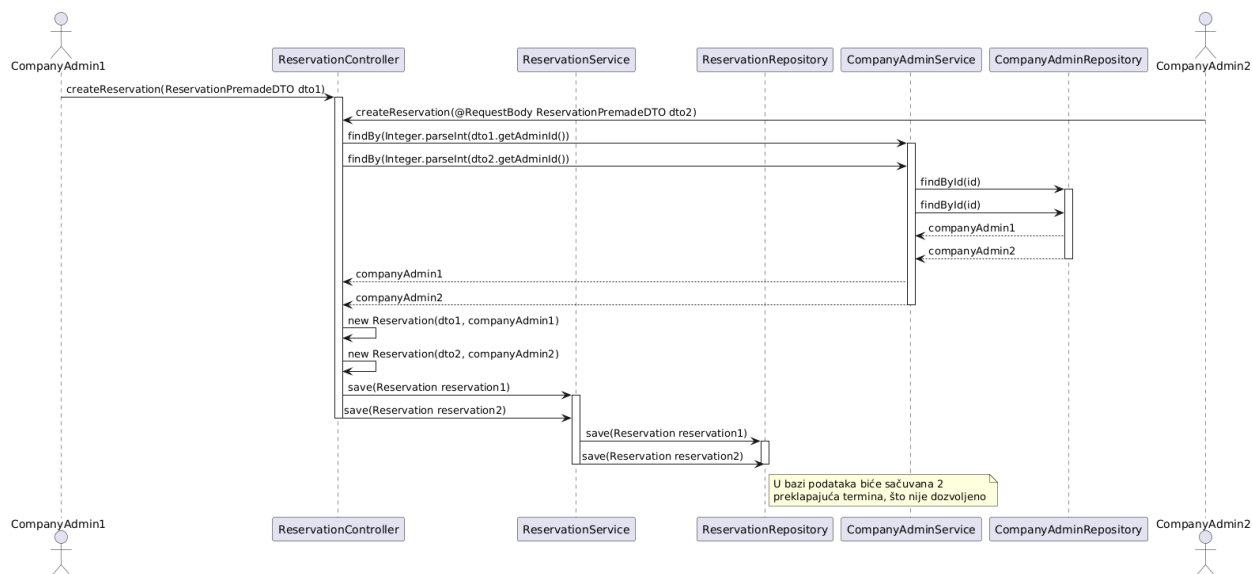


Student 3- Konfliktne situacije

Scenario 1:

Više administratora kompanije ne mogu unapred definisati termine u isto ili preklapajuće vreme.

