

Primer rešavanja konfliktnih situacija

Internet softverske arhitekture, Bojan Popržen, SW 16-2017

Kreiranje upita za pregled

Kreiranje upita za pregled podrazumeva da je pacijent pronašao unapred definisani pregled ili pak da je kreirao svoj pregled i da, zatim, podnosi upit za njegovu rezervaciju.

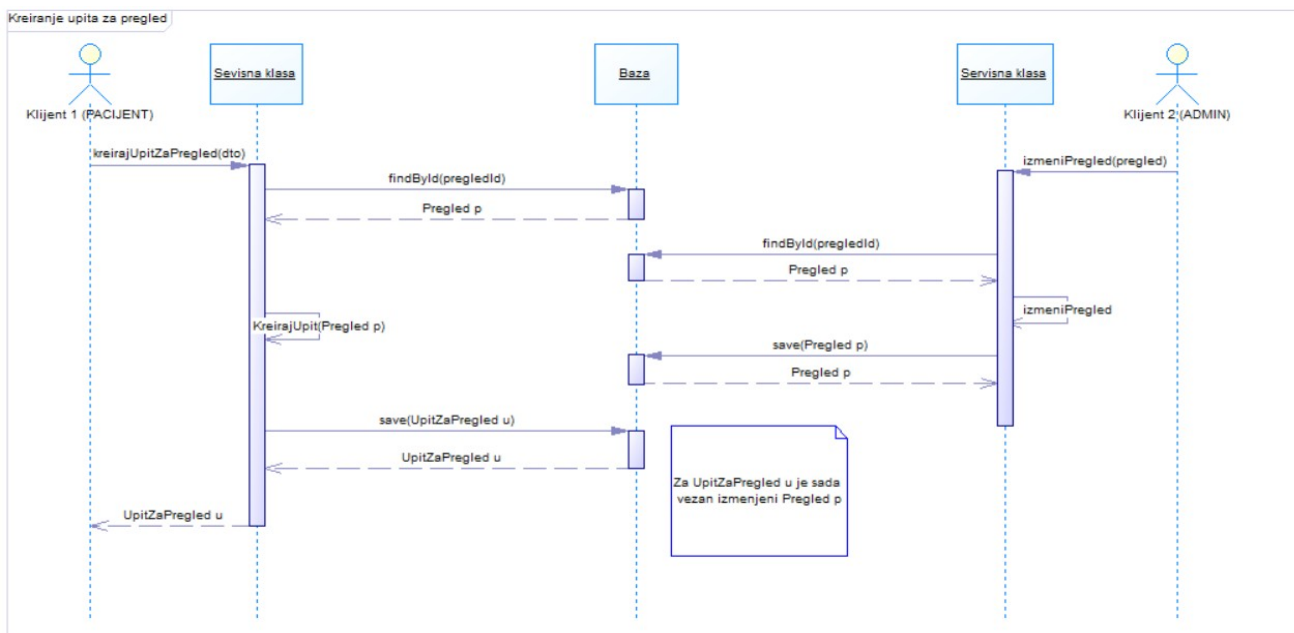
1. Opis konfliktnih situacija

Konfliktne situacije koje nastaju prilikom kreiranja upita za pregled vezane su za **konkurentnu izmenu entiteta** sa kojim je takav upit u vezi: Pregled, Lekar I TipPregleda. Moguće konfliktne situacije su:

1. Prilikom kreiranja upita za unapred definisani pregled P, drugi korisnik sistema (admin) konkurentno menja odabrani pregled P i pacijent naposljetku rezerviše neželjeni pregled. Primer štetne situacije je ona u kojoj admin konkurentno menja vreme pregleda I pacijent na kraju rezerviše pregled u neodgovarajuće vreme.
2. Prilikom kreiranja upita za unapred definisani pregled P, drugi korisnik sistema (admin, pacijent) konkurentno menja odabranog lekara L i pacijent naposljetku rezerviše pregled kod neželjenog lekara. Primer štetne situacije je ona u kojoj lekar L konkurentno dobija veoma lošu ocenu – ocenu koju bi pacijent uzeo u obzir pri izboru tog lekara.
3. Prilikom kreiranja upita za unapred definisani pregled P, drugi korisnik sistema (admin) konkurentno menja tip pregleda T i pacijent naposljetku rezerviše pregled koji ima neželjeni tip. Primer štetne situacije je ona u kojoj admin konkurentno menja naziv tipa pregleda i pacijent na kraju rezerviše potpuno neodgovarajući pregled po tipu.

2. Crteži tokova zahteva

Na slici ispod prikazan je tok zahteva dva korisnika aplikacije (pacijent i admin). Pacijent želi da kreira upit za pregled P koji istovremeno admin menja. Ovde je prikazan samo konflikt vezan za vezani entitet Pregled, međutim isti se konflikti mogu desiti sa entitetima TipPregleda i Lekar.



3. Rešenje

Dakle, problem definisan u 1.2 svodi se na tzv. NON-REPEATABLE READ problem koji se rešava zaključavanjem entiteta obuhvaćenih problemom. Zaključavanje entiteta podrazumeva zajednički (shared) ključ nad entitetom (dozvoljeno je čitanje svih onih koji imaju ključ) ili jedinstveni ključ (exclusive).

