Alati za konceptualno modeliranje podataka

Antolović, Goran

Undergraduate thesis / Završni rad

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Pula / Sveučilište Jurja Dobrile u Puli**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:137:186290

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-06**

Repository / Repozitorij:

Digital Repository Juraj Dobrila University of Pula

Sveučilište Jurja Dobrile u Puli
Fakultet ekonomije i turizma
«Dr. Mijo Mirković»

GORAN ANTOLOVIĆ

ALATI ZA KONCEPTUALNO MODELIRANJE PODATAKA

Završni rad

Pula, 2016.

Sveučilište Jurja Dobrile u Puli
Fakultet ekonomije i turizma
«Dr. Mijo Mirković»

GORAN ANTOLOVIĆ

ALATI ZA KONCEPTUALNO MODELIRANJE PODATAKA

Završni rad

JMBAG: 0303001597, izvanredni student
Studijski smjer: Poslovna Informatika

Predmet: Baze Podataka
Mentor: Vanja Bevanda

Pula, veljača 2016.

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani Goran Antolović, kandidat za prvostupnika Poslovne Informatike ovime izjavljujem da je ovaj Završni rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Završnog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student:

U Puli, ____. ____. 2016.

_____

IZJAVA O KORIŠTENJU AUTORSKOG DJELA

Ja, Goran Antolović dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj završni rad pod nazivom Alati za konceptualno modeliranje podataka koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, _____ (datum)

Potpis

_____

Sadržaj

UVOD **1**

1. PODACI I MODELIRANJE PODATAKA **2**

2. KONCEPTUALNO(E-R) MODELIRANJE PODATAKA **4**

2.1. MODEL ENTITETI-VEZE **4**

2.2. OSNOVNI KONCEPTI E-R MODELA **5**

2.2.1. ENTITETI **5**

2.2.2. VEZE **7**

2.2.3. ATRIBUTI **13**

2.3. NOTACIJE E-R MODELA **14**

2.3.1. CHENOV PRIKAZ **14**

2.3.2. MARTINOV PRIKAZ **17**

2.3.3. BARKEROV PRIKAZ **20**

3. LOGIČKO MODELIRANJE PODATAKA **23**

3.1 PRETVARANJE KONCEPTUALNOG MODELA U LOGIČKI MODEL PODATAKA **24**

4. ALATI **29**

4.1. TOAD **29**

4.2. NAVICAT **33**

4.3. DBFORGE STUDIO **38**

4.4. USPOREDBA ALATA **42**

5. ZAKLJUČAK **44**

6. LITERATURA **45**

7. SAŽETAK **46**

UVOD

Svakim danom, informacijske se tehnologije sve brže razvijaju i napreduju. Samim time dolazi do potrebe za složenijim i zahtjevnijim informacijskim sustavima. Svaki informacijski sustav ima bazu podataka kojom raspolaže i iz koje izvlači podatke koji su mu potrebni za rad. Sama baza podataka nastaje tako da se, najprije, analiziraju i prikupe određeni podaci. Nakon analize, prikupljeni se podaci oblikuju. Prvo dolazi do konceptualnog oblikovanja, gdje se stvara konceptualni model samih podataka. Zatim se konceptualni model pretvara u relacijski model podataka, te tu dolazi do logičkog oblikovanja podataka. Nakon toga se stvara fizička shema, te tu dolazi do fizičkog oblikovanja podataka. Zatim, dolazi do implementacije podataka. Te se na kraju ti podaci, i sama baza podataka, testiraju.

Razvojem baze podataka, modeliranje podataka poprima sve veću ulogu i važnost. Dobar konceptualni model podataka ključan je za rad i razvoj baze podataka. Zbog povećanja važnosti i potrebe konceptualnog modela, sve više dolazi do razvoja alata sa grafičkim sučeljem, s kojima je izrada modela i baze podataka uvelike olakšana i pojednostavljena. Danas već postoji veliki broj alata za konceptualno modeliranje podataka koji se mogu spajati na razne sustave za upravljanje bazom podataka.

