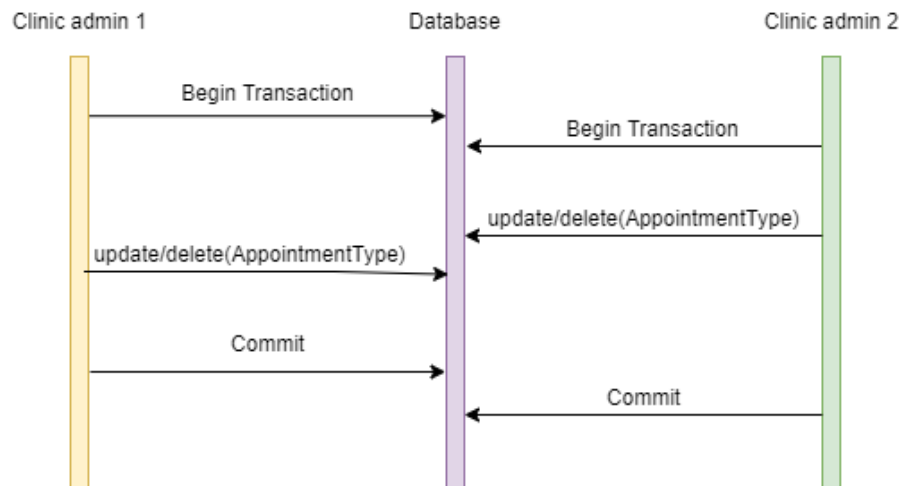


Konfliktne situacije – student 2 (Ksenija Prčić sw10/2017)

1) Više administratora klinike istovremeno briše¹/menja isti tip pregleda

Opis konfliktne situacije: Nakon što se administrator klinike uloguje u aplikaciju na svom početnom ekranu može da izabere pregled svih tipova pregleda koji postoje u njegovoj klinici. Na stranici za pregled tipova pregleda, može da odabere dodavanje novog tipa pregleda ili izmenu/brisanje nekog od postojećih. Potrebno je napomenuti da se izmena/brisanje tipa pregleda može izvršiti samo ako u tom trenutku nijedan zakazan pregled nije tog tipa pregleda. Do konfliktne situacije dolazi ukoliko dva ili više administratora klinike pokušaju istovremeno da obrišu/izmene isti tip pregleda.

Crtež konfliktne situacije: Administrator klinike 1 započinje svoju transakciju, pa zatim administrator klinike 2 započinje svoju transakciju. Administrator klinike 1 menja/briše tip pregleda i komituje svoje promene, zatim administrator klinike 2 pokušava da uradi isto. Ukoliko dozvolimo da oba administratora urade željene promene dolazi do fenomena LostUpdate u kom jedan administrator pregazi izmene



drugog, nakon prvog komita u bazu.

Opis rešenja konfliktne situacije: Problem LostUpdate je rešen zaključavanjem objekta AppointmentType. Kada jedan administrator klinike uzme objekat AppointmentType nijedan drugi administrator klinike ne bi trebao da mu pristupi i menja njegove podatke dok se prva transakcija ne završi i njene izmene komituju. Način na koji je rešena ova konfliktna situacija je optimističko zaključavanje. Klasi AppointmentType je dodato polje Long Version sa anotacijom @Version. Nakon svake izmene nad objektom vrednost atributa Version se uvećava za jedan. Kada prvi administrator klinike uzme i izmeni objekat AppointmentType njegova verzija će se povećati. Ukoliko bilo koja druga

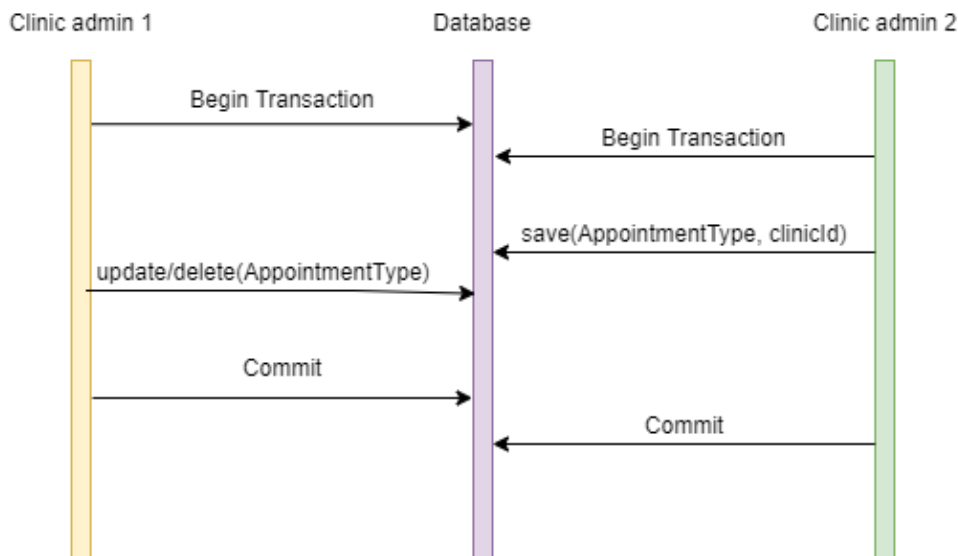
¹ Brisanje se radi logički, te predstavlja izmenu vrednosti polja active tipa pregleda

