

# Programsko inženjerstvo

Ak. god. 2022./2023.

## SatCom

Dokumentacija, Rev. 2.0

Grupa: EspressoDepresso

Voditelj: Mihael Pristav

Datum predaje: 13.1.2023.

Nastavnik: Goran Rajić

# Sadržaj

<b>1 Dnevnik promjena dokumentacije</b>	<b>3</b>
<b>2 Opis projektnog zadatka</b>	<b>8</b>
<b>3 Specifikacija programske potpore</b>	<b>11</b>
3.1 Funkcionalni zahtjevi . . . . .	11
3.1.1 Obrasci uporabe . . . . .	13
3.1.2 Sekvencijski dijagrami . . . . .	31
3.2 Ostali zahtjevi . . . . .	34
<b>4 Arhitektura i dizajn sustava</b>	<b>35</b>
4.1 Baza podataka . . . . .	36
4.1.1 Opis tablica . . . . .	36
4.1.2 Dijagram baze podataka . . . . .	40
4.2 Dijagram razreda . . . . .	41
4.3 Dijagram stanja . . . . .	43
4.4 Dijagram aktivnosti . . . . .	45
4.5 Dijagram komponenti . . . . .	47
<b>5 Implementacija i korisničko sučelje</b>	<b>49</b>
5.1 Korištene tehnologije i alati . . . . .	49
5.2 Ispitivanje programskog rješenja . . . . .	51
5.2.1 Ispitivanje komponenti . . . . .	51
5.2.2 Ispitivanje sustava . . . . .	56
5.3 Dijagram razmještaja . . . . .	60
5.4 Upute za puštanje u pogon . . . . .	61
<b>6 Zaključak i budući rad</b>	<b>64</b>
<b>Popis literature</b>	<b>65</b>
<b>Indeks slika i dijagrama</b>	<b>67</b>

**Dodatak: Prikaz aktivnosti grupe**

**68**

# 1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak. Popunjeni osnovni podaci. Osvježen dnevnik sastajanja	Mihael Pristav	23.10.2022.
0.1.1	Osvježen dnevnik sastajanja	Mihael Pristav	25.10.2022.
0.1.2	Osvježen dnevnik sastajanja	Mihael Pristav	26.10.2022.
0.2	Započeta cjelina Arhitektura i dizajn. Opis tablica baze podataka	Marta Vidas Ivan Žgela	28.10.2022.
0.3	Započeta cjelina Opis projektnog zadatka	Maria Carmen Belušić Gonzalez	30.10.2022.
0.3.1	Osvježen dnevnik sastajanja	Mihael Pristav	30.10.2022.
0.3.2	Osvježen dnevnik sastajanja	Ivan Žgela	31.10.2022.
0.4	Započeta cjelina funkcionalni zahtjevi	Leona Salihović	1.11.2022.
0.4.1	Osvježen dnevnik sastajanja Ispravljene prijašnje greške u dokumentaciji	Mihael Pristav	2.11.2022.
0.4.2	Dodan sadržaj u cjelinu funkcionalni zahtjevi	Leona Salihović	3.11.2022.
0.4.3	Promijenjen sadržaj u cjelini funkcionalni zahtjevi	Leona Salihović	3.11.2022.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
0.4.4	Osvježen dnevnik sastajanja Ispravljene prijašnje greške u dokumentaciji Dodani dijagrami obrazaca uporabe	Mihael Pristav	3.11.2022.
0.4.5	Dorađen opis postojećih tablica baze podataka i dodane nove	Marta Vidas	7.11.2022.
0.4.6	Upisano pola obrazaca uporabe Dopunjena tablica uložениh sati Popravljen uređjenje dijelova dokumentacije	Mihael Pristav Marta Vidas	7.11.2022.
0.4.7	Osviježen opis baze podataka	Mihael Pristav	8.11.2022.
0.4.8	Dodani obrasci uporabe	Mihael Pristav	9.11.2022.
0.5	Osvježeni obrasci uporabe Osviježen dnevnik sastajanja Dodani prvi sekvencijski dijagrami Ispravljene primjećene greške u tekstu Ispravljena neka upozorenja i pogreške u Latex sintaksi	Mihael Pristav	11.11.2022.
0.5.1	Osvježeni obrasci uporabe Podešeni sekvencijski diagrami za bolju preglednost i čitljivost Dodani opisi sekvencijskih dijagrama Podešena custom naredba za referenciranje i njezino korištenje u dokumentaciji	Mihael Pristav	12.11.2022.
0.5.2	Manje promjene u oblikovanju stranica Popravljen 2. sekvencijski dijagram	Mihael Pristav	13.11.2022.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
0.5.3	Dopunjeni funkcionalni zahtjevi običnog uporabnog korisnika Osvježen opis i atributi baze Dodani obrasci uporabe	Mihael Pristav	14.11.2022.
0.5.4	Osvježen dijagram baze Dodani obrasci uporabe	Mihael Pristav	15.11.2022.
0.5.5	Dodavanje obrazaca uporabe za sate- lite (prikaz liste satelita, edit, delete i dodavanje novog satelita) Dodani sati provedeni na kodiranje aplikacije	Ivan Žgela	15.11.2022.
0.5.6	Dodavanje obrazaca uporabe za prikaz informacija satelita Ispravljene pogrešno podešene reference Ispravljene manje greške u oblikovanju Osvježeni opisi obrazaca uporabe u odnosu na novi dijagram obrazaca uporabe	Mihael Pristav	15.11.2022.
0.5.7	Dodavanje obrazaca uporabe za tran- smitere (prikaz liste transmitera, edit, delete i dodavanje novog tran- smitera)	Leona Salihović	15.11.2022.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
0.5.8	Dodani dijagrami razreda i njihovi opisi Proširen opis projektnog zadatka ispravljene greške u svom dosadašnjem tekstu popravljen formatiranje teksta u dokumentu	Maria Carmen Belušić Gonzales Leticija Crnković Mihael Pristav Marta Vidas	17.11.2022.
1.0	Dodavanje dosadašnje literature Završna kontrola teksta	Mihael Pristav	18.11.2022.
1.0.1	Ispravljene manje greške u opisu baze podataka	Mihael Pristav	6.12.2022.
1.0.2	Dopunjeni ostali zahtjevi	Mihael Pristav	9.1.2023.
1.0.3	Usklađen opis zadatka s trenutnim stanjem aplikacije	Mihael Pristav	9.1.2023.
1.0.4	Započeta cjelina Korištene tehnologije i alati	Leona Salihović	9.1.2023.
1.0.5	Promjene u cjelini Korištene tehnologije i alati	Leona Salihović	10.1.2023.
1.1.0	Započeta cjelina Zaključak	Leona Salihović	11.1.2023.
1.1.1	Dovršena Aktualizacija obrazaca uporabe	Mihael Pristav	11.1.2023.
1.1.2	Promjene u cjelinama Zaključak i Korištene tehnologije i alati	Leona Salihović	12.1.2023.
1.2.0	Započeta cjelina Dijagram razmještaja	Leona Salihović	12.1.2023.
1.2.1	Dodan dijagram razmještaja	Leona Salihović	12.1.2023.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
1.2.2	Ispravljenenovo nastale pogreške kod referenciranja i foramtiranja dokumenta	Mihael Pristav	12.1.2023.
1.3.0	Dodana cjelina ispitivanje komponenti	Marta Vidas	12.1.2023.
1.3.1	Osvježeni dijagrami razreda	Leticija Crnković	12.1.2023.
1.3.2	Osvježeni sekvencijski dijagrami Ispravljeno referenciranje slika	Mihael Pristav	13.1.2023.
1.4.0	Dodana cjelina za Dijagram komponenti Dopunjen Dijagram aktivnosti	Dalen Grdić	13.1.2023.
1.4.1	Dodana cjelina Ispitivanje sustava	Maria Carmen Belušić Gonzalez	13.1.2023.
1.5.0	Dovršene upute za puštanje u pogon	Leticija Crnković	13.1.2023.
2.0	Dodavanje grafova aktivnosti na gitLabu Završna kontrola teksta	Mihael Pristav	13.1.2023.



## 2. Opis projektnog zadatka

Cilj ovog projekta je izrada programske potpore za web aplikaciju *SatCom* koja svojim korisnicima omogućuje olakšan i intuitivan način komunikacije sa simuliranim satelitima uz pomoć *SatNOGS* mreže. Zainteresirani korisnici mogu biti dio istraživačke, znanstvene i akademske zajednice ili bilo koja skupina zainteresirana za testiranje komunikacije sa satelitima.

Zbog velike koristi satelita, njihove široke primjene i povećane troškovne dostupnosti, u posljednjem desetljeću pojavljuje se sve veći broj satelita otvorenog, akademskog tipa, a uz to i pitanje ostvarivanja komunikacije s njima. Općenito, sateliti komuniciraju koristeći se radio komunikacijskim vezama za razošiljanje poruka prema zemaljskim stanicama, nakon čega stanice prime poruku i obrađuju informacije koje poruka sadrži (podatci o satelitu, lokacija satelita, satelitske slike i sl).

U aplikaciji postoje tri uloge:

- običan uporabni korisnik,
- administrator aplikacije,
- administrator satelita.

Funkcionalnosti kojima može pristupiti *običan uporabni korisnik* mogu pristupati i korisnici s preostalim ulogama. Razlika je što korisnici s ulogama *administrator aplikacije* i *administrator satelita* imaju pristup dodatnim funkcionalnostima.

Registraciju novih korisnika može obaviti samo korisnik s ulogom *administrator aplikacije* klikom na gumb *Add New User* na stranici *Users*. Pri registriranju novog korisnika potrebno je upisati e-mail adresu, korisničko ime, lozinku i dodijeliti mu jednu od uloga.

Osim registracije korisnika, korisnik s ulogom *administrator aplikacije* može pregledati sve korisnike aplikacije i brisati ih putem stranice *Users* na kojoj se prikazuje lista svih korisnika aplikacije u tabličnom prikazu. Svaki redak tablice sadrži ikonu za brisanje odabranog korisnika.

Neregistrirani korisnici ne mogu pristupiti funkcionalnostima aplikacije.

Registriranom i prijavljenom korisniku omogućene su razne aktivnosti u aplikaciji. Korisnik može pristupiti početnoj stranici na kojoj je opisan cilj *SatCom*

projekta i objašnjenje osnovnih pojmova koji se koriste u aplikaciji:

- satelit,
- zemaljska stanica,
- komunikacijski link,
- transponder.

Korisnik inicijalizira postupak slanja poruka na satelit klikom na gumb

*Send message* koji se nalazi u zaglavlju i na početnoj stranici.

Stranica *Send Message* sadrži popis svih satelita i njihovih atributa u tabličnom prikazu i omogućuje korisniku pretraživanje satelita putem tražilice na vrhu tablice. Korisnik odabire satelit s kojim želi ostvariti komunikaciju klikom na redak u tablici. Nakon odabira stanice, korisniku se na vrhu stranice prikaže odabrani satelit, njegove informacije i informacije o njegovim transponderima, a ispod se nalazi lista svih kompatibilnih komunikacijskih linkova od kojih mora odabrati jedan. Potom korisnik mora odabrati želi li odabir zemaljske stanice ostvariti automatski ili manualno. Automatski odabir podrazumijeva da se o odabiru zemaljske stanice brine sustav. Kod manualnog, korisniku se prepušta odabir jedne od kompatibilnih zemaljskih stanica. U zadnjem koraku korisnik unosi poruku koju želi poslati i zatim čeka odgovor od satelita.

Običan korisnik također može pristupiti svom profilu na stranici *My Profile* koja sadrži korisnikove osobne podatke i gumbove kojima pristupa uređivanju profila te pregledu povijesti poslanih zahtjeva komunikacije sa satelitima. Korisnik može mijenjati korisničko ime i lozinku klikom na gumb *Edit profile*. Pregled poslanih zahtjeva nalazi se na stranici *Message History*, gdje se korisniku prikazuje zapis satelita s kojima je uspješno komunicirao, kao i svi parametri komunikacije (vrijeme slanja poruke, zemaljska stanica, komunikacijski link te sadržaj poruke i odgovora). Povijest komunikacija može se obrisati klikom na gumb *Erase All*.

Korisnik s ulogom *administrator satelita* može obavljati razne akcije nad satelitima, komunikacijskim linkovima i transponderima.

Satelitima pristupa putem stranice *Satellites* na kojoj su prikazani svi sateliti i njihovi atributi u tabličnom prikazu. Na toj stranici, administratoru satelita ponuđene su opcije pregleda podataka, dodavanja novih satelita, brisanja i uređivanja satelita. Klikom na redak u tablici, otvara se stranica *Satellite Details*. Na vrhu navedene stranice nalaze se svi atributi odabranog satelita, a ispod lista povezanih transpondera i njihove informacije. Klikom na jedan od transpondera administratoru se prikazuju njegova svojstva i opcije za brisanje i promjenu podataka.

Opcijama se pristupa gumbima *Delete transmitter* i *Edit*. Formi za dodavanje satelita pristupa se putem gumba *Add New Satellite* gdje se upisuju parametri satelita. Brisanje i uređivanje satelita obavlja se klikom na gumbove koji se nalaze na vrhu stranice s podacima o satelitu. Pri uređivanju satelita, administrator satelita može uređivati njegove podatke i dodavati i brisati njegove transmitere.

Tabličnom prikazu liste komunikacijskih linkova i njihovih informacija administrator pristupa putem stranice *Links*. Na toj stranici administrator satelita može dodavati nove satelite klikom na gumb *Add New link* čime pristupa formi u koju upisuje podatke novog linka. Klikom na link, prikazuju se informacije o tom linku i opcije za brisanje i uređivanje odabranog linka pomoću gumbova *Delete* i *Edit* koji se nalaze na kraju svakog retka tablice.

Zadnje, administrator može zatražiti prikaz svih transmitera putem stranice *Transmitters*. Tamo može odabrati transmitere, te ih potom brisati i uređivati.

Bitno je napomenuti da zemaljske stanice ne može niti jedan korisnik mijenjati ili brisati. Sustav se svakog dana u ponoć spaja na *SatNOGS* aplikacijsko sučelje i dohvaća podatke aktivnih zemaljskih stanica.

Postoje razne mogućnosti proširenja sadašnjeg rješenja koje bi poboljšale aplikaciju, ali nisu nužne za njenu funkcionalnost:

- ostvarenje komunikacije sa stvarnim satelitima koji se nalaze na *SatNOGS* mreži,
- registriranje vlastitih satelita putem aplikacije na *SatNOGS* mrežu,
- omogućavanje korisnicima da satelite na koje često šalju poruke mogu označiti kao *omiljene*,
- implementacija mobilne aplikacije,
- slanje zahtjeva administratoru stranice za promjenu e-mail adrese,
- slanje zahtjeva za registracijom.

## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

#### Dionici:

1. Vlasnik (naručitelj)
2. Korisnici
  - (a) opći uporabni korisnik
  - (b) administrator aplikacije
  - (c) administrator satelita
3. Razvojni tim

#### Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik (inicijator) može:
  - (a) se prijaviti u sustav, za što su mu potrebni lozinka (*password*) i korisničko ime (*username*)
2. Obični uporabni korisnik (inicijator) može:
  - (a) se prijaviti u sustav unosom lozinke (*password*) i korisničkog imena (*username*) te promijeniti inicijalo mu dodijeljenu lozinku i korisničko ime
  - (b) odabrati satelit na kojeg želi poslati poruku i link za komunikaciju
  - (c) odlučiti hoće li samostalno odabrati zemeljsku stanicu putem koje se komunicira sa satelitom ili će to sustav napraviti umjesto njega
  - (d) poslati željeni tekst kao poruku na prethodno odabranu točku (satelit, link, stanica)
  - (e) pregledavati i brisati sve poslane poruke i primljene odgovore
3. Administrator aplikacije (inicijator) može:
  - (a) kreirati nove korisnike sustava pri čemu im dodijeljuje adresu e-pošte (*e-mail*), korisničko ime (*username*), lozinku (*password*) i ulogu (*role*)
  - (b) ostvariti sve funkcionalne zahtjeve običnih uporabnih korisnika

4. Administrator satelita (inicijator) može:

- (a) kreirati i izbrisati satelite te promijenjiti njihove parametre
- (b) kreirati i izbrisati linkove te promijenjiti njihove parametre
- (c) kreirati i izbrisati transmitere te promijenjiti njihove parametre
- (d) ostvariti sve funkcionalne zahtjeve običnih uporabnih korisnika

5. Baza podataka (sudionik) može:

- (a) pohraniti sve podatke o korisnicima, njihovim ovlastima i razmijenjenim porukama
- (b) pohraniti sve podatke o satelitima, zemaljskim stanicama, linkovima, antenama i transponderima

6. SatNOGS Network API (sudionik) može:

- (a) dohvatiti podatke o stanicama i antenama

7. Satelit server (sudionik) može:

- (a) primiti poruke koje generira korisnik
- (b) odlučiti o prihvatanju iste poruke
- (c) generirati povratnu poruku korisniku

### 3.1.1 Obrasci uporabe

#### UC1 - Prijava

- **Glavni sudionik:** Neregistrirani korisnik (*guest\_user*)
- **Cilj:** Dobiti pristup korisničkom sučelju
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Postojanje korisničkog računa u bazi
- **Opis osnovnog tijeka:**
  1. Unos korisničkog imena (*username*) i lozinke (*password*)
  2. Sustav potvrđuje ispravnost unesenih podataka
  3. Sustav omogućava pristup korisničkim funkcijama koje definira uloga (*role*) prijavljenog korisnika
- **Opis mogućih odstupanja:**
  - 1 Neispravno korisničko ime i/ili lozinka
    - 1.1 Sustav obavještava korisnika o neuspjelom upisu i prikazuje prikladnu poruku: "Username or password incorrect"

