

Konkurentan pristup bazi podataka

Đorđe Jovanović SW42/2019

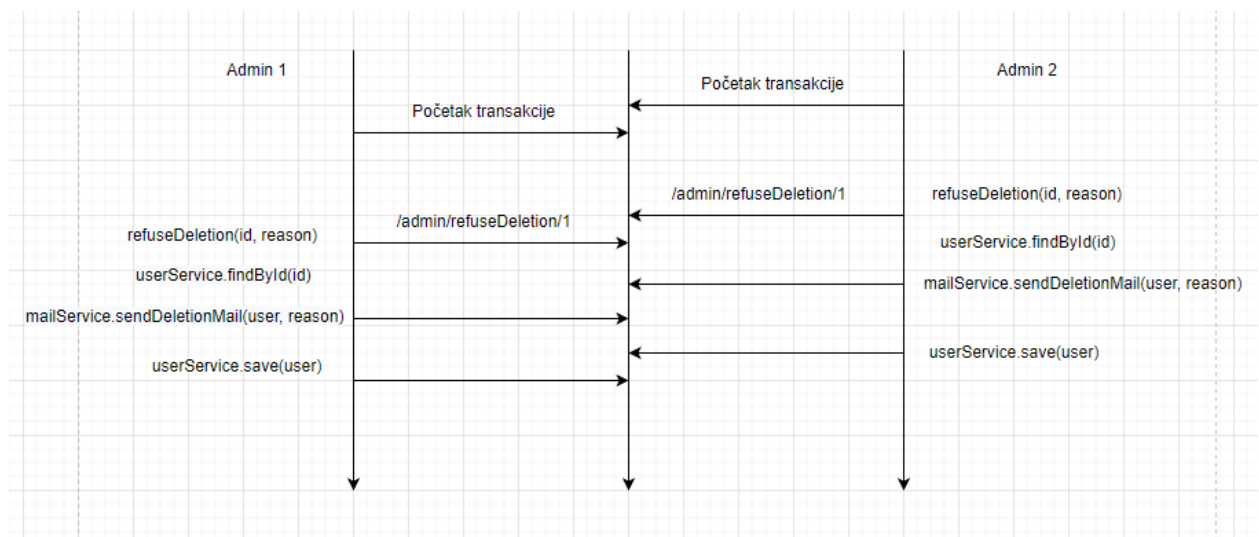
Rešavane konfliktne situacije:

1. Na jedan zahtev za brisanje naloga može da odgovori samo jedan administrator sistema
2. Na jednu žalbu može da odgovori samo jedan administrator sistema
3. Na jednu reviziju može da odgovori samo jedan administrator sistema
4. Na jedna zahtev za registraciju može da odgovori jedan administrator sistema

Problemi:

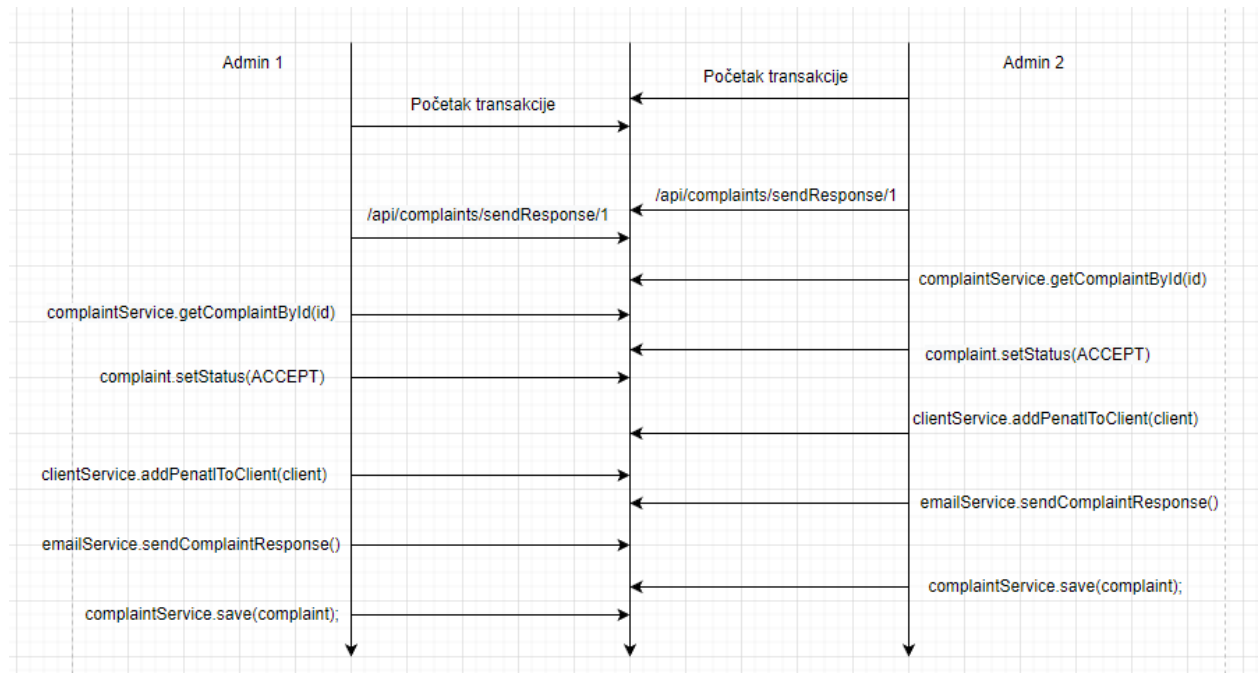
1. *Na jedan zahtev za brisanje naloga može da odgovori samo jedan administrator sistema*

