

Oblikovanje programske potpore

Ak. god. 2019./2020.

Giger

Dokumentacija, Rev. 1

Grupa: *ChillCrew*

Voditelj: *Ivan Juren*

Datum predaje: 15. 11. 2019.

Nastavnik: *Tomislav Jukić*

Sadržaj

1 Dnevnik promjena dokumentacije	2
2 Opis projektnog zadatka	4
2.1 Tijek i opseg aplikacije	4
2.1.1 Opcionalna proširenja aplikacije	5
2.2 Slična rješenja problema	6
2.2.1 Amy	6
2.2.2 BandFriend	6
3 Specifikacija programske potpore	8
3.1 Funkcionalni zahtjevi	8
3.1.1 Obrasci uporabe	10
3.1.2 Sekvencijski dijagrami	25
3.2 Ostali zahtjevi	29
4 Arhitektura i dizajn sustava	31
4.1 Baza podataka	33
4.1.1 Opis tablica	34
4.1.2 Dijagram baze podataka	42
4.2 Dijagram razreda	43
Popis literature	49
Indeks slika i dijagrama	50
Dodatak: Prikaz aktivnosti grupe	51

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak, modificirana glavna .tex datoteka	Lanča	26.10.2019.
0.2	Unesen dnevnik sastanaka	Lanča	27.10.2019.
0.3	Dodani <i>Use Case</i> dijagrami br. 2, 3, 4, 5, 6	Jurić	29.10.2019.
0.4	Dodan <i>Use Case</i> dijagram br. 9	Nosil	29.10.2019.
0.5	Dodani <i>Use Case</i> dijagrami br. 10, 11, 12, 13, 17, 18	Gaši	29.10.2019.
0.6	Dodani <i>Use Case</i> dijagrami br. 1, 7, 8, 14, 15, 16, 19, 20	Zec	30.10.2019.
0.7	Dodani funkcionalni zahtjevi	Zec	04.11.2019.
0.8	Nadopunjen zapisnik sastanka, dodani ostali zahtjevi	Lanča	04.11.2019.
0.9	Dodan opis projektnog zadatka	Gaši	05.11.2019.
0.10	Dodani <i>Use Case</i> dijagrami br. 21-35	Zec	07.11.2019.
0.11	Rastavljeni neki <i>Use Case</i> -ovi na više njih	Gaši	08.11.2019.
0.12	Dodane opisne tablice baze	Gaši	08.11.2019.
0.13	Uneseni sekvencijski dijagrami njihov opis	Jurić, Gaši	12.11.2019.
0.14	Unesen opis arhitekture	Lanča	12.11.2019.
0.15	Unesen opis tablica	Krmek	12.11.2019.
0.16	Dodani dijagrami obrazaca uporabe	Zec	12.11.2019.
0.17	Promijenjeni opisi aktora i use case 1 do 15	Nosil	12.11.2019.
0.18	Dodana slika u opis arhitekture te dodan zahtjev	Lanča	12.11.2019.
0.19	Promjenjeni Use case-ovi 16 do 36	Nosil	12.11.2019.
0.20	Izbrisan duplikat arhitekture, osvježen dnevnik sastajanja	Lanča	14.11.2019.
0.21	Popunjena tablica aktivnosti	Lanča	14.11.2019.
0.22	Dodan odjeljak o deploymentu	Juren	14.11.2019.
0.23	Uklonjeni suvišni dijelovi predloška	Juren	14.11.2019.

Rev.	Opis promjene/dodatka	Autori	Datum
0.24	Dodani DTO i Controllers dijagrami	Zec	14.11.2019.
0.25	Dodani ostali opisi tablica baze podataka	Krmek	15.11.2019.
0.26	Lektorirana dokumentacija	Lanča	15.11.2019.
0.27	Dodani dijagrami Service	Zec	15.11.2019.
0.28	Dodani dijagrami razreda entiteta	Zec	15.11.2019.
0.29	Dodani security dijagram	Zec	15.11.2019.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	Gaši	15.11.2019.

2. Opis projektnog zadatka

Cilj ovog projekta je razviti programsku podršku za stvaranje web aplikacije *Giger* koja je namijenjena glazbenicima, ljudima kojima su potrebne usluge glazbenika (organizatori različitih proslava, menadžeri) te ljudima koji su zainteresirani za obližnje događaje. Na taj način na jednom mjestu se omogućava:

- upoznavanje glazbenika i njihovo povezivanje u bendove
- olakšana komunikacija i organizacija unutar bendova
- promocija glazbenika
- dogovaranje nastupa s organizatorima koji jednostavno mogu pronaći prikladan bend za određeni događaj
- pregled zanimljivih nadolazećih događaja dostupan svima
- uvid u recenzije bendova/organizatora

Organizacija osobnog kalendara i dogovaranje termina koji zahtijevaju prisustvo više ljudi je težak zadatak, a s tim se problemom na gotovo dnevnoj razini susreću glazbenici kada dogovaraju nastupe. Isto tako, ljudi koji organiziraju razne proslave za koje trebaju glazbenike, kao i ljudi koji žele otići na neku živu svirku, ponekad ne znaju kakva je glazbena ponuda u njihovoj okolini, a ako su i čuli za neki događaj u blizini, ne postoji jedinstveno mjesto gdje mogu pročitati recenzije o glazbenicima, bendu ili organizatoru kako bi bili sigurni u kvalitetu događaja. *Giger* je platforma koja rješava navedene probleme integrirajući kalendar glazbenika sa servisom namijenjenim za sastajanje bendova i organizatora nastupa.

Česta je situacija da je jedan glazbenik član više bendova pa tako dostupnost svakog benda ovisi o dostupnosti svih njegovih članova. Kalendar svakog glazbenika ujedinjuje sve obaveze iz bendova u kojima je član pa dostupnost benda postaje trivijalna informacija.

2.1 Tijek i opseg aplikacije

Prilikom pokretanja aplikacije, neregistriranom korisniku prikazuju se opće informacije o javnim događajima te mu se nudi mogućnost prijavljivanja u sustav s

postojećim računom (potrebno upisati email i lozinku) te kreiranje novog računa. Za stvaranje novog računa potrebni su:

- korisničko ime
- email adresa
- lozinka

Registrirani korisnik može pregledati, mijenjati osobne podatke i izbrisati svoj korisnički račun. Takvom korisniku nudi se mogućnost pisanja recenzija te razmjenjivanja poruka s drugim korisnicima. On u svojim postavkama može postati glazbenik i/ili organizator.

Glazbenik može odabrati instrumente koje svira, osnovati bend, pridružiti se postojećem, dodati članove u bend te uređivati svoj profil i kalendar. On na svojoj stranici može dodavati različite medije te se tako promovirati.

Organizator ima mogućnost kreiranja nastupa. On može filtrirati bendove prema vrsti glazbe, tipu nastupa, lokaciji i drugo. Nakon što dobije listu raspoloživih bendova, može pregledavati njihove profile. Profil benda prikazuje osnovne informacije o bendu, razne novosti koje uređuju njegovi članovi, popis nadolazećih javnih nastupa te recenzije korisnika. Organizator ima mogućnost kontaktiranja benda putem poruka ugrađenih u aplikaciju. Imajući takav skup podataka, *Giger* svojim korisnicima, običnim ljudima željnim zabave, može preporučiti nadolazeće događaje u njihovoj blizini.

Uz glazbenika i organizatora postoji i uloga administratora koji ima mogućnost uređivanja popisa ponuđenih instrumenata te blokiranja korisnika.

Aplikacija će biti izvedena kao web aplikacija prilagođena (engl. responsive) mobilnom uređaju i podržavat će rad više paralelnih korisnika sa sučeljem koji je jednostavan za korištenje kako bi korisnici imali što bolje iskustvo.

2.1.1 Opcionalna proširenja aplikacije

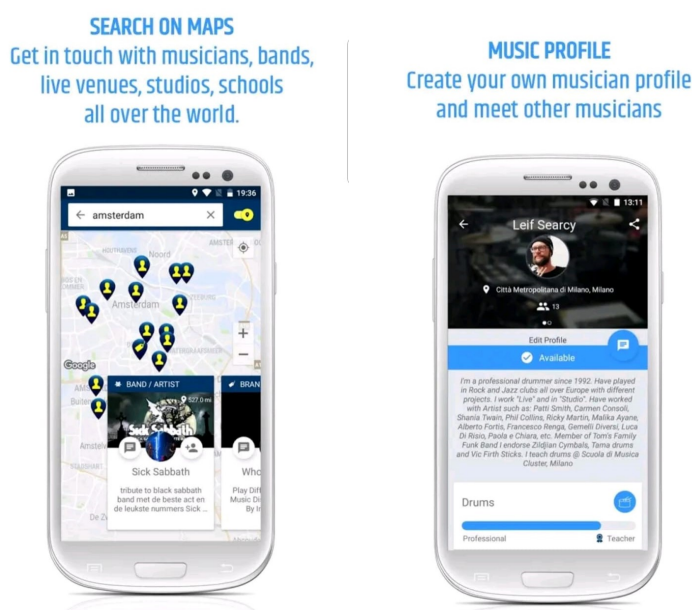
- Bendovi mogu postaviti oglas da traže nove članove, na koje se glazbenici mogu javljati
- Moguće organizirati „Nasumična druženja“ da se hrpa glazbenika nađe i sviraju zajedno bez da su dio istog benda.
- Dodavanje komentara na objave

- Dodavanje novih funkcionalnosti za organizatore (mogućnost unajmljivanja ton majstora, fotografa itd.)
- Dodati „Podijeli na Facebook“ mogućnost za javne nastupe

2.2 Slična rješenja problema

2.2.1 Amy

Amy je mobilna aplikacija koja nakon registracije nudi različite opcije vrste korisnika (glazbenik, DJ i slično). Nakon odabira vrste korisnika nudi i unos instrumenata koje korisnik svira te odabir razine profesionalnosti na pojedinom instrumentu. Također, nudi i odabir glazbenih žanrova, kalendar s obavezama te pregledavanje glazbenika u blizini. Aplikacija ima jako puno potencijala, ali se često ruši što utječe na iskustvo korisnika.

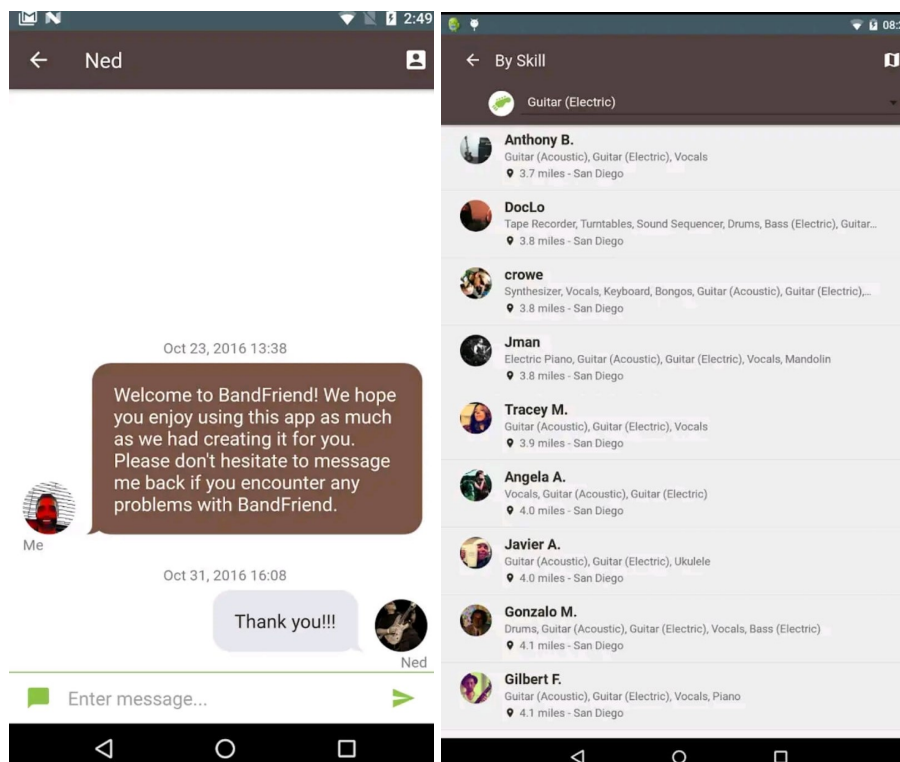


Slika 2.1: Prikaz *Amy* aplikacije

2.2.2 BandFriend

BandFriend je mobilna aplikacija koja prilikom registracije traži dosta informacija o korisniku. Nakon izrade profila, mogu se pretraživati novi glazbenici, glazbenici najbliži trenutnom korisniku, najbliži po lokaciji i slično. Aplikacija nudi

i razgovor porukama s drugim korisnicima. Neke korisnike tolika lista zahtjeva prilikom registracije može odbiti te će *Giger* tražiti samo korisničko ime, email i lozinku, a kasnije svaki korisnik, ukoliko to želi, može dodati više informacija o sebi. Također, *BandFriend* ne nudi kalendar s obavezama korisnika, stvaranje bendova ili događaja.



