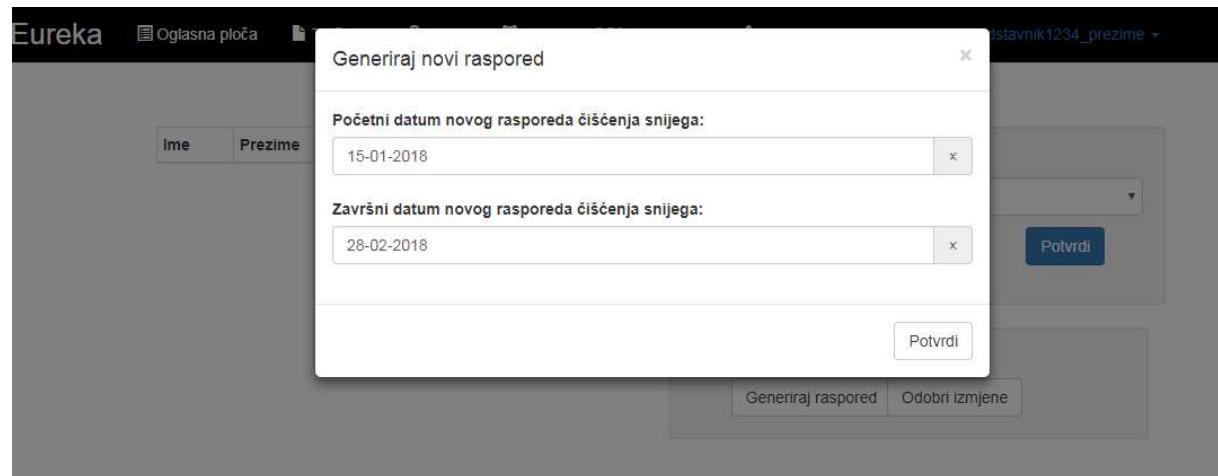
Slika 48 Uspješna promjena osobnih podataka

Slučaj 6:

Opis: Generiranje rasporeda čišćenja snijega.

Očekivani rezultat: Raspored je uspješno generiran.

Dobiveni rezultat: Predstavnik odabire od kojeg do kojeg datuma želi generirati raspored, te dobiva nakon potvrde, generirani raspored.



Slika 49 Generiranje rasporeda čišćenja snijega

The screenshot shows a table of employees with columns: Ime (Name), Prezime (Surname), and Datum (Date). The table contains 10 rows of data. To the right, there are two modal windows: one for requesting a schedule change by date, with a dropdown menu showing 'Odaberi' and a 'Potvrdi' button; and another for updating the schedule, with buttons for 'Generiraj raspored' (Generate schedule) and 'Odobri izmjene' (Approve changes).

Ime	Prezime	Datum
Mate	Ivić	15-01-2018
Stanar4_ime	Stanar4_prezime	16-01-2018
Stanar2_ime	Stanar2_prezime	17-01-2018
Stanar3_ime	Stanar3_prezime	18-01-2018
Predstavnik1234_ime	Predstavnik1234_prezime	19-01-2018
Mate	Ivić	20-01-2018
Stanar4_ime	Stanar4_prezime	21-01-2018
Stanar2_ime	Stanar2_prezime	22-01-2018
Stanar3_ime	Stanar3_prezime	23-01-2018

Slika 50 Uspješno generiran raspored

Slučaj 7:

Opis: Izmjena rasporeda čišćenja snijega.

Očekivani rezultat: Zatražene izmjene su uspješno proslijeđene predstavniku, te on radi izmjenu rasporeda.

Dobiveni rezultat: Predstavnik dobiva zahtjeve od stanara, no tek kada dobije minimalno dva zahtjeva može napraviti izmjenu rasporeda, te uspješno zamijeniti dva stanara u rasporedu.

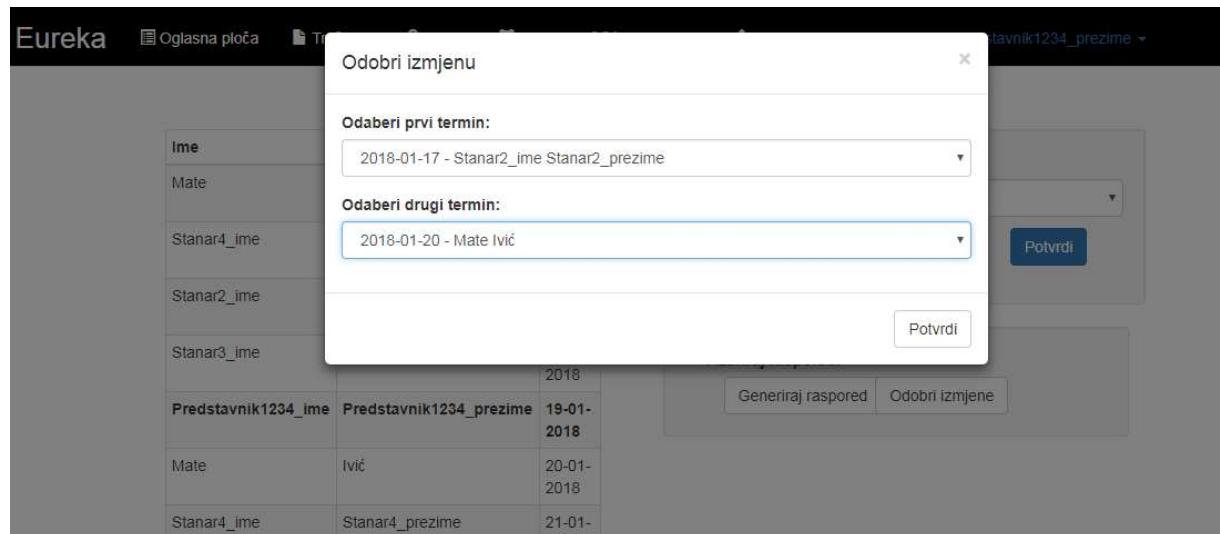
The screenshot shows a table of employees with columns: Ime (Name), Prezime (Surname), and Datum (Date). The table contains 10 rows of data. To the right, there is a single modal window for requesting a schedule change by date, with a dropdown menu showing '2018-01-20' and a 'Potvrdi' button.

Ime	Prezime	Datum
Mate	Ivić	15-01-2018
Stanar4_ime	Stanar4_prezime	16-01-2018
Stanar2_ime	Stanar2_prezime	17-01-2018
Stanar3_ime	Stanar3_prezime	18-01-2018
Predstavnik1234_ime	Predstavnik1234_prezime	19-01-2018
Mate	Ivić	20-01-2018
Stanar4_ime	Stanar4_prezime	21-01-2018

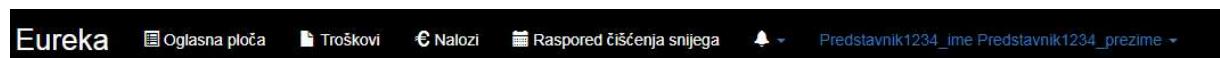
Slika 51 Stanar1 traži promjenu rasporeda



Slika 52 Stanar2 traži promjenu rasporeda



Slika 53 Predstavnik radi zamjenu u rasporedu



Ažuriraj raspored:

Generiraj raspored Odobri izmjene

Ime	Prezime	Datum
Mate	Ivić	15-01-2018
Stanar4_ime	Stanar4_prezime	16-01-2018
Mate	Ivić	17-01-2018
Stanar3_ime	Stanar3_prezime	18-01-2018
Predstavnik1234_ime	Predstavnik1234_prezime	19-01-2018
Stanar2_ime	Stanar2_prezime	20-01-2018

Slika 54 Zamjena uspješno napravljena

Slučaj 8:

Opis: Unos novog naloga.

Očekivani rezultat: Novi nalog je uspješno unesen.

Dobiveni rezultat: Upravitelj je uspješno unio novi nalog za plaćanje.

