

Programsko inženjerstvo

Ak. god. 2020./2021.

Zamjena soba

Dokumentacija, Rev. 1

Grupa: *springRolice*

Voditelj: *Mateja Iveta*

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

Nastavnik: *Ivana Cepetić*

Sadržaj

1	Dnevnik promjena dokumentacije	3
2	Opis projektnog zadatka	4
2.1	Opseg projekta	5
2.2	Postojeća slična rješenja	8
3	Specifikacija programske potpore	10
3.1	Funkcionalni zahtjevi	10
3.1.1	Obrasci uporabe	12
3.1.2	Sekvencijski dijagrami	19
3.2	Ostali Zahtjevi	21
4	Arhitektura i dizajn sustava	22
4.1	Baza podataka	23
4.1.1	Opis tablica	24
4.1.2	Dijagram baze podataka	29
4.2	Dijagram razreda	30
4.3	Dijagram stanja	31
4.4	Dijagram aktivnosti	32
4.5	Dijagram komponenti	33
5	Implementacija i korisničko sučelje	34
5.1	Korištene tehnologije i alati	34
5.2	Ispitivanje programskog rješenja	35
5.2.1	Ispitivanje komponenti	35
5.2.2	Ispitivanje sustava	35
5.3	Dijagram razmještaja	36
5.4	Upute za puštanje u pogon	37
6	Zaključak i budući rad	38
	Popis literature	39

Indeks slika i dijagrama	40
Dodatak: Prikaz aktivnosti grupe	41

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napisani obrasci uporabe	Bortas	02.11.2020.
0.2	Napisan Opis projektnog zadatka	Horvat	03.11.2020.
0.5	Dodani Use Case dijagrami i sekvencijski dijagrami	Bortas	04.11.2020.

Moraju postojati glavne revizije dokumenata 1.0 i 2.0 na kraju prvog i drugog ciklusa. Između tih revizija mogu postojati manje revizije već prema tome kako se dokument bude nadopunjavao. Očekuje se da nakon svake značajnije promjene (dodatka, izmjene, uklanjanja dijelova teksta i popratnih grafičkih sadržaja) dokumenta se to zabilježi kao revizija. Npr., revizije unutar prvog ciklusa će imati oznake 0.1, 0.2, ..., 0.9, 0.10, 0.11.. sve do konačne revizije prvog ciklusa 1.0. U drugom ciklusu se nastavlja s revizijama 1.1, 1.2, itd.

2. Opis projektnog zadatka

dio 1. revizije

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

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

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

2.1 Opseg projekta

Glavni je cilj razvoja aplikacije "Zamjena sobe" svim studentima koji su tijekom studiranja smješteni u studentskim domovima ponuditi rješenje za dva najveća problema s kojima se susreću. To su traženje sobe, to jest osobe za zamjenu te sam proces prijave te zamjene u Studentskim centrima.

Nakon provedenog postupka natječaja za smještaj i objave liste ostvarenja prava na studentski smještaj, studenti imaju mogućnost zamjene doma ili sobe unutar dodijeljenog doma. Studenti potencijalne zamjene traže uglavnom nasumično, usmenim putem ili putem raznih foruma i društvenih mreža, na posebnim grupama za pojedine gradove i studentske domove. Ovakvim izuzetno nepraktičnim i nestrukturiranim načinom zamjene vrlo je lako da se dva, moguće idealna oglasa, nikada ne povežu. Također, studentima se ne omogućuje postavljanje i traženje zamjene po više željenih kriterija.

Ova bi aplikacija bila dostupna svima, ne samo studentima na određenom fakultetu ili gradu. Neregistrirani korisnik ima mogućnost pregleda svih oglasa i glavnih informacija o pojedinom oglasu: ime, prezime i profilna slika osobe koja je predala oglas, grad, paviljon i broj ponuđene sobe te željene kriterije nove sobe. Za korištenje svih ostalih funkcionalnosti neregistrirani korisnik može se registrirati, to jest stvoriti novi račun ili se prijaviti u sustav s već stvorenim računom. Kod prijave u sustav potrebno je upisati korisničko ime ili e-mail adresu i lozinku. Za registraciju korisnik mora upisati sljedeće podatke:

- ime
- prezime
- korisničko ime
- e-mail adresa
- lozinka
- jmbag

Prijavljenom korisniku omogućen je pregled njegova profila na kojemu može vidjeti i uređivati svoje osobne podatke i sliku profila, ili izbrisati svoj račun. Također, na pregledu "Moji oglasi" ima uvid u vlastite predane oglase koji mogu biti aktivni ili neaktivni. Aktivni oglas može dodatno uređivati ili ga učiniti neaktivnim.

Kod predaje oglasa za zamjenu sobe student o sobi koju 'nudi' navodi sljedeće informacije:

- grad
- dom
- paviljon
- kat
- broj kreveta
- tip kupaonice

Dodatno, student u tekstualno polje ima opciju upisa dodatnih značajki sobe, poput najbliže menze i slično. Osim informacija o svojoj sobi, student navodi i informacije o sobi koju želi, to jest željene kriterije. Jedino se ograničenje odnosi na grad - student ne može tražiti zamjenu u drugom gradu. Ostali kriteriji mogu biti i višestruki, na primjer može označiti da mu odgovaraju sobe s jednim, dva ili tri kreveta.

Nakon što preda oglas, student ima uvid u ostale aktivne oglase koji odgovaraju njegovim željenim kriterijima. Ti oglasi mogu biti prikazani kao pojedinačni ili ulančani. Sve ponuđene oglase student može ocijeniti po stupnjevima podudaranja kriterija:

- 1 - sviđa mi se
- 2 - sviđa mi se jako
- 3 - to je to
- Ne prikazuj više ovaj oglas

Osim pregleda ponuđenih oglasa, student također ima uvid u zaprimljene ponude za zamjenu - koji su studenti zainteresirani za njegov oglas te njihove oglase.

Sustav periodički pretražuje nove dodane oglase i ažurira uparivanja po kriterijima. Kao što je već spomenuto, uz izravne, pojedinačne zamjene sustav nudi i potencijalne lance razmjene, ako takvi postoje.

Nakon što oba studenta (ili svi studenti u slučaju lanca razmjene) ocijene razmjenu, sustav im šalje e-mail poruku kojom se traži da konačno potvrde zamjenu soba. Poveznica na određenu zamjenu sadržana je u e-mail poruci.

Sve su potvrđene zamjene dostupne na uvid ovlaštenim djelatnicima pojedinog Studentskog centra. Nakon prijave u sustav zaključane zamjene izravno evidentiraju u sustavu Studentskog centra, bez potrebe za osobnom intervencijom studenata.

2.2 Postojeća slična rješenja

Mnoga strana sveučilišta imaju slična rješenja za zamjenu soba, ali unutar svojih privatnih sustava koji su dostupni samo studentima toga sveučilišta. Jedan je takav primjer 'Room Swap' sveučilišta *The University of North Carolina at Chapel Hill*. 'Room Swap' je također dostupan samo studentima na sveučilištu 'North Carolina'. Glavna je razlika između ovog sustava i naše aplikacije u tome što putem *Room Swapa* student ne može provesti 'pravu' zamjenu sobe – može izabrati neku slobodnu sobu na kampusu ili sobu koja nije do kraja popunjena, ne može se direktno zamijeniti s drugom osobom, što bi naša aplikacija omogućavala. Student u sustavu *Room Swap* prvo izabire zgradu te se klikom na zgradu otvara pregled katova i dostupnih slobodnih soba.

