

KONKURENTNI PRISTUP

Resenje za konkurentne pristupe bazi- **pesimisticko zakljucavanje i optimisticko zakljucavanje na bazi version polja**:

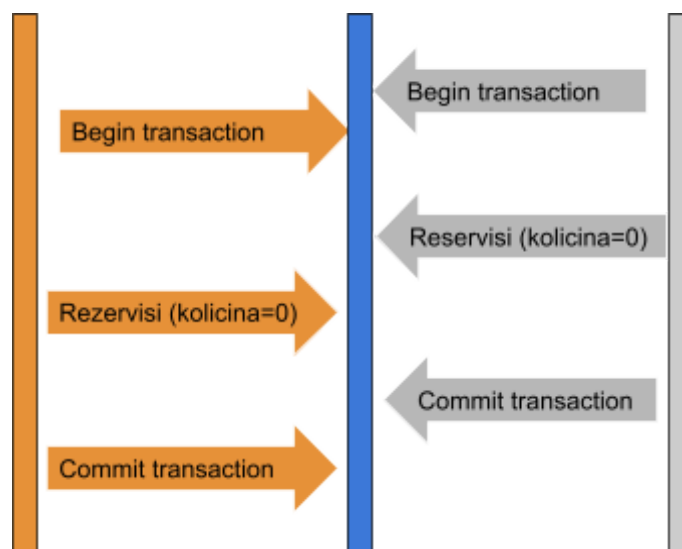
Jedna od osnovnih komponenti koje nudi TypeORM je *QueryBuilder* koji nam omogućava formiranje SQL upita ka bazi. Sledeci kod ekvivalentan je SELECT upitu gde se trazi korisnik cije ID polje ima vrednost 1.

```
const firstUser = await connection
  .getRepository(User)
  .createQueryBuilder("user")
  .where("user.id = :id", { id: 1 })
  .getOne();
```

TypeORM takodje nudi *@VersionColumn* anotaciju koja funkcioniše analogno *@Version* anotaciji iz JPA. Iako *QueryBuilder* nudi pesimisticko zakljucavanje, sa razlicitim nivoima izolacije, uvođenjem version polja u entitete dobijamo mogucnost da koristimo optimisticko zakljucavanje koje je efikasnije. Sledeci kod predstavlja primer SELECT upita za svim korisnicima koji koristi optimisticko zakljucavanje. Kao parametar zakljucavanja koristi se version polje korisnika.

```
const users = await getRepository(User)
  .createQueryBuilder("user")
  .setLock("optimistic", existUser.version)
  .getMany();
```

Opis konfliktne situacije 1: Više istovremenih korisnika aplikacije pokušava da rezerviše lek koji je u međuvremenu postao nedostupan

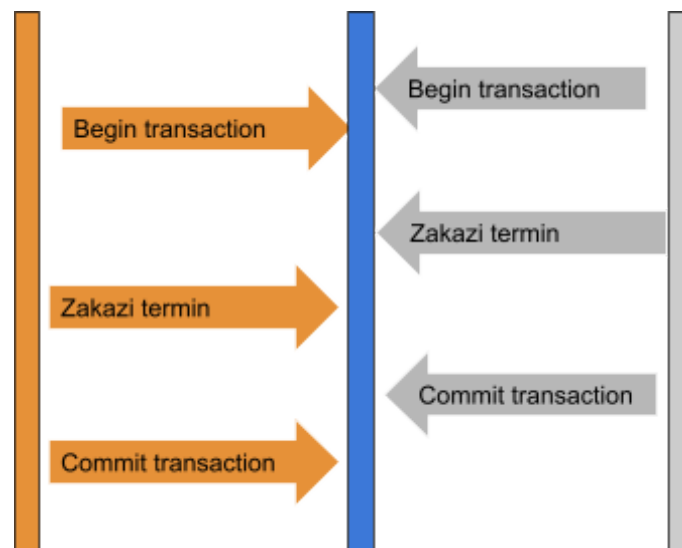


Resenje: **Optimisticko zakljucavanje Inventory** entiteta. Prilikom cuvanja rezervacije leka (tj odgovarajuće stavke inventara apoteke) ako se promenila verzija te stavke transakcija se obustavlja.