Dodaj novi nalog

Iznos
Unesi iznos

Opis
Unesci opis

Datum prispjeća
DD-MM-YYYY

Datum plaćanja
DD MM YYYY

Platitelj
Odaberi

Primatelj
Odaberi

Potvrdi

Slika 55 Unos novog naloga

Platitelj	Primatelj	Iznos	Opis	Datum dospijeća	Dan plaćanja
Stanar2_ime Stanar2_prezime	Upravitelj5678_ime Upravitelj5678_prezime	152.5	Pričuva	01-02-2018	30-04-2018

Dodaj novi nalog Ažuriraj podatke

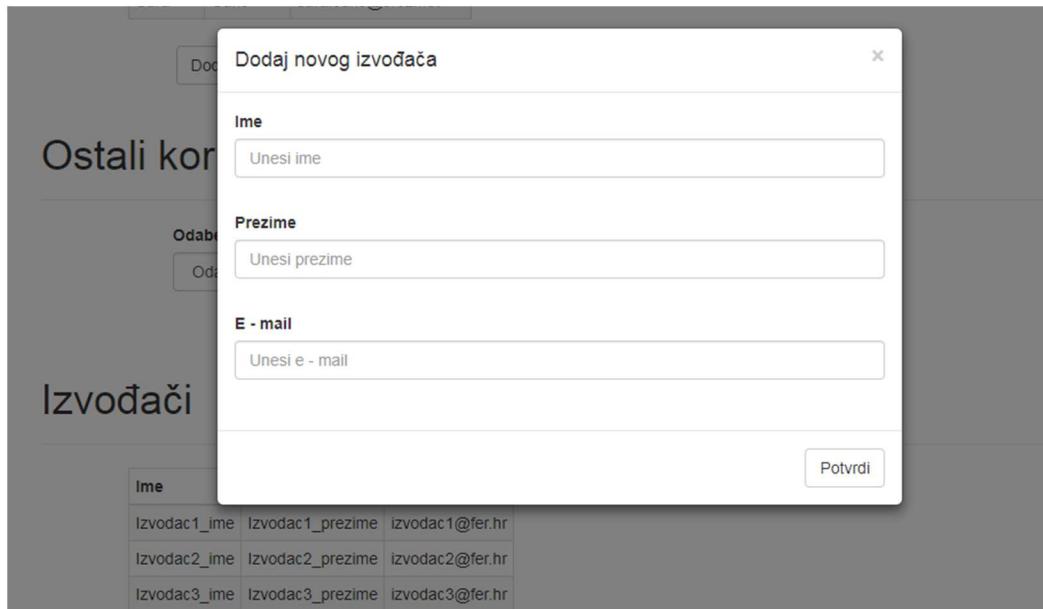
Slika 56 Novi nalog uspješno unesen

Slučaj 9:

Opis: Administrator dodaje novog izvođača.

Očekivani rezultat: Dodana je novi izvođač.

Dobiveni rezultat: Administrator unosi sve potrebne podatke, te je novi izvođač uspješno unesen.



Slika 57 Dodavanje novog izvođača

Izvođači

Ime	Prezime	E - Mail
Izvodac1_ime	Izvodac1_prezime	izvodac1@fer.hr
Izvodac2_ime	Izvodac2_prezime	izvodac2@fer.hr
Izvodac3_ime	Izvodac3_prezime	izvodac3@fer.hr
Izvodac4_ime	Izvodac4_prezime	izvodac4@fer.hr
Izvodac5_ime	Izvodac5_prezime	izvodac5@fer.hr
MAR	MAR	izvodac6@fer.hr

[Dodaj izvođača](#) [Obriši izvođača](#)

Slika 58 Novi izvođač uspješno dodan

Slučaj 10:

Opis: Unos troškova.

Očekivani rezultat: Novi trošak je unesen.

Dobiveni rezultat: Korisnik unosi ispravne podatke te je trošak uspješno unesen.

Slika 59 Unos novog troška

Trenutni iznos zajedničkog novca zgrade "Prva zgrada": 100000 kn

Kreator	Iznos	Opis	Datum kreiranja	Hitnost	Status
Stanar1_ime Stanar1_prezime	1500 kn	Nova garažna vrata	02-08-2016	Hitno ▲	Sredstva skupljena
Stanar2_ime Stanar2_prezime	1500 kn	Nova garažna vrata	02-08-2016	Hitno ▲	Sredstva skupljena
Stanar3_ime Stanar3_prezime	1500 kn	Nova garažna vrata	02-08-2016	Hitno ▲	Sredstva skupljena
Stanar4_ime Stanar4_prezime	1500 kn	Nova garažna vrata	02-08-2016	Hitno ▲	Sredstva skupljena
Predstavnik1234_ime Predstavnik1234_prezime	1500 kn	Nova garažna vrata	02-08-2016	Hitno ▲	Sredstva skupljena
Stanar1_ime Stanar1_prezime	30000 kn	Novo dizalo	14-01-2018	Nije hitno	Prikupljanje sredstava

[Dodaj novi trošak](#)

[Ažuriraj podatke](#)

Slika 60 Novi trošak uspješno unesen

Slučaj 11:

Opis: Administrator dodaje novog administratora.

Očekivani rezultat: Novi administrator je dodan.

Dobiveni rezultat: Administrator unosi sve potrebne podatke za novog administratora te potvrđuje te podatke i uspješno ga dodaje.

Ime	Prezime	E-mail
Ivan	Biškup	ivan.biska@fer.hr
Ivana	Mršić	ivana.mrsic@fer.hr
Luka	Hrgović	luka.hrgovic2@fer.hr
Marko	Cavalli	marko.cavalli@fer.hr
Marko	Čurlić	marko.curlic@fer.hr
Toni	Martinčić	toni.martincic@fer.hr
Hrvoje	Šimić	hrvoje.simic@croz.net
Katarina	Potkonjak	ksismanovic@croz.net
Sara	Ćurić	sara.curic@croz.net
Jure	Ivić	jure.ivic@fer.hr

Slika 61 Dodavanje novog administratora



Administratori

Ime	Prezime	E - Mail
Ivan	Biškup	ivan.biska@fer.hr
Ivana	Mršić	ivana.mrsic@fer.hr
Luka	Hrgović	luka.hrgovic2@fer.hr
Marko	Cavalli	marko.cavalli@fer.hr
Marko	Čurlić	marko.curlic@fer.hr
Toni	Martinčić	toni.martincic@fer.hr
Hrvoje	Šimić	hrvoje.simic@croz.net
Katarina	Potkonjak	ksismanovic@croz.net
Sara	Ćurić	sara.curic@croz.net
Jure	Ivić	jure.ivic@fer.hr

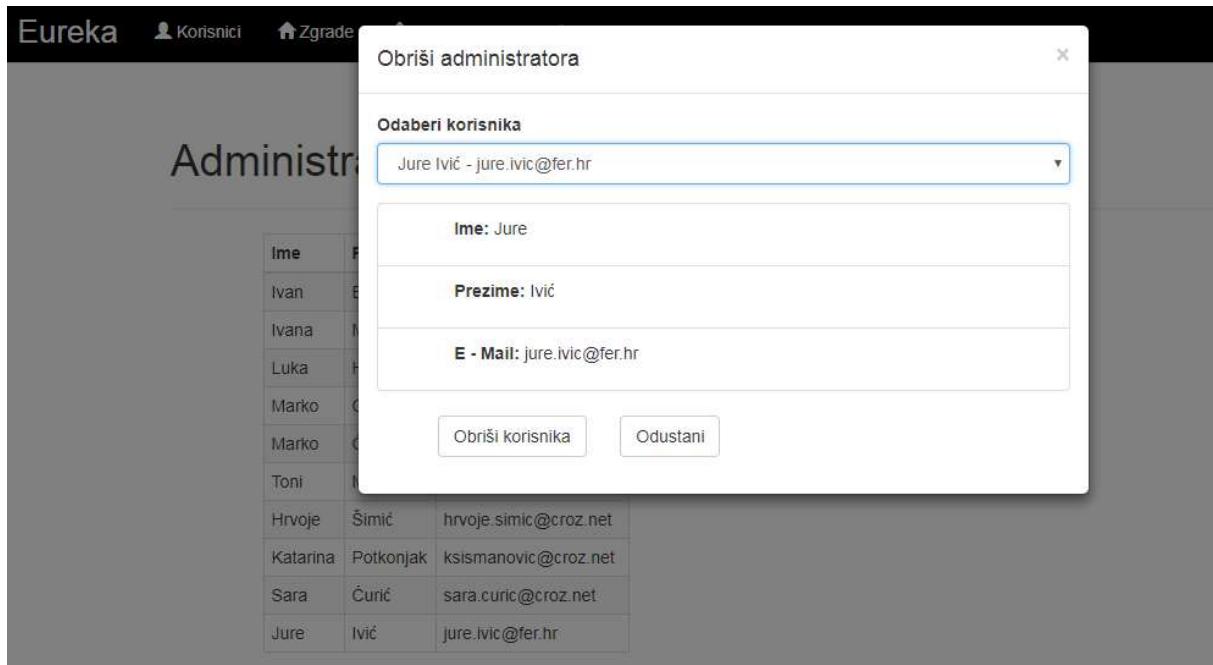
Slika 62 Uspješan unos novog administratora

Slučaj 12:

Opis: Obriši administratora.