Room Selection - Room Swap (Step 9 of 9)

Main | Renew Residence Hall | General Room Selection | **Room Swap**

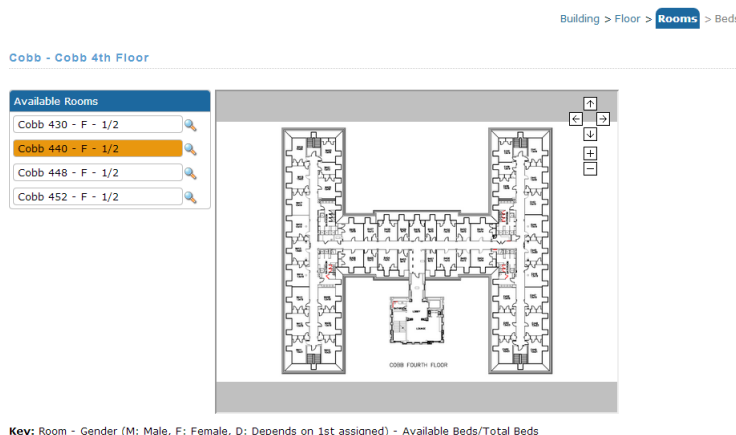
Building > **Floor** > Rooms > Beds

Cobb

Floor/Suite	Single Occupancy Spaces	Double Occupancy Spaces	Triple Occupancy Spaces
Cobb 1st Floor	0	1	1
Cobb 2nd Floor	0	1	1
Cobb 4th Floor	0	4	0

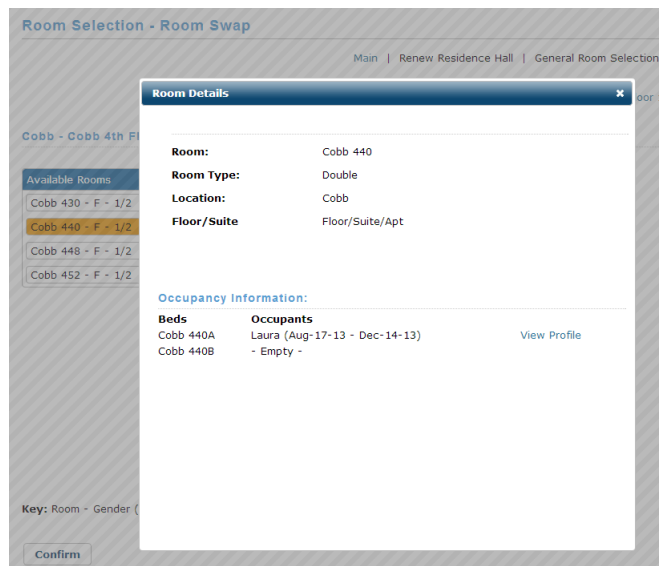
Slika 2.1: Prikaz katova i dostupnih slobodnih soba na pojedinom katu

Još je jedna razlika u odnosu na našu aplikaciju to što nisu sve sobe dostupne svima. Na primjer, postoje 'First Year Experience Buildings' koje su dostupne samo brucosima. Također, sobe s više kreveta podijeljene su po spolu. Nakon izbora kata aplikacija prikazuje tlocrt toga kata sa svim dostupnim sobama.



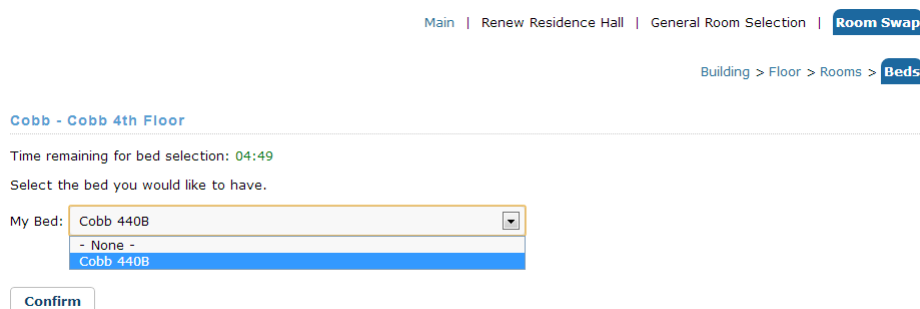
Slika 2.2: Tlocrt odabranog kata

Klikom na sobu moguć je prikaz detalja o sobi ali i studentu kojemu je trenutno dodijeljena soba. Naša bi aplikacija, uz osnovne informacije o sobi poput doma, paviljona, kata i vrste, nudila i mogućnost detaljnijeg opisa - tip kupaonice, blizina menza itd.



Slika 2.3: Detalji o sobi i studentima u sobi

Nakon odabira sobe pokreće se odbrojavanje od 5 minuta unutar kojih student mora izabrati krevet u sobi i konačno potvrditi svoj potpuni odabir. Ako se odabir unutar 5 minuta ne potvrdi zamjena se neće provesti. Naša aplikacija ne bi postavljala vremensko ograničenje na potvrdu zamjene, već bi se čekala potvrda obje strane.



Slika 2.4: Odbrojavanje i konačni odabir

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Studenti
2. Djelatnici Studentskih centara
3. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani korisnik (inicijator) može:
 - (a) vidjeti sve oglase
 - (b) se registrirati, za stvaranje korisničkog računa potrebni su: ime, prezime, korisničko ime, lozinka, e-mail, JMBAG
2. Registrirani korisnik (inicijator) može:
 - (a) se prijaviti u sustav
 - (b) stvoriti oglas
 - (c) vidjeti svoj profil i mijenjati podatke
 - (d) izbrisati svoj profil
 - (e) vidjeti oglase pojedinačne ili ulančane koji odgovaraju njegovim kriterijima
 - (f) pregledati sve svoje oglase (aktivne i neaktivne)
 - (g) uređivati svoje oglase
 - (h) učiniti svoje oglase aktivnim i neaktivnim
 - (i) brisati svoj oglas
 - (j) "lajkati" oglase po stupnjevima:
 - i. 1-sviđa mi se
 - ii. 2-sviđa mi se jako
 - iii. 3-to je to
3. Djelatnik SC-a (inicijator) može:

- (a) se prijaviti u sustav
- (b) pregledati sve zaključane zamjene

4. Timer (inicijator) može:

- (a) pokrenuti uparivanje studenata s oglasima koji odgovaraju njegovim kriterijima

5. Baza podataka (sudionik):

- (a) pohranjuje sve podatke o korisnicima i njihovim ovlastima
- (b) pohranjuje sve podatke o oglasima i sobama

6. Poslužitelj (sudionik):

- (a) obrađuje zahtjeve korisnika

3.1.1 Obrasci uporabe

UC1 -Pregled oglasa

- **Glavni sudionik:** Neregistrirani korisnik
- **Cilj:**Pregledati oglase soba
- **Sudionici:** baza podataka
- **Preduvjet:**
- **Opis osnovnog tijeka:**
 1. Prilikom otvaranja aplikacije sustav prikazuje sve oglase

