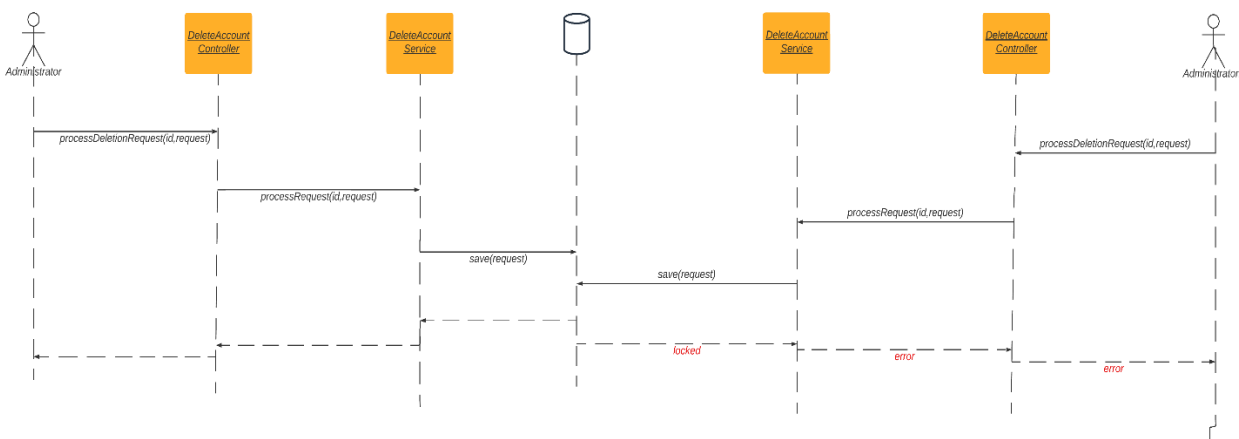


## 1. Na jedan zahtev za brisanje naloga može da odgovori samo jedan administrator sistema

Administrator sistema ima pregled svih zahtjeva za brisanje naloga i procesira ih - odobrava ili odbija.

### Opis situacije:

Posmatrajmo sledeći scenario: dva administratora sistema, Admin1 i Admin2, pregledaju zahtjeve za brisanje naloga. Admin1 klikne na dugme za odobravanje zahtjeva. U tom trenutku šalje se HTTP zahtjev na server i vrši se provjera da li je taj zahtjev već obrađen. Ukoliko to nije slučaj, zahtjev biva odobren i izmjene se čuvaju u bazu. **Problem:** Admin2 za isti zahtjev za brisanje naloga klikne na dugme za odbijanje zahtjeva i šalje drugi HTTP zahtjev na server. I on će provjeriti da li je zahtjev već obrađen, ali u tom trenutku izmjena Admina1 još uvijek nije sačuvana u bazu - zahtjev će proći validaciju i finalno stanje u bazi biće "odbijen", ali će korisnik dobiti 2 obaveštenja - prvo da je zahtjev prihvaćen, zatim da je odbijen.



### Rješenje: optimističko zaključavanje resursa

U klasu *DeleteAccountRequest* koja predstavlja zahtjev za brisanje naloga dodat je atribut sa anotacijom `@Version`, tipa `Integer`. Tako Hibernate-u pri svakoj izmjeni torke vrši inkrement atributa `version` ako je njena vrijednost ista kao i na početku konverzacije. U suprotnom, ako se verzije ne poklapaju dobija se

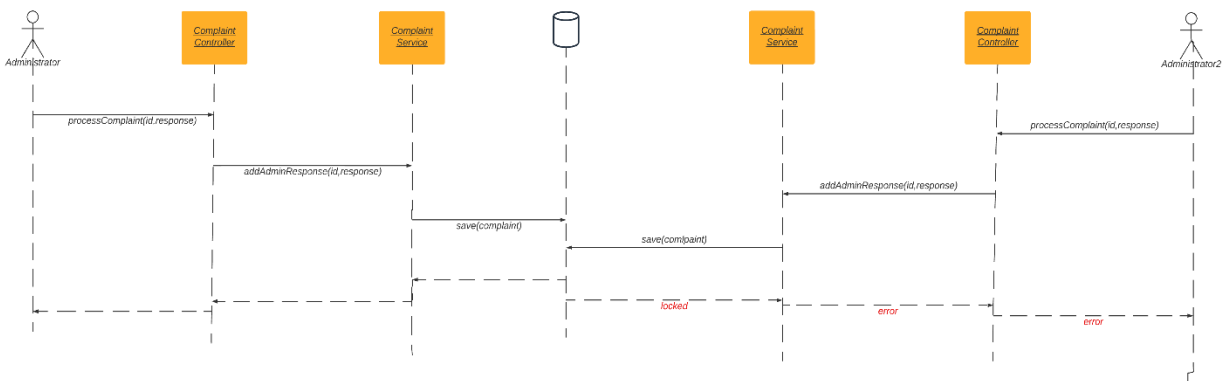
**ObjectOptimisticLockingFailureException.** Tako će u prethodnom scenariju Admin2 izazvati ovaj izuzetak kada u se u okviru servisne klase pozove čuvanje zahtjeva.

