

Konkurentni pristup bazi podataka

Konfliktne situacije

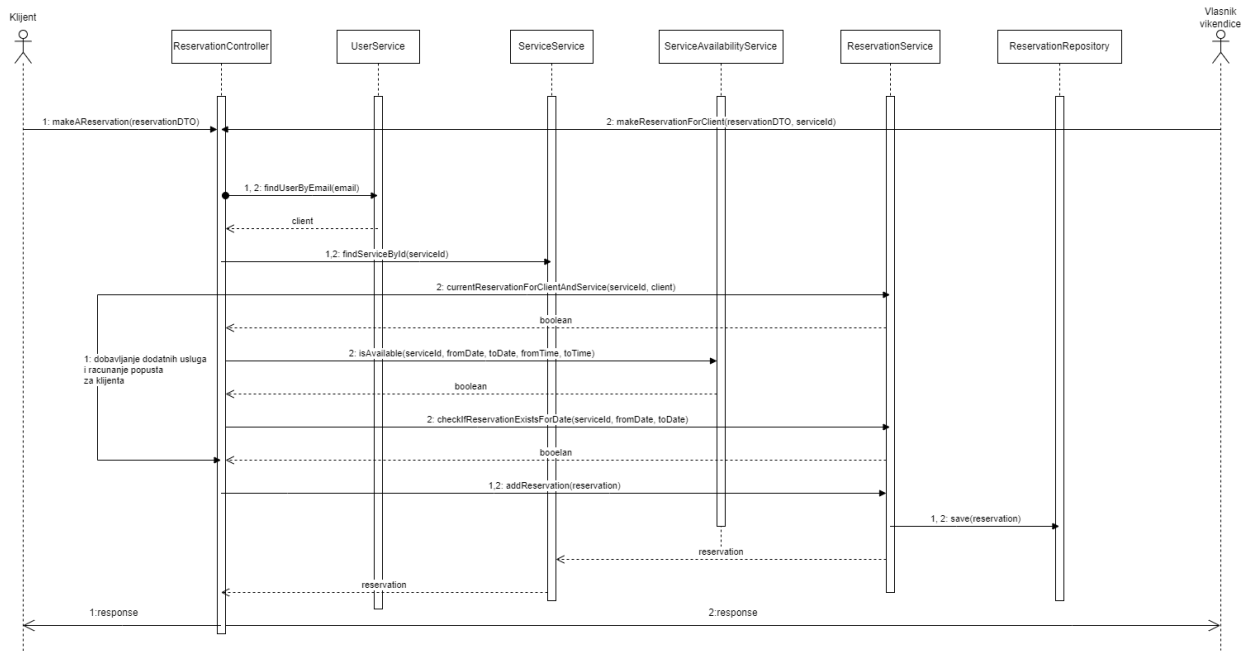
Student: Matija Zarić

Broj indeksa: SW 24/2019

Situacija 1:

Vlasnik vikendice/broda pokušava da rezerviše svoju vikendicu/brod za nekog klijenta dok istovremeno drugi klijent pokušava da rezerviše njegovu uslugu za sebe. Posledica može biti da se kreiraju dve rezervacije u istom terminu.

Tok akcije: Vlasnik vikendice/broda istovremeno pritiska dugme za rezervaciju svog entiteta za nekog klijenta dok drugi klijent pritiska dugme za rezervaciju istog entiteta u isto vreme. Na server stižu oba zahteva, a ako oba prođu sve provere koje se rade pre upisa, obe rezervacije će biti upisane.



