Oblikovanje programske potpore

Ak. God. 2017./2018.

*Upravljanje stambenom zgradom*

Dokumentacija, Rev. 1

Grupa: *Eureka*

Voditelj: *Toni Martinčić*

Datum predaje:

Mentor: *Katarina Potkonjak*

Popis članova grupe i zadatci:

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

Ivana Mršić – arhitektura sustava, UML dijagrami, baza podataka

Marko Ćurlin – arhitektura sustava, UML dijagrami, baza podataka

Marko Cavalli – arhitektura sustava, UML dijagrami, baza podataka

Luka Hrgović – arhitektura sustava, UML dijagrami, baza podataka

Ivan Biškup – arhitektura sustava, UML dijagrami, baza podataka

Sadržaj

# 1. Dnevnik promjena dokumentacije

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev.** | **Opis promjene/dodatka** | **Autor(i)** | **Datum** |
| 0.1 | Napravljen predložak.  Dodan opis projektnog zadatka, popis članova i zadatci. | Biškup | 30.10.2017. |
| 0.2 | Dodan pojmovnik. | Cavalli | 30.10.2017. |
| 0.3 | Dodani funkcionalni zahtjevi. | Mršić,  Hrgović | 30.10.2017. |
| 0.4 | Dodan popis literature. | Martinčić | 31.10.2017. |
| 0.5 | Opis obrazaca uporabe. | Biškup, Ćurlin | 4.11.2017. |
| 0.6 |  |  |  |
| 0.7 |  |  |  |
| 0.8 |  |  |  |
| 0.9 |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  | . |

**2. Opis projektnog zadatka**

Cilj projekta je praktično primjeniti postupke oblikovanja programske potpore na rješavanje konkretnih problema u stambenim zgradama. Također izraditi projektnu dokumentaciju i implemanticiju 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 se želi postići vođenje obveza stanara kao što su uplata pričuve, raspored čišćenja snijega i objave izgubljenih ili pronađenih stvari. Isto tako bi se omogućila transparentnost svih aspekata upravljanja zgradom.

Predviđa se izrada web aplikacije koja bi se sastojala od početne stranice gdje bi se korisnici, odnosno stanari, prijavljivali pomoću svojih korisničkih imena i lozinki. Nakon prijave bi se korisniku otvorio ekran sa oglasnom pločom gdje bi mogao vidjeti sve obavijesti. Isto tako bi mogao otvoriti ekran sa nalazoma gdje ... . I zadnji ekran će predstavljati sve inforamcije o samoj aplikaciiji i laksem korištenju iste.

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

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

1. *Predstavnik stanara* – osoba koja vodi brigu o cijelokupnoj zgradi i potvrđuje radne naloge.

2. *Upravitelj zgrade*  - ovlaštena firma koja vodi brigu o financijama zgrade i slanju osoba za određene popravke na samoj zgradi.

3. *Administrator* – osoba koja se brine o samoj stranici i promjenama na njoj. Isto tako ima mogućnost dodavanja novih stanara i postavljanja neke unaprijed određene osobe za predstavnika.

**3. Pojmovnik**

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

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

**Git**: distribuirani sustav za upravljanje izvronim kodom.

**Hibernate**: objektno/relacijska tehnologija koja povezuje objekte iz Jave s realacijskim 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
* Ulogirava 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 i troškove
* Objavljuje na oglasnoj ploči
* Briše vlastite objave
* Predstavnik stanara – inicijator
* Ulogirava se lozinkom i korisničkim imenom
* Prijavljuje štetu, kvar i izgubljene/pronađene stvari
* Određuje važnost i redoslijed popravaka i radova
* Generira naloge i uplate
* Zatražuje promjenu termina čišćenja snijega po potrebi
* Objavljuje raspored čišćenja snijega
* Uvid u raspored čišćenja snijega, naloge i troškove
* Objavljuje na oglasnoj ploči
* Briše objave po potrebi
* Upravitelj zgrade – inicijator
* Ulogirava se lozinkom i korisničkim imenom
* Obrađuje naloge i financije
* Dogovara posao sa izvođačima radova
* Administrator – inicijator
* Upisuje novog korisnika (stanara) u bazu
* Daje ovlasti predstavniku stanara
* Daje ovlasti upravitelju zgrade
* Održava stranicu
* Baza podataka – sudionik
* Čuva podatke o svim stanarima, troškovima i nalozima
* Čuva aktivnosti na oglasnoj ploči i raspored čišćenja snijega
* Čuva podatke o svim zgradam i stanovima

**Opis obrazaca uporabe:**

* **UC1** – PrijavljivanjeKorisnika
* **Glavni sudionik:** Stanar, Predstavnik, Upravitelj, Administrator.
* **Cilj:** Prijaviti se u sustav kako bi mogao koristiti aplikaciju.
* **Sudionici:** Baze 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) Provodi se provjera podataka s bazom, korisnika se upućuje na glavnu stranicu.