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

Administrator sistema ima pregled svih žalbi koje su podnijeli korisnici po završetku rezervacija i šalje odgovore na žalbe i klijentima i vlasnicima.

### Opis situacije:

Posmatrajmo sledeći scenario: dva administratora sistema, Admin1 i Admin2, pregledaju žalbe korisnika. Admin1 klikne na dugme za unos odgovora na žalbu i zatim klikne na dugme za potvrdu. U tom trenutku šalje se HTTP zahtjev na server i vrši se provjera da li je na tu žalbu već odgovoreno. Ukoliko to nije slučaj, mijenja se status žalbe, šalju se odgovori klijentu i vlasniku, i izmjene se čuvaju u bazu. Problem: Admin2 za istu žalbu unosi odgovor i šalje drugi HTTP zahtjev na server. I on će provjeriti da li je na žalbu već odgovoreno, ali u tom trenutku izmjena Admina1 još uvijek nije sačuvana u bazu - zahtjev će proći validaciju i klijent i vlasnik će dobiti dva odgovora - po jedan od svakog administratora.



Rješenje: Optimističko zaključavanje resursa

U klasu Complain koja predstavlja žalbu dodat je atribut sa anotacijom @Version, tipa Integer. Tako će u opisanom scenariju Admin2 izazvati ObjectOptimisticLockingFailureException kada u se u okviru servisne klase pozove čuvanje žalbe (analogno 1.)

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

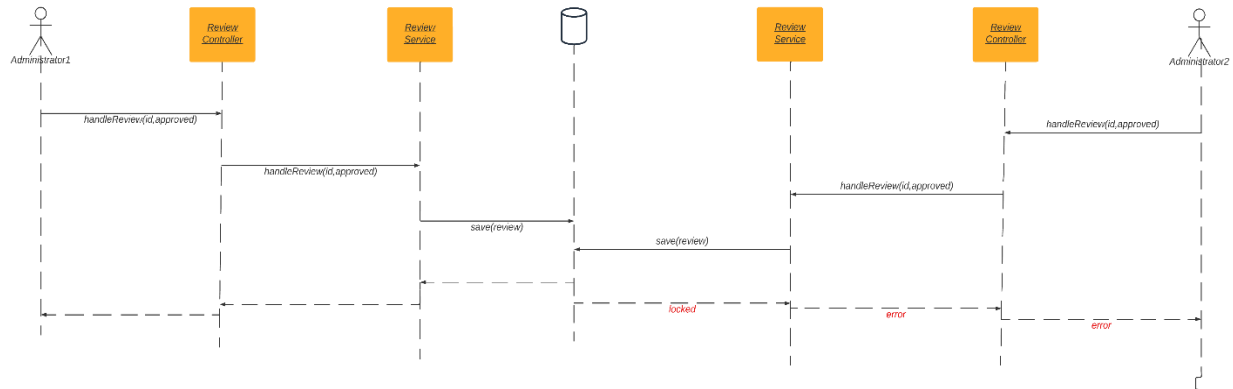
Administrator sistema ima pregled svih revizija klijenata po završetku rezervacija i procesira ih – odobrava ili odbija za publikaciju.

Opis situacije:

Posmatrajmo sledeći scenario: dva administratora sistema, Admin1 i Admin2, pregledaju revizije korisnika. Admin1 klikne na dugme za odobravanje revizije. U tom trenutku šalje se HTTP zahtjev na server i vrši se provjera da li je ta revizija već obrađena. Ukoliko to nije slučaj, revizija biva odobrena, šalje se mejl vlasniku/instruktoru i izmjene se čuvaju u bazu. Problem: Admin2 za istu reviziju klikne na dugme za odbijanje revizije i šalje drugi HTTP zahtjev na server. I on će provjeriti da li je revizija već obrađena, ali u tom trenutku izmjena Admina1 još uvijek nije sačuvana u bazu - revizija će proći validaciju i upisati izmjene u bazu. Finalno, vlasnik će dobiti mejl s odlukom prvog administratora da je revizija odobrena, iako stanje u bazi govori da je revizija odbijena za publikaciju.

Rješenje: Optimističko zaključavanje resursa

U klasu ClientReview koja predstavlja reviziju dodat je atribut sa anotacijom @Version, tipa Integer. Tako će u opisanom scenariju Admin2 izazvati ObjectOptimisticLockingFailureException kada u se u okviru servisne klase pozove čuvanje revizije (analogno 1. i 2.)



Student 3

Ruljić Božana, RA173/2018