



MODEL ENTITETA I ODNOSA

Baze podataka 1

dr Miloš CVETANOVIĆ

Dizajn baze podataka



1. Inicijalna faza – karakterisanje zahteva za podacima iz perspektive krajnjih korisnika
2. Druga faza – projektovanje baze podataka (konceptualni dizajn)
 - izbor modela podataka i primena konceptata odabranog modela podataka
 - prevođenje zahteva za podacima u konceptualni model baze podataka (potrebno je razumeti operacija tj. transakcije koje će se obavljati nad podacima)
3. Finalna faza – prevođenje sa apstraktnog (konceptulanog) modela na „niži“ model
 - logički dizajn (odlučivanje o **šemi baze podataka**)
određivanje „dobre“ kolekcije **relacionih šema**, tj. relacija na osnovu poslovne logike/pravila/ograničenja i očekivanih podataka i na osnovu inženjerskih odluka o rasporedu atributa po relacionim šemama
 - fizički dizajn (odlučivanje o fizičkom rasporedu baze podataka)
distribucija podataka i načini pristupa podacima

Dizajn alternative

- Prilikom dizajna baze podatak treba izbeći dva najveća problema: redundantnost podataka, nepotpunost podataka
- Izbegavanje lošeg dizajna nije dovoljno. Postoji veliki broj dobrih dizajna -> treba odabrati

Dizajn pristupi

- Odozgo naniže – Model entiteta i odnosa (ER model)
- Odozdo naviše – Normalizaciona teorija

Model entiteta i odnosa



- Entity Relationship model – ER modela
- Omogućava sagledavanje potreba i ustrojstva podataka
- ER model koristi tri osnovna koncepta:
 1. skup entiteta
 2. skup odnosa
 3. skup atributa
- Entiteti – ima fiksni broj atributa (koje poseduju svi članovi skupa entiteta)
- Veze – asocijacija između dva ili više entiteta (n -arni odnos, $n > 1$), može da ima attribute
- Atributi – prosti, kompozitni, više-vrednosni, izvedeni
- ER model je grafička notacija (postoji više različitih) koja najčešće obuhvata simbole za:
 1. jake entiteta (pravougaonik sa navedenim nazivom i atributima)
 2. slabe entitete (kao i jak entitet izuzev što je oivičen duplom linijom)
 3. zavisnosti – direktne odnose (usmerena isprekidana ili puna linija)
 4. veze – indirektne odnose (romb sa navedenim nazivom i opcionalnim atributima)
 5. napredne koncepte: specijalizacija, agregacija (trougao, uokvirena veza)
 6. dodatno specificiranje odnosa: kardinalnosti (par zagrade unutar kojih su dva broja)



Fudbalski savez

Posmatrani sistem je fudbalski savez koji evidentira utakmice odigrane u toku jedne sezone. Za svaku utakmicu se trajno čuvaju podaci o tome koji timovi su igrali, u kom kolu je ta utakmica odigrana (i koje godine), kakav je ishod utakmice bio, koji su sve igrači igrali i na kojim pozicijama. Za svakog fudbalera se pamti ime i tim za koji igra, dok se za timove pamti naziv i mesto iz koga dolaze. Pretpostavka je da fudbaleri ne mogu da menjaju tim u kome igraju, u toku sezone. Takođe je potrebno trajno evidentirati svaki postignuti gol kao i to koji fudbaler ga je postigao, u kom minutu i koji je to gol bio po redu na posmatranoj utakmici. Pored golova trajno se evidentiraju i svi kartoni, žuti i crveni, dodeljeni na utakmici i to kom fudbaleru i u kom minutu.



FUDBALER -> Ime, Naziv tima

Ime	Naziv tima
Edwin van der Sar	Manchester United
Rio Ferdinand	Manchester United
Nemanja Vidic	Manchester United
Wayne Rooney	Manchester United
Darren Fletcher	Manchester United
Cesc Fabregas	Arsenal
Dejan Stankovic	Inter
Lionel Messi	Barselona
Zlatan Ibrahimovic	Barselona
Andres Iniesta	Barselona

Relacioni model:

- relacija (tabela)
- atributi (kolone)
- torke (redovi)

relacioni šema, tj. relacija FUDBALER (Ime, Naziv)

instanca relacije šeme sadrži elemente koji se zovu torke i predstavljene su redovima u tabeli

Redosled toki unutar relacije je nevažan (proizvoljan je)

domen atributa -> skup mogućih vrednosti atributa

vrednosti atributa moraju biti atomične

specijalna „vrednost“ **null** je član svakog domena null označava da je vrednost „nepoznata“

šema baze – logička struktura baze

instanca baze – „snimak“ podataka u bazi u trenutku x

Šta ako tim promeni naziv?