Opis konfliktne situacije 2: Nekorektno azuriranje količine leka na stanju u apoteci. Uzrok je istovremeno finalizovanje 2 od sledećih funkcija: rezervacija leka, otkazivanje rezervacije, izdavanje leka preko recepta/eRecepta, prihvatanje ponude od dobavljača

Rešenje: Analogno prethodnoj konfliktnoj situaciji izbegavamo nekorektno azuriranje uvođenjem **optimističkog zaključavanja za stavke inventara** apoteke (koje su oblika [apoteka,lek,kolicina]).

Opis konfliktne situacije 3/4: Jedan dermatolog/farmaceut ne može da istovremeno bude prisutan na više termina. Više aktivnosti može da bude uzrok, npr. moguće je da dva pacijenta pokušaju u isto vreme da zakazu termine, kod istog farmaceuta, koji se preklapaju.



Rešenje: Kako ne postoji entitet Kalendar za dermatologe i farmaceute ovaj problem se ne može rešiti optimističkim zaključavanjem. Zbog toga prilikom čuvanja novog termina vršimo **pesimističko zaključavanje termina** tokom kog proveravamo da li se novi termin preklapa s nekim od postojećih.

Opis konfliktne situacije 5: Više korisnika pokušava da zakaze unapred definisan termin

Rešenje: **Optimističko zaključavanje** na osnovu verzije **termina**. Ukoliko primetimo da je termin već neko izmenio transakcija će biti zaustavljena, u suprotnom proveravamo za termin da li je polje patient Null i ako jeste zauzimamo termin.

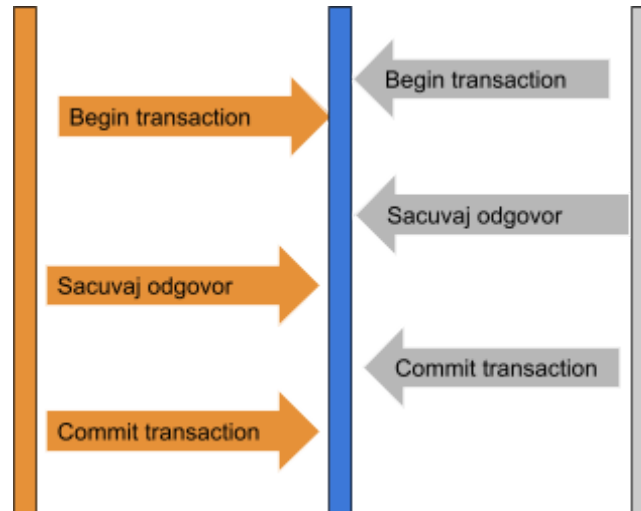
Opis konfliktne situacije 6: Više korisnika pokušava da zakaze termin kod istog farmaceuta u istom periodu (ili s presekom).

Rešenje: Kao u situaciji 4 kako termini nastaju prvi put možemo jedino koristiti **pesimističko zaključavanje termina**.

Opis konfliktne situacije 7: Prilikom izdavanja eRecepta neki od lekova nije dostupan- potrebno odbiti ceo recept

Resenje: Kako se ovde ne radi o pojedinacnim stavkama u inventaru apoteke moramo izvršiti **pesimisticko zakljucavanje inventara**. Dok je inventar apoteke zakljućan proveravamo dostupnost svih lekova i tek ako je moguće za sve azuriramo kolićine.

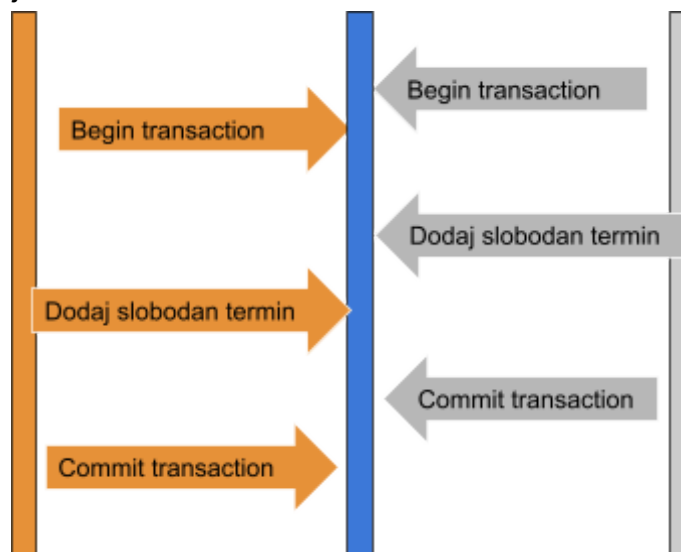
Opis konfliktne situacije 8: Više administratora sistema je pokušalo da odgovori na istu žalbu