UC2 -Registracija

- **Glavni sudionik:** Neregistrirani korisnik
- **Cilj:**Registrirati korisnika i ovlastiti ga
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** dostupnost poslužitelja, korisnik nije registriran
- **Opis osnovnog tijeka:**
 1. Korisnik pritisće tipku "Registriraj se"
 2. Sustav otvara stranicu za registraciju
 3. Korisnik unosi potrebne podatke
 4. Sustav provjerava ispravnost podataka
 5. Sustav registrira korisnika
 6. Korisnik prima podatke o uspješnoj registraciji
- **Opis mogućih odstupanja:**
 - 2.a Korisnik unosi neispravne podatke ili već zauzeto korisničko ime ili e-mail
 1. Sustav obavještava korisnika o pogrešci
 2. Sustav vraća korisnika na stranicu za registraciju sa crveno označenom greškom
 3. Korisnik mijenja neispravne podatke te završava unos ili odustaje od registracije

UC3 -Prijava u sustav

- **Glavni sudionik:** Registrirani korisnik, Djelatnik SC-a
- **Cilj:**Prijaviti se u sustav
- **Sudionici:** baza podataka
- **Preduvjet:** Korisnik se je registrirao

- **Opis osnovnog tijeka:**
 1. Unos korisničkog imena i lozinke
 2. Sustav provjerava ispravnost podataka
 3. Sustav korisniku otvara početnu stranicu i korisnik ima pristup korisničkim funkcijama
- **Opis mogućih odstupanja:**
 - 2.a Unos neispravnih podataka
 1. Sustav obavještava o neispravnim podacima i vraća na stranicu za prijavu s crveno označenom pogreškom
 2. Korisnik ispravlja podatke

UC4 -Objavljivanje oglasa

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:**Objaviti oglas
- **Sudionici:** baza podataka,poslužitelj
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju "Objavi novi oglas"
 2. Sustav otvara stranicu za objavljivanje oglasa
 3. Korisnik upisuje potrebne podatke za sobu koju nudi i kriterije za sobu koju traži. Sve podatke odabire iz padajućih izbornika osim polja za proizvoljni komentar
 4. Korisnik pritišće tipku "objavi"
 5. Sustav pohranjuje oglas u bazu

UC5 -Pregled mogućih zamjena

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:**Pregledati ponuđene moguće zamjene koje odgovaraju kriterijima korisnika
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen i ima aktivan oglas
- **Opis osnovnog tijeka:**
 1. Korisnik otvara početnu stranicu
 2. Sustav prikazuje sve oglase koji odgovaraju njegovim kriterijima

UC6 - Uparivanje oglasa

- **Glavni sudionik:** Timer
- **Cilj:** Upariti studenta s oglasima koji odgovaraju njegovim kriterijima
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:**
- **Opis osnovnog tijeka:**
 1. Timer pokreće uparivanje svakih 5 sati
 2. Sustav za svakog studenta stvara listu svih oglasa koji direktno ili lančano odgovaraju njegovim kriterijima

UC7 -Lajkanje oglasa

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** "Lajkati" oglase soba koje korisnika zanimaju
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen, ima aktivan oglas i barem jedan oglas odgovara njegovim kriterijima
- **Opis osnovnog tijeka:**
 1. Korisnik oglase koji ga zanimaju označava("lajka") po stupnjevima od 1 do 3
 2. Sustav sprema njegov odabir

UC8 -Mjenjanje Lajka

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Promijeniti razinu "lajka" ili maknuti lajk
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen, ima aktivan oglas, barem jedan oglas odgovara njegovim kriterijima
- **Opis osnovnog tijeka:**
 1. Korisnik odabire neki drugi stupanj "lajka" ili odustaje od "lajka" micanjem oznake
 2. Njegov odabir se sprema u bazu podataka

UC9 -Pregled mojih oglasa

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Vidjeti sve svoje oglase
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen i ima oglas

- **Opis osnovnog tijeka:**

1. Korisnik odabire opciju "Moji oglasi"
2. Sustav otvara stranicu gdje se prikazuju svi korisnikovi oglasi

UC10 -Uređivanje oglasa

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:**Urediti već postojeći oglas
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen i ima aktivan oglas
- **Opis osnovnog tijeka:**
 1. Korisnik odlazi pod "Moji oglasi"
 2. Sustav prikazuje stranicu sa listom njegovih oglasa
 3. Korisnik odabire opciju uredi
 4. Korisnik mijenja podatke ili atribut je li oglas aktivan ili neaktivan
 5. Korisnik potvrđuje promjene odabirom opcije spremi
 6. Promjene se spremaju u bazu podataka

UC11 -Brisanje oglasa

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:**Izbrisati oglas
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen i ima oglas
- **Opis osnovnog tijeka:**
 1. Korisnik odlazi pod "Moji oglasi"
 2. Sustav prikazuje stranicu sa listom njegovih oglasa
 3. Korisnik odabire opciju izbriši
 4. Sustav briše oglas iz baze podataka

UC12 -Pregled profila

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:**Vidjeti svoje osobne podatke
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odlazi pod "Moj profil"
 2. Sustav otvara stranicu korisnikova profila

UC13 -Promjena profila

- **Glavni sudionik:** Prijavljeni korisnik
- **Cilj:** Vidjeti svoje osobne podatke
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odlazi pod "Moj profil"
 2. Sustav otvara stranicu korisnikova profila
 3. Korisnik odabire opciju "uredi"
 4. Korisnik mijenja osobne podatke
 5. Korisnik pritišće tiku "spremi promjene"
 6. Sustav sprema promjene

UC14 -Brisanje profila

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Vidjeti svoje osobne podatke
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odlazi pod "Moj profil"
 2. Sustav otvara stranicu korisnikova profila
 3. Korisnik odabire opciju "izbriši profil"
 4. Sustav briše korisnikov profil i otvara početnu stranicu

UC15 -Potvrđivanje zamjene

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Potvrditi zamjenu sobe
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Korisnik je prijavljen, sve uključene strane su "lajkale" oglas
- **Opis osnovnog tijeka:**
 1. Korisnik dobiva obavijest da su sve strane "lajkale" zamjenu
 2. Korisnik pomoću linka iz obavijesti dolazi do oglasa
 3. Korisnik potvrđuje zamjenu
 4. Potvrda se sprema u bazu podataka