Tim treba da ima i podatak u kom gradu se nalazi.

Da li se može „uštedeti“ prostor? ...da kodiramo ili „šifriramo“ podatak o timu?

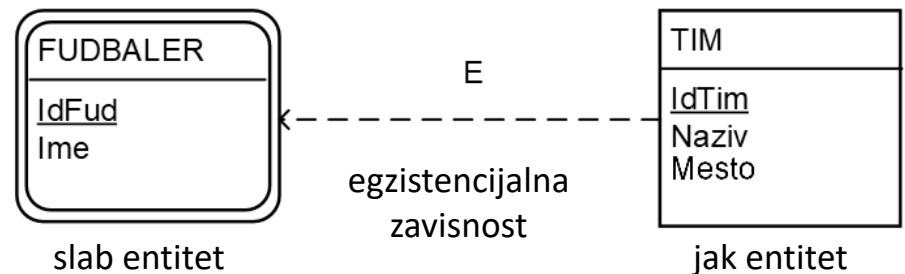


FUDBALER -> Ime, Šifra tima

Ime	Šifra tima
Edwin van der Sar	1
Rio Ferdinand	1
Nemanja Vidic	1
Wayne Rooney	1
Darren Fletcher	1
Cesc Fabregas	2
Dejan Stankovic	3
Lionel Messi	4
Zlatan Ibrahimovic	4
Andres Iniesta	4

Šifra tima	Naziv tima	Mesto
1	Manchester United	Manchester
2	Arsenal	London
3	Inter	Milano
4	Barselona	Barselona

primarni ključ
kandidat ključ – alternativni ključ
strani ključ



Šta koja šifra tima znači?

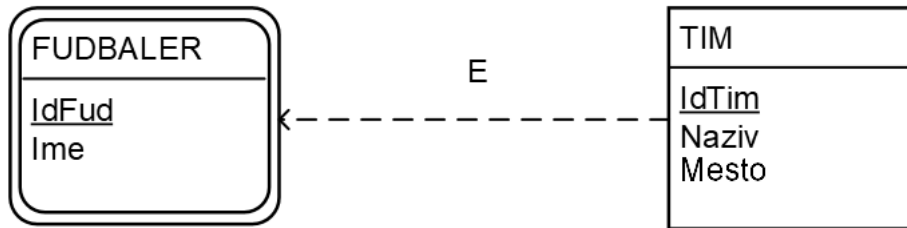
TIM -> Šifra tima, Naziv, Mesto

Referencijalni integritet?

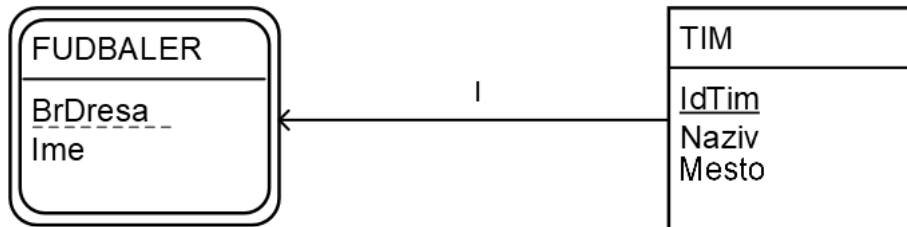
Identifikacioni integritet?

Opšti integritet?

TIM (IdTim, Naziv, Mesto)
FUDBALER (IdFud, Ime, IdTim)

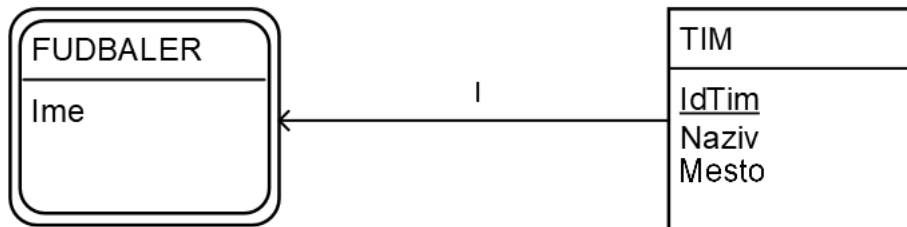


TIM (IdTim, Naziv, Mesto)
FUDBALER (IdFud, Ime, IdTim)



TIM (IdTim, Naziv, Mesto)
FUDBALER (IdTim, BrDresa, Ime)

diskriminator



TIM (IdTim, Naziv, Mesto)
FUDBALER (IdTim, Ime)

maksimalno jedan fudbaler u timu?

