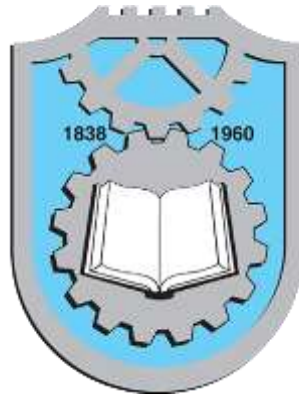


Univerzitet u Kragujevcu
Fakultet inženjerskih nauka



Baze podataka

Projektni zadatak:
IS Mašinijada

Student:

Uroš Stanojkov 601/2018

Predmetni nastavnici:

Prof. Dr Milan Erić
Doc. Dr Aleksandar Đorđević
Saradnik Nenad Pantelić

Sadržaj

1. Uvod

1.1 Relevantni dokumenti i informacione potrebe realnog sistema	3
---------------------------------------------------------------------------	---

2. ER Dijagram

2.1 Entiteti	4
2.2 Veze	10
2.3 Kompletan ER dijagram	17

3. Logička šema relacione baze podataka

3.1 Entiteti	18
3.2 Veze nasleđivanja	19
3.3 Gerundi	19
3.4 Preostale veze	19
3.5 Korigovanje relacija radi poštovanja normalnih formi.	21
3.6 Međurelaciona ograničenja	22
3.7 Kompletan logička šema	22

4. Fizička šema relacione baze podataka

4.1 Predstavljanje fizičkog modela	25
----------------------------------------------	----

5. Implementacija projektovane baze podataka sa testnim podacima u MySQL

5.1 Slike implementacije u MySQL-u	26
----------------------------------------------	----

6. Literatura

6.1 Literatura	39
--------------------------	----

Rezime

U ovom projektnom zadatku biće opisan ceo sistem baze podataka koji je potreban za organizovanje takmičenja studenata mašinskih fakulteta iz zemalja regiona, Mašinijadama.

Sama struktura baze podataka predstavlja celu organizaciju mašinijade, gde se nalaze razni učesnici, koji se mogu takmičiti u raznim disciplinama, kao i profesorima, organizatorima, itd. Prvo će biti predstavljen ER dijagram, iz njega će se izvesti ekvivalentna šema relacije baze podataka, tj. logičke šeme in a kraju će se definisati njena fizička šema.

Ključne reči: Baza podataka, ER dijagram, šema relacije baze podataka, SQL.

1. Uvod

Mašinijada je opšte poznato takmičenje kome prisustvuje većina zemalja iz regiona, kao što su Srbija, Makedonija, Bugarska, Crna Gora, Bosna i Hercegovina, itd. Mašinijadi prisustvuju studenti iz ovih zemalja, koji se takmiče u raznim disciplinama. Nisu svi studenti takmičari, deo studenata ide da posmatra ove događaje. Discipline u kojima se studenti takmiče su uglavnom sportovi – fudbal, košarka, rukomet, odbojka, ali takođe postoje i takmičenja u raznim naučnim disciplinama, poput, matematike, mehanike, i ostalih.

Na mašinijadama se mogu naći i profesori, kao i organizatori same mašinijade, koje ubrajamo u učesnike. Organizatori pripadaju raznim sektorima, recimo, za organizaciju takmičenja, za turističke obilaske, za organizaciju žurki...

Takmičarima je cilj da osvoje što bolje nagrade koje odnosi fakultet, za većinu takmičarskih disciplina postoje nagrade za prvo, drugo i treće mesto, dok recimo u naučnim disciplinama postoji više prvih, drugih i trećih mesta. Sve discipline se održavaju na nekim mestima održavanja, a to mogu biti teren i sala za takmičarske discipline i učionica za naučne.

Mašinijadi mogu prisustvovati i gledaoci. Osobe koje će kupiti kartu da bi mogle da prisustvuju na određenim mestima gde se održavaju takmičenja.

2. ER dijagram

Modelovanje realnog sistema započinje se identifikovanjem entiteta koji postoje u sistemu, nakon toga se određuju veze između postojećih entiteta.

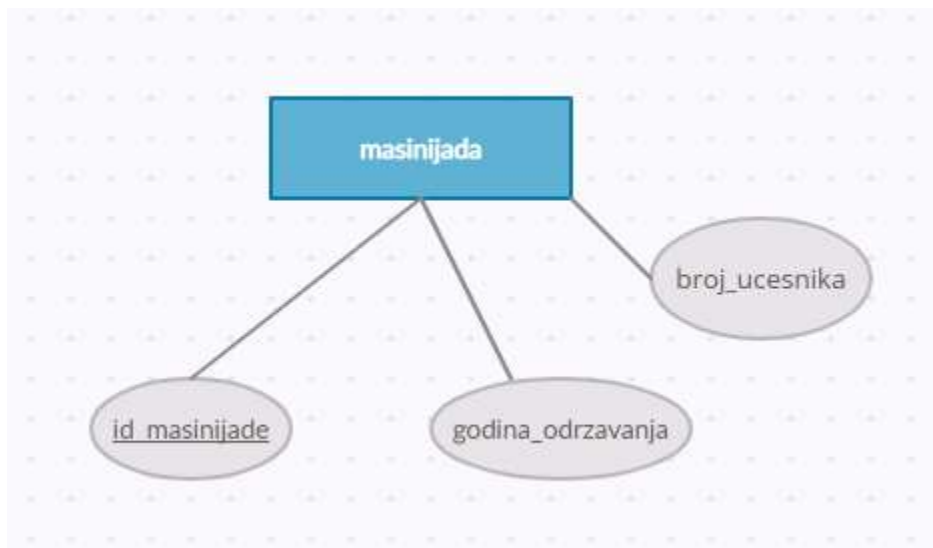
2.1 Entiteti

Identifikovani su dole navedeni entiteti, počev od najosnovnijeg.

- **Mašinjada**

Entitet *Mašinjada* je okarakterisan svojim identifikatorom (*id_masinijade*), *godinom održavanja* i *brojem učesnika* na mašinjadi.

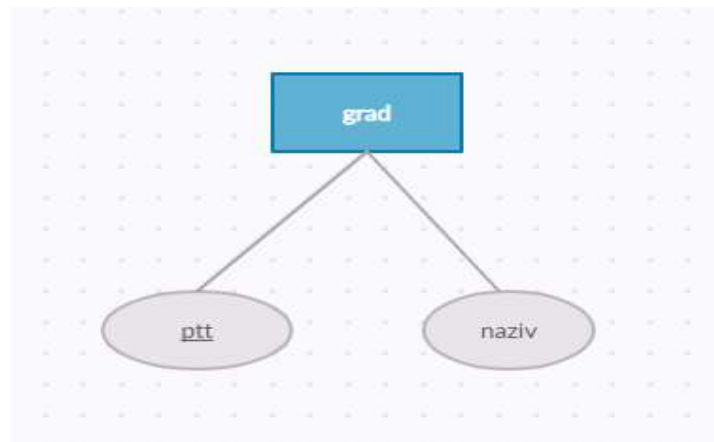
Ovaj entitet nije jedinstven, u bazi podataka se mogu naći više različitih Mašinjada – slika 2.1.



Slika 2.1 – Entitet Mašinjada

- **Grad**

Entitet *grad* je predstavljen svojim identifikatorom (*ptt*) i svojim *nazivom* – slika 2.2.



Slika 2.2 – Entitet grad

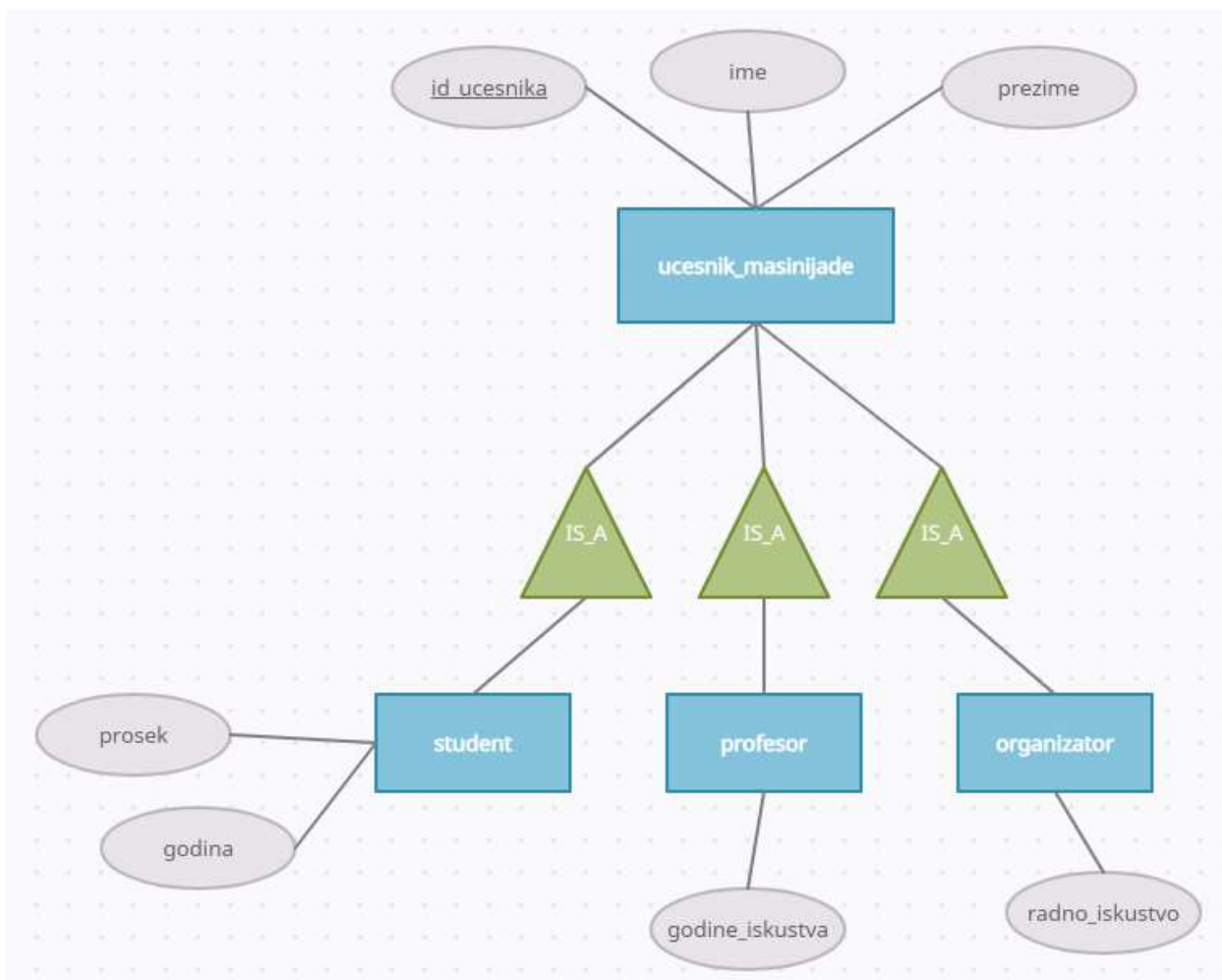
- **Učesnik mašinijade i tipovi učesnika mašinijade**

Entitet *učesnik mašinijade* sadrži svoj identifikator (*id_ucesnika*), kao i *ime* i *prezime* učesnika. Učesnici mašinijade mogu biti *studenti*, *profesori* ili sami *organizatori* mašinijade. Dakle entiteti *student*, *profesor* i *organizator* su *podentiteti* entiteta *učesnik mašinijade*.

Student takođe sadrži atribut *prosek* studenta tokom školovanja i njegovu trenutnu *godinu studiranja*.

Entitet *profesor* sadrži atribut *godine_iskustva*.

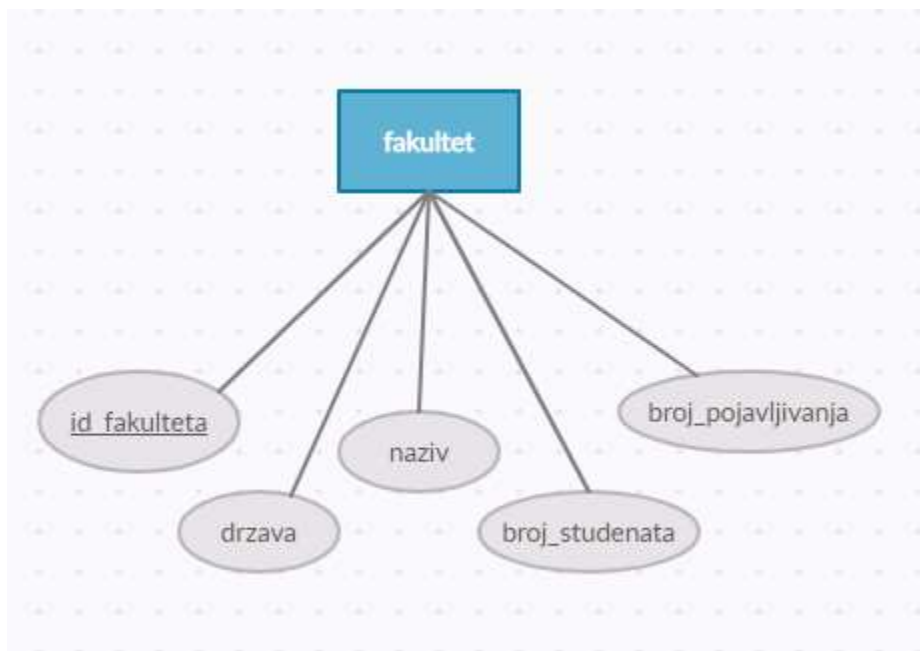
Entitet *organizator* sadrži atribut *radno_iskustvo* – slika 2.3.