#### UC2 - Pregled osobnih podataka

- **Glavni sudionik:** Korisnik (*user*)
- **Cilj:** Pregledati osobne podatke
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen
- **Opis osnovnog tijeka:**
  1. Korisnik odabire opciju "User details"
  2. Sustav pokazuje osobne podatke korisnika (*username*, *e-mail* i *role*)

#### UC3 - Promjena osobnih podataka

- **Glavni sudionik:** Korisnik (*user*)
- **Cilj:** Promijeniti osobne podatke
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen

- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju "Edit user details"
  2. Sustav korisniku nudi formu za popunjavanje u kojoj su njegovi trenutni podaci
  3. Korisnik mijenja svoje osobne podatke (*username* i/ili *password*)
  4. Korisnik odabere opciju "Save"
  5. Sustav ažurira bazu podataka
- **Opis mogućih odstupanja:**
  - 1 Korisnik promijeni svoje osobne podatke, ali ne odabere opciju "Save"
    - 1.1 Sustav obavještava korisnika da nije spremio podatke prije izlaska iz prozora
  - 2 Korisnik mijenja *username*, a on je već zauzet
    - 2.1 Sustav obavještava korisnika da je *username* već zauzet i prikazuje prikladnu poruku: "Username already taken"

#### UC4 - Slanje poruke na satelit

- **Glavni sudionik:** Korisnik (*user*)
- **Cilj:** Slanje poruke na odabrani satelit i određivanje parametara
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen
  2. U bazi podataka postoji kompatibilni trio (satelit, link i stanica)
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju za slanje poruke na satelit
  2. Sustav ispisuje listu satelita s nekim njihovim informacijama
  3. Korisnik odabere jedan od satelita
  4. Sustav ispisuje sve podatke o satelitu i njegovim transponderima
  5. Sustav ispod informacija o satelitu ispisuje listu kompatibilnih linkova sa svim njihovim informacijama (*frequency, mode, baud*)
  6. Korisnik odabire jedan od linkova
  7. Sustav nudi korisniku opciju da sam odabere željenu stanicu ili da odabir prepusti sustavu
  8. Odabire se kompatibilni satelit ( Vidi UC5)
  9. Sustav vodi korisnika na stranicu s poljem za unošenje teksta poruke

10. Korisnik unosi poruku i odabire "Send"
  11. Sustav provjerava sa SatNOGS mrežom podudaraju li se naši podaci o odabranoj stanici s trenutnim podacima u njihovoj bazi podataka
  12. Sustav generira sadržaj poruke na temelju vremena slanja, odabranog linka i unesenog teksta
  13. Poruka se označava kao *UPLOAD* i pridružuju joj se parametri komunikacije
  14. Sustav šalje poruku na server(satelit), koji generira povratnu informaciju sličnog formata s oznakom *DOWNLOAD*
  15. Obje poruke se spremaju u bazu podataka s ID oznakom korisnika
- **Opis mogućih odstupanja:**
    - 1 U bazi podataka ne postoji niti jedan satelit
      - 1.1 Sustav obavještava korisnika o nepostojanju satelita
    - 2 Poruka poslana na satelit nema definirane sve parametre ili je praznog sadržaja
      - 2.1 Sustav u bazu podataka sprema poruku označenu kao *FAILED UPLOAD*
      - 2.2 Satelit ne generira povratnu poruku zbog manjka potrebnih informacija o komunikaciji
    - 3 U bazi podataka ne postoji niti jedan link s odgovarajućim atributima za odabrani satelit
      - 3.1 Sustav obavještava korisnika o nepostojanju odgovarajućeg linka
      - 3.2 Sustav vraća korisnika na odabir satelita
    - 4 Sustav je tijekom provjere aktualnosti informacija o stanici, pronašao razliku u podacima bitnim za komunikaciju
      - 4.1 Sustav obavještava korisnika o nemogućnosti slanja poruke odabranim parametrima te ga potiče da pokuša ponovo
      - 4.2 Sustav osvježava podatke o stanicama i njihovim antenama u bazi podataka

### UC5 -Biranje stanice

- **Glavni sudionik:** Korisnik (*user*)
- **Cilj:** Odabir zemaljske stanice putem koje se planira poslati poruka
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen



2. Korisnik je odabrao željeni satelit i link

- **Opis osnovnog tijeka:**

1. Korisnik ima opciju odabrati "Manual selection" ili "Automatic selection" za odabir stanice
2. Kod automatskog pretraživanja, sustav odabire stanicu s najvećim brojem obzervacija, koja je kompatibilna s odabranim linkom
3. Kod ručnog odabira:
  - a) Sustav ispisuje listu stanica čije antene podržavaju komunikaciju na način propisan odabranim linkom
  - b) Korisnik odabire stanicu kojom želi poslati poruku
4. Nastavljajući proces u UC4, sustav vodi korisnika na stranicu s poljem za unošenje teksta poruke.

- **Opis mogućih odstupanja:**

- 1 U bazi podataka ne postoji niti jedna stanica, koja podržava komunikaciju definiranu odabranim linkom
  - 1.1 Sustav obavještava korisnika o nepostojanju adekvatne stanice
  - 1.2 Sustav vraća korisnika na ponovni izbor satelita (Vidi UC4)

### UC6 - Pregled poslanih zahtjeva i rezultata

- **Glavni sudionik:** Korisnik (*user*)
- **Cilj:** Pregled vlastitih komunikacija putem mreže
- **Sudionici:** Baza podataka
- **Preduvjet:**

1. Korisnik je prijavljen

- **Opis osnovnog tijeka:**

1. Korisnik odabere opciju "Message history"
2. Sustav ispisuje tablicu prošlih komunikacija korisnika
3. U tablici korisnik može vidjeti tekst poruke, vrijeme slanja poruke, smjer komunikacije (*UPLOAD/DOWNLOAD/FAILED UPLOAD*), imena satelita i stanice putem koje se komuniciralo te informacije o korištenom linku

- **Opis mogućih odstupanja:**

- 1 Korisnik nema nikakvih prijašnjih poruka
  - 1.1 Sustav obavještava korisnika o nepostojanju prijašnje komunikacije sa satelitima

### UC7 - Brisanje poslanih zahtjeva

- **Glavni sudionik:** Korisnik (*user*)
- **Cilj:** Brisanje kompletne povjesti vlastitih poruka
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen
  2. Korisnik u bazi ima zabilježenu barem jednu poruku
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju "Erase all"
  2. Sustav provjerava je li korisnik siguran da želi obrisati sve poruke
  3. Sustav iz baze briše sve poruke povezane s korisnikom
- **Opis mogućih odstupanja:**
  - 1 korisnik odabere opciju za brisanje, ali odustane od brisanja kod potvrde
    - 1.1 Sustav obavještava Korisnika da se brisanje nije dogodilo

### UC8 - Pregled korisnika

- **Glavni sudionik:** Administrator stranice (*super\_admin*)
- **Cilj:** Pregledati popis korisnika i njihovih podataka
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator stranice
- **Opis osnovnog tijeka:**
  1. Administrator stranice odabere opciju "Users"
  2. Sustav ispisuje listu svih ostalih korisnika s njihovim osobnim podacima
- **Opis mogućih odstupanja:**
  - 1 Ne postoje drugi korisnici
    - 1.1 Sustav Ispod prazne tablice korisnika ispisuje obavijest:  
"Application has no other users."

### UC9 - Brisanje odabranog korisnika

- **Glavni sudionik:** Administrator stranice (*super\_admin*)
- **Cilj:** Izbrisati podatke o korisniku iz baze podataka
- **Sudionici:** Baza podataka

- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator stranice
  2. Postoji barem jedan drugi korisnik u bazi podataka
- **Opis osnovnog tijeka:**
  1. Administrator stranice odabere željenog korisnika na listi korisnika
  2. Administrator stranice odabere opcije "Erase"
  3. Administrator stranice potvrđuje svoj odabir
  4. Sustav briše podatke u bazi vezane uz korisnika (osobni podaci i zabilježene poruke)
- **Opis mogućih odstupanja:**
  - 1 Administrator stranice odabere korisnika, ali odustane od brisanja kod potvrde
    - 1.1 Sustav obavještava administratora da se brisanje nije dogodilo

#### UC10 - Dodavanje novog korisnika

- **Glavni sudionik:** Administrator stranice (*super\_admin*)
- **Cilj:** Kreiranje i dodavanje novog korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator stranice
- **Opis osnovnog tijeka:**
  1. Administrator stranice odabere opciju "Add new user"
  2. Sustav administratoru nudi formu za popunjavanje
  3. Administrator stranice unosi osnovne podatke o novom korisniku ( *username*, *e-mail*, *password* i *role*)
  4. Administrator stranice odabere opciju "Create"
  5. Sustav u bazi podataka stvara novog korisnika
- **Opis mogućih odstupanja:**
  - 1 Administrator stranice odabere *e-mail* koji već koristi neki drugi korisnik
    - 1.1 Sustav provjerava jedinstvenost ovih podataka u bazi i vraća Administratora stranice na unos podataka s porukom o pogrešci: "E-mail already in use"
  - 2 Administrator stranice odabere *username* koji već koristi neki drugi korisnik

- 2.1 Sustav provjerava jedinstvenost ovih podataka u bazi i vraća Administratora stranice na unos podataka s porukom o pogrešci:  
"Username already in use"

### UC11 - Prikaz liste transmitera

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Pregled liste već postojećih transmitera
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Transmitters"
  2. Sustav ispisuje listu transmitera i njihovih atributa (*name, frequency, mode i baud*)
  3. Administrator satelita može odabrati pojedini transmieter( Vidi UC24)
- **Opis mogućih odstupanja:**
  - 1 U bazi podataka ne postoji niti jedan transmieter
    - 1.1 Sustav Ispod prazne tablice transmitera ispisuje obavijest:  
"Application has no registered Transmitters."

### UC12 - Dodavanje novog transmitera

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Stvaranje novog transmitera
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
- **Opis osnovnog tijeka:**
  1. Administratr satelita odabere opciju "Add Transmitter"
  2. Sustav administratoru nudi formu za popunjavanje podataka
  3. Administrator satelita unosi podatke o novom transmieteru(*name, frequency, mode i baud*)
  4. Administrator satelita odabere opciju "Submit"
  5. Sustav kreira novi zapis u bazi podataka na temelju unesenih podataka.
  6. Ako postoji link s identičnim atributima, sustav zapisuje *linkId* tog linka u naš zapis

7. Sustav vraća administratora na prikaz satelita (Vidi UC26)

- **Opis mogućih odstupanja:**

- 1 U bazi podataka postoji transponder koji ima isto ime
  - 1.1 Sustav ne dozvoljava korisniku da pritisne "Submit"
  - 1.2 Sustav obavještava administratora porukom "Transmitter with this name already exists"

### UC13 - Brisanje transpondera

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Brisanje postojećeg transpondera
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
  2. Postoji barem jedan transponder u bazi podataka
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Delete" na odabranom transponderu
  2. Sustav provjerava je li administrator siguran da želi obrisati transponder
  3. Sustav iz baze briše sve podatke o transponderu
  4. Sustav vraća Administratora na prikaz liste transpondera (Vidi UC11)
- **Opis mogućih odstupanja:**
  - 1 Administrator stranice odabere transponder, ali odustane od brisanja kod potvrde
    - 1.1 Sustav obavještava administratora da se brisanje nije dogodilo

### UC14 - Promjena parametara transpondera

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Promjena parametara transpondera
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
  2. U bazi postoji barem jedan transponder
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Edit"

2. Sustav administratoru nudi formu za popunjavanje u kojoj su trenutni podaci o transponderu
  3. Administrator satelita mijenja podatke o transponderu (*name, frequency, mode* i *baud*)
  4. Administrator satelita odabere opciju "Submit"
  5. Sustav mijenja zapis u bazi podataka na temelju unesenih podataka
- **Opis mogućih odstupanja:**
    - 1 U bazi podataka postoji drugi transponder koji ima isto ime
      - 1.1 Sustav ne dozvoljava korisniku da pritisne "Submit"
      - 1.2 Sustav obavještava administratora porukom "Transmitter with this name already exists"

### UC15 - Prikaz liste linkova

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Pregled liste već postojećih linkova
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Links"
  2. Sustav ispisuje listu linkova i njihovih atributa(*frequency, mode* i *baud*)
  3. Administrator satelita može odabrati pojedini link( Vidi UC25)
  4. Iznad same tablice Administratoru satelita se nudi opcija "Add Link"
- **Opis mogućih odstupanja:**
  - 1 U bazi podataka ne postoji niti jedan link
    - 1.1 Sustav ispod prazne tablice linkova ispisuje obavijest:  
"Application has no registered Links."

### UC16 - Dodavanje novog linka

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Stvaranje novog linka
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
- **Opis osnovnog tijeka:**

1. Administrator satelita odabere opciju "Add Link"
  2. Sustav administratoru nudi formu za popunjavanje podataka
  3. Administrator satelita unosi podatke o novom linku( *frequency*, *mode*, *baud*)
  4. Administrator satelita odabere opciju "Submit"
  5. Sustav kreira novi zapis u bazi podataka na temelju unesenih podataka i administratorovog Id-a
  6. Sustav vraća administratora na prikaz liste linkova (Vidi UC15)
- **Opis mogućih odstupanja:**
    - 1 U bazi podataka postoji link koji sadrži iste podatke za sva tri unesena podatka
      - 1.1 Sustav ne dozvoljava korisniku da pritisne "Submit"
      - 1.2 Sustav obavještava administratora porukom "Link with these attributes already exists"

#### UC17 - Brisanje linka

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Brisanje postojećeg linka
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
  2. Postoji barem jedan link u bazi podataka
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Delete" na odabranom linku
  2. Sustav provjerava je li administrator siguran da želi obrisati link
  3. Sustav u bazi podataka briše sve podatke o linku
  4. Sustav vraća administratora na prikaz liste linkova (Vidi UC15)
- **Opis mogućih odstupanja:**
  - 1 Administrator odabere link, ali odustane od brisanja kod potvrde
    - 1.1 Sustav obavještava administratora da se brisanje nije dogodilo

#### UC18 - Promjena parametara linka

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Promjena parametara linka
- **Sudionici:** Baza podataka

- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
  2. U bazi postoji barem jedan link
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Edit"
  2. Sustav administratoru nudi formu za popunjavanje u kojoj su trenutni podaci o linku
  3. Administrator satelita mijenja podatke o linku (*frequency, mode, baud*)
  4. Administrator satelita odabere opciju "Submit"
  5. Sustav mijenja zapis u bazi podataka na temelju unešenih podataka
  6. Sustav vraća administratora na prikaz liste linkova (Vidi UC15)
- **Opis mogućih odstupanja:**
  - 1 U bazi podataka postoji link s drugim *linkId*-em koji sadrži iste parametre
    - 1.1 Sustav ne dozvoljava korisniku da pritisne "Submit"
    - 1.2 Sustav obavještava administratora porukom "Link with these attributes already exists"

#### UC19 - Prikaz liste satelita

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Pregled liste već postojećih satelita
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Satellites"
  2. Sustav ispisuje popis postojećih satelita te njihove osnovne attribute
  3. Administrator može kliknuti na zapis satelita u prikazanoj tablici i svi njegovi podaci će biti prikazani ( Vidi UC26)
  4. Iznad same tablice administratoru satelita se nudi opcija "Add Satellite"
- **Opis mogućih odstupanja:**
  - 1 U bazi podataka ne postoji niti jedan satelit
    - 1.1 Sustav ispod prazne tablice linkova ispisuje obavijest:  
"Application has no registered Satellites."