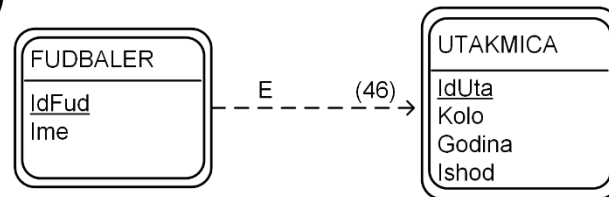
Fudbaler postoji samo ako je poznat tim za koji igra?

Fudbaler je deo tima za koji igra?

UTAKMICA -> Šifra utakmice, Kolo, Godina, Ishod, Šifra domaćeg tima, Šifra gostujućeg tima

UTAKMICA je egzistencijalno zavisna od TIM (i to dva tima)

Kako evidentirati podatak da fudbaler igra na utakmici?



UTAKMICA -> Šifra utakmice, Kolo, Godina, Ishod, Šifra domaćeg tima, Šifra gostujućeg tima,
+ Šifra igrača 1, Šifra igrača 2, ... Šifra igrača X

Koji je maksimalan broj fudbalera koje treba evidentirati na utakmici? 2 x 11? Izmene? 2 x 23?
Ako ne igraju svi? Varijabilni broj kolona u tabeli? ...ali tabela treba da ima fiksnu strukturu!

Tabela koja evidentira koji fudbaler igra na kojoj utakmici
IGRA (SifF, SifU)

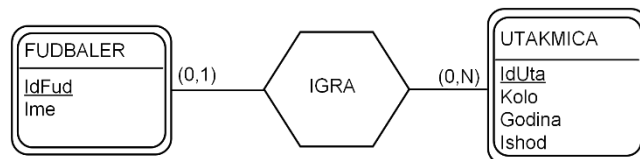
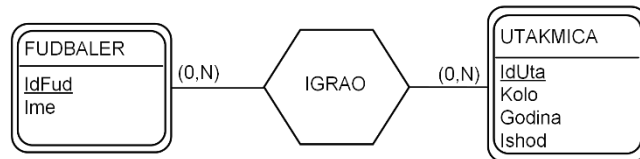


Tabela koja evidentira koji fudbaler je igrao na kojoj utakmici
IGRAO (SifF, SifU)



Šifra fudbalera	Šifra utakmice
1	1
2	1
3	2
2	2

Jedan fudbaler na **jednoj** utakmici
Jedan fudbaler na **više** utakmica

IGRA? Ili IGRAO?

Fudbalski savez

Posmatrani sistem je fudbalski savez koji evidentira utakmice odigrane u toku jedne sezone. Za svaku utakmicu se trajno čuvaju podaci o tome koji timovi su igrali, u kom kolu je ta utakmica odigrana (i koje godine), kakav je ishod utakmice bio, koji su sve igrači igrali i na kojim pozicijama. Za svakog fudbalera se pamti ime i tim za koji igra, dok se za timove pamti naziv i mesto iz koga dolaze. Pretpostavka je da fudbaleri ne mogu da menjaju tim u kome igraju, u toku sezone. Takođe je potrebno trajno evidentirati svaki postignuti gol kao i to koji fudbaler ga je postigao, u kom minutu i koji je to gol bio po redu na posmatranoj utakmici. Pored golova trajno se evidentiraju i svi kartoni, žuti i crveni, dodeljeni na utakmici i to kom fudbaleru i u kom minutu.

Da li fudbaler može da postigne gol na utakmici na kojoj nije igrao?

FUDBALER (IdFud, Ime, IdTim)