Slika 2.3 – Entitet Učesnik mašinijade sa podentitetima

- **Fakultet**

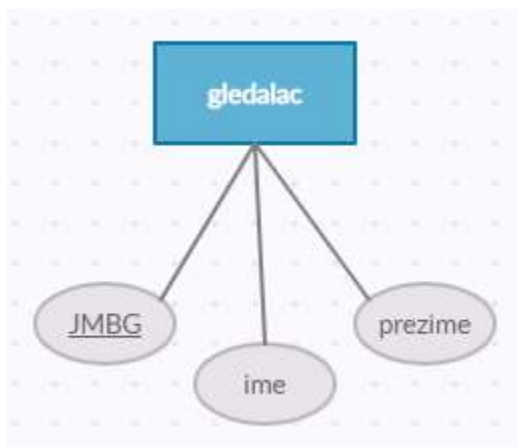
Entitet *fakultet* sadrži od atributa svoj jedinstveni identifikator (*id fakulteta*), kao i svoj *naziv*, *državu* u kojoj se nalazi, *broj pojavljivanja* na Mašinijadama zaključno sa ovom i *broj studenata* koji pohađaju fakultet – slika 2.4.



Slika 2.4 – Entitet fakultet

- **Gledalac**

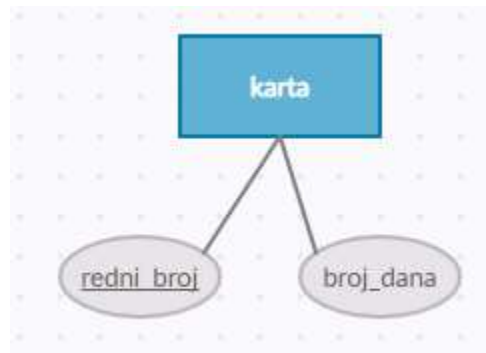
Entitet *gledalac* označava određenu osobu koja je došla da isprati događaje na mašinijadi. Od atributa sadrži svoj identifikator – *JMBG*, *ime* i *prezime* – slika 2.5.



Slika 2.5 – Entitet gledalac

- **Karta**

Karta je entitet koji od atributa sadrži, pored identifikatora koji predstavlja njen redni broj, i atribut *broj dana* koji predstavlja koliko dana se ta karta može koristiti za ulaz na događaje mašinijade – slika 2.6.



Slika 2.6 – Entitet karta

- **Disciplina**

Entitet *disciplina* predstavlja disciplinu u kojoj se određeni student takmiči. Entitet od atributa sadrži svoj identifikator (id_discipline), njen *naziv* i njeno *prosečno trajanje* – slika 2.7.



Slika 2.7 – Entitet disciplina

- **Mesto održavanja discipline (Mesto održavanja)**

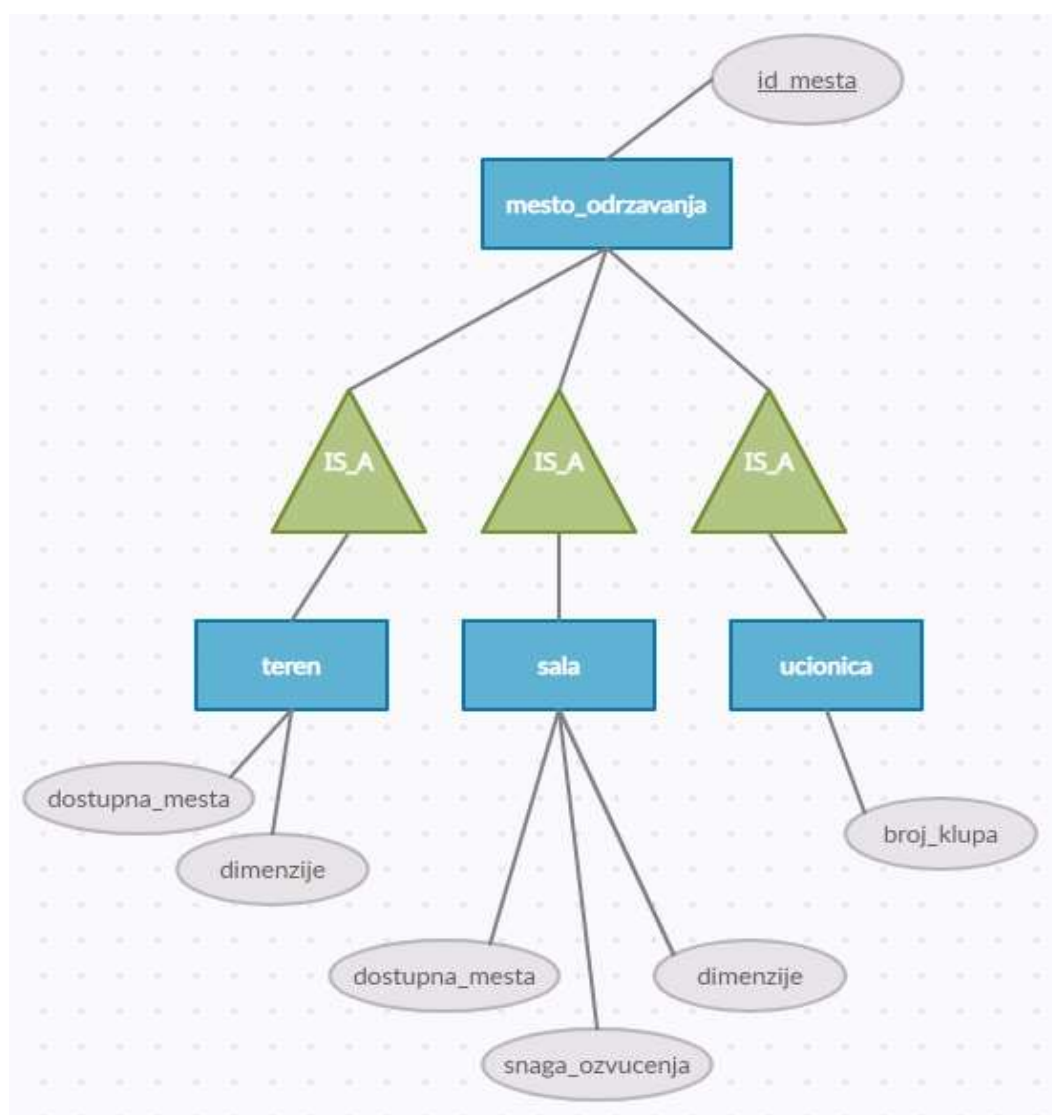
Entitet *mesto održavanja* nam govori gde se određena disciplina održava. Entitet od atributa ima samo svoj identifikator (*id_mesta*). Mesto održavanja se razlikuje, ukupno postoje tri različita mesta koja su pogodna za održavanje određenih disciplina.

To su *teren*, *sala* i *učionica* i oni predstavljaju *podentitete* mesta održavanja.

Teren od atributa sadrži *broj dostupnih mesta* za gledaoce i *dimenzije* samog terena.

Sala od atributa sadrži takođe *broj dostupnih mesta* za gledaoce, *snagu ozvučenja* u sali i *dimenzije* sale.

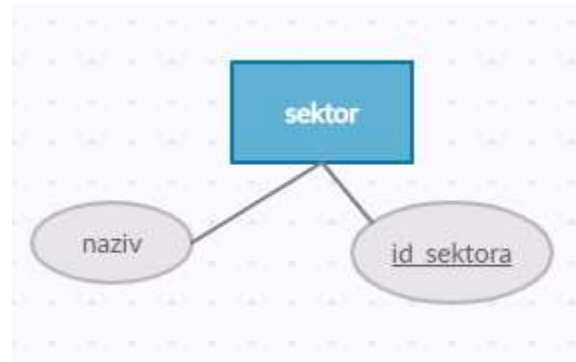
Učionica od atributa sadrži *broj klupa* u toj učionici – slika 2.8.



Slika 2.8 – Entitet Mesto održavanja sa podentitetima

- **Sektor**

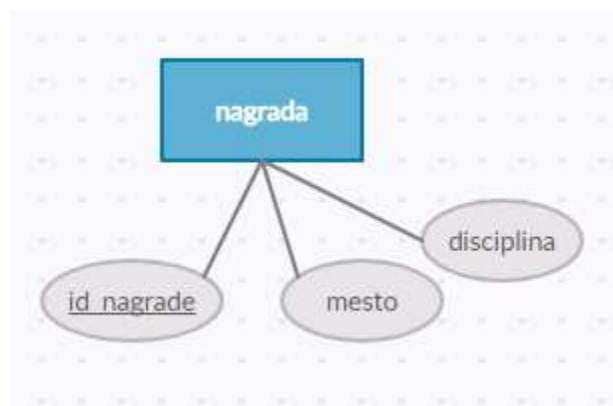
Entitet *sektor* predstavlja grupu koju čine organizatori, postoje sektor za IT, sektor za sport, sektor za organizaciju žurki, itd... Od atributa ima jedinstveni identifikator (*id sektora*) i *naziv* sektora – slika 2.9.



Slika 2.9 – Entitet sektor

- **Nagrada**

Entitet *nagrada* sadrži informacije koje označavaju *rang mesto* fakulteta koji dobija nagradu, kao i *disciplinu* za koju dobija nagradu. Entitet nagrada ima jedinstveni identifikator (*id nagrade*) – slika 2.10.



Slika 2.10 – Entitet nagrada

2.2 Veze

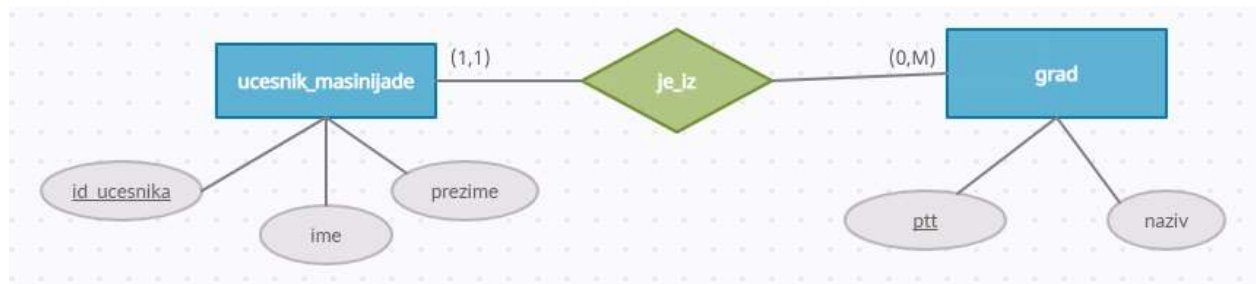
Nakon definisanja entiteta, njihovih podentiteta, kao i njihovih atributa, sada je na redu definisanje povezanosti entiteta, tj. *veza*. U konkretnom primeru IS Mašinjada će imati entitet *Učesnik mašinjade* koji ima podentitete: *Student*, *Profesor* i *Organizator* i entitet *Mesto održavanja* koji ima svoje podentitete: *Teren*, *Sala* i *Učionica*. Priroda relacije nasleđivanja je takva da podentitet pored atributa njihovih nadtipova, nasleđuju sve veze koje se odnose na njihove nadtipove.

Sledi opis veza između entiteta u sistemu IS Mašinjada.

- **Veza JE_IZ**

Postojeće više veza koje povezuju entitet Grad zajedno sa nekim drugim entitetom. Veza *JE_IZ* povezuje *Grad* i *Učesnika Mašinjade*, tako što svaki učesnik mašinjade je iz jednog i samo jednog grada, a jedan grad ne mora imati nijednog učesnika mašinjade a može ih imati više.

Kardinalnost ove veze je 1:M – slika 2.11.

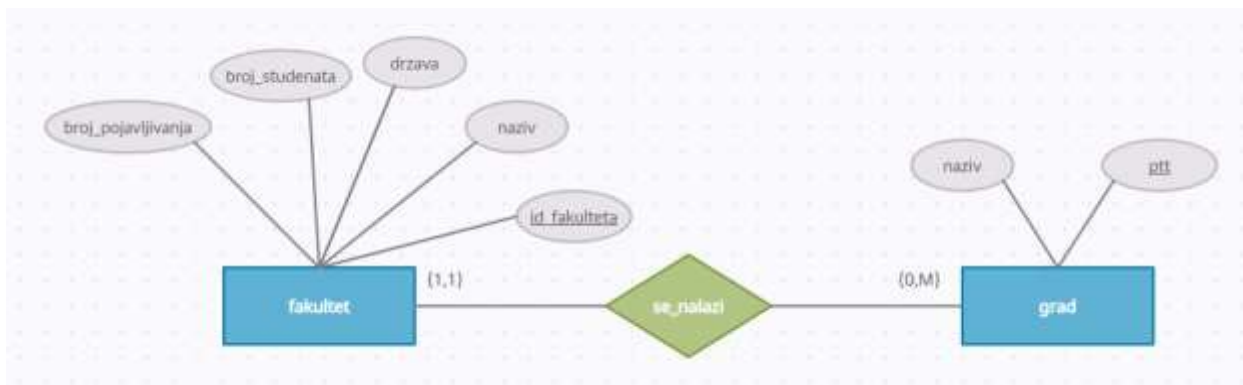


Slika 2.11 – Veza JE_IZ

- **Veza SE_NALAZI**

Veza *SE_NALAZI* je druga veza koja povezuje *Grad* i još neki entitet. U ovom slučaju je to entitet *Fakultet*. Slično kao kod veze *JE_IZ*, fakultet se nalazi u jednom i samo jednom gradu, dok se u nekom gradu može nalaziti više fakulteta, a ne mora ni jedan.