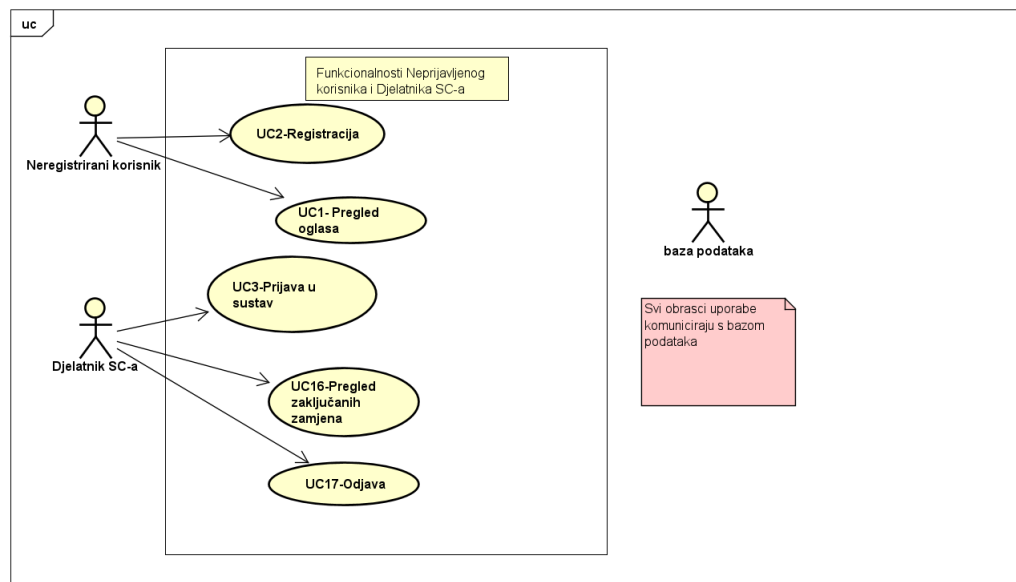
UC16 -Pregled zaključenih zamjena

- **Glavni sudionik:** Djelatnik SC-a
- **Cilj:** Vidjeti sve zaključane zamjene kako bi se mogle provesti u sustavu SC-a
- **Sudionici:** baza podataka, poslužitelj
- **Preduvjet:** Djelatnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Sustav na početnoj stranici prikazuje listu svih zaključenih zamjena
 2. Korisnik označuje sve zamjene koje je proveo
 3. Sustav miče označene zamjene sa stranice

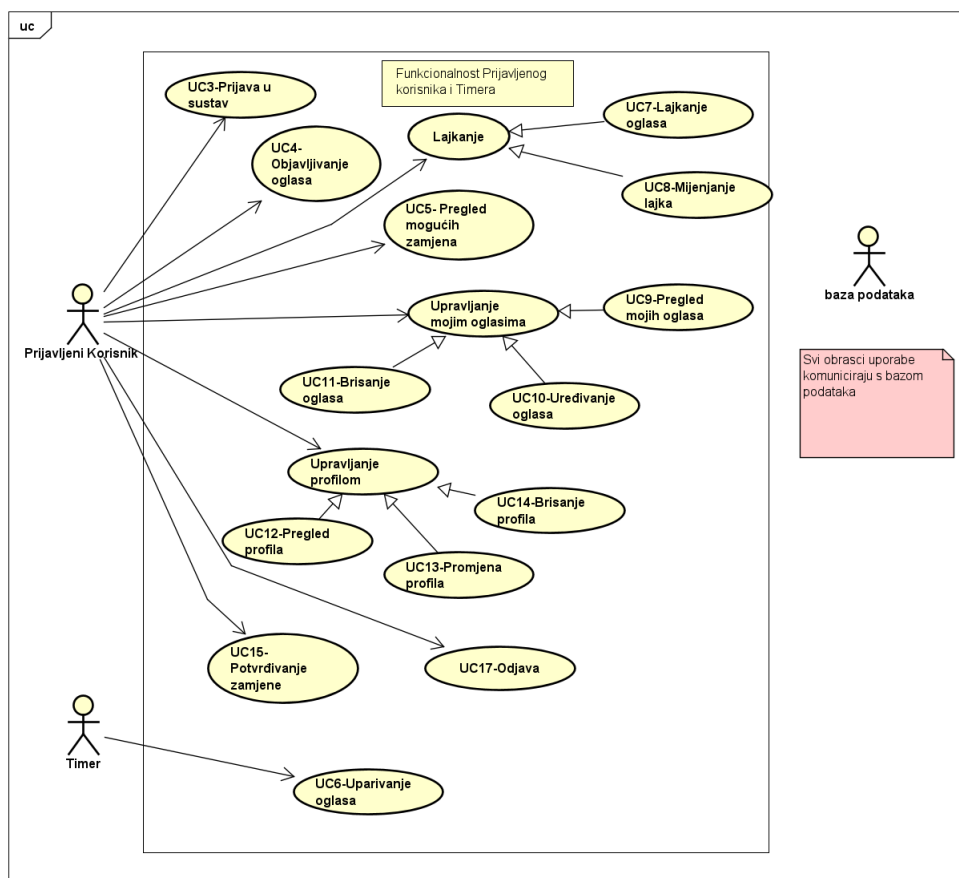
UC17 -Odjava

- **Glavni sudionik:** Djelatnik SC-a, Prijavljeni korisnik
- **Cilj:** Odjaviti se iz sustava
- **Sudionici:** poslužitelj
- **Preduvjet:** Djelatnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik pritišće tipku "odjavi se"
 2. Sustav odjavljuje korisnika te otvara početnu stranicu

Dijagrami obrazaca uporabe



Slika 3.1: Dijagrami obrazaca uporabe, funkcionalnost neregistriranog korisnika i djelatnika SC-a

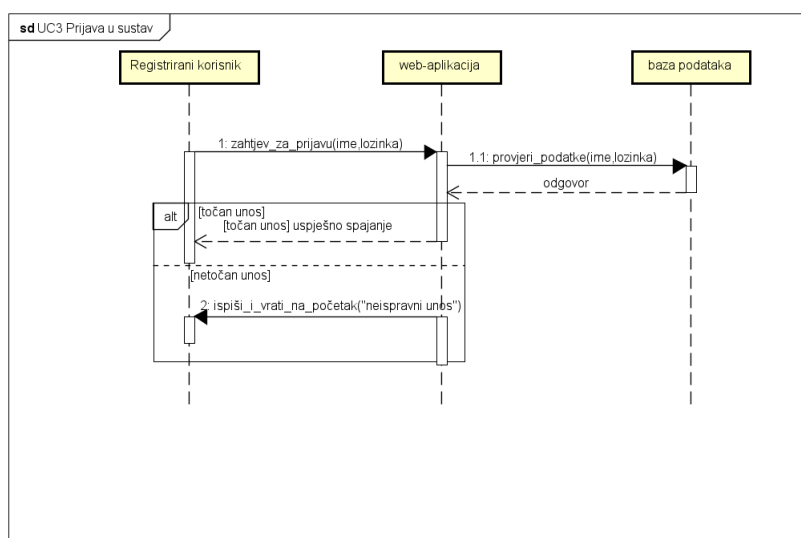


Slika 3.2: Dijagrami obrazaca uporabe, funkcionalnost prijavljenog korisnika i timera

3.1.2 Sekvencijski dijagrami

Obrazac uporabe UC3(Prijava u sustav)

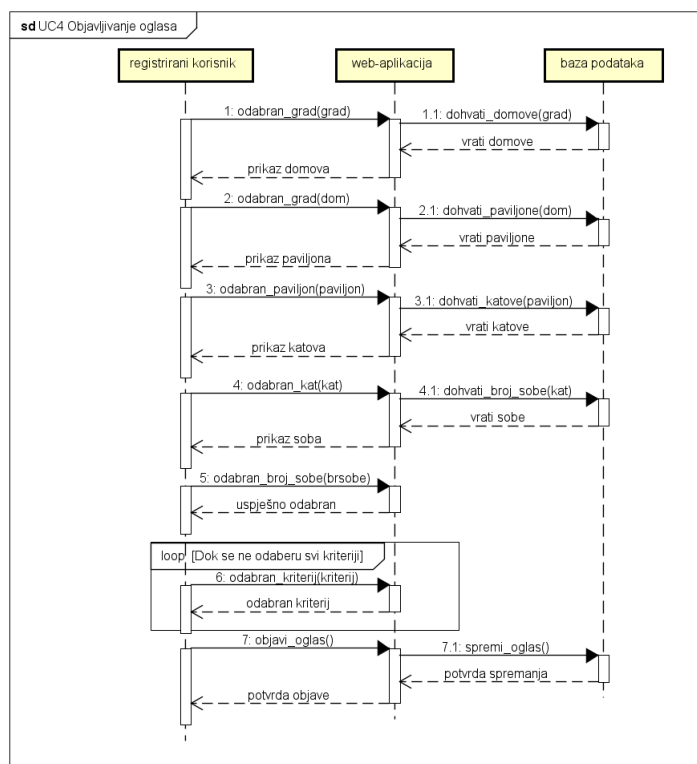
Neprijavljeni korisnik šalje zahtjev za prijavu s korisničkim imenom i lozinkom. Provjerava se ako je korisničko ime u bazi podataka. Ako ime ne postoji dojavljuje se greška u suprotnom provjerava se ako mu odgovara unesena lozinka. Ako lozinka odgovara tada se korisniku dodjeljuju ovlasti u suprotnom se dojavljuje greška.