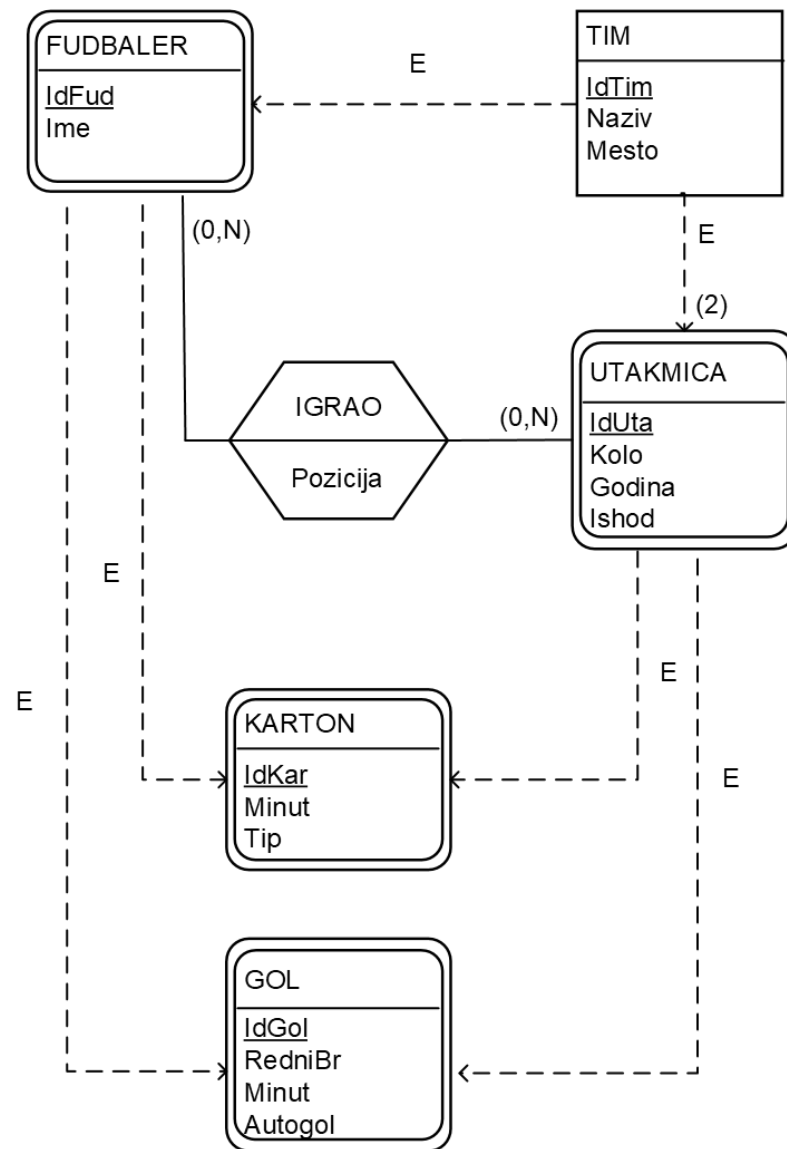
TIM (IdTim, Naziv, Mesto);

IGRAO (IdFud, IdUta, Pozicija);

GOL (IdGol, IdUta, IdFud, RedniBr, Minut, Autogol);

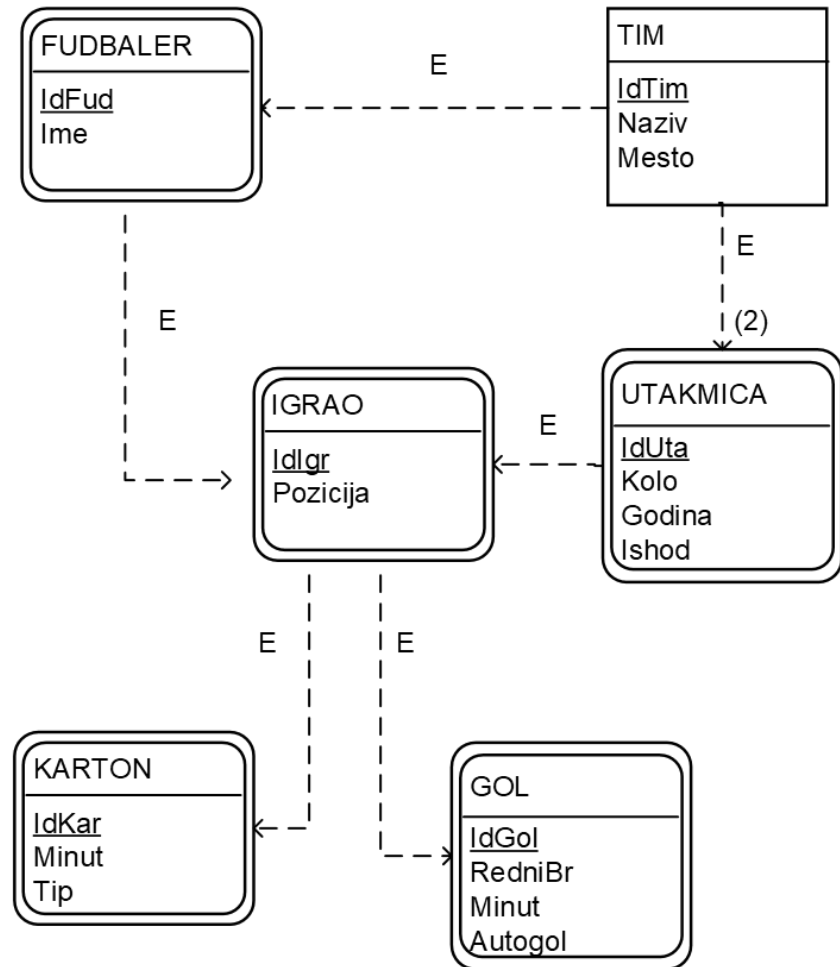
KARTON (IdKar, IdUta, IdFud, Tip, Minut);

UTAKMICA (IdUta, IdTDomaci, IdTGost, Kolo, Ishod, Godina);