Kardinalnost u ovom slučaju je takođe 1:M – slika 2.12.



Slika 2.12 – Veza SE_NALAZI

- **Veza SE_ODRZAVA**

Poslednja veza koja uključuje entitet *Grad* i u ovom slučaju entitet *Mašinjada*, je veza *SE_ODRZAVA*. Veza nam govori to da se jedna Mašinjada može održavati u jednom i samo jednom gradu, dok se u jednom gradu može održavati više Mašinjada istovremeno.

Kardinalnost ove veze je takođe 1:M – slika 2.13.

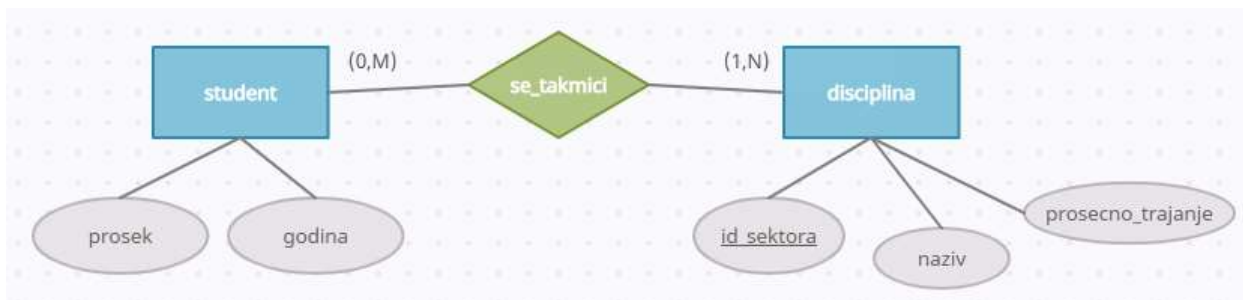


Slika 2.13 – Veza SE_ODRZAVA

- **Veza SE_TAKMICI**

Svaki *student* se može takmičiti u više *disciplina*, a ne mora ni u jednoj (oni koji su došli samo da prisustvuju događajima). Da bi se neka disciplina našla u bazi podataka bitno je da se barem jedan student takmiči u toj disciplini, a obično ih bude i više. Kao što se zaključuje veza *SE_TAKMICI* povezuje entitet *Student* i *Disciplina*.

Njena kardinalnost je M:N – slika 2.14.

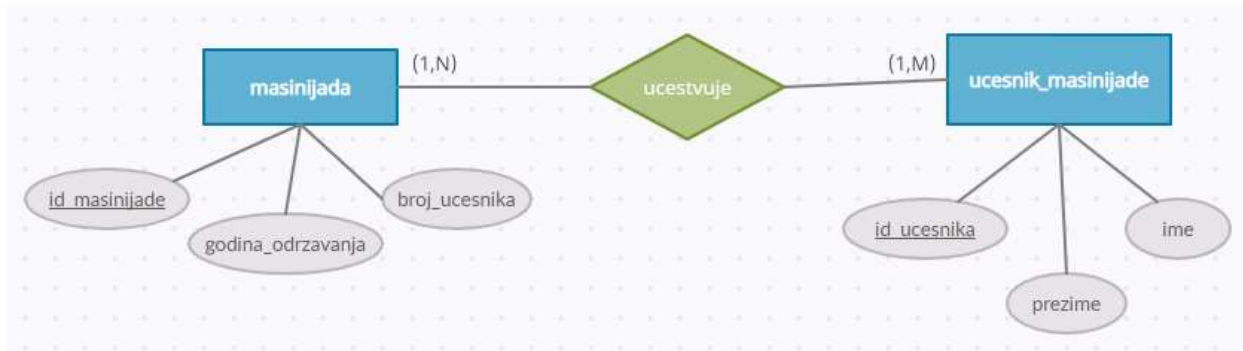


Slika 2.14 – Veza SE_TAKMICI

- **Veza UCESTVUJE**

Veza *UCESTVUJE* povezuje entitete *Mašinjada* i *Učesnika mašinjade*. Na jednoj Mašinjadi mora da bude barem jedan učesnik, a uvek ih je više. Učesnik mašinjade da bi bio u bazi podataka mora da ide na barem jednu Mašinjadu, a može ići i na više Mašinjada.

Kardinalnost veze UCESTVUJE je M:N – slika 2.15.

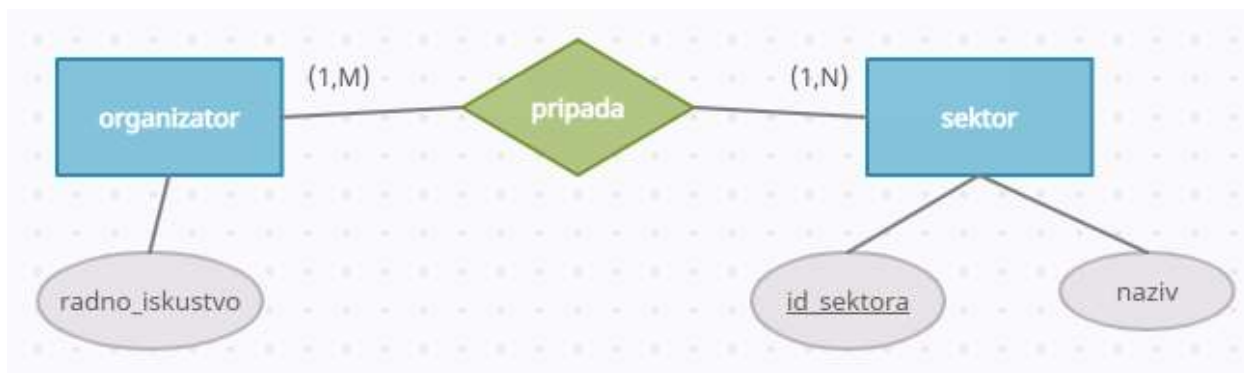


Slika 2.15 – Veza UCESTVUJE

- **Veza PRIPADA**

Veza *PRIPADA* označava koji *organizator* ili *organizatori* pripadaju kom *sektoru*. Organizator mora da pripada barem jednom sektoru, ali može biti i u više sektora. U jednom sektoru se nalazi barem jedan organizator, obično ih je i više.

Kardinalnost veze PRIPADA je M:N – slika 2.16.

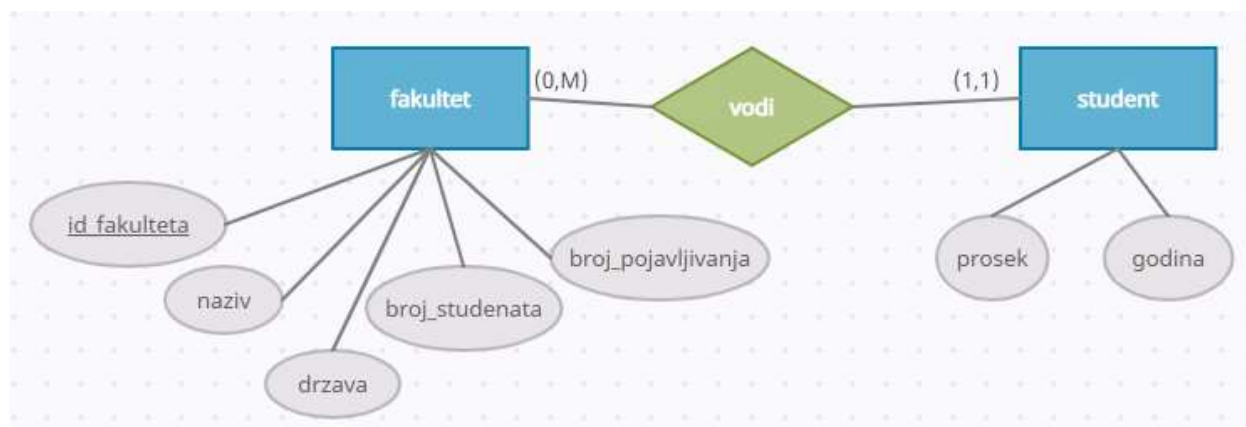


Slika 2.16 – Veza PRIPADA

- **Veza VODI**

Veza *VODI* označava da svaki *fakultet* vodi više *studenata*, a ne mora nijednog. *Studenta* mora voditi jedan i samo jedan *fakultet*.

Kardinalnost veze VODI je M:1 – slika 2.17.

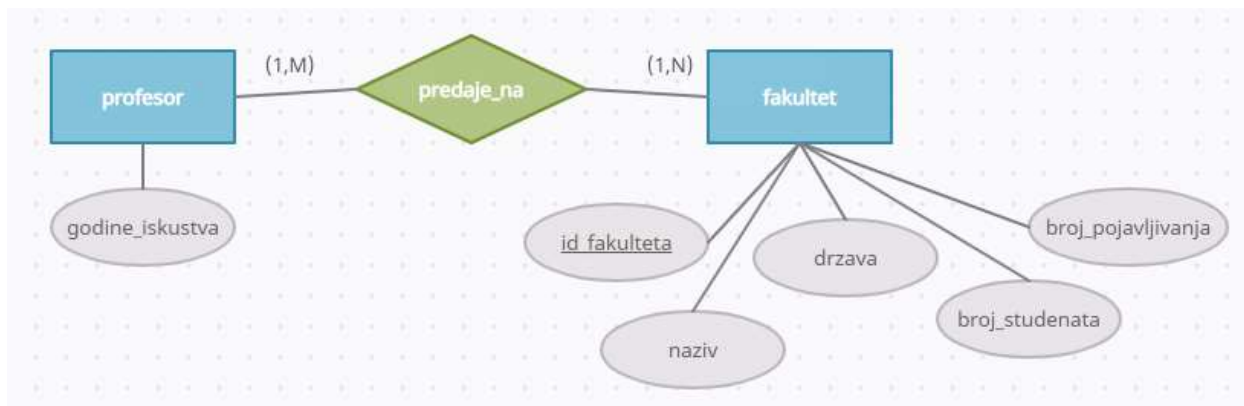


Slika 2.17 – Veza VODI

- **Veza PREDAJE_NA**

Veza koja predstavlja koji *profesor* predaje na kom *fakultetu*. Jedan profesor mora predavati na barem jednom fakultetu da bi bio profesor, a može predavati i na više. Na jednom fakultetu ima barem jedan profesor, uvek ih je više.

Kardinalnost veze PREDAJE_NA je M:N – slika 2.18.

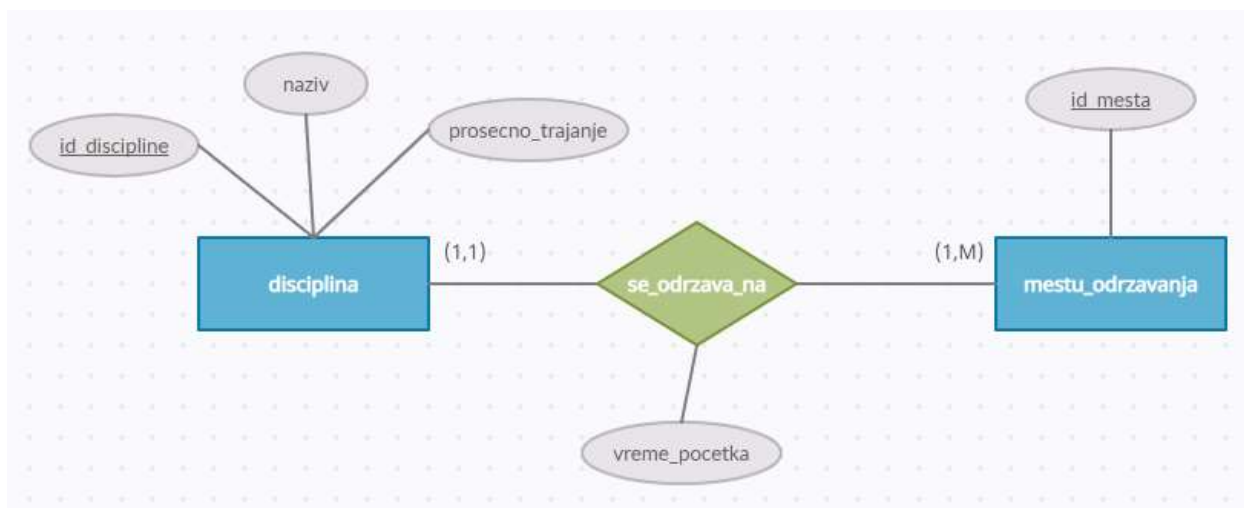


Slika 2.18 – Veza PREDAJE_NA

- **Veza SE_ODRZAVA_NA**

Veza koja predstavlja na kom *Mestu održavanja* se održava koja *disciplina*. Ova veza ima svoj *atribut veze* koji označava *vreme početka održavanja* neke discipline. Jedna disciplina se može održavati na jednom i samo jednom mestu održavanja, a na jednom jestu održavanja se održava barem jedna disciplina, a može i više.

Kardinalnost veze SE_ODRZAVA_NA je 1:M – slika 2.19.

