

Student 2

Konfliktne situacije iz specifikacije:

1. vlasnik vikendice/broda instruktor ne moze da napravi rezervaciju u isto vreme kad i drugi klijent
2. vlasnik vikendice/broda ne moze da napravi akciju u isto vreme kad i drugi klijent vrši rezervaciju postojećeg entiteta
3. vlasnik vikendice/broda ne moze da mijenja vikendicu/brod u isto vreme kad i drugi klijent vrši rezervaciju postojećeg entiteta
4. vlasnik vikendice/broda ne moze da obrise akciju u isto vreme kad i drugi klijent vrši rezervaciju postojećeg entiteta

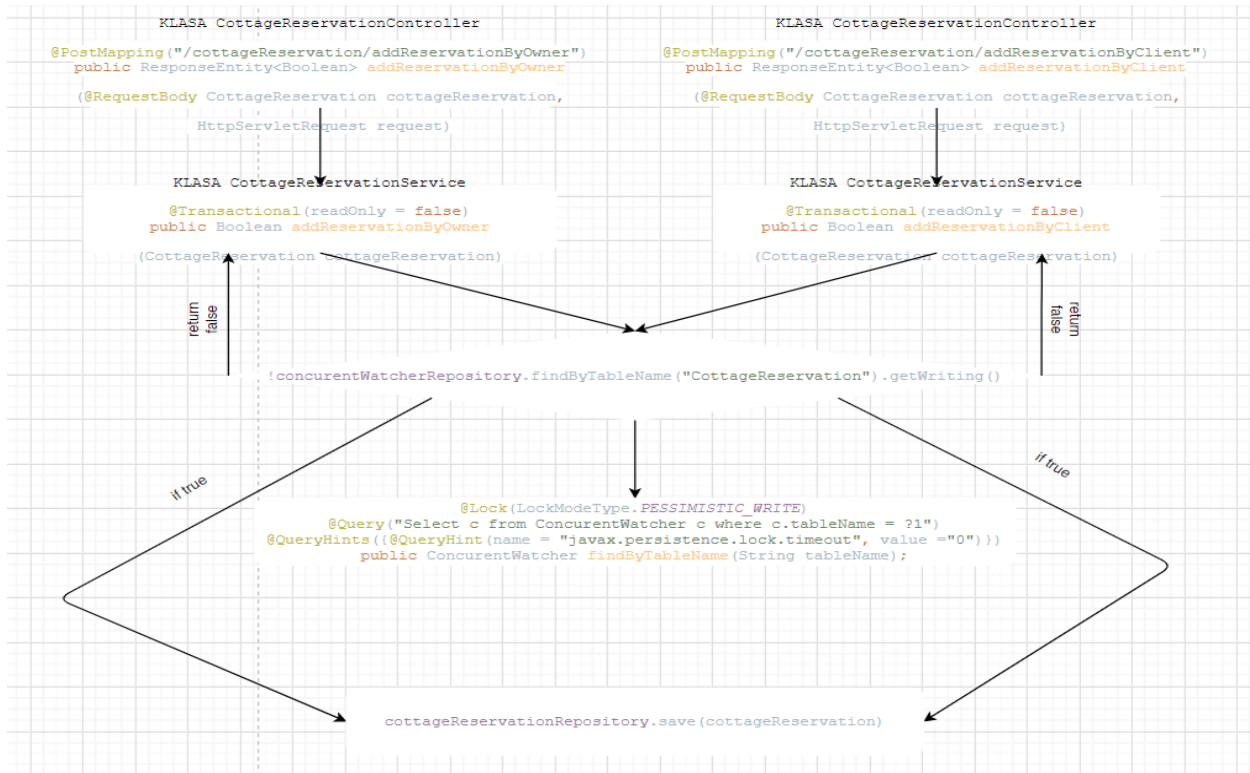
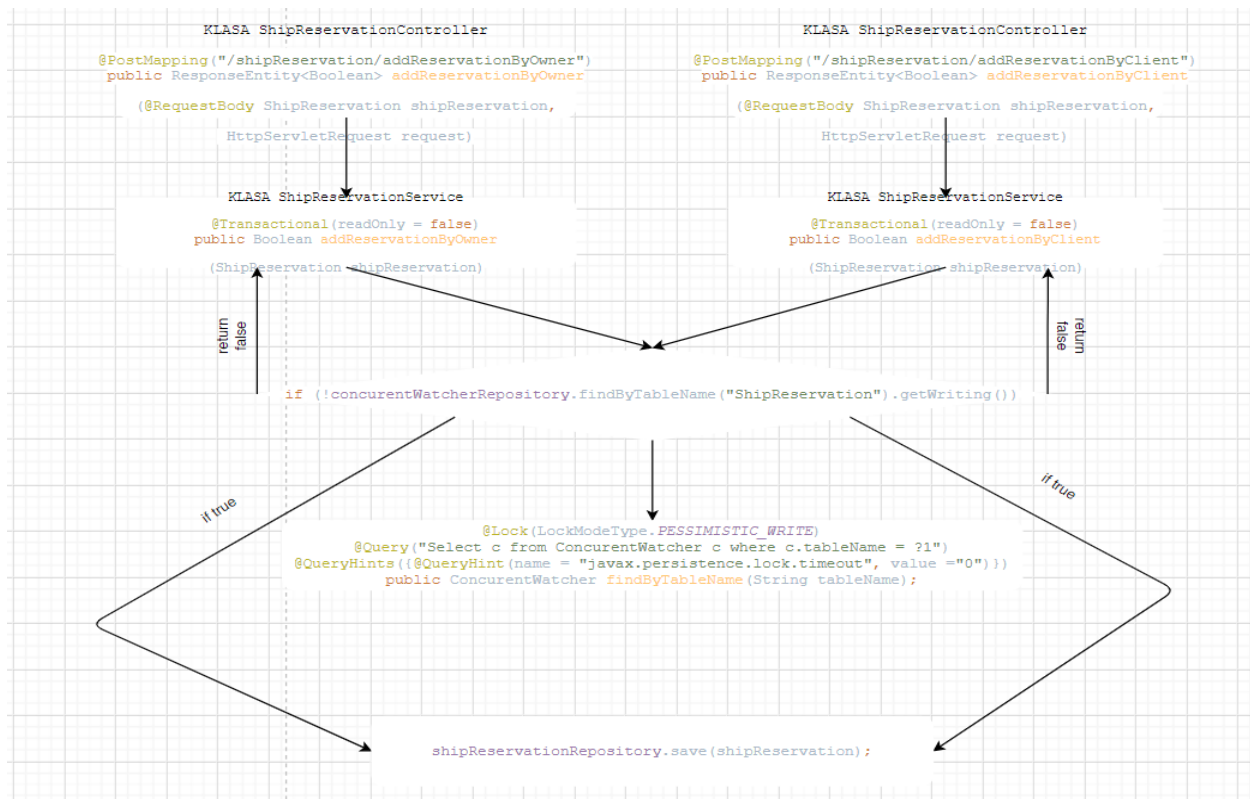
1. Ako bi se desilo da vlasnik i klijent naprave rezervaciju u isto vrijeme istog entiteta u bazi bi se moglo desiti da se sacuvaju obje rezervacije iako im se termini preklapaju.

