

Univerzitet u Banjoj Luci

Prirodno-matematički fakultet

Matematika i informatika – opšti smjer

Baze podataka 2

Baza podataka apotekarske ustanove

april 2023.

Ivana Vujić

Sadržaj

1. Uvod	1
2. EER konceptualni model	2
3. Relacioni model.....	3
4. EER fizički model	5
5. Trigeri	6
6. Transakcije	7

1. Uvod

Projektuje se baza podataka za potrebe rada apotekarske ustanove.

Za svakog zaposlenog apotekara čuvaju se ime, prezime, telefon, email, plata i zvanje.

Svaki šef službe je ujedno i apotekar.

Apotekar može da vrši prodaju, prilikom koje izdaje račun.

Svaki račun ima broj, datum, informaciju o apotekaru koji ga je izdao, te listu prodatih lijekova i pratećih količina.

Za svaki lijek čuvaju se naziv, proizvođač, stanje, bar-kod, cijena, lista i šifra fonda.

Takođe, za svaki lijek čuva se informacija o njegovom sastavu, odnosno supstancama koje ga čine, te pratećim količinama.

Za svaku supstancu čuvaju se šifra i naziv.

Za neke lijekove postoje i zamjenski lijekovi.

Lijekovi mogu da se izdaju na recept, te se za svaki recept prate broj, datum, ljekar, dijagnoza, da li je lijek izdat ili ne, informacija o osiguraniku, te informacija o zdravstvenoj ustanovi.

Za svakog osiguranika čuvaju se id osiguranika, id kartice, ime, prezime, telefon, email i jmbg.

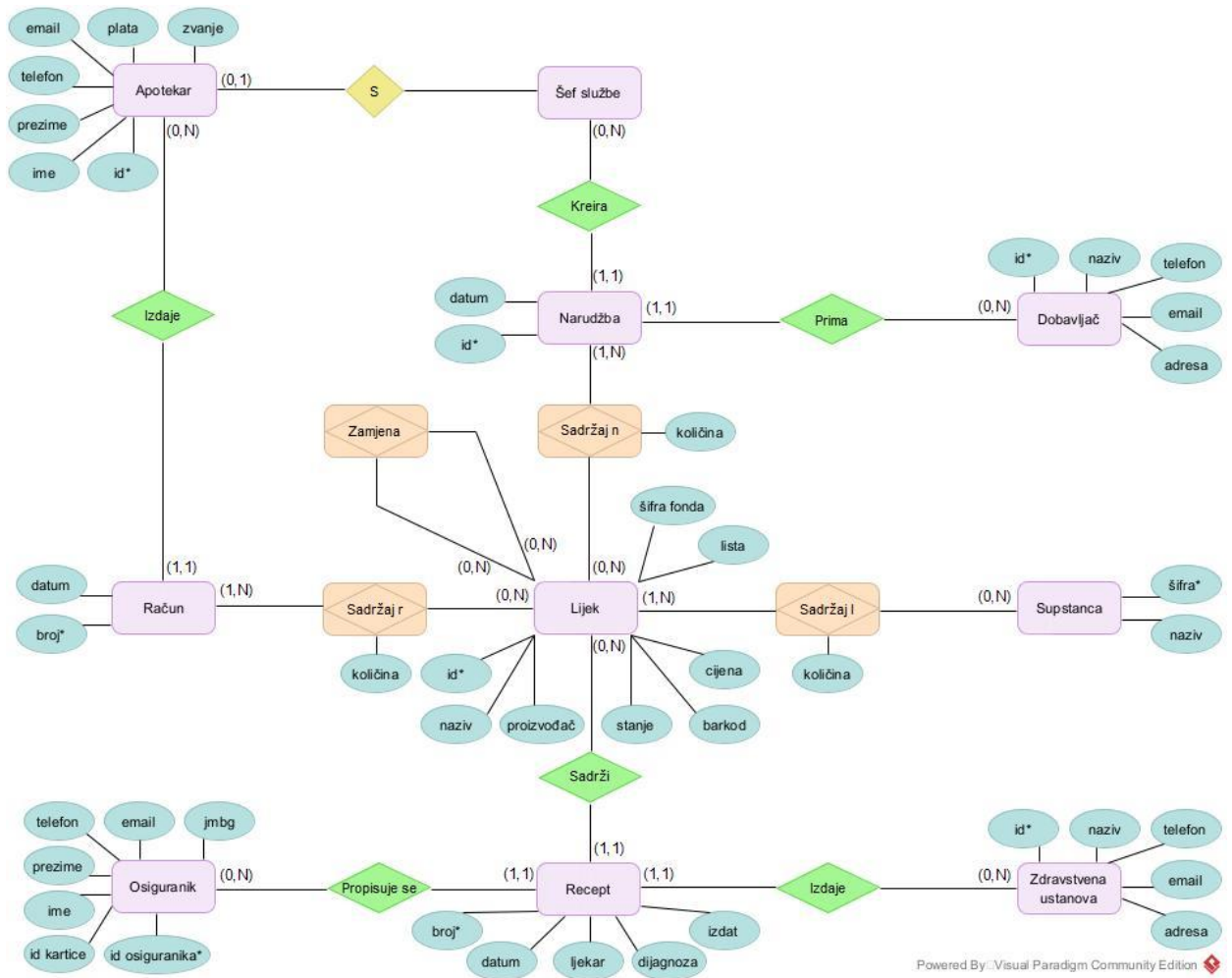
Za svaku zdravstvenu ustanovu čuvaju se naziv, telefon, email i adresa.