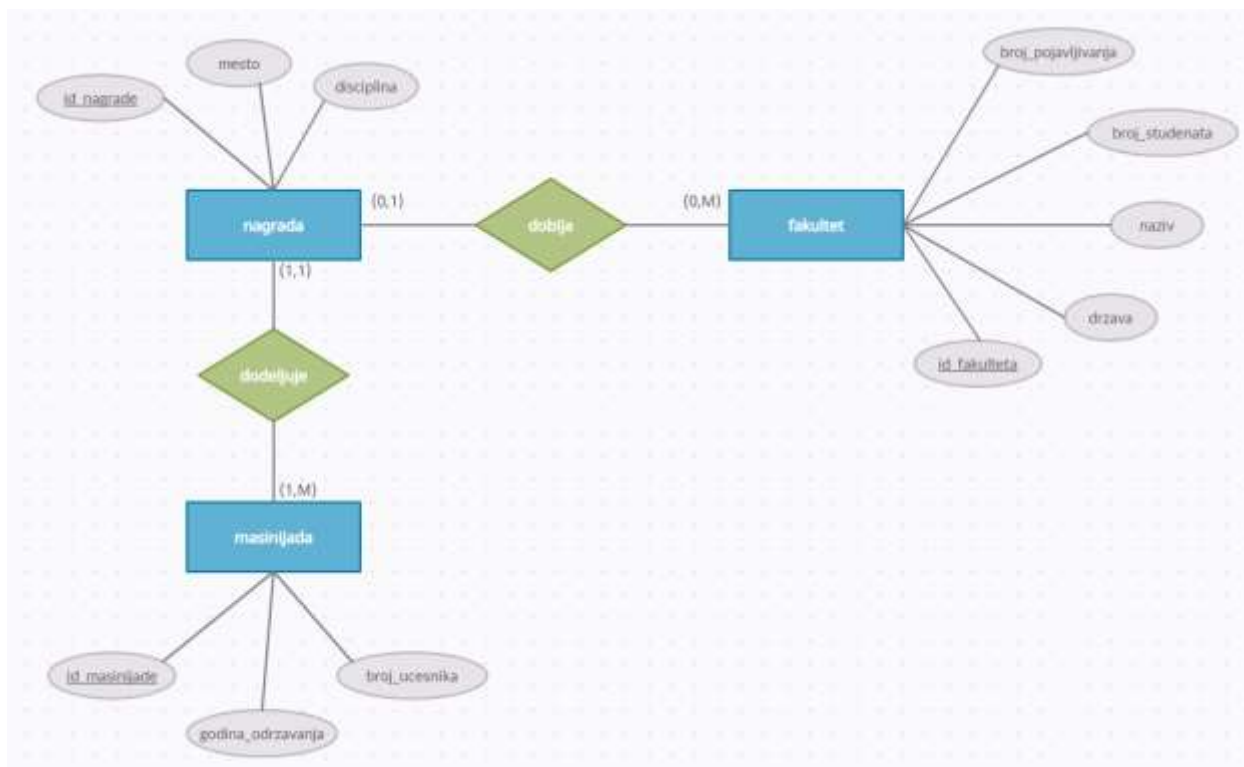


Sliak 2.19 – Veza SE_ODRZAVA_NA

- **Veza DOBIJA i veza DODELJUJE**

Veza *DOBIJA* predstavlja koji *fakultet* dobija *nagradu*. Fakultet ne mora da dobije nijednu nagradu, a može dobiti više nagrada. Jedna nagrada se dobija samo jednom, a ne mora nijednom. Kardinalnost veze DOBIJA je 1:M.

Veza *DODELJUJE* označava koja *nagrada* se dodeljuje od strane *Mašinjade*. Mašinjada dodeljuje barem jednu nagradu, a obično dodeljuje više. Svaka nagrada mora biti dodeljena jednom i samo jednom. Kardinalnost veze DODELJUJE je takođe 1:M – slika 2.20.



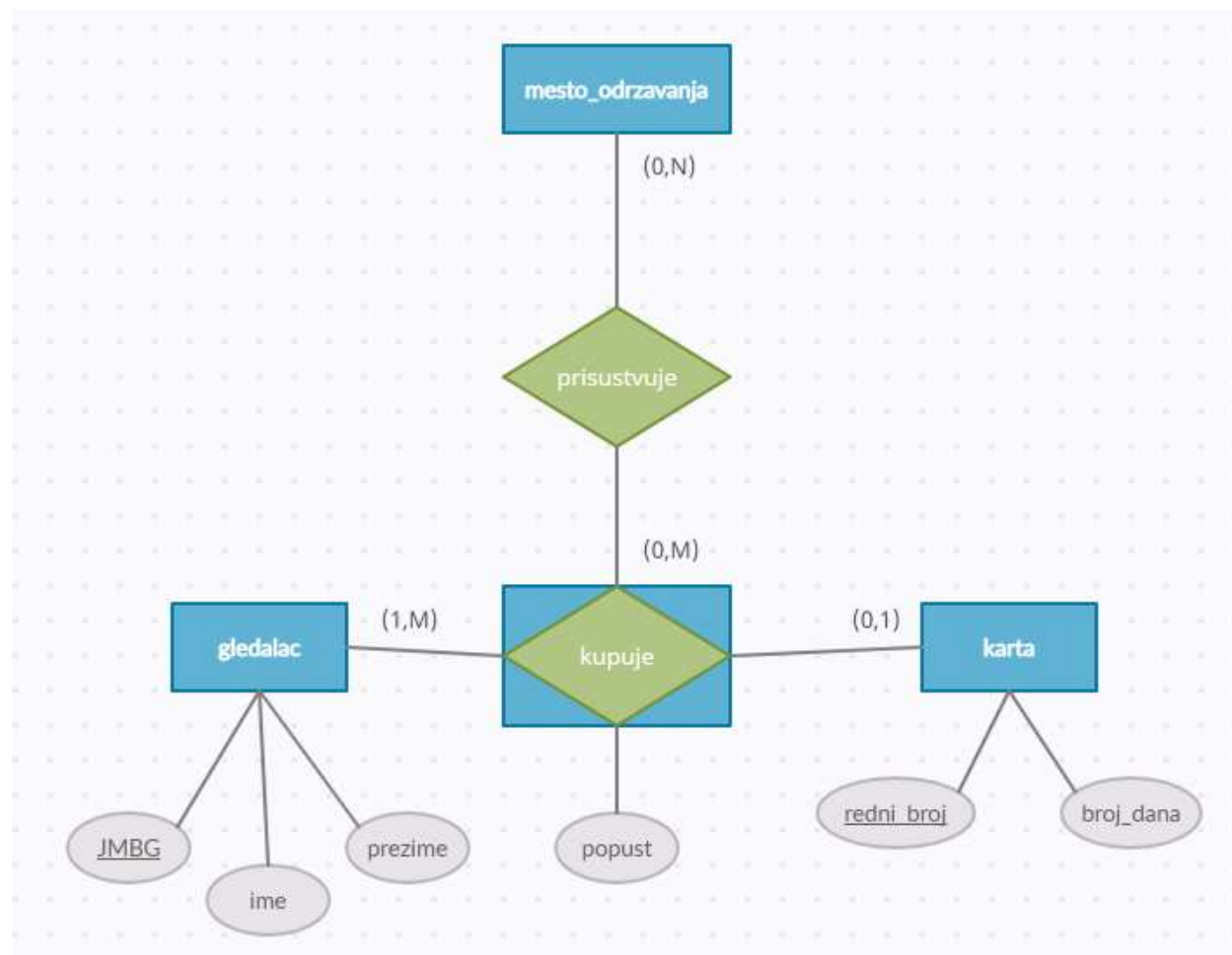
Slika 2.20 – Veze DOBIJA i DODELJUJE

- **Veza PRISUSTVUJE i Gerund KUPUJE**

Da bi gledalac prisustvovao nekom mestu održavanja on mora kupiti kartu. Zbog ovakvog načina predstavlja se koristi *gerund*, jer je gledalac u mogućstvu da prisustvuje samo ako kupi kartu.

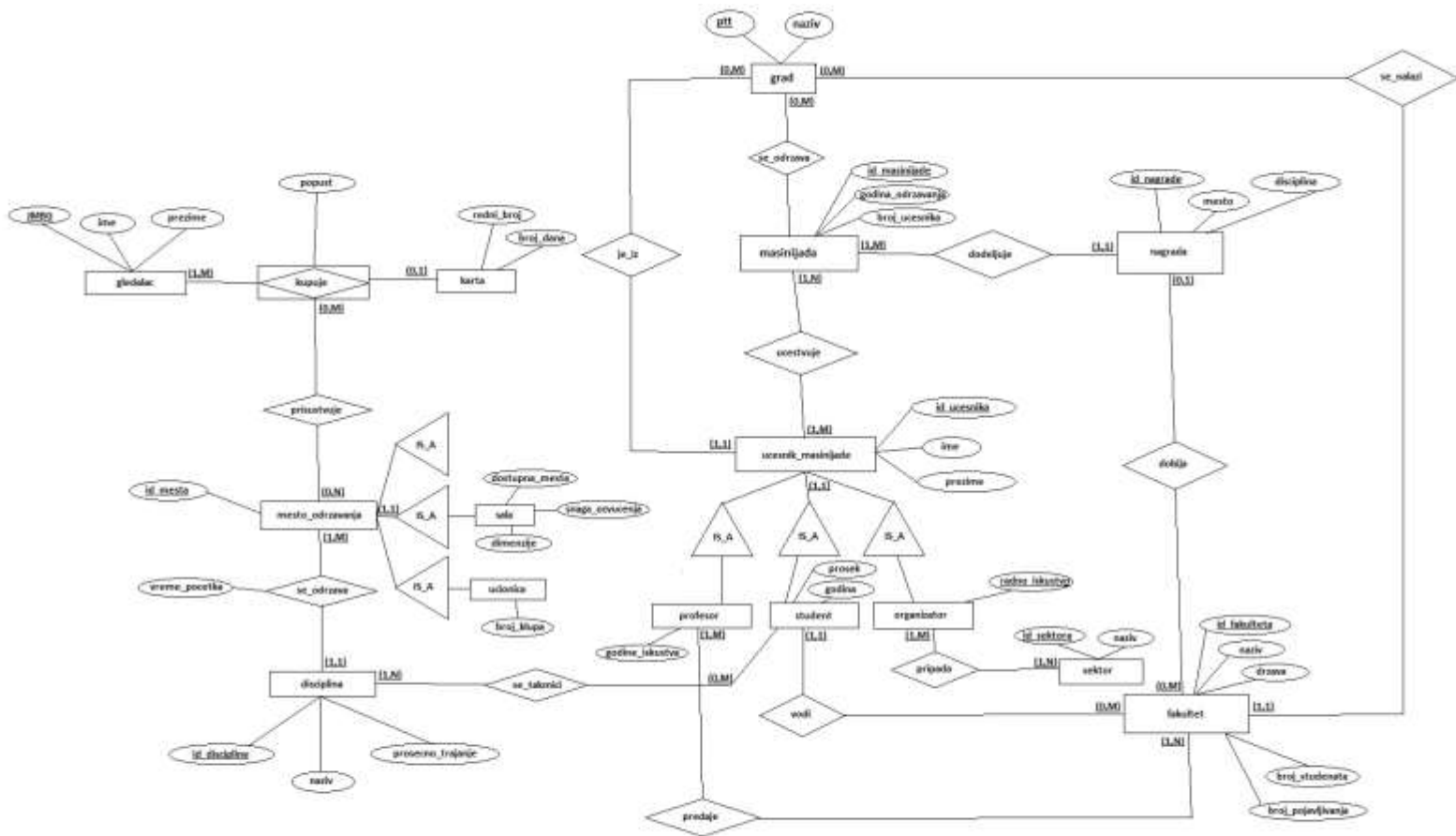
Da bi se *gledalac* našao u bazi podataka mora kupiti barem jednu *kartu*, a može i više karata. Veza *KUPUJE* ima svoj atribut koji označava određeni *popust*. Jedna karta se može prodati samo jednom a ne mora ni jednom. Kardinalnost Veze KUPUJE je M:1.

Kada *gledalac* kupi *kartu* on može *prisustvovati* nekom *mestu održavanja*. On ne mora prisustvovati nijednom mestu održavanja, a može biti i na više. Na jednom mestu održavanja ne mora biti nijedan gledalac, a obično ih bude više. Kardinalnost veze PRISUSTVUJE je M:N – slika 2.21.



Slika 2.21 - Veza PRISUSTVUJE i Gerund KUPUJE

2.3 Kompletan ER diagram



Slika 2.22 – Kompletan ER dijagram

3. Logička šema relacione baze podataka

Da bi se dobila logička šema relacione baze podataka, moraju svi entiteti i veze da se prevedu iz modela objekti-veze na njihov ekvivalentni oblik u relacionom modelu – relacije i međurelaciona ograničenja.

Elementi će biti prevedeni u relacioni oblik sledećim redosledom.

- Entiteti
- Veze nasleđivanja
- Gerundi
- Preostale veze

3.1 Entiteti

Svi entiteti postaju nezavisne šeme relacija. Ime tipa entiteta postaje *ime šeme relacije*, obeležja tipa objekta su *obeležja šeme relacije (atributi)*. Identifikator entiteta postaje *primarni ključ* šeme relacije.