Slika 2.2: Prikaz *BandFriend* aplikacije

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Naručitelj
2. Javnost
3. Korisnici
4. Organizator
5. Glazbenik
6. Voditelj benda
7. Administrator
8. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neprijavljeni/neregistrirani korisnik (javnost) može:
 - (a) Pregledati javne događaje
 - (b) Pregledati profil benda
 - (c) Pročitati recenzije
 - (d) Registrirati se na aplikaciji
2. Korisnik (inicijator) može:
 - (a) Sve što i Javnost
 - (b) Prijaviti se na aplikaciju
 - (c) Pregledavati i uređivati korisnički profil
 - (d) Pregledati postojeće poruke s drugim korisnicima
 - (e) Pregledati svoj kalendar
 - (f) Slati nove poruke drugim korisnicima
 - (g) Recenzirati događaje
 - (h) Komentirati objave glazbenika
 - (i) Postati glazbenik i/ili organizator

- (j) Pisati objave na svom profilu
- (k) Odjaviti se

3. Glazbenik (inicijator) može:

- (a) Sve što i Korisnik
- (b) Prihvatiti ili odbiti poziv u bend
- (c) Postati članom jednog ili više bendova
- (d) Održati samostalan nastup
- (e) Stvoriti bend

4. Organizator (inicijator) može:

- (a) Sve što i Korisnik
- (b) Pretraživati bendove na aplikaciji
- (c) Stvarati događaje
- (d) Uređivati događaje
- (e) Recenzirati bend kao organizator događaja

5. Voditelj benda (inicijator) može:

- (a) Sve što i Glazbenik
- (b) Pozivati i dodavati nove članove u bend
- (c) Dodavati obaveze u kalendar benda
- (d) Vidjeti kalendare članova benda
- (e) Recenzirati organizatora u ime benda
- (f) Napisati objave na stranici benda
- (g) Uređivati profil benda

6. Administrator (inicijator) može:

- (a) Blokirati korisnike koji su prekršili uvjete korištenja aplikacije
- (b) Uređivati popis instrumenata
- (c) Dodati događaje koji su stvoreni na drugim platformama

7. Baza podataka (sudionik) može:

- (a) Pohraniti podatke o korisnicima, glazbenicima, organizatorima i bendovima
- (b) Pohraniti podatke o nastupima, recenzijama, objavama, komentarima
- (c) Pohraniti razmijenjene poruke između korisnika

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1 - Pristup listi javnih događaja

- **Glavni sudionik:** Javnost
- **Cilj:** Vidjeti listu javnih događaja na aplikaciji
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Javnost na početnoj stranici odabire prikaz događaja
 2. Aplikacija prikazuje javne događaje

UC2 - Pregled profila benda

- **Glavni sudionik:** Javnost
- **Cilj:** Vidjeti profil benda
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Javnost odabire bend čiji profil želi pregledati
 2. Aplikacija prikazuje profil odabranog benda

UC3 - Pregled recenzija

- **Glavni sudionik:** Javnost
- **Cilj:** Vidjeti pregled recenzija
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Prioritet:** Medium
- **Opis osnovnog tijeka:**
 1. Javnost odabire popis recenzija
 2. Aplikacija prikaže popis recenzija
 3. Javnost odabere recenziju koju želi vidjeti
 4. Aplikacija korisniku prikaže odabranu recenziju

UC4 - Registracija

- **Glavni sudionik:** Javnost
- **Cilj:** Registrirati se
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Javnost odabire opciju za registraciju
 2. Aplikacija prikazuje prozor za unos podataka
 3. Javnost unosi tražene korisničke podatke
 4. Aplikacija korisniku šalje email za verifikaciju email-a
 5. Korisnik prima obavijest o uspješnoj registraciji
- **Opis mogućih odstupanja:**
 - 2.a Odabir već zauzetog korisničkog imena i/ili e-maila, unos podataka u nedozvoljenom formatu
 1. Sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za registraciju.
 2. Korisnik mijenja potrebne podatke ili odustaje od registracije.

UC5 - Prijava u sustav

- **Glavni sudionik:** Korisnik
- **Cilj:** Korištenje aplikacije
- **Sudionici:** Baza podataka
- **Preduvjet:** Napravljena registracija
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju "Prijava"
 2. Aplikacija prikazuje prozor za prijavu
 3. Korisnik unosi podatke
 4. Baza autorizira unesene podatke te dozvoljava korisniku korištenje aplikacije
 5. Aplikacija korisniku prikazuje početnu stranicu
- **Opis mogućih odstupanja:**
 - 2.a Aplikacija ne dozvoljava Korisniku korištenje aplikacije ako su podaci neispravni

UC6 - Uvid u popis dodanih instrumenata

- **Glavni sudionik:** Administrator
- **Cilj:** Izmjena krivo unesenih instrumenata u bazi
- **Sudionici:** Baza podataka
- **Preduvjet:** Administrator je prijavljen u sustav
- **Prioritet:** Low
- **Opis osnovnog tijeka:**
 1. Administrator odabire pregled dodanih instrumenata
 2. Baza ispisuje dodane instrumente
 3. Administrator mijena krivo unesene zapise

UC7 - Blokiranje korisnika koji su prekršili uvjete korištenja

- **Glavni sudionik:** Administrator
- **Cilj:** Blokirati korisnike koji krše uvjete korištenja aplikacije
- **Sudionici:** Baza podataka
- **Preduvjet:** Administrator je prijavljen u sustav
- **Prioritet:** Low
- **Opis osnovnog tijeka:**
 1. Administrator uočava korisnika koji krši opća pravila korištenja sustava
 2. Administrator blokira korisnički račun određenoga korisnika na neodređeno vrijeme

UC8 - Dodavanje događaja

- **Glavni sudionik:** Administrator
- **Cilj:** Dodati događaje koji su kreirani van sustava u sustav
- **Sudionici:** Baza podataka
- **Preduvjet:** Administrator je prijavljen u sustav
- **Prioritet:** Low
- **Opis osnovnog tijeka:**
 1. Administrator odabire opciju za dodavanje događaja u sustav
 2. Administrator dodaje događaj u sustav

UC9 - Pregled poruka

- **Glavni sudionik:** Korisnik
- **Cilj:** Uvid u postojeće poruke
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen

- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Korisnik odabire "Poruke"
 2. Aplikacija prikazuje osobe s kojima se već vodio razgovor
 3. Korisnik odabire osobu te se prikazuju poruke
- **Opis mogućih odstupanja:**
 - 2.a Korisnik nema prošlih poruka
 1. Aplikacija prikazuje poruku „Nema poruka“

UC10 - Pisanje poruke

- **Glavni sudionik:** Korisnik
- **Cilj:** Komunikacija među korisnicima
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Korisnik odabire poruke
 2. Aplikacija prikazuje korisnikove razgovore
 3. Korisnik odabire drugog korisnika s kojim želi komunicirati
 4. Aplikacija prikazuje postojeće poruke s odabranom osobom
 5. Korisnik unosi novu poruku
 6. Korisnik odabere „Pošalji“ za slanje poruke
- **Opis mogućih odstupanja:**
 - 3.a Korisnik želi poslati poruku osobi s kojom još nije komunicirao
 1. Korisnik odabire opciju „Nova poruka“
 2. Pronalazi korisnika kojemu želi napisati poruku

UC11 - Pisanje recenzije o bendu

- **Glavni sudionik:** Korisnik
- **Cilj:** Napisati recenziju o bendu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti prijavljen
- **Prioritet:** Medium
- **Opis osnovnog tijeka:**
 1. Korisnik pristupa profilu benda
 2. Korisnik u okvir za recenzije napiše svoje mišljenje te da ocjenu

3. Korisnik odabere "Završi recenziju"

UC12 - Pisanje recenzije za događaj

- **Glavni sudionik:** Korisnik
- **Cilj:** Napisati recenziju za događaj
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Prioritet:** Medium
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za recenziranje događaja
 2. Aplikacija otvara prozor za unos recenzije
 3. Korisnik u okvir za recenzije napiše svoje mišljenje te da ocjenu
 4. Korisnik odabere "Završi recenziju"
 5. Aplikacija sprema recenziju

UC13 - Pregled profila korisnika

- **Glavni sudionik:** Korisnik
- **Cilj:** Pregled profilne stranice korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Korisnik odabire prikaz profila korisnika
 2. Aplikacija prikazuje profil korisnika
 - (a) Ako je glazbenik onda mu se prikazuju dodatne informacije (kalendar, instrumenti, popis događaja na kojima svira)
 - (b) Ako je organizator onda mu se prikazuju dodatne informacije (menager name i recenzije kao organizator)

UC14 - Komentiranje objava glazbenika

- **Glavni sudionik:** Korisnik
- **Cilj:** Komentiranje objave glazbenika
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Prioritet:** Medium
- **Opis osnovnog tijeka:**

1. Korisnik odabire "Komentiraj" na objavi glazbenika
2. Aplikacija otvara prozor za unos komentara
3. Korisnik piše komentar
4. Korisnik odabere "Spremi"
5. Aplikacija sprema objavu

UC15 - Uređivanje profila korisnika

- **Glavni sudionik:** Korisnik
- **Cilj:** Promjeniti informacije na profilu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za uređivanje profila
 2. Korisnik mijenja podatke
 3. Korisnik odabirom opcije za spremanje potvrđuje i sprema načinjene promjene
- **Opis mogućih odstupanja:**
 - 3.a Korisnik nije unio obavezne podatke ili su podatci nevaljani
 1. Korisnik ponovno unosi obavezne podatke
 2. Korisnik odustaje od uređivanja profila

UC16 - Stvaranje profila organizatora

- **Glavni sudionik:** Korisnik
- **Cilj:** Dodavanje privilegija organizatora korisniku
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Korisnik odabire uređivanje profila
 2. Korisnik odabire opciju "Organizator"
 3. Aplikacija prikazuje dodatna polja za unos informacija
 4. Korisnik popunjava potrebne informacije
 5. Korisnik spremanjem promjena postaje Organizator
- **Opis mogućih odstupanja:**
 - 5.a Korisnik nije unio obavezne podatke ili su podatci nevaljani

1. Korisnik ponovno unosi obavezne podatke
2. Korisnik odustaje od izmjene profila

UC17 - Stvaranje profila glazbenika

- **Glavni sudionik:** Korisnik
- **Cilj:** Dodavanje privilegija glazbenika korisniku
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Korisnik odabire uređivanje profila
 2. Korisnik odabire opciju "Glazbenik"
 3. Aplikacija prikazuje dodatna polja za unos informacija
 4. Korisnik popunjava potrebne informacije
 5. Korisnik spremanjem promjena postaje Glazbenik
- **Opis mogućih odstupanja:**
 - 4.a Korisnik nije unio obavezne podatke ili su podatci nevaljani
 1. Korisnik ponovno unosi obavezne podatke
 2. Korisnik odustaje od izmjene profila

UC18 - Odjava

- **Glavni sudionik:** Korisnik
- **Cilj:** Odjavljivanje iz aplikacije
- **Sudionici:** -
- **Preduvjet:** Korisnik je prijavljen
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za odjavu

UC19 - Uređivanje profila glazbenika

- **Glavni sudionik:** Glazbenik
- **Cilj:** Promijeniti informacije na profilu
- **Sudionici:** Baza podataka
- **Preduvjet:** Glazbenik je prijavljen u sustav
- **Prioritet:** High
- **Opis osnovnog tijeka:**

1. Glazbenik odabire opciju za uređivanje profila
 2. Glazbenik mijenja podatke
 3. Glazbenik odabirom opcije za spremanje potvrđuje i sprema načinjene promjene
- **Opis mogućih odstupanja:**
 - 3.a Glazbenik nije unio obavezne podatke ili su podatci nevaljani
 1. Glazbenik ponovno unosi obavezne podatke
 2. Glazbenik odustaje od uređivanja profila

UC20 - Promjena liste instrumenata

- **Glavni sudionik:** Administrator
- **Cilj:** Promijeniti listu instrumenata
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijavljen je administrator
- **Prioritet:** Medium
- **Opis osnovnog tijeka:**
 1. Administrator dobiva popis prijedloga instrumenata
 2. Administrator označi koje instrumente želi dodati
 3. Administrator odabire "Dodaj"

UC21 - Pisanje objava

- **Glavni sudionik:** Korisnik
- **Cilj:** Objaviti objavu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Prioritet:** Medium
- **Opis osnovnog tijeka:**
 1. Korisnik u izborniku bira opciju "Post"
 2. Aplikacija prikaže prozor za unos sadržaja objave
 3. Korisnik piše objavu
 4. Korisnik odabire opciju "Objavi"
 5. Aplikacija sprema objavu i prikazuje ju na profilu korisnika

UC22 - Poziv u bend

- **Glavni sudionik:** Voditelj benda
- **Cilj:** Dodati glazbenika u bend

- **Sudionici:** Baza podataka, Glazbenik
- **Preduvjet:** Voditelj benda je prijavljen
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Voditelj benda nalazi glazbenika kojeg želi dodati u bend u listi glazbenika
 2. Voditelj benda odabire opciju "Dodaj glazbenika"
 3. Glazbenik dobiva poziv za učlanjenje u bend

UC23 - Učlanjenje u bend

- **Glavni sudionik:** Glazbenik
- **Cilj:** Ući u bend
- **Sudionici:** Baza podataka
- **Preduvjet:** Glazbenik je prijavljen u sustav
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Glazbenik odabire prikaz obavijesti o pozivu u bend
 2. Aplikacija prikazuje obavijest
 3. Glazbenik odabire opciju "Prihvati" ili "Odbij"

UC24 - Pregled kalendara

- **Glavni sudionik:** Glazbenik
- **Cilj:** Pregledati vlastiti kalendar
- **Sudionici:** Baza podataka
- **Preduvjet:** Glazbenik je prijavljen u sustav
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Glazbenik odabire opciju "Prikaži kalendar"
 2. Aplikacija prikazuje glazbenikov kalendar