Šef službe može da naručuje lijekove, prilikom čega kreira narudžbu.

Za svaku narudžbu prate se datum, informacija o šefu službe koji ju je kreirao, te informacija o dobavljaču od koga se naručuje.

Za svakog dobavljača čuvaju se naziv, telefon, email i adresa.

2. EER konceptualni model



3. Relacioni model



Apotekar

Apotekar(id, ime, prezime, telefon, email, plata, zvanje)



Šef službe

SefSluzbe(apotekar_id)

SefSluzbe[apotekar_id] \subseteq Apotekar[id]



Narudžba

Narudzba(id, datum, sef_sluzbe_apotekar_id, dobavljac_id)

Narudzba[sef_sluzbe_apotekar_id] \subseteq SefSluzbe[apotekar_id]

Narudzba[dobavljac_id] \subseteq Dobavljac[id]



Dobavljač

Dobavljac(id, naziv, telefon, email, adresa)



Račun

Racun(broj, datum, apotekar_id)

Racun[apotekar_id] \subseteq Apotekar[id]



Lijek

Lijek(id, naziv, proizvođač, stanje, barkod, cijena, lista, sifra_fonda)



Supstanca

Supstanca(sifra, naziv)



Osiguranik

Osiguranik(id_osiguranika, id_kartice, ime, prezime, telefon, email, jmbg)



Recept

Recept(broj, datum, ljekar, dijagnoza, lijek_id, osiguranik_id_osiguranika, zdravstvena_ustanova_id)

Recept[lijek_id] \subseteq Lijek[id]

Recept[osiguranik_id_osiguranika] \subseteq Osiguranik[id_osiguranika]

Recept[zdravstvena_ustanova_id] \subseteq ZdravstvenaUstanova[id]



Zdravstvena ustanova

ZdravstvenaUstanova(id, naziv, telefon, email, adresa)

Sadržaj narudžbe

SadržajNarudzbe(narudzba_id, narudzba_sef_sluzbe_apotekar_id,
narudzba_dobavljac_id, lijek_id, kolicina)

SadržajNarudzbe[narudzba_id] \subseteq Narudzba[id]

SadržajNarudzbe[narudzba_sef_sluzbe_apotekar_id] \subseteq
Narudzba[sef_sluzbe_apotekar_id]

SadržajNarudzbe[narudzba_dobavljac_id] \subseteq Narudzba[dobavljac_id]

SadržajNarudzbe[lijek_id] \subseteq Lijek[id]

Sadržaj računa

SadržajRacuna(racun_broj, racun_apotekar_id, lijek_id, kolicina)

SadržajRacuna [racun_broj] \subseteq Racun[broj]

SadržajRacuna [racun_apotekar_id] \subseteq Racun[apotekar_id]

SadržajRacuna [lijek_id] \subseteq Lijek[id]

Sadržaj lijeka

SadržajLijeka(lijek_id, supstanca_sifra, kolicina)

SadržajLijeka [lijek_id] \subseteq Lijek[id]

SadržajLijeka [supstanca_sifra] \subseteq Supstanca[sifra]