Očekivani rezultat: Administrator je uspješno obrisan.

Dobiveni rezultat: Administrator odabire kojeg korisnika želi maknuti s liste administratora te ga briše.



Slika 63 Brisanje administratora



Administratori

Ime	Prezime	E - Mail
Ivan	Biškup	ivan.biska@fer.hr
Ivana	Miršić	ivana.mrsic@fer.hr
Luka	Hrgović	luka.hrgovic2@fer.hr
Marko	Cavalli	marko.cavalli@fer.hr
Marko	Čurlin	marko(curlin)@fer.hr
Toni	Martinčić	toni.martincic@fer.hr
Hrvoje	Šimić	hrvoje.simic@croz.net
Katarina	Potkonjak	ksismanovic@croz.net
Sara	Ćurić	sara.curic@croz.net

[Dodaj administratora](#) [Obriši administratora](#)

Slika 64 Uspješno obrisan administrator

Slučaj 13:

Opis: Dodavanje novog stana u zgradi.

Očekivani rezultat: Novi stan je uspješno dodan.

Dobiveni rezultat: Korisnik unosi samo kvadraturu stanu te je stan unesen potvrdom tog unosa.

Br.	Površina	Ime vlasnika	Prezime vlasnika	E - Mail vlasnika
1	25.23	Pero	Peric	pero.peric@fer.hr
2	25.23	Iva	Ivic	iva.ivic@fer.hr
3	25.23	Ivan	Horvat	ivan.horvat@fer.hr
4	25.23	Marko	Cavalli	marko.cavalli@fer.hr
5	56.5	Luka	Hrgović	luka.hrgovic2@fer.hr

Slika 65 Dodavanje novog stana

Br.	Površina	Ime vlasnika	Prezime vlasnika	E - Mail vlasnika
1	25.23	Pero	Peric	pero.peric@fer.hr
2	25.23	Iva	Ivic	iva.ivic@fer.hr
3	25.23	Ivan	Horvat	ivan.horvat@fer.hr
4	25.23	Marko	Cavalli	marko.cavalli@fer.hr
5	56.5	Luka	Hrgović	luka.hrgovic2@fer.hr
6	45.2			

Slika 66 Novi stan uspješno dodan

Slučaj 14:

Opis: Dodaj novog korisnika.

Očekivani rezultat: Novi korisnik je uspješno dodan.

Dobiveni rezultat: Administrator unosi sve potrebne podatke za novog korisnika, te ako je unio e-mail koji već postoji unos mu neće biti moguć te će biti upozoren da taj e-mail već postoji. U suprotnom korisnik će biti uspješno unesen.

The screenshot shows a modal dialog box titled "Dodaj novog korisnika". Inside the dialog, there are fields for "Ime" (Name) containing "Ivan", "Prezime" (Last Name) containing "Horvat", and "E - mail" (Email) containing "ivan.horvat@fer.hr". Below the email field, a red error message box displays the text "Unesena e - mail adresa već postoji." (The entered email address already exists.). There are dropdown menus for "Odaberi privilegiju" (Select privilege) set to "Stanar" and "Odaber jedan od praznih stanova:" (Select one of the empty rooms) set to "22". A blue "Potvrdi" (Confirm) button is at the bottom right.

Slika 67 Prikaz neuspješnog dodavanja novog korisnika

Ostali korisnici: (po zgradama)

Odaberi zgradu:

Ilica 256

Ime	Prezime	E - Mail	Privilegija
Upravitelj568_ime	Upravitelj568_prezime	upravitelj568@fer.hr	Upravitelj
Pero	Peric	pero.peric@fer.hr	Stanar
Iva	Ivic	iva.ivic@fer.hr	Stanar
Ivan	Horvat	ivan.horvat@fer.hr	Stanar
Marko	Cavalli	marko.cavalli@fer.hr	Administrator
Luka	Hrgovic	luka.hrgovic2@fer.hr	Administrator
Ivan	Horvat	ivan.horvat1@fer.hr	Stanar

Dodaj novog korisnika

Ažuriraj podatke

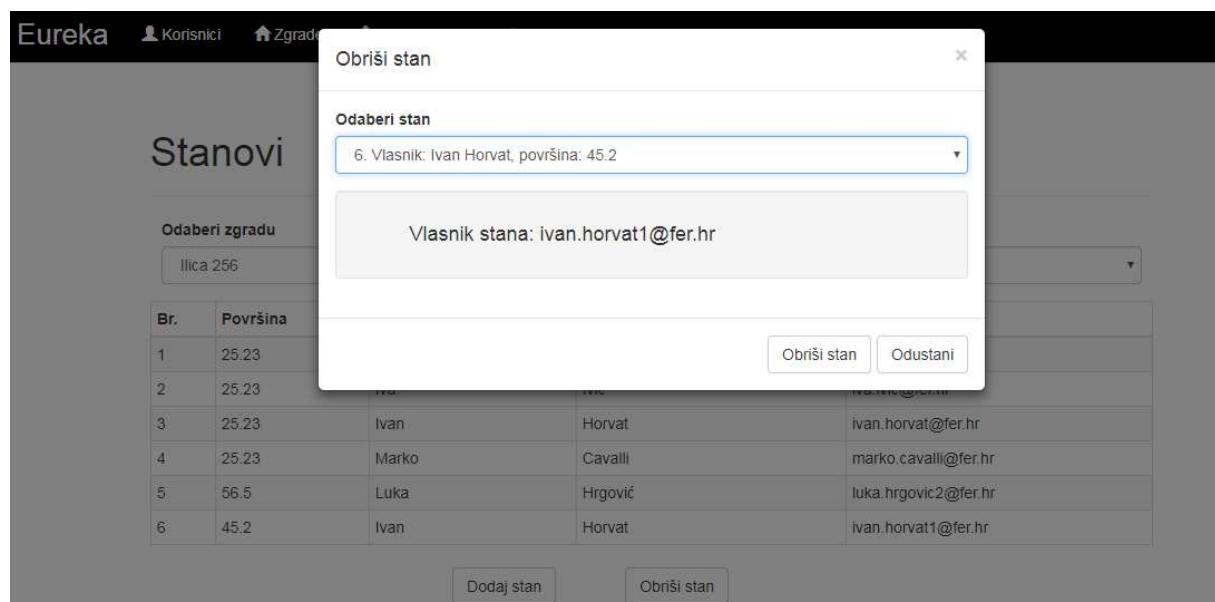
Slika 68 Korisnik je uspješno dodan

Slučaj 15:

Opis: Brisanje stana.

Očekivani rezultat: Stan je uspješno obrisan.

Dobiveni rezultat: Administrator odabire koji stan želi obrisati te potvrđuje odabir i briše stan.



Slika 69 Brisanje stana



Stanovi

Odaberite zgradu

Ulica 256

Br.	Površina	Ime vlasnika	Prezime vlasnika	E - Mail vlasnika
1	25.23	Pero	Perić	pero.peric@fer.hr
2	25.23	Iva	Ivić	iva.ivic@fer.hr
3	25.23	Ivan	Horvat	ivan.horvat@fer.hr
4	25.23	Marko	Cavalli	marko.cavalli@fer.hr
5	56.5	Luka	Hrgović	luka.hrgovic2@fer.hr

