ŠTUDENT: Blaž Jerebic

Mentor: Robert malačič

Sistem ambulant

Projektna naloga pri OPB2

Vsebina

[1. UVOD 2](#_Toc11242862)

[2. NAMEN, CILJI TER METODE DELA 2](#_Toc11242863)

[3. MYSQL 3](#_Toc11242864)

[4. IZDELAVA PODATKOVNE BAZE 4](#_Toc11242865)

[**4.1** **Načrtovanje ER diagrama** 4](#_Toc11242866)

[**4.2** **Relacije med entitetami** 5](#_Toc11242867)

[**4.3** **View** 6](#_Toc11242868)

[5. POIZVEDBE 7](#_Toc11242869)

[**5.1** **Iz tabele napotnica izpišite podatke o napotnicah kjer idbolezen ni NULL** 7](#_Toc11242870)

[**5.2** **Iz tabele pacienti izpišite vse podatke o pacientih, ki so naroceni** 7](#_Toc11242871)

[6. Zaključek 8](#_Toc11242872)

[7. Viri 8](#_Toc11242873)

[Slika 1: Logo MySQL https://sl.wikipedia.org/wiki/MySQL#/media/Slika:MySQKL.jpg 3](#_Toc11242874)

[Slika 2: Končni ER diagram 4](#_Toc11242875)

[Slika 3: Prikaz pogleda 6](#_Toc11242876)

[Slika 4: Poizvedba v tabeli napotnica 7](#_Toc11242877)

[Slika 5: Poizvedba iz tabel pacienti ter naroceni 7](#_Toc11242878)

# UVOD

Pri predmetu Zbirke Podatkovnih Baz 1 smo si lahko izbrali poljubno temo projektne naloge, s katero naj predstavimo letošnje nabrano znanje. Za našo temo smo izbrali dermatološko ambulanto, saj poznamo nekoga, ki dela v dermatološki ambulanti. Zastavili smo nekaj vprašanj, da si lahko pomagamo z odgovori ter naredimo približek podatkovne baze.

# NAMEN, CILJI TER METODE DELA

Namen naloge je povzeti pridobljeno znanje predmeta Zbirke Podatkovnih Baz 1, ter ga predstaviti na strnjen način, v obliki ER diagrama ter z vzorci napolnjeno podatkovno bazo.  
Cilj naloge je ta ER diagram sestaviti, postaviti relacije med entitetami, ki nastopajo v podatkovni bazi, uvoziti podatkovno bazo (jo sestaviti iz ER diagrama) ter napolniti z vzorčnimi podatki.  
Uporabljene metode dela so delo z programsko opremo za upravljanje podatkovnih baz (MySQL), povpraševanje med. osebja v dermatološki ambulanti, iskanje ustreznih rešitev za probleme, ki so nastali med izdelavo podatkovne baze (brskalnik Google).

# MYSQL

MySQL je ustvarilo švedsko podjetje MySQL AB, je odprto kodni upravljalski sistem za relacijske baze. Prva verzija je izšla 23. maja 1995. Je komponenta LAMPa. MySQL je napisan v C in C++, razčlenjevalnik je napisan v »yacc«, uporablja pa domače napisan leksični analizator. (<https://en.wikipedia.org/wiki/MySQL> 11.6.2019)

Leta 2008 je podjetje Sun Microsystems kupilo MySQL AB, na začetku leta 2010 pa je Oracle kupil Sun Microsystems, tako da je trenutno MySQL produkt Oraclea. ([https://en.wikipedia.org/wiki/MySQL 11.6.2019](https://en.wikipedia.org/wiki/MySQL%2011.6.2019))