UC25 - Dodavanje obaveza u kalendar

- **Glavni sudionik:** Voditelj benda
- **Cilj:** Dodati obaveze u kalendar benda
- **Sudionici:** Baza podataka
- **Preduvjet:** Voditelj benda je prijavljen u sustav
- **Prioritet:** High

- **Opis osnovnog tijeka:**

1. Voditelj benda odabire opciju "Dodaj obavezu" na profilu benda
2. Aplikacija prikazuje polje unos događaja u kalendar
3. Voditelj benda unosi događaje u kalendar

- **Opis mogućih odstupanja:**

- 3.a Voditelj benda nije unio obavezne podatke ili su podatci nevaljani
 1. Voditelj benda ponovno unosi podatke
 2. Voditelj benda odustaje od uređivanja profila

UC26 - Stvaranje benda

- **Glavni sudionik:** Glazbenik

- **Cilj:** Stvoriti bend

- **Sudionici:** Baza podataka

- **Preduvjet:** Glazbenik je prijavljen u sustav

- **Prioritet:** High

- **Opis osnovnog tijeka:**

1. Glazbenik bira opciju "Stvori bend"
2. Aplikacija prikazuje polja za unos podataka
3. Glazbenik u polja za unos unosi naziv benda i žanr
4. Glazbenik bira opciju "Spremi"
5. Aplikacija pohranjuje bend

- **Opis mogućih odstupanja:**

- 4.a Glazbenik nije unio obavezne podatke ili su podatci nevaljani
 1. Glazbenik ponovno unosi podatke
 2. Glazbenik odustaje od uređivanja profila

UC27 - Objavljivanje objava benda

- **Glavni sudionik:** Voditelj benda

- **Cilj:** Objaviti objavu na profilu benda

- **Sudionici:** Baza podataka

- **Preduvjet:** Voditelj benda je prijavljen u sustav

- **Prioritet:** Medium

- **Opis osnovnog tijeka:**

1. Voditelj benda bira opciju "Post"
2. Aplikacija prikazuje prozor za unos sadržaja objave
3. Voditelj benda piše objavu

4. Voditelj bira opciju "Objavi"
5. Aplikacija sprema objavu i prikazuje ju na profilu benda

UC28 - Pregled kalendara članova benda

- **Glavni sudionik:** Voditelj benda
- **Cilj:** Pregledati kalendar bilo kojeg člana benda
- **Sudionici:** Baza podataka
- **Preduvjet:** Voditelj benda prijavljen je u sustav
- **Prioritet:** Medium
- **Opis osnovnog tijeka:**
 1. Voditelj benda odabire opciju za pregled kalendara članova benda
 2. Aplikacija prikazuje kalendare članova

UC29 - Uređivanje profila benda

- **Glavni sudionik:** Voditelj benda
- **Cilj:** Promijeniti informacije na profilu benda
- **Sudionici:** Baza podataka
- **Preduvjet:** Voditelj benda prijavljen je u sustav
- **Prioritet:** Medium
- **Opis osnovnog tijeka:**
 1. Voditelj benda odabire opciju za uređivanje profila benda
 2. Voditelj benda mijenja podatke
 3. Voditelj benda odabirom opcije za spremanje potvrđuje i sprema načinjene promjene
- **Opis mogućih odstupanja:**
 - 3.a Voditelj benda nije unio obavezne podatke ili su podatci nevaljani
 1. Voditelj benda ponovno unosi obavezne podatke
 2. Voditelj benda odustaje od uređivanja profila

UC30 - Pretraga bendova

- **Glavni sudionik:** Organizator
- **Cilj:** Pregledati bendove prema određenom kriteriju
- **Sudionici:** Baza podataka
- **Preduvjet:** Organizator je prijavljen u sustav
- **Prioritet:** High
- **Opis osnovnog tijeka:**

1. Organizator bira opciju "Bendovi"
2. Aplikacija prikazuje popis bendova
3. Organizator može odabrati filtrirati bendove

UC31 - Pregled povijesti benda

- **Glavni sudionik:** Javnost
- **Cilj:** Pregledati povijest benda
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Prioritet:** Low
- **Opis osnovnog tijeka:**
 1. Javnost bira opciju "Prikaži biografiju" na stranici benda
 2. Aplikacija prikazuje biografiju benda

UC32 - Kreirati događaj

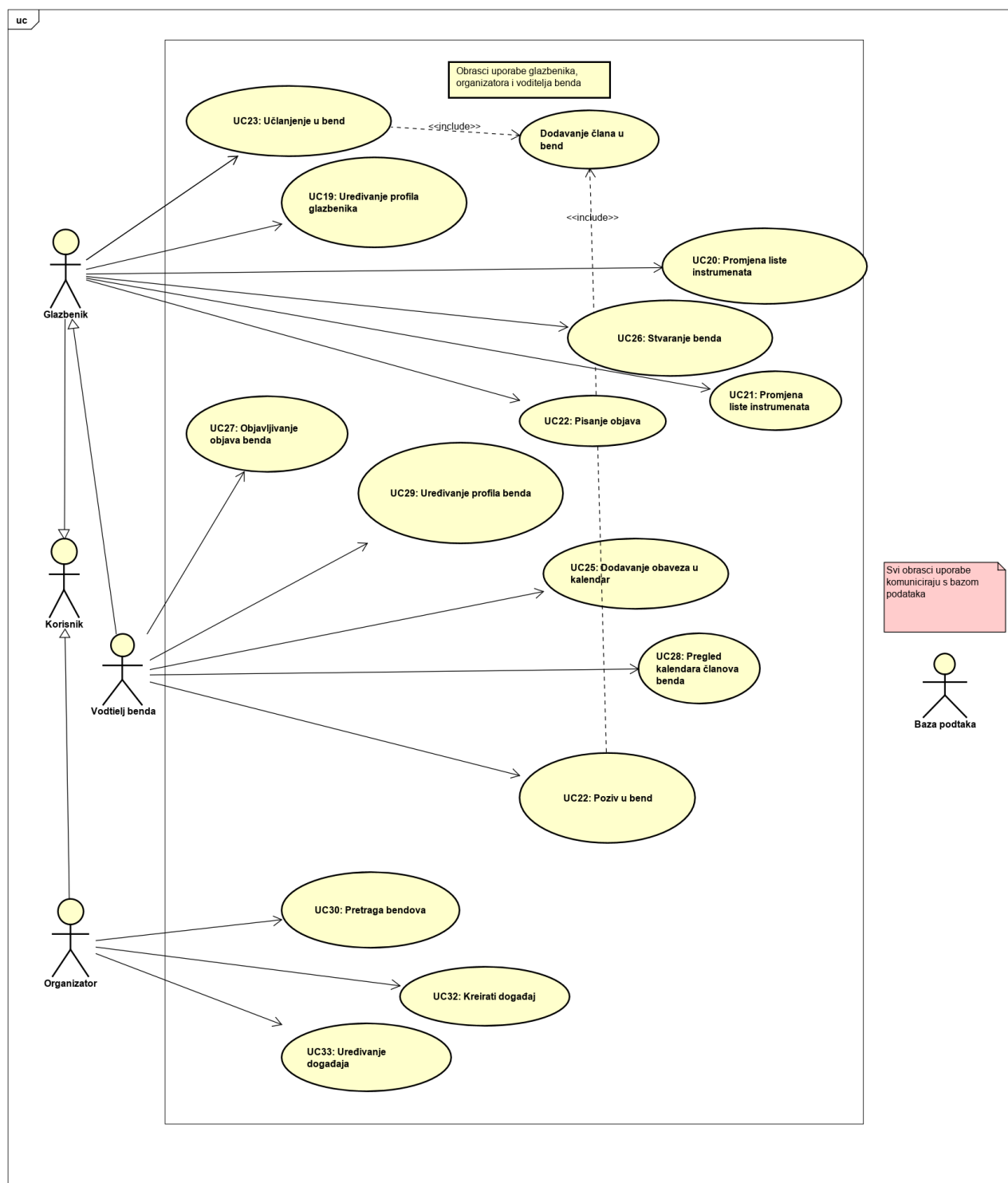
- **Glavni sudionik:** Organizator
- **Cilj:** Kreirati događaj
- **Sudionici:** Baza podataka
- **Preduvjet:** Organizator je prijavljen u sustav
- **Prioritet:** High
- **Opis osnovnog tijeka:**
 1. Organizator bira opciju "Kreiraj događaj"
 2. Aplikacija prikazuje prozor za unos događaja
 3. Organizator unosi potrebne informacije
 4. Organizator bira opciju "Spremi"
 5. Aplikacija sprema događaj u bazu
- **Opis mogućih odstupanja:**
 - 4.a Organizator nije unio obavezne podatke ili su podatci nevaljani
 1. Organizator ponovno unosi obavezne podatke
 2. Organizator odustaje od kreiranja događaja

UC33 - Uređivanje događaja

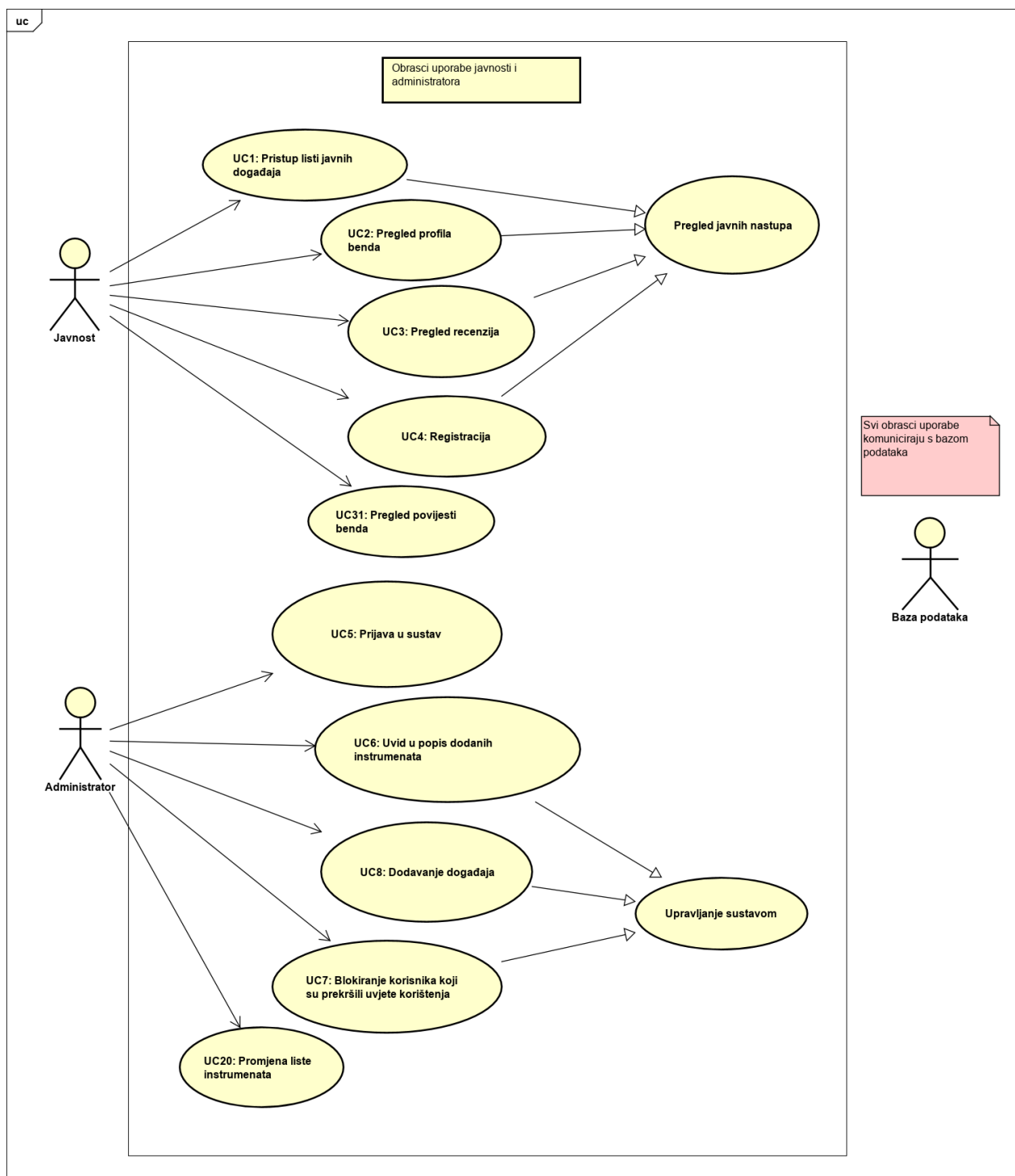
- **Glavni sudionik:** Organizator
- **Cilj:** Urediti informacije o događaju
- **Sudionici:** Baza podataka
- **Preduvjet:** Organizator je prijavljen u sustav

- **Prioritet:** Medium
- **Opis osnovnog tijeka:**
 1. Organizator odabire opciju "Uredi" na stranici događaja
 2. Aplikacija prikazuje prozor za izmjenu događaja
 3. Organizator bira opciju "Spremi"
 4. Aplikacija sprema promjene u bazu
- **Opis mogućih odstupanja:**
 - 3.a Organizator želi unijeti nevaljane podatke
 1. Organizator ponovno unosi podatke
 2. Organizator odustaje od izmjene događaja
 - 3.b Organizator želi promijeniti podatke za koje nije dozvoljena izmjena
 1. Organizator ponovno unosi podatke
 2. Organizator odustaje od izmjene događaja