Dodaj stan

Obriši stan

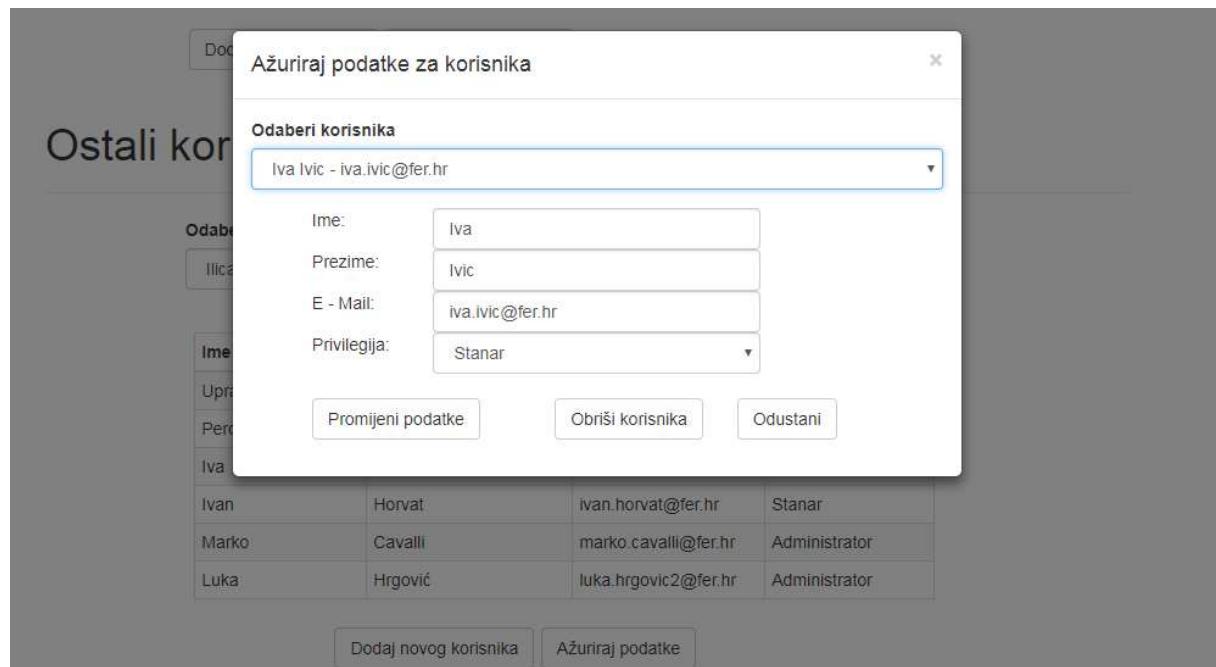
Slika 70 Stan je uspješno obrisan

Slučaj 16:

Opis: Brisanje korisnika.

Očekivani rezultat: Korisnik je uspješno obrisan.

Dobiveni rezultat: Administrator označuje koje korisnika želi obrisati te klikom na „Obriši korisnika“ korisnik biva obrisan.



Slika 71 Brisanje korisnika

Ostali korisnici: (po zgradama)

Odaberite zgradu:

Ime	Prezime	E - Mail	Privilegija
Upravitelj568_ime	Upravitelj568_prezime	upravitelj568@fer.hr	Upravitelj
Pero	Peric	pero.peric@fer.hr	Stanar
Ivan	Horvat	ivan.horvat@fer.hr	Stanar
Marko	Cavalli	marko.cavalli@fer.hr	Administrator
Luka	Hrgović	luka.hrgovic2@fer.hr	Administrator

[Dodaj novog korisnika](#)

[Ažuriraj podatke](#)

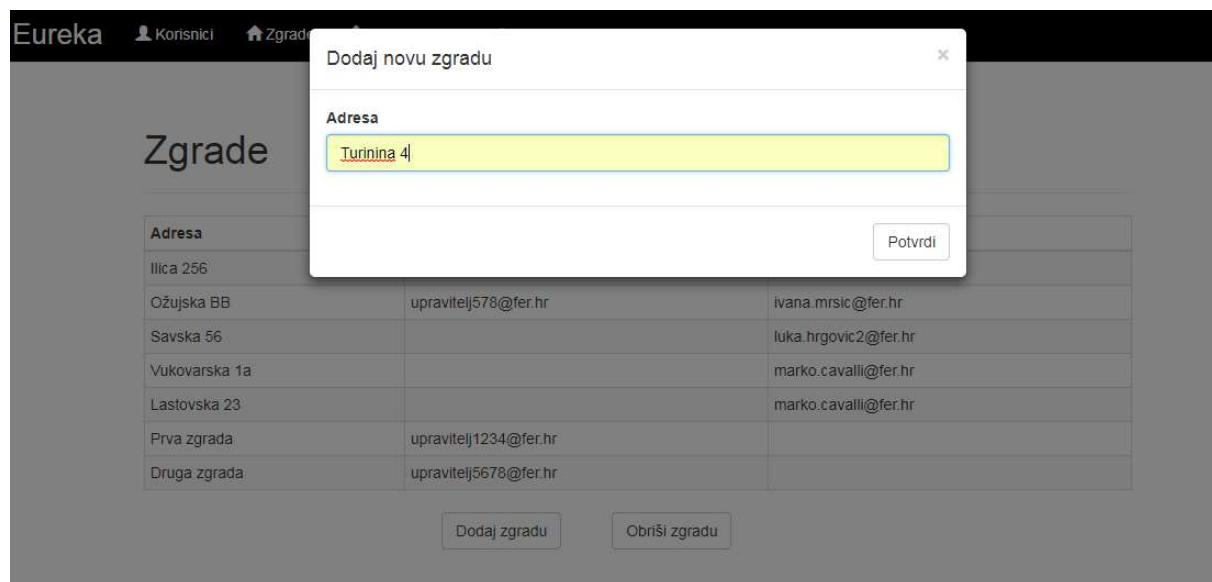
Slika 72 Korisnik je uspješno obrisan

Slučaj 17:

Opis: Dodaj novu zgradu.

Očekivani rezultat: Nova zgrada je uspješno dodana.

Dobiveni rezultat: Administrator upisuje adresu gdje se nova zgrada nalazi te je potvrdom unosi. Ukoliko je unesena zgrada sa istom adresom, unos mu je odbijen sa porukom.



Slika 73 Dodavanje nove zgrade



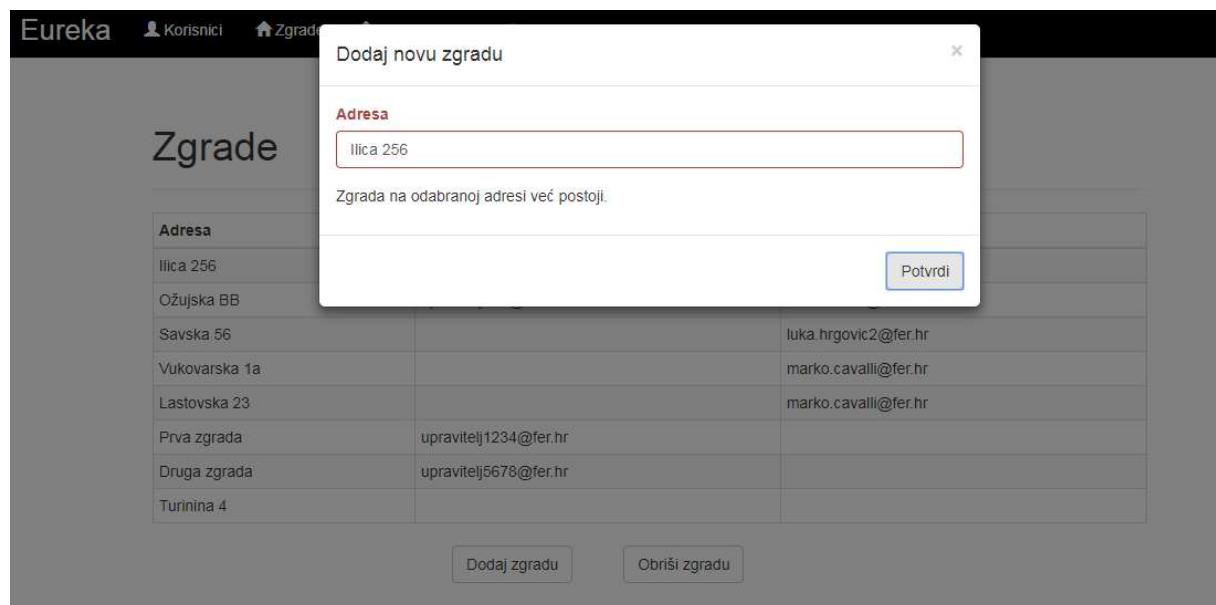
Zgrade

Adresa	Upravitelj	Predstavnik stanara
Ilica 256	upravitelj568@fer.hr	ivan.biska@fer.hr
Ožujska BB	upravitelj578@fer.hr	ivana.mrsic@fer.hr
Savska 56		luka.hrgovic2@fer.hr
Vukovarska 1a		marko.cavalli@fer.hr
Lastovska 23		marko.cavalli@fer.hr
Prva zgrada	upravitelj1234@fer.hr	
Druga zgrada	upravitelj5678@fer.hr	
Turinina 4		