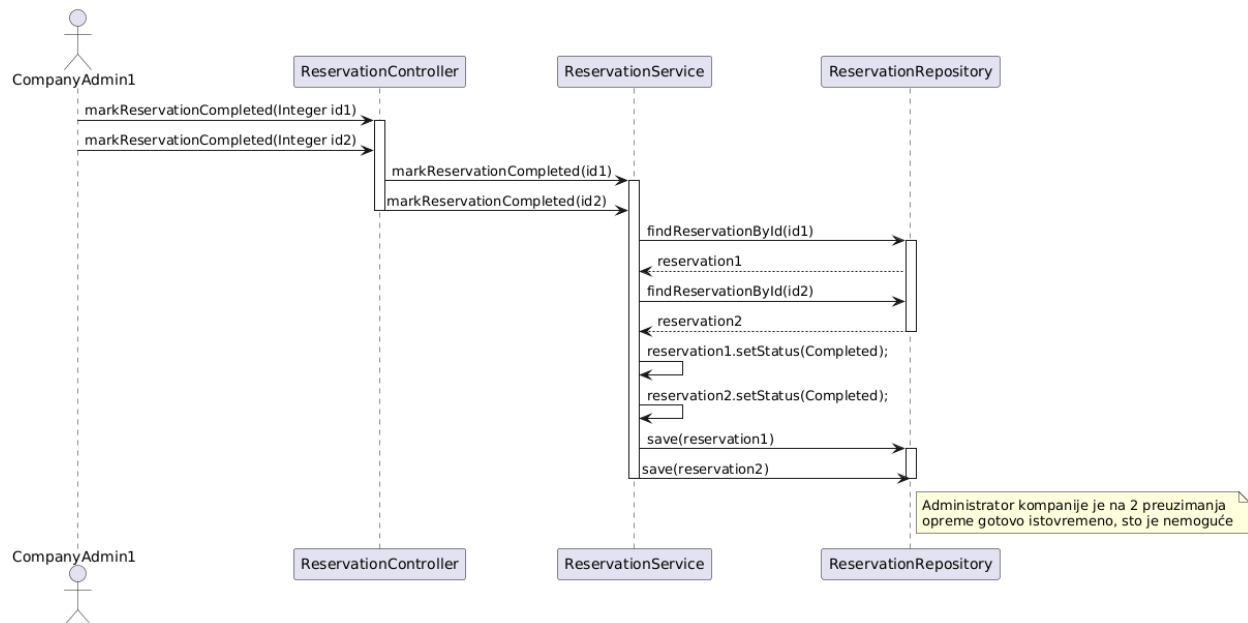


Ovaj problem rešen je putem Pessimistic Write Lock mehanizma, gde će se u bazi podataka zaključati tabela rezervacija u celosti dok se vrši upis metode koja je prva poslala zahtev za čuvanje termina. Dok se ovaj upis ne završi tabela rezervacija je zaključana.

Konflikt situacija je simulirana prijavljivanjem na aplikaciju sa privatnog i normalnog prozora pretračivača na 2 različita naloga admina kompanije. Ukoliko se termini preklapaju

Scenario 2:

Jedan administrator ne može biti prisutan istovremeno na više različitih termina preuzimanja opreme. Količina opreme na stanju se mora ispravno ažurirati nakon preuzimanja opreme od strane korisnika.



Ovaj problem je rešen tako što se pomoću Pessimistic Write Lock mehanizma zaključaju sve rezervacije koje su spremne za preuzimanje i pripadaju prijavljenom administratoru. Ovakav način zaključavanja sprečava preuzimanje drugih rezervacija od strane administratora dok se ne završi trenutno preuzimanje.