Dijagrami obrazaca uporabe

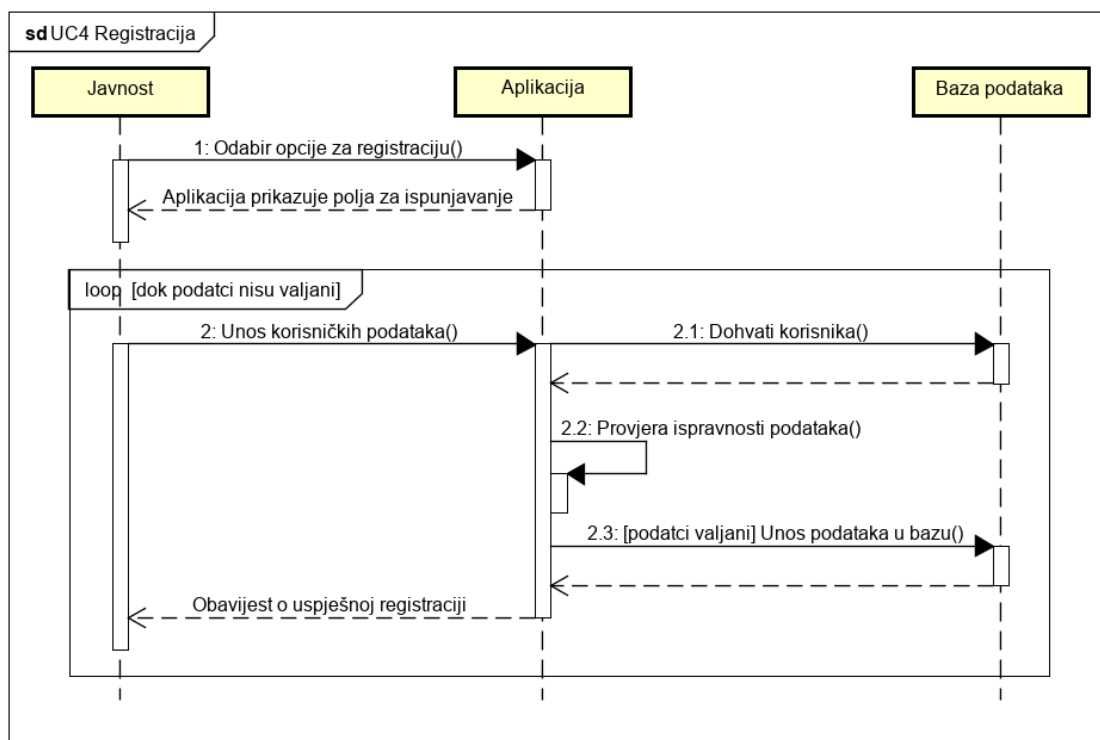


Slika 3.1: Obrasci uporabe za korisnika i organizatora



Slika 3.2: Obrasci uporabe za javnost i administratora

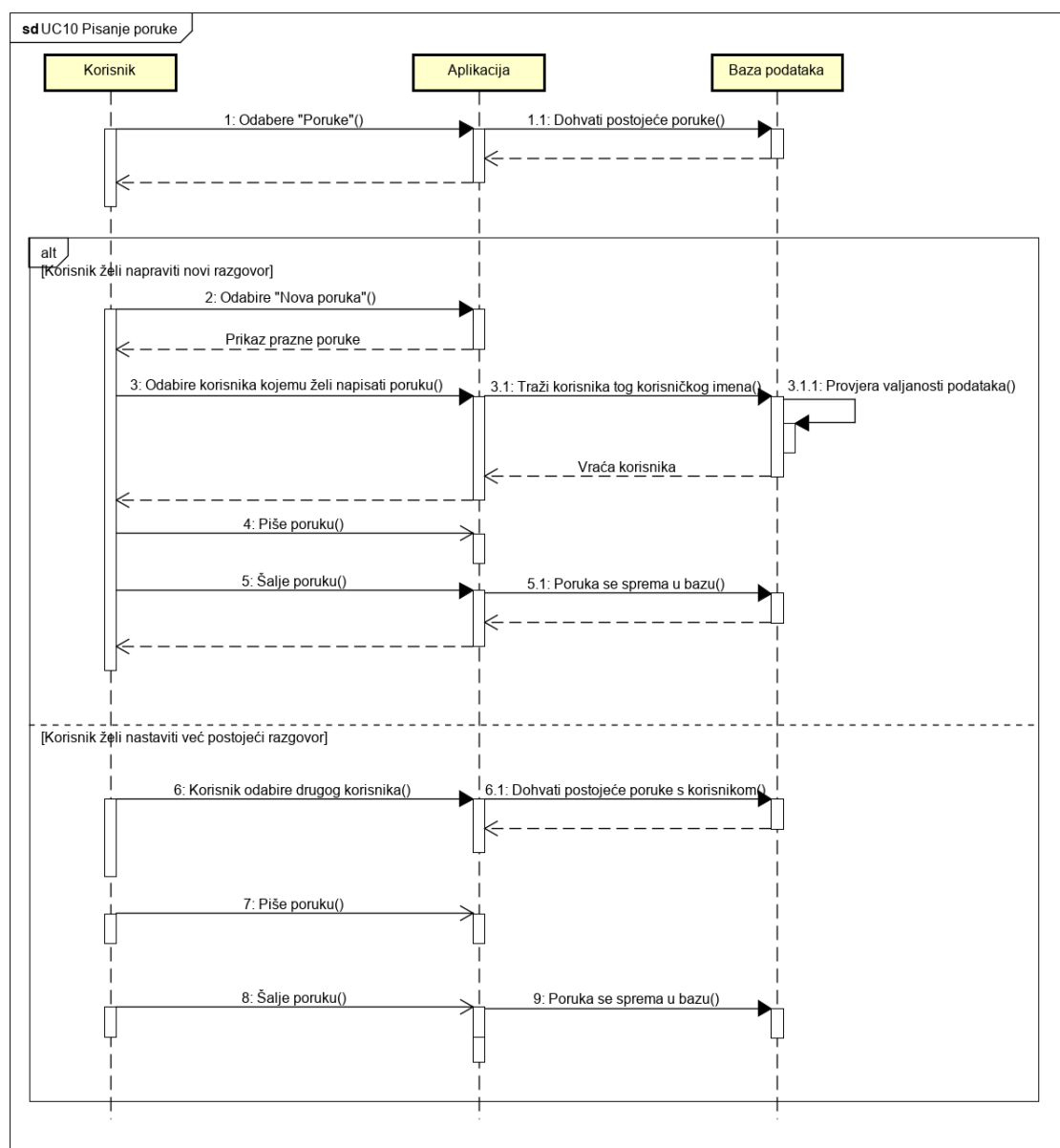
nike iz baze nakon čega provjerava postoji li korisnik istog korisničkog imena ili emaila. Vrš se provjera ispravnosti podataka te ukoliko su podatci valjani, unose se u bazu. Nakon unosa podataka u bazu, korisniku se šalje obavijest o uspješnoj registraciji.



Slika 3.4: Sekvencijski dijagram za UC4

Obrazac uporabe UC10 - Pisanje poruke

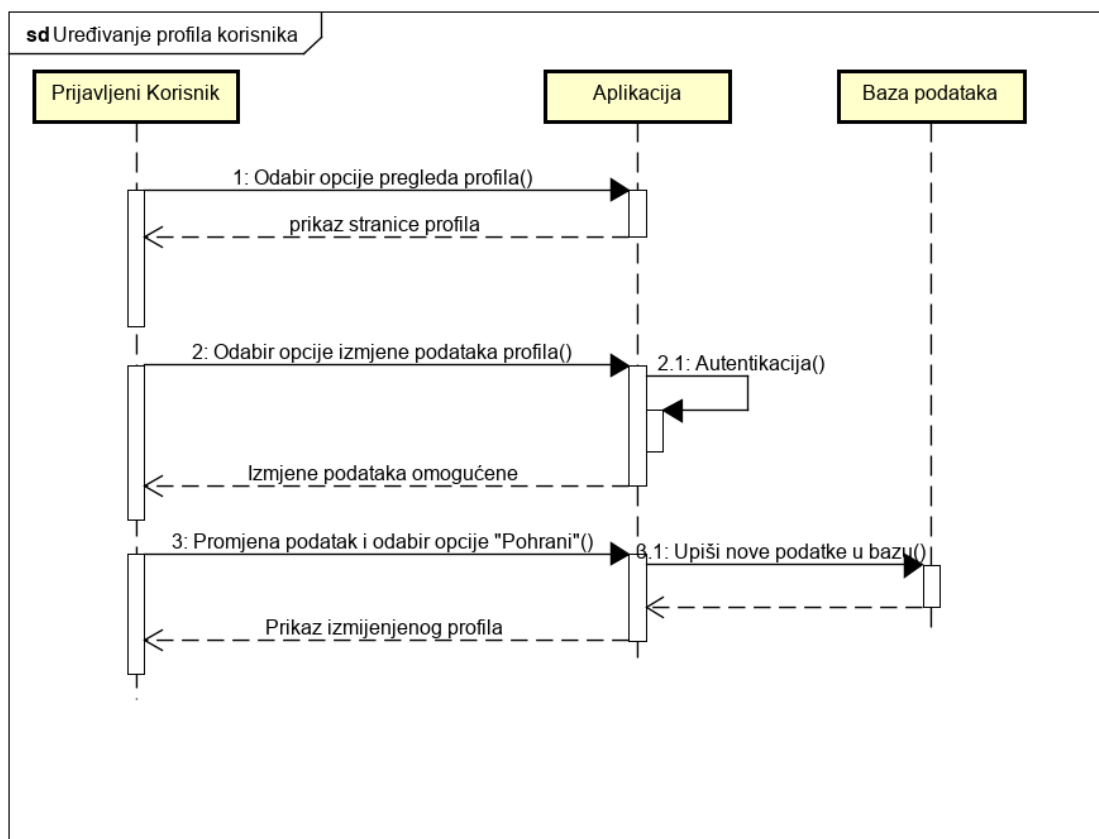
Korisnik odabire "Poruke" nakon čega se iz baze dohvaćaju postojeće poruke korisnika. Korisnik može ili napraviti novi razgovor, ili odabrati postojeći. Ukoliko želi napraviti novi razgovor, odabire opciju "Nova poruka" te mu se prikaže prazna poruka. Nakon toga, odabire korisnika kojemu želi napisati poruku, a taj korisnik se traži u bazi podataka. Ukoliko korisnik želi razgovarati s korisnikom s kojim je već komunicirao, odabire razgovor s tim korisnikom. Aplikacija dohvaća iz baze podataka prošle poruke s odabranim korisnikom. U oba opisana slučaja, korisnik piše željenu poruku te ju šalje nakon čega se poruka sprema u bazu.



Slika 3.5: Sekvencijski dijagram za UC10

Obrazac uporabe UC15 - Uređivanje profila korisnika

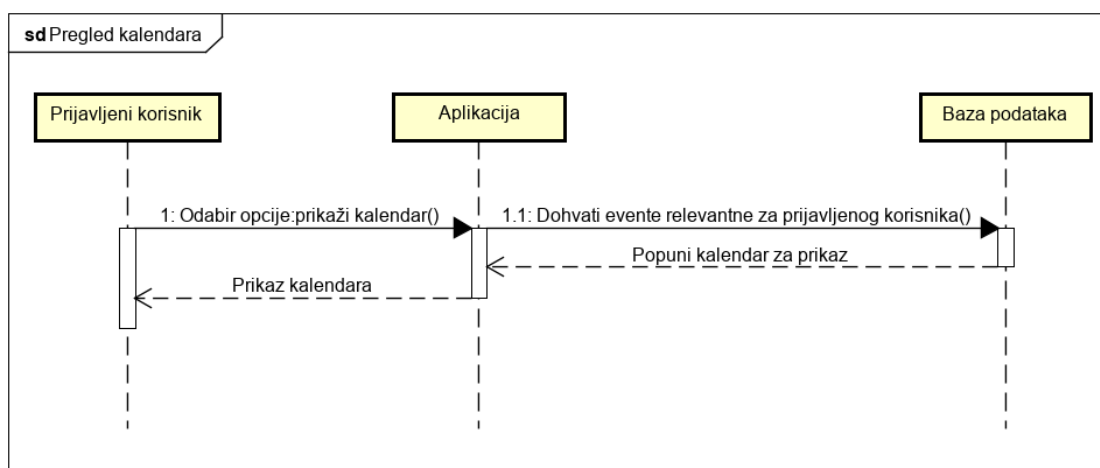
Prijavljeni korisnik ide na stranicu svog profila. Korisnik potom bira opciju izmjene podataka na svom profilu, nakon čega slijedi proces autentikacije korisnika. Ukoliko je autentikacija prošla, omogućene su izmjene podataka korisniku. Korisnik po želji mijenja podatke nakon čega odabire opciju "Izmijeni". Promijenjeni podatci se zapisuju u bazu, a korisniku se prikazuje izmijenjena stranica profila.



Slika 3.6: Sekvencijski dijagram za UC15

Obrazac uporabe UC24 - Pregled kalendara

Prijavljeni korisnik bira opciju prikaži kalendar, nakon čega aplikacija uzima iz baze podataka sve događaje koji su relevantni za korisnika (one na kojima je bio/će biti ili je svirao/će svirati). Aplikacija puni kalendar tim događajima i prikazuje ih korisniku.