Scenario problema: Dva različita admina imaju mogućnost pregleda svih pristiglih zahteva za brisanje naloga. Ako admin želi da odbije zahtev za brisanje dužan je da unese razlog za odbijanje zahteva kako bi se taj razlog prosledio klijentu putem mail-a. Kako bi se izbegla mogućnost da korisnik primi dva različita mail-a sa dva različita razloga potrebno je uvesti konkurentan pristup bazi podataka. Ukoliko je jedan admin već poslao odgovor klijnetu drugi admin toga neće biti svestan jer će njemu i dalje biti prikazana lista gde taj zahtev i dalje čeka na odgovor.



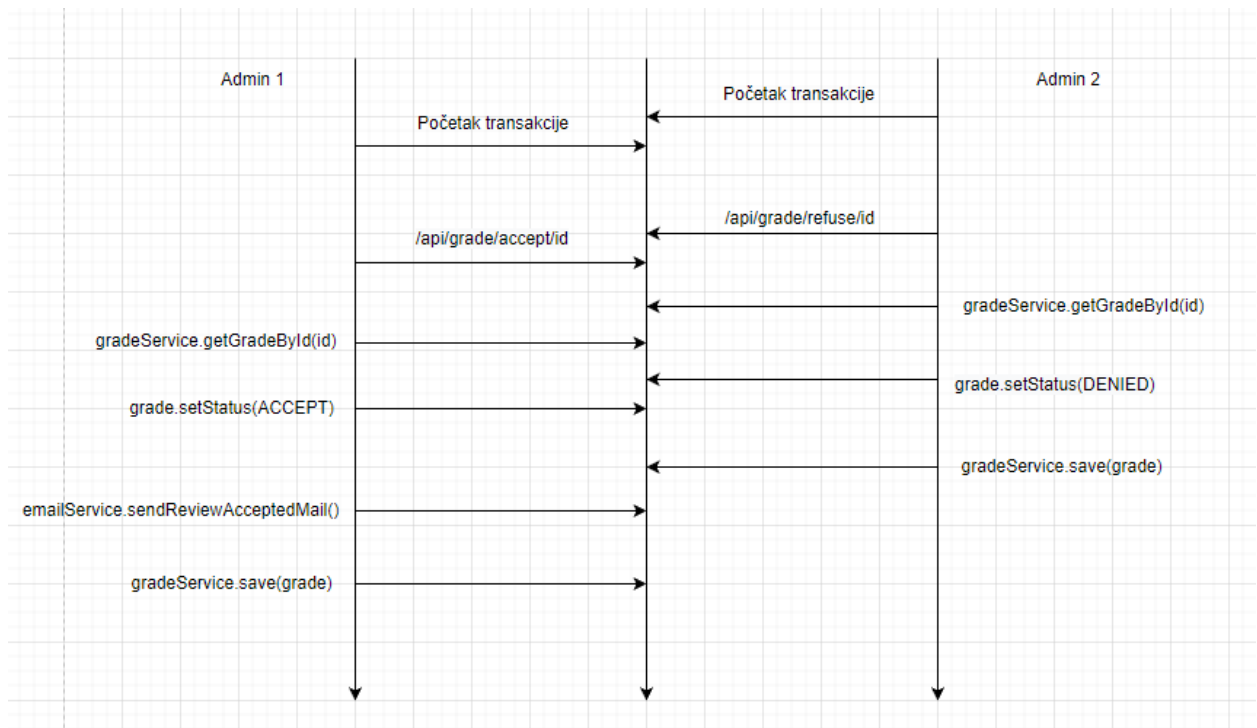
2. Na jednu žalbu može da odgovori samo jedan administrator sistema

Scenario problema: Dva različita admina imaju mogućnost pregleda svih pristiglih žalbi na klijente i žalbi na pružaoce usluga. Ako admin želi da prihvati žalbu, a u žalbi se zahteva kažnjavanje klijenta, klijentu će biti dodat *penalty point*. Kako bi se izbegao slučaj gde klijent može dobiti dva penalty point-a za jednu žalbu potrebno je uvesti konkurentan pristup bazi podataka.