#### UC20 - Dodavanje novog satelita



- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Stvaranje novog satelita
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Add satellite"
  2. Sustav Administratoru nudi formu za popunjavanje podataka
  3. Administrator satelita unosi podatke o novom satelitu (*name*)
  4. Administrator satelita odabere opciju "Submit"
  5. Sustav kreira novi zapis u bazi podataka na temelju unesenih podataka i Administratorovog Id-a
  6. Sustav vraća Administratora na prikaz liste satelita
- **Opis mogućih odstupanja:**
  - 1 U bazi podataka postoji satelit koji ima isto ime
    - 1.1 Sustav ne dozvoljava korisniku da pritisne "Submit"
    - 1.2 Sustav obavještava administratora porukom "Satellite with this name already exists"

### UC21 - Brisanje satelita

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Brisanje postojećeg satelita
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
  2. Postoji barem jedan satelit u bazi podataka
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabire opciju "Delete" na odabranom satelitu
  2. Sustav provjerava je li administrator siguran da želi obrisati satelit
  3. Sustav iz baze podataka briše sve podatke o satelitu
  4. Sustav vraća administratora na prikaz liste satelita (Vidi UC19)
- **Opis mogućih odstupanja:**
  - 1 Administrator satelita odabere satelit, ali odustane od brisanja kod potvrde
    - 1.1 Sustav obavještava administratora da se brisanje nije dogodilo

### UC22 - Promjena parametara satelita

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Promjena parametara satelita
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
  2. U bazi postoji barem jedan satelit
- **Opis osnovnog tijeka:**
  1. Administrator satelita odabere opciju "Edit"
  2. Sustav administratoru nudi formu za popunjavanje u kojoj su trenutni podaci o satelitu
  3. Administrator satelita mijenja podatke o satelitu (*name*)
  4. Administrator satelita odabere opciju "Submit"
  5. Sustav mijenja zapis u bazi podataka na temelju unesenih podataka
  6. Sustav vraća administratora na prikaz liste satelita (Vidi UC19)
- **Opis mogućih odstupanja:**
  - 1 U bazi podataka postoji satelit sa drugim *satelliteId*-em koji sadrži isto ime
    - 1.1 Sustav ne dozvoljava korisniku da pritisne "Submit"
    - 1.2 Sustav obavještava administratora porukom "Satellite with this name already exists"

### UC23 - Osvježavanje informacija o stanicama

- **Glavni sudionik:** Baza podataka (*Data base*)
- **Cilj:** Osvježavanje tablice stanica i tablice antena
- **Sudionici:** Satnogs network api
- **Preduvjet:**
  1. Vrijeme sustava je ponoć(00:00)
- **Opis osnovnog tijeka:**
  1. Sustav se spaja na SatNOGS Network API
  2. Sustav briše sve podatke o satelitima i antenama iz baze podataka (uključujući tablice linkAntenna i statAntenna)
  3. Sustav šalje Get zahtjev za stanice SatNOGS Network API-ju
  4. SatNOGS Network API vraća listu stanica sa svim njihovim atributima

5. Sustav parsira podatke o svakoj stanici i stvara njihove zapise: (*Id, name, number of observations, latitude, longitude, altitude* i *status*)
  6. Sustav također na temelju podataka o anteni stvara nove zapise o antenama (*type, min frequency, max frequency*)
  7. Sustav pohranjuje nove podatke u bazu podataka
- **Opis mogućih odstupanja:**
    - 1 Sustav se ne uspijeva spojiti na SatNOGS Network API
      - 1.1 Sustav prekida s procesom osvježavanja tablica
      - 1.2 Sustav pokušava ponoviti proces sljedećeg dana u ponoć(00:00)

#### UC24 - Prikaz transmitera

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Pregled postojećeg transmitera
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
- **Opis osnovnog tijeka:**
  1. Sustav ispisuje attribute transmitera (*name, frequency, mode* i *baud*) i satelit na kojem se transponder nalazi
  2. U kutu postoje dvije opcije ("Edit" i "Delete")

#### UC25 - Prikaz linka

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Pregled postojećeg linka
- **Sudionici:** Baza podataka
- **Preduvjet:**
  1. Korisnik je prijavljen kao administrator satelita
- **Opis osnovnog tijeka:**
  1. Sustav ispisuje attribute linka (*frequency, mode* i *baud*)
  2. U kutu postoje dvije opcije ("Edit" i "Delete")

#### UC26 - Prikaz satelita

- **Glavni sudionik:** Administrator satelita (*satellite\_admin*)
- **Cilj:** Pregled postojećeg satelita
- **Sudionici:** Baza podataka

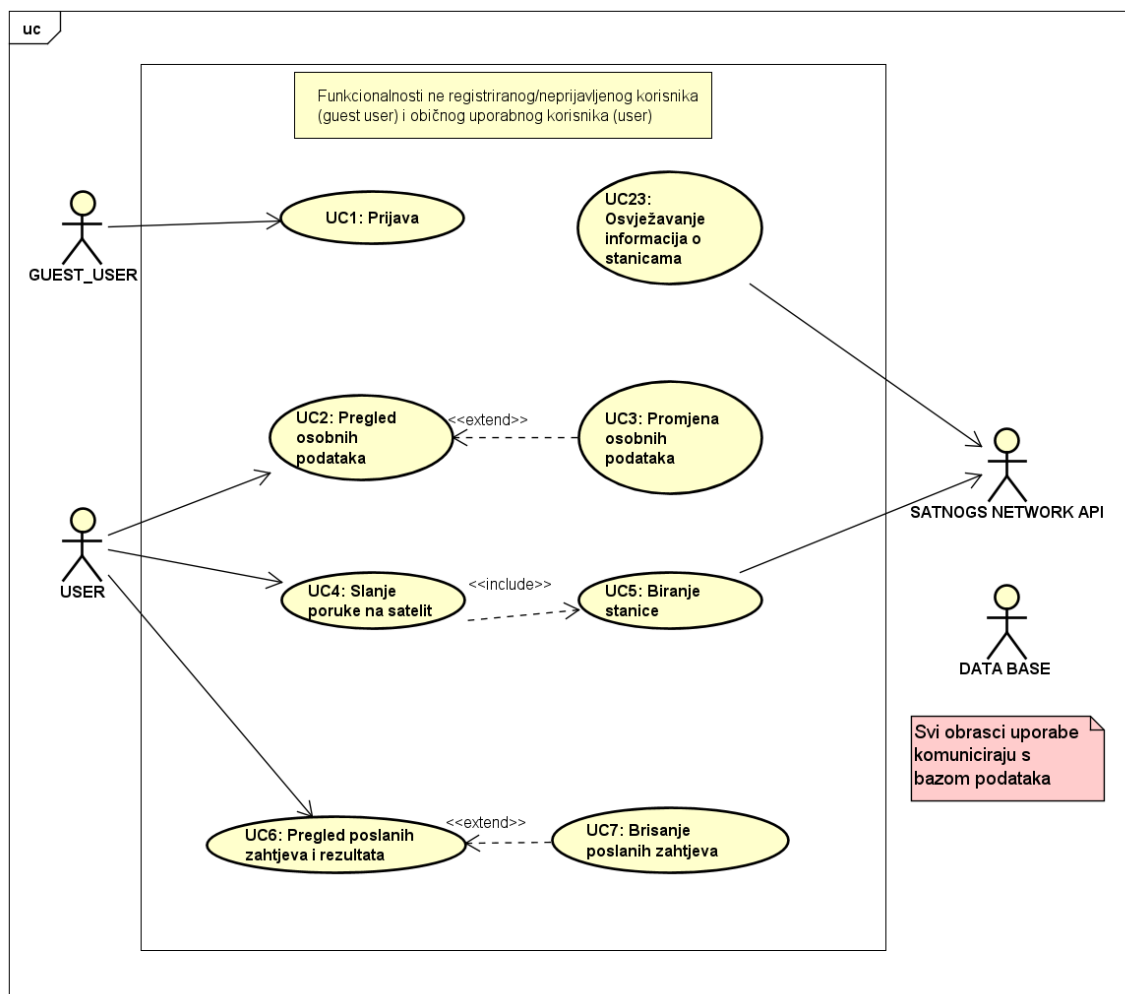
- **Preduvjet:**

1. Korisnik je prijavljen kao administrator satelita

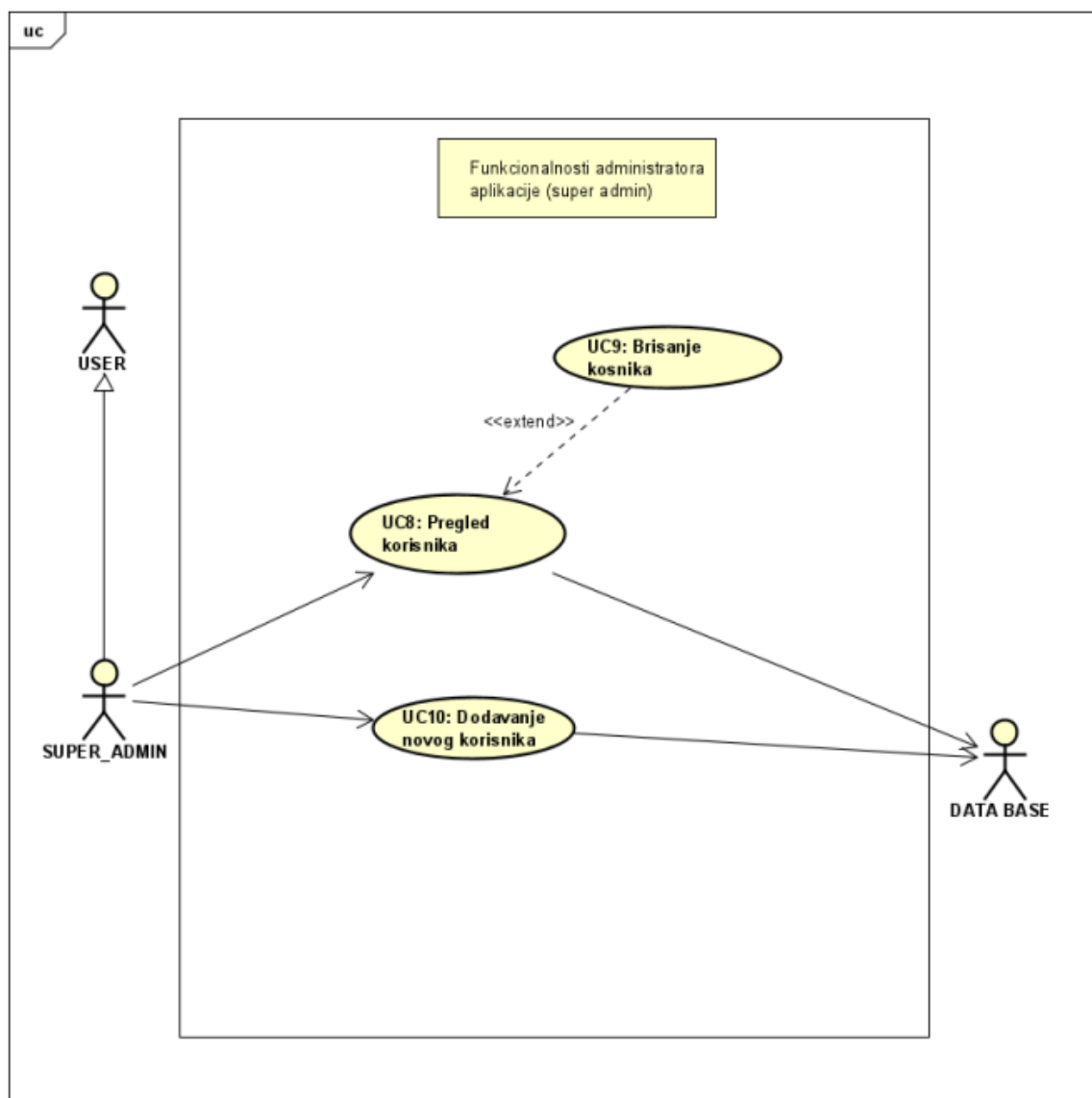
- **Opis osnovnog tijeka:**

1. Sustav ispisuje osnovne attribute satelita
2. U kutu postoje dvije opcije ("Edit" i "Delete")
3. Ispod tih informacija sustav ispisuje listu transmitera povezanih s ovim satelitom i njihove informacije
4. Iznad transmitera nalazi se opcija za dodavanje transmitera("Add transmitter")
5. Administrator ima opciju odabrati jedan transmieter i pristupiti njegovim podacima( VidiUC24)

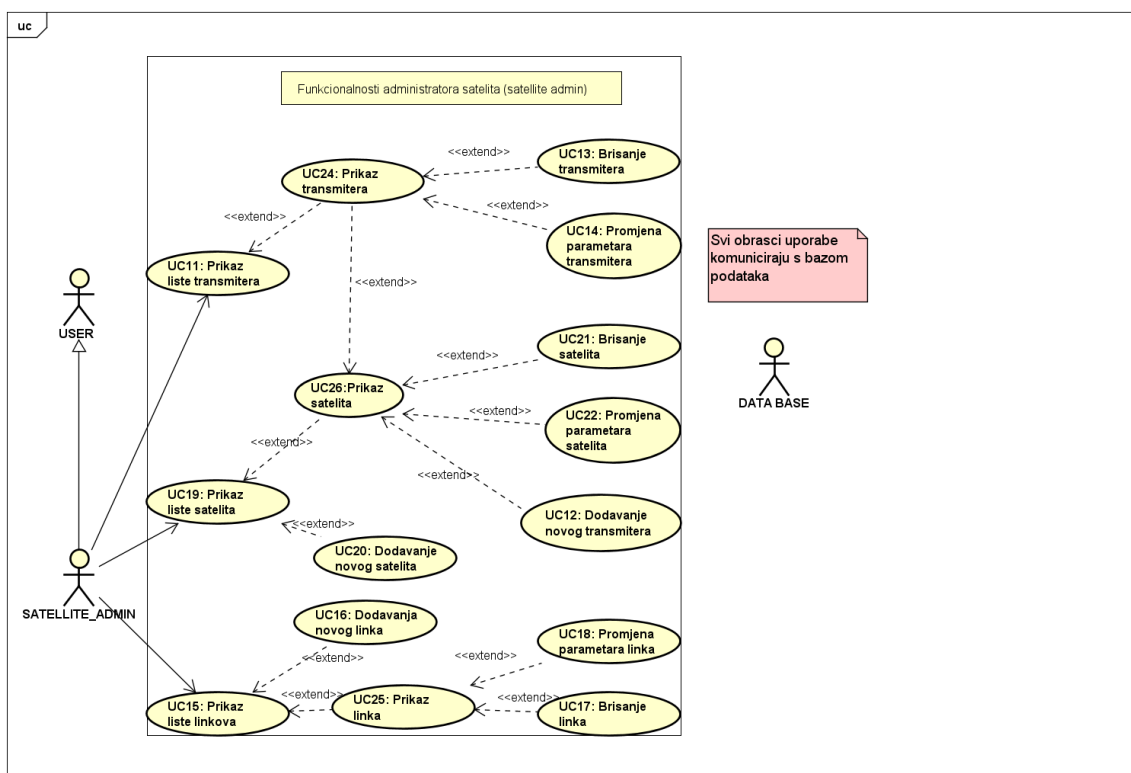
## Dijagrami obrazaca uporabe



Slika 3.1: Dijagram obrasca uporabe, funkcionalnost neprijavljenog korisnika i običnog uporabnog korisnika



Slika 3.2: Dijagram obrasca uporabe, funkcionalnost Administratora stranice

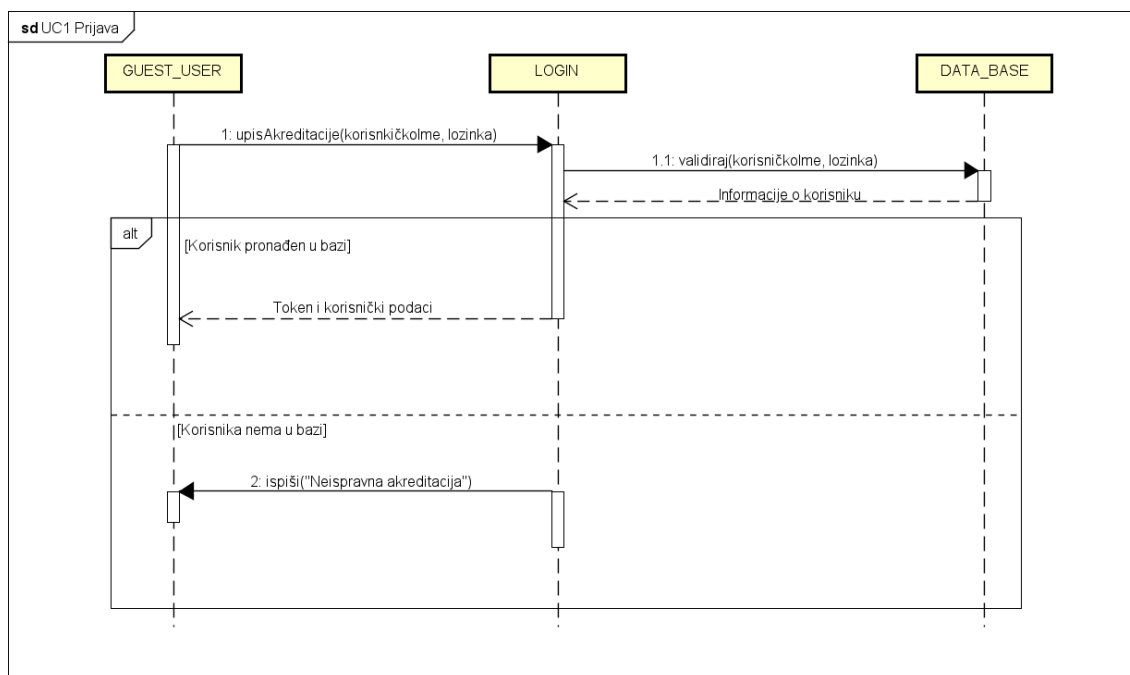


Slika 3.3: Dijagram obrasca uporabe, funkcionalnost Administratora satelita

### 3.1.2 Sekvencijski dijagrami

#### Obrazac uporabe UC1-Prijava

Neregistrirani korisnik upisuje korisničko ime i lozinku. Klikom na gumb "Log in" ti podaci se šalju na validaciju. Ako su uneseni podaci pronađeni u bazi, aplikacija generira token sesije. U suprotnom, šalje se poruka o neispravnoj akreditaciji.