Slika 3.7: Sekvencijski dijagram za UC24

3.2 Ostali zahtjevi

- Sustav treba omogućiti rad više korisnika u isto vrijeme
- Neispravno korištenje sučelja ne smije na bilo koji način promijeniti ili obustaviti rad sustava
- Sustav mora biti jednostavan za uporabu, tj. mora biti intuitivan
- Aplikacija mora podržavati hrvatske dijakritičke znakove
- Sustav mora biti u stanju brzo obraditi dobivene podatke kako korisnik ne bi dugo čekao promjenu
- Sustav treba biti implementiran kao web aplikacija, s tim da bi joj većina korisnika pristupala preko mobilnih uređaja
- Web aplikacija treba biti implementirana koristeći objektno-orijentirane jezike

- Treba osigurati sigurnost aplikacije, tj. lozinke moraju biti enkriptirane te mora biti osigurana sigurnost veze između korisnika aplikacije i baze podataka

4. Arhitektura i dizajn sustava

Da bi dugoročno uštedjeli vrijeme, uložili smo dio vremena na konfiguriranje CI (engl. continuous integration) i CD (engl. continuous delivery) procesa. Za isporuku aplikacije odabrali smo Heroku.

Heroku je jedna od najpoznatijih platformi za isporučivanje aplikacija koja se ističe svojom jednostavnošću. Za razliku od AWS-ovih servisa, koje smo također razmatrali, Heroku se sam brine oko instanci i arhitekture sustava na kojem se izvodi naša aplikacija. Zbog ograničenih resursa odlučili smo isporučiti aplikaciju na besplatnu instancu Heroku-a.

Besplatna instanca Heroku-a ima određena ograničenja, a najupečatljivije od njih je način na koji se pokreće projekt. Heroku sam prepoznaje kojeg je tipa projekt pa da pokrenemo frontend i backend trebamo dvije instance. CD je integriran putem GitLab-ovih pipelinesa za koje je bilo potrebno napisati `.gitlab-ci.yml` datoteku u kojoj smo konfigurirali GitLab-ov pipeline. GitLab-ov pipeline konfiguriran je tako da se na svaki commit u dev grani izgradi aplikacija, pokrenu i uspješno završe testovi te krene isporuka aplikacije na Heroku.

Da bi osigurali maksimalno vrijeme dostupnosti naše platforme, `.yml` datoteka također je konfigurirana tako da prilikom commita u master isporuči aplikaciju na druge dvije instance. Ukupno imamo četiri pokrenute instance Heroku-a, od koje su dvije backend, a dvije frontend. Backend i frontend imaju svaki svoju razvojnu i produkcijsku instancu. Poveznice na instance:

- <http://giger-fer.herokuapp.com/>
- <https://giger-fer-dev.herokuapp.com/>
- <https://giger-backend-dev.herokuapp.com/>
- <http://giger-backend.herokuapp.com/>

U skladu s time, Spring Boot aplikacija ima dvije `.properties` datoteke. Jedna od njih je namijenjena lokalnom izvođenju aplikacije te sadrži postavke lokalne PostgreSQL baze, dok je druga konfigurirana tako da postavke čita iz varijabli

okruženja. Varijable okruženja postavljene su na Heroku tako da čak niti pristupom u git repozitorij vanjski korisnik ne može doći do akreditacije (credentials) kojima bi mogao pristupiti bazi.

Prednost ostvarena automatiziranjem procesa isporuke jest povećanje vjerojatnosti uspješnosti iste te povećanje udjela dostupnosti aplikacije zbog dva para instanci servera.

Uvidjevši prednosti korištenja CD-a, primijenili smo to znanje i na prevođenje dokumentacije. Unutar repozitorija, osim pipeline-a za isporuku aplikacije, postoji pipeline za automatsko prevođenje dokumentacije čiji je rezultat .pdf dokument. Potencijalni prostor za napredak bio bi korištenje Docker tehnologije tako da prilikom pokretanja aplikacije korisnik ne mora imati instaliranu PostgreSQL bazu već ju pokrene u Docker kontejneru.

Od mogućih arhitektura sustava, za svoj projekt smo odabrali objektno usmjerenu arhitekturu. Tu arhitekturu smo odabrali zato što se koristi u industriji te je de facto standard razvoja složenih programskih rješenja. Osim toga, ona je fleksibilna, omogućuje recikliranje koda te logički razdjeljuje sustav na više cjelina, što je bitno s obzirom da više ljudi radi na implementaciji aplikacije. Zahvaljujući modularnosti programskog rješenja, greške su lako ispravljive, a nove mogućnosti dodaju se bez poteškoća.

Odlučili smo se za web aplikaciju, koja je prilagođena mobilnim uređajima, obzirom da glazbenici, a time i bendovi nemaju uvijek pristup računalu, a ne želimo da je korisnik ograničen samo na mobilne uređaje.

Arhitekturu sustava možemo podijeliti na četiri podsustava:

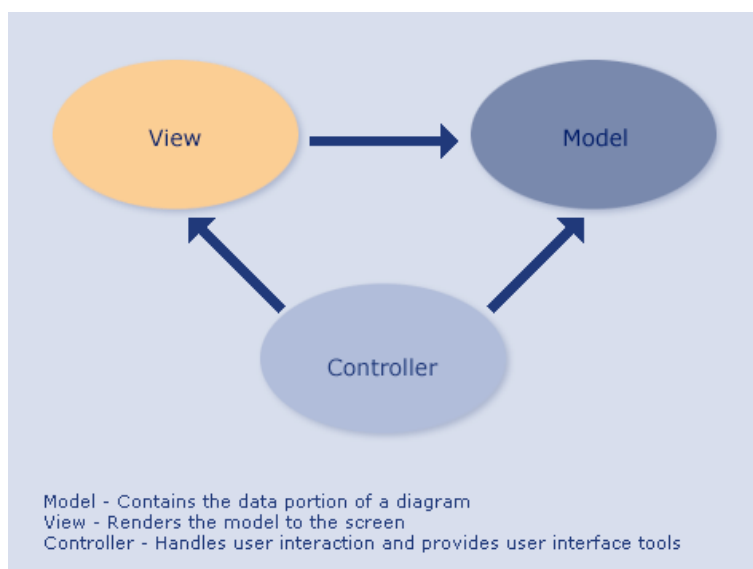
- Web preglednik
- Web poslužitelj
- Web aplikacija
- Baza podataka

Korisnik (javnost, glazbenik, bend, administrator) pristupa web aplikaciji uz pomoć svog web preglednika, s time da se u sredini nalazi web poslužitelj. Na njemu se nalazi aplikacija koju on pokreće, te uz pomoć protokola komunicira s korisnicima.

Klijentski (frontend) dio aplikacije omogućuje da korisnik korištenjem sučelja može pristupiti serveru (backend) aplikacije. Ovisno o tome što korisnik hoće,

taj server ima mogućnost spajanja na bazu podataka kako bi korisniku prikazao informacije.

Backend je napisan u Javi 11, a kao razvojni okvir koristimo Java Spring Boot 2.2.0. Dodani su projekti Spring Data JPA kako bi backend mogao lako komunicirati s bazom, Spring Web MVC za rukovanje zahtjevima te Spring Security kako bi zaštitili aplikaciju od vanjskih napada. Za pregledniji kod koristimo Lombok.



Slika 4.1: Pojednostavljeni prikaz MVC-a

Za frontend koristimo React. On je moderan i jednostavan framework koji koristi HTML, CSS, JSX i JavaScript uz pomoć kojeg smo napravili sučelje za našu aplikaciju. Uz pomoć React-a možemo lagano komunicirati s backendom koristeći REST.

4.1 Baza podataka

Za potrebe razvoja *Gigera* koristit će se objektno relacijsko mapiranje. To je metoda koja se koristi u objektno-orijentiranim jezicima te se na taj način stvara virtualna objektna baza podataka. Za implementaciju baze podataka odabrali smo PostgreSQL, zbog generalno pozitivnog iskustva u korištenju te implementacije baze podataka u dosadašnjem fakultetskom obrazovanju. Bitno je naglasiti da na osobnim računalima u svrhu razvijanja aplikacije koristimo istu implementaciju baze kao i na web poslužitelju kako bi minimizirali neočekivano ponašanje.

Baza podataka sastoji se od sljedećih tablica:

- Comment
- Message
- Conversation
- Conversation_user
- System_person
- Person
- System_person_roles
- Organizer
- Band
- Gig_type
- Band_occasions
- Occasion
- Musician_occasions
- Band_invited_back_up_members
- Band_invited
- Band_back_up_members
- Musician_bands
- Post
- Musician
- Instruments
- Instrument
- Musician_gig_history
- Gig
- Review_gig
- Review

4.1.1 Opis tablica

Comment Ovaj entitet sadrži jedan komentar. Sadrži attribute: id komentara, id autora, sadržaj te vrijeme objavljivanja. Ovaj entitet je u *Many-to-One* vezi s entitetima: Post i Person.

Comment		
id	BIGINT	jedinstveni identifikator komentara
content	VARCHAR	sadržaj komentara

posted_on	TIMESTAMP	datum i vrijeme objave komentara
author_id	BIGINT	jedinstveni identifikator autora komentara
fk_post	BIGINT	jedinstveni identifikator komentara

Message Ovaj entitet sadrži informacije o poruci. Sadrži attribute: id poruke, sadržaj poruke, vrijeme kada je poruka poslana, id pošiljatelja i id razgovora. Ovaj entitet je u *Many-to-One* vezi s entitetima: Conversation, Person, Band. Ako je bend poslao poruku, tada je fk_sender null, a ako ju je poslao korisnik tada je fk_sender_band null.

Message		
id	BIGINT	jedinstveni identifikator poruke
content	VARCHAR	sadržaj poruke
sent_time	TIMESTAMP	vrijeme kada je poruka poslana
fk_sender	BIGINT	jedinstveni identifikator pošiljatelja
fk_sender_band	BIGINT	jedinstveni identifikator benda pošiljatelja
fk_conversation	BIGINT	jedinstveni identifikator razgovora

Conversation Ovaj entitet sadrži informacije o razgovoru. Sadrži attribute: id razgovora i ime razgovora. Ovaj entitet je u *One-to-Many* vezi s entitetima: Message i *Many-to-One* vezi s entitetima: Band i Person.

Conversation		
id	BIGINT	jedinstveni identifikator razgovora
fk_band	BIGINT	jedinstveni identifikator benda
picture_url	VARCHAR	url slike razgovora
title	VARCHAR	naziv razgovora

Conversation_user Ova vezna tablica sadrži informacije o sudjelovanju korisniku u razgovoru. Sadrži attribute: id korisnika i id razgovora.

Conversation_user		
fk_user	BIGINT	jedinstveni identifikator korisnika
fk_conversation	BIGINT	jedinstveni identifikator razgovora

System.Person Ovaj entitet sadrži podatke korisnika potrebne sustavu za sis-

temsku logiku. Sadrži attribute: id korisnika, email, locked i verified zastavice te šifriranu lozinku. Ovaj entitet je u *One-to-Many* vezi s entitetom: Roles i *One-to-One* vezi s entitetima: Person, Musician i Organizer.

System_person		
id	BIGINT	jedinstveni identifikator sustavskih podataka o korisniku
email	VARCHAR	email adresa osobe
locked	BOOLEAN	korisnik ima zabranu korištenja aplikacije ili ne
password_hash	VARCHAR	hash lozinke osobe
verified	BOOLEAN	email adresa potvrđena ili ne

Person Ovaj entitet sadrži podatke korisnika potrebne za poslovne svrhe. Sadrži attribute: telefonski broj, url slike korisnika, korisničko ime kojim se predstavlja javnosti. Ovaj entitet je u *One-to-Many* vezi s entitetima: Conversation, Message i *One-to-One* vezi s entitetima: Musician, System_person i Organizer.

Person		
id	BIGINT	jedinstveni identifikator korisnika
phone_number	VARCHAR	telefonski broj korisnika
picture_url	VARCHAR	url slike korisnika
username	VARCHAR	korisničko ime korisnika

System_person_roles Ovo je vezna tablica koja sadrži n-torke iz kojih možemo iščitati dodijeljene uloge pojedinim korisnicima. Sadrži attribute: system_person_id i cijeli broj uloge koji predstavlja enumeraciju.

System_person_roles		
system_person_id	BIGINT	jedinstveni identifikator sustavskih podataka o korisniku
roles	INT	uloga korisnika

Organizer Ovaj entitet sadrži informacije o organizatoru. Sadrži attribute: id organizatora te ime organizatora. Ovaj entitet je u *One-to-One* vezi s entitetima: Musician, System_person i Organizer te u *One-to-Many* vezi s entitetom Gig.

Organizer		
id	BIGINT	jedinstveni identifikator organizatora
manager_name	VARCHAR	ime organizatora

Band Ovaj entitet sadrži podatke o kreiranom bendu. Sadrži attribute: id, opis, datum formiranja, adresu sjedišta, dodatni opis adrese, par koordinata, maksimalnu udaljenost, naziv benda, url slike benda i id voditelja benda. Ovaj entitet je u *Many-to-Many* vezi s Musician za potrebe liste članova, u *Many-to-One* vezi s Musician za potrebu evidencije voditelja benda te *One-to-Many* vezi s entitetima: Message, Conversation, Gig i Post.