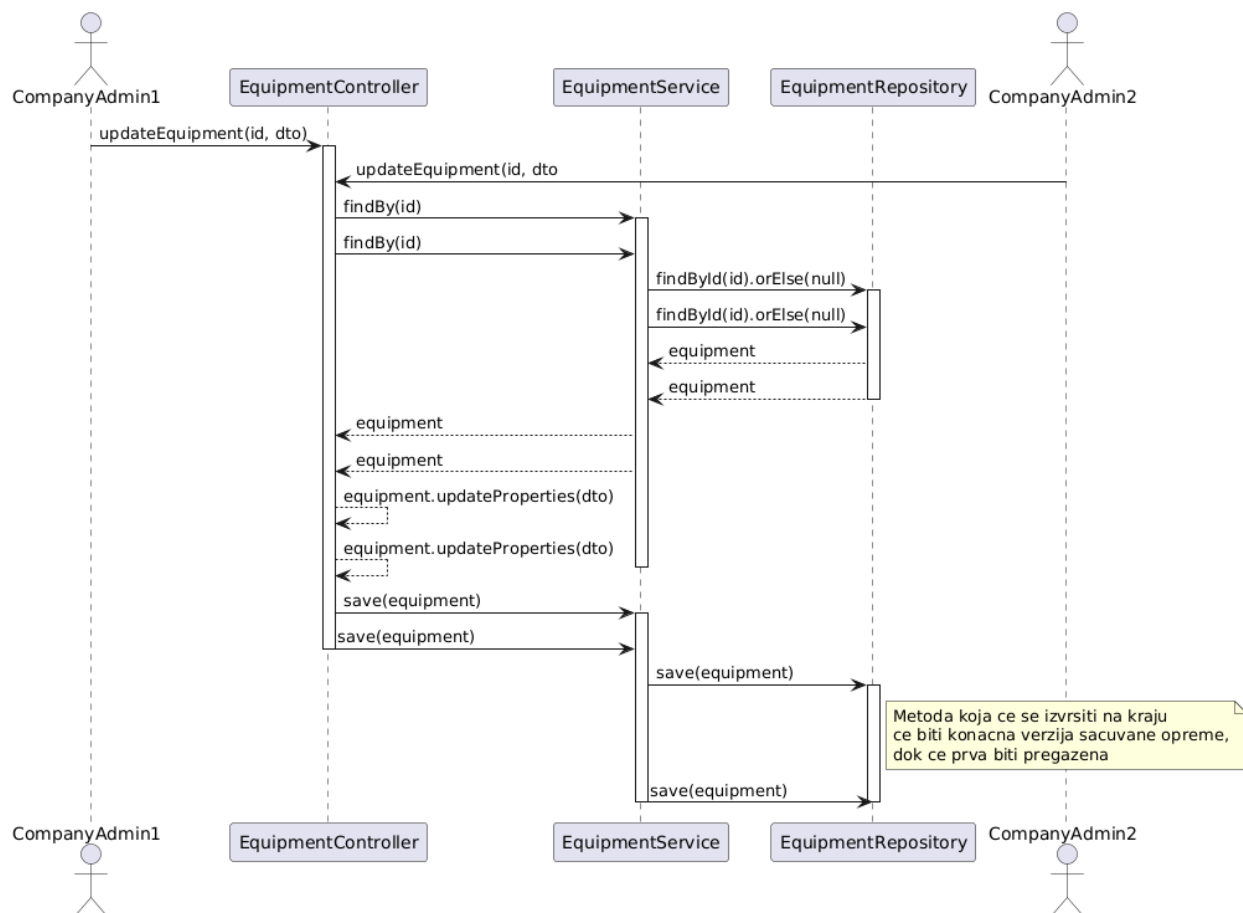
Tokom izvršavanja ove metode sva oprema koja pripada rezervaciji koja je prva došla biće pesimistički zaključana, odgovarajućom anotacijom iznad metode u repozitorijumu, tako da samo ta funkcija može pristupiti i menjati njeno stanje. Ovako je očuvana konzistentost stanja opreme u bazi podataka. Ukoliko nema dovoljno opreme na stanju za rezervaciju, korisnik nije u stanju da preuzme opremu.

Konflikt je testiran tako što je dodato dugme koje će gotovo istovremeno označiti sve rezervacije preuzetim i poslati zahteve ka backend-u dovoljno brzo da bi došlo do izuzetka. Jedan zahtev će proći dok će ostali izazvati `PessimisticLockingFailure` izuzetak koji se hvata i šalje korisniku odgovorajuća poruka u vidu iskačućeg prozora.

Napomena: Ovaj konflikt ne bi trebalo biti moguće izazvati jer nije dozvoljeno kreiranje preklapajućih termina. Konflikt je u ovom primeru izazvan veštačkim kreiranjem podataka, ubacivanjem preklapajućih rezervacija direktno u bazu.

Scenario 3:

Dva administratora kompanije ažuriraju istu instancu opreme u istom trenutku



Ovaj problem rešen je optimističnim zaključavanjem reda tabele opreme. Dodato je version polje sa anotacijom `@Version` u entitet Opreme . Ovo polje prati verziju svake pojave entiteta I prilikom svakog ažuriranja verzija se automatski inkrementira za jedan. Pre ažuriranja se proverava verzija opreme koja je dobavljena sa front end-a sa onom iz baze, ukoliko su verzije različite back end baca `CustomOptimisticLockingException`. Front end onda prikazuje grešku i automatski dobavlja novu verziju opreme iz baze i prikazuje je korisniku.

U situaciji da jedan admin briše opremu dok drugi istovremeno želi da je ažurira, back-end baca `EquipmentNotFoundException` i korisniku prikazuje odgovarajuću poruku.

Ovaj problem je testiran logovanjem jednog administratora kompanije u normalnom pretraživaču, dok je drugi administrator iste kompanije ulogovan preko incognito prozora istog pretraživača.