Dodaj zgradu

Obriši zgradu

Slika 74 Uspješan unos zgrade



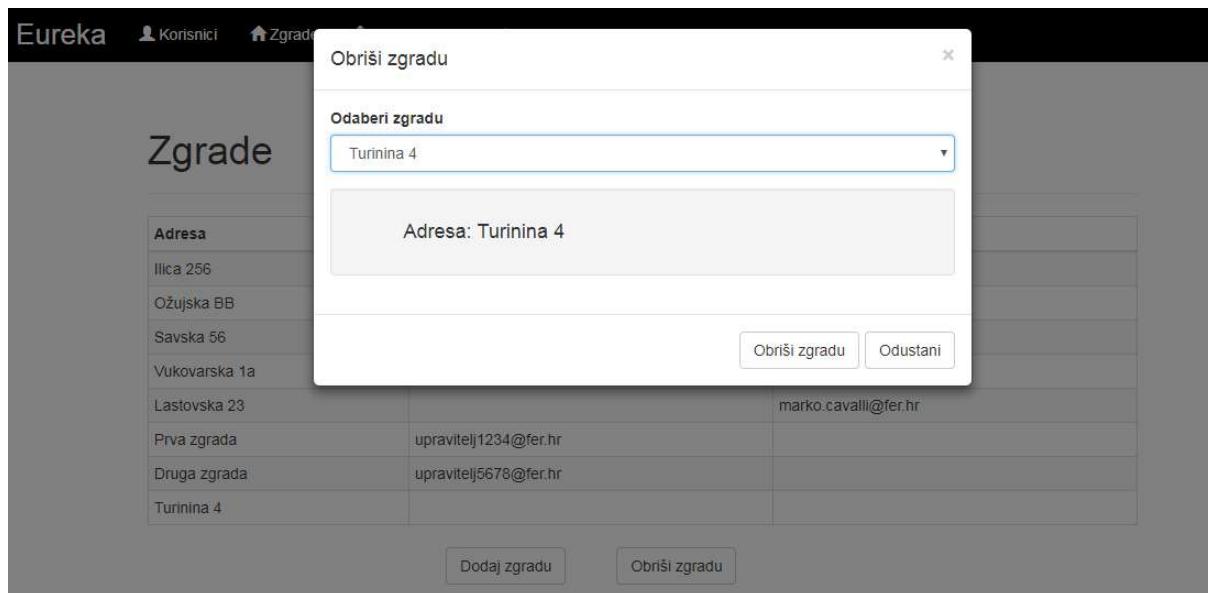
Slika 75 Pogršan unos zgrade

Slučaj 18:

Opis: Obriši zgradu.

Očekivani rezultat: Zgrada je uspješno izbrisana.

Dobiveni rezultat: Administrator odabire koju zgradu želi obrisati te potvrdom na „Obriši zgradu“ je briše.



Slika 76 Brisanje zgrade

Adresa	Upravitelj	Predstavnik stanara
Ilica 256	upravitelj568@fer.hr	ivan.biska@fer.hr
Ožujska BB	upravitelj578@fer.hr	ivana.mrsic@fer.hr
Savska 56		luka.hrgovic2@fer.hr
Vukovarska 1a		marko.cavallii@fer.hr
Lastovska 23		marko.cavallii@fer.hr
Prva zgrada	upravitelj1234@fer.hr	
Druga zgrada	upravitelj5678@fer.hr	
Turinina 4		

Slika 77 Uspješno obrisana zgrada

7.5. Upute za instalaciju

Web aplikaciju je moguće otvoriti pomoću linka: <https://eureka-building-management.herokuapp.com/>. Nakon toga korisnik se može ulogirati i ući u aplikaciju samo ako je unesen u bazu podataka od administratora, te se onda može prijaviti sa svojim korisničkim imenom i lozinkom koji su mu dodijeljeni od strane administratora.

Isto tako samu aplikaciju možete pokrenuti lokalno tako da skinete projekt sa Gitlaba, sa stranice: <https://gitlab.com/Toni222/Eureka.git> i pokrenete je u bilo kojem IDE programu, kao što je na primjer IntelliJ, te nakon toga otvorite web preglednik te u njemu upišete <https://localhost:3000/> i nakon toga vam se aplikacija otvara na stranici za prijavu.

7.6. Korisničke upute

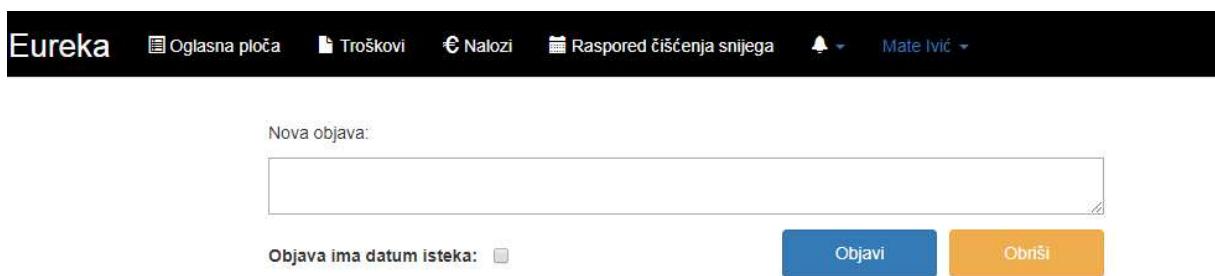
7.6.1. Upute stanaru, predstavniku i upravitelju

Ulaskom na stranicu aplikacije, korisniku se otvara stranica za prijavu, gdje korisnik unosi svoje ispravno korisničko ime i lozinku kako bi se uspješno prijavio.



Slika 78 Stranica za prijavu

Nakon uspješne prijave, korisniku se otvara početna stranica, odnosno oglasna ploča, gdje ako nešto želi objaviti ima mogućnost to i napraviti unosom željenog teksta i datuma isteka objave, te klikom na „Objavi“ na kraju svega. Isto tako korisnik nakon unosa, ima mogućnost i brisanja i uređivanja te objave pomoću alata pored objave.



Slika 79 Početna stranica(Oglasna ploča)

Nova objava:

Objava ima datum isteka: **Objavi****Obriši**

Izgubljeni ključevi	ALATI ZA UREĐIVANJE I BRISANJE
Stanar1_ime Stanar1_prezime	<input checked="" type="checkbox"/> <input type="button" value="Delete"/>
	14-01-2018

Slika 80 Prikaz objave i alata za uređivanje iste

Na drugoj stranici, naziva „Troškovi“, korisnik može unijeti neke štete koje su se dogodile na zgradi i određene kvarove ili neke planirane troškove u budućnosti klikom na „Dodaj novi trošak“. Isto tako ih može uređivati i ažurirati klikom na „Ažuriraj podatke“.

Eureka Oglasna ploča Troškovi Nalozi Raspored čišćenja snijega Mate Ivić ▾

Trenutni iznos zajedničkog novca zgrade "Prva zgrada": 100000 kn

Kreator	Iznos	Opis	Datum kreiranja	Hitnost	Status
Mate Ivić	1500 kn	Nova garažna vrata	02-08-2016	Hitno	Sredstva skupljena
Stanar2_ime Stanar2_prezime	1500 kn	Nova garažna vrata	02-08-2016	Hitno	Sredstva skupljena
Stanar3_ime Stanar3_prezime	1500 kn	Nova garažna vrata	02-08-2016	Hitno	Sredstva skupljena
Stanar4_ime Stanar4_prezime	1500 kn	Nova garažna vrata	02-08-2016	Hitno	Sredstva skupljena

Dodaj novi trošak **Ažuriraj podatke**

Slika 81 Stranica „Troškovi“

Treća stranica, „Nalozi“, prikazuje stanarove naloge koje ima za platiti, i prikazuje zgradine naloge koji se trebaju platiti. Nove naloge klikom na „Dodaj novi nalog“ može dodavati samo Upravitelj, koji ih jedini može i ažurirati isto tako klikom na „Ažuriraj podatke“.