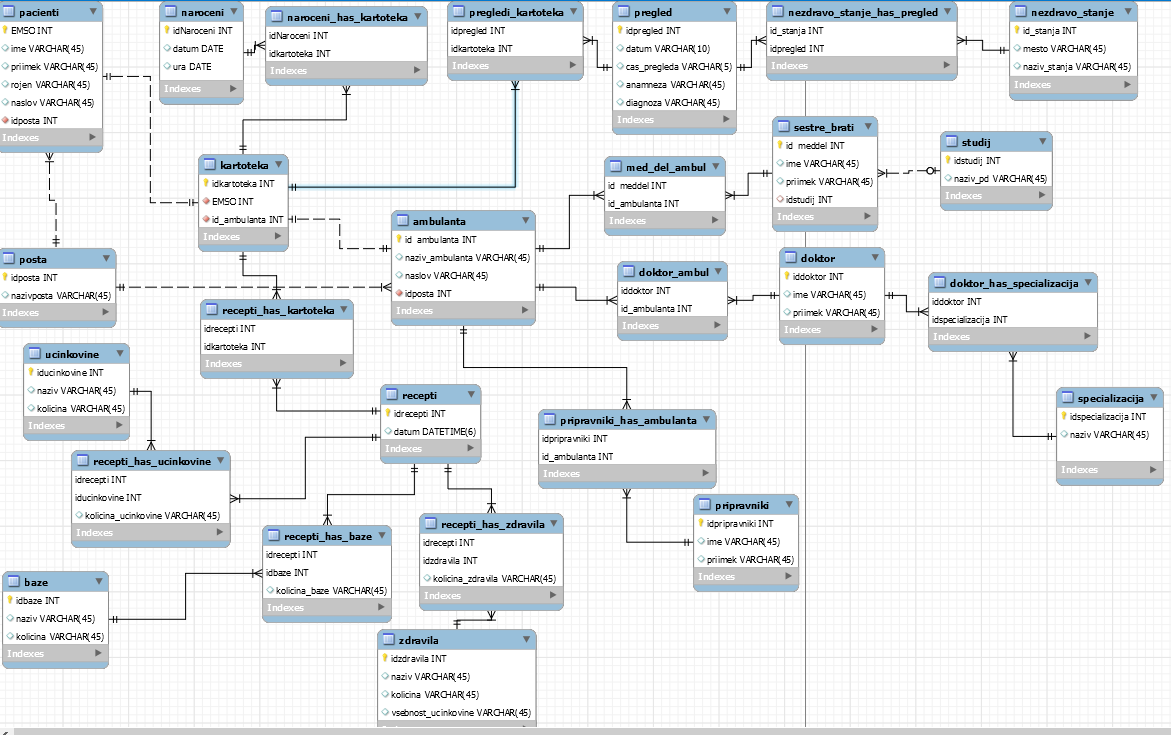
Slika 1: Logo MySQL <https://sl.wikipedia.org/wiki/MySQL#/media/Slika:MySQKL.jpg>

# IZDELAVA PODATKOVNE BAZE

Za končno podatkovno bazo rabimo kar nekaj korakov. Te korake bomo opisali v naslednjih delih.

## **Načrtovanje ER diagrama**

Pomembno je, da analiziramo kaj vse rabimo za našo podatkovno bazo. To vključuje entitete, atribute, ter relacije med entitetami.



Slika 2: Končni ER diagram

## **Relacije med entitetami**

Pacient ima lahko samo eno poštno št., eno poštno št. pa ima lahko en ali več pacientov.

En pacient ima lahko samo eno kartoteko, ena kartoteka ima lahko samo enega pacienta.

Ena ali več kartotek ima lahko enega ali več naročenih. En ali več naročenih ima lahko eno ali več kartotek.  
Ena poštna št. ima lahko eno ali več ambulant. Ena ali več ambulant ima lahko eno poštno št.

Ena ali več kartotek ima lahko en ali več receptov.

En ali več receptov ima lahko eno ali več kartotek.

En ali več receptov ima lahko eno ali več učinkovin.

Ena ali več učinkovin ima lahko en ali več receptov.

En ali več receptov ima lahko eno ali več zdravil.

Ena ali več zdravil ima lahko en ali več receptov.

En ali več receptov ima lahko eno ali več baz.

Ena ali več baz ima lahko en ali več receptov.

Ena ali več ambulant ima lahko enega ali več pripravnikov.

En ali več pripravnikov ima lahko eno ali več ambulant.

Ena ali več ambulant ima lahko enega ali več medicinskih delavcev (sestre\_brati).

En ali več medicinskih delavcev ima lahko en študij.

En študij ima lahko en ali več medicinskih delavcev.

En ali več doktorjev ima lahko eno ali več ambulant.

Ena ali več ambulant ima lahko enega ali več doktorjev.

En ali več doktorjev ima lahko eno ali več specializacij.

Ena ali več specializacij ima lahko en ali več doktorjev.

Ena ali več kartotek ima lahko en ali več pregledov.

En ali več pregledov ima lahko eno ali več kartotek.

En ali več pregledov ima lahko eno ali več nezdravih stanj.

Eno ali več nezdravih stanj ima lahko en ali več pregledov.

