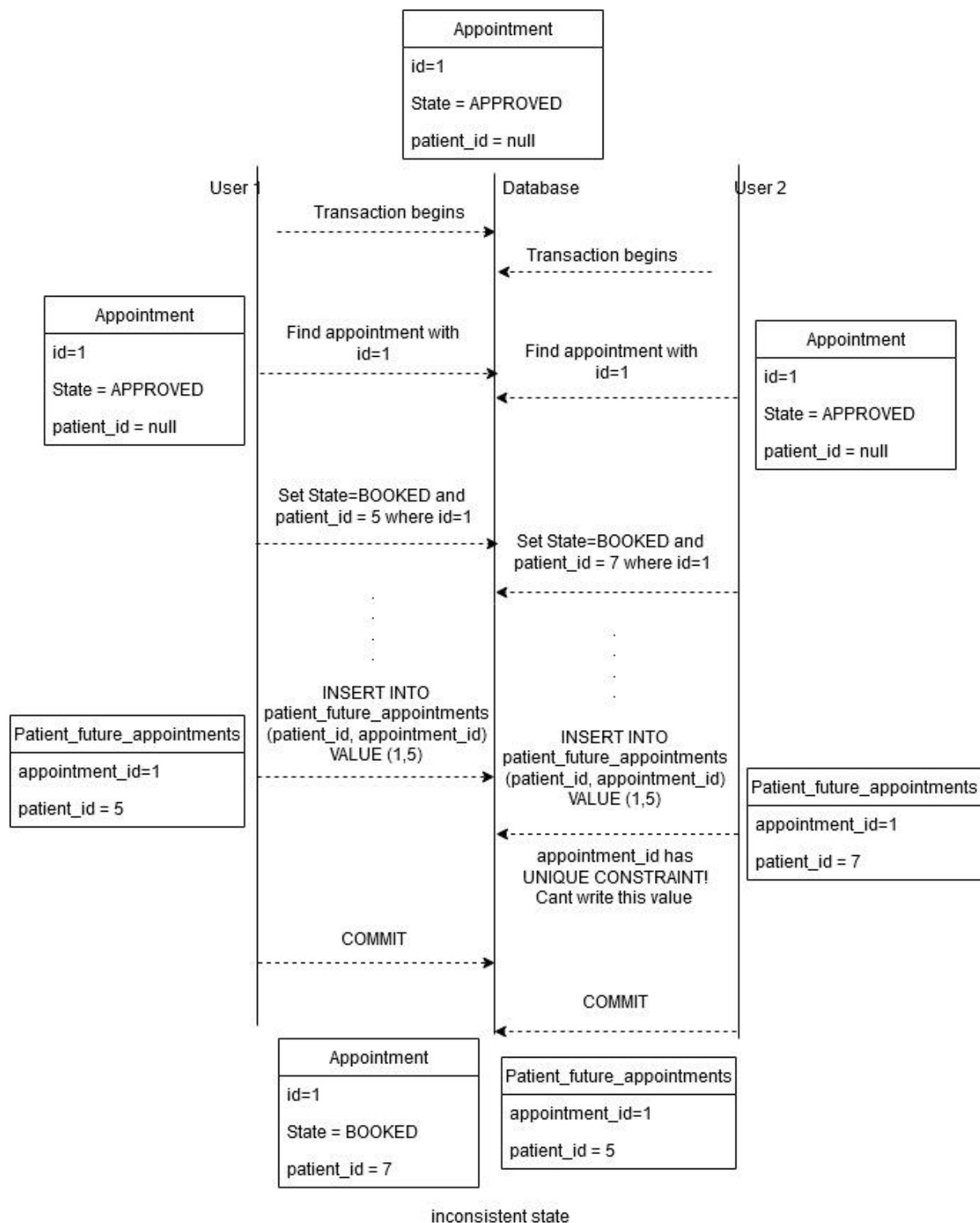


KONKURENTAN PRISTUP RESURSIMA U BAZI

STUDENT 1: Veljko Maksimović

Prva konfliktna situacija: **Rezervisanje unapred definisanih termina.** Proces rezervisanja unapred definisanih termina sastoji se iz:

1. Provere da li termin sa datim id-em postoji u bazi, i da li je i dalje slobodan.
2. Promene statusa termina u "ZAUZET" i upisivanja id-a pacijenta u termin
3. Iz tabele koja spaja klinike sa predefinisanim terminima treba izbaciti torku koja se odnosi na malopre zakazan termin
4. I u tabelu koja spaja pacijente sa terminima na koje treba da dođu unosimo novu torku koja se sastoji od id-a pacijenta i id-a pregleda.



Opis situacije u kojoj može da dođe do konflikta:

- Dva pacijenta "istovremeno" šalju zahtev za rezervisanje termina. Zahtev koji je došao drugi prolazi proveru u koraku 1) pre nego što je prvi zahtev stigao da promeni njegovo stanje na "ZAUZET" u koraku

2). Prvi zahtev upisuje svoj id u polje pacijenta, ali ga drugi zahtev odma nakon toga prepravlja. Na kraju prvi zahtev u tabelu koja spaja pacijente sa predstojećim pregledima upisuje id svog pacijenta i id termina. Kada drugi zahtev dodje do 4. koraka baza mu ne dozvoljava da u malopre pomenutu tabelu upiše id svog pacijenta zajedno sa id-em termina, jer je na id termina postavljen UNIQUE constraint, a obeležje sa tom vrednošću se već nalazi u tabeli jer ga je upisao zahtev broj jedan. Na kraju smo ostali sa bazom koja u tabeli "pacijent-termin" ukazuje na to da pregled koji je u pitanju pripada jednom pacijentu, a ako pogledamo u tabelu pregleda, taj isti pregled kod sebe u polju "id pacijenta" referencira pacijenta iz drugog zahteva: Baza nije konzistentna! Da se u trenutku kada user 2 dobije unique constraint gresku primeni rollback na sve operacije koje je on uradio u okviru ove transakcije, baza bi bila konzistentna jer bi se polje patient_id u Appointment torki vratilo na vrednost 5.

Druga konfliktna situacija: **Odgovor na zahtev za odmor (Vacation Request) od strane kliničkog admina.** Proces odgovaranja na zahtev sastoji se iz:

1. Provera da li zahtev sa datim id-em postoji u bazi, i da li je već neko odgovorio na isti
2. Promena polj "accepted" na true ili false
3. Zahtev se briše iz tabele koja povezuje admina klinike sa zahtevima na koje tek treba da odgovore.
4. Ako je zahtev odobren, ubacuje se nova torka u tabelu koja spaja zaposlenog sa odobrenim godišnjim odmorom.

Opis konfliktne situacije:

-Dva admina iste klinike "istovremeno" odgovaraju na zahtev za godišnji odmor: jedan odobrava zahtev, a drugi odbija. I jedan i drugi zahtev stižu do prvog koraka, proveravaju da li se zahtev za odmor nalazi u bazi, i da još niko nije odgovorio na isti. Nakon toga prvi zahtev odobrava godišnji odmor i briše ga iz tabele koja spaja admina sa zahtevima na koje još nije odgovorio. Drugi zahtev pravi ista dva koraka, samo što prepravlja polje "accepted" na fals. Za kraj, prvi zahtev, koji je odobrio godišnji odmor u tabelu koja spaja zaposlene sa odobrenim godišnjim odmorom upisuje novu torku, gre će id "Vacation Requesta" ukazivati na zahtev za godišnji odmor koji nije odobre- nastaje konfliktna situacija.