2. Ako bi se desilo da u isto vrijeme za isti entitet vlasnik napravi akciju i klijent napravi rezervaciju u bazi bi se moglo desiti da se sacuvaju i akcija i rezervacija iako im se termini preklapaju.

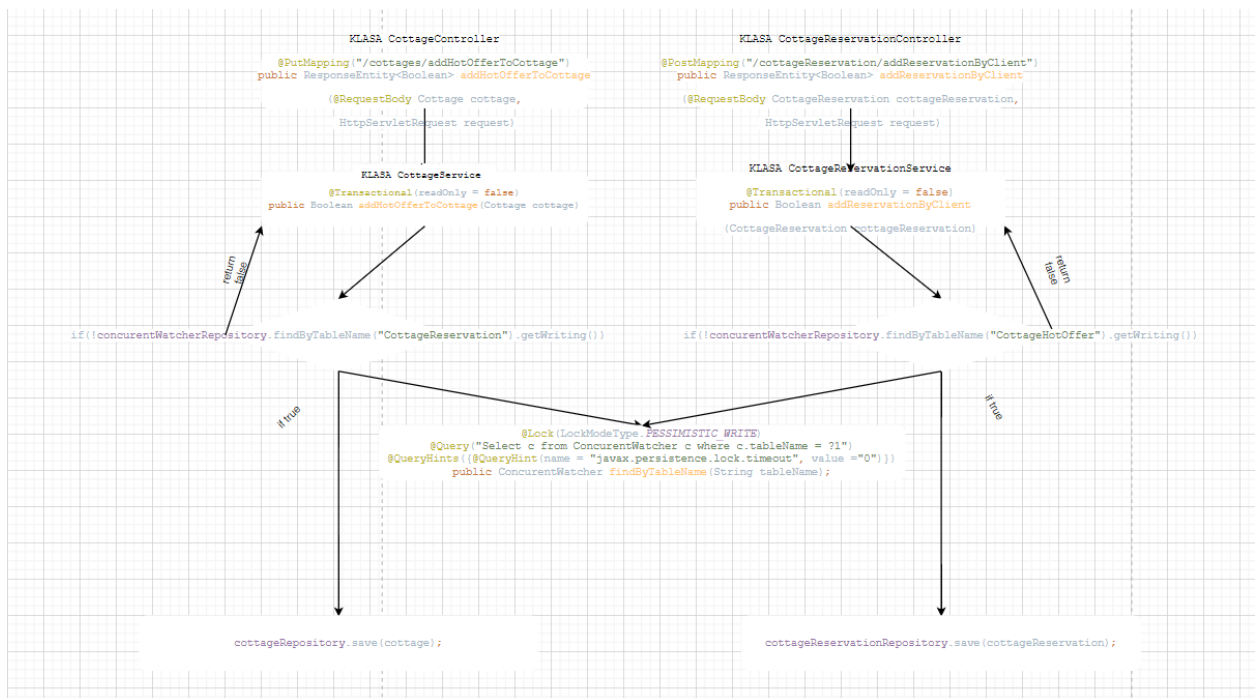
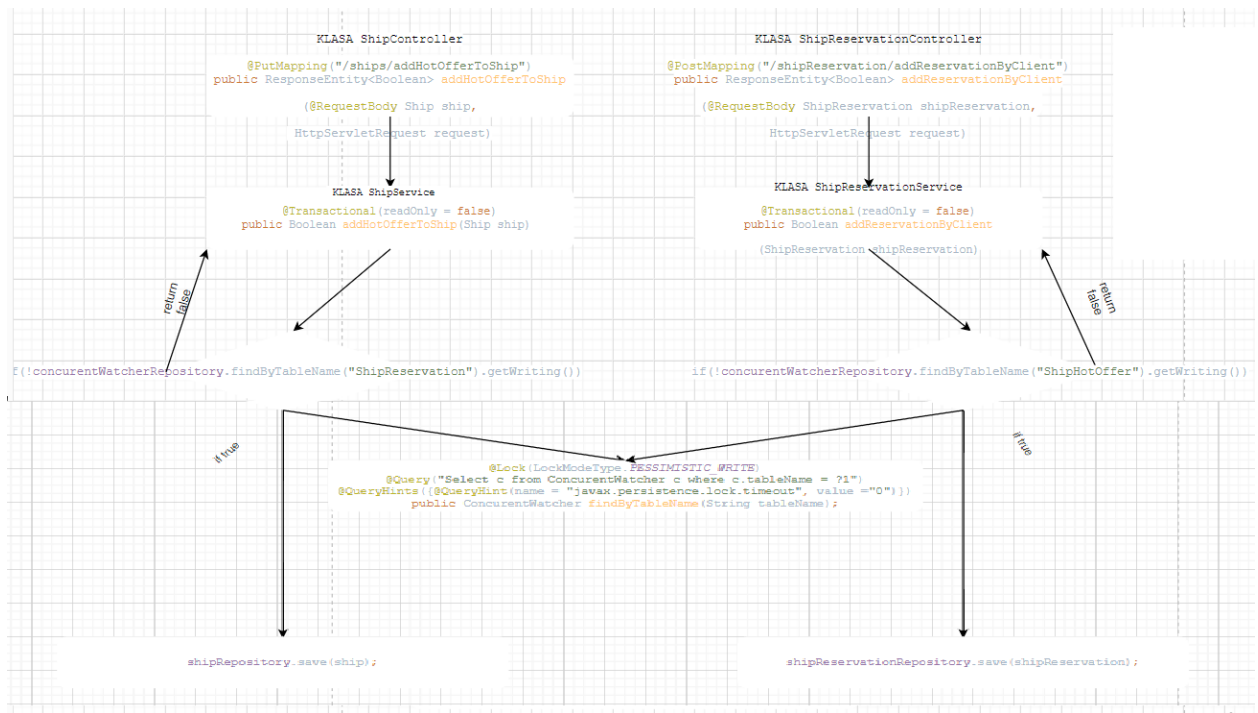
3. Ako bi se desilo da u isto vrijeme za isti entitet vlasnik napravi promjene i klijent napravi rezervaciju u bazi bi se moglo desiti da se sacuvaju i promene entiteta i rezervacija iako im se podaci ne podudaraju (rezervacija bi mogla da ima neku uslugu koju entitet ne pruza, razlika u ceni).