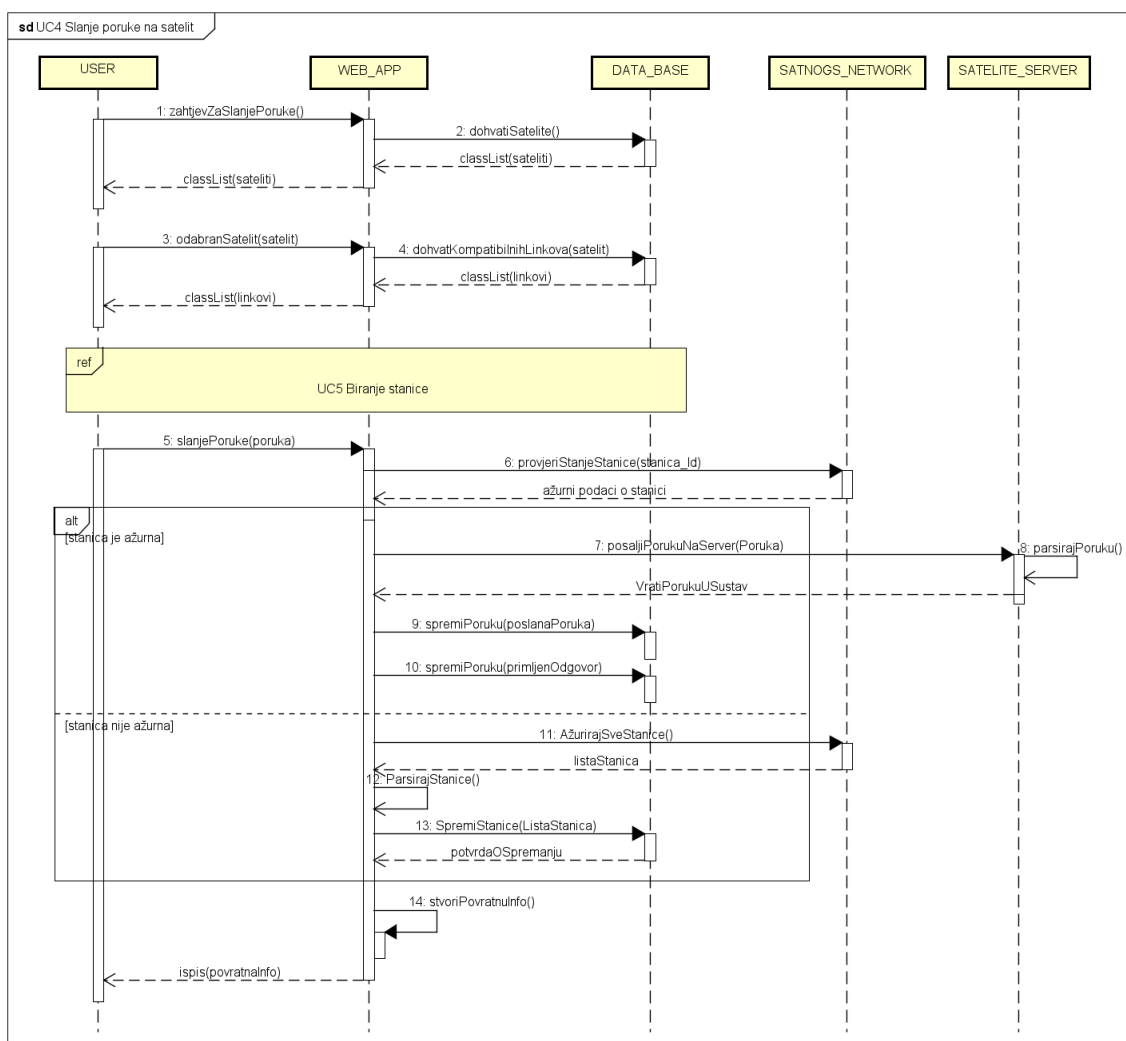


Slika 3.4: Sekvencijski dijagram prijave korisnika



### Obrazac uporabe UC4-Slanje poruke na satelit

Korisnik šalje zahtjev za slanje poruke na satelit. Sustav na taj zahtjev iz baze dohvaća podatke o satelitima te ih šalje natrag korisniku. Korisnik potom bira satelit na koji želi poslati poruku. Aplikacija na taj zahtjev dohvaća listu kompatibilnih linkova za odabrani satelit i šalje ih korisniku. Nakon toga, korisnik odabere jedan link, a zatim se ostvaruje odabir stanice (Vidi UC5). Na posljetku, korisnik unosi tekst poruke i pritiskom na gumb "Send" pokreće simulaciju komunikacije sa satelitom. Aplikacija simulira slanje i odgovor koje sprema u bazu, a nakon toga šalje odgovor korisniku.

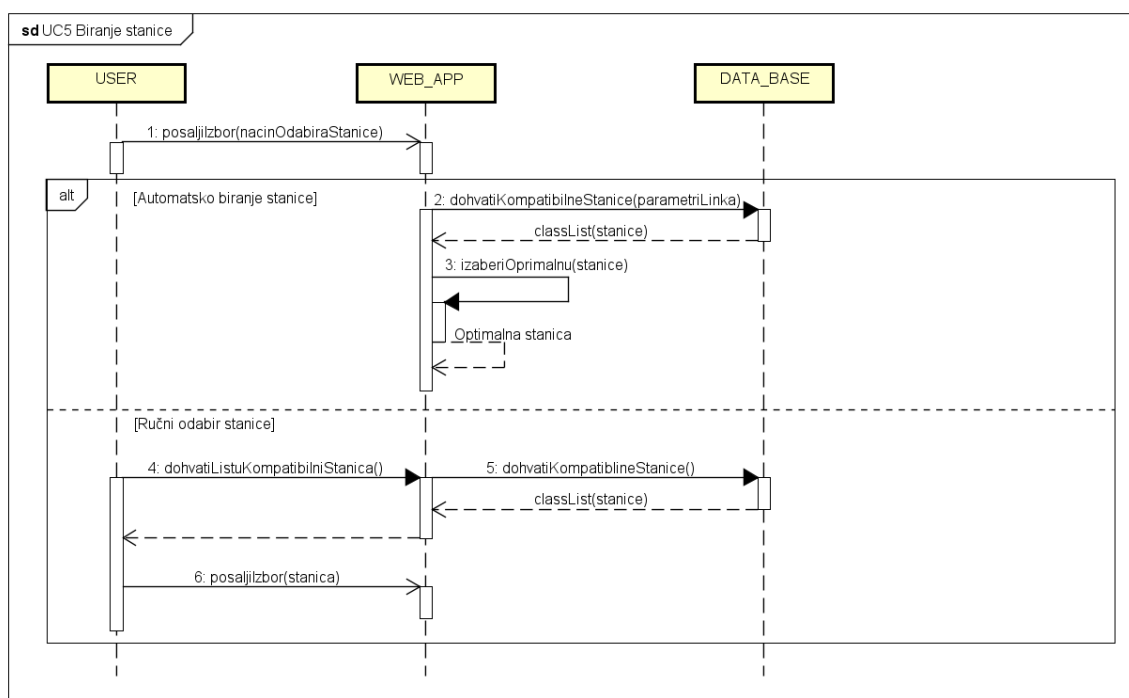


Slika 3.5: Sekvencijski dijagram slanja poruke

### Obrazac uporabe UC5-Biranje stanice

Korisnik odlučuje hoće li odabir stanice prepustiti aplikaciji (automatski) ili samostalno izabrati preko koje stanice će se komunikacija odvijati. Pri odabiru automatske opcije, aplikacija dohvaća iz baze stanice koje su kompatibilne s prethodno odabranim linkom. Odabir optimalne stanice temelji se na atributu broja obzervacija. Iz SatNOGS mreže dohvaćaju se ažurni podaci o odabranoj optimalnoj stanici. Aplikacija uspoređuje podatke o stanici iz baze s dobivenim iz SatNOGS mreže. Ako nije došlo do promjena, korisniku se šalje poruka o pronalasku stanice i on može nastaviti sa slanjem poruke. Ukoliko je došlo do promjena podataka koji utječu na komunikaciju, podaci se u bazi ažuriraju i nastavlja se s traženjem optimalne stanice. U slučaju da stanica više ne postoji u mreži, njen zapis se briše iz baze i ponovno se nastavlja traženje optimalne stanice.

U drugom slučaju, aplikacija iz baze dohvaća kompatibilne stanice pa šalje upit na SatNOGS mrežu o tim stanicama i po primitku odgovora uspoređuje dobivene podatke s pročitanim podacima iz baze. Ukoliko je došlo do promjena u podacima, ažurira ih u bazi. Zatim, šalje korisniku listu stanice iz koje on treba odabrati željenu stanicu za komunikaciju.



Slika 3.6: Sekvencijski dijagram biranja stanice

## 3.2 Ostali zahtjevi

- Sustav treba biti jednostavan/intuitivan za korištenje, korisnici se moraju znati koristiti sučeljem bez opširnih uputa.
- Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava.
- Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava.
- Lozinke moraju biti kriptirane pri pohrani u bazu.
- Sustav treba biti implementiran kao web aplikacija koristeći objektno-orijentirane jezike.
- Korisničko sučelje i sustav treba podržavati hrvatske diakritičke znakove za unos i prikaz tekstualnog sadržaja
- Sustavu se treba moći pristupati iz javne mreže pomoću HTTPS
- Sustav pri logiranju korisniku dodjeljuje JWT Token kako bi se garantirao siguran prijenos informacija putem web-a
- Sustav omogućava istovremeni rad više korisnika

## 4. Arhitektura i dizajn sustava

Arhitektura sustava može se podijeliti na tri glavna podsustava, a to su: web preglednik, web poslužitelj i baza podataka.

- **Web preglednik** je program koji služi za pristup web stranicama. Putem web preglednika, korisnik šalje zahtjeve za resursima (npr. HTML kod web stranice) ili šalje podatke (npr. putem neke forme), web preglednik dohvaća te datoteke s web poslužitelja, a potom ih interpretira i prikazuje na ekranu korisnika ili ih pohranjuje na poslužitelju.
- **Web poslužitelj** glavni je dio web aplikacije. To je namjensko računalo ili softver koji šalje i prima podatke od mnogostrukih klijenata. Komunikacija s klijentima (korisnicima i bazom podataka) odvija se preko HTTP protokola. Na korisnikov zahtjev, web preglednik dohvaća resurse i vraća u obliku HTML dokumenta ili obrađuje podatke predane u formi te ih sprema u bazu podataka.
- **Baza podataka** koristi se za pohranjivanje podataka sustava. Web aplikacija u svom radu vrlo često komunicira s bazom te iz nje dohvaća podatke ili ih u nju sprema.

Pri oblikovanju aplikacije koristili smo MVC (Model-View-Controller) obrazac softverske arhitekture. Po principu MVC-a, aplikaciju dijelimo na tri komponente:

- **Model** je glavna komponenta sustava. Predstavlja strukturu podataka (Java objekti) i njihovu funkcionalnost.
- **View** odlučuje kako će se dohvaćeni podaci reprezentirati.
- **Controller** zaprima zahtjeve za resursima (HTTP zahtjevi) od klijenta koje prilagođava i prosljeđuje Modelu ili Viewu.

Programski jezik kojeg smo odabrali za izradu backenda naše aplikacije je Java zajedno sa Spring Boot radnim okvirom u razvojnom okruženju IntelliJ, a u izradi frontenda koristili smo jezike JavaScript i TypeScript te biblioteke React, Redux i Axios uz razvojno okruženje Visual Studio Code.

## 4.1 Baza podataka

U ovom projektu koristit ćemo relacijsku bazu u kojoj će gradivne jedinice biti entiteti, definirani imenom i skupom atributa. Zadaća baze je pohrana podataka o satelitima, linkovima, baznim stanicama i korisnicima. Sukladno time entiteti koje ćemo kreirati su:

- User
- Satellite
- Transmitter
- Link
- Station
- Antenna
- Message

### 4.1.1 Opis tablica

**User** Ovaj entitet sadrži podatke o korisniku aplikacije. Njegovi atributi su: `userId` (PRIMARY KEY), `username`, `email`, `password` i `roleName`. Ovaj entitet u vezi je *One-to-Many* s entitetom `Satellite` preko atributa `userId`, u vezi *One-to-Many* s entitetom `Link` preko atributa `userId` i u vezi *One-to-Many* s entitetom `Message` preko atributa `userId`.

User		
<code>userId</code>	INT	Jedinstveni brojevi identifikator, autogeneriran od strane baze
<code>username</code>	VARCHAR	Jedinstveno korisničko ime
<code>email</code>	VARCHAR	Jedinstvena email adresa
<code>password</code>	VARCHAR	Šifrirana lozinka

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

User		
roleName	VARCHAR	SUPER_ADMIN (admin aplikacije), SATELLITE_ADMIN (admin satelita) ili USER (običan korisnik)

**Satellite** Ovaj entitet sadrži podatke o satelitima u Zamljinoj orbiti. Njegovi atributi su: satelliteId (PRIMARY KEY), satName i satelliteStatus. Ovaj entitet u vezi je *Many-to-One* s entitetom User preko atributa userId, u vezi *Many-to-Many* s entitetom Transmitter preko atributa satelliteId i transId.

Satellite		
satelliteId	INT	Jedinstveni brožani identifikator, autogeneriran od strane baze
satName	VARCHAR	Naziv satelita
satelliteStatus	VARCHAR	Active (dostupan/aktivan) ili inactive (nedostupan/neaktivan)

**Transmitter** Ovaj entitet sadrži podatke o transponderima koji se nalaze na satelitu. Njegovi atributi su: transId (PRIMARY KEY), transFreq, transMode, transBaud i transName. Ovaj entitet u vezi je *Many-to-Many* s entitetom Satellite preko atributa satelliteId i transId i u vezi *Many-to-One* s entitetom Link preko atributa linkId.

Transmitter		
transId	INT	Jedinstveni brožani identifikator, autogeneriran od strane baze
transFreq	VARCHAR	Frekvencija na kojoj transponder može komunicirati
transMode	VARCHAR	Radio-frekvencijska modulacija transpondera (oznaka za način kodiranja informacije u u radio-signalu)

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Transmitter		
transmBaud	VARCHAR	Broj kodiranih informacija (definiranih modeom transmitera) koje se mogu promijeniti u jedinici vremena
transmName	VARCHAR	naziv transmitera

**Link** Ovaj entitet sadrži podatke o linkovima koji predstavljaju vezu između entiteta Satellite i entiteta Station. Njegovi atributi su: linkId (PRIMARY KEY), linkMode, linkBaud i linkFreq. Ovaj entitet u vezi je *Many-to-Many* s entitetom Antenna preko atributa linkId i antennaId, u vezi *Many-to-One* s entitetom User preko atributa userId i u vezi *One-to-Many* s entitetom Transmitter preko atributa linkId.

Link		
linkId	INT	Jedinstveni brojčani identifikator, autogeneriran od strane baze
linkMode	VARCHAR	Radio-frekvencijska modulacija linka (oznaka za način kodiranja informacije u u radio-signalu)
linkBaud	VARCHAR	Broj kodiranih informacija (definiranih modeom linka) koje se mogu promijeniti u jedinici vremena
linkFreq	INT	Frekvencija na kojoj se uspostavlja komunikacija između satelita i bazne stanice

**Station** Ovaj entitet sadrži podatke o zemaljskim stanicama (dobivene iz sat-NOGS Network API-ja) koje komuniciraju sa Zemaljskim satelitima preko linkova. Njegovi atributi su: stationId (PRIMARY KEY), numOfObservations, statName, altitude, longitude, latitude i stationStatus. Ovaj entitet je u vezi *Many-to-Many* s entitetom Antenna preko atributa antennaId i stationId.

Station		
stationId	INT	jedinstveni brožčani identifikator
numOf- Observations	INT	broj uspješno obavljenih slanja i dohvaćanja podataka sa satelita
statName	VARCHAR	Naziv Zemaljste stanice
altitude	INT	Nadmorska visina lokacije na kojoj se nalazi stanica
longitude	DECIMAL(3,2)	Geografska dužina lokacije na kojoj se nalazi stanica
latitude	DECIMAL(3,2)	Geografska širina lokacije na kojoj se nalazi stanica
stationStatus	VARCHAR	Offline (nije u mreži), testing (u fazi testiranja) ili online (u mreži je)

**Antenna** Ovaj entitet sadrži podatke o anteni koja se nalazi na stanici (dobivene iz satNOGS Network API-ja). Njegovi atributi su: antennaId (PRIMARY KEY), antennaType, antennaFreqHigh i antennaFreqLow. Ovaj entitet u vezi je *Many-to-Many* s entitetom Station preko atributa antennaId i stationId, u vezi *Many-to-Many* je s entitetom Link preko atributa linkId i antennaId.

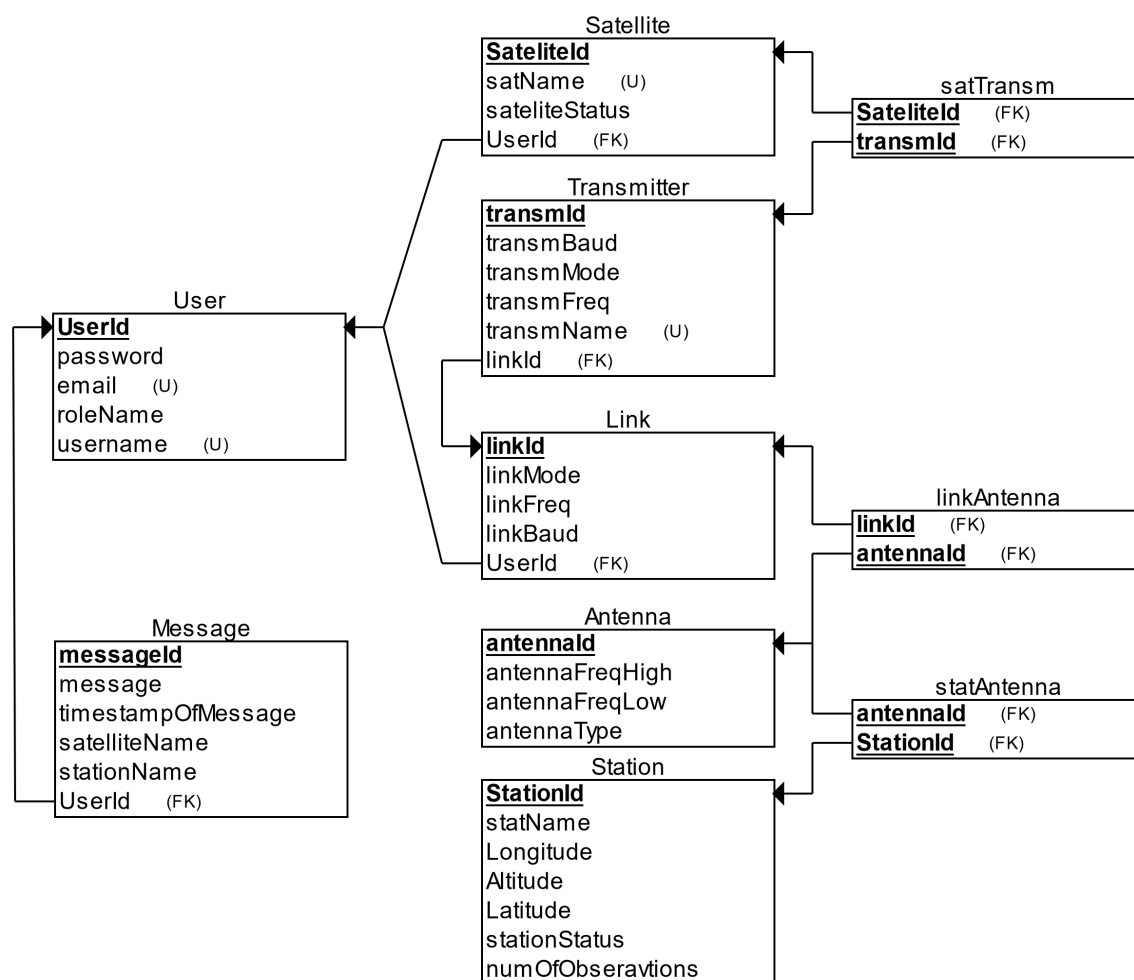
Antenna		
antennaId	INT	Jedinstveni brožčani identifikator, autogeneriran od strane baze
antennaType	VARCHAR	Tip antene
antennaFreq- High	INT	Gornja granica frekvencije na kojoj antena može komunicirati
antennaFreq- Low	INT	Donja granica frekvencije na kojoj antena može komunicirati

**Message** Ovaj entitet sadrži podatke o porukama koje je korisnik slao na neki od satelita i rezultatima koje je od njega dobio. Njegovi atributi su: messageId (PRIMARY KEY), satelliteId, message i timestampOfMessage. Ovaj entitet u vezi je *Many-to-One* s entitetom User preko atributa userId.