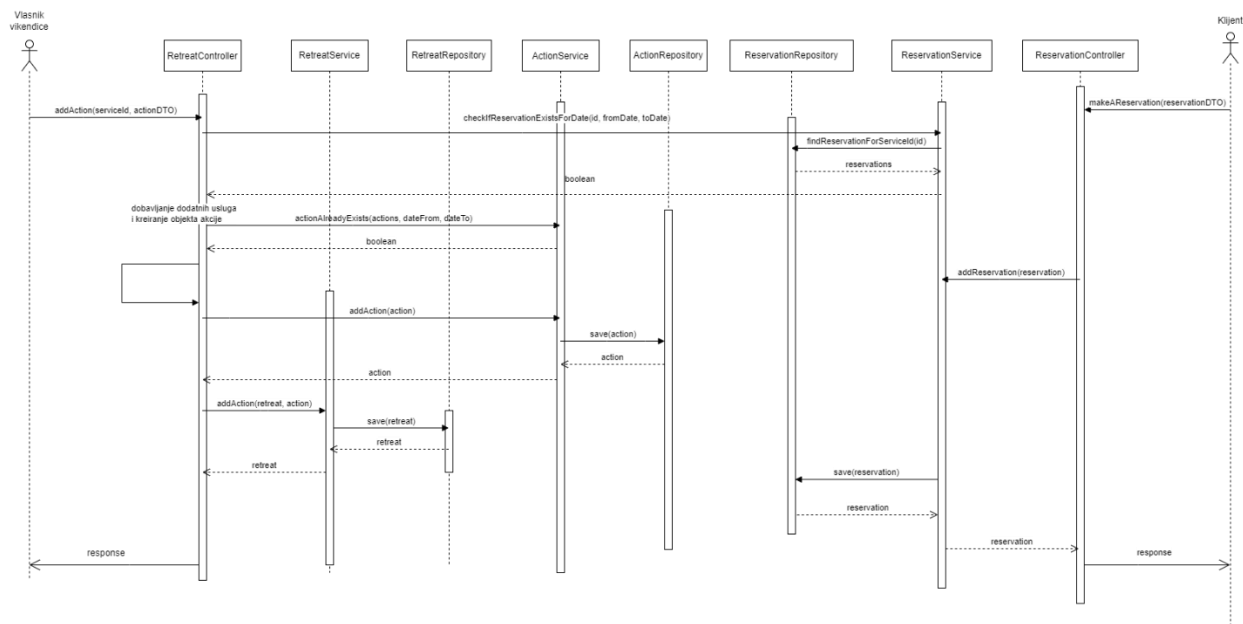
Rešenje: Ovaj problem je rešen korišćenjem optimističkog zaključavanja, u klasi *Service*, koja je bazna klasa za klase *Retreat* i *Ship*, dodat je atribut *version* koji modeluje trenutnu verziju objekta vikendice ili broda. Takođe u klasi *Service* je dodat atribut *numOfReservations* koji se uvećava prilikom kreiranja nove rezervacije i time izaziva promenu verzije objekta vikendice ili

broda, čime se postiže optimističko zaključavanje tog objekta, pa zahtev koji je drugi krenuo da doda rezervaciju, ne uspeva jer pokušava da izmeni staru verziju objekta vikendice/broda i dobija *OptimisticLockingFailureException*.

Situacija 2:

Vlasnik vikendice/broda pokušava da napravi akciju za vikendicu/brod istovremeno kada neki klijent pokušava da rezerviše taj entitet. Posledica može biti kreiranje akcije za period kada je klijent kreirao rezervaciju.

Tok akcije: Vlasnik vikendice/broda istovremeno pritiska dugme za kreiranje akcije kao i klijent koji pritiska dugme za rezervisanje te vikendice/broda za isto vreme kao i akcija. Na server stižu oba zahteva, a ako oba prođu sve provere koje se rade pre upisa, biće upisana i akcija i rezervacija.