Platitelj	Primatelj	Iznos	Opis	Datum dospijeća	Dan plaćanja
Stanar2_ime Stanar2_prezime	Upravitelj5678_ime Upravitelj5678_prezime	152.5	Pričuva	01-02-2018	30-04-2018
Upravitelj5678_ime Upravitelj5678_prezime	Izvodac2_ime Izvodac2_prezime	1984.42	Popravak dizala	28-02-2018	30-06-2018

[Dodaj novi nalog](#) [Ažuriraj podatke](#)

Slika 82 Stranica „Nalozi“

I na posljednjoj stranici, „Raspored čišćenja snijega“, pristup imaju samo stanar i predstavnik. Predstavnik ima opciju za generiranje rasporeda klikom na „Generiranje rasporeda“ nakon čega odabire od kojeg do kojeg datuma želi generirati raspored, te isto tako ima opciju za odobravanje izmjene kada mu stignu više od jednog zahtjeva stanara za izmjenu rasporeda. Stanar s druge strane ima opciju samo odabrati svoj datum koji želi zamijeniti u rasporedu te to potvrditi klikom na „Potvrđi“, te čekati da mu predstavnik odobri zamjenu sa nekim drugim stanarom.

The screenshot shows a user interface for managing snow cleaning schedules. At the top, there's a navigation bar with links like 'Oglasna ploča', 'Troškovi', 'Nalozi', 'Raspored čišćenja snijega', and a notification bell. Below the navigation is a search bar with fields for 'Ime', 'Prezime', and 'Datum'. To the right, there's a modal window titled 'Zatraži promjenu rasporeda za datum' with a dropdown menu for selecting a date ('Odaberi') and a 'Potvrdi' button. Another modal window below it is titled 'Ažuriraj raspored:' with buttons for 'Generiraj raspored' and 'Odobri izmjene'.

Slika 83 Stranica "Raspored čišćenja snijega"

The screenshot shows a table of scheduled tasks and a date selection modal. The table has columns for 'Ime', 'Prezime', and 'Datum'. The tasks listed are: Mate Ivić (15-01-2018), Stanar4_ime Stanar4_prezime (16-01-2018), Stanar2_ime Stanar2_prezime (17-01-2018), Stanar3_ime Stanar3_prezime (18-01-2018), Predstavnik1234_ime Predstavnik1234_prezime (19-01-2018), Mate Ivić (20-01-2018), and Stanar4_ime Stanar4_prezime (21-01-2018). To the right, a modal window titled 'Zatraži promjenu rasporeda za datum' shows the date '2018-01-20' selected in a dropdown and a 'Potvrdi' button.

Ime	Prezime	Datum
Mate	Ivić	15-01-2018
Stanar4_ime	Stanar4_prezime	16-01-2018
Stanar2_ime	Stanar2_prezime	17-01-2018
Stanar3_ime	Stanar3_prezime	18-01-2018
Predstavnik1234_ime	Predstavnik1234_prezime	19-01-2018
Mate	Ivić	20-01-2018
Stanar4_ime	Stanar4_prezime	21-01-2018

Slika 84 Prikaz nakon generiranja rasporeda

7.6.2. Upute administratoru

Ulaskom na stranicu aplikacije, administratoru se otvara stranica za prijavu, gdje on unosi svoje ispravno korisničko ime i lozinku kako bi se uspješno prijavio.



Slika 85 Stranica za prijavu

Uspješnom prijavom administratoru se otvara početna stranica „Korisnici“ gdje se nalaze svi administratori, kao i stanari po zgradama i izvođači radova. Administrator ima mogućnosti dodavanja i brisanja administratora klikom na „Dodavanje administratora“ ili „Brisanje administratora“.

Administratori

Ime	Prezime	E - Mail
Ivan	Biškup	ivan.biska@fer.hr
Ivana	Mršić	ivana.mrsic@fer.hr
Luka	Hrgović	luka.hrgovic2@fer.hr
Marko	Cavalli	marko.cavalli@fer.hr
Marko	Ćurlin	marko(curlin)@fer.hr
Toni	Martinčić	toni.martincic@fer.hr
Hrvoje	Šimić	hrvoje.simic@croz.net
Katarina	Potkonjak	ksismanovic@croz.net
Sara	Ćurić	sara.curic@croz.net

Dodaj administratora Obriši administratora

Slika 86 Prikaz administratora

Također na istoj stranici administrator ima mogućnost da odabere zgradu, te nakon toga može odabrati opciju „Dodaj novog korisnika“ te unos svih potrebnih podataka ili opciju „Ažuriraj podatke“ te time promijeniti neke podatke pojedinog korisnika ili obrisati korisnika i isto tako odrediti privilegiju korisniku.

Ostali korisnici: (po zgradama)

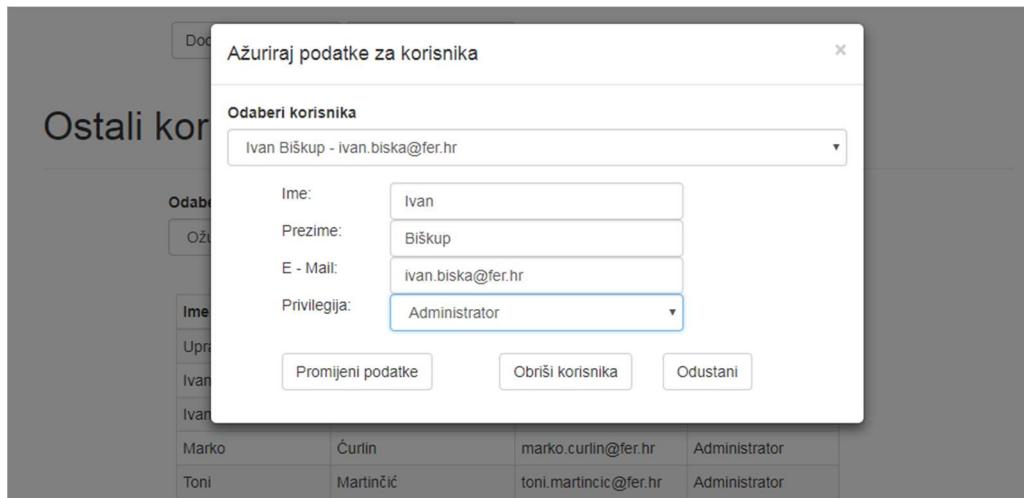
Odaberi zgradu:

Ožujska BB

Ime	Prezime	E - Mail	Privilegija
Upravitelj578_ime	Upravitelj578_prezime	upravitelj578@fer.hr	Upravitelj
Ivana	Mršić	ivana.mrsic@fer.hr	Administrator
Ivan	Biškup	ivan.biska@fer.hr	Administrator
Marko	Čurlin	marko(curlin@fer.hr	Administrator
Toni	Martinčić	toni.martincic@fer.hr	Administrator

Dodaj novog korisnika
Ažuriraj podatke

Slika 87 Prikaz korisnika po zgradama



Slika 88 Ažuriranje podataka korisnika

Kao i u prošla dva primjera, na istoj stranici administrator još ima mogućnost dodavanja i brisanja izvođača radova, pomoću opcija „Dodaj izvođača“ i „Obriši izvođača“.

Izvođači

Ime	Prezime	E - Mail
Izvodac1_ime	Izvodac1_prezime	izvodac1@fer.hr
Izvodac2_ime	Izvodac2_prezime	izvodac2@fer.hr
Izvodac3_ime	Izvodac3_prezime	izvodac3@fer.hr
Izvodac4_ime	Izvodac4_prezime	izvodac4@fer.hr
Izvodac5_ime	Izvodac5_prezime	izvodac5@fer.hr

[Dodaj izvođača](#) [Obriši izvođača](#)

Slika 89 Prikaz izvođača

Na drugoj stranici se nalaze podatci o zgradama i administrator ima mogućnost dodati i obrisati zgradu pomoću opcija „Dodaj zgradu“ i „Obriši zgradu“.

Zgrade

Adresa	Upravitelj	Predstavnik stanara
Ilica 256	upravitelj568@fer.hr	ivan.biska@fer.hr
Ožujska BB	upravitelj578@fer.hr	ivana.mrsic@fer.hr
Savska 56		luka.hrgovic2@fer.hr
Vukovarska 1a		marko.cavalli@fer.hr
Lastovska 23		marko.cavalli@fer.hr
Prva zgrada	upravitejj1234@fer.hr	
Druga zgrada	upravitejj5678@fer.hr	

[Dodaj zgradu](#) [Obriši zgradu](#)

Slika 90 Podaci o zgradama

Te na zadnjoj stranici, „Stanovi“, administrator prvo odabere zgradu koju želi, te nakon toga ima mogućnost dodati stan sa opcijom „Dodaj stan“ ili obrisati stan opcijom „Obriši stan“.