Fudbalski savez

Posmatrani sistem je fudbalski savez koji evidentira utakmice odigrane u toku jedne sezone. Za svaku utakmicu se trajno čuvaju podaci o tome koji timovi su igrali, u kom kolu je ta utakmica odigrana (i koje godine), kakav je ishod utakmice bio, koji su sve igrači igrali i na kojim pozicijama. Za svakog fudbalera se pamti ime i tim za koji igra, dok se za timove pamti naziv i mesto iz koga dolaze. Pretpostavka je da fudbaleri ne mogu da menjaju tim u kome igraju, u toku sezone. Takođe je potrebno trajno evidentirati svaki postignuti gol kao i to koji fudbaler ga je postigao, u kom minutu i koji je to gol bio po redu na posmatranoj utakmici. Pored golova trajno se evidentiraju i svi kartoni, žuti i crveni, dodeljeni na utakmici i to kom fudbaleru i u kom minutu.



Da li je pozicija atribut ili entitet?

FUDBALER (IdFud, Ime, IdTim)

TIM (IdTim, Naziv, Mesto);

IGRAO (IdIgr, IdFud, IdUta, Pozicija);

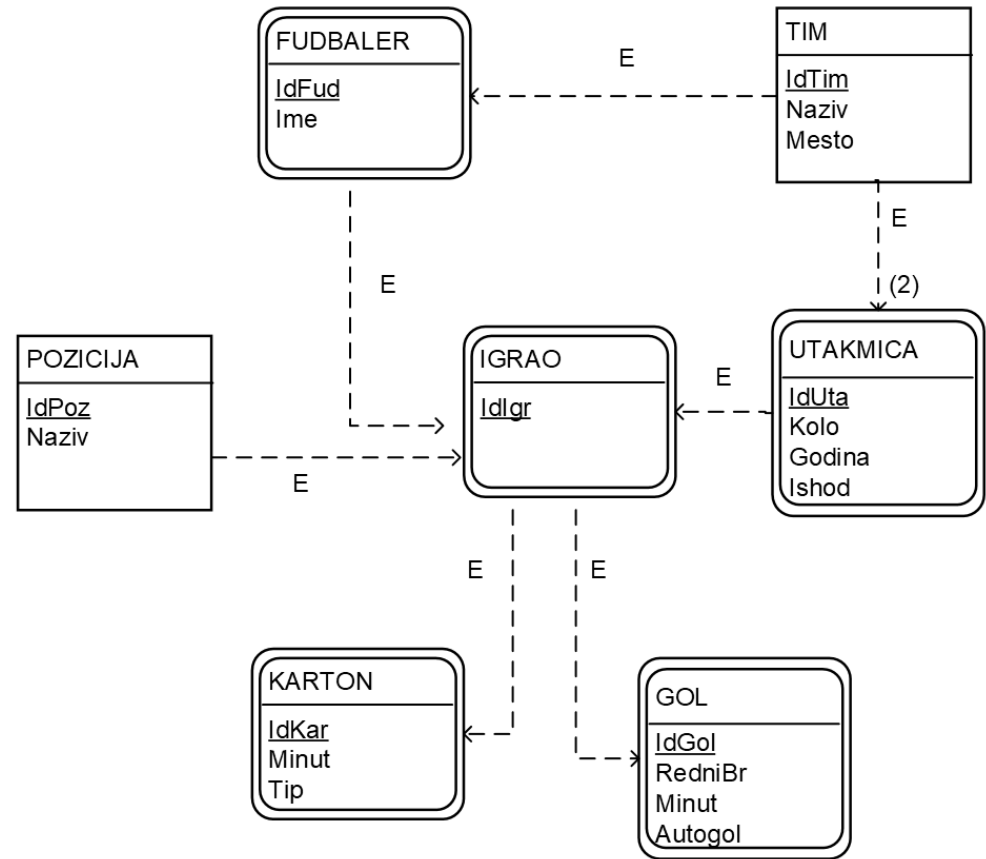
GOL (IdGol, IdIgr, RedniBr, Minut, Autogol);

KARTON (IdKar, IdIgr, Tip, Minut);

UTAKMICA (IdUta, IdTDomaci, IdTGost, Kolo, Godina, Ishod);

Fudbalski savez

Posmatrani sistem je fudbalski savez koji evidentira utakmice odigrane u toku jedne sezone. Za svaku utakmicu se trajno čuvaju podaci o tome koji timovi su igrali, u kom kolu je ta utakmica odigrana (i koje godine), kakav je ishod utakmice bio, koji su sve igrači igrali i na kojim pozicijama. Za svakog fudbalera se pamti ime i tim za koji igra, dok se za timove pamti naziv i mesto iz koga dolaze. Pretpostavka je da fudbaleri ne mogu da menjaju tim u kome igraju, u toku sezone. Takođe je potrebno trajno evidentirati svaki postignuti gol kao i to koji fudbaler ga je postigao, u kom minutu i koji je to gol bio po redu na posmatranoj utakmici. Pored golova trajno se evidentiraju i svi kartoni, žuti i crveni, dodeljeni na utakmici i to kom fudbaleru i u kom minutu.