Message		
messageId	INT	Jedinstveni brojčani identifikator, autogeneriran od strane baze
satelliteName	VARCHAR	Ime satelita koji se koristio u komunikaciji
stationName	VARCHAR	Ime stanice kroz koju teče komunikacija
message	VARCHAR	tekst poruke koja je poslana/primljena
timestampOf-Message	TIMESTAMP	Vrijeme i datum slanja/primanja poruke

#### 4.1.2 Dijagram baze podataka

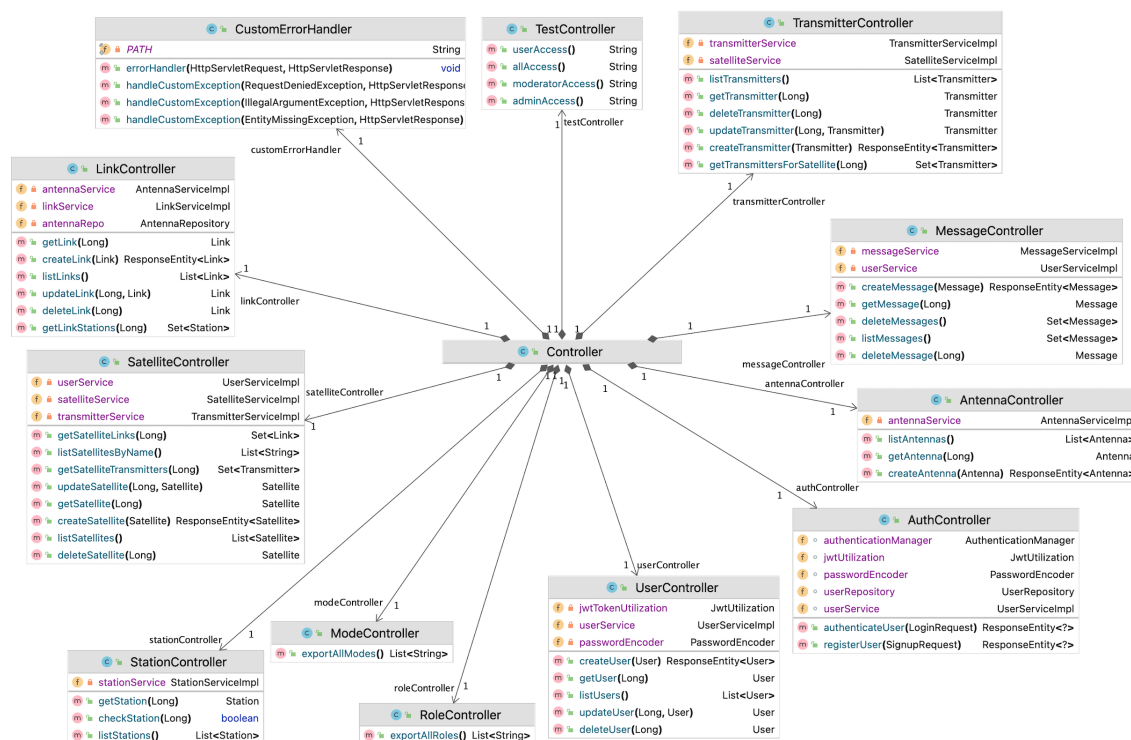


Slika 4.1: Dijagram baze podataka

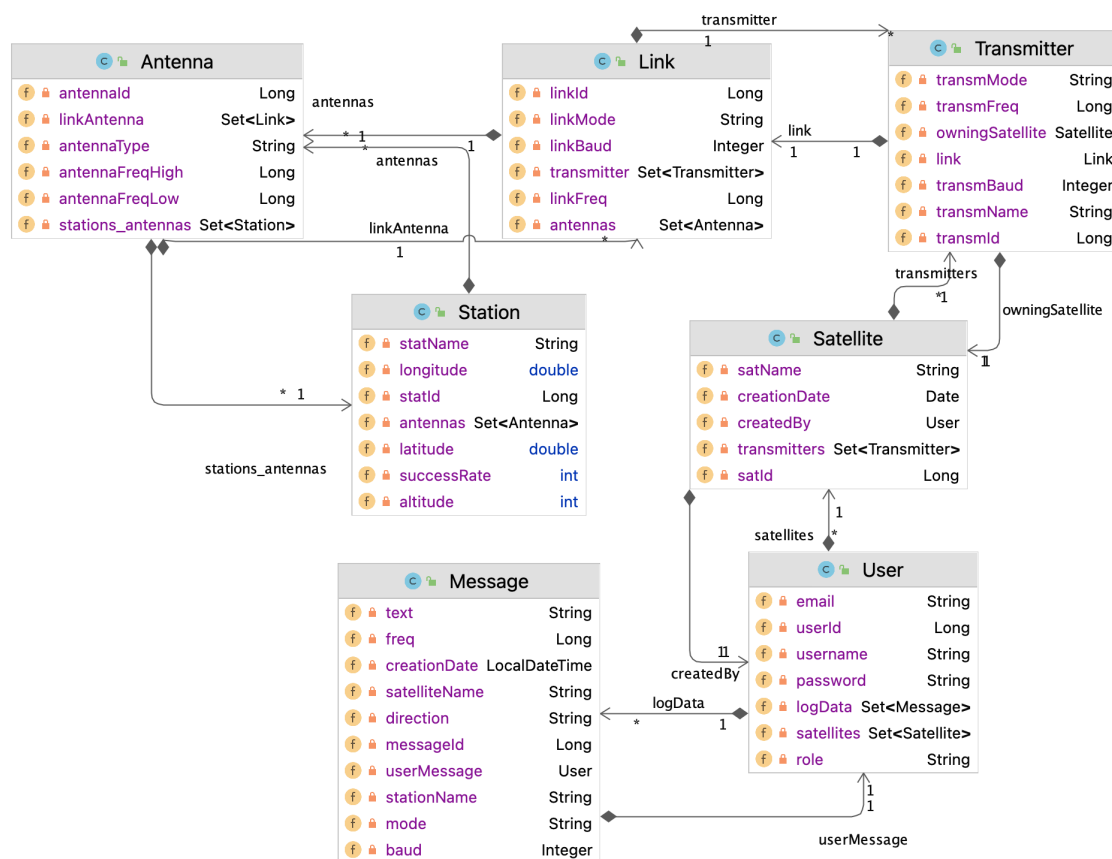
## 4.2 Dijagram razreda

Na slikama 4.2 i 4.3 prikazani su razredi koji odgovaraju *backend* dijelu MVC arhitekture. Razredi prikazani na slici 4.2 nasljeđuju Controller razred. Metode implementirane u tim razredima manipuliraju modelima i vraćaju zatražene podatke koji su reprezentirani modelima u listama.

Model razredi preslikavaju strukturu baze podataka u aplikaciji. Razredi prikazani na 4.3 su Java klase koji predstavljaju entitete iz baze podataka. Članske varijable svake klase su atributi odgovarajućeg entiteta iz baze.



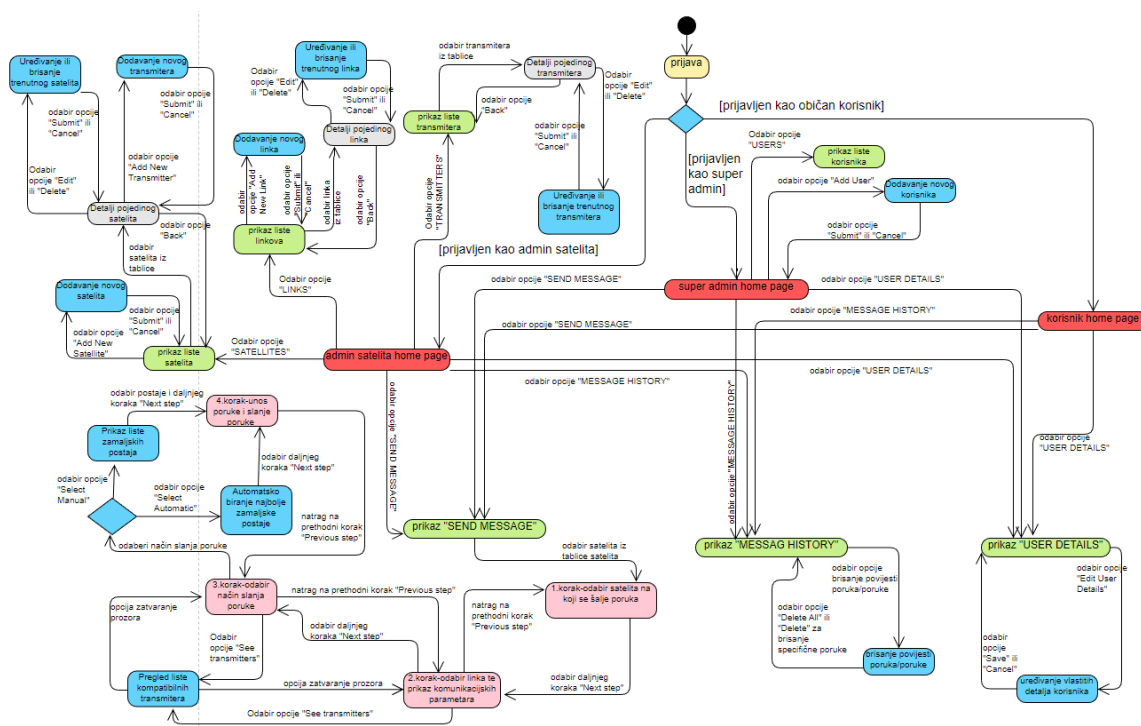
Slika 4.2: Dijagram razreda Controllers



Slika 4.3: Dijagram razreda Models

## 4.3 Dijagram stanja

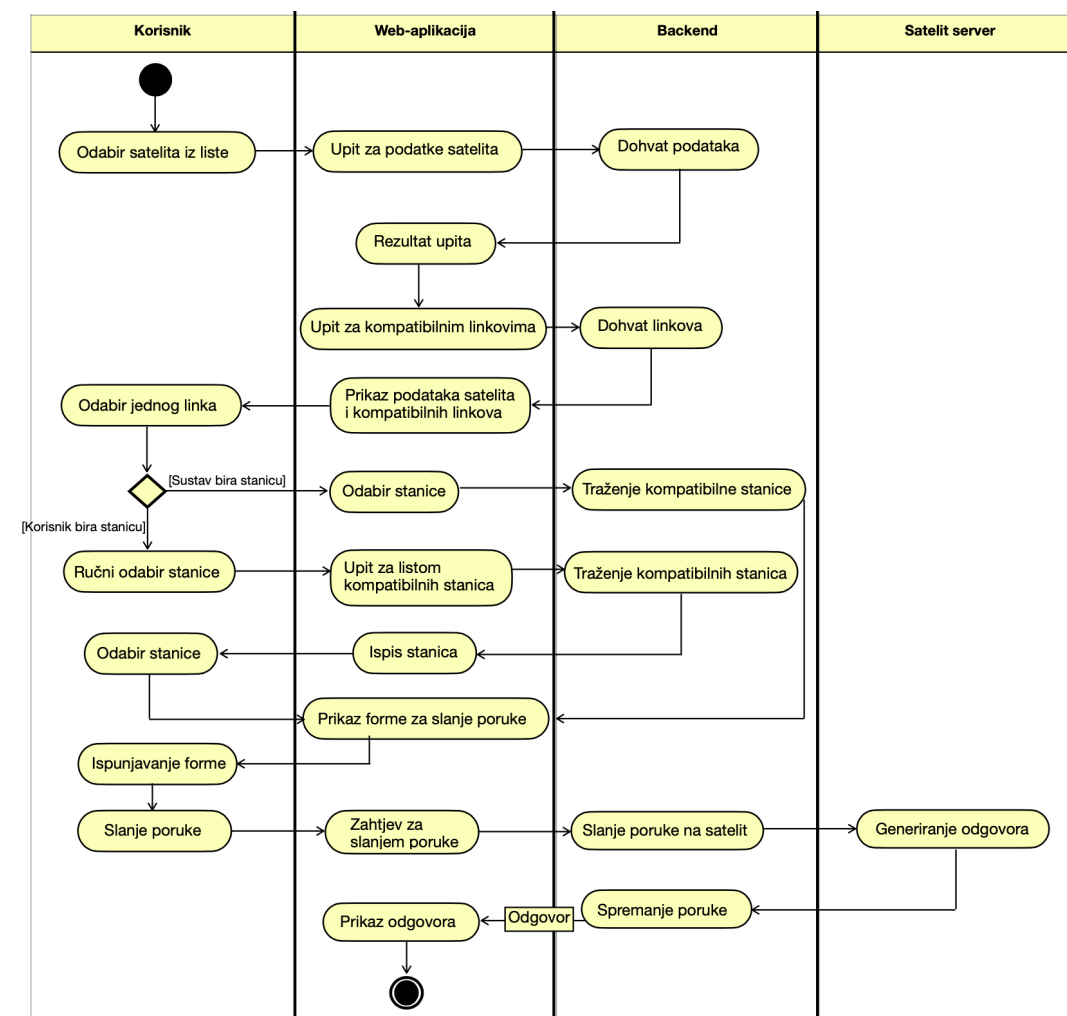
Na slici 4.4 je prikaz dijagrama stanja koji predstavlja stanja kroz koje korisnik (superadmin, admin satelita ili običan korisnik) prolaze od trenutka prijave, pa sve do trenutka odjave iz web aplikacije. Početno stanje za sve korisnike je prijava. Nakon prijave svim korisnicima je prikazana početna stranica "Home page" gdje se nalaze opcije za daljnje korištenje ovisno o ulozi. Svim korisnicima je zajedničko slanje poruka na prikazu "SEND MESSAGE" gdje se kroz 4 koraka odvija odabir satelita, odabir odgovarajućeg linka, zatim odabira načina slanja poruke te u konačnici unos poruke koja će se poslati i slanje poruke. Osim slanja poruke svima je omogućen pregled povijesti poslanih poruka na prikazu "MESSAGE HISTORY" gdje svaki korisnik ima mogućnost brisanja poruka iz tablice poruka. Također, svima je omogućen prikaz vlastitih korisničkih podataka i njihovo uređivanje. Superadminu je dodatno omogućena opcija dodavanje novih korisnika opcijom "Add User" te prikaz liste trenutno postojećih korisnika u sustavu. Korisnik koji ima ulogu admin satelita je u mogućnosti obavljati CRUD akcija kreiranja (eng. create), pregled postojećih podataka (eng. read), uređivanje (eng. update) i brisanje podataka (eng. delete) nad satelitima, linkovima i transmitterima. Zbog preglednosti na dijagramu nije dodana opcija odjava jer iz bilo kojeg prikaza se korisnik može u gornjem desnom kutu odjaviti sa stranice. Također, korisnik se u bilo kojem trenutku može vratiti na početnu stranicu "Home page"