Band		
id	BIGINT	jedinstveni identifikator benda
bio	VARCHAR	opis benda
formed_date	DATE	datum osnutka benda
address	VARCHAR	adresa benda
extra_description	VARCHAR	dodatak opis benda
x	DOUBLE	x koordinata lokacije
y	DOUBLE	y koordinata lokacije
max_distance	DOUBLE	najveća udaljenost koju bend želi prijeći zbog gaže
name	VARCHAR	ime benda
picture_url	VARCHAR	url slike benda
leader_id	BIGINT	jedinstveni identifikator voditelja benda

Gig_type Ovo je vezna tablica iz koje se mogu iščitati sve vrste nastupa koje izvodi određeni bend. Sadrži attribute: band_id i naziv tipa nastupa.

Gig_type		
band_id	BIGINT	jedinstveni identifikator benda
gig_type	VARCHAR	vrsta nastupa

Band_occasions Ovo je vezna tablica iz koje se može iščitati zauzetost benda. Sadrži attribute: identifikator benda i identifikator termina.

Band_occasions		
occasion_id	BIGINT	jedinstveni identifikator događaja
band_id	BIGINT	jedinstveni identifikator benda koji sudjeluje na događaju

Occasion Ovaj entitet sadrži podatke o događaju. Sadrži attribute: id događaja, opis, datum, zastavicu privatnosti. Ovaj entitet je u *Many-to-One* vezi s entitetima: Musician i Band.

Occasion		
id	BIGINT	jedinstveni identifikator događaja
description	VARCHAR	opis događaja
local_date	DATE	datum održavanja događaja
personal_occasion	BOOLEAN	privatan događaj ili ne

Musician_occasions Ovo je vezna tablica iz koje se može iščitati zauzetost glazbenika. Sadrži attribute: identifikator glazbenika i identifikator termina.

Musician_occasions		
musician_id	BIGINT	jedinstveni identifikator glazbenika
occasions_id	BIGINT	jedinstveni identifikator termina

Band_invited_back_up_members Ovo je vezna tablica iz koje se mogu iščitati poslane i neodgovorene pozivnice za pričuvnog člana benda. Sadrži attribute: identifikator glazbenika i identifikator benda.

Band_invited_back_up_members		
band_id	BIGINT	jedinstveni identifikator benda
invited_back_up_members_id	BIGINT	jedinstveni identifikator glazbenika pozvanih u bend kao pričuvni član

Band_invited Ovo je vezna tablica iz koje se mogu iščitati poslane i neodgovorene pozivnice za člana benda. Sadrži attribute: identifikator glazbenika i identifikator benda.

Band_invited		
band_id	BIGINT	jedinstveni identifikator benda
invited_id	BIGINT	jedinstveni identifikator glazbenika pozvanih u bend

Band_back_up_members Ovo je vezna tablica iz koje se mogu iščitati pričuvni članovi bendova. Sadrži attribute: identifikator glazbenika i identifikator benda.

Band_back_up_members		
band_id	BIGINT	jedinstveni identifikator benda
invited_back_up_members_id	BIGINT	jedinstveni identifikator glazbenika koji su pričuvni članovi

Musician_bands Ovo je vezna tablica iz koje se mogu iščitati članovi bendova. Sadrži attribute: identifikator glazbenika i identifikator benda.

Musician_bands		
fk_musician	BIGINT	jedinstveni identifikator glazbenika
fk_band	BIGINT	jedinstveni identifikator benda

Post Ovaj entitet sadrži podatke o objavi. Sadrži attribute: id objave, sadržaj, datum i vrijeme objave, identifikator korisnika ili benda (autora). Ovaj entitet je u *One-to-Many* vezi s entitetima: Comment te u *Many-to-one* s entitetima: Person, Band.

Post		
id	BIGINT	jedinstveni identifikator objave
content	VARCHAR	sadržaj objave
published_on	TIMESTAMP	datum i vrijeme objave
<i>fk_band</i>	BIGINT	jedinstveni identifikator benda
<i>fk_user</i>	BIGINT	jedinstveni identifikator korisnika koji je napisao objavu

Musician Ovaj entitet sadrži informacije o glazbeniku. Sadrži attribute: id glazbenika, oznaku za privatni kalendar te opis glazbenika. Ovaj entitet je u *One-to-One* vezi s entitetima: System_person, Organizer i Person, u *Many-to-Many* vezi s entitetima: Instrument i Band te u *One-to-Many* vezi s entitetima: Band (u potrebe

pohranjivanja voditelja benda) i Occasions.

Musician		
id	BIGINT	jedinstveni identifikator glazbenika
bio	VARCHAR	opis glazbenika
public_calendar	BOOLEAN	kalendar glazbenika javan ili ne

Instruments Ovo je vezna tablica iz koje se može iščitati koje instrumente svira pojedini glazbenik. Sadrži attribute: id instrumenta te id glazbenika.

Instruments		
instruments_id	BIGINT	jedinstveni identifikator instrumenta
musician	BIGINT	jedinstveni identifikator glazbenika

Instrument Ovaj entitet sadrži informacije o instrumentima. Sadrži attribute: id instrumenta, ime instrumenta te vrstu instrumenta. Ovaj entitet je u *One-to-Many* vezi s entitetom Musician.

Instrument		
id	BIGINT	jedinstveni identifikator instrumenta
name	VARCHAR	ime instrumenta
type	INT	vrsta instrumenta

Musician_gig_history Ovo je vezna tablica iz koje se može iščitati povijest nastupa glazbenika. Sadrži attribute: identifikator glazbenika i identifikator nastupa.

Musician_gig_history		
fk_musician	BIGINT	jedinstveni identifikator glazbenika
fk_gig	BIGINT	jedinstveni identifikator nastupa

Gig Ovaj entitet sadrži informacije o nastupima. Sadrži attribute: id nastupa, datum i vrijeme održavanja nastupa, opis nastupa, očekivano trajanje nastupa, oznaku za postignut dogovor, vrstu nastupa, adresa održavanja nastupa, dodatan opis nastupa, x koordinata lokacije, y koordinata lokacije, oznaku za privatni nastup, preporučenu cijenu ulaznice, id organizatora te id benda. Ovaj entitet je u *Many-to-One* vezi s entitetima Organizer i Band te *One-to-Many* vezi s entitetom Review i u *Many-to-Many* vezi s entitetom Musician.

Gig		
id	BIGINT	jedinstveni identifikator nastupa
date_time	TIMESTAMP	datum i vrijeme održavanja nastupa
description	VARCHAR	opis nastupa
expected_duration	VARCHAR	očekivano trajanje nastupa
final_deal_achieved	BOOLEAN	dogovor postignut ili ne
gig_type	INT	vrsta nastupa
address	VARCHAR	adresa održavanja nastupa
extra_description	VARCHAR	dodatan opis nastupa
x	DOUBLE	x koordinata lokacije
y	DOUBLE	y koordinata lokacije
private_gig	BOOLEAN	nastupa privatno ili ne
proposed_price	INT	preporučena cijena ulaznice
organizer_id	BIGINT	jedinstveni identifikator organizatora
band_id	BIGINT	jedinstveni identifikator benda

Review_gig Ovo je vezna tablica iz koje pronalazimo pripadnost recenzije pojedinom nastupu. Sadrži attribute: id nastupa i id recenzije.

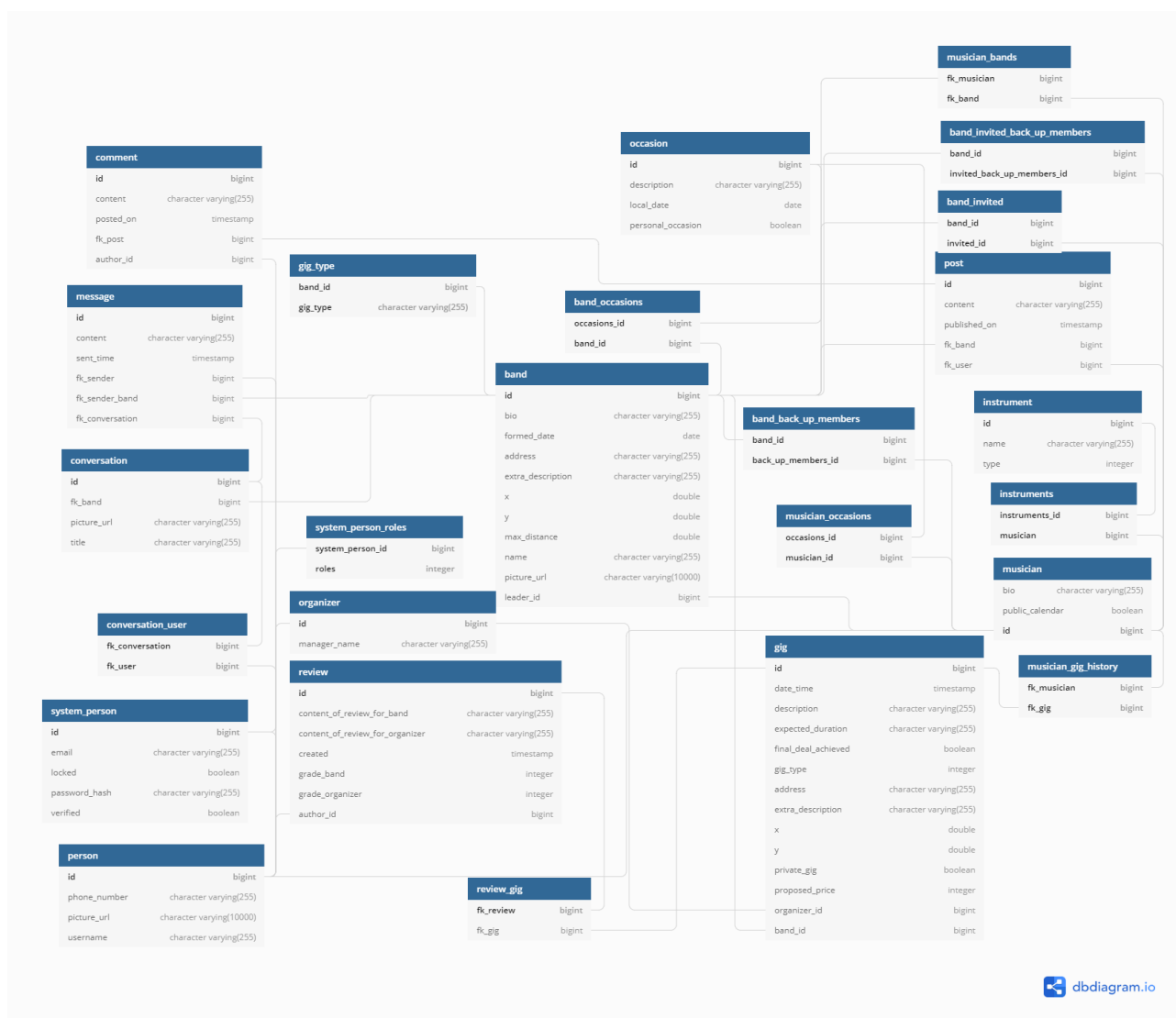
Review_gig		
fk_gig	BIGINT	jedinstveni identifikator nastupa
fk_review	BIGINT	jedinstveni identifikator recenzije

Review Ovaj entitet sadrži informacije za recenziju. Sadrži attribute: id recenzije, sadržaj recenzije benda, sadržaj recenzije organizatora, vrijeme objave recenzije, ocjenu benda, ocjenu organizatora te id autora. Ovaj entitet je u *Many-to-One* vezi s entitetima: Musician i Organizer.

Review		
id	BIGINT	jedinstveni identifikator recenzije
content_of_review_for_band	VARCHAR	sadržaj komentara benda
content_of_review_for_organizer	VARCHAR	sadržaj komentara organizatora
created	TIMESTAMP	vrijeme objave komentara
grade_band	INT	ocjena benda

grade_organizer	INT	ocjena organizatora
author_id	BIGINT	jedinstveni identifikator korisnika koji je autor recenzije