U ovom će radu najprije pisati o podacima i informacijama, te koji je proces modeliranja podataka i na koji se sve način ti podaci modeliraju. Zatim, će biti objašnjeno što je to konceptualno modeliranje, koji su njegovi koncepti i kakvi se grafički prikazi koriste. Na kraju će biti obrađen i objašnjen rad 3 alata za konceptualno modeliranje podataka: TOAD, NAVICAT I DBFORGE STUDIO, i ti će alati biti uspoređeni. Alati, osim konceptualnog, podržavaju i logičko modeliranje podataka, pa će prije opisa samih alata, biti spomenuto ponešto i o logičkom modeliranju podataka.

1. PODACI I MODELIRANJE PODATAKA

Napretkom i razvojem informacijske tehnologije javlja se potreba za sve složenijim i naprednijim informacijskim sustavima. Informacijski sustav je model poslovnog sustava. To je skup elemenata koji prikazuju dijelove stvarnog svijeta. Informacijski sustav čine "skup ljudi i opreme" povezanih aktivnostima prikupljanja, obrade, pohranjivanja i distribucije podataka i informacija, s krajnjim ciljem generiranja informacija u različitim problemskim situacijama i na različitim razinama organiziranosti nekog objektnog sustava vodeći računa o temeljnim ekonomskim načelima. Pod-modeli informacijskog sustava su: Model podataka- njime se naznačuje koji su podaci kojima organizacija raspolaže, Model procesa- opisuje kako se i na koji način obrađuju, prikupljaju i distribuiraju podaci, te Model izvršitelja odnosno resursa.

Podaci su simbolički prikaz elemenata stvarnog svijeta. Oni predstavljaju jedno obilježje nekog subjekta ili događaja. Opisuje li neki podatak stvarni svijet ili ne, ovisi o samom korisniku i njegovoj percepciji. Ako nekom podatku dodamo kontekst i predstavimo na određeni način, tada on postaje informacija. Informacija je, dakle, skup podataka koji imaju neko značenje. Podatak se odnosi na vrijednost obilježja, a razvrstavanje podataka po kategorijama čini tip nekog obilježja. Odnosi između tipova obilježja odnosno kategorija podataka su predmet modeliranja.

Modeliranje informacijskog sustava započinje utvrđivanjem i analizom zahtjeva korisnika za informacijama, a završava izgradnjom stabilne i prilagodljive baze podataka. Model podataka pokazuje koji su podaci kojima raspolaže organizacija. To je formalan i ujednačen način opisa podataka informacijskog sustava. Dakle, model podataka je formalan sustav i ima tri komponente: Skup koncepata za opis statičke strukture podataka, Skup koncepata za opis ograničenja podataka koji osigurava integritet podataka, te Skup operatora za mijenjanje podataka uz uvažavanje strukture i pravila ograničenja. Model podataka može biti Model sa potpuno kategoriziranim podacima i Model sa nepotpuno kategoriziranim podacima. U modelu sa potpuno kategoriziranim podacima svaki podatak pripada nekoj kategoriji. Dopuštene su kategorije određene unaprijed i ne mijenjaju se. U modelu s nepotpuno kategoriziranim podacima kategorije su dopuštene, ali nisu nužne, a podaci o kategorijama promatraju se kao podaci.

Podaci informacijskog sustava spremaju se u bazu podataka. Baza podataka je model podataka poslovnog sustava, koji je segment stvarnog svijeta. To je skup podataka koji su obrađeni u jednoj organizaciji. Baza podataka je, također, skup istovrsnih podataka koji imaju višestruku namjenu. Dakle, kao što smo prije već spomenuli, korisnik je taj koji određuje da li neki podatak opisuje ili ne stvarni svijet. Jednom korisniku, u njegovom poslu, nisu potrebni svi podaci nego može koristi samo jednu datoteku koja se nalazi u mnogo složenijoj strukturi podataka. Svaki korisnik može koristiti drugu datoteku, a sve te datoteke su dio iste baze podataka.