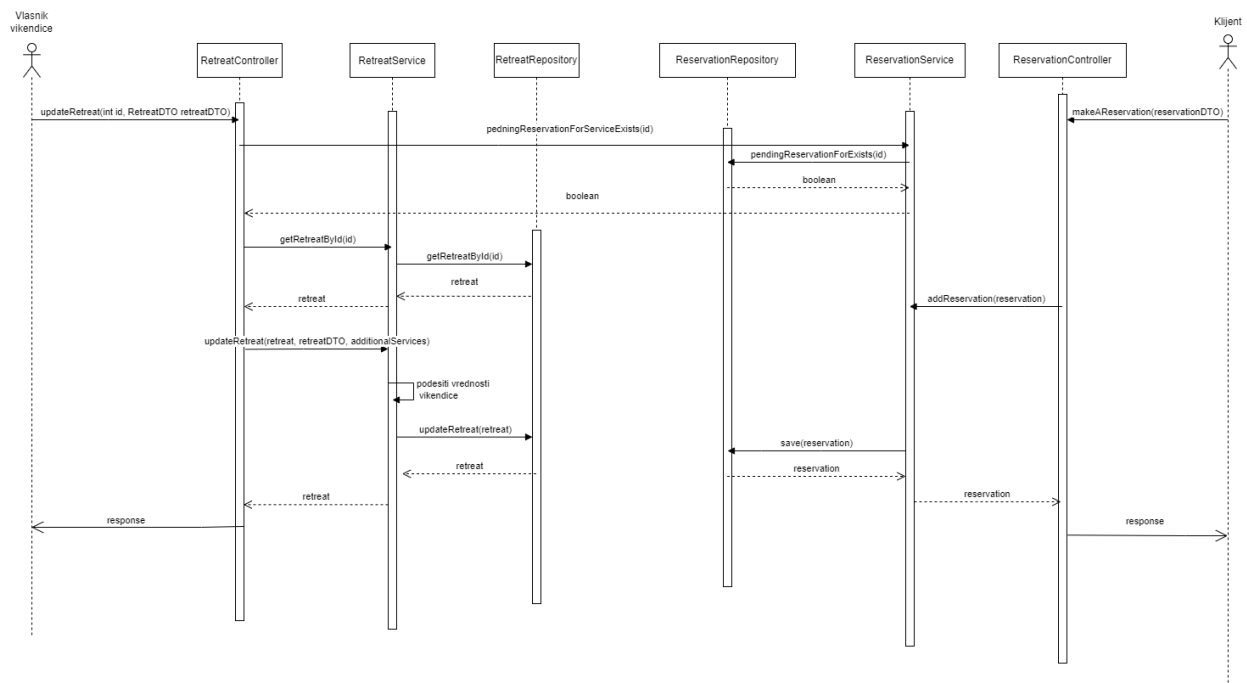


Rešenje: Ovaj problem je rešen korišćenjem optimističkog zaključavanja, u klasi *Service*, koja je bazna klasa za klase *Retreat* i *Ship*, dodat je atribut *version* koji modeluje trenutnu verziju objekta vikendice ili broda. Takođe u klasi *Service* je dodat atribut *numOfReservations* koji se uvećava prilikom kreiranja nove akcije i time izaziva promenu verzije objekta vikendice ili broda čime se postiže optimističko zaključavanje tog objekta, pa zahtev koji je krenuo da doda rezervaciju, ne uspeva jer pokušava da izmeni staru verziju objekta vikendice/broda i dobija *OptimisticLockingFailureException*.

Situacija 3:

Klijent pokušava da rezerviše vikendicu dok istovremeno vlasnik pokušava da modifikuje, ili obriše vikendicu. Posledica akcije može biti da je klijent rezervisao vikendicu koja je izbrisana ili joj je izmenjeno neko bitno svojstvo poput cene ili neke dodatne usluge.

Tok akcije: Vlasnik vikendice menja podatak o dodatnim uslugama vikendice dok istovremeno klijent pritiska dugme za rezervaciju te iste vikendice. U bazi se upisuje izmena usluga koje pruža vlasnik, a istovremeno se upisuje rezervacija koja ima neku dodatnu uslugu koju vlasnik potencijalno više ne nudi.