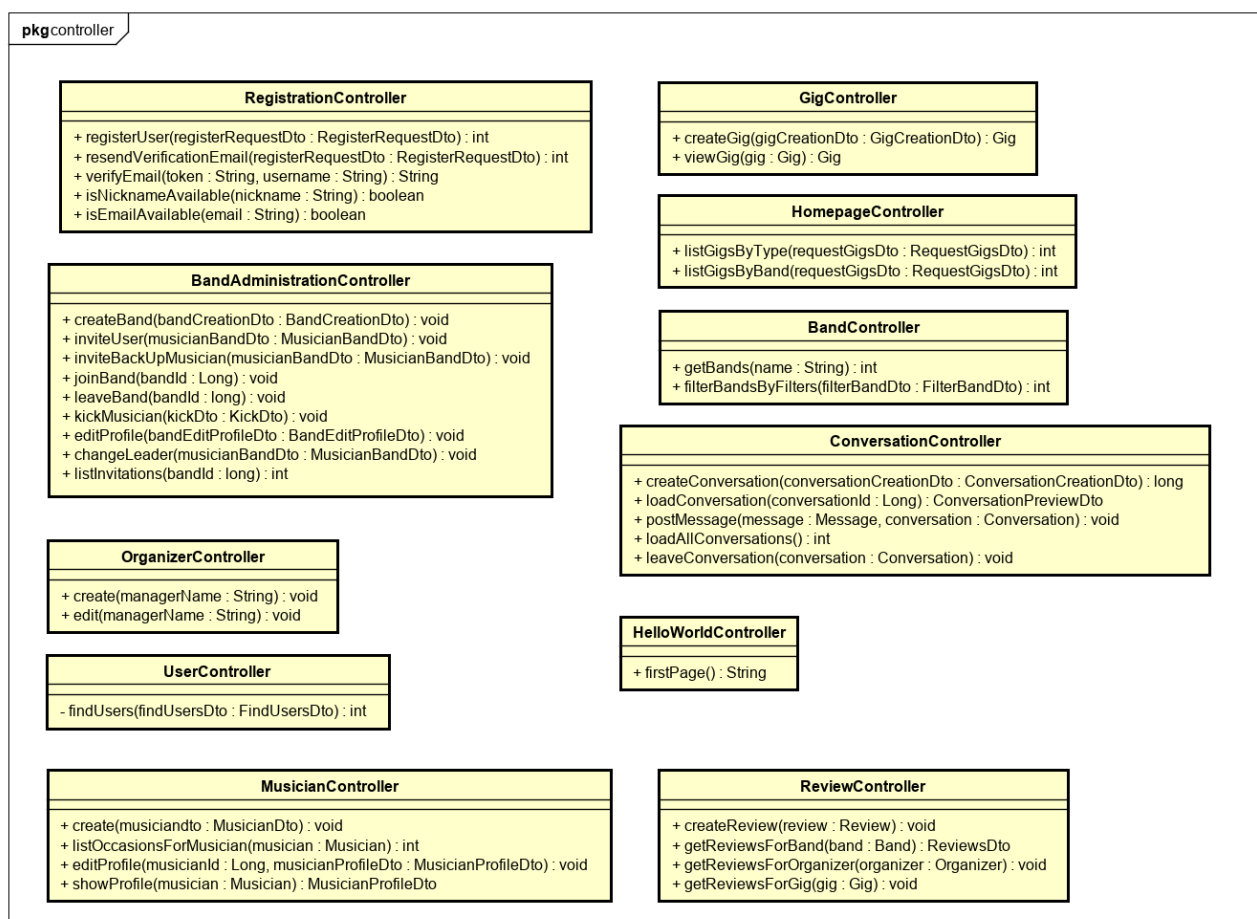
4.1.2 Dijagram baze podataka



Slika 4.2: Dijagram baze podataka

4.2 Dijagram razreda

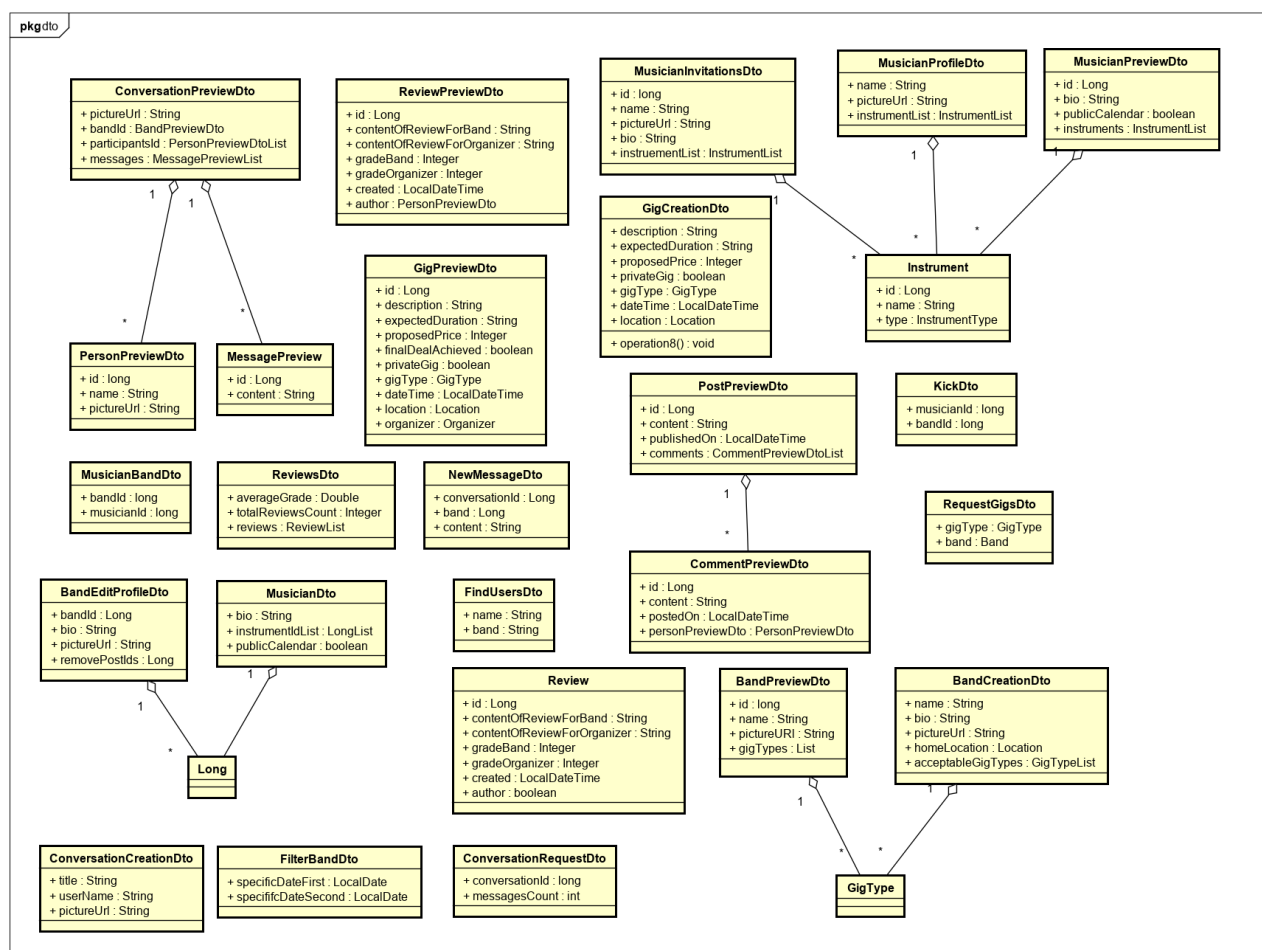
Na slici 4.3 prikazan je Controllers dio backend aplikacije. Controlleri su jedina izložena točka u aplikaciji te nad njima frontend izvršava upite. Svi Controller-i su zaštićeni Spring Security-jem te se prije svakog propuštanja zahtjeva na Controller autorizira token koji se nalazi u zaglavlju zahtjeva. Jedina iznimka su Controller-i koji služe za registraciju i prijavu. Nakon što se zahtjev autorizira Controller-i pozivaju servisni sloj aplikacije te od njih zahtjevaju da izvrše dio poslovne logike za koju su napisani. Povratni tip Controller-a su DTO-ovi (Data Transfer Objects) prikazani na slici 4.4. Njima se na frontend vraća samo dio informacije prikupljene od servisa.



Slika 4.3: Dijagram razreda - dio Controllers

Slika 4.4 prikazuje DTO-ove kojima backend dio aplikacije komunicira s frontendom. DTO-ove smo modelirali tako da izbjegnemo kružne reference objekata

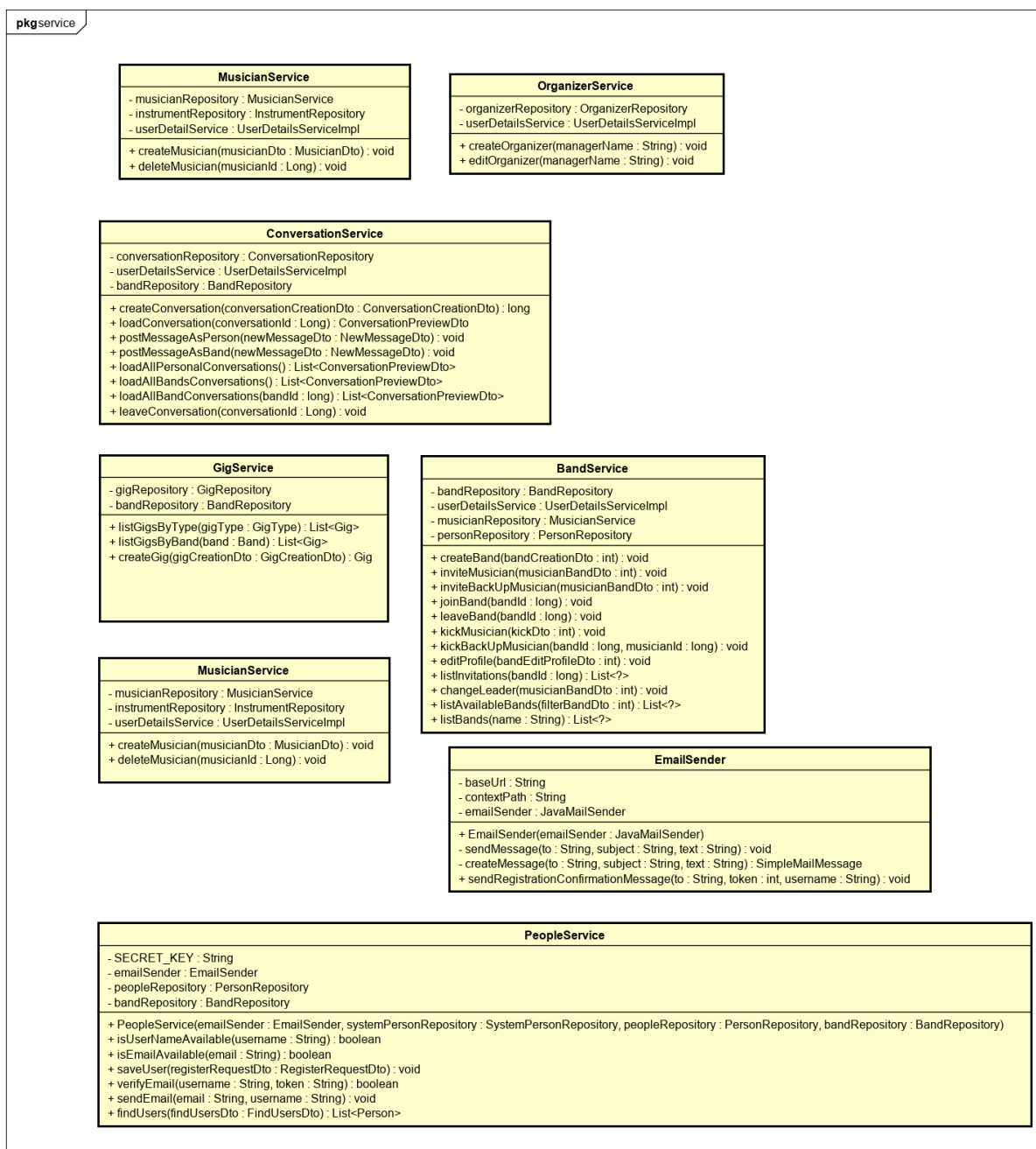
koje dobijemo iz baze podataka. Kao posljedica, DTO-ovi sadrže uglavnom primitivne tipove ili neke druge DTO-ove (npr. ConversationPreviewDTO sadrži listu PersonPreviewDTO koji predstavljaju sudionike razgovora). DTO-ove koristimo u oba smjera komunikacije backenda i frontenda.



Slika 4.4: Dijagram razreda - dio Data transfer objects

Slika 4.5 prikazuje dijagram razreda servisnog sloja. Servisi komuniciraju s repozitorijima koji pristupaju bazi i Controller-ima od kojih dobivaju i kojima vraćaju podatke. Servisi iz baze dobivaju instance objekata koji mogu biti povezani s drugim podacima, itd. i njihov je cilj poštivajući poslovnu logiku obraditi te podatke i kao rezultat svog izvođenja vraćaju DTO-ove. Servisi sadrže svu poslovnu logiku. Gotovo svaki Controller ima pripadajući servis, a svaki je servis logički objedinjen skup funkcija poslovne logike. Iznimke koje se bacaju u servisima omataju se GigerException-om koji nasljeđuje RuntimeException. Ako se pogreška propagira iz sustava, možemo provjeriti njezin tip te ako je zamotana u