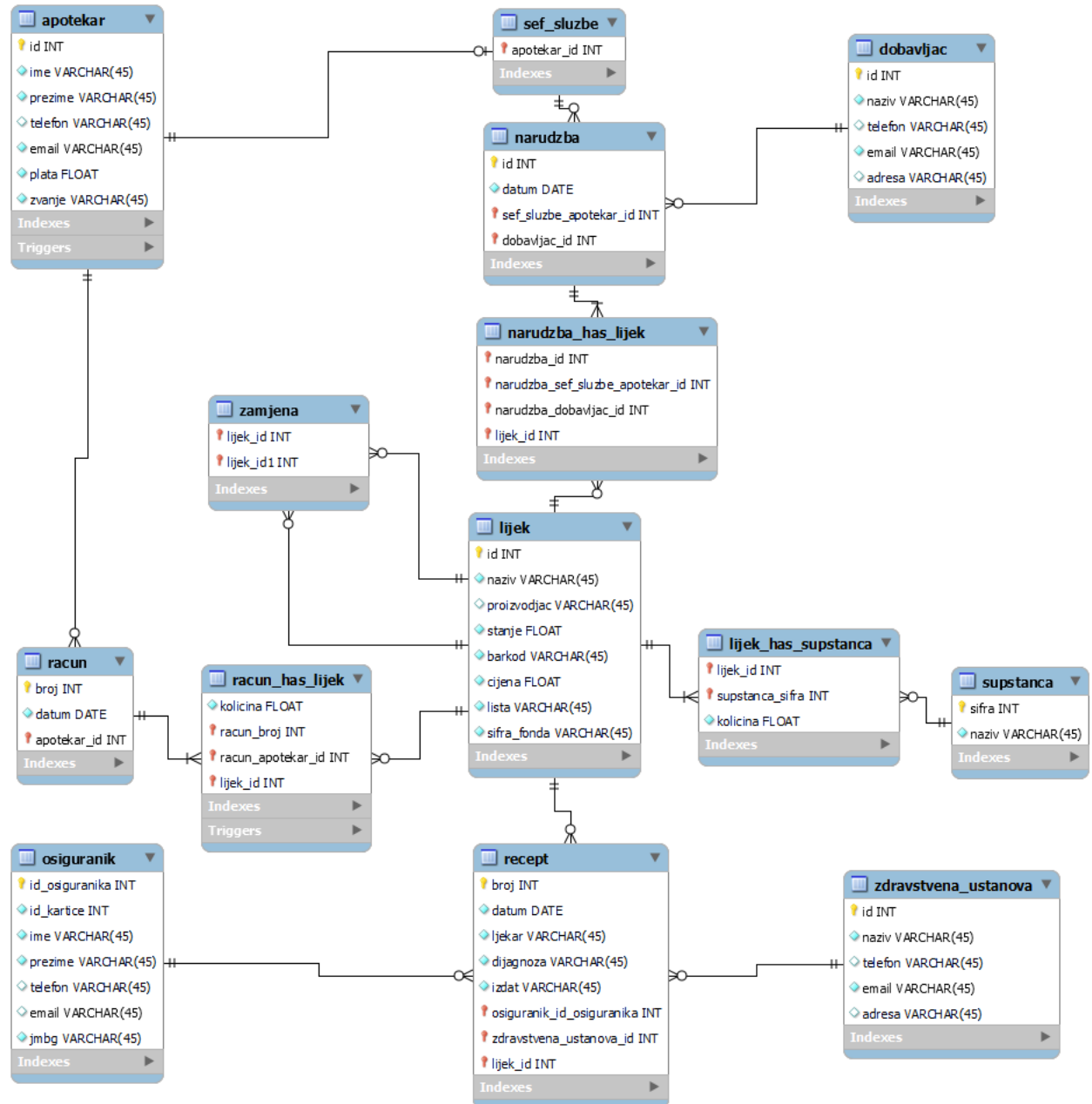
Zamjena

Zamjena(lijek_id, lijek_id1)

Zamjena [lijek_id] \subseteq Lijek[id]

Zamjena [lijek_id1] \subseteq Lijek[id]

4. EER fizički model



5. Trigeri

- ✚ Triger kojim se za svakog unesenog apotekara u tabelu *apotekar*, čija je plata veća ili jednaka 1800 i zvanje magistar farmacije, dodaje id tog apotekara u tabelu *sef_sluzbe*.

```
CREATE DEFINER = CURRENT_USER TRIGGER `apoteka`.`apotekar_AFTER_INSERT` AFTER  
INSERT ON `apotekar` FOR EACH ROW  
BEGIN  
    IF new.plata >= 1800 and new.zvanje = 'magistar farmacije' THEN  
        INSERT INTO `sef_sluzbe`(apotekar_id) VALUES(new.id);  
    END IF;  
END
```

- ✚ Triger kojim se za svaki uneseni lijek i količinu u tabelu *racun_has_lijek* stanje tog lijeka umanjuje za navedenu količinu u tabeli *lijek*.

```
CREATE DEFINER = CURRENT_USER TRIGGER `apoteka`.`racun_has_lijek_AFTER_INSERT`  
AFTER INSERT ON `racun_has_lijek` FOR EACH ROW  
BEGIN  
    UPDATE `lijek` SET stanje = stanje - new.kolicina WHERE lijek.id = new.lijek_id;  
END
```


6. Transakcije

- ✚ Procedura kojom se dodaje zamjena lijeku u tabelu *zamjena* pomoću transakcije. U slučaju da lijekovi nisu sa iste liste transakcija se poništava.

```
CREATE PROCEDURE `dodavanje_zamjene` (IN id1 INT, IN id2 INT)
BEGIN
  DECLARE lista1 VARCHAR(45);
  DECLARE lista2 VARCHAR(45);
  START TRANSACTION;
  SELECT `lijek`.`lista` INTO lista1 FROM `lijek` WHERE id = id1;
  SELECT `lijek`.`lista` INTO lista2 FROM `lijek` WHERE id = id2;
  INSERT INTO `zamjena`(lijek_id, lijek_id1) VALUES(id1, id2);
  INSERT INTO `zamjena`(lijek_id, lijek_id1) VALUES(id2, id1);
  IF lista1 = lista2 THEN
    COMMIT;
    SELECT 'Uspjesno dodavanje.';
  ELSE
    ROLLBACK;
    SELECT 'Lijekovi nisu sa iste liste.';
  END IF;
END
```