Slika 4.4: Dijagram stanja

## 4.4 Dijagram aktivnosti

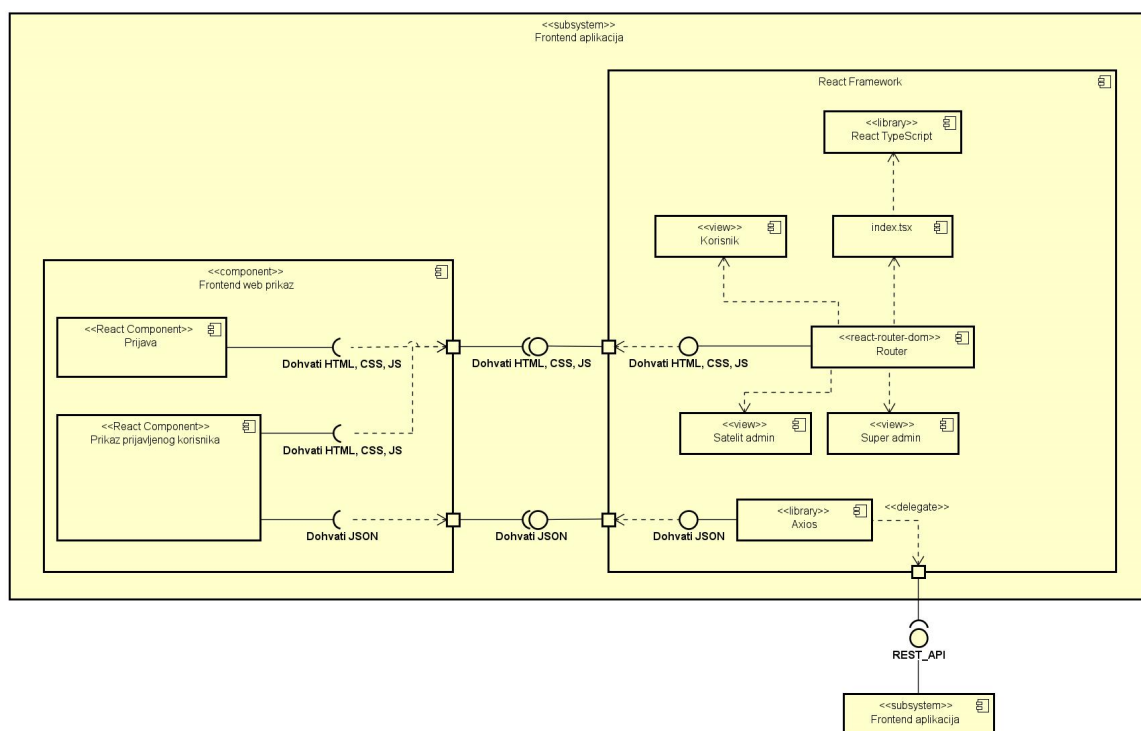
Dijagram aktivnosti primjenjujemo za opise modela toka upravljanja ili toka podataka, ali ne i za modeliranje događajima poticanog ponašanja. Pri modeliranju toka upravljanja svaki korak prethodi onom sljedećem u lancu i tim se redom izvršavaju uz naglašenu jednostavnost. Na dijagramu aktivnosti prikazan na slici 4.5 prikazan je proces slanja poruke na satelit. Prijavljeni korisnik odabire jedan od ponuđenih satelita. Sustav vraća listu kompatibilnih linkova za odabrani satelit. Korisnik zatim odabire link pomoću kojeg želi poslati poruku. Nakon što se odabere link, odabire se i način slanja poruke. Za ručni odabir, sustav vraća stanice koje mogu provesti slanje poruke na satelit uz odabrani link, korisnik zatim odabire željenu stanicu i nastavlja na sljedeći korak. Kod automatskog slanja poruke, stanicu odabire sustav prema najvećem broju obzervacija i korisnik nastavlja na sljedeći korak. Sljedeći korak je ispunjavanje forme za slanje poruke gdje korisnik upisuje željenu poruku i šalje zahtjev za slanje poruke. Kada sustav zaprimi slanje poruke, poruka se prosljeđuje na satelit server gdje se simulira odgovor satelita. Odgovor satelita i poslana poruka se spremaju u bazu podataka te se korisniku prikaže poslana poruka i primljeni odgovor.



Slika 4.5: Dijagram aktivnosti

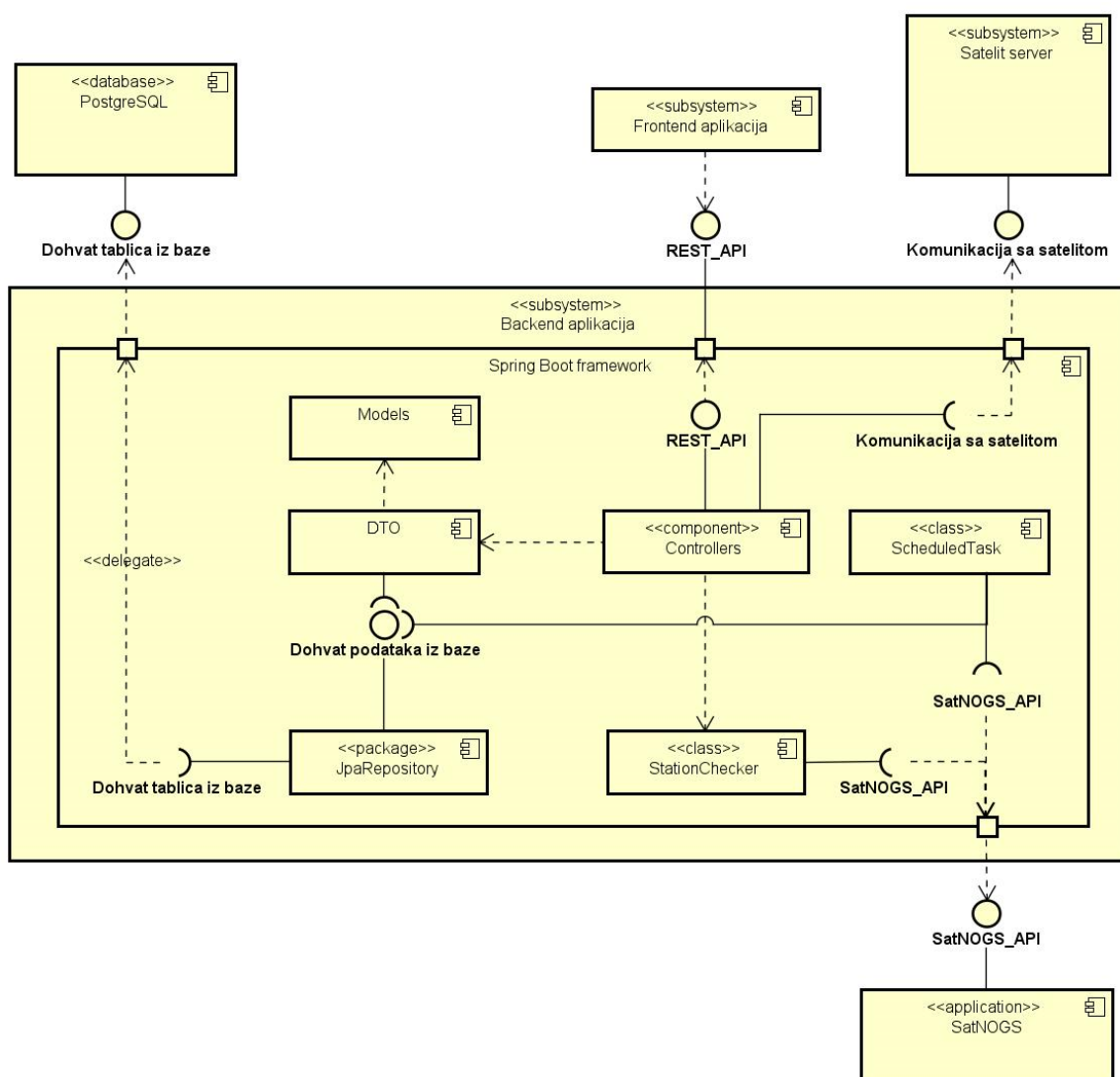
## 4.5 Dijagram komponenti

Dijagram komponenti prikazani na slikama 4.6 i 4.7 opisuju međuovisnost i organizaciju i odnose u internoj strukturi. Sustavu se pristupa preko internet preglednika preko sučelja za dohvat HTML, CSS i JS datoteka koje služe za prikaz i funkcionalnost grafičkog sučelja. Router je komponenta kojom se upravlja prikaz internet stranice prema ulozi korisnika. Sučelju za primanje JSON podataka pristupa se preko REST API komponenti. Pomoću Axios biblioteke dobivene podatke s *backenda* prenosimo na korisnikov preglednik. Na *backendu* nam je potrebna komponenta Controllers koja služi da modele koje pretvaramo u DTO (*Data transfer object*) pošaljemo na *frontend* i da prenesemo odgovore sa satelita. Paket JpaRepository koristimo za pisanje i čitanje podataka iz vanjske baze podataka. Klase StationChecker i ScheduledTask koriste vansjku aplikaciju SatNOGS za najsvježije podatke potrebne za rad naše aplikacije.



Slika 4.6: Dijagram komponenti frontend dijela aplikacije





Slika 4.7: Dijagram komponenti backend dijela aplikacije

## 5. Implementacija i korisničko sučelje

### 5.1 Korištene tehnologije i alati

Komunikacija u timu je ostvarena korištenjem aplikacija Discord<sup>1</sup> i GoogleMeet<sup>2</sup>. Komunikacija s asistentom grupe je ostvarena korištenjem aplikacije MicrosoftOutlook<sup>3</sup>. Za izradu sekvencijskih UML dijagrama i UML dijagrama obrazaca upotrebe korišten je alat Astah UML<sup>4</sup>. Za izradu UML dijagrama razreda korišten je alat IntelliJ<sup>5</sup>. Za izradu dijagrama razmještaja korišten je alat VisualParadigmOnline<sup>6</sup>, a za izradu dijagrama aktivnosti korišten je program Pages razvijen od strane Apple Inc.<sup>7</sup>. Za izradu modela baze podataka korišten je alat ERDPlus<sup>8</sup>. Za upravljanje izvornim kodom korišten je alat Git<sup>9</sup>. Udaljeni repozitorij projekta nalazi se na platformi Gitlab<sup>10</sup> koja je korištena i za upravljanje zadacima na projektu. Kao razvojno okruženje na backendu je korišten IntelliJ, a na frontendu je korišten Visual Studio Code<sup>11</sup>. Tehnologije korištene za razvoj backenda su radni okvir Spring Boot<sup>12</sup> i programski jezik Java<sup>13</sup>. Tehnologije korištene za razvoj frontenda su biblioteka React<sup>14</sup> i programski jezik JavaScript<sup>15</sup>.

---

<sup>1</sup><https://discord.com/>

<sup>2</sup><https://meet.google.com/>

<sup>3</sup><https://outlook.live.com/owa/>

<sup>4</sup><https://astah.net/products/astah-uml/>

<sup>5</sup><https://www.jetbrains.com/idea/>

<sup>6</sup><https://online.visual-paradigm.com/>

<sup>7</sup><https://www.apple.com/pages/>

<sup>8</sup><https://erdplus.com/>

<sup>9</sup><https://git-scm.com/>

<sup>10</sup><https://gitlab.com/>

<sup>11</sup><https://code.visualstudio.com/>

<sup>12</sup><https://spring.io/projects/spring-boot>

<sup>13</sup><https://www.java.com/en/>

<sup>14</sup><https://reactjs.org/>

<sup>15</sup><https://www.javascript.com/>

Za automatizirano testiranje cjelokupne aplikacije korišten je alat Selenium IDE<sup>16</sup> te programski jezik Java. Za automatizirano testiranje komponenti korišten je alat JUnit<sup>17</sup> te programski jezik Java. Za ručno testiranje korišten je alat ReqBin<sup>18</sup> te programski jezik Java. Za puštanje aplikacije u pogon korištena je platforma Microsoft Azure<sup>19</sup>.

---

<sup>16</sup><https://www.selenium.dev/selenium-ide/>

<sup>17</sup><https://junit.org/junit5/>

<sup>18</sup><https://reqbin.com/>

<sup>19</sup><https://azure.microsoft.com/en-us>

## 5.2 Ispitivanje programskog rješenja

### 5.2.1 Ispitivanje komponenti

Ispitivanje komponenti ostvareno je pomoću Spring Boot i JUnit 5 alata za ispitivanje.

Na slikama 5.1 i 5.2 prikazani su isječki koda iz testa koji provjerava ispravnost spajanja i dohvaćanja kompatibilne trojke (satelit, link i stanica). Slika 5.1 prikazuje dva slučaja: 1. kada satelit, onosno njegovi transmiteri, nisu kompatibilni s linkovima koji se nalaze u bazi. Očekivani rezultat ovog slučaja je prazan set kompatibilnih likova. Te 2. kada postoji link koji je kompatibilan s transponderom uz očekivani rezultat da set kompatibilnih likova sadrži točno jedan link koji je jednak linku kojeg stvaramo u ovom testu. Slika 5.2 također prikazuje dva slučaja: 1. kada za određeni link ne postoje kompatibilne stanice, odnosno niti jedna antena neke stanice nije kompatibilna sa zadanim linkom. Očekivani rezultat u ovom slučaju je prazan set kompatibilnih stanica. Te 2. kada postoji stanica s kojom je zadani link kompatibilan. Očekivani rezultat ovog testa je da će set kompatibilnih stanica za zadani link sadržavati jednu stanicu koja je jednaka stanici koju na početku testa stvaramo.

```
@Test
@DisplayName("Testing when satellite has no compatible links.")
public void testSatelliteLinkNotComp() {
    Pair ids = createSatelliteAndTransmitter(10L);
    assertTrue(transmitterService.findAllCompatibleLinks(satelliteService.fetch(ids.first).getTransmittersObj()).isEmpty());
    deleteSatelliteAndTransmitter(ids.first, ids.second);
}

@Test
@DisplayName("Testing when satellite has compatible links.")
public void testSatelliteLinkComp() {
    Pair ids = createSatelliteAndTransmitter(103L);
    Link l = linkService.createNewLink(new Link(linkMode: "PSK", linkFreq: 103L, linkBaud: 100));
    Set<Link> compLinks = transmitterService.findAllCompatibleLinks(satelliteService.fetch(ids.first).getTransmittersObj());
    assertTrue(condition: compLinks.contains(l) && compLinks.size()==1);
    deleteSatelliteAndTransmitter(ids.first, ids.second);
    linkService.deleteLink(l.getLinkId());
}
```

Slika 5.1: Testiranje spajanja i dohvaćanja kompatibilnih satelita i linkova

```
@Test
@DisplayName("Testing when link has no compatible stations.")
public void testLinkStationNotComp() {
    Link l = linkService.createNewLink(new Link( linkMode: "PSK", linkFreq: -100L, linkBaud: 100));
    assertTrue(antennaService.findAllCompatibleStations(l.getAntennasObj()).isEmpty());
    linkService.deleteLink(l.getLinkId());
}

@Test
@DisplayName("Testing when link has compatible stations.")
public void testLinkStationComp() {
    Pair ids = createStationAndAntenna(1000L, 10L);
    Link l = linkService.createNewLink(new Link( linkMode: "PSK", linkFreq: 122L, linkBaud: 100));
    Set<Station> compStat = antennaService.findAllCompatibleStations(l.getAntennasObj());
    for(Station st:compStat) {
        assertTrue(st.equals2(stationService.fetch(ids.first)));
    }
    linkService.deleteLink(l.getLinkId());
    deleteStationAndAntenna(ids.first, ids.second);
}
```

Slika 5.2: Testiranje spajanja i dohvaćanja kompatibilnih stanica i linkova

Na slici 5.3 prikazano je testiranje funkcionalnosti ispitivanja jednakosti stanica koje su spremljene u bazi u odnosu na stanice koje dohvaćamo sa SatNOGS API-ja. Za prvi test pretpostavka je da stanica s id-jem 6 postoji i u bazi satcom aplikacije i u bazi SatNOGS-a pa je sukladno tome očekivani rezultat istina, tj. stanice su jednake. Pretpostavka za drugi test je da smo u satnogs bazi promijenili barem jedan podatak stanice s id-jem 9 pa je zbog toga očekivai rezultat laž, odnosno da stanice nisu jednake. Pretpostavka za treći test je da smo u satnogs bazi promijenili barem jedan podatak neke antene koja pripada stanici s id-jem 12, tako da možemo očekivati da će rezultat ponovno biti laž, tj. stanice nisu jednake.

```
@Test
@DisplayName("Test checker for identical stations")
void testStationCheckerTrue(){
    boolean result=StationChecker.checkStations((long) 6);
    assertTrue(result);
}

@Test
@DisplayName("Test checker for stations with difference in station data")
void testStationCheckerFalseStation(){
    //pobrinuti se da prije testiranja barem jedan podatak o ovoj stanici je krivi u bazi
    boolean result=StationChecker.checkStations((long) 9);
    assertFalse(result);
}

@Test
@DisplayName("Test checker for stations with difference in antenna data")
void testStationCheckerFalseAntenna(){
    //pobrinuti se da prije testiranja barem jedan podatak o antenama ove stanice krivi
    boolean result=StationChecker.checkStations((long) 12);
    assertFalse(result);
}
```

Slika 5.3: Testiranje ispitivanja jednakosti stanica u bazi i na SatNOGS API-ju

Na slici 5.4 prikazano je testiranje spremanja poruke u bazu. Prvi test pokriva slučaj spremanja kada su svi atributi modela Message ispravno postavljeni i očekivani rezultat nakon spremanja je da se u bazi nalazi poruka koja je jednaka onoj koju smo željeli spremiti. Drugi test prikazuje da se u slučaju pokušaja spremanja poruke koja nema sve atribute zadane, konkretno u ovom testnom primjeru atribut `satelliteName` je null, baca iznimka `IllegalArgumentException`.

```
@Test
@DisplayName("Testing creation of message")
public void TestMessageSaveToDBOK()
{
    Message m = new Message( stationName: "TestStatName", satelliteName: "TestSatName", LocalDateTime.now(),
        text: "Testing creaion of message", direction: "UPLOAD", freq: 3000, mode: "BPSK", baud: 150000);
    m.setUserMessage(userService.fetch( userId: 4L));
    messageService.createNewMessage(m);
    Long mId = m.getMessageId();
    assertEquals(m, messageService.fetch(mId));
    messageService.deleteMessage(mId);
}

@Test
@DisplayName("Testing creation of message")
public void TestMessageSaveToDBNoSatelliteName()
{
    Message m = new Message( stationName: "TestStatName", satelliteName: null, LocalDateTime.now(),
        text: "Testing creaion of message", direction: "UPLOAD", freq: 3000, mode: "BPSK", baud: 150000);
    m.setUserMessage(userService.fetch( userId: 4L));
    assertThrows(IllegalArgumentException.class, () -> messageService.createNewMessage(m));
}
```

Slika 5.4: Testiranje spremanja poruka u bazu.