Stanovi

Odaberite zgradu				
Ilica 256				
1	25.23	Pero	Peric	pero.peric@fer.hr
2	25.23	Iva	Ivic	iva.ivic@fer.hr
3	25.23	Ivan	Horvat	ivan.horvat@fer.hr
4	25.23	Marko	Cavalli	marko.cavalli@fer.hr
5	56.5	Luka	Hrgović	luka.hrgovic2@fer.hr

[Dodaj stan](#) [Obriši stan](#)

Slika 91 Podatci o stanovima

Treba napomenuti da administrator kod dodavanja stanova samo unosi površinu, a vlasnik stana se određuje tek kod unosa novog korisnika za tu zgradu na stranici „Korisnici“. Isto tako kod dodavanja nove zgrade, unosi se samo adresa zgrade, dok upravitelja i predstavnika se određuje kod korisnika na stranici „Korisnici“, gdje se samom stanaru, odnosno upravitelju određuje privilegija koju ima u toj zgradi.

8. Zaključak i budući rad

Zadatak ovog projekta je bio razviti web aplikaciju koja će omogućiti jednostavniju i lakšu međusobnu komunikaciju svih stanara, te svih ostalih zainteresiranih sudionika, poput upravitelja zgrade. Također je bio cilj omogućiti funkcionalnosti poput objavljivanja na oglasnoj ploči, pregled troškova i mogućnost unosa novih, pregled naloga te rasporeda čišćenja snijega i još mnogih drugih. Izrada samog projekta radila se u dvije faze.

Prva faza izrade sastojala se od upoznavanja samog tima, te analiziranja projektnog zadatka na samom početku. Drugi korak u prvoj fazi je bio određivanje funkcionalnih i nefunkcionalnih zahtjeva. Nakon uspješno definiranih zahtjeva krenulo se u izradu obrazaca uporabe i sekvensijski dijagrama, te definiranja arhitekture sustava i entiteta baze podataka. Zatim se krenulo sa izradom same implementacije, odnosno baze podataka i backenda. Završni dio prve faze je bilo odrediti dijagrame razreda i objekata. Time je u prvom dijelu napravljen velik dio grafičkog prikaza samog zadatka i bitan dio za samu izradu implementacije sustava.

Druga faza je bila nešto kraća vremenski i nešto jednostavnija od prve faze što se tiče izrade same dokumentacije, jer je bitniji i teži dio dokumentacije obavljen u prvoj fazi, te je druga faza bila finalizacija projekta. Druga faza je započela izradom ostalih dijagrama, kao što su dijagrami stanja, aktivnosti, razmještaja i komponenti, te komunikacijski dijagrami. Paralelno s time radila se izrada frontenda i dovršavao se backend. Nakon što je implementacija bila većim dijelom gotova, moglo se obaviti opisivanje isječaka programskih kodova vezanih za temeljnu funkcionalnost, te krenuti u ispitivanje samih programskih rješenja na aplikaciji. I kao završni dio dokumentacije je bilo napisati upute za instalaciju same aplikacije te detaljne upute za korisnike iste. Cjelokupno znanje stečeno na predmetu pridonijelo je lakšoj izradi dokumentacije, te uspješnoj realizaciji same aplikacije.

Komunikacija između članova najviše se vodila preko društvenih mreža (Facebook), što je uvelike pridonijelo razvoju aplikacije i samom dogovoru oko posla oko same izrade. Također se prilikom izrade koristio sustav Git za upravljanje raznim verzijama koda i dokumentacije.

Kao moguća proširenja i poboljšanje same aplikacije vidimo moguću izradu podrške za korištenje aplikacije preko mobilnih uređaja, te poboljšanje nekih dijelova same aplikacije koje bi se odnosile na neke sitne promjene u izvedbi određenih dijelova aplikacije.

Kao krajnji zaključak možemo reći da je ovo bilo jedno poučno iskustvo. Najzahtjevniji dio samog projekta je bilo određivanje funkcionalnih zahtjeva sustava te izrada obrazaca uporabe, jer se time uvelike određivao izgled i implementacija same aplikacije. Projekt je u konačnici uspješno priveden kraju i sve je uspješno završeno i zasigurno će nam naučene vještine pomoći u dalnjem studiranju i životu.

9. Popis literature

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
2. Oblikovanje programske potpore, FER ZEMRIS, <https://moodle.fer.hr/>
3. I. Sommerville, „Software engineering“, 8th ed, Addison Wesley, 2007.
4. Spring Boot tutorijal, <https://spring.io/guides/gs/spring-boot/>
5. Spring Boot tutorijal, <http://www.mkyong.com/tutorials/spring-boot-tutorials/>
6. React tutorijal, <https://reactjs.org/tutorial/tutorial.html>
7. React-Bootstrap, <https://react-bootstrap.github.io/>
8. Redux, <http://redux.js.org/docs/basics/>
9. Code School, <https://www.codeschool.com/>
10. Stackoverflow, <https://stackoverflow.com/>

Dodatak A: Indeks (slika, dijagrami, tablica, ispis koda)

Tablica 1 Dnevnik promjena dokumentacije	5
Slika 1 Funkcionalnosti aplikacije	16
Slika 2 Sekvencijski dijagram obrasca PrijavljanjeKorisnika	17
Slika 3 Sekvencijski dijagram obrasca PogledajOglasnuPloču	18
Slika 4 Sekvencijski dijagram obrasca DodajNaOglasnuPloču.....	19
Slika 5 Sekvencijski dijagram obrasca BrišiSaOglasnePloče	20
Slika 6 Sekvencijski dijagram obrasca PregledajRasporedČišćenjaSnijega.....	21
Slika 7 Sekvencijski dijagram obrasca ZatražioZamjenuTerminaRasporedaČišćenjaSnijega	22
Slika 8 Sekvencijski dijagram obrasca OdgovorNaZamjenuTerminaRasporedaČišćenjaSnijega.....	23
Slika 9 Sekvencijski dijagram obrasca PregledajOpćeInformacije.....	24
Slika 10 Sekvencijski dijagram obrasca IzmjeniOpćeInforamcije	25
Slika 11 Sekvencijski dijagram obrasca PogledajNaloge	26
Slika 12 Sekvencijski dijagram obrasca AžurirajNaloge	27
Slika 13 Sekvencijski dijagram obrasca DodajNaloge	28
Slika 14 Sekvencijski dijagram obrasca PrijaviŠtetu	29
Slika 15 Sekvencijski dijagram obrasca OdjavljivanjeKorisnika	30
Slika 16 Sekvencijski dijagram obrasca DodavanjeKorisnika.....	31
Slika 17 Sekvencijski dijagram obrasca BrisanjeKorisnika	32
Slika 18 Sekvencijski dijagram obrasca UrediObjavuNaOglasnojPloči	33
Slika 19 Skica sustava	35
Slika 20 ER model baze podataka.....	38
Slika 21 Dijagram razreda paketa domain.....	41
Slika 22 Dijagram razreda paketa service.....	43
Slika 23 Dijagram razreda paketa controller	46
Slika 24 Dijagram objekata stvarnog stanja sustava	47
Slika 25 Dijagram Aktivnosti-Prijava Korisnika	48
Slika 26 Dijagram Aktivnosti - Prikaz liste troškova.....	49
Slika 27 Dijagram Aktivnosti - Dodavanje nove objave	50
Slika 28 Dijagram Stanja - Korisnički prikaz	51

Slika 29 Komunikacijski dijagram - Brisanje objave.....	52
Slika 30 Komunikacijski dijagram - Prikaz korisnika lokacije(zgrade).....	53
Slika 31 Dijagram komponenti	54
Slika 32 Dijagram komponenti - Objave	54
Slika 33 Dijagram Razmještaja.....	55
Slika 34 UserRepository.....	57
Slika 35 UserService.....	58
Slika 36 UserController.....	59
Slika 37 UserDomain.....	60
Slika 38 Pogrešna prijava	61
Slika 39 Uspješna prijava	61
Slika 40 Neuspješan unos nove objave	62
Slika 41 Uspješan unos objave	62
Slika 42 Brisanje objave	63
Slika 43 Objava obrisana	63
Slika 44 Pregled informacija-prvi korak.....	63
Slika 44 Pregled informacija	64
Slika 45 Uređivanje postavki	64
Slika 46 Promjena osobnih podataka	65
Slika 47 Promjena lozinke.....	65
Slika 48 Uspješna promjena osobnih podataka	66
Slika 49 Generiranje rasporeda čišćenja snijega	66
Slika 50 Uspješno generiran raspored.....	67
Slika 51 Stanar1 traži promjenu rasporeda	67
Slika 52 Stanar2 traži promjenu rasporeda	68
Slika 53 Predstavnik radi zamjenu u rasporedu	68
Slika 54 Zamjena uspješno napravljena	68
Slika 55 Unos novog naloga.....	69
Slika 56 Novi nalog uspješno unesen	69
Slika 57 Dodavanje novog izvođača	70
Slika 58 Novi izvođač uspješno dodan.....	70