Slika 3.3: Sekvencijski dijagram za UC3

Obrazac uporabe UC4(Objavljivanje oglasa)

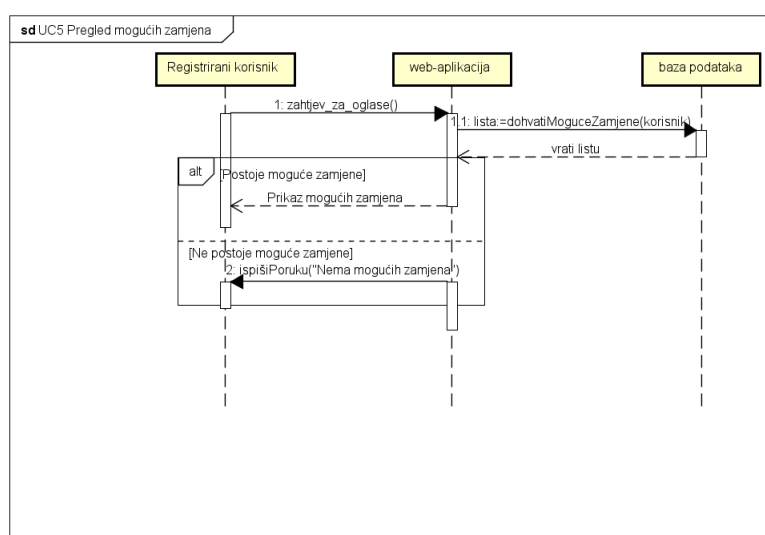
Korisnik odabire opciju "Objavi novi oglas". Zatim u obrascu iz padajućih izbornika bira grad zatim mu se nude svi domovi. Nakon odabira doma prikazuju mu se paviljoni i tako dalje za kat i broj sobe. Zatim korisnik označava iste kriterije za sobu koju traži. Pritiskom na oznaku spremi sprema se oglas u bazu podataka.



Slika 3.4: Sekvencijski dijagram za UC4

Obrazac uporabe UC5(Pregled mogućih zamjena)

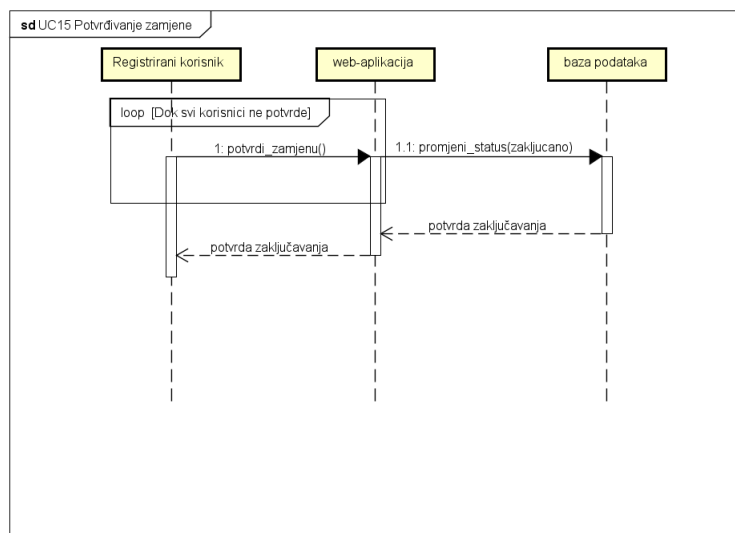
Korisnik nakon što napravi oglas odlazi na početnu stranicu gdje dobiva listu svih oglasa koji odgovaraju njegovim kriterijima.



Slika 3.5: Sekvencijski dijagram za UC5

Obrazac uporabe UC15(Potvrđivanje zamjene)

Korisnik potvrđuje zamjenu odabirom opcije "zaključaj zamjenu". Potvrda se sprema u bazu podataka promjenom statusa oglasa tek kad sve strane potvrde zamjenu.



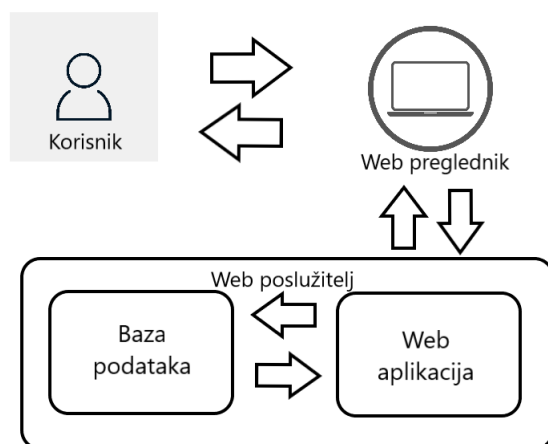
Slika 3.6: Sekvencijski dijagram za UC15

3.2 Ostali Zahtjevi

1. Sustav treba omogućiti rad više korisnika u stvarnom vremenu
2. Sustav bi se trebao moći koristiti bez dodatnih uputa
3. Sustav i korisničko sučelje trebaju podržavati hrvatsku abecedu
4. Izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi
5. Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava
6. Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava
7. Aplikacija treba biti izvedena kao web aplikacija prilagođena (engl. responsive) mobilnom uređaju

4. Arhitektura i dizajn sustava

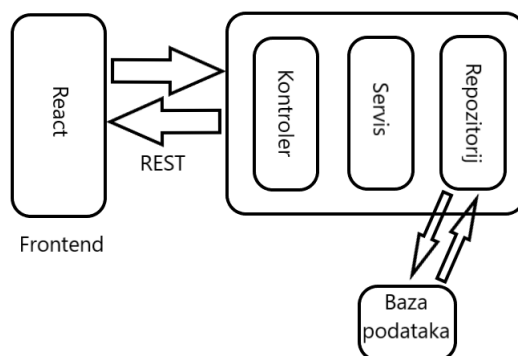
Arhitektura sustava je web aplikacija kojoj će korisnici pristupiti pomoću web preglednika. Odлучili smo se na takvu arhitekturu jer je cilj sustava da bude što jednostavniji za korištenje i da mu se može pristupiti sa svih mjesta.