Odatle, prim skup relacija će se sastojati iz:

masinijada(id_masinijade, godina_odrzavanja, broj_ucesnika)
grad(ptt, naziv)
fakultet(id_fakulteta, drzava, naziv, broj_studenata, broj_pojavljivanja)
ucesnik_masinijade(id_ucesnika, ime, prezime)
student(prosek, godina)
profesor(godine_iskustva)
organizator(radno_iskustvo)
disciplina(id_discipline, naziv, prosecno_trajanje)
mesto_odrzavanja(id_mesta)
teren(dostupna_mesta, dimenzije)
sala(dostupna_mesta, snaga_ozvucenja, dimenzije)
ucionica(broj_klupa)
nagrada(id_nagrade, mesto, disciplina)
sektor(id_sektora, naziv)
gledalac(JMBG, ime, prezime)
karta(redni_broj, broj_dana)

3.2 Veze nasleđivanja

Tipovi podentiteta nasleđuju identifikator tipa nadentiteta. Ako tip podentiteta već ima svoj identifikator, primarni ključ se bira između identifikatora nadentiteta i identifikatora podentiteta. Ako tip podentiteta nema svoj identifikator, primarni ključ relacije podentiteta postaje identifikator tipa nadentiteta.

Izmenjene su sledeće relacije podentiteta:

student(id_ucesnika, prosek, godina)
profesor(id_ucesnika, godine_iskustva)
organizator(id_ucesnika, radno_iskustvo)
teren(id_mesta, dostupna_mesta, dimenzije)
sala(id_mesta, dostupna_mesta, snaga_ozvucenja, dimenzije)
ucionica(id_mesta, broj_klupa)

U prva tri slučaja, za relacije student, profesor i organizator primarni ključ nije morao biti *id_ucesnika*, već je mogao da bude i *br_indeksa*, *id_profesora* i *id_organizatora*, redom.

3.3 Gerundi

Agregirani objekti (mešoviti tip objekat veza, tj. *Gerund*) se posmatra na isti način kao i odgovarajuća veza. To je tip entiteta koji se ponaša kao veza ili tip veze koja može da učestvuje u drugim vezama – da se ponaša kao entitet. Gerund uvažava samo tipove *entiteta* koje povezuje prilikom prevođenja u relacioni oblik, dok tipove *veza* ignoriše.

U ovom konkretnom modelu, gerund postaje nova relacija, sa sledećim atributima. Atribut popust je u stvari atribut veze KUPUJE.

kupuje(redni_broj, JMBG, popust)

3.4 Preostale veze

Prilikom prevođenja u relacioni model takođe je potrebno obratiti pažnju na sve ostale veze u modelu.

Sledi opis za prevođenje svake veze iz modela:

Veze *JE_IZ*, *SE_ODRZAVA* (u kom gradu se održava mašinijada) i *SE_NALAZI*

Sve tri veze su povezane sa entitetom GRAD i entitetima, redom, UCESNIK_MASINIJADE, MASINIJADA i FAKULTET. Kardinalnost veze je svuda (0,M):(1,1).

Znači da ne nastaje nova šema relacija, već se u postojanu šemu relacije dodaje identifikator kao jedan od atributa koja je na strani kardinalnosti (1,1).

ucesnik_masinijade(id_ucesnika, **ptt**, ime, prezime)
masinijada(id_masinijade, **ptt**, godina_odrzavanja, broj_ucesnika)
fakultet(id_fakulteta, **ptt**, drzava, naziv, broj_studenata, broj_pojavljivanja)

Veza UCESTVUJE

Označava koji učesnik mašinijade učestvuje na kojoj mašinijadi. Kardinalnost ove veze je (1,N):(1,M). Kada je ova kardinalnost u pitanju (kao i (0,N):(1,M) ili (1,N):(0,M)) tada nastaje nova šema relacija od veze:

ucestvuje(id_masinijade, id_ucesnika)

Veza VODI

Veza koja kazuje koji fakultet vodi studenta. Kardinalnost veze je (0,M):(1,1) posmatrajući od entiteta FAKULTET ka entitetu STUDENT.

student(id_ucesnika, id_fakulteta, prosek, godina)

Veza PRIPADA

Povezuje entitet ORGANIZATOR i entitet SEKTOR. Kardinalnost je (1,M):(1,M). Nastaje nova relacija od veze, jedan od ključeva će biti id_organizatora koji dolazi iz šeme relacije organizator.

pripada(id_ucesnika, id_sektora)

Veza PREDAJE

Povezuje entitete PROFESOR i FAKULTET. Kardinalnost ove veze je (1,M):(1,N).

predaje(id_ucesnika, id_fakulteta)

Veza SE_TAKMICI

Povezuje entitete STUDENT i DISCIPLINA i kardinalnost veze je (0,M):(1,N), posmatrajući od entiteta STUDENT ka entitetu DISCIPLINA.

se_takmici(id_ucesnika, id_discipline)

Veza SE_ODRZAVA (na kom mestu održavanja se održava disciplina)

Kardinalnost veze je (1,1):(1,M) gledajući od entiteta DISCIPLINA ka entitetu MESTO_ODRZAVANJA.

To znači sledeće:

disciplina(id_discipline, id_mesta, vreme_pocetka, naziv, prosečno_trajanje)

Primećuje se da je pored atributa id_mesta, dodat i atribut vreme_pocetka, koji je u stvari atribut veze SE_ODRZAVA.

Veza *PRISUSTVUJE*

Veza prisustvuje povezuje gerund KUPUJE i entitet MESTO_ODRZAVANJA. Pošto je kardinalnost ove veze (0,M):(0,N) nastaje nova relacija:

prisustvuje(redni_broj, id_mesta)

Veza *DODELJUJE*

Veza koja povezuje entitete MASINIJADA i NAGRADA. Njena kardinalnost je od entiteta MASINIJADA ka entitetu NAGRADA (1,M):(1,1). To znači da nemamo novu relaciju već identifikator objekta sa strane za koju je gornja granica kardinaliteta GG = M postaje obeležje šeme relacije koja odgovara objektu sa strane za koju je GG = 1.

nagrada(id_nagrade, id_masinijade, mesto, disciplina)

Veza *DOBIJA*

Kardinalnost ove veze je (0,1):(0,M) posmatrajući od entiteta NAGRADA ka entitetu FAKULTET. To znači da imamo novu šemu relacije i ključ ove šeme relacije je identifikator objekta za koji je GG = 1.

dobija(id_nagrade, id_fakulteta)

3.5 Korigovanje relacija radi poštovanja normalnih formi

Primećuje se da u trenutnom sistemu ne postoji nijedan složeni atribut, dakle sve vrednosti su atomične i normalizovane, tj. poštuju *prvu normalnu formu* (1NF).

Relacija je u drugoj normalnoj formi ako i samo ako je već u prvoj normalnoj formi i ako svi njeni neključni atributi potpuno i funkcionalno zavise od primarnog ključa. U ovom slučaju je i *druga normalna forma* ispunjena, jer je u većini slučajeva primarni ključ samo jedan atribut (nije složeni ključ) ili ako je složeni ključ sastoji se od dva atributa i relacije nema neključne attribute.

Da bi treća normalna forma bila ispunjena potrebno je da je relacija u 2NF i da svi neključni atributi netranzitivno funkcionalno zavise od primarnog ključa, što je i ispunjeno. Pošto su sve tri normalne forme ispunjene neće biti korigovanja relacija.

3.6 Međurelaciona ograničenja

Sada se mogu definisati međurelaciona ograničenja. Ona se definišu za sve nasleđene attribute neke relacije – da moraju biti podskup skupa vrednosti nasleđenog atributa u originalnoj relaciji od koje je taj atribut nasleđen.

To uključuje sve strane ključeve svake relacije.

Opis svih međurelacionih ograničenja u datom modelu:

student[id_ucesnika] \subseteq ucesnik_masinijade[id_ucesnika]
profesor[id_ucesnika] \subseteq ucesnik_masinijade[id_ucesnika]
organizator[id_ucesnika] \subseteq ucesnik_masinijade[id_ucesnika]
teren[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
sala[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
ucionica[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
kupuje[redni_broj] \subseteq karta[redni_broj]
kupuje[JMBG] \subseteq gledalac[JMBG]
ucesnik_masinijade[ptt] \subseteq grad[ptt]
masinijada[ptt] \subseteq grad[ptt]
fakultet[ptt] \subseteq grad[ptt]
ucestvuje[id_masinijade] \subseteq masinijada[id_masinijade]
ucestvuje[id_ucesnika] \subseteq ucesnik_masinijade[id_ucesnika]
student[id_fakulteta] \subseteq fakultet[id_fakulteta]
pripada[id_organizatora] \subseteq organizator[id_organizatora]
pripada[id_sektora] \subseteq sektor[id_sektora]
predaje[id_profesora] \subseteq profesor[id_profesora]
predaje[id_fakulteta] \subseteq fakultet[id_fakulteta]
se_takmici[br_indeksa] \subseteq student[br_indeksa]
se_takmici[id_discipline] \subseteq disciplina[id_discipline]
disciplina[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
prisustvuje[redni_broj] \subseteq kupuje[redni_broj]
prisustvuje[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
nagrada[id_masinijade] \subseteq masinijada[id_masinijade]
dobija[id_nagrade] \subseteq nagrada[id_nagrade]
dobija[id_fakulteta] \subseteq fakultet[id_fakulteta]

3.7 Kompletna logička šema

Na osnovu dosadašnjeg izlaganja dolazi se do kompletnog relacionog modela sistema:

S = {

masinijada(id_masinijade, ptt, godina_odrzavanja, broj_ucesnika)
grad(ptt, naziv)
fakultet(id_fakulteta, ptt, drzava, naziv, broj_studenata, broj_pojavljivanja)
ucelnik_masinijade(id_ucelnika, ptt, ime, prezime)
student(id_ucelnika, id_fakulteta, prosek, godina)
profesor(id_ucelnika, godine_iskustva)
organizator(id_ucelnika, radno_iskustvo)
disciplina(id_discipline, id_mesta, naziv, prosečno_trajanje)
mesto_odrzavanja(id_mesta)
teren(id_mesta, dostupna_mesta, dimenzije)
sala(id_mesta, dostupna_mesta, snaga_ozvucenja, dimenzije)
ucionica(id_mesta, broj_klupa)
nagrada(id_nagrade, id_masinijade, mesto, disciplina)
sektor(id_sektora, naziv)
gledalac(JMBG, ime, prezime)
karta(redni_broj, broj_dana)
ucestvuje(id_masinijade, id_ucelnika)
pripada(id_ucelnika, id_sektora)
se_takmici(id_ucelnika, id_discipline)
prisustvuje(redni_broj, id_mesta)
dobija(id_nagrade, id_fakulteta)
kupuje(redni_broj, JMBG, popust)
predaje(id_ucelnika, id_fakulteta)