## **Poizvedbe**

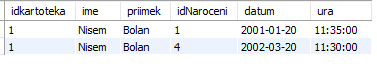
select k.idkartoteka, p.ime, p.priimek, n.idNaroceni, n.datum, n.ura

from naroceni n, kartoteka k

inner join pacienti p on p.EMSO = k.EMSO

inner join naroceni\_has\_kartoteka nhk on nhk.idkartoteka = k.idkartoteka

where p.EMSO = "1" and n.idNaroceni = nhk.idNaroceni;



Slika 3: Kdaj se je prvi pacient naročil na pregled?

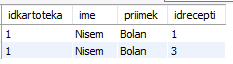
select k.idkartoteka, p.ime, p.priimek, rhk.idrecepti

from kartoteka k

inner join pacienti p on p.EMSO = k.EMSO

inner join recepti\_has\_kartoteka rhk on rhk.idkartoteka = k.idkartoteka

where k.idkartoteka = 1 and p.emso = 1;



Slika 4: Katere recepte je pacient prejel?

select r.idrecepti as ID\_Recepta, b.naziv as Baza, rhb.kolicina\_baze as Kol\_Baze, u.naziv as Ucinkovina, rhu.kolicina\_ucinkovine as Kol\_Ucin, z.naziv as Zdravilo, rhz.kolicina\_zdravila as Kol\_Zdrav

from baze b, ucinkovine u, zdravila z, recepti r

inner join recepti\_has\_ucinkovine rhu on rhu.idrecepti = r.idrecepti

inner join recepti\_has\_zdravila rhz on rhz.idrecepti = r.idrecepti

inner join recepti\_has\_baze rhb on rhb.idrecepti = r.idrecepti

inner join recepti\_has\_kartoteka rhk on rhk.idrecepti = r.idrecepti

where r.idrecepti IN (select rhk.idrecepti

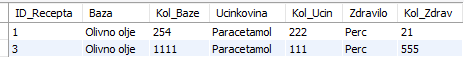
from kartoteka k

inner join pacienti p on p.EMSO = k.EMSO

inner join recepti\_has\_kartoteka rhk on rhk.idkartoteka = k.idkartoteka

where k.idkartoteka = 1 and p.emso = 1)

group by r.idrecepti;



Slika 5: Recepti za pacienta 1, kjer so navedena vsaj 1 baza, 1 učinkovina, 1 zdravilo

## **Triggerji**

Ker želimo, da se nam zaloge same znižajo, ko pride do uporabe zdravil, učinkovin, baz, smo uporabili tri podobne triggerje, ki delujejo isto ampak za različne tabele.

CREATE DEFINER=`root`@`localhost` TRIGGER `recepti\_has\_baze\_BEFORE\_INSERT` BEFORE INSERT ON `recepti\_has\_baze` FOR EACH ROW BEGIN

update baze

set kolicina = kolicina - NEW.kolicina\_baze

where baze.idbaze = new.idbaze;

END

CREATE DEFINER=`root`@`localhost` TRIGGER `recepti\_has\_ucinkovine\_BEFORE\_INSERT` BEFORE INSERT ON `recepti\_has\_ucinkovine` FOR EACH ROW BEGIN

update ucinkovine

set kolicina = kolicina - NEW.kolicina\_ucinkovine

where ucinkovine.iducinkovine = new.iducinkovine;

END

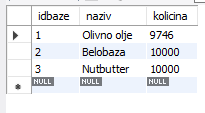
CREATE DEFINER=`root`@`localhost` TRIGGER `recepti\_has\_zdravila\_BEFORE\_INSERT` BEFORE INSERT ON `recepti\_has\_zdravila` FOR EACH ROW BEGIN

update zdravila

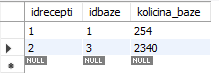
set kolicina = kolicina - NEW.kolicina\_zdravila

where zdravila.idzdravila = new.idzdravila;

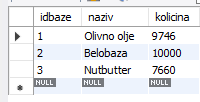
END



Slika 6: Tabela baze pred porabo količine



Slika 7: Z idrecept = 2, bomo znižali kolicino tretje baze



Slika 8: Kolicina je znižana

# Zaključek

S prikazom delovanja naše podatkovne baze smo zadovoljni. Tako smo zaključili to projektno nalogo. Izdelava baze ter pisanje triggerjev je bilo zanimivo.

# Viri

<https://en.wikipedia.org/wiki/MySQL>