Na slikama 5.5 i 5.6 prikazano je testiranje komunikacije sa serverom kojem šaljem poruke, tj. koji "glumi" satelit. Test na slici 5.5 prikazuje da ako nedostaje neki od argumenata u poruci koju šaljem, server će vratiti null kao odgovor. Test na slici 5.6 prikazuje da ako pošaljemo ispravnu poruku, server će vratiti odgovor koji nije null.

```
@Test
@DisplayName("Test missing message or communication parameters")
void testSendMessageWithMissingParameters(){
    //For this test Server application must be running
    Message message = new Message( stationName: "TestStatName", satelliteName: "TestSatName", LocalDateTime.now(),
        text: "Testing creasion of message", direction: "UPLOAD", freq: 3000, mode: "BPSK", baud: 150000);
    message.setUserMessage(userService.fetch( userId: 4L));

    SatelliteConnector connector = new SatelliteConnector();
    message.setSatelliteName(null);
    assertNull(connector.sendMessage(message));
    message.setSatelliteName("TestStatName");
    message.setStationName(null);
    assertNull(connector.sendMessage(message));
    message.setStationName("TestSatName");
    message.setText(null);
    assertNull(connector.sendMessage(message));
    message.setText("");
    assertNull(connector.sendMessage(message));
    message.setText("Testing parameters");
    message.setDirection(null);
    assertNull(connector.sendMessage(message));
    message.setDirection("UPLOAD");
    message.setFreq(null);
    assertNull(connector.sendMessage(message));
    message.setFreq((Long) 3000);
    message.setMode(null);
    assertNull(connector.sendMessage(message));
    message.setMode("BPSK");
    message.setBaud(null);
    assertNull(connector.sendMessage(message));
    message.setBaud(150);
}
```

Slika 5.5: Testiranje neispravne komunikacije sa serverom koji "glumi" satelit.

```
@Test
@DisplayName("Test sending a valid message to erver")
void testSendMessageWithAkkParameters(){
    //For this test Server application must be running
    Message message = new Message( stationName: "TestStatName", satelliteName: "TestSatName", LocalDateTime.now(),
        text: "Testing creasion of message", direction: "UPLOAD", freq: 3000, mode: "BPSK", baud: 150000);
    message.setUserMessage(userService.fetch( userId: 4L));
    SatelliteConnector connector = new SatelliteConnector();
    Message receivedMessage=connector.sendMessage(message);
    assertNotNull(receivedMessage);
}
```

Slika 5.6: Testiranje ispravne komunikacije sa serverom koji "glumi" satelit.



## 5.2.2 Ispitivanje sustava

Ispitivanje sustava provedeno je pomoću dodatka za preglednik Selenium IDE. U nastavku su opisani provedeni ispitni slučajevi.

Na slikama 5.7 i 5.8 prikazan je prvi ispitni slučaj u kojem je ispitan pokušaj prijave registriranog korisnika čiji podaci postoje u bazi podataka. Očekivani izlaz je uspješno provedena prijava u sustav i preusmjerenje na početnu stranicu korisnika.

http://localhost:3000			
	Command	Target	Value
1	✓ open	/login	
2	✓ set window size	754x824	
3	✓ click	id=username	
4	✓ type	id=username	superAdmin1
5	✓ type	css=p-password-input	superAdmin1
6	✓ click	css=p-button-label	
7	✓ mouse over	css=p-button-label	
8	✓ mouse out	css=p-button-label	
9	✓ verify element not present	css=#login-error-message	

Slika 5.7: Parametri prvog ispitnog slučaja

### Running 'loginSuccess'

1. open on /login OK
2. setWindowSize on 754x824 OK
3. click on id=username OK
4. type on id=username with value superAdmin1 OK
5. type on css=p-password-input with value superAdmin1 OK
6. click on css=p-button-label OK
7. Trying to find css=p-button-label... OK
8. mouseOut on css=p-button-label OK
9. verifyElementNotPresent on css=#login-error-message OK

'loginSuccess' completed successfully

Slika 5.8: Rezultat prvog ispitnog slučaja

Slike 5.9 i 5.10 prikazuju drugi ispitni slučaj u kojem je ispitan pokušaj prijave korisnika koji ne postoji u bazi podataka. Očekivani izlaz je neuspješna prijava korisnika u sustav te prikaz poruke pogreške.

http://localhost:3000			
	Command	Target	Value
1	✓ open	/login	
2	✓ set window size	754x824	
3	✓ click	id=username	
4	✓ type	id=username	nepostojeciUser
5	✓ type	css=p-password-input	superAdmin1
6	✓ click	css=p-button-label	
7	✓ mouse over	css=p-button-label	
8	✓ mouse out	css=p-button-label	
9	✗ verify element not present	css=#login-error-message	

Slika 5.9: Parametri drugog ispitnog slučaja

#### Running 'loginFail'

1. open on /login OK
2. setWindowSize on 754x824 OK
3. click on id=username OK
4. type on id=username with value nepostojeciUser OK
5. type on css=p-password-input with value superAdmin1 OK
6. click on css=p-button-label OK
7. mouseOver on css=p-button-label OK
8. mouseOut on css=p-button-label OK
9. verifyElementNotPresent on css=#login-error-message Failed:  
Element with locator css=#login-error-message was found

'loginFail' ended with 1 error(s)

Slika 5.10: Rezultat drugog ispitnog slučaja

Na slikama 5.11 i 5.12 vidimo treći ispitni slučaj koji ispituje registraciju novog korisnika sa valjanim podacima u ispravnom formatu. Očekivani izlaz je uspješna registracija te preusmjeravanje na stranicu Users na kojoj se nalazi lista svih korisnika.

http://localhost:3000/user-creation			
	Command	Target	Value
1	✓ open	http://localhost:3000/user-creation	
2	✓ set window size	758x824	
3	✓ click	id=in1	
4	✓ type	id=in1	email1@gmail.com
5	✓ type	id=in2	noviUser1
6	✓ type	css=.p-password-input	noviUser123
7	✓ click	css=.p-dropdown-label	
8	✓ click	css=.p-dropdown-item:nth-child(1)	
9	✓ click	css=.p-button-label	
10	✓ verify element not present	css=.error-message	

Slika 5.11: Parametri trećeg ispitnog slučaja

#### Running 'createUserSuccess'

1. open on http://localhost:3000/user-creation OK
2. setWindowSize on 758x824 OK
3. click on id=in1 OK
4. type on id=in1 with value email1@gmail.com OK
5. type on id=in2 with value noviUser1 OK
6. type on css=.p-password-input with value noviUser123 OK
7. click on css=.p-dropdown-label OK
8. click on css=.p-dropdown-item:nth-child(1) OK
9. click on css=.p-button-label OK
10. verifyElementNotPresent on css=.error-message OK

'createUserSuccess' completed successfully

Slika 5.12: Rezultat trećeg ispitnog slučaja

Konačno, Na slikama 5.13 i 5.14 prikazan je četvrti ispitni slučaj u kojem ispitujemo pokušaj registracije korisnika sa neispravnim podacima, odnosno sa e-mail adresom u krivom formatu. Očekivani izlaz je neuspješna registracija te prikaz poruke pogreške.

http://localhost:3000/user-creation			
	Command	Target	Value
1	✓ open	http://localhost:3000/user-creation	
2	✓ set window size	761x824	
3	✓ click	id=in1	
4	✓ type	id=in1	invalidEmail
5	✓ type	id=in2	invalidEmail
6	✓ type	css=p-password-input	invalidEmail
7	✓ click	css=p-dropdown-label	
8	✓ click	css=p-dropdown-item:nth-child(2)	
9	✓ click	css=p-button-label	
10	✗ verify element not present	css=error-message	Invalid e-mail address.

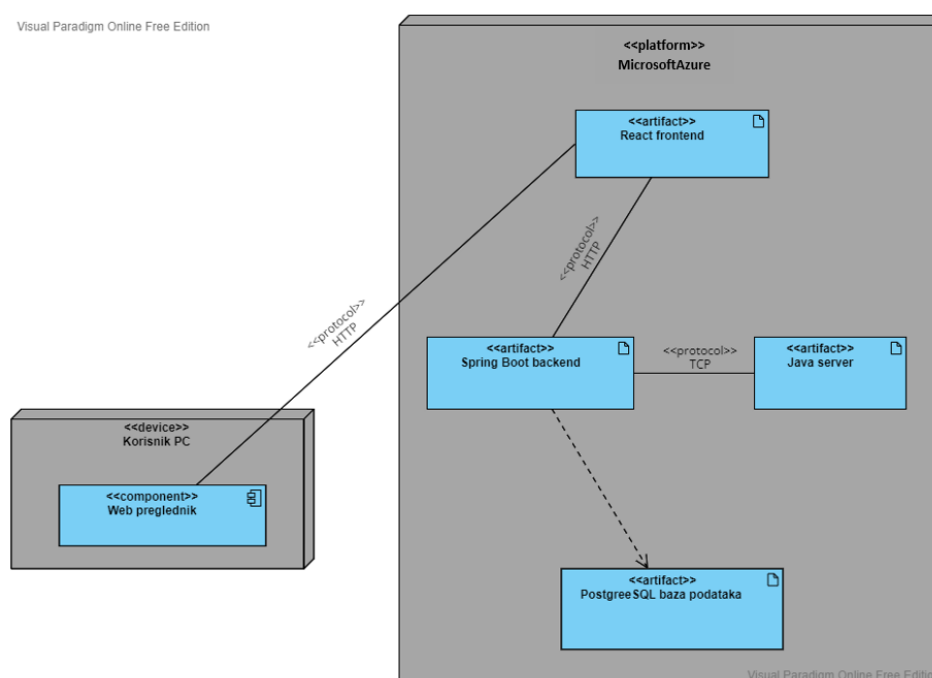
Slika 5.13: Parametri četvrtog ispitnog slučaja

```
Running 'createUserFail - invalid address'
1. open on http://localhost:3000/user-creation OK
2. setWindowSize on 761x824 OK
3. click on id=in1 OK
4. type on id=in1 with value invalidEmail OK
5. type on id=in2 with value invalidEmail OK
6. type on css=p-password-input with value invalidEmail OK
7. click on css=p-dropdown-label OK
8. click on css=p-dropdown-item:nth-child(2) OK
9. click on css=p-button-label OK
10. verifyElementNotPresent on css=error-message with value Invalid e-mail address. Failed:
    Element with locator css=error-message was found
'createUserFail - invalid address' ended with 1 error(s)
```

Slika 5.14: Rezultat četvrtog ispitnog slučaja

## 5.3 Dijagram razmještaja

Na slici 5.15 prikazan je dijagram razmještaja. Sustav je baziran na arhitekturi "klijent-poslužitelj". Korisnici pristupaju aplikaciji korištenjem web preglednika. Na platformi Microsoft Azure se nalaze poslužitelji za frontend, backend, bazu podataka te server koji implementira funkcionalnosti satelita. Komunikacija između korisnika i poslužitelja za frontend, te poslužitelja za frontend i poslužitelja za backend, ostvaruje se korištenjem protokola HTTP. Komunikacija između poslužitelja za backend i servera koji implementira funkcionalnosti satelita odvija se protokolom TCP. Komunikacija između poslužitelja za backend i baze podataka odvija se protokolom PostgreSQL.



Slika 5.15: Dijagram razmještaja

## 5.4 Upute za puštanje u pogon

Za deploy cjelokupne aplikacije korišten je cloud servis Microsoft Azure čime je omogućeno da aplikacija bude javno dostupna. Na virtualnoj mašini (distribucija Linuxa - Ubuntu 20.04.5), prije samog puštanja aplikacije u pogon, nužno je instalirati potrebnu programsku podršku. Budući da je deployment rađen na virtualnoj mašini u oblaku, sav potrebni softver instaliran je preko komandne linije. U sljedećim odlomcima bit će opisan kompletan postupak postavljanja okruženja za puštanje aplikacije u pogon.

### Konfiguracija backenda

Za potrebe naše aplikacije nužno je instalirati odgovarajuću Java verziju. U našem slučaju dovoljna je Java 11. Nadalje, potrebno je preuzeti Postgres bazu podataka. Korištena je verzija Postgres 15.1. Uz instalaciju Postgres baze potrebno je provesti standardni setup (stvoriti korisnika, podesiti port, dodijeliti memoriju). Uputstva za setup baze lako je pronaći online u službenoj dokumentaciji.

```

azureuser@satcomVM:~$ sudo -u postgres psql
psql (15.1 (Ubuntu 15.1-1.pgdg20.04+1))
Type "help" for help.

postgres=# \l

```

Name	Owner	Encoding	Collate	Ctype	ICU Locale	Locale Provider	Access privileges
dbname	postgres	UTF8	C.UTF-8	C.UTF-8		libc	
postgres	postgres	UTF8	C.UTF-8	C.UTF-8		libc	
template0	postgres	UTF8	C.UTF-8	C.UTF-8		libc	=c/postgres +
template1	postgres	UTF8	C.UTF-8	C.UTF-8		libc	=c/postgres +
							postgres=Ct/postgres

```

(4 rows)

postgres=# CREATE DATABASE satcom;
CREATE DATABASE
postgres=#

```

Slika 5.16: Prikaz konfiguracije baze

Kako bi backend dio aplikacije uspješno spremao i dohvaćao podatke potrebno je u prethodno instaliranoj Postgres bazi prije njegova pokretanja stvoriti bazu podataka pod nazivom 'satcom'. Za to je najjednostavnije koristiti defaultnog postgres korisnika koji dolazi s instalacijom postgres baze podataka te koristeći psql CLI generirati novu bazu podataka preko komandne linije. Ta baza podataka bit će popunjena podacima od strane aplikacije.

```
azureuser@satcomVM:~/satcom/IzvorniKod/Backend/satcom$ mvn install -Dmaven.test.skip=true
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to method java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Scanning for projects...
[INFO]
[INFO] -----< hr.fer.progi:satcom >-----
[INFO] Building satcom 0.0.1-SNAPSHOT
[INFO] [ jar ]-----
[INFO]
[INFO] --- maven-resources-plugin:3.2.0:resources (default-resources) @ satcom ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Using 'UTF-8' encoding to copy filtered properties files.
[INFO] Copying 1 resource
[INFO] Copying 0 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.10.1:compile (default-compile) @ satcom ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 61 source files to /home/azureuser/satcom/IzvorniKod/Backend/satcom/target/classes
[INFO] /home/azureuser/satcom/IzvorniKod/Backend/satcom/src/main/java/hr/fer/progi/satcom/security/WebSecurityConfig.java: /home/azureuser/satcom/IzvorniKod/Backend/satcom/src/main/java/hr/fer/progi/satcom/security/WebSecurityConfig.java uses or overrides a deprecated API.
[INFO] /home/azureuser/satcom/IzvorniKod/Backend/satcom/src/main/java/hr/fer/progi/satcom/security/WebSecurityConfig.java: Recompile with -Xlint:deprecation for details.
[INFO]
[INFO] --- maven-resources-plugin:3.2.0:testResources (default-testResources) @ satcom ---
```

Slika 5.17: Pokretanje maven builda

Za jednostavniji build backenda koristili smo Apache Maven project manager. Njega je također potrebno preuzeti kako bi se cjelokupna aplikacija uspješno izgradila prema postavkama definiranim unutar maven datoteke pom.xml. Ona definira određene ovisnosti (dependencies), build direktorij, izvorni direktorij, testni izvorni direktorij te ostale dodatke (plugins) koji su potrebni našoj aplikaciji da bi funkcionirala.

```
azureuser@satcomVM:~/satcom/IzvorniKod/Backend/satcom$ mvn spring-boot:run 1>/dev/null 2>&1 &
[1] 971590
azureuser@satcomVM:~/satcom/IzvorniKod/Backend/satcom$
```

Slika 5.18: Pokretanje spring aplikacije

### Konfiguracija frontenda

Za pokretanje frontend dijela aplikacije također je potrebno instalirati dodatnu programsku podršku. Prvotno je potrebno instalirati Node, također preko komandne linije, a zatim i npm koji služi kao package manager za Node aplikacije.

```
azureuser@satcomVM:~/satcom/IzvorniKod/FrontEnd$ npm install

changed 2 packages, and audited 1498 packages in 6s

219 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (1 moderate, 7 high)

To address issues that do not require attention, run:
  npm audit fix

To address all issues possible (including breaking changes), run:
  npm audit fix --force

Some issues need review, and may require choosing
a different dependency.

Run `npm audit` for details.
npm notice
npm notice New major version of npm available! 8.1.2 -> 9.3.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.3.0
npm notice Run npm install -g npm@9.3.0 to update!
npm notice
```

Slika 5.19: Instaliranje node modula

Nakon uspješno provedenih svih navedenih koraka potrebno je povući programski kod s odgovarajućeg git repozitorija.

```
azureuser@satcomVM:~/satcom$ git pull
Username for 'https://gitlab.com': laetit
Password for 'https://laetit@gitlab.com':
Updating 619d9f8B..f750d8dc
Fast-forward
 .gitignore | 3 +
 .idea/jpa-buddy.xml | 6 +
 .idea/misc.xml | 6 +
 .idea/modules.xml | 8 +
 .idea/satcom.iml | 9 +
 .idea/vcs.xml | 6 +
 .idea/workspace.xml | 93 +++
 Dokumentacija/Arhitektura.tex | 57 +-
 Dokumentacija/Dnevnik_promjena_dokumentacije.tex | 23 +-
 Dokumentacija/Dodatak.tex | 78 +-
 Dokumentacija/Implementacija.tex | 179 +++-
 Dokumentacija/Literatura.tex | 29 +
 Dokumentacija/Opis_projektnog_zadatka.tex | 10 +-
 Dokumentacija/PROGI_2022_ExpressoDepresso.log | 2225 -
 Dokumentacija/PROGI_2022_ExpressoDepresso.out | 23 -
 Dokumentacija/PROGI_2022_ExpressoDepresso.pdf | Bin 1115867 -> 3485164 bytes
 Dokumentacija/PROGI_2022_ExpressoDepresso_sunstex.pdf | Bin 201168 -> 6 bytes
```

Slika 5.20: Prikaz pullanja koda s gita

Nakon uspješnog pulla imamo sve potrebne resurse za pokretanje aplikacije te je samo potrebno runnati maven projekte i node aplikaciju u odgovarajućim direktorijima na virtualnoj mašini. Poželjno je aplikaciju staviti da se vrti kao pozadinski proces na virtualnoj mašini.