Slika 4.1: Skica sustava

Programski jezik koji smo odabrali za izradu aplikacije je java s razvojnim okvirom Spring Boot te JavaScript. Odabrano razvojno okruženje je IntelliJ IDEA. Aplikacija je organizirana u dva sloja: frontend i backend. Za izradu frontenda koristi se React. Frontend i backend komuniciraju pomoću RESTa. Backend se sastoji od tri sloja:

- Kontroler - služi za komunikaciju s frontendom. Zaprma HTTP zahtjev te određuje koja će se funkcionalnost izvršavati
- Servis - u njima se odvijaju poslovne logike i sve funkcionalnosti aplikacije
- Repozitorij - dohvaća i sprema podatke u bazu podataka



Slika 4.2: Skica aplikacije

4.1 Baza podataka

Za potrebe sustava za zamjenu soba koristit ćemo relacijsku bazu podataka koja nam omogućuje oblikovanje objekata iz stvarnog svijeta pomoću povezanih tablica - relacija. Svaka je tablica definirana vlastitim nazivom i skupom različitih atributa koji je opisuju. Glavna je zadaća baze podataka pohrana, brzo pronalaženje i dohvaćanje te dodavanje i brisanje podataka. Baza podataka ovog sustava sastoji se od entiteta:

- Student
- Oglas
- Soba
- Grad
- Dom
- Paviljon
- StudentskiCentar
- Obavijest
- Zaposlenik SC
- TrazeniUvjeti

- Lajkovi
- StudentObavijesti

4.1.1 Opis tablica

Student Entitet sadrži informacije o korisniku aplikacije - studentu. Sadrži sljedeće attribute: identifikator studenta, korisničko ime, ime, prezime, e-mail adresu, JMBAG, lozinku te oznaku za primanje mailova. Entitet je u vezi *Many-to-Many* s entitetom Obavijest preko identifikatora obavijesti, u vezi *One-to-One* s entitetom Oglas preko atributa statusa oglasa, u vezi *One-to-One* s entitetom TrazeniUvjeti preko atributa identifikatora uvjeta te u vezi *One-to-One* s entitetom Oglas.

Student		
idKorisnik	UUID	jedinstveni identifikator studenta (korisnika)
korisnickoIme	VARCHAR	jedinstveno korisničko ime
jmbag	VARCHAR	jedinstveni JMBAG studenta
ime	VARCHAR	ime studenta
prezime	VARCHAR	prezime studenta
email	VARCHAR	e-mail adresa studenta
hashLozinke	VARCHAR	hash lozinka
obavijestiNaMail	BOOLEAN	oznaka želi li student primati obavijesti na mail
viseOvlasti	BOOLEAN	aa
idStatusOglasa	BOOLEAN	oznaka potvrde
idTrazeniUvjeti	VARCHAR	traženi kriteriji za sobu za zamjenu

Oglas Entitet sadrži informacije koje su vezane uz oglas koji student predaje. Sadrži attribute: ID oglasa, naslov oglasa, opis te datum objave oglasa. Entitet je u vezi *One-to-One* s entitetom Status preko atributa identifiaktora statusa oglasa i u vezi *One-to-Many* s entitetom Obavijest preko atributa identifikatora oglasa.

Oglas		
idOglas	UUID	jedinstveni identifikator oglasa
naslov	VARCHAR	naslov oglasa
opis	VARCHAR	opis oglasa

Oglas		
objavljen	DATE	datum objavljivanja oglasa
godina	INTEGER	godina objavljivanja oglasa
<i>idStatusOglasa</i>	VARCHAR	status oglasa

Soba Entitet sadrži informacije o sobi u studentskom domu. Sadrži attribute: broj sobe, kat na kojemu se soba nalazi, broj kreveta i vrstu kupaonice koja pripada sobi, kategoriju te identifikator paviljona i doma. Entitet je u vezi *Many-to-One* s entitetom Paviljon preko atributa identifikatora paviljona i doma.

Soba		
broj	INTEGER	broj sobe
kat	INTEGER	kat na kojemu se soba nalazi
brojKreveta	VARCHAR	broj kreveta u sobi
tipKupaonice	VARCHAR	vrsta dostupne kupaonice
kategorija	VARCHAR	kategorija sobe
<i>idPaviljon</i>	UUID	identifikator paviljona kojemu soba pripada
<i>idDom</i>	UUID	identifikator doma kojemu soba pripada

Grad Entitet sadrži informacije o pojedinom gradu. Sadrži attribute: identifikator grada i naziv te identifikator studentskog centra tog grada. Entitet je u vezi *One-To-One* s entitetom Studentski Centar preko atributa identifikatora studentskog centra i u vezi *One-to-Many* s entitetom Dom preko identifikatora grada.

Grad		
idGrad	UUID	jedinstveni identifikator grada
naziv	VARCHAR	ime grada
<i>idSc</i>	UUID	identifikator gradskog studentskog centra

Dom Entitet sadrži sve važne informacije o pojedinom studentskom domu. Sadrži attribute: ID doma, naziv doma, ID grada u kojemu se dom nalazi te oznaku ima li dom vlastitu menzu. Entitet je u vezi *Many-to-One* s entitetom Grad preko atributa identifikatora grada i u vezi *One-To-Many* s entitetom Paviljon preko identifikatora doma.

Dom		
idDom	UUID	jedinstveni identifikator studentskog doma
naziv	VARCHAR	ime studentskog doma
imaMenzu	BOOLEAN	oznaka ima li dom vlastitu menzu
<i>idGrad</i>	UUID	identifikator grada u kojemu se dom nalazi

Paviljon Entitet sadrži sve informacije o pojedinom paviljonu studentskog doma. Sadrži attribute: identifikator paviljona, naziv te identifikator doma. Entitet je u vezi *Many-to-One* s entitetom Dom preko atributa identifikatora doma te u vezi *One-to-Many* s entitetom Soba preko indetifikatora paviljona.

Paviljon		
idPaviljon	UUID	jedinstveni identifikator paviljona
naziv	VARCHAR	naziv paviljona
<i>idDom</i>	UUID	identifikator doma u kojemu se paviljon nalazi

Studentski centar Entitet sadrži informacije o studentskom centru. Sadrži attribute: identifikator studentskog centra, naziv te identifikator grada u kojemu se studentski centar nalazi. Entitet je u vezi *One-to-One* s entitetom Grad preko atributa identifikatora grada te u vezi *One-to-Many* s entitetom Zaposlenik SC preko identifikatora studentskog centra.

Studentski Centar		
idSc	UUID	jedinstveni identifikator studentskog centra
naziv	VARCHAR	ime studentskog centra
<i>idGrad</i>	UUID	identifikator grada u kojemu se nalazi studentski centar

Obavijest Entitet sadrži informacije o obavijestima koje aplikacija šalje studentima. Sadrži entitete: identifikator obavijesti, tekst, oznaku je li obavijest procitana, vrijeme slanja obavijesti te listu studenata kojima se obavijest šalje i identifikator oglasa za koji se obavijest šalje. Entitet je u vezi *Many-to-Many* s entitetom Student te u vezi *Many-to-One* s entitetom Oglas preko identifikatora oglasa.

Obavijest		
idObavijest	UUID	jedinstveni identifikator obavijesti
tekst	VARCHAR	tekst obavijesti
procitana	BOOLEAN	oznaka je li poruka pročitana
vrijeme	DATE	vrijeme slanja obavijesti
<i>idOglas</i>	UUID	identifikator oglasa za koji se obavijest generira

Zaposlenik SC Entitet sadrži informacije o zaposleniku u studentskom centru. Sadrži attribute: identifikator zaposlenika, korisničko ime i lozinku za prijavu u sustav, ime i prezime zaposlenika, e-mail adresu te identifikator studentskog centra u kojemu je zaposlen.