Odabrani JPA mehanizmi zaključavanja entiteta koji su se pokazali kao odgovarajući su:

- **PESSIMISTIC_READ** za zaključavanje entiteta Pregled, Lekar, TipPregleda. Ovaj mehanizam definiše zajednički ključ nad entitetom. Na ovaj način, obezbedili smo to da se konkurentno entiteti ne mogu izmeniti, ali se smeju čitati, što je dovoljno restriktivno za naš slučaj. Na ovaj način otklonili smo problem NON-REPEATABLE READ-a.

Ocenjivanje klinika i lekara

Ocenjivanje klinika i lekara podrazumeva da je korisnik imao posetu kod nekog Lekara L i da sada želi da ga oceni, i/ili želi da oceni kliniku kojoj taj lekar pripada.

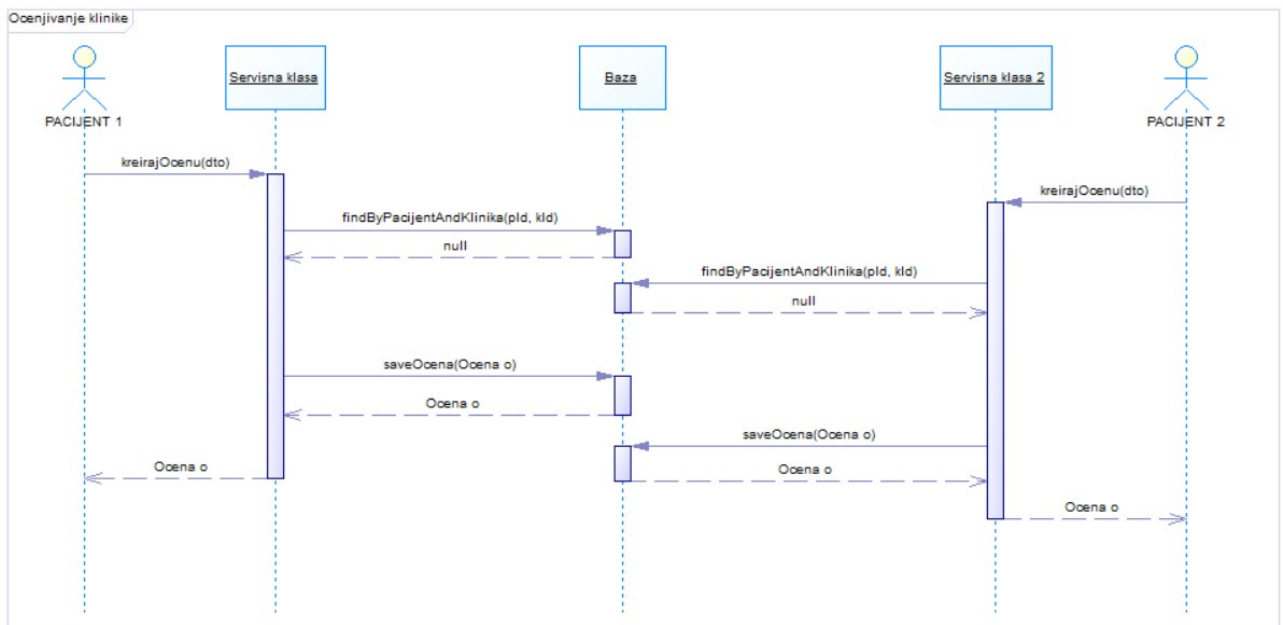
1. Opis konfliktnih situacija

Konfliktne situacije vezane su za ograničenje da pacijent može imati najviše po jednu ocenu za određenog lekara i određenu kliniku. Primer konfliktne situacije je:

- Pacijent u malim vremenskim razmacima dva puta šalje ocenu za istu kliniku. Dešava se NON-REPEATABLE READ problem, gde obe provere da pacijent ima po najviše jednu ocenu za istog kliniku prođu i ocene se sačuvaju čime se ograničenja prekršavaju.

2. Crteži tokova zahteva

Na slici ispod prikazano je kako isti pacijent želi da oceni istu kliniku istovremeno. Metoda `findByPacijentAndKlinika` će u oba slučaja vratiti NULL kao indikator da takve ocene nema, te će se kasnije obe ocene dodati uspešno.



3. Rešenje

Odabrani JPA mehanizmi zaključavanja entiteta koji su se pokazali kao odgovarajući su:

PESSIMISTIC_WRITE za zaključavanje entiteta Pacijent I Klinika i/ili Lekar. Ovaj mehanizam definiše ekskluzivni ključ nad entitetom. Na ovaj način, obezbedili smo to da se konkurentno entiteti ne mogu izmeniti ni čitati, dok imamo ključ nad njim. Na ovaj način otklonili smo problem NON-REPEATABLE READ-a.

Odabrao sam ovaj metod zaključavanja zato što nijedna druga transakcija ne sme da ima pristup ocenama pacijenta, klinike ili lekara u toku transakcije koja dodaje novu ocenu.