}

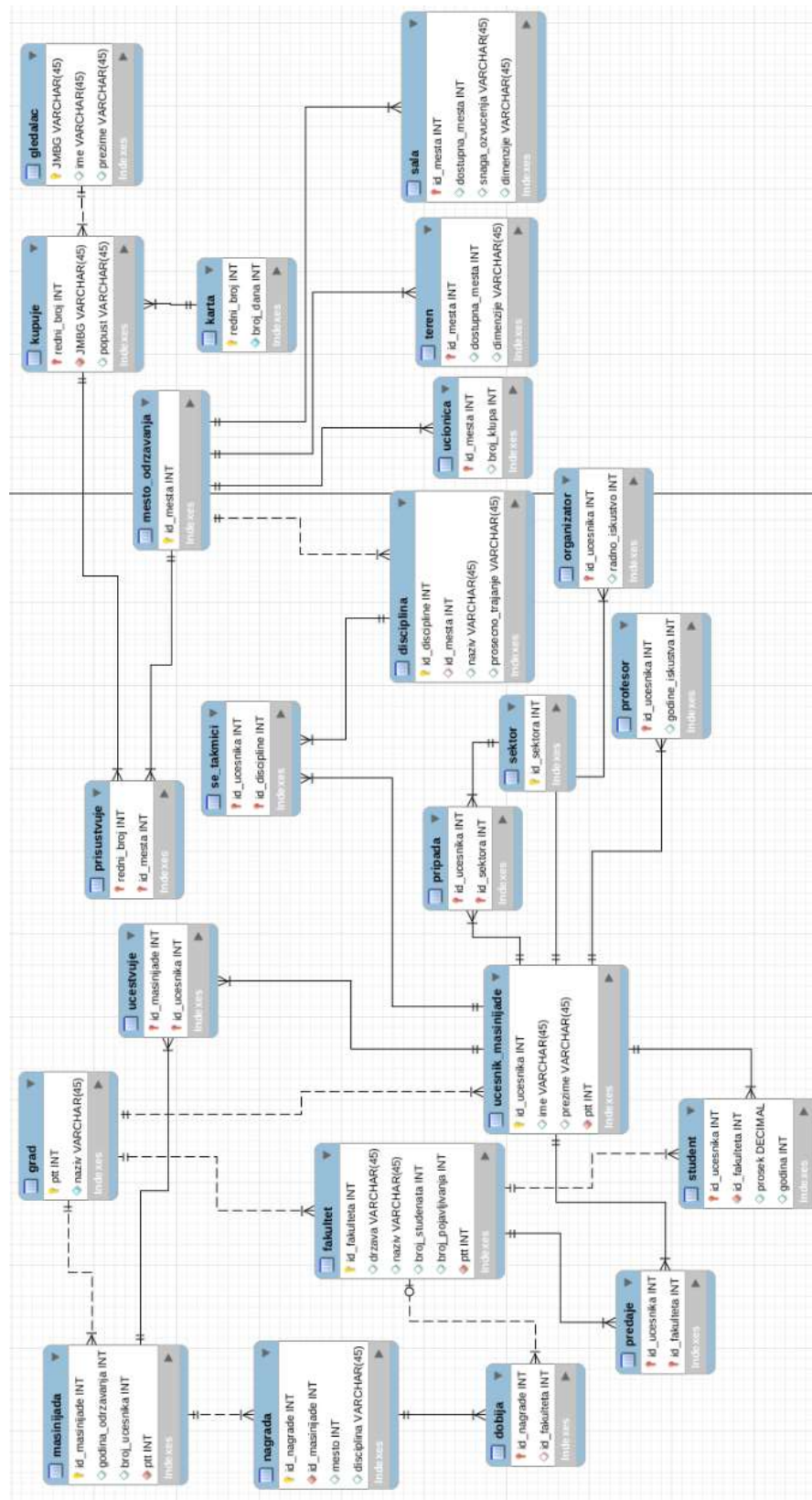
I = {

student[id_ucelnika] \subseteq ucelnik_masinijade[id_ucelnika]
profesor[id_ucelnika] \subseteq ucelnik_masinijade[id_ucelnika]
organizator[id_ucelnika] \subseteq ucelnik_masinijade[id_ucelnika]
teren[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
sala[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
ucionica[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
kupuje[redni_broj] \subseteq karta[redni_broj]
kupuje[JMBG] \subseteq gledalac[JMBG]
ucelnik_masinijade[ptt] \subseteq grad[ptt]
masinijada[ptt] \subseteq grad[ptt]
fakultet[ptt] \subseteq grad[ptt]
ucestvuje[id_masinijade] \subseteq masinijada[id_masinijade]
ucestvuje[id_ucelnika] \subseteq ucelnik_masinijade[id_ucelnika]
student[id_fakulteta] \subseteq fakultet[id_fakulteta]

pripada[id_organizatora] \subseteq organizator[id_organizatora]
pripada[id_sektora] \subseteq sektor[id_sektora]
predaje[id_profesora] \subseteq profesor[id_profesora]
predaje[id_fakulteta] \subseteq fakultet[id_fakulteta]
se_takmici[br_indeksa] \subseteq student[br_indeksa]
se_takmici[id_discipline] \subseteq disciplina[id_discipline]
disciplina[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
prisustvuje[redni_broj] \subseteq kupuje[redni_broj]
prisustvuje[id_mesta] \subseteq mesto_odrzavanja[id_mesta]
nagrada[id_masinijade] \subseteq masinijada[id_masinijade]
dobija[id_nagrade] \subseteq nagrada[id_nagrade]
dobija[id_fakulteta] \subseteq fakultet[id_fakulteta]
predaje[id_ucesnika] \subseteq profesor[id_ucesnika]
predaje[id_fakulteta] \subseteq fakultet[id_fakulteta]

}

4. Fizička šema relacione baze podataka



Fizička šema je najniži nivo apstrakcije podataka u bazi. Pomoću fizičke šeme se opisuje realan siste. To je poslednji korak pre implementacije samog sistema baze podataka.

Fizička šema je vrlo slična logičkoj, pa se vrši direktno mapiranje između logičke i fizičke šeme. Izuzeci su izmene koje olakšavaju održavanje podataka i povećane efikasnosti operacija sistema za upravljanje bazom podataka. Na datoj slici se vidi kompletna fizička šema, izrađena preko "MySQL Workbench" programa.

5. Implementacija projektovane baze podataka sa testnim podacima u MySQL

Kada implementiramo fizičku šemu u MySQL Workbenchu, postoji opcija da se izgeneriše SQL kod koji implementira fizičku šemu, zajedno sa testnim podacima koji su ručno ubačeni preko istog softverskog alata. Sledi kompletan SQL kod sa testnim primerima, koji implementira razmatrani realni sistem.

```
1 -- MySQL Workbench Forward Engineering
2
3 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
4 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
5 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,
6 NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
7
8 --
9 -- Schema masinijada
10 --
11
12 --
13 -- Schema masinijada
14 --
15 CREATE SCHEMA IF NOT EXISTS `masinijada` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci ;
16 USE `masinijada` ;
17
18 --
19 -- Table `masinijada`.`mesto_odrzavanja`
20 --
21 CREATE TABLE IF NOT EXISTS `masinijada`.`mesto_odrzavanja` (
22   `id_mesta` INT NOT NULL AUTO_INCREMENT,
23   PRIMARY KEY (`id_mesta`))
24 ENGINE = InnoDB
25 DEFAULT CHARACTER SET = utf8mb4
26 COLLATE = utf8mb4_0900_ai_ci;
27
```

```

28
29 -- -----
30 -- Table `masinijada`.`disciplina`
31 -- -----
32 CREATE TABLE IF NOT EXISTS `masinijada`.`disciplina` (
33   `id_discipline` INT NOT NULL AUTO_INCREMENT,
34   `id_mesta` INT NULL DEFAULT NULL,
35   `naziv` VARCHAR(45) NULL DEFAULT NULL,
36   `prosecno_trajanje` VARCHAR(45) NULL DEFAULT NULL,
37   PRIMARY KEY (`id_discipline`),
38   INDEX `id_mesta_idx` (`id_mesta` ASC) VISIBLE,
39   CONSTRAINT `stranil`
40     FOREIGN KEY (`id_mesta`)
41     REFERENCES `masinijada`.`mesto_odrzavanja` (`id_mesta`)
42     ON DELETE CASCADE
43     ON UPDATE CASCADE)
44 ENGINE = InnoDB
45 DEFAULT CHARACTER SET = utf8mb4
46 COLLATE = utf8mb4_0900_ai_ci;
47
48
49 -- -----
50 -- Table `masinijada`.`grad`
51 -- -----
52 CREATE TABLE IF NOT EXISTS `masinijada`.`grad` (
53   `ptt` INT NOT NULL,
54   `naziv` VARCHAR(45) NULL DEFAULT NULL,
55   PRIMARY KEY (`ptt`))
56 ENGINE = InnoDB
57 DEFAULT CHARACTER SET = big5;
58
59
60 -- -----
61 -- Table `masinijada`.`masinijada`
62 -- -----
63 CREATE TABLE IF NOT EXISTS `masinijada`.`masinijada` (
64   `id_masinijade` INT NOT NULL AUTO_INCREMENT,
65   `godina_odrzavanja` INT NULL DEFAULT NULL,
66   `broj_ucesnika` INT NULL DEFAULT NULL,
67   `ptt` INT NULL DEFAULT NULL,
68   PRIMARY KEY (`id_masinijade`),
69   INDEX `ptt_idx` (`ptt` ASC) VISIBLE,
70   CONSTRAINT `ptt`
71     FOREIGN KEY (`ptt`)
72     REFERENCES `masinijada`.`grad` (`ptt`)
73     ON DELETE SET NULL
74     ON UPDATE SET NULL)
75 ENGINE = InnoDB
76 DEFAULT CHARACTER SET = utf8mb4
77 COLLATE = utf8mb4_0900_ai_ci;
78
79

```



```

80 -- -----
81 -- Table `masinijada`.`nagrada`
82 -- -----
83 CREATE TABLE IF NOT EXISTS `masinijada`.`nagrada` (
84   `id_nagrada` INT NOT NULL AUTO_INCREMENT,
85   `id_masinijade` INT NOT NULL,
86   `mesto` INT NULL DEFAULT NULL,
87   `disciplina` VARCHAR(45) NULL DEFAULT NULL,
88   PRIMARY KEY (`id_nagrada`),
89   INDEX `id_masinijade_idx` (`id_masinijade` ASC) VISIBLE,
90   CONSTRAINT `strani8`
91     FOREIGN KEY (`id_masinijade`)
92     REFERENCES `masinijada`.`masinijada` (`id_masinijade`)
93     ON DELETE CASCADE
94     ON UPDATE CASCADE)
95 ENGINE = InnoDB
96 DEFAULT CHARACTER SET = utf8mb4
97 COLLATE = utf8mb4_0900_ai_ci;
98
100 -- -----
101 -- Table `masinijada`.`fakultet`
102 -- -----
103 CREATE TABLE IF NOT EXISTS `masinijada`.`fakultet` (
104   `id_fakulteta` INT NOT NULL AUTO_INCREMENT,
105   `drzava` VARCHAR(45) NULL DEFAULT NULL,
106   `naziv` VARCHAR(45) NULL DEFAULT NULL,
107   `broj_studenata` INT NULL DEFAULT NULL,
108   `broj_pojavljivanja` INT NULL DEFAULT NULL,
109   `ptt` INT NOT NULL,
110   PRIMARY KEY (`id_fakulteta`),
111   INDEX `ptt_idx` (`ptt` ASC) VISIBLE,
112   CONSTRAINT `strani4`
113     FOREIGN KEY (`ptt`)
114     REFERENCES `masinijada`.`grad` (`ptt`)
115     ON DELETE CASCADE
116     ON UPDATE CASCADE)
117 ENGINE = InnoDB
118 DEFAULT CHARACTER SET = utf8mb4
119 COLLATE = utf8mb4_0900_ai_ci;
120
121
122 -- -----
123 -- Table `masinijada`.`dobija`
124 -- -----
125 CREATE TABLE IF NOT EXISTS `masinijada`.`dobija` (
126   `id_nagrada` INT NOT NULL,
127   `id_fakulteta` INT NULL DEFAULT NULL,
128   PRIMARY KEY (`id_nagrada`),
129   INDEX `id_fakulteta_idx` (`id_fakulteta` ASC) VISIBLE,
130   CONSTRAINT `strani2`
131     FOREIGN KEY (`id_nagrada`)
132     REFERENCES `masinijada`.`nagrada` (`id_nagrada`)
133     ON DELETE CASCADE
134     ON UPDATE CASCADE,
135   CONSTRAINT `strani3`
136     FOREIGN KEY (`id_fakulteta`)
137     REFERENCES `masinijada`.`fakultet` (`id_fakulteta`)
138     ON DELETE CASCADE
139     ON UPDATE CASCADE)
140 ENGINE = InnoDB
141 DEFAULT CHARACTER SET = utf8mb4
142 COLLATE = utf8mb4_0900_ai_ci;

```

```

145 -- -----
146 -- Table `masinijada`.`gledalac`
147 -- -----
148 CREATE TABLE IF NOT EXISTS `masinijada`.`gledalac` (
149   `JMBG` VARCHAR(45) NOT NULL,
150   `ime` VARCHAR(45) NULL DEFAULT NULL,
151   `prezime` VARCHAR(45) NULL DEFAULT NULL,
152   PRIMARY KEY (`JMBG`))
153 ENGINE = InnoDB
154 DEFAULT CHARACTER SET = utf8mb4
155 COLLATE = utf8mb4_0900_ai_ci;
156
157
158 -- -----
159 -- Table `masinijada`.`karta`
160 -- -----
161 CREATE TABLE IF NOT EXISTS `masinijada`.`karta` (
162   `redni_broj` INT NOT NULL AUTO_INCREMENT,
163   `broj_dana` INT NOT NULL,
164   PRIMARY KEY (`redni_broj`))
165 ENGINE = InnoDB
166 DEFAULT CHARACTER SET = utf8mb4
167 COLLATE = utf8mb4_0900_ai_ci;
168
169
170 -- -----
171 -- Table `masinijada`.`kupuje`
172 -- -----
173 CREATE TABLE IF NOT EXISTS `masinijada`.`kupuje` (
174   `redni_broj` INT NOT NULL,
175   `JMBG` VARCHAR(45) NOT NULL,
176   `popust` VARCHAR(45) NULL DEFAULT NULL,
177   PRIMARY KEY (`redni_broj`),
178   INDEX `JMBG_idx` (`JMBG` ASC) VISIBLE,
179   CONSTRAINT `strani5`
180     FOREIGN KEY (`redni_broj`)
181     REFERENCES `masinijada`.`karta` (`redni_broj`)
182     ON DELETE CASCADE
183     ON UPDATE CASCADE,
184   CONSTRAINT `strani6`
185     FOREIGN KEY (`JMBG`)
186     REFERENCES `masinijada`.`gledalac` (`JMBG`)
187     ON DELETE CASCADE
188     ON UPDATE CASCADE)
189 ENGINE = InnoDB
190 DEFAULT CHARACTER SET = utf8mb4
191 COLLATE = utf8mb4_0900_ai_ci;