Zaposlenik SC		
idZaposlenik	UUID	jedinstveni identifikator zaposlenika studentskog centra
korisnickoIme	VARCHAR	jedinstveno korisničko ime zaposlenika studentskog centra
ime	VARCHAR	ime zaposlenika studentskog centra
prezime	VARCHAR	prezime zaposlenika studentskog centra
email	VARCHAR	e-mail adresa zaposlenika studentskog centra
hashLozinke	VARCHAR	hash lozinke
<i>idSc</i>	UUID	identifikator studentskog centra u kojemu je zaposlen

TrazeniUvjeti		
idTrazeniUvjeti	UUID	jedinstveni identifikator skupa traženih uvjeta
brojKreveta	VARCHAR	traženi broj kreveta
tipKupaonice	VARCHAR	traženi tip kupaonice
kategorija	VARCHAR	tražena kategorija
godina	INTEGER	godina za koju se predaje oglas
komentar	VARCHAR	dodatni komentari vezani uz tražene uvjete

Lajkovi Entitet sadrži informacije vezane uz 'lajkove' oglasa. Sadrži attribute: identifikator oglasa i identifikator studenta te ocjenu. Entitet je u vezi *Many-to-One* s entitetom Student preko identifikatora studenta i u vezi *Many-to-One* s entitetom Oglas preko identifikatora oglasa.

Lajkovi		
idOglas	UUID	jedinstveni identifikator oglasa koji se ocjenjuje
idStudent	UUID	jedinstveni identifikator studenta koji je 'dao lajk'
ocjena	INTEGER	iznos ocjene

Status oglasa		
idStautsOglasa	UUID	jedinstveni identifikator statusa oglasa
status	INTEGER	aa
idOglas	UUID	identifikator oglasa
idStudent	UUID	identifikator studenta

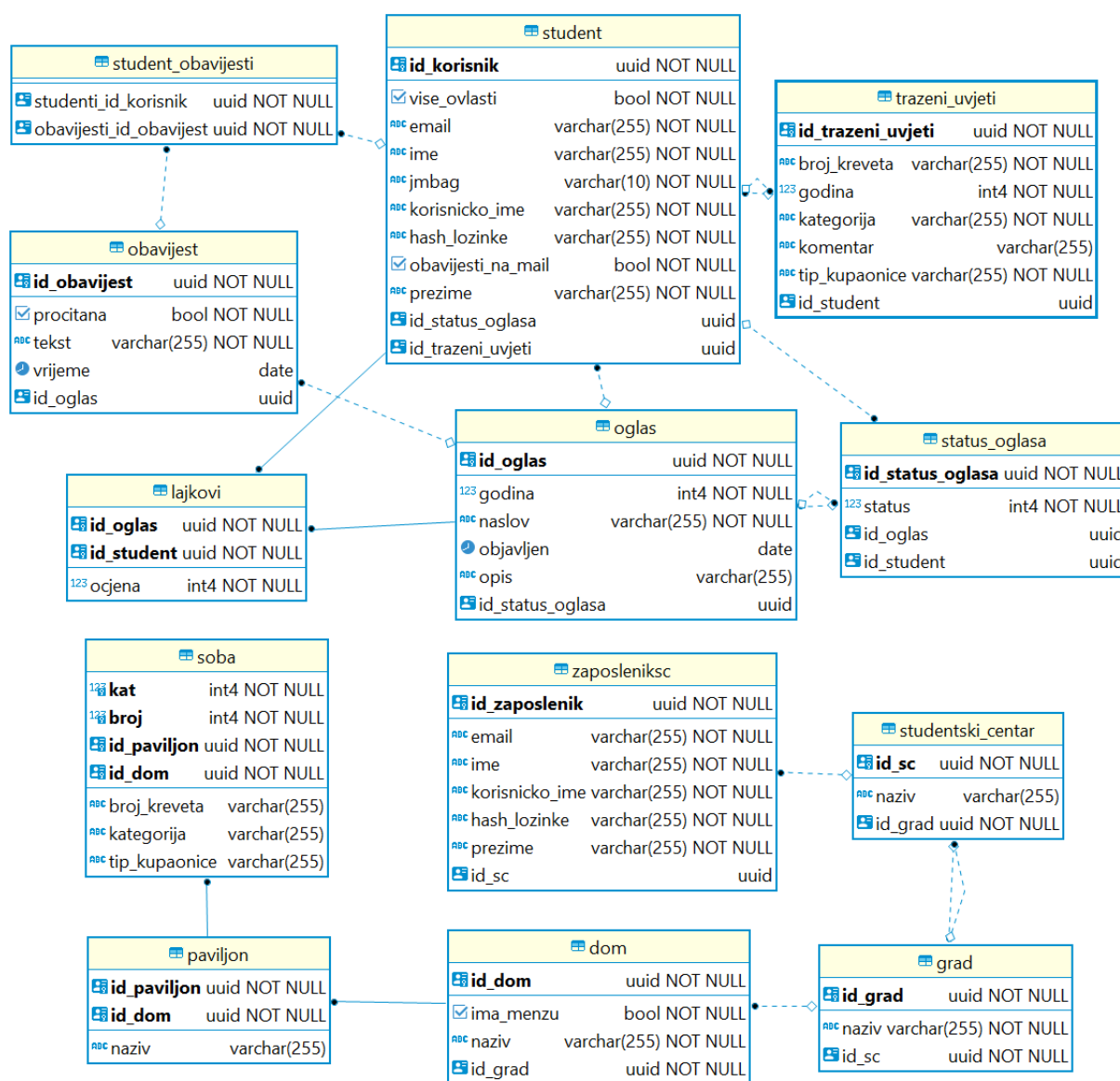
StudentObavijesti		
studentIdKorisnik	UUID	aa
obavijestiIdObavijes	UUID	aa

Broj kreveta

Oznake kategorija

Tip kupaonice

4.1.2 Dijagram baze podataka



Slika 4.3: ER dijagram baze podataka

4.2 Dijagram razreda

Potrebno je priložiti dijagram razreda s pripadajućim opisom. Zbog preglednosti je moguće dijagram razlomiti na više njih, ali moraju biti grupirani prema sličnim razinama apstrakcije i srodnim funkcionalnostima.

dio 1. revizije

*Prilikom prve predaje projekta, potrebno je priložiti potpuno razrađen dijagram razreda vezan uz **generičku funkcionalnost** sustava. Ostale funkcionalnosti trebaju biti idejno razrađene u dijagramu sa sljedećim komponentama: nazivi razreda, nazivi metoda i vrste pristupa metodama (npr. javni, zaštićeni), nazivi atributa razreda, veze i odnosi između razreda.*

dio 2. revizije

Prilikom druge predaje projekta dijagram razreda i opisi moraju odgovarati stvarnom stanju implementacije

4.3 Dijagram stanja

dio 2. revizije

*Potrebno je priložiti dijagram stanja i opisati ga. Dovoljan je jedan dijagram stanja koji prikazuje **značajan dio funkcionalnosti** sustava. Na primjer, stanja korisničkog sučelja i tijekom korištenja neke ključne funkcionalnosti jesu značajan dio sustava, a registracija i prijava nisu.*

4.4 Dijagram aktivnosti

dio 2. revizije

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

4.5 Dijagram komponenti

dio 2. revizije

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

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

dio 2. revizije

*Detaljno navesti sve tehnologije i alate koji su primijenjeni pri izradi dokumentacije i aplikacije. Ukratko ih opisati, te navesti njihovo značenje i mjesto primjene. Za svaki navedeni alat i tehnologiju je potrebno **navesti internet poveznicu** gdje se mogu preuzeti ili više saznati o njima.*

5.2 Ispitivanje programskog rješenja

dio 2. revizije

U ovom poglavlju je potrebno opisati provedbu ispitivanja implementiranih funkcionalnosti na razini komponenti i na razini cijelog sustava s prikazom odabranih ispitnih slučajeva. Studenti trebaju ispitati temeljnu funkcionalnost i rubne uvjete.