Da bi se baza podataka mogla implementirati, njen razvoj prije mora proći kroz različite faze. Određeni se podaci prvo analiziraju i prikupljuju. Podaci, nakon analize i prikupljanja kreću u oblikovanje. Oblikovanje baze podataka sastoji se od tri faze: konceptualno modeliranje, logičko modeliranje i fizičko modeliranje. U konceptualnom modelu se određuju entiteti, atributi i veze. Konceptualni model se pretvara u relacijski model tj. sastavljanja tablica. Tu dolazi do logičkog modeliranja. Javlja se, zatim, faza normalizacije gdje se nastale relacije prilagođavaju i popravljaju da se unutarnja osobina podataka bolje i kvalitetnije izrazi. To sve vodi do fizičkog modeliranja gdje se naredbama u jeziku SQL stvara fizički model. Tek sada je baza podataka spremna za implementaciju.

Oblikovanje podataka je vrlo bitno i ključno u razvoju baze podataka. Cilj modeliranja podataka je da se odredi pogodna struktura baze za određene podatke i da ti podaci budu međusobno nezavisni.

2. KONCEPTUALNO(E-R) MODELIRANJE PODATAKA

Modeliranje podataka je ključna faza u razvoju i implementaciji baze podataka. Isto je tako, konceptualno modeliranje podataka bitan i osnovni dio modeliranja podataka. "Konceptualno modeliranje polazi od specifikacija informacijskih zahtjeva, koje čine zahtjevi za strukturu podataka i zahtjevi za korištenjem podataka, a rezultira izrađenim konceptualnim modelom podataka."(Varga,M.,1994.) Konceptualnim modeliranjem, dakle, dobivamo konceptualni model podataka. On prikazuje stvarni svijet tj. prikazuje podatke koji su dio stvarnog svijeta. Konceptualni model opisuje podatke koji su dio strukture cijelog informacijskog sustava, a izgrađuju ga projektanti koji surađuju sa korisnicima kako bi bolje razumjeli koji su podaci i koje informacije važne i potrebne. Konceptualni se model ne može implementirati i neovisan je o implementaciji. Podaci, u konceptualnom modelu, opisuju jednu činjenicu i grupirani su na jednom mjestu, a međusobno su nezavisni sa podacima neke druge činjenice.

Za modeliranje konceptualnog modela koriste se koncepti: entiteti, veze i atributi, a za prikaz modela upotrebljavaju se dijagrami. Konceptualni model može se sastojati od jednog ili više dijagrama, a mora sadržavati i detaljan opis elemenata koji se prikazuju u dijagramima.

2.1. MODEL ENTITETI-VEZE

E-R model odnosno model entiteti-veze je model za konceptualno modeliranje podataka koji za opis konceptualnog modela podataka upotrebljava koncepte: entitet, veza i atribut. Dakle, E-R model proučava, opisuje i promatra stvarni svijet preko entiteta, odnosa između entiteta, i atributa entiteta ili njihovog odnosa. Za promatranje stvarnog svijeta koristi se apstrakcija pomoću koje se lakše uočava i odvaja ono što je bitno od onoga što je nebitno ili manje bitno. Za konceptualno modeliranje podataka u E-R modelu koriste se tri vrste ili postupka apstrakcije. To su klasifikacija, generalizacija i agregacija.

"Klasifikacija je vrsta apstrakcije u kojoj se entiteti klasificiraju, opisuju i grupiraju u klase, razrede odnosno tipove, prema zajedničkim obilježjima."(Varga,M.,1994.) Dakle, ako

u E-R modelu postoje neki entiteti koji imaju zajednička obilježja, oni se grupiraju u tip entiteta, ali vrijednost njihovih obilježja su i dalje različita. Klasifikacija se opisuje vezom "jest pojava". (npr. entiteti Ivica, Marko i Karla klasificiraju se kao tip entiteta učenici, ali oni imaju različiti datum rođenja, visinu, težinu, ocjene, spol) Obrnuti postupak od klasifikacije, gdje se klase ili tipovi entiteta odvajaju u entitete, zove se instantacija.