* **Mogući drugi scenarij:**

1) Korisnika nema u bazi podataka, te se vraća na stranicu za prijavu.

2) Korisnik je unio krivu lozinku i korisničko ime, te se vraća na stranicu za prijavu.

* **UC2** – PogledajOglasnuPloču
* **Glavni sudionik:** Stanar, predstavnik, upravitelj.
* **Cilj:** Pogledati trenutno stanje na oglasnoj ploči.
* **Sudionici:** Baze podataka.
* **Preduvjeti:** Korisnik je prijavljen.
* **Rezultat:** Prikaz svih objava na oglasnoj ploči.
* **Željeni scenarij:**

1) Prilikom učitavanja stranice prikazuje se oglasna ploča sa objavama.

* **Mogući drugi scenarij:**

-

* **UC3** – DodajNaOglasnuPloču
* **Glavni sudionik:** Stanar, predstavnik, upravitelj.
* **Cilj:** Dodati nove objave, obrisati objave na oglasnoj ploči.
* **Sudionici:** Baze podataka.
* **Preduvjeti:** Korisnik je ulogiran i ima privilegiju objavljivanja.
* **Rezultat:** Korisnik dodaje podatke na oglasnu ploču.
* **Željeni scenarij:**

1) Korisnik odabire opciju dodavanja objava.

2) Sustav bilježi dodavanje te osvježava oglasnu ploču.

* **Mogući drugi scenarij:** -
* **UC4** – BrišiSaOglasnePloče
* **Glavni sudionik:** Stanar, predstavnik, upravitelj.
* **Cilj:** Obrisati objave na oglasnoj ploči.
* **Sudionici:** Baze podataka.
* **Preduvjeti:** Korisnik je ulogiran i ima privilegiju brisanja.
* **Rezultat:** Korisnik briše podatke na oglasnoj ploči.
* **Željeni scenarij:**

1) Korisnik odabire opciju brisanja objava.

2) Sustav bilježi brisanje te osvježava oglasnu ploču.

* **Mogući drugi scenarij:**.

1) Korisnik nema privilegiju za brisanje objave, te mu sustav odbija akciju.

* **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) Prikaz trenutnog stanja rasporeda čišćenja snijega, na zahtjev korisnika.

* **Mogući drugi scenarij:** -
* **UC6** – IzmjeniRasporedČišćenjaSnijega
* **Glavni sudionik:** Stanar i predstavnik.
* **Cilj:** Mogućnost zamjene u rasporedu čišćenja snijega.
* **Sudionici:** Baza podataka.
* **Preduvjeti:** Korisnik je prijavljen, te se nalazi na toj stranici.
* **Rezultat:** Korisnik traži izmjenu rasporeda.
* **Željeni scenarij:**

1) Korisnik zatražuje izmjenu u rasporedu.

2) Izmjena prihvaćena.

3) Sustav osvježava raspored čišćenja snijega.

* **Mogući drugi scenarij:**

1) Izmjena odbijena.

2) Sustav javlja korisniku da je izmjena odbijena.

* **UC7** – PregledajOpćeInformacije
* **Glavni sudionik:** Stanar, predstavnik, upravitelj.
* **Cilj:** Dobiti uvid u opće informacije korisnika i zgrade.
* **Sudionici:** Baze podataka.
* **Preduvjeti:** Korisnik mora bit prijavljen.
* **Rezultat:** Otvara se stranica sa općim informacijama korisnika i zgrade.
* **Željeni scenarij:**

1) Prikaz informacija o korisniku i zgradi, na zahtjev korisnika.

* **Mogući drugi scenarij:** -
* **UC8**– IzmjeniOpćeInformacije
* **Glavni sudionik:** Stanar, predstavnik, upravitelj.
* **Cilj:** Izmjeniti opće informacije korisnika ili zgrade.
* **Sudionici:** Baze 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) Unosi nove podatke.

3) Sustav ažurira unesene podatke.