## 6. Zaključak i budući rad

Zadatak naše grupe bio je razvoj web aplikacije pod nazivom "SatCom". Ideja aplikacije je da se korisnicima omogući komunikacija sa satelitima u mreži SatNogs. Nakon 12 tjedana timskog rada, ostvarili smo zadani cilj i projekt je završen. Projekt je imao tri faze.

Prva faza je uključivala okupljanje tima, razgovor o generalnim idejama za aplikaciju, izražavanje pojedinačnih interesa i želja za radom u prvom ciklusu predaje, te konačno podjela zadataka za prvi ciklus. Formirala su se dva podtima. U prvom podtimu članovi su radili na implementaciji funkcionalnosti na frontendu. U drugom podtimu su članovi su radili na implementaciji funkcionalnosti na backendu. Rad na dokumentaciji podijeljen je na više manjih zadataka koje su pojedinci samostalno preuzimali. Prva je faza projekta trajala do kolokviranja prvog ciklusa projekta.

U drugoj je fazi naglasak bio na implementaciji aplikacije, te se i ovdje zadržala podjela u dva podtima koji su radili na programskom ostvarenju aplikacije. U ovoj je fazi dovršena implementacija aplikacije, a trajala je do demonstracije alfa inačice aplikacije.

Treća i konačna faza je uključivala izradu raznih UML dijagrama, ispitivanje sustava, pronalazak i ispravak grešaka. U ovoj fazi također je zadržana podjela na dva podtima te su zadatke vezane uz dokumentaciju članovi tima preuzimali samostalno. Ova je faza trajala do završetka projekta, prije konačne predaje i kolokviranja drugog ciklusa.

Izgrađenu aplikaciju je moguće proširiti na mnogo načina. Jedna od ideja je da se implementira stranica koja bi prikazivala geolokacije pojedinih stanica i informacije o njima poput uspješnosti u slanju poruka. Sudjelovanje u ovom projektu je bilo vrijedno iskustvo za sve članove tima. Svima nama je ovo bio prvi ozbiljniji grupni projekt, i snašli smo se jako dobro. Konflikata u timu nije bilo, a suradnja i komunikacija su bili na iznimno zadovoljavajućoj razini. Većini nas je ovaj projekt bio prvi ozbiljniji dodir s tehnologijama poput Gita i Latex. Naučili smo koristiti neke moderne radne okvire pri izradi web aplikacija. Iznimno smo zadovoljni postignutim rezultatima i timskim radom koji je do tih rezultata doveo.

# Popis literature

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. SatNOGS projekt, <https://satnogs.org/about/>
3. SatNOGS Network, <https://network.satnogs.org/>
4. SatNOGS Database, <https://db.satnogs.org/>
5. SatNOGS Network API, <https://librespacefoundation.gitlab.io/-/satnogs/satnogs-network/-/jobs/3313398089/artifacts/satnogs-network-api-client/html2/index.html>
6. The Unified Modeling Language, <https://www.uml-diagrams.org/>
7. Astah Community, <http://astah.net/editions/uml-new>
8. BezKoder, <https://www.bezkoder.com/spring-boot-jwt-authentication/>
9. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
10. Junit5 <https://junit.org/junit5/>
11. Unit testiranje, Informacija, logika i jezici , FER ZTEL, [https://www.fer.unizg.hr/predmet/ilj\\_b](https://www.fer.unizg.hr/predmet/ilj_b)
12. Java Networking, <https://docs.oracle.com/javase/8/docs/technotes/guides/net/index.html>
13. Scheduling Tasks, Spring.io, <https://spring.io/guides/gs/scheduling-tasks/>

# Indeks slika i dijagrama

3.1	Dijagram obrasca uporabe, funkcionalnost neprijavljenog korisnika i običnog uporabnog korisnika . . . . .	28
3.2	Dijagram obrasca uporabe, funkcionalnost Administratora stranice	29
3.3	Dijagram obrasca uporabe, funkcionalnost Administratora satelita .	30
3.4	Sekvencijski dijagram prijave korisnika . . . . .	31
3.5	Sekvencijski dijagram slanja poruke . . . . .	32
3.6	Sekvencijski dijagram biranja stanice . . . . .	33
4.1	Dijagram baze podataka . . . . .	40
4.2	Dijagram razreda Controllers . . . . .	41
4.3	Dijagram razreda Models . . . . .	42
4.4	Dijagram stanja . . . . .	44
4.5	Dijagram aktivnosti . . . . .	46
4.6	Dijagram komponenti frontend dijela aplikacije . . . . .	47
4.7	Dijagram komponenti backend dijela aplikacije . . . . .	48
5.1	Testiranje spajanja i dohvaćanja kompatibilnih satelita i linkova . .	51
5.2	Testiranje spajanja i dohvaćanja kompatibilnih stanica i linkova . .	52
5.3	Testiranje ispitivanja jednakosti stanica u bazi i na SatNOGS API-ju	53
5.4	Testiranje spremanja poruka u bazu. . . . .	54
5.5	Testiranje neispravne komunikacije sa serverom koji "glumi" satelit.	55
5.6	Testiranje ispravne komunikacije sa serverom koji "glumi" satelit. .	55
5.7	Parametri prvog ispitnog slučaja . . . . .	56
5.8	Rezultat prvog ispitnog slučaja . . . . .	56
5.9	Parametri drugog ispitnog slučaja . . . . .	57
5.10	Rezultat drugog ispitnog slučaja . . . . .	57
5.11	Parametri trećeg ispitnog slučaja . . . . .	58
5.12	Rezultat trećeg ispitnog slučaja . . . . .	58
5.13	Parametri četvrtog ispitnog slučaja . . . . .	59
5.14	Rezultat četvrtog ispitnog slučaja . . . . .	59
5.15	Dijagram razmještaja . . . . .	60

5.16 Prikaz konfiguracije baze . . . . .	61
5.17 Pokretanje maven builda . . . . .	62
5.18 Pokretanje spring aplikacije . . . . .	62
5.19 Instaliranje node modula . . . . .	63
5.20 Prikaz pullanja koda s gita . . . . .	63
6.1 Graf gitLab aktivnosti . . . . .	74
6.2 Graf gitLab aktivnosti-nastavak . . . . .	75

# Dodatak: Prikaz aktivnosti grupe

## Dnevnik sastajanja

### 1. sastanak

- Datum: 20.10.2022.
- Trajanje: 2h
- Prisustvovali:  
M. C. Belušić Gonzales, L. Crnković, D. Grdić, M. Pristav, L. Salihović, M. Vidas, I. Žgela
- Teme sastanka:
  - Upoznavanje s zadatkaom
  - Izlučivanje zahtjeva aplikacije
  - Razriješavanje inicijalnih nedoumica
  - Odabir baznih tehnologija
  - Rasprava i opisivanje relacijskog modela baze podataka
  - Opisivanje osnovnog UI izgleda
  - Proučavanje API dokumentacije

### 2. sastanak

- Datum: 25.10.2022.
- Trajanje: 3h
- Prisustvovali:  
M. C. Belušić Gonzales, L. Crnković, D. Grdić, M. Pristav, L. Salihović, M. Vidas, I. Žgela
- Teme sastanka:
  - Upoznavanje s detaljima dokumentacije
  - Započeta podjela posla oku dokumentacije
  - Određeni načini praćenja rada u timu

### 3. sastanak

- Datum: 26.10.2022.
- Trajanje: 1.5h

- Prisustvovali:  
M. C. Belušić Gonzales, L. Salihović, I. Žgela
- Teme sastanka:
  - Definiranje općenit dizaj i boje web stranice
  - Definiranje komponente koje je potrebno implementirati
  - Dogovor oko raspodjele poslova do slijedećeg sastanka

#### 4. sastanak

- Datum: 30.10.2022.
- Trajanje: 2.5h
- Prisustvovali:  
L. Crnković, D. Grdić, M. Pristav, M. Vidas
- Teme sastanka:
  - Otkriveni i prodiskutirani problemi sa spajanjem na API
  - Osvježen UML baze podataka
  - Istraživanje okolnosti našeg zadatka  
(mogućnosti transmitera, antena, satelita...)
  - Istraživanje konkretnog formata potrebnih podataka

#### 5. sastanak

- Datum: 31.10.2022.
- Trajanje: 1h
- Prisustvovali:  
M. C. Belušić Gonzales, I. Žgela
- Teme sastanka:
  - Zajedničko rješavanje konflikata
  - Zajedničko rješavanje problema prijave korisnika u aplikaciju
  - Raspodjela zadataka i uređivanje gitlab board-a
  - Međusobno pokazivanje koda i konzultacija za nastavak rada na aplikaciji

#### 6. sastanak

- Datum: 2.11.2022.
- Trajanje: 3.5h
- Prisustvovali:  
L. Crnković, D. Grdić, M. Pristav, M. Vidas
- Teme sastanka:

- Testiranje get zahtjeva na API-ju
- Planiranje daljnjeg rada backenda
- Pojašnjenje novih aspekata dokumentacije
- ispravljanje prijašnjih grešaka u radu i shvaćanju zadatka
- Određen termin za zajedničko stvaranje dijela dokumentacija

#### 7. sastanak

- Datum: 3.11.2022.
- Trajanje: 4h
- Prisustvovali:  
L. Crnković, D. Grdić, M. Pristav, L. Salihović, M. Vidas, I. Žgela
- Teme sastanka:
  - Sastanak sa asistentom
  - Provjera dosadašnjeg rada i uspunjenja zahtjeva zadatka
  - Odlučen način i redovitost povezivanja na satnogs network
  - Dovršen model baze podataka
  - Napravljeni dijagrami obrazaca uporabe
  - Usklađivanje workspaceova na frontendu
  - Dogovor oko načina predaje formi

#### 8. sastanak

- Datum: 10.11.2022.
- Trajanje: 4h
- Prisustvovali:  
M. C. Belušić Gonzales, L. Crnković, D. Grdić, M. Pristav, L. Salihović, M. Vidas, I. Žgela
- Teme sastanka:
  - Usklađivanje frontend i backend ideja i dovršenog posla
  - Spajanje frontend i backend koda
  - Dovođenje funkcionalnosti LogIn/LogOut
  - Dodavanje previđenih Obrazaca Uporabe
  - Započeta modelacija Sekvencijskih dijagrama
  - Zajednička provjera/pregled dijelova dokumentacije
  - Dodane manje preinake u bazu podataka
  - Raspravljene moguće promjene o finalnom načinu rada aplikacije

#### 9. sastanak

- Datum: 18.11.2022.

- Trajanje: 3h
- Prisustvovali:  
L. Crnković, D. Grdić, M. Vidas, I. Žgela
- Teme sastanka:
  - Deployment aplikacije
  - Testiranje deployane aplikacije

## 10. sastanak

- Datum: 19.12.2022.
- Trajanje: 4h
- Prisustvovali:  
M. C. Belušić Gonzales, L. Crnković, D. Grdić, M. Pristav, L. Salihović, M. Vidas, I. Žgela
- Teme sastanka:
  - Osvrt na demonstraciju alfa inačice
  - Raspodjela preostale implementacije nakon sastanka s asistentom
  - Postavljen rok za dovršetak implementacije i početak dokumentacije
  - Dogovoren termin sljedećeg sastanka

## 11. sastanak

- Datum: 5.1.2023.
- Trajanje: 2h
- Prisustvovali:  
M. C. Belušić Gonzales, L. Crnković, D. Grdić, M. Pristav, L. Salihović, M. Vidas, I. Žgela
- Teme sastanka:
  - Zajednički pregled funkcionalnosti aplikacije
  - Pregled potrebnih znanja za daljnju dokumentaciju
  - Raspodjela posla oko dokumentacije
  - Predložene manje promjene u dizajnu aplikacije



## Tablica aktivnosti

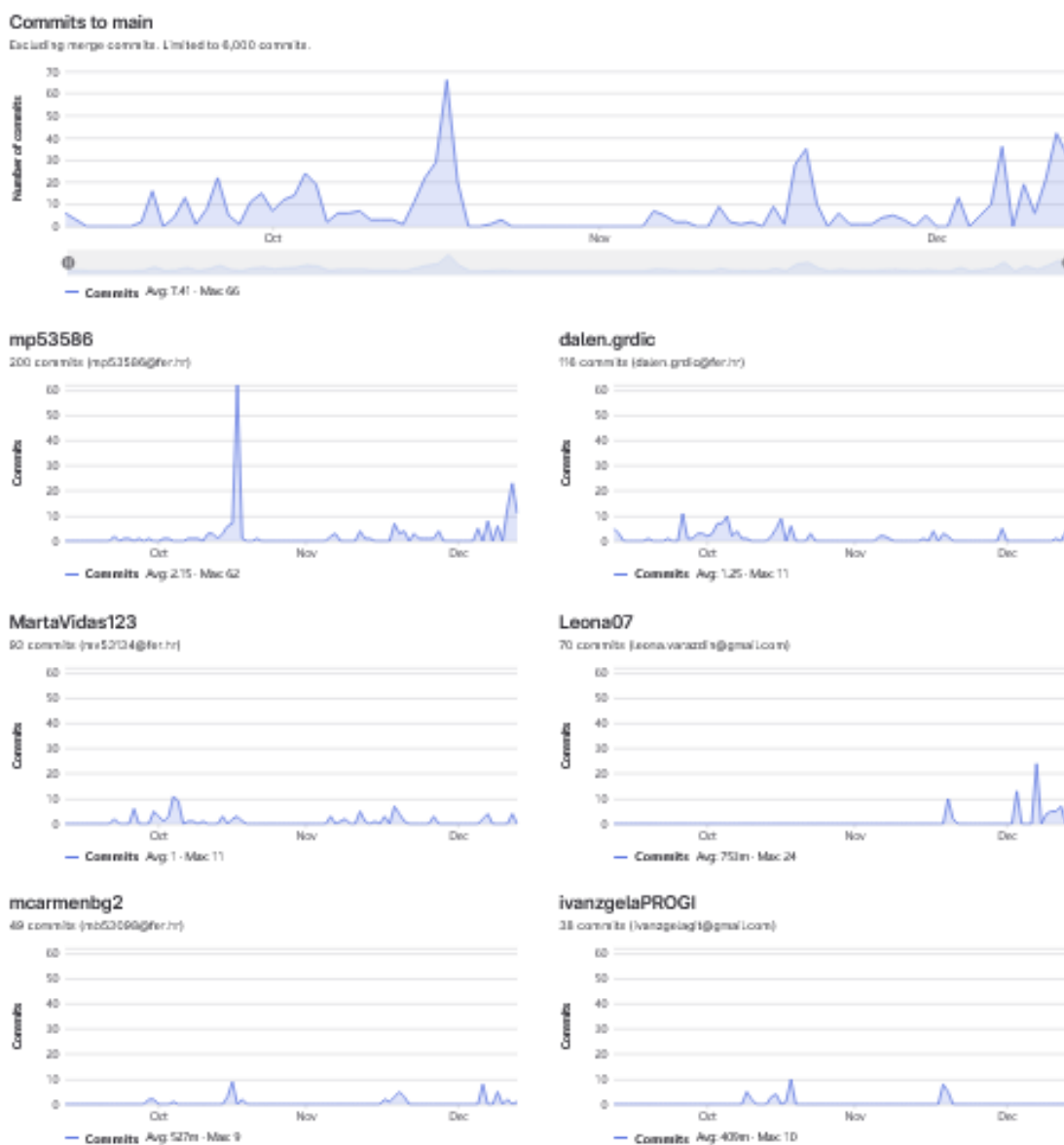
	Mihael Pristav	Maria Carmen Belušić Gonzales	Leticija Crnković	Dalen Grdić	Leona Salihović	Marta Vidas	Ivan Žgela
Opis projektnog zadatka	1	5					
Funkcionalni zahtjevi	1				2	0.5	
Opis pojedinih obrazaca	11				2.5	2	0.5
Dijagram obrazaca	1.5						
Sekvencijski dijagrami	1						
Opis ostalih zahtjeva						0.5	
Arhitektura i dizajn sustava						0.5	
Baza podataka	2					4.5	2.5
Dijagram razreda			1.5			0.5	
Dijagram stanja							2
Dijagram aktivnosti			1	1			
Dijagram komponenti				1		1	
Korištene tehnologije i alati	0.5				2		
Ispitivanje programskog rješenja	1	2.5				1	
Dijagram razmještaja					1		
Upute za puštanje u pogon			4				

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Mihael Pristav	Maria Carmen Belušić Gonzales	Leticija Crnković	Dalen Grdić	Leona Salihović	Marta Vidas	Ivan Žgela
Dnevnik sastajanja	2						0.5
Zaključak i budući rad					2		
Popis literature	1						
<b>Ukupno dokumentacija</b>	22	7.5	6.5	2	9.5	10.5	5.5
<i>Upravljanje projektom</i>	11						
<i>back end</i>	24		17	30		36	
<i>front end</i>		28			22		42
<i>deployment</i>		4	17	15	5	10	3
<i>Istraživanje informacija/tehnologija</i>	10	12	7.5	7.5	10	6.5	11
<i>izrada baze podataka</i>	2		1	1		1	
<i>izrada dijagrama</i>	10	3.5	8	11.5	3	7	4
<i>izrada dizajna aplikacije u Figmi</i>					6		
<b>Ukupno razvoj projekta</b>	79	55	57	67	55.5	71	65.5

## Dijagrami pregleda promjena



Slika 6.1: Graf gitLab aktivnosti



Slika 6.2: Graf gitLab aktivnosti-nastavak