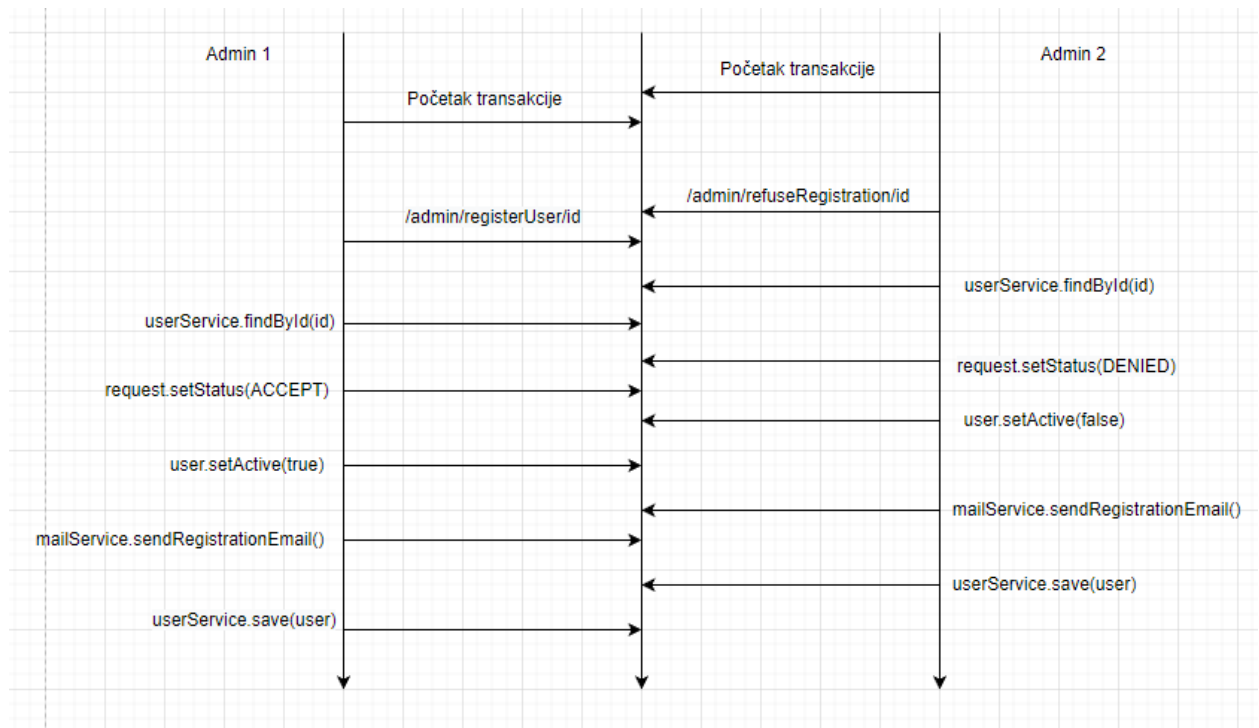
3. Na jednu reviziju može da odgovori samo jedan administrator sistema

Scenario problema: Dva različita admina imaju mogućnost pregleda svih pristiglih revizija za usluge koje pružaju vlasnici brodova, vikendica i instruktori. U slučaju da revizija bude prihvaćena od strane admina biće prikazana na profilu usluge. Kako bi se izbegla situacija u kojoj admin broj dva odbije reviziju, a zatim admin broj jedan prihvati reviziju i ona bude prikazana na profilu usluge i ako ne bi trebala biti jer je već odbijena, potrebno je uvesti konkurentan pristup bazi podataka.



4. Na jedan zahtev za registraciju može da odgovori jedan administrator sistema

Scenario problema: Dva različita admina imaju mogućnost pregleda svih pristiglih zahteva za registraciju naloga. Ako admin želi da odbije zahtev za registraciju dužan je da unese razlog za odbijanje zahteva kako bi se taj razlog prosledio klijentu putem mail-a. Kako bi se izbegla mogućnost da korisnik primi dva različita mail-a sa dva različita razloga potrebno je uvesti konkurentan pristup bazi podataka. Ukoliko je jedan admin već poslao odgovor klijnetu drugi admin toga neće biti svestan jer će njemu i dalje biti prikazana lista gde taj zahtev i dalje čeka na odgovor. Takođe potrebno je izbeći slučaj da će klijent moći da se prijavi na nalog i ako je njegov zahtev prvobitno odbijen. Do ovakovog slučaja će doći ako admin broj dva odbije zahtev za registraciju, a admin jedan prihvati zahtev za registraciju ne znajući da je taj zahtev već odbijen.



Rešenje:

Za rešenje problema korišćeno je optimističko zaključavanje. Kako je predloženo na vežbama u slučaju da menjamo atribute postojećih entiteta u bazi podatak optimalnije je korišćenje optimističkog zaključavanja. U tabelu entiteta sa kojima je rađeno u primerima iznad dodata je kolona *version* sa anotacijom *@Version*. Naime, izmenom nekog od atributa entiteta iz baze podataka, *version* kolona će biti inkrementirana. Npr. ako su dva admina preuzeli zahtev za brisanje čija je verzija bila 0, sledećim pozivom *save* metode i ukoliko je došlo do izmene nekog od atributa, verzija ovog zahteva postaje 1. Admin koji pokuša da uradi *save* entiteta nakon što je verzija entiteta već promenjena neće uspeti u tome jer je verzija entiteta koju on poseduje zastarela. Doći će do *ObjectOptimisticLockingFailureException* greške. Po ugledu na vežbe, dodati su odgovarajući testovi koji potvrđuju rad optimističkog zaključavanja.