* **Mogući drugi scenarij:**

1) Uneseni podatci su neispravni, te sustav odbija akciju.

2) Korisnik nema privilegiju za izmjenu odabranih podataka.

* **UC9** – 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) Prikaz osobnih naloga.

2) Prikaz naloga ostalih stanara.

3) Prikaz budućih naloga.

* **Mogući drugi scenarij:** -
* **UC10**– ObradiNaloge
* **Glavni sudionik:** Upravitelj.
* **Cilj:** Provesti naloge.
* **Sudionici:** Baza podataka.
* **Preduvjeti:** Korisnik mora bit ulogiran i nalaziti se na toj stranici.
* **Rezultat:** Korisnik obrađuje odabrane naloge.
* **Željeni scenarij:**

1) Upravitelj vidi trenutne naloge i njihov status.

2) Obrađuje naloge po važnosti.

3) Sustav osvježava promjene.

* **Mogući drugi scenarij:**

1) Nema dovoljno novaca na računu za obradu naloga.

* **UC11** – 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.

1) Korisnik unosi sve potrebne podatke za opis štete.

2) Sustav provjerava hitnost štete i šalje zahtjev za obradom iste.

* **Mogući drugi scenarij:**

1) Ista šteta je već unesena, te se zahtjev odbija i vraća se na početni ekran.

* **UC12** – OdjavljivanjeKorisnika
* **Glavni sudionik:** Stanar, predstavnik, upravitelj.
* **Cilj:** Odjaviti korisnika iz aplikacije.
* **Sudionici:** -
* **Preduvjeti:** Korisnik je prijavljen.
* **Rezultat:** Otvara se početna stranica i korisnik ima ponovno mogućnost prijave.
* **Željeni scenarij:**

1) Odabir opcije za odjavu.

1) Odjava iz sustava.

2) Prikaz stranice za prijavu.

* **Mogući drugi scenarij:** -
* **UC13** – 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.

1) Unos svih potrebnih podataka za novog korisnika.

2) Sustav osvježava bazu podataka.

* **Mogući drugi scenarij:**

1) Unos krivih podatak.

2) Korisnik je već dodan.

3) Postoji korisnik s istim privilegijama(predstavnik, upravitelj).

* **UC14** – 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) Odabir korisnika koji se briše.

2) Sustav osvježava bazu podataka.

* **Mogući drugi scenarij:** -
* **UC15** – UrediObjavuNaOglasnojPloči
* **Glavni sudionik:** Stanar, predstavnik, upravitelj, administrator.
* **Cilj:** Urediti objave na oglasnoj ploči.
* **Sudionici:** Baze podataka.
* **Preduvjeti:** Korisnik je ulogiran i ima privilegiju uređivanja.
* **Rezultat:** Korisnik uređuje podatke na oglasnoj ploči.
* **Željeni scenarij:**

1) Korisnik odabire opciju uređivanja objava.

2) Sustav bilježi uređivanje te osvježava oglasnu ploču.

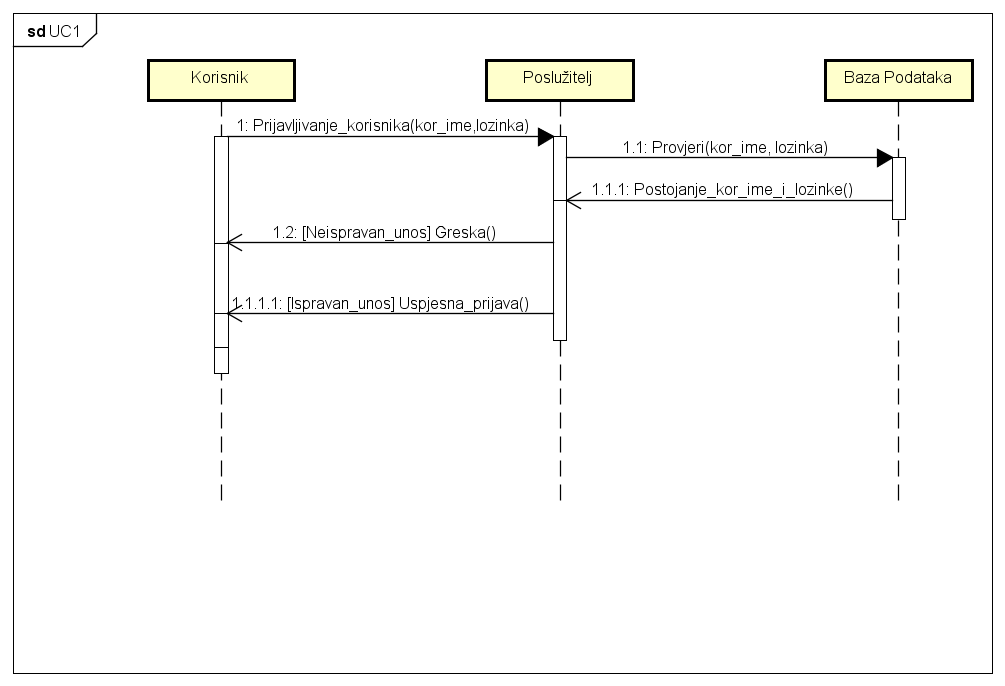
* **Mogući drugi scenarij:**.