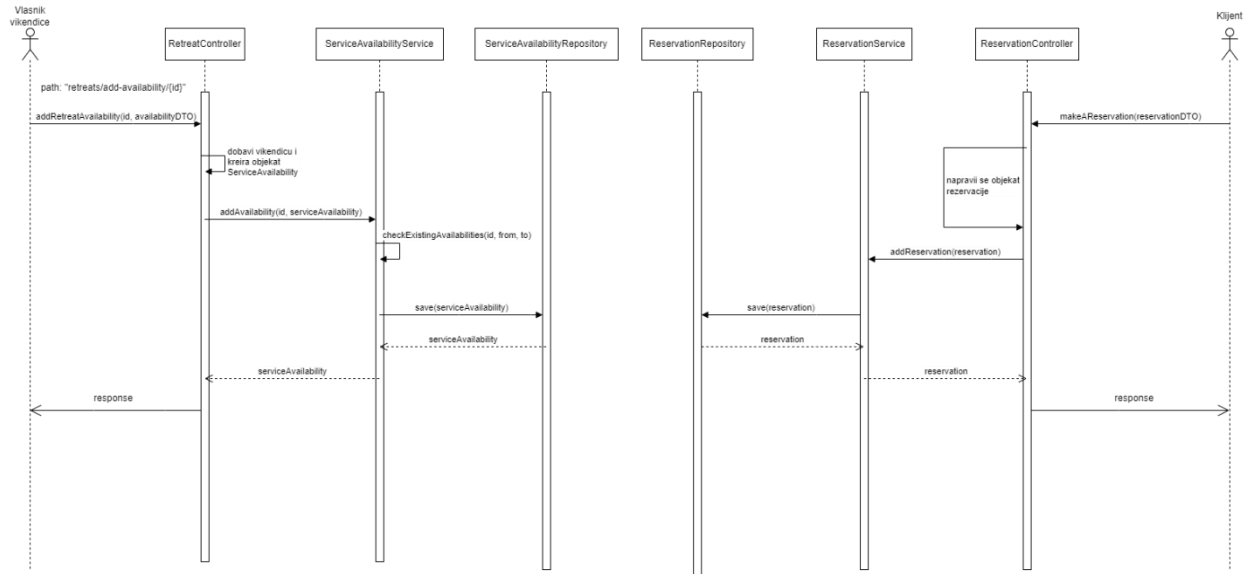


Rešenje: Ovaj problem je rešen korišćenjem optimističnog zaključavanja, u klasi *Service*, koja je bazna klasa za klasu *Retreat*(vikendica), dodat je atribut *version* koji modeluje trenutnu verziju objekta vikendice. Kada se izmeni vikendica vrednost *version* atributa se povećava i ako se pokuša kreiranje rezervacije za stariju verziju vikendice dobiće se *OptimisticLockingFailureException*. Takođe u klasi *Service* dodat je atribut *numOfReservations* koji se menja kada se dodaje nova rezervacija da bi se mogao zaključati servis za koji se dodaje vikendica.

Situacija 4:

Klijent pokušava da rezerviše vikendicu dok istovremeno vlasnik pokušava da doda termin dostupnosti za istu. Posledica akcije može biti da klijent rezerviše vikendicu u potencijalno nedostupnom periodu koji je korisnik definisao.

Tok akcije: Prethodno nije definisan period dostupnosti za vikendicu, pa se smatra da je vikendica uvek dostupna. Klijent pokušava da rezerviše vikendicu, dok istovremeno vlasnik dodaje termin dostupnosti vikendice, pa se može desiti da se vikendica rezerviše u periodu za koji je vlasnik naveo da nije dostupan.



Rešenje: Ovaj problem je rešen korišćenjem optimističnog zaključavanja, u klasi *Service*, koja je bazna klasa za klasu *Retreat* (vikendica), dodat je atribut *version* koji modeluje trenutnu verziju objekta vikendice, takođe je dodat atribut *numOfAvailabilities*. Kada se doda period dostupnosti za vikendicu poveća se vrednost atributa *numOfAvailabilities* što dovodi do ažuriranja verzije vikendice i njenog atributa *version*, pa klijent ne može upisati rezervaciju za stariju verziju vikendice, odnosno dobija *OptimisticLockingFailureException*.