4. Ako bi se desilo da u isto vrijeme za isti entitet vlasnik izbrise akciju i klijent napravi rezervaciju u bazi bi se moglo desiti da se sacuva rezervacija iako je vlasnik uspjesno izbrisao akciju.

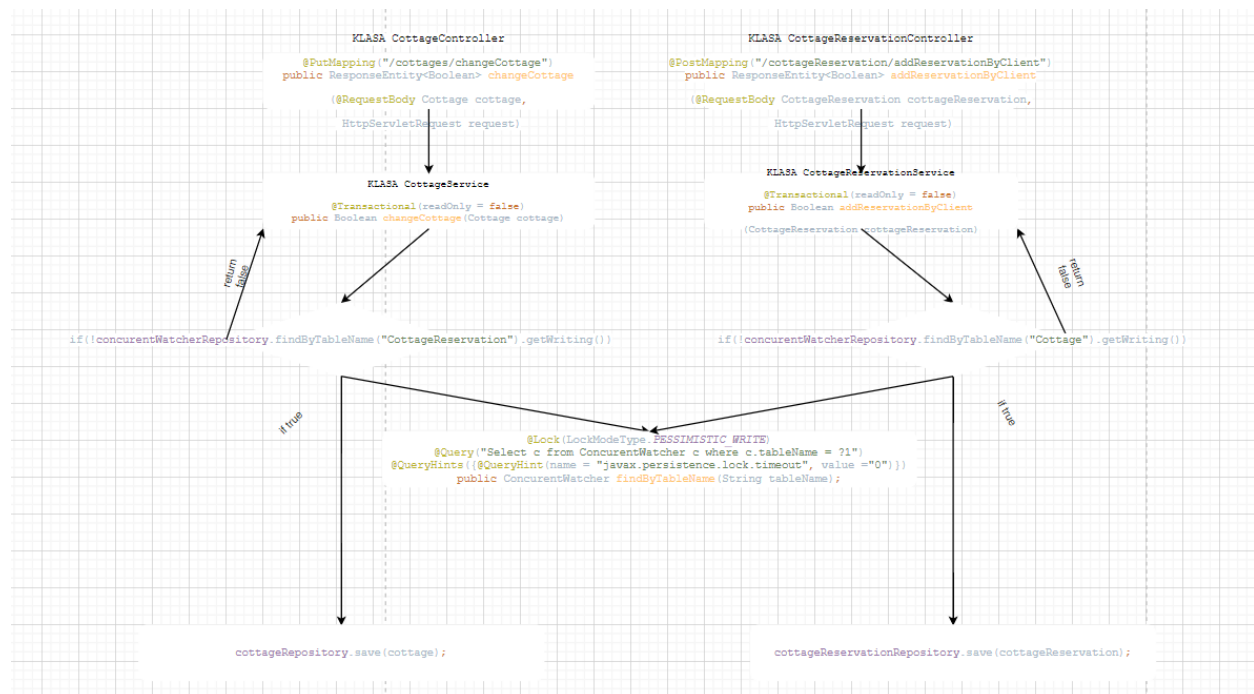
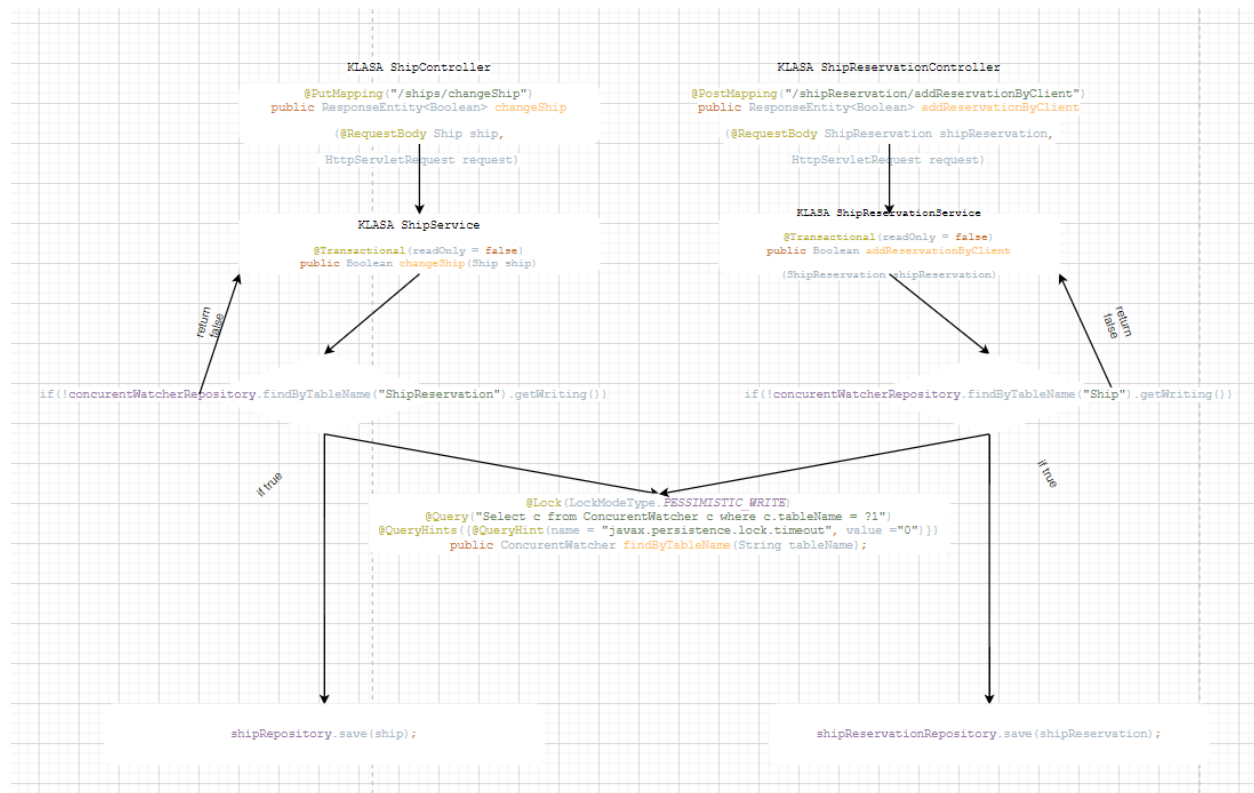
1.