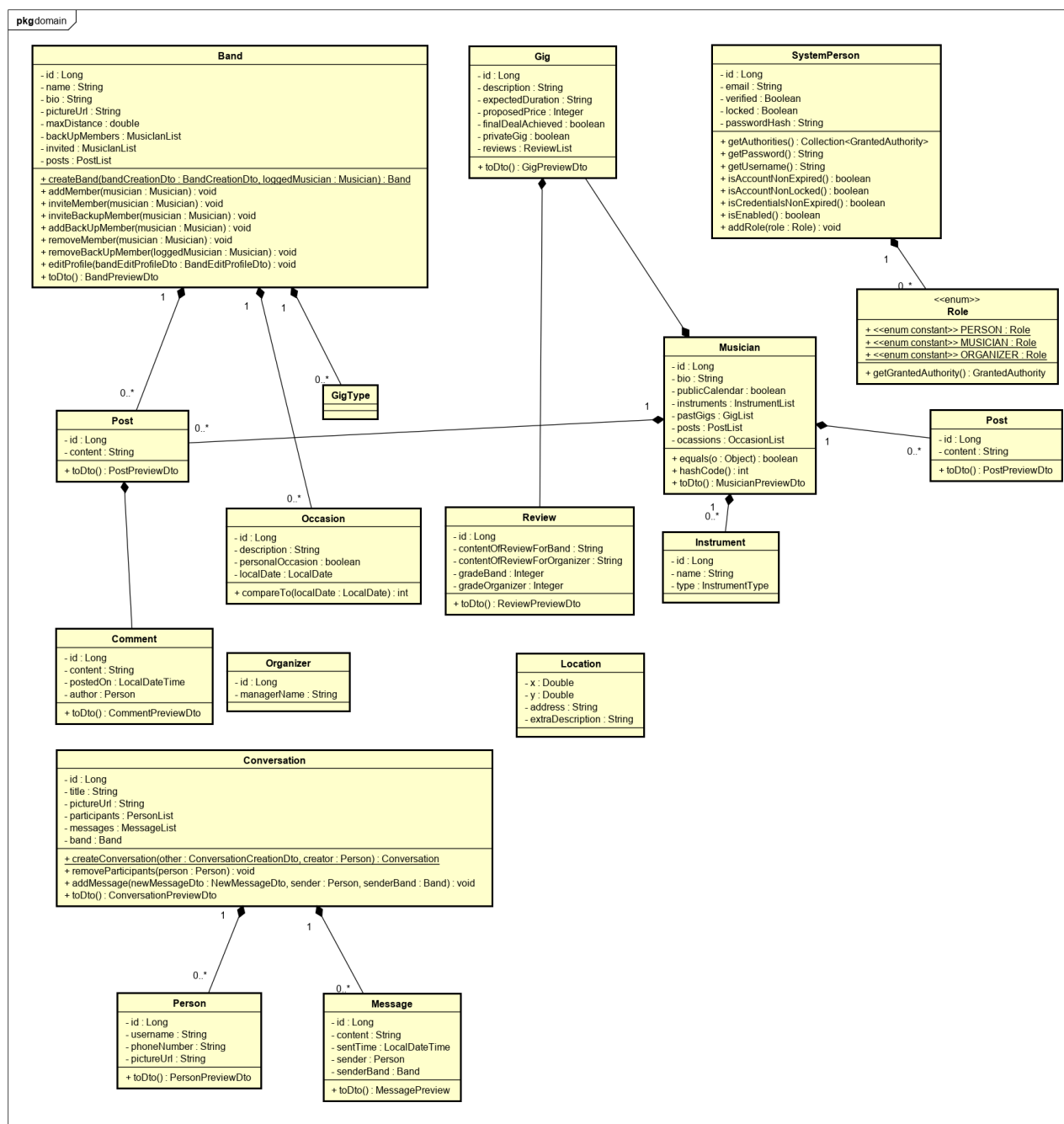
GigerException, znači da je to iznimka koju smo očekivali, u protivnom je došlo do neočekivane situacije u sustavu.



Slika 4.5: Dijagram razreda - dio Service

Slika 4.6 prikazuje razrede koji predstavljaju enitete enitetsko - relacijskog modela u bazi podataka. Razred Band predstavlja bend kojim upravlja glazbenik koji je postavljen za voditelja benda. Razred SystemPerson predstavlja razred u kojem

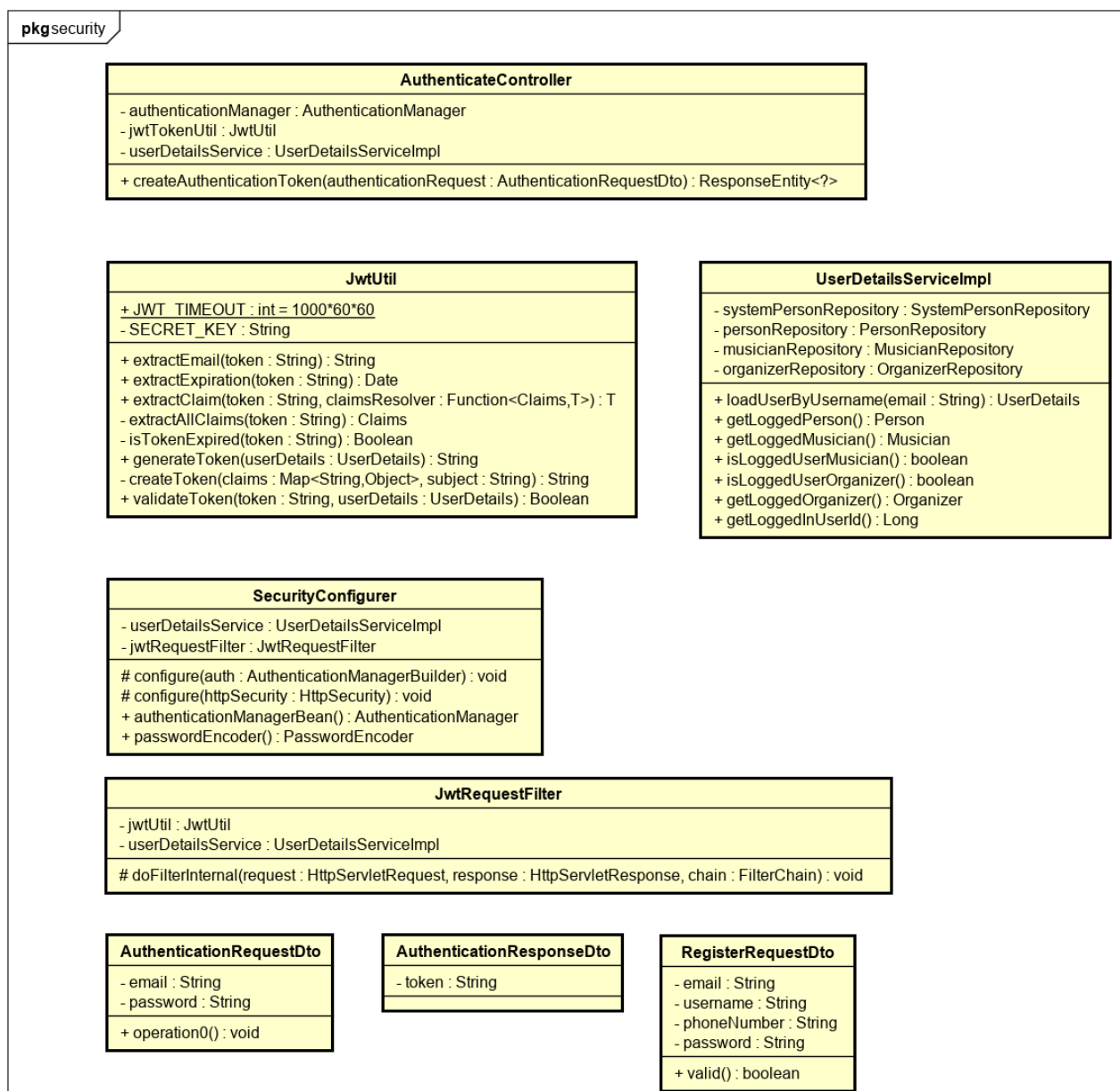
su objedinjeni svi sustavski podaci vezani za osobu kao što su: email, hash lozinke i id. Gig predstavlja nastup nekog benda koji organizira određeni organizator i pri tom enkapsulira sve logističke informacije o tom nastupu. Razred Conversation omogućuje komunikaciju između aktora glazbenika i organizatora. Glazbenik je opisan razredom Musician, dok je organizator opisan razredom Organizer. Razredi Post, Comment, Location, Instrument, Message služe za enkapsulaciju informacija kako bi dopunili razrede poput Band, Conversation i Musician. GigType i Role predstavljaju enumeracije.



Slika 4.6: Dijagram razreda - razredi entiteta

Na slici 4.7 prikazan je dijagram razreda zaduženih za Spring Security. Ulazna točka za autorizaciju zahtjeva je definirana u `JwtRequestFilteru` koji poziva `UserDetailsServiceImpl` da provjeri postoji li u bazi podataka zapis s akreditacijama koje se dekodiraju iz jwt tokena. `AuthenticateController` je zadužen za pružanje jwt tokena ukoliko u bazi pronade zapis s odgovarajućim vrijednostima. Aplikacija

cija ne pamti sesiju niti stanje (STATELESS) tako da korisnik prilikom postavljanja zahtjeva mora dostaviti svoj jwt token putem kojeg se autentificira i autorizira. JwtUtil klasa je puna pomoćnih metoda za baratanje jwt tokenom, a UserDetailsServiceImpl je posrednik između baze korisnika i AuthenticateControllera. Vezano uz autorizaciju, modelirana su tri DTO objekta koji definiraju objekte koje backend prima i daje kao zahtjev ili odgovor.



Slika 4.7: Dijagram razreda - security

Popis literature

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
2. Astah Community, <http://astah.net/editions/uml-new>
3. GitLab Pipelines, <https://docs.gitlab.com/ee/ci/pipelines.html>
4. Sprint Boot, <https://docs.spring.io/spring-boot/docs/current-SNAPSHOT/reference/htmlsingle/>

Indeks slika i dijagrama

2.1	Prikaz <i>Amy</i> aplikacije	6
2.2	Prikaz <i>BandFriend</i> aplikacije	7
3.1	Obrasce uporabe za korisnika i organizatora	23
3.2	Obrasce uporabe za javnost i administratora	24
3.3	Obrasce uporabe za korisnika	25
3.4	Sekvencijski dijagram za UC4	26
3.5	Sekvencijski dijagram za UC10	27
3.6	Sekvencijski dijagram za UC15	28
3.7	Sekvencijski dijagram za UC24	29
4.1	Pojednostavljeni prikaz MVC-a	33
4.2	Dijagram baze podataka	42
4.3	Dijagram razreda - dio Controllers	43
4.4	Dijagram razreda - dio Data transfer objects	44
4.5	Dijagram razreda - dio Service	45
4.6	Dijagram razreda - razredi entiteta	47
4.7	Dijagram razreda - security	48

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: 3. listopada 2019.
- Prisutni: I. Juren, T. Krmek, M. Jurić, M. Zec, S. Gaši, M. Nosil, P. Lanča
- Teme sastanka:
 - predlaganje ideja za projektni zadatak
 - odabir između web ili mobilne aplikacije
 - svaki član je iznio koja predznanja ili iskustva ima vezano za stvaranje aplikacije

2. sastanak

- Datum: 9. listopada 2019.
- Prisutni: I. Juren, T. Krmek, M. Jurić, M. Zec, S. Gaši, M. Nosil, P. Lanča
- Teme sastanka:
 - upoznavanje s mentorima i demonstratorom
 - razgovor o tehnologijama koje ćemo koristiti
 - dogovoren način komunikacije s asistentom i demonstratorom
 - upoznavanje s ponuđenim projektom te razgovor o tome kako poboljšati naš prijedlog projekta

3. sastanak

- Datum: 14. listopada 2019.
- Prisutni: I. Juren, M. Jurić, M. Zec, S. Gaši, P. Lanča
- Teme sastanka:
 - sastanak s asistentom
 - nacrtana gruba shema različitih korisnika s pripadajućim potrebnim pristupom
 - predlaganje funkcionalnosti, dogovoreno što se obavezno mora implementirati

4. sastanak

- Datum: 22. listopada 2019.
- Prisutni: I. Juren, T. Krmek, M. Jurić, M. Zec, S. Gaši, M. Nosil, P. Lanča
- Teme sastanka:
 - razrada must i could have funkcionalnosti
 - izjašnjavanje svojih nedoumica te njihovo razrješavanje, eventualno stavljene na popis za pitanja na sastanku s asistentom
 - razriješena problematika benda (glavni i rezervni članovi)
 - nakon internog, sastanak s asistentom: dogovorena detaljnija implementacija, napravljen popis zadataka koje treba odraditi do idućeg sastanka

5. sastanak

- Datum: 28. listopada 2019.
- Prisutni: I. Juren, M. Jurić, M. Zec, S. Gaši, M. Nosil, P. Lanča
- Teme sastanka:
 - nabranje usecase-ova
 - podjela rada
 - određena pitanja za idući sastanak s asistentom

6. sastanak

- Datum: 29. listopada 2019.
- Prisutni: I. Juren, M. Jurić, M. Zec, S. Gaši, M. Nosil, P. Lanča
- Teme sastanka:
 - sastanak s asistentom: pokazano što je sve napravljeno
 - napravljen popis zadataka koji moraju biti gotovi do idućeg sastanka s asistentom i sve što još treba za prvu verziju
 - riješena dilema oko recenzija
 - razriješen problem solista, bit će jednočlani bend
 - rasprava oko baze podataka, što treba promijeniti i poboljšati

7. sastanak

- Datum: 11. studenog 2019.
- Prisutni: I. Juren, M. Jurić, M. Zec, S. Gaši, M. Nosil, P. Lanča, T. Krmek
- Teme sastanka:
 - određivanje preostalih poslova te njihov raspored po članovima
 - na sastanku dovršen frontend te je aplikacija isporučena

8. sastanak

- Datum: 12. listopada 2019.

- Prisutni: I. Juren, M. Jurić, M. Zec, S. Gaši, M. Nosil, P. Lanča, T. Krmek
- Teme sastanka:
 - sastanak s asistentom: pokazana aplikacija te dokumentacija
 - razriježene neke nedoumice oko baze podataka

Tablica aktivnosti

	Ivan Juren	Stela Gaši	Marin Jurić	Tomislav Krnek	Paolo Lanča	Mihael Nosil	Mario Zec
Upravljanje projektom	2	1		1			1
Opis projektnog zadatka	3	2					
Funkcionalni zahtjevi	1	2				1	2
Opis pojedinih obrazaca		2	1		2	10	4
Dijagram obrazaca					1		3
Sekvencijski dijagrami		2	2		1		
Opis ostalih zahtjeva			1		1		
Arhitektura i dizajn sustava	3				2		
Baza podataka	6	5		3			
Dijagram razreda	2	1					5
Dnevnik sastajanja	1	1			3		
Popis literature	1						
<i>Frontend</i>	1		2	15		2	
<i>Backend</i>	20						
<i>Vrijeme provedeno na sastancima</i>	10	10	10	7	10	9	10