```

```

194 -- -----
195 -- Table `masinijada`.`ucesnik_masinijade`
196 -- -----
197 CREATE TABLE IF NOT EXISTS `masinijada`.`ucesnik_masinijade` (
198   `id_ucesnika` INT NOT NULL AUTO_INCREMENT,
199   `ime` VARCHAR(45) NULL DEFAULT NULL,
200   `prezime` VARCHAR(45) NULL DEFAULT NULL,
201   `ptt` INT NOT NULL,
202   PRIMARY KEY (`id_ucesnika`),
203   INDEX `ptt_idx` (`ptt` ASC) VISIBLE,
204   CONSTRAINT `strani23`
205     FOREIGN KEY (`ptt`)
206     REFERENCES `masinijada`.`grad` (`ptt`)
207     ON DELETE CASCADE
208     ON UPDATE CASCADE)
209 ENGINE = InnoDB
210 DEFAULT CHARACTER SET = utf8mb4
211 COLLATE = utf8mb4_0900_ai_ci;
212
213
214 -- -----
215 -- Table `masinijada`.`organizator`
216 -- -----
217 CREATE TABLE IF NOT EXISTS `masinijada`.`organizator` (
218   `id_ucesnika` INT NOT NULL,
219   `radno_iskustvo` INT NULL DEFAULT NULL,
220   PRIMARY KEY (`id_ucesnika`),
221   CONSTRAINT `strani9`
222     FOREIGN KEY (`id_ucesnika`)
223     REFERENCES `masinijada`.`ucesnik_masinijade` (`id_ucesnika`)
224     ON DELETE CASCADE
225     ON UPDATE CASCADE)
226 ENGINE = InnoDB
227 DEFAULT CHARACTER SET = utf8mb4
228 COLLATE = utf8mb4_0900_ai_ci;
229
230
231 -- -----
232 -- Table `masinijada`.`predaje`
233 -- -----
234 CREATE TABLE IF NOT EXISTS `masinijada`.`predaje` (
235   `id_ucesnika` INT NOT NULL,
236   `id_fakulteta` INT NOT NULL,
237   PRIMARY KEY (`id_ucesnika`, `id_fakulteta`),
238   INDEX `id_fakulteta_idx` (`id_fakulteta` ASC) VISIBLE,
239   CONSTRAINT `strani10`
240     FOREIGN KEY (`id_ucesnika`)

```



```

241 REFERENCES `masinijada`.`ucesnik_masinijade` (`id_ucesnika`)
242 ON DELETE CASCADE
243 ON UPDATE CASCADE,
244 CONSTRAINT `strani11`
245 FOREIGN KEY (`id_fakulteta`)
246 REFERENCES `masinijada`.`fakultet` (`id_fakulteta`)
247 ON DELETE CASCADE
248 ON UPDATE CASCADE)
249 ENGINE = InnoDB
250 DEFAULT CHARACTER SET = utf8mb4
251 COLLATE = utf8mb4_0900_ai_ci;
252
253
254 -----
255 -- Table `masinijada`.`sektor`
256 -----
257 CREATE TABLE IF NOT EXISTS `masinijada`.`sektor` (
258   `id_sektora` INT NOT NULL AUTO_INCREMENT,
259   PRIMARY KEY (`id_sektora`))
260 ENGINE = InnoDB
261 DEFAULT CHARACTER SET = utf8mb4
262 COLLATE = utf8mb4_0900_ai_ci;
263
264
265 -----
266 -- Table `masinijada`.`pripada`
267 -----
268 CREATE TABLE IF NOT EXISTS `masinijada`.`pripada` (
269   `id_ucesnika` INT NOT NULL,
270   `id_sektora` INT NOT NULL,
271   PRIMARY KEY (`id_ucesnika`, `id_sektora`),
272   INDEX `id_sektora_idx` (`id_sektora` ASC) VISIBLE,
273   CONSTRAINT `strani12`
274     FOREIGN KEY (`id_ucesnika`)
275     REFERENCES `masinijada`.`ucesnik_masinijade` (`id_ucesnika`)
276     ON DELETE CASCADE
277     ON UPDATE CASCADE,
278   CONSTRAINT `strani13`
279     FOREIGN KEY (`id_sektora`)
280     REFERENCES `masinijada`.`sektor` (`id_sektora`)
281     ON DELETE CASCADE
282     ON UPDATE CASCADE)
283 ENGINE = InnoDB
284 DEFAULT CHARACTER SET = utf8mb4
285 COLLATE = utf8mb4_0900_ai_ci;

```

```

288 -- -----
289 -- Table `masinijada`.`prisustvuje`
290 -- -----
291 CREATE TABLE IF NOT EXISTS `masinijada`.`prisustvuje` (
292   `redni_broj` INT NOT NULL,
293   `id_mesta` INT NOT NULL,
294   PRIMARY KEY (`redni_broj`, `id_mesta`),
295   INDEX `id_mesta_idx` (`id_mesta` ASC) VISIBLE,
296   CONSTRAINT `strani14`
297     FOREIGN KEY (`redni_broj`)
298     REFERENCES `masinijada`.`kupuje` (`redni_broj`)
299     ON DELETE CASCADE
300     ON UPDATE CASCADE,
301   CONSTRAINT `strani15`
302     FOREIGN KEY (`id_mesta`)
303     REFERENCES `masinijada`.`mesto_odrzavanja` (`id_mesta`)
304     ON DELETE CASCADE
305     ON UPDATE CASCADE)
306 ENGINE = InnoDB
307 DEFAULT CHARACTER SET = utf8mb4
308 COLLATE = utf8mb4_0900_ai_ci;
309
310
311 -- -----
312 -- Table `masinijada`.`profesor`
313 -- -----
314 CREATE TABLE IF NOT EXISTS `masinijada`.`profesor` (
315   `id_ucesnika` INT NOT NULL,
316   `godine_iskustva` INT NULL DEFAULT NULL,
317   PRIMARY KEY (`id_ucesnika`),
318   CONSTRAINT `strani16`
319     FOREIGN KEY (`id_ucesnika`)
320     REFERENCES `masinijada`.`ucesnik_masinijade` (`id_ucesnika`)
321     ON DELETE CASCADE
322     ON UPDATE CASCADE)
323 ENGINE = InnoDB
324 DEFAULT CHARACTER SET = utf8mb4
325 COLLATE = utf8mb4_0900_ai_ci;
326
327
328 -- -----
329 -- Table `masinijada`.`sala`
330 -- -----
331 CREATE TABLE IF NOT EXISTS `masinijada`.`sala` (
332   `id_mesta` INT NOT NULL,
333   `dostupna_mesta` INT NULL DEFAULT NULL,
334   `snaga_ozvucenja` VARCHAR(45) NULL DEFAULT NULL,

```



```

335 `dimenzije` VARCHAR(45) NULL DEFAULT NULL,
336 PRIMARY KEY (`id_mesta`),
337 CONSTRAINT `strani17`
338     FOREIGN KEY (`id_mesta`)
339     REFERENCES `masinijada`.`mesto_odrzavanja` (`id_mesta`)
340     ON DELETE CASCADE
341     ON UPDATE CASCADE)
342 ENGINE = InnoDB
343 DEFAULT CHARACTER SET = utf8mb4
344 COLLATE = utf8mb4_0900_ai_ci;
345
346
347 -----
348 -- Table `masinijada`.`se_takmici`
349 -----
350 CREATE TABLE IF NOT EXISTS `masinijada`.`se_takmici` (
351     `id_ucesnika` INT NOT NULL,
352     `id_discipline` INT NOT NULL,
353     PRIMARY KEY (`id_ucesnika`, `id_discipline`),
354     INDEX `id_discipline_idx` (`id_discipline` ASC) VISIBLE,
355     CONSTRAINT `strani18`
356     FOREIGN KEY (`id_ucesnika`)
357     REFERENCES `masinijada`.`ucesnik_masinijade` (`id_ucesnika`)
358     ON DELETE CASCADE
359     ON UPDATE CASCADE,
360     CONSTRAINT `strani19`
361     FOREIGN KEY (`id_discipline`)
362     REFERENCES `masinijada`.`disciplina` (`id_discipline`)
363     ON DELETE CASCADE
364     ON UPDATE CASCADE)
365 ENGINE = InnoDB
366 DEFAULT CHARACTER SET = utf8mb4
367 COLLATE = utf8mb4_0900_ai_ci;
368
369
370 -----
371 -- Table `masinijada`.`student`
372 -----
373 CREATE TABLE IF NOT EXISTS `masinijada`.`student` (
374     `id_ucesnika` INT NOT NULL,
375     `id_fakulteta` INT NOT NULL,
376     `prosek` DECIMAL(10,0) NULL DEFAULT NULL,
377     `godina` INT NULL DEFAULT NULL,
378     PRIMARY KEY (`id_ucesnika`),
379     INDEX `id_fakulteta_idx` (`id_fakulteta` ASC) VISIBLE,
380     CONSTRAINT `strani20`
381     FOREIGN KEY (`id_ucesnika`)

```

```

382 REFERENCES `masinijada`.`ucesnik_masinijade` (`id_ucesnika`)
383 ON DELETE CASCADE
384 ON UPDATE CASCADE,
385 CONSTRAINT `strani21`
386 FOREIGN KEY (`id_fakulteta`)
387 REFERENCES `masinijada`.`fakultet` (`id_fakulteta`)
388 ON DELETE CASCADE
389 ON UPDATE CASCADE)
390 ENGINE = InnoDB
391 DEFAULT CHARACTER SET = utf8mb4
392 COLLATE = utf8mb4_0900_ai_ci;
393
394
395 -----
396 -- Table `masinijada`.`teren`
397 -----
398 CREATE TABLE IF NOT EXISTS `masinijada`.`teren` (
399   `id_mesta` INT NOT NULL,
400   `dostupna_mesta` INT NULL DEFAULT NULL,
401   `dimenzije` VARCHAR(45) NULL DEFAULT NULL,
402   PRIMARY KEY (`id_mesta`),
403   CONSTRAINT `strani22`
404   FOREIGN KEY (`id_mesta`)
405   REFERENCES `masinijada`.`mesto_odrzavanja` (`id_mesta`)
406   ON DELETE CASCADE
407   ON UPDATE CASCADE)
408 ENGINE = InnoDB
409 DEFAULT CHARACTER SET = utf8mb4
410 COLLATE = utf8mb4_0900_ai_ci;
411
412
413 -----
414 -- Table `masinijada`.`ucestvuje`
415 -----
416 CREATE TABLE IF NOT EXISTS `masinijada`.`ucestvuje` (
417   `id_masinijade` INT NOT NULL,
418   `id_ucesnika` INT NOT NULL,
419   PRIMARY KEY (`id_masinijade`, `id_ucesnika`),
420   INDEX `id_ucesnika_idx` (`id_ucesnika` ASC) VISIBLE,
421   CONSTRAINT `strani24`
422   FOREIGN KEY (`id_masinijade`)
423   REFERENCES `masinijada`.`masinijada` (`id_masinijade`)
424   ON DELETE CASCADE
425   ON UPDATE CASCADE,
426   CONSTRAINT `strani25`
427   FOREIGN KEY (`id_ucesnika`)
428   REFERENCES `masinijada`.`ucesnik_masinijade` (`id_ucesnika`)

```

```

429     ON DELETE CASCADE
430     ON UPDATE CASCADE)
431 ENGINE = InnoDB
432 DEFAULT CHARACTER SET = utf8mb4
433 COLLATE = utf8mb4_0900_ai_ci;
434
435
436 -- -----
437 -- Table `masinijada`.`ucionica`
438 -- -----
439 CREATE TABLE IF NOT EXISTS `masinijada`.`ucionica` (
440   `id_mesta` INT NOT NULL,
441   `broj_klupa` INT NULL DEFAULT NULL,
442   PRIMARY KEY (`id_mesta`),
443   CONSTRAINT `strani26`
444     FOREIGN KEY (`id_mesta`)
445     REFERENCES `masinijada`.`mesto_odrzavanja` (`id_mesta`)
446     ON DELETE CASCADE
447     ON UPDATE CASCADE)
448 ENGINE = InnoDB
449 DEFAULT CHARACTER SET = utf8mb4
450 COLLATE = utf8mb4_0900_ai_ci;
451
452
453 SET SQL_MODE=@OLD_SQL_MODE;
454 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
455 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
456
457 -- -----
458 -- Data for table `masinijada`.`mesto_odrzavanja`
459 -- -----
460 START TRANSACTION;
461 USE `masinijada`;
462 INSERT INTO `masinijada`.`mesto_odrzavanja` (`id_mesta`) VALUES (1);
463 INSERT INTO `masinijada`.`mesto_odrzavanja` (`id_mesta`) VALUES (2);
464 INSERT INTO `masinijada`.`mesto_odrzavanja` (`id_mesta`) VALUES (3);
465 INSERT INTO `masinijada`.`mesto_odrzavanja` (`id_mesta`) VALUES (4);
466 INSERT INTO `masinijada`.`mesto_odrzavanja` (`id_mesta`) VALUES (5);
467 INSERT INTO `masinijada`.`mesto_odrzavanja` (`id_mesta`) VALUES (6);
468
469 COMMIT;
470
471
472 -- -----
473 -- Data for table `masinijada`.`disciplina`
474 -- -----
475 START TRANSACTION;
476 USE `masinijada`;