Slika 59 Unos novog troška	71
Slika 60 Novi trošak uspješno unesen	71
Slika 61 Dodavanje novog administratora.....	72
Slika 62 Uspješan unos novog administratora	72
Slika 63 Brisanje administratora.....	73
Slika 64 Uspješno obrisan administrator.....	73
Slika 65 Dodavanje novog stana.....	74
Slika 66 Novi stan uspješno dodan	74
Slika 67 Prikaz neuspješnog dodavanja novog korisnika.....	75
Slika 68 Korisnik je uspješno dodan	76
Slika 69 Brisanje stana	76
Slika 70 Stan je uspješno obrisan	77
Slika 71 Brisanje korisnika	77
Slika 72 Korisnik je uspješno obrisan.....	78
Slika 73 Dodavanje nove zgrade	78
Slika 74 Uspješan unos zgrade	79
Slika 75 Pogršan unos zgrade	79
Slika 76 Brisanje zgrade	80
Slika 77 Uspješno obrisana zgrada	80
Slika 78 Stranica za prijavu	82
Slika 79 Početna stranica(Oglasna ploča)	82
Slika 80 Prikaz objave i alata za uređivanje iste	83
Slika 81 Stranica „Troškovi“.....	83
Slika 82 Stranica „Nalozi“	84
Slika 83 Stranica "Raspored čišćenja snijega"	84
Slika 84 Prikaz nakon generiranja rasporeda	84
Slika 85 Stranica za prijavu	85
Slika 86 Prikaz administratora	85
Slika 87 Prikaz korisnika po zgradama.....	86
Slika 88 Ažuriranje podataka korisnika.....	86
Slika 89 Prikaz izvođača	87

Slika 90 Podatci o zgradama.....	87
Slika 91 Podatci o stanovima.....	88
Tablica 2 Članovi grupe (abecednim redom) -aktivnost.....	99
Slika 92 Prikaz aktivnosti	99

Dodatak B: Dnevnik sastajanja

1.sastanak – 18.10.2017.

Prisutni: Ivan Biškup, Marko Cavalli, Marko Ćurlin, Luka Hrgović, Toni Martinčić, Ivana Mršić

Sažetak: Instaliravanje svih potrebnih tehnologija za rad na projektu. Kloniranje projekta s Gitlaba u lokalni repozitorij.

2. sastanak – 21.10.2017.

Prisutni: Ivan Biškup, Marko Cavalli, Marko Ćurlin, Luka Hrgović, Toni Martinčić, Ivana Mršić

Sažetak: Učenje korištenja Gitlaba, Spring Boota. Rasprava i proučavanje projektnog zadatka. Početak rada na bazi podataka i backendu.

3. sastanak – 28.10.2017.

Prisutni: Ivan Biškup, Marko Cavalli, Marko Ćurlin, Luka Hrgović, Toni Martinčić, Ivana Mršić

Sažetak: Učenje Reacta i Reduxa. Rad na backendu, početak rada na frontendu i početak pisanja dokumentacije.

4. sastanak – 4.11.2017.

Prisutni: Ivan Biškup, Marko Cavalli, Marko Ćurlin, Luka Hrgović, Toni Martinčić, Ivana Mršić

Sažetak: Rad na dokumentaciji, obrascima uporaba i sekvencijskim dijagramima.

5. sastanak – 11.11.2017.

Prisutni: Ivan Biškup, Marko Cavalli, Marko Ćurlin, Luka Hrgović, Toni Martinčić, Ivana Mršić

Sažetak: Rad na dokumentaciji, ostali zahtjevi, svrha, opći prioriteti, skica sustava, dijagram razreda s opisom, dijagram objekata.

6. sastanak – 15.11.2017.

Prisutni: Ivan Biškup, Marko Cavalli, Marko Ćurlin, Luka Hrgović, Toni Martinčić, Ivana Mršić

Sažetak: Dovršetak izrade dokumentacije, dodavanje dodataka, te zaključak i daljni rad na projektu.

7. sastanak – 10.1.2018.

Prisutni: Ivan Biškup, Marko Cavalli, Marko Ćurlin, Luka Hrgović, Toni Martinčić, Ivana Mršić

Sažetak: Prezentacija trenutnog stanja aplikacije asistentu.

8. sastanak – 13.1.2018.

Prisutni: Ivan Biškup, Marko Cavalli, Marko Ćurlin, Luka Hrgović, Toni Martinčić, Ivana Mršić

Sažetak: Dogovor oko završnih poslova na dokumentaciji i aplikaciji.

Dodatak C: Prikaz aktivnosti grupe

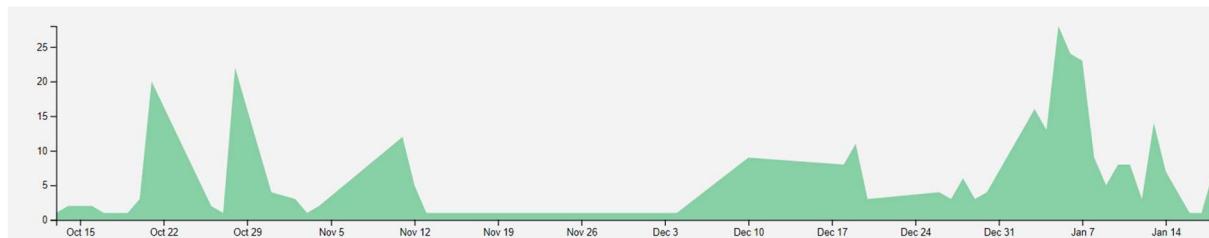
Popis aktivnosti	Toni Martinčić	Ivan Biškup	Marko Cavalli	Marko Ćurlin	Luka Hrgović	Ivana Mršić	
Upravljanje projektom	50%					50%	
Opis projektnog zadatka	50%					50%	
Rječnik pojmoveva	20%		80%				
Opis funkcionalnih zahtjeva					50%	50%	
Opis ostalih zahtjeva		50%		50%			
Arhitektura i dizajn sustava							
Svrha, opći prioriteti i skica sustava	50%	50%					
Dijagram razreda s opisom							
Dijagram objekata					100%		
Ostali UML dijagrami			50%		50%		
Implementacija i korisničko sučelje							
Dijagram razmještaja				100%			
Korištene tehnologije i alati						100%	
Isječak programskog kôda		100%					
Ispitivanje programskog rješenja		25%		50%	25%		
Upute za instalaciju	100%						
Korisničke upute			50%			50%	
Plan rada							
Pregled rada i stanje	10%	50%	10%	10%	10%	10%	

ostvarenja									
Zaključak i budući rad	16.66%	16.66%	16.66%	16.66%	16.66%	16.66%	16.66%	16.66%	
Popis literature	50%	10%	10%	10%	10%	10%	10%	10%	
Dodaci									
Indeks							100%		
Dnevnik sastajanja	50%	50%							

Tablica 2 Članovi grupe (abecednim redom) -aktivnost

October 13, 2017 – January 18, 2018

Commits to master, excluding merge commits. Limited to 6,000 commits.



Slika 92 Prikaz aktivnosti

Dodatak D: Pregled rada i stanje ostvarenja

U rev. 1 smo ostvarili u dokumentaciji:

- opis projektnog zadatka,
- funkcionalne zahtjeve i obrasce uporabe,
- ostale zahtjeve
- arhitekturu i dizajn sustava i
- dijagrame razreda, objekata, te sekvencijske dijagrame.

Također je napravljen dio implementacije koji se odnosi većim dijelom na backend, a jednim manjim dijelom na frontend.

U rev. 2 smo ostvarili u dokumentaciji:

- ostale UML dijagrame,
- dijagram razmještaja,
- isječke programskog koda vezanog za temeljne funkcionalnosti
- ispitivanje programskog rješenja
- upute za instalaciju i korisničke upute
- zaključak
- popis literature

Također je napravljena cijela implementacija do kraja.