Pozicija je “šifrirana”

FUDBALER (IdFud, Ime, IdTim)

TIM (IdTim, Naziv, Mesto);

POZICIJA (IdPoz, Naziv)

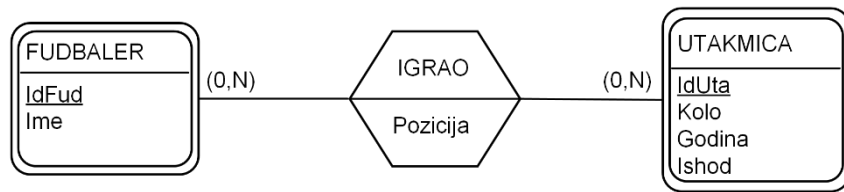
IGRAO (IdIgr, IdFud, IdUta, IdPoz);

GOL (IdGol, IdIgr, RedniBr, Minut, Autogol);

KARTON (IdKar, IdIgr, Tip, Minut);

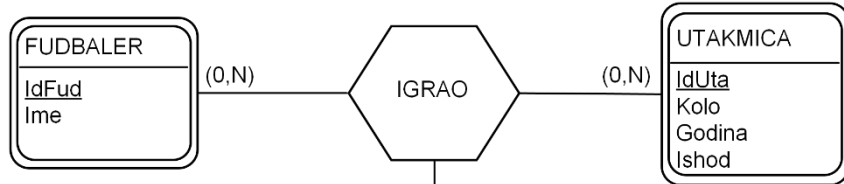
UTAKMICA (IdUta, IdTDomaci, IdTGost, Kolo, Godina, Ishod);

Da li fudbaler može da igra na više pozicija na jednoj utakmici?

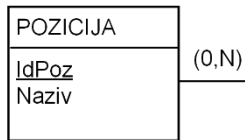


FUDBALER (IdFud, ...)
UTAKMICA (IdUta, ...)
IGRAO (IdFud, IdUta, Pozicija)

U vezama, učesnik sa (0,1) definiše PK



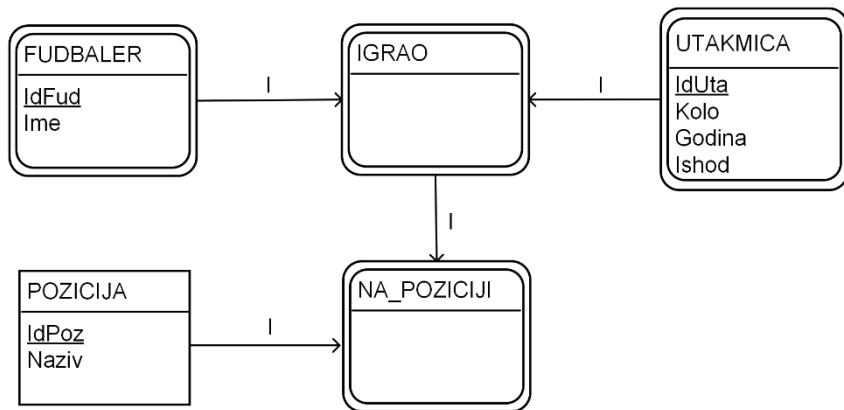
FUDBALER (IdFud, ...)
UTAKMICA (IdUta, ...)
POZICIJA (IdPoz, ...)
IGRAO (IdFud, IdUta, IdPoz)



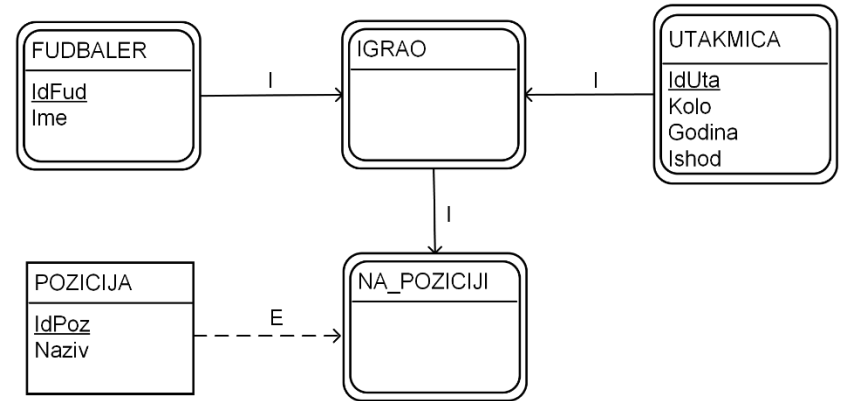
Fudbaler na više pozicija na jednoj utakmici.