1) Korisnik nema privilegiju za uređivanje objave, te mu sustav odbija akciju.

**Sekvencijski dijagrami :**

**Obrazac uporabe UC1 (PrijavljivanjeKorisnika)**

Korisnik sa postojećim korisničkim imenom i lozinkom unosi svoje korisničko ime i lozinku i šalje zahtjev za prijavu poslužitelju. Poslužitelj provjerava da li postoje unešeni podatci u bazi podataka. Ako podatci ne postoje ili su krivi dojavljuje grešku, dok u suprotnom se korisnik upješno prijavljuje.



Slika 1 Sekvencijski dijagram obrasca PrijavljivanjeKorisnika

**Obrazac uporabe UC2 (PogeldajOglasnuPloču)**

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



Slika 2 Sekvencijski dijagram obrasca PogledajOglasnuPloču

**Obrazac uporabe UC3 (DodajNaOglasnuPloču)**

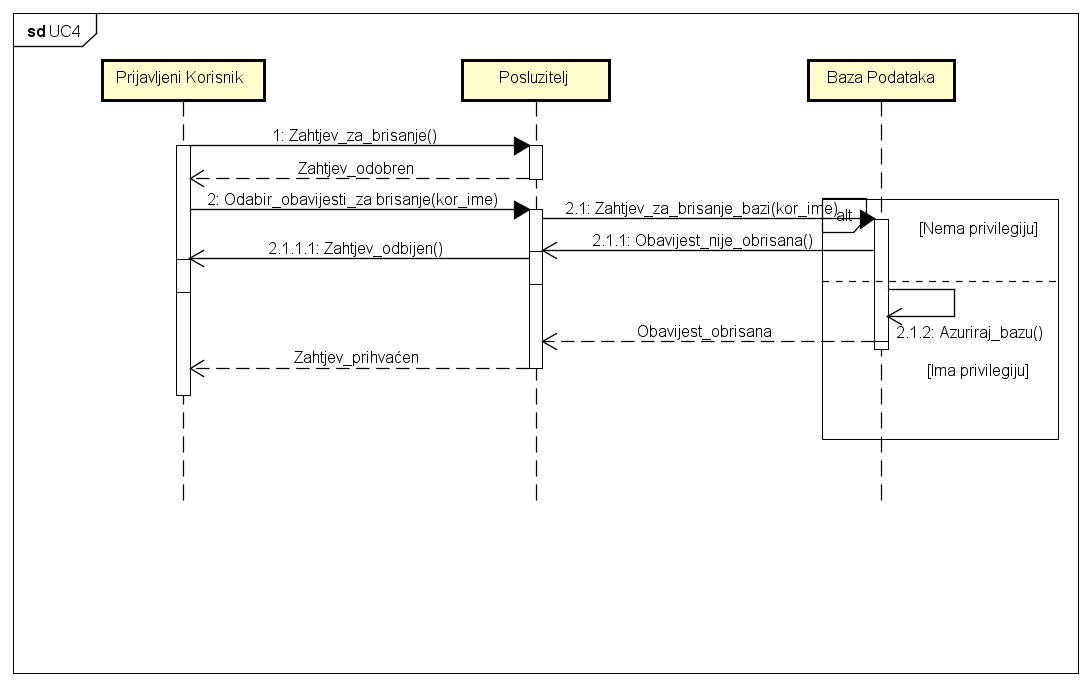
Korisnik zahtjeva dodavanje obavijesti na oglasnu ploču, što mu poslužitelj dozvoljava. Zatim korisnik unosi sve potrebne podatke koje želi dodati, poslužitelj iz prosljeđuje bazi podataka koja ih unosi i sprema. I na kraju korisnik dobiva obavijest da je njegov unos uspješno napravljen.



Slika 3 Sekvencijski dijagram obrasca DodajNaOglasnuPloču

**Obrazac uporabe UC4 (BrišiSaOglasnePloče)**

Korisnik zahtjeva brisanje što mu poslužitelj odobrava. Nakon toga korisnik odabire obavijest koju želi obrisati, poslužitelj prosljeđuje to bazi, koja provjerava ima li taj korisnik privilegiju obrisati obavijest koju je odabrao. Ako nema privilegiju, obavijest neće biti obrisana, a korisniku će zahtjev biti odbijen. S druge strane, ako ima privilegiju, obavijest će biti obrisana, baza će se ažurirati i korisnik dobiti obavijest da je uspješno obavijest obrisana i njegov zahtjev prihvaćen.



Slika 4 Sekvencijski dijagram obrasca BrišiSaOglasnePloče

**Obrazac uporabe UC5 (PogledajRasporedČišćenjaSnijega)**

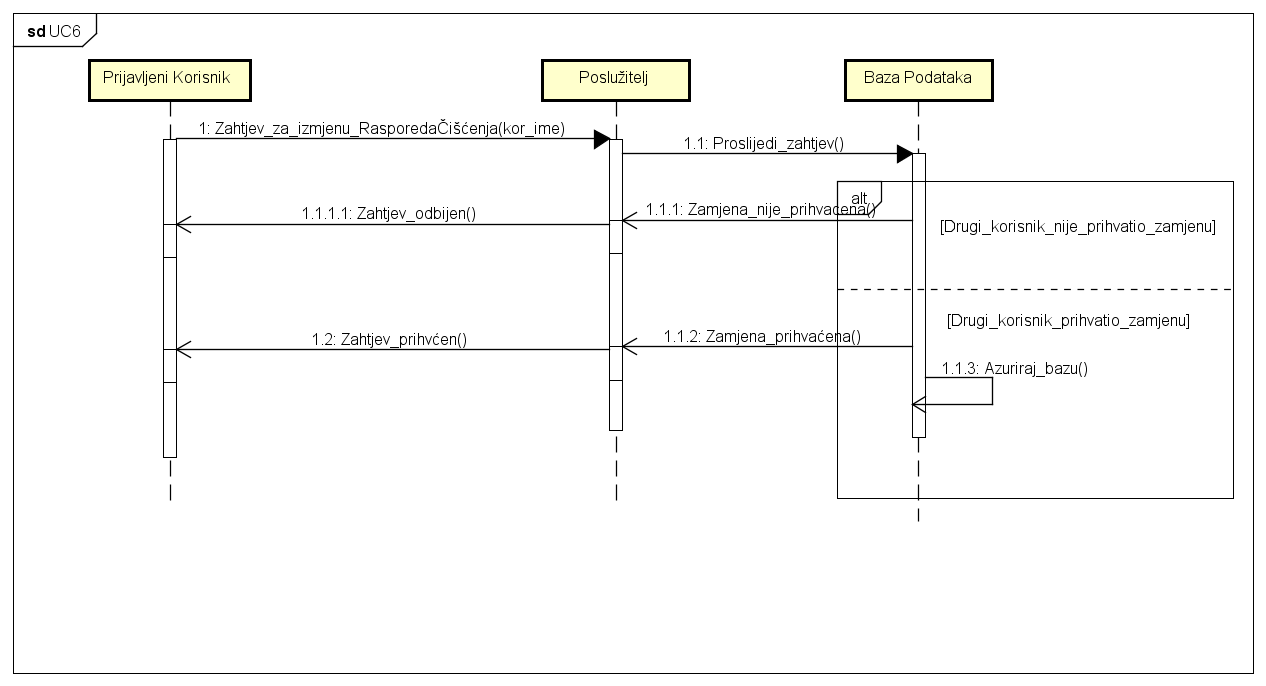
Prijavljeni korisnik zahtjeva pregled rasporeda čišćenja snijega. Poslužitelj dohvaća sve podatke iz baze podataka te ih daje na uvid korisniku.



Slika 5 Sekvencijski dijagram obrasca PregledajRasporedČišćenjaSnijega

**Obrazac uporabe UC6 (IzmjeniRasporedČišćenjaSnijega)**

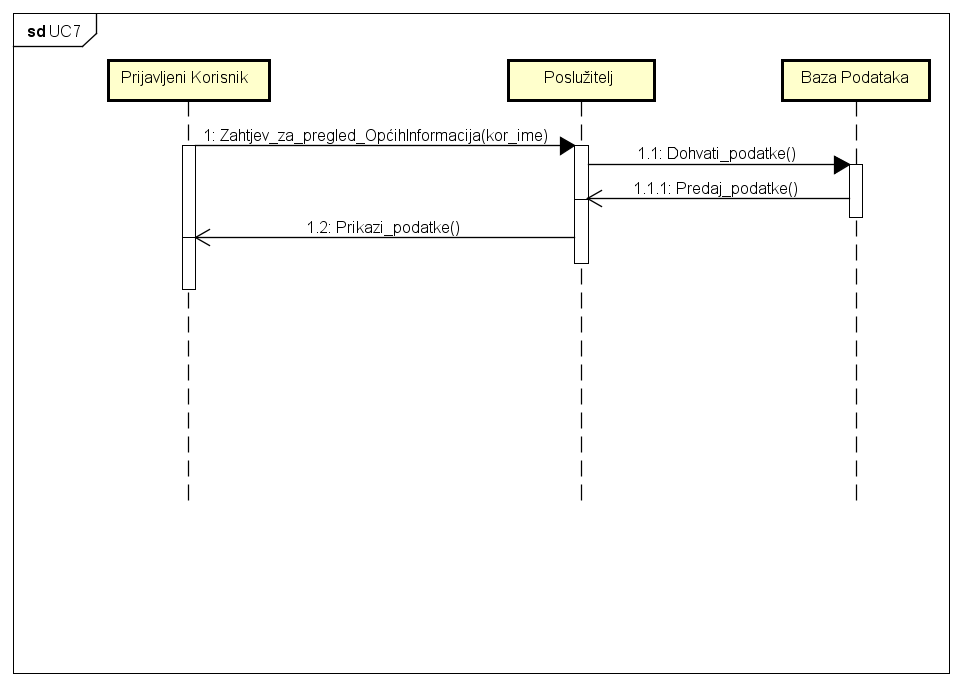
Korisnik šalje zahtjev za izmjenu rasporedu. Poslužitelj šalje zahtjev bazi gdje čeka da drugi korisnik potvrdi ili odbije zamjenu. Ako drugi korisnik nije prihvatio zamjenu, poslužitelj vraća da je zahtjev odbijen. U drugom slučaju, da je zahtjev prihvaćen, poslužitelj javlja korisniku da je zahtjev prihvaćen i baza podataka se ažurira.



Slika 6 Sekvencijski dijagram obrasca IzmjeniRasporedČišćenjaSnijega

**Obrazac uporabe UC7 (PregledajOpćeInformacije)**

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



Slika 7 Sekvencijski dijagram obrasca PregledajOpćeInformacije

**5. Ostali zahtjevi**

* Sustav mora podržavati paralelan rad više korisnika
* Sustav i korisničko sučelje moraju podržavati sve znakove hrvatske abecede, kao i interpukcijske znakove
* Sustav mora biti napravljen tako da omogućava jednostavno korištenje korisniku bez detaljnih uputa
* Pretraga baze podataka ne smije trajati duže od 5 sekundi

**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.

Podsustavi arhitekture:

* Web poslužitelj
* Web aplikacija
* Baza podataka

**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 posljužitelj 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 spriječava 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 pream 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 atribute 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 atribute: 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 atribute: ID, IDKorisnika, Opis. Entitet sadrži jednu Many-to-One vezu preko IDKorisnika prema atributu ID u entitetu **Korisnik.**

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



Slika 8 ER model baze podataka

**7. Implementacija i korisničko sučelje**

7.1. Dijagram razmeštanja

7.2. Korištene tehnologije i alati

7.3. Isječak programskog koda vezan za temeljnu funkcionalnost

7.4. Ispitivanje programskog rješenja

7.5. Upute za instalaciju

7.6. Korisničke upute

**8. Zaključak i budući rad**

Analiziran je projektni zadatak, napravljeni su dijagrami obrazaca uporabe, sekvencijski dijagrami, model sustava, dijagrami razreda, dijagrami objekata, baza podataka, te dio predviđene arhitekture i logike sustava.

Daljni dio rada bi trebao biti usmjeren na dovršavanje same arhitekture i logike sustava i većim dijelom modeliranje i implementacija samog izgleda web stranice za interakcije korisnika.

**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)**

**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.

**Dodatak C: Prikaz aktivnosti grupe**