5.2.1 Ispitivanje komponenti

*Potrebno je provesti ispitivanje jedinica (engl. unit testing) nad razredima koji implementiraju temeljne funkcionalnosti. Razraditi **minimalno 6 ispitnih slučajeva** u kojima će se ispitati redovni slučajevi, rubni uvjeti te izazivanje pogreške (engl. exception throwing). Poželjno je stvoriti i ispitni slučaj koji koristi funkcionalnosti koje nisu implementirane. Potrebno je priložiti izvorni kôd svih ispitnih slučajeva te prikaz rezultata izvođenja ispita u razvojnom okruženju (prolaz/pad ispita).*

5.2.2 Ispitivanje sustava

*Potrebno je provesti i opisati ispitivanje sustava koristeći radni okvir Selenium¹. Razraditi **minimalno 4 ispitna slučaja** u kojima će se ispitati redovni slučajevi, rubni uvjeti te poziv funkcionalnosti koja nije implementirana/izaziva pogrešku kako bi se vidjelo na koji način sustav reagira kada nešto nije u potpunosti ostvareno. Ispitni slučaj se treba sastojati od ulaza (npr. korisničko ime i lozinka), očekivanog izlaza ili rezultata, koraka ispitivanja i dobivenog izlaza ili rezultata.*

Izradu ispitnih slučajeva pomoću radnog okvira Selenium moguće je provesti pomoću jednog od sljedeća dva alata:

- *dodatak za preglednik **Selenium IDE** - snimanje korisnikovih akcija radi automatskog ponavljanja ispita*
- ***Selenium WebDriver** - podrška za pisanje ispita u jezicima Java, C#, PHP koristeći posebno programsko sučelje.*

Detalji o korištenju alata Selenium bit će prikazani na posebnom predavanju tijekom semestra.

¹<https://www.seleniumhq.org/>

5.3 Dijagram razmještaja

dio 2. revizije

*Potrebno je umetnuti **specifikacijski** dijagram razmještaja i opisati ga. Moguće je umjesto specifikacijskog dijagrama razmještaja umetnuti dijagram razmještaja instanci, pod uvjetom da taj dijagram bolje opisuje neki važniji dio sustava.*

5.4 Upute za puštanje u pogon

dio 2. revizije

*U ovom poglavlju potrebno je dati upute za puštanje u pogon (engl. deployment) ostvarene aplikacije. Na primjer, za web aplikacije, opisati postupak kojim se od izvornog kôda dolazi do potpuno postavljene baze podataka i poslužitelja koji odgovara na upite korisnika. Za mobilnu aplikaciju, postupak kojim se aplikacija izgradi, te postavi na neku od trgovina. Za stolnu (engl. desktop) aplikaciju, postupak kojim se aplikacija instalira na računalo. Ukoliko mobilne i stolne aplikacije komuniciraju s poslužiteljem i/ili bazom podataka, opisati i postupak njihovog postavljanja. Pri izradi uputa preporučuje se **naglasiti korake instalacije uporabom natuknica** te koristiti što je više moguće **slike ekrana** (engl. screenshots) kako bi upute bile jasne i jednostavne za slijediti.*

Dovršenu aplikaciju potrebno je pokrenuti na javno dostupnom poslužitelju. Studentima se preporuča korištenje neke od sljedećih besplatnih usluga: Amazon AWS, Microsoft Azure ili Heroku. Mobilne aplikacije trebaju biti objavljene na F-Droid, Google Play ili Amazon App trgovini.

6. Zaključak i budući rad

dio 2. revizije

U ovom poglavlju potrebno je napisati osvrt na vrijeme izrade projektnog zadatka, koji su tehnički izazovi prepoznati, jesu li riješeni ili kako bi mogli biti riješeni, koja su znanja stečena pri izradi projekta, koja bi znanja bila posebno potrebna za brže i kvalitetnije ostvarenje projekta i koje bi bile perspektive za nastavak rada u projektnoj grupi.

Potrebno je točno popisati funkcionalnosti koje nisu implementirane u ostvarenoj aplikaciji.

Popis literature

Kontinuirano osvježavanje

Popisati sve reference i literaturu koja je pomogla pri ostvarivanju projekta.

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

Indeks slika i dijagrama

2.1	Prikaz katova i dostupnih slobodnih soba na pojedinom katu	8
2.2	Tlocrt odabranog kata	8
2.3	Detalji o sobi i studentima u sobi	9
2.4	Odbrojavanje i konačni odabir	9
3.1	Dijagrami obrazaca uporabe, funkcionalnost neregistriranog korisnika i djelatnika SC-a	17
3.2	Dijagrami obrazaca uporabe, funkcionalnost prijavljenog korisnika i timera	18
3.3	Sekvencijski dijagram za UC3	19
3.4	Sekvencijski dijagram za UC4	20
3.5	Sekvencijski dijagram za UC5	20
3.6	Sekvencijski dijagram za UC15	21
4.1	Skica sustava	22
4.2	Skica aplikacije	23
4.3	ER dijagram baze podataka	29

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

Kontinuirano osvježavanje

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

1. sastanak

- Datum: u ovom formatu: 12. studenoga 2020.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - opis prve teme
 - opis druge teme

2. sastanak

- Datum: u ovom formatu: 12. studenoga 2020.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - opis prve teme
 - opis druge teme

Tablica aktivnosti

Kontinuirano osvježavanje

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

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
Upravljanje projektom							
Opis projektnog zadatka							
Funkcionalni zahtjevi							
Opis pojedinih obrazaca							
Dijagram obrazaca							
Sekvencijski dijagrami							
Opis ostalih zahtjeva							
Arhitektura i dizajn sustava							
Baza podataka							
Dijagram razreda							
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							
Dijagram razmještaja							
Upute za puštanje u pogon							
Dnevnik sastajanja							
Zaključak i budući rad							
Popis literature							

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
<i>Dodatne stavke kako ste podijelili izradu aplikacije</i>							
<i>npr. izrada početne stranice</i>							
<i>izrada baze podataka</i>							
<i>spajanje s bazom podataka</i>							
<i>back end</i>							

Dijagrami pregleda promjena

dio 2. revizije

Prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je na kraju projekta generirane grafove s gitlaba prenijeti u ovo poglavlje dokumentacije. Dijagrami za vlastiti projekt se mogu preuzeti s gitlab.com stranice, u izborniku Repository, pritiskom na stavku Contributors.