Resenje: **Optimisticko zakljucavanje žalbe**. Zahtev jednog administratora će biti odbijen nakon čega neće više moći da vidi žalbu jer je neko već odgovorio.

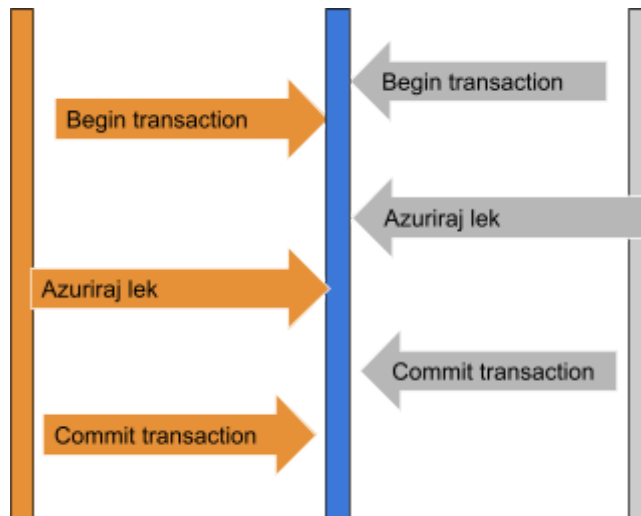
Dodatne konfliktne situacije:

1. Dva administratora apoteke pokušavaju za istog dermatologa da definišu slobodne termine koji se preklapaju



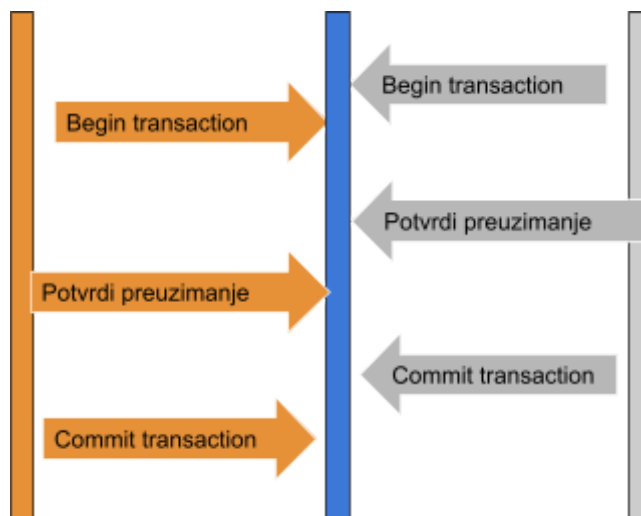
Resenje: Analogno problemu gde dva korisnika zakazuju savetovanje kod istog farmaceuta. Vrsimo **pesimisticko zakljucavanje termina** i ukoliko ne postoji termin s kojim se novi slobodan termin preklapa dodajemo ga. U suprotnom obavestavamo administratora da novi termin nije moguć.

2. Dva administratora sistema istovremeno redefinišu lek u sifarniku.



Resenje: **Optimisticko zakljucavanje entiteta Medicine** na osnovu version polja.
 Administrator ciji zahtev za izmenu leka stigne kasnije ce dobitja obavestenje da je lek redefinisan nakon cega moze ponovo da pokuša da redefiniše lek.

3. Dva farmaceuta pokusavaju da potvrde preuzimanje iste rezervacije leka



Resenje: **Optimisticko zakljucavanje rezervacije**. Za farmaceuta ciji zahtev za potvrdu rezervacije stigne kasnije transakcija se obustavlja na osnovu izmenjenog version polja.