PK: {IdFud, IdUta} -> fudbaler na jednoj pozicija na jednoj utakmici

Slab entitet umesto veze? ...koristiti zavisnosti?



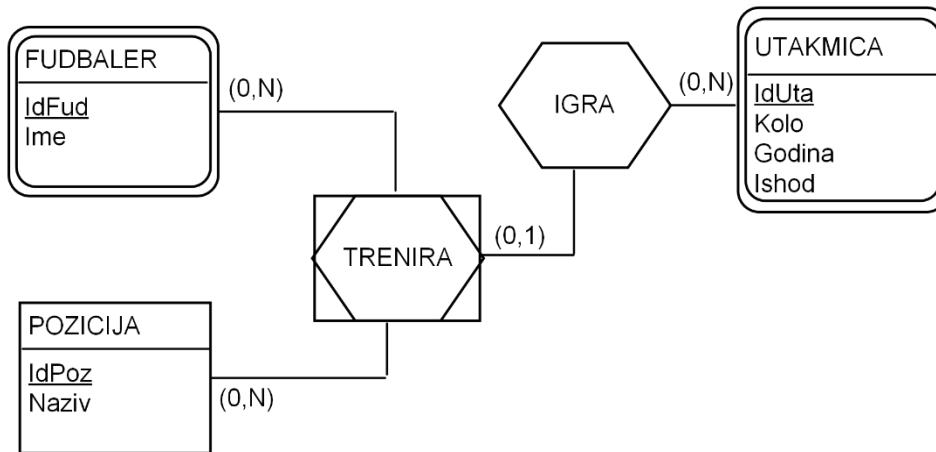
IGRAO (IdFud, IdUta)
NA_POZICIJI (IdFud, IdUta, IdPoz)



IGRAO (IdFud, IdUta)
NA_POZICIJI (IdFud, IdUta, IdPoz)



Agregacija – koncept da veza može biti učesnika u nekoj drugoj vezi



Fudbaler može na utakmici da igra više od pozicija, ali samo one koje inače trenira

UTAKMICA (IdUta, ...)

FUDBALER (IdFud, ...)

POZICIJA (IdPoz, ...)

TRENIRA (IdFud, IdPoz)

IGRA (IdFud, IdPoz, IdUta)



Veze -> trenutno stanje

Entiteti -> trenutno stanje + istorija

Zavisnosti -> egzistencijalna

identifikaciona <- Celina/deo

Ograničenja atributa ili skupa atributa:

1. jedinstvenost U – unique
2. obaveznost NN – not null

Pravila imenovanja:

- imenice (jedinina / množina)
- glagolski obilici

Razumljiv model?

MESTO (IdMes, Naziv)

POZICIJA (IdPoz, Naziv)

TIM (IdTim, Naziv, IdMes)

FUDBALER (IdFud, Ime, IdTim)

UTAKMICA (IdUta, Kolo, Godina, Ishod, IdTimD, IdTimG) KK: {IdTimD, IdTimG}

IGRANJE (Idlgr, IdFud, IdUta, IdPoz)