```



```

477 INSERT INTO 'masinijada', 'disciplina' ('id_disciplina', 'id_mesta', 'naziv', 'prosecno_trajanje') VALUES (1, 3, 'Rukomet', '40min');
478 INSERT INTO 'masinijada', 'disciplina' ('id_disciplina', 'id_mesta', 'naziv', 'prosecno_trajanje') VALUES (2, 1, 'Kosarka', '1h');
479 INSERT INTO 'masinijada', 'disciplina' ('id_disciplina', 'id_mesta', 'naziv', 'prosecno_trajanje') VALUES (3, 1, 'Fudbal', '1h');
480 INSERT INTO 'masinijada', 'disciplina' ('id_disciplina', 'id_mesta', 'naziv', 'prosecno_trajanje') VALUES (4, 2, 'Biohemija', '3h');
481
482 COMMIT;
483
484
485 -----
486 -- Data for table 'masinijada'. 'grad'
487 -----
488 START TRANSACTION;
489 USE 'masinijada';
490 INSERT INTO 'masinijada', 'grad' ('ptt', 'naziv') VALUES (34000, 'Kragujevac');
491 INSERT INTO 'masinijada', 'grad' ('ptt', 'naziv') VALUES (11000, 'Beograd');
492 INSERT INTO 'masinijada', 'grad' ('ptt', 'naziv') VALUES (11400, 'Mladenovac');
493
494 COMMIT;
495
496
497 -----
498 -- Data for table 'masinijada'. 'masinijada'
499 -----
500 START TRANSACTION;
501 USE 'masinijada';
502 INSERT INTO 'masinijada', 'masinijada' ('id_masinijade', 'godina_odrzavanja', 'broj_ucesnika', 'ptt') VALUES (1, 2019, 5000, 34000);
503 INSERT INTO 'masinijada', 'masinijada' ('id_masinijade', 'godina_odrzavanja', 'broj_ucesnika', 'ptt') VALUES (2, 2020, 5500, 11000);
504
505 COMMIT;
506
507
508 -----
509 -- Data for table 'masinijada'. 'nagrada'
510 -----
511 START TRANSACTION;
512 USE 'masinijada';
513 INSERT INTO 'masinijada', 'nagrada' ('id_nagrade', 'id_masinijade', 'mesto', 'disciplina') VALUES (1, 1, 1, 'Rukomet');
514 INSERT INTO 'masinijada', 'nagrada' ('id_nagrade', 'id_masinijade', 'mesto', 'disciplina') VALUES (2, 1, 2, 'Rukomet');
515 INSERT INTO 'masinijada', 'nagrada' ('id_nagrade', 'id_masinijade', 'mesto', 'disciplina') VALUES (3, 1, 3, 'Rukomet');
516 INSERT INTO 'masinijada', 'nagrada' ('id_nagrade', 'id_masinijade', 'mesto', 'disciplina') VALUES (4, 2, 1, 'Kosarka');
517 INSERT INTO 'masinijada', 'nagrada' ('id_nagrade', 'id_masinijade', 'mesto', 'disciplina') VALUES (5, 2, 2, 'Kosarka');
518 INSERT INTO 'masinijada', 'nagrada' ('id_nagrade', 'id_masinijade', 'mesto', 'disciplina') VALUES (6, 2, 3, 'Kosarka');
519
520
521 -----
522 -- Data for table 'masinijada'. 'fakultet'
523 -----
524
525 START TRANSACTION;
526 USE 'masinijada';
527 INSERT INTO 'masinijada', 'fakultet' ('id_fakulteta', 'drzava', 'naziv', 'broj_studenata', 'broj_pojavljivanja', 'ptt') VALUES (1, 'Srbija', 'Fakultet Inzenjerskih nauka', 1100, 12, 34000);
528 INSERT INTO 'masinijada', 'fakultet' ('id_fakulteta', 'drzava', 'naziv', 'broj_studenata', 'broj_pojavljivanja', 'ptt') VALUES (2, 'Srbija', 'Masinski Fakultet Beograd', 3000, 9, 11000);
529
530 COMMIT;
531
532
533 -----
534 -- Data for table 'masinijada'. 'dobiya'
535 -----
536
537 START TRANSACTION;
538 USE 'masinijada';
539 INSERT INTO 'masinijada', 'dobiya' ('id_nagrade', 'id_fakulteta') VALUES (1, 1);
540 INSERT INTO 'masinijada', 'dobiya' ('id_nagrade', 'id_fakulteta') VALUES (2, 1);
541 INSERT INTO 'masinijada', 'dobiya' ('id_nagrade', 'id_fakulteta') VALUES (3, 1);
542 INSERT INTO 'masinijada', 'dobiya' ('id_nagrade', 'id_fakulteta') VALUES (4, 2);
543 INSERT INTO 'masinijada', 'dobiya' ('id_nagrade', 'id_fakulteta') VALUES (5, 2);
544 INSERT INTO 'masinijada', 'dobiya' ('id_nagrade', 'id_fakulteta') VALUES (6, 1);
545
546 COMMIT;
547
548
549 -----
550 -- Data for table 'masinijada'. 'gledalac'
551 -----
552 START TRANSACTION;
553 USE 'masinijada';
554 INSERT INTO 'masinijada', 'gledalac' ('JMBG', 'ime', 'prezime') VALUES ('190009', 'Miro', 'Ivanovic');
555 INSERT INTO 'masinijada', 'gledalac' ('JMBG', 'ime', 'prezime') VALUES ('151000', 'Nikola', 'Brosevic');
556
557 COMMIT;
558
559
560 -----
561 -- Data for table 'masinijada'. 'karta'
562 -----
563 START TRANSACTION;
564 USE 'masinijada';
565 INSERT INTO 'masinijada', 'karta' ('redni_broj', 'broj_dana') VALUES (1, 3);
566 INSERT INTO 'masinijada', 'karta' ('redni_broj', 'broj_dana') VALUES (2, 3);
567 INSERT INTO 'masinijada', 'karta' ('redni_broj', 'broj_dana') VALUES (3, 1);
568 INSERT INTO 'masinijada', 'karta' ('redni_broj', 'broj_dana') VALUES (4, 2);
569 INSERT INTO 'masinijada', 'karta' ('redni_broj', 'broj_dana') VALUES (5, 1);
570
571 COMMIT;

```

```

574 -- -----
575 -- Data for table 'masinijada'. 'kupuje'
576 -- -----
577 START TRANSACTION;
578 USE 'masinijada';
579 INSERT INTO 'masinijada'. 'kupuje' ('redni_broj', 'JMBG', 'popust') VALUES (1, '190099', NULL);
580 INSERT INTO 'masinijada'. 'kupuje' ('redni_broj', 'JMBG', 'popust') VALUES (2, '251000', '15%');
581
582 COMMIT;
583
584
585 -- -----
586 -- Data for table 'masinijada'. 'ucesnik_masinijade'
587 -- -----
588 START TRANSACTION;
589 USE 'masinijada';
590 INSERT INTO 'masinijada'. 'ucesnik_masinijade' ('id_ucesnika', 'ime', 'prezime', 'ptt') VALUES (1, 'Petar', 'Petrovic', 34000);
591 INSERT INTO 'masinijada'. 'ucesnik_masinijade' ('id_ucesnika', 'ime', 'prezime', 'ptt') VALUES (2, 'Milan', 'Milanovic', 11000);
592 INSERT INTO 'masinijada'. 'ucesnik_masinijade' ('id_ucesnika', 'ime', 'prezime', 'ptt') VALUES (3, 'Danica', 'Mlric', 11000);
593 INSERT INTO 'masinijada'. 'ucesnik_masinijade' ('id_ucesnika', 'ime', 'prezime', 'ptt') VALUES (4, 'Matea', 'Belojevic', 34000);
594 INSERT INTO 'masinijada'. 'ucesnik_masinijade' ('id_ucesnika', 'ime', 'prezime', 'ptt') VALUES (5, 'Nikola', 'Mirovic', 11000);
595 INSERT INTO 'masinijada'. 'ucesnik_masinijade' ('id_ucesnika', 'ime', 'prezime', 'ptt') VALUES (6, 'Anastasija', 'Milenkovic', 11000);
596 INSERT INTO 'masinijada'. 'ucesnik_masinijade' ('id_ucesnika', 'ime', 'prezime', 'ptt') VALUES (7, 'Petko', 'Mihajlovic', 34000);
597
598 COMMIT;
599
600
601 -- -----
602 -- Data for table 'masinijada'. 'organizator'
603 -- -----
604 START TRANSACTION;
605 USE 'masinijada';
606 INSERT INTO 'masinijada'. 'organizator' ('id_ucesnika', 'radno_iskustvo') VALUES (6, 3);
607 INSERT INTO 'masinijada'. 'organizator' ('id_ucesnika', 'radno_iskustvo') VALUES (7, 5);
608
609 COMMIT;
610
611
612 -- -----
613 -- Data for table 'masinijada'. 'predaje'
614 -- -----
615 START TRANSACTION;
616 USE 'masinijada';
617 INSERT INTO 'masinijada'. 'predaje' ('id_ucesnika', 'id_fakulteta') VALUES (4, 2);
618 INSERT INTO 'masinijada'. 'predaje' ('id_ucesnika', 'id_fakulteta') VALUES (5, 1);
619
620 COMMIT;
621
622
623 -- -----
624 -- Data for table 'masinijada'. 'sektor'
625 -- -----
626 START TRANSACTION;
627 USE 'masinijada';
628 INSERT INTO 'masinijada'. 'sektor' ('id_sektora') VALUES (1);
629 INSERT INTO 'masinijada'. 'sektor' ('id_sektora') VALUES (2);
630
631 COMMIT;
632
633
634 -- -----
635 -- Data for table 'masinijada'. 'pripada'
636 -- -----
637 START TRANSACTION;
638 USE 'masinijada';
639 INSERT INTO 'masinijada'. 'pripada' ('id_ucesnika', 'id_sektora') VALUES (6, 2);
640 INSERT INTO 'masinijada'. 'pripada' ('id_ucesnika', 'id_sektora') VALUES (7, 1);
641
642 COMMIT;
643
644
645 -- -----
646 -- Data for table 'masinijada'. 'profesor'
647 -- -----
648 START TRANSACTION;
649 USE 'masinijada';
650 INSERT INTO 'masinijada'. 'profesor' ('id_ucesnika', 'godine_iskustva') VALUES (4, 15);
651 INSERT INTO 'masinijada'. 'profesor' ('id_ucesnika', 'godine_iskustva') VALUES (5, 12);
652
653 COMMIT;
654
655
656 -- -----
657 -- Data for table 'masinijada'. 'sala'
658 -- -----
659 START TRANSACTION;
660 USE 'masinijada';
661 INSERT INTO 'masinijada'. 'sala' ('id_mesta', 'dostupna_mesta', 'snaga_ozvucenja', 'dimenzije') VALUES (5, 250, '300W', '30x55');
662
663 COMMIT;

```

```

666 -- -----
667 -- Data for table 'masinijada'. 'se_takmici'
668 -- -----
669 START TRANSACTION;
670 USE 'masinijada';
671 INSERT INTO 'masinijada'. 'se_takmici' ('id_uesnika', 'id_discipline') VALUES (1, 2);
672 INSERT INTO 'masinijada'. 'se_takmici' ('id_uesnika', 'id_discipline') VALUES (2, 2);
673 INSERT INTO 'masinijada'. 'se_takmici' ('id_uesnika', 'id_discipline') VALUES (1, 1);
674 INSERT INTO 'masinijada'. 'se_takmici' ('id_uesnika', 'id_discipline') VALUES (3, 3);
675 INSERT INTO 'masinijada'. 'se_takmici' ('id_uesnika', 'id_discipline') VALUES (2, 1);
676
677 COMMIT;
678
679
680 -- -----
681 -- Data for table 'masinijada'. 'student'
682 -- -----
683 START TRANSACTION;
684 USE 'masinijada';
685 INSERT INTO 'masinijada'. 'student' ('id_uesnika', 'id_fakulteta', 'prosek', 'godina') VALUES (1, 1, 8.5, 3);
686 INSERT INTO 'masinijada'. 'student' ('id_uesnika', 'id_fakulteta', 'prosek', 'godina') VALUES (2, 1, 9, 4);
687 INSERT INTO 'masinijada'. 'student' ('id_uesnika', 'id_fakulteta', 'prosek', 'godina') VALUES (3, 2, 8.7, 2);
688
689 COMMIT;
690
691
692 -- -----
693 -- Data for table 'masinijada'. 'teren'
694 -- -----
695 START TRANSACTION;
696 USE 'masinijada';
697 INSERT INTO 'masinijada'. 'teren' ('id_mesta', 'dostupna_mesta', 'dimenzije') VALUES (1, 100, '25x50');
698 INSERT INTO 'masinijada'. 'teren' ('id_mesta', 'dostupna_mesta', 'dimenzije') VALUES (3, 120, '30x55');
699
700 COMMIT;
701
702
703 -- -----
704 -- Data for table 'masinijada'. 'ucestvuje'
705 -- -----
706 START TRANSACTION;
707 USE 'masinijada';
708 INSERT INTO 'masinijada'. 'ucestvuje' ('id_masinijade', 'id_uesnika') VALUES (1, 1);
709 INSERT INTO 'masinijada'. 'ucestvuje' ('id_masinijade', 'id_uesnika') VALUES (1, 2);
710 INSERT INTO 'masinijada'. 'ucestvuje' ('id_masinijade', 'id_uesnika') VALUES (1, 3);
711 INSERT INTO 'masinijada'. 'ucestvuje' ('id_masinijade', 'id_uesnika') VALUES (2, 6);
712 INSERT INTO 'masinijada'. 'ucestvuje' ('id_masinijade', 'id_uesnika') VALUES (2, 7);
713
714 COMMIT;
715
716 -- -----
717 -- Data for table 'masinijada'. 'ucionica'
718 -- -----
719 -- -----
720 START TRANSACTION;
721 USE 'masinijada';
722 INSERT INTO 'masinijada'. 'ucionica' ('id_mesta', 'broj_klupa') VALUES (2, 30);
723
724 COMMIT;

```

6. Implementacija projektovane baze podataka sa testnim podacima u MySQL

[1] <https://dev.mysql.com/doc/workbench/en/wb-getting-started-tutorial-creating-a-model.html>

[2] <https://dev.mysql.com/doc/>