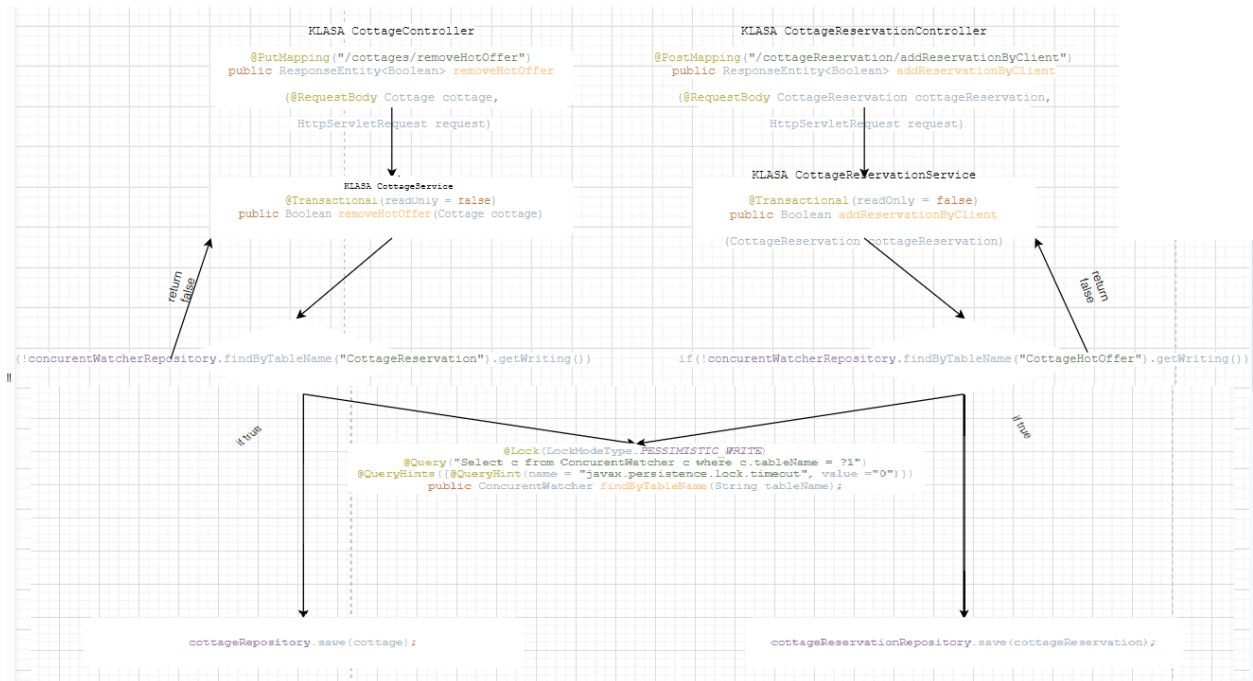
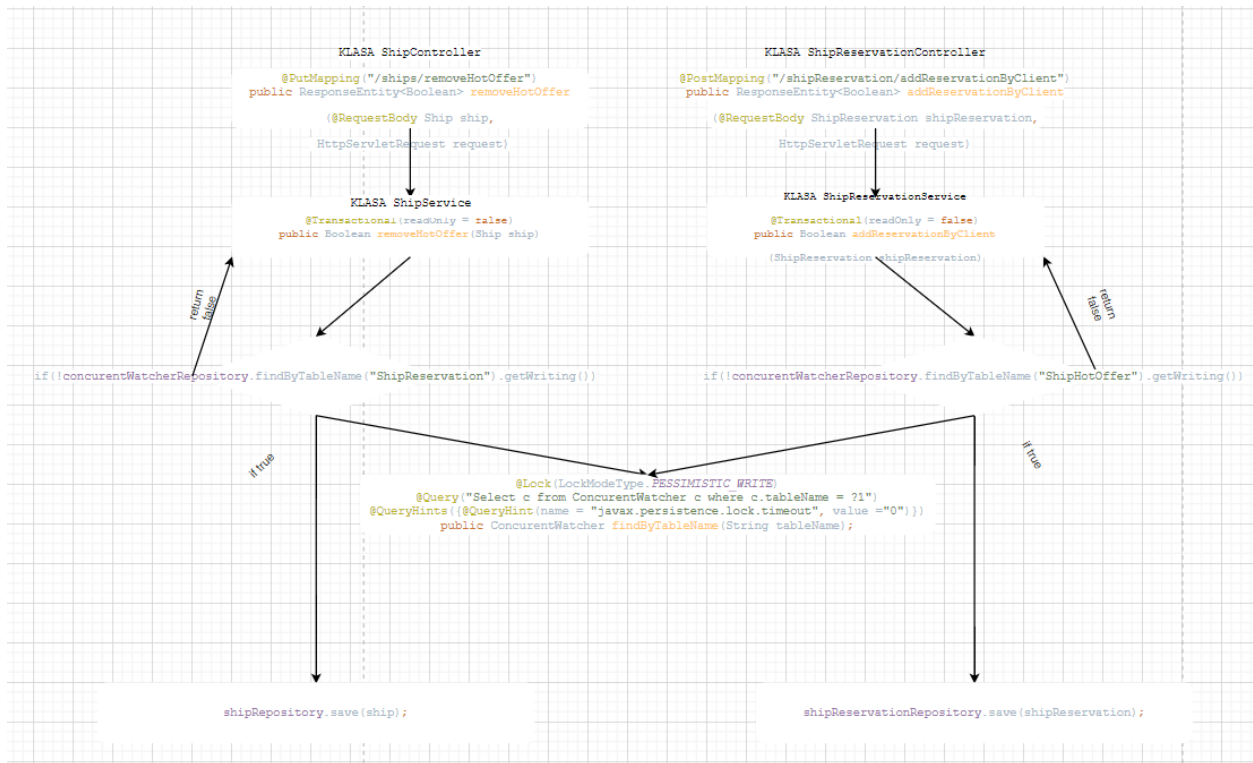
2.



3.



4.



Sve 4 konfliktne situacije su riješene na sličan način. Napravio smo entitet i tabelu ConcurrentWatcher u kojoj imamo po jedan red za tabele Ship, Cottage, CottageHotOffer, ShipHotOffer, CottageReservation, i ShipReservation.

Svaki od navedenih redova sadrži polje isWriting koje postavim na true kada neka metoda iz servisa će da vrši write te tabele. Kada ta metoda završi korišćenjem tabele to polje isWriting vratim na false.

Ako već neka metoda iz servisa vrši write neke tabele, ostale metode koje mogu izazvati konfliktnu situaciju neće moći da izvrše write te iste tabele.

U repozitorijumu ConcurrentWatcherRepository koristio sam :

```
@Lock(LockModeType.PESSIMISTIC_WRITE)
```

```
@QueryHints({@QueryHint(name = "javax.persistence.lock.timeout", value = "0")})
```