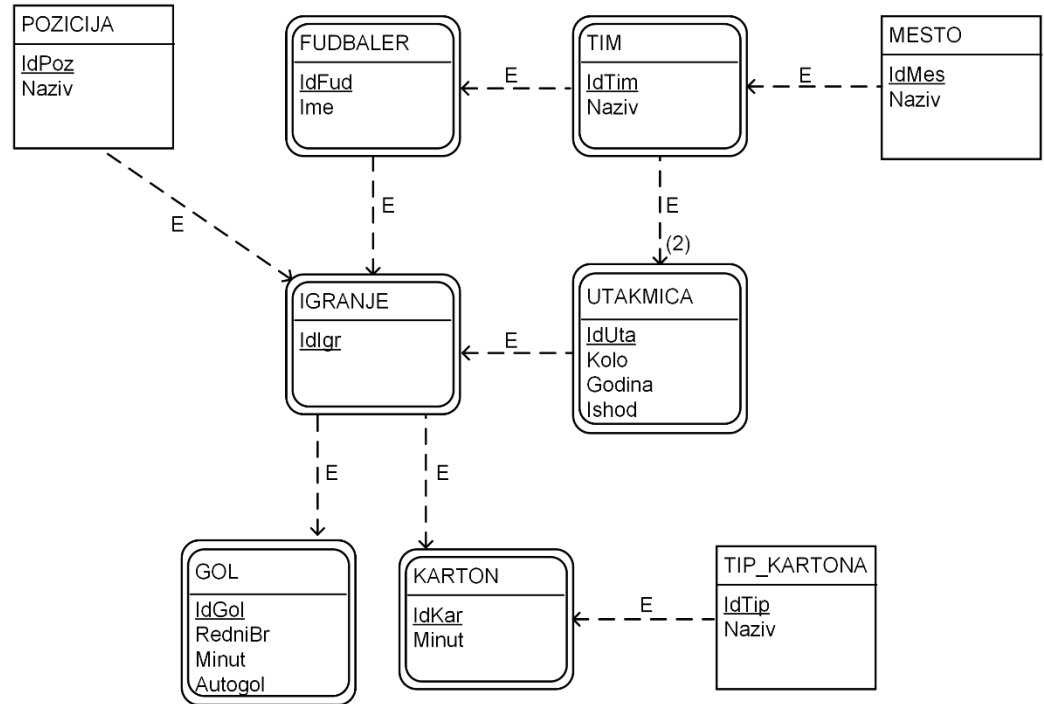
KK: {IdFud, IdUta}

TIP_KARTONA (IdTip, Naziv)

KARTON (IdKar, Minut, IdTip, Idlgr)

KK: {Idlgr, IdTip}

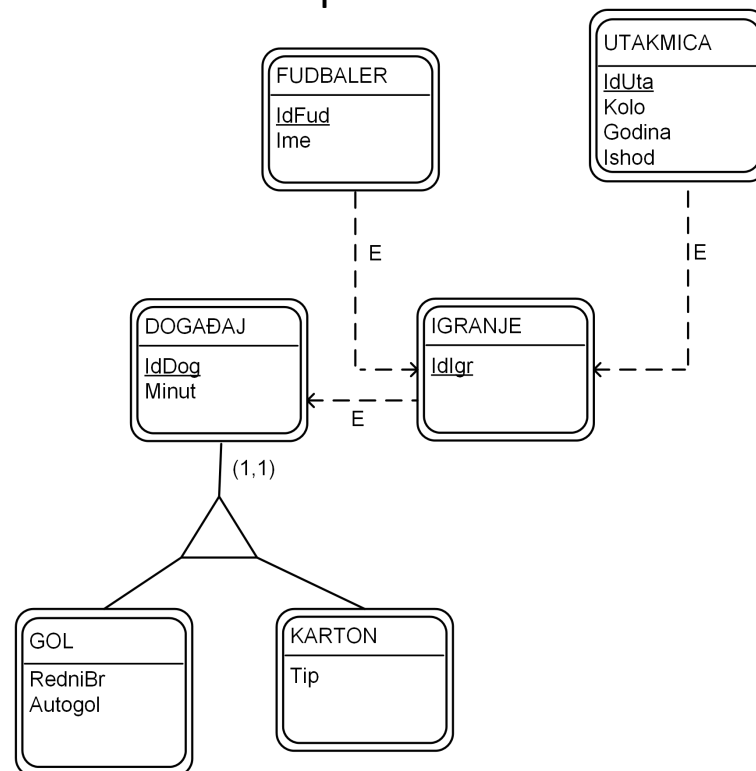
GOL (IdGol, RedniBr, Minut, Autogol, Idlgr)



Specijalizacija / generalizacija – koncept apstrahovanja zajedničkih atributa i/ili odnosa dva ili više skupova entiteta



UTAKMICA (IdUta, ...)
FUDBALER (IdFud, ...)
IGRANJE (Idlgr, ...)
DOGAĐAJ (IdDog, Idlgr, Minut)
GOL (IdDog, RedniBr, Autogol)
KARTON (IdDog, Tip)



Događaj na utakmici može biti ili gol ili karton -> ekskluzivna specijalizacija -> kardinalnost (x,1)

Inkluzivna specijalizacija -> događaj može biti istovremeno i gol i karton -> kardinalnost (x,2)

Parcijalna -> ekskluzivna (0,1), odnosno inkluzivna (0,2)

Totalna -> ekskluzivna (1,1) odnosno inkluzivna (1, 2)

Prevođenje u relacione šeme:

N+1 u N+1 – kao u primeru

N+1 u N – ne formira se relacione šema za nad entitet

N+1 u 1 – redukcija DOGAĐAJ (IdDog, Idlgr, Minut, **ŠtaJe**, RedniBr, Autogol, Tip)



Kardinalnosti – različite od (0,1) i (0, N)?

Provera kardinalnosti (0,1) i (0, N) – "jednostavna" ograničenja na nivou tabele

Proverava kardinalnosti (x, y), gde je $x > 0$, $y \notin \{1, N\}$ često zahteva:

1. Složena ograničenja na nivou tabele
2. Ograničenja na nivou baze podataka (nad više tabela)

Kardinalnost (1, 1) – egzistencijalna zavisnost?