Generalizacija je slična postupku klasifikacije. To je vrsta apstrakcije gdje se tipovi entiteta niže razine spajaju u tipove entiteta više razine. Tip entiteta niže razine je podtip, a tip entiteta više razine je nadtip. Generalizacija se opisuje vezom "jest". (npr. tipovi entiteta nogometaš, rukometaš i tenisač generaliziraju se u tip entiteta sportaš. Obrnuti postupak od generalizacije, gdje tipovi entiteta prelaze iz nadtipova entiteta u podtipove entiteta, zove se specijalizacija.

"Agregacija je postupak ili vrsta apstakcije kojom se sastavlja novi pojam višeg stupnja, na temelju odnosa postojećih pojmova."(Varga,M.,1994.) Agregacija se koristi kao: agregacija jednostavnih atributa i agregacija entiteta u novi entitet. Agregacijom jednostavnih atributa formira se novi pojam od atributa koji opisuju entitet (npr. atributi br. indexa, ocjena opisuju entitet student). Agregacijom entiteta u novi entitet stvara se entitet koji predstavlja odnos više entiteta. (npr. plaćanje računa za struju-entiteti osoba, račun i struja formiraju se u entitet uplata)

2.2. OSNOVNI KONCEPTI MODELA ENTITETI-VEZE

Već je spomenuto da je E-R model za konceptualno modeliranje podataka model koji pri modeliranju i prikazivanju podataka koristi koncepte entitet, veza i atribut. E-R je skraćenica od eng. Entity-Relationship pa se ovaj model zove još i model entiteti-veze gdje veze pokazuju odnose između entiteta, a atributi opisuju entitete.

2.2.1. ENTITETI

Entitet je nešto što postoji i što se u stvarnom svijetu može identificirati. "To je stvaran ili apstraktan predmet ili događaj o kojemu se u informacijskom sustavu prikupljaju podaci."(Varga,M.,1994.) Dakle, možemo reći da je entitet neki objekt koji prikazuje stvarni svijet i čiji se podaci upisuju u bazu podataka. Može biti objekt koji ima mnogo instanci u

bazi podataka, može biti sastavljen od više atributa i može biti objekt kojeg sami modeliramo. Entitet nikako ne bi trebao biti korisnik baze podataka i izlaz iz baze podataka. Entitet se u E-R modelu prikazuje u kvadratnom obliku, **Slika 2.1.**

Slika 2.1. Simbol entiteta u E-R modelu

Svaki entitet ima svoje atribute. Apstraktinim postupkom klasifikacije grupiraju se, odnosno klasificiraju entiteti koji imaju zajednička obilježja tj. atribute te tako nastaje klasa ili tip entiteta. Tip entiteta je znači, skup entiteta koji imaju zajedničke atribute. Iako entiteti u tipu entiteta imaju zajedničke atribute, oni se razlikuju i moguće ih je prepoznati po vrijednostima atributa koje su različite za svaki entitet. Dakle, tip entiteta se prepoznaje i definira nazivom i popisom atributa, a pojedini entitet vrijednošću njegovih atributa. Za naziv tipa entiteta koristi se imenica, piše se u jednini i koriste se velika slova. Tip entiteta može biti jaki i slabi tip, a postoji i asocijacijski entitet, no o njima će više biti riječ kasnije u poglavlju o vezama. Postoje različiti tipovi entiteta, **Primjer 2.1.**

Primjeri tipova entiteta:

- ljudi (njihove uloge):
  - OSOBA - skup svih osoba
  - DJELATNIK - skup zaposlenih osoba u poduzeću
  - KUPAC - skup fizičkih ili pravnih osoba koji kupuju robu
- organizacije (organizirani skup ljudi):
  - PODUZEĆE - skup poduzeća u gradu, regiji, republici
  - AGENCIJA - skup poduzeća koja se bave agencijskim poslovima
- stvari (materijalni predmeti):
  - PROIZVOD - skup proizvoda koje proizvodi ili prodaje poduzeće
  - REZERVNI DIO - skup rezervnih dijelova
  - AUTOMOBIL - skup automobila registriranih u jednom gradu
- koncepti (nematerijalni predmeti):
  - RAČUN - skup računa jedne trgovine
  - RUTA - skup mogućih ruta avionskih letova
- mjesta (zemljopisna ili druga područja):
  - GRAD - skup gradova
  - URED - skup lokacija poduzeća
- događaji:
  - NARUDŽBA - skup narudžbi upućenih dobavljaču
  - PROJEKT - skup projekata
  - LET - skup letova avionske kompanije

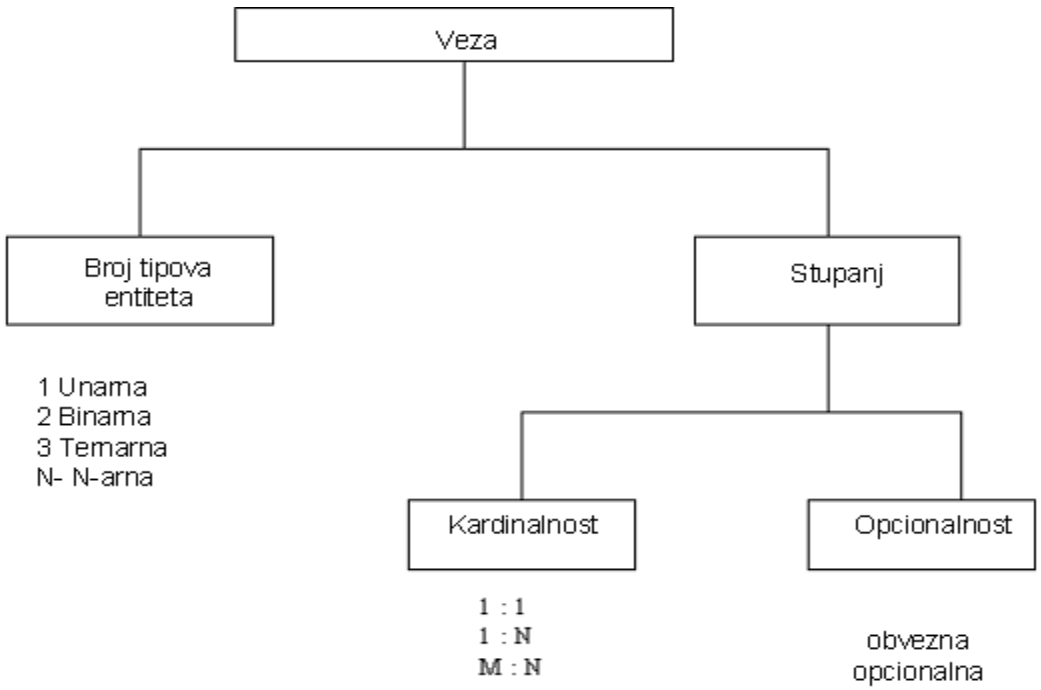
Primjer 2.1. Primjeri tipova entiteta

Izvor: "Varga,M.,1994.,Baze Podataka"

2.2.2 VEZE

Veza opisuje odnos između entiteta. Ona predstavlja agregaciju između dva ili više entiteta u jedan novi entitet tj. vezu. "Veza se opisuje matematičkom relacijom k>2 skupova entiteta, ne nužno različitih. Ako su $E_1,E_2,\ldots$, Ek skupovi entiteta, veza se opisuje skupom n-torki($e_1,e_2,\ldots$,ek) u kojoj je e1 pojava entiteta E_1, e_2 pojava entiteta $E_2,\ldots$,ek pojava entiteta Ek." (Varga,M.,1994.)