transakcija pokušava da izmeni objekat pre nego što je prethodna komitovala svoje izmene doći će do greške jer se vrednost atributa Version u transakciji i bazi neće podudarati. Na ovaj način je rešena konfliktna situacija istovremene izmene/brisanja istog AppointmentType od strane dva ili više administratora klinike.

2) Više administratora klinike istovremeno kreira tip pregleda sa istim nazivom

Opis konfliktna situacije: Kada se administrator klinike uloguje u aplikaciju na svojoj početnoj stranici može da odabere pregled svih tipova pregleda koji su dostupni u njegovoj klinici. Na stranici za pregled svih tipova može da odabere da kreira novi tip pregleda. Jedino ograničenje pri kreiranju novog tipa pregleda je da njegov naziv mora da bude jedinstven u okviru klinike. Do konfliktna situacije dolazi ukoliko dva ili više administratora klinike pokušaju da istovremeno kreiraju tip pregleda sa istim nazivom.

Crtež konfliktna situacije: Administrator klinike 1 započinje svoju transakciju, pa zatim administrator klinike 2 počinje svoju transakciju. Administrator klinike 1 dodaje/menja tip pregleda i komituje svoje promene, zatim administrator klinike 2 pokušava da doda novi tip pregleda sa istim nazivom kao izmenjeni tip pregleda. Ukoliko dozvolimo da oba administratora klinike komituju svoje promene u bazi će se nalaziti dva tipa pregleda sa istom nazivom što dovodi do nedozvoljenog stanja baze i neispravnog rada aplikacije.



Rešenje konfliktna situacije: Pošto objekat AppointmentType pre kreiranja ne postoji u bazi podataka ova konfliktna situacija se ne može rešiti pomoću optimističnog zaključavanja. Da bi ova konfliktna situacija bila rešena potrebno je podići nivo izolacije metode za kreiranje novog tipa pregleda na nivo SERIALIZABLE. U metodi za kreiranje novog tipa se šalje upit ka bazi koji dobavlja sve entitete tabele AppointmentType koji se nalaze u klinici administratora koji želi da kreira novi tip pregled i koji imaju isti naziv kao izmenjeni tip pregleda. Ukoliko upit ka bazi ne pronađe nijednu takvu torku novi AppointmentType se čuva u bazi i transakcija se komituje. U slučaju da za vreme trajanja dodavanja

novog tipa pregleda, druga transakcija izmeni neki postojeći tip pregleda tako da mu se ime poklapa sa novim, doći će do konflikta koji će biti sprečen postavljenim nivoom izolacije, to jest, drugoj transakciji (izmeni) neće biti dozvoljen komit. Na ovaj način je rešen problem konkurentnog kreiranja tipa pregleda i kombinacije kreiranja tipa pregleda + izmena tipa pregleda sa istim nazivom u istoj klinici od strane dva ili više administratora klinike.

- 3) Više administratora klinike istovremeno briše/menja istu salu, menja istu kliniku, briše isti predefinisani pregled, briše istog doktora ili briše istu medicinsku sestru

Rešenje konfliktne situacije: Sve konfliktne situacije konkurentnog brisanja/izmene od strane dva ili više administratora klinike su rešene na isti način kao za tip pregleda. Da bi optimističko zaključavanje bilo moguće svakoj tabeli entiteta koji se briše/menja je dodat atribut Long Version koji je anotiran sa @Version.

Izmjena/brisanje sale: Administrator klinike može da briše/menja salu ukoliko ona nije rezervisana za neki predstojeći pregled ili operaciju.

Brisanje/doktora: Administrator klinike ne može da briše doktora koji u svom random kalendaru ima zakazanih pregleda.

- 4) Više administratora klinike istovremeno kreira doktora, medicinsku sestru ili kreira salu sa istim jedinstvenim obeležjima

Rešenje konfliktne situacije: Sve konfliktne situacije konkurentnog upisa entiteta sa istim jedinstvenim obeležjima od strane dva ili više administratora klinike su rešene na isti način kao za dodavanje tipa pregleda. Nivo izolacije metode za dodavanje novog entiteta je podignut na SERIALIZABLE što sprečava PhantomRead.

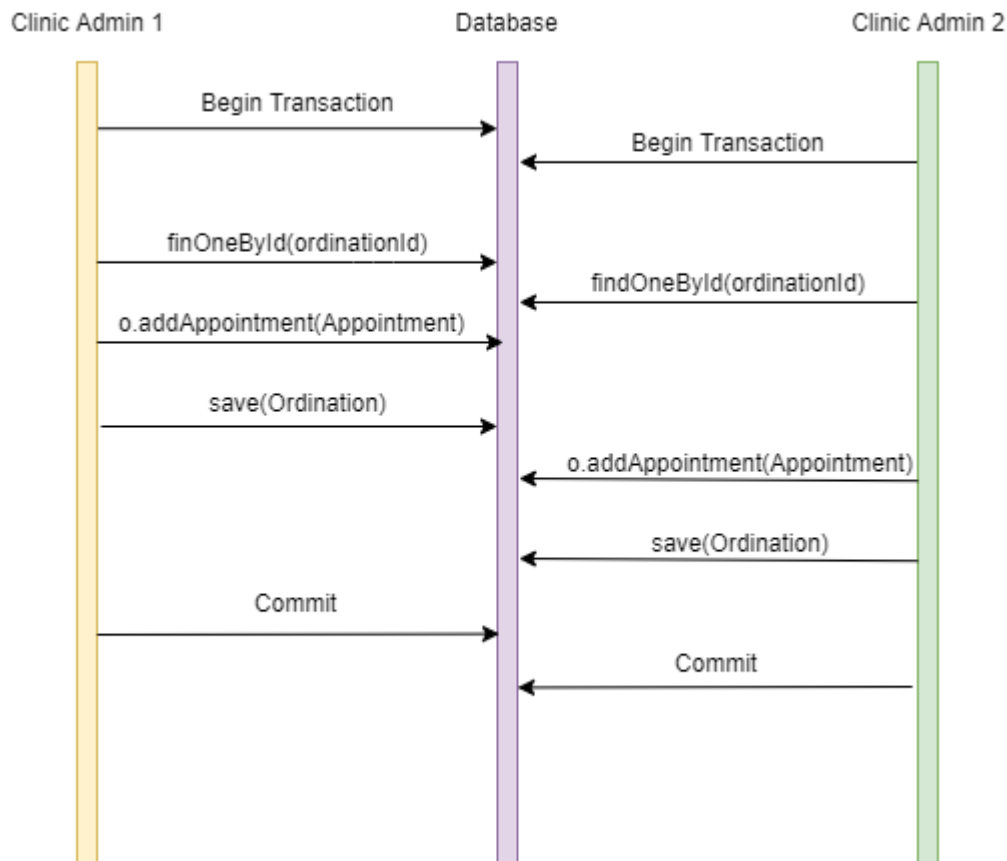
Kreiranje doktora/medicinske sestre : Pri kreiranju doktora/medicinske sestre SecurityNumber i Email moraju biti jedinstveni u sistemu.

Kreirane sale: Pri kreiranju sale kombinacija Name i OrdinationNumber mora biti jedinstvena u klinici.

- 5) Više administratora klinike istovremeno zakazuju istu salu za pregled/operaciju

Opis konfliktne situacije: Nakon što se administrator klinike uloguje u aplikaciju na svom početnom ekranu može da izabere pregled svih zahteva za pregled/operaciju koji postoje u njegovoj klinici. Na stranici za pregled zahteva za preglede, može da odabere zakazivanje sale za jedan od pregleda. Do konfliktne situacije dolazi ukoliko dva administratora klinike pokušaju istovremeno da rezervišu istu salu za isto vreme i datum.

Crtež konfliktne situacije: Administrator klinike 1 započinje svoju transakciju, zatim administrator klinike 2 započinje svoju transakciju. Administrator klinike 1 i 2 dobavljaju ordinaciju, zatim administrator klinike 1 dodaje novi pregled ordinaciji (zakazuje je) i čuva izmene, i nakon toga administrator klinike 2 pokuša da uradi isto. Oba pregleda se dešavaju istog datuma u preklapajuće vreme. Ukoliko dozvolimo da oba administratora klinike urade željeno zakazivanje, ordinacija bi bila zakazana za dva pregleda koja se dešavaju istovremeno, što dovodi do nedozvoljenog stanja baze i neispravnog rada aplikacije.



Opis rešenja konfliktne situacije: Nakon što jedan administrator klinike pristupi ordinaciji potrebno ga je zaključati za sve ostale transakcije koje žele da je menjaju. Nakon što administrator završi rad sa zauzetim objektom i komituje svoje izmene on ga otključava i ostale transakcije mogu da ga menjaju. Ovaj problem je rešen primenom optimističkog zaključavanja objekta. Klasi Ordination je dodato polje Long Version sa anotacijom @Version. Kada prva transakcija zauzme objekat, obavi izmene nad njime i komituje ih, njegov Version atribut će se uvećati za jedan. Ukoliko naredna transakcija pokuša da izmeni isti objekat dobiće izuzetak jer se vrednost polja verzije u transakciji i bazi ne podudaraju. Na ovaj način je rešen problem konkurentnog zauzimanja sale za više pregleda koji se dešavaju istovremeno.