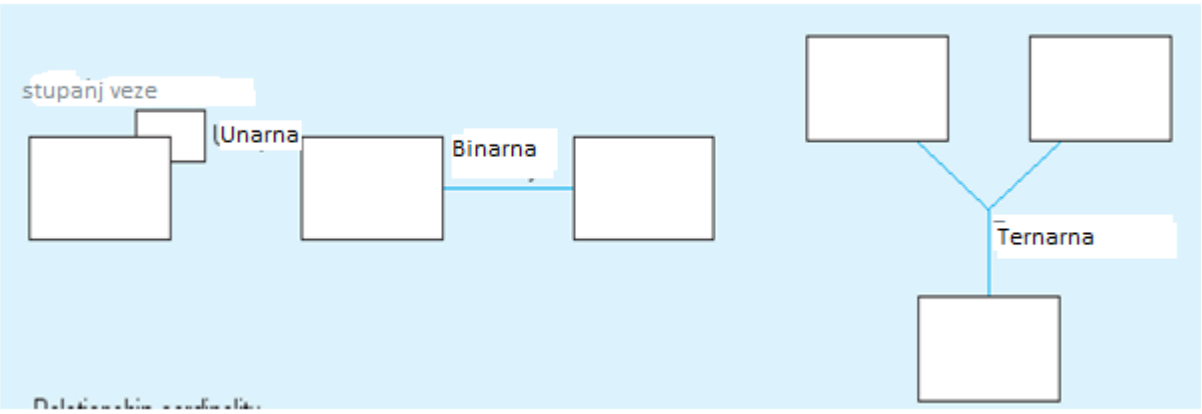
Veze mogu imati svoje atribute, a dva entiteta mogu imati više tipova veze. Tipovi veze dijele se na one koje su veze između više tipova entiteta i one koji su veze između istog tipa entiteta. Ovisno o broju entiteta mogu biti unarna, binarna, ternarna ili k-narna veza i takva podjela zove se podjela po stupnju veze. Veze između entiteta dijele se još po kardinalnosti i opcionalnosti. **Slika 2.3.** prikazuje klasifikaciju veze po stupnju veze, kardinalnosti i opcionalnosti.



Slika 2.3. Klasifikacija veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Stupanj veze ovisi o broju entiteta koji se u toj vezi nalazi. Unarna veza je veza u kojoj su povezana dva entiteta istog tipa odnosno entitet je povezan sa samim sobom. Binarna veza je veza u kojoj su povezana dva različita tipa entiteta. Ternarna veza je veza u kojoj sudjeluju i u kojoj su povezana tri različita tipa entiteta. **Slika 2.4.** prikazuje primjer unarne, binarne i ternarne veze



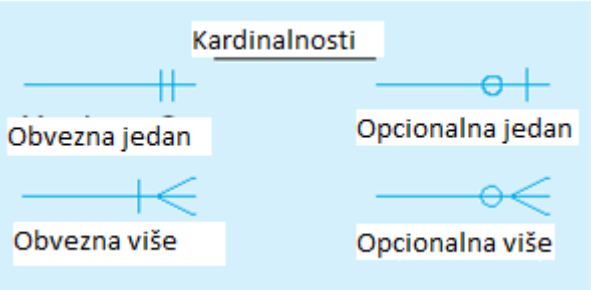
Slika 2.4. Stupanj veze

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Kardinalnost je svojstvo koje određuje koliki se broj instanci jednog entiteta povezuje sa instancom drugog entiteta. Krdinalnost može biti "jedan-jedan", "jedan-više", "više-jedan" i "više-više". Kardinalnost "jedan-jedan" prikazuje se oznakom 1:1 što znači da jedna instanca prvog entiteta može biti povezana sa samo jednom instancom drugog entiteta, a isto tako jedna instanca drugog entiteta može biti povezana samo sa jednom instancom prvog entiteta. Kardinalnost "jedan-više" prikazuje se oznakom 1:M i govori nam da jedna instanca prvog entiteta može biti povezana sa više instanci drugog entiteta, dok jedna instanca drugog entiteta može biti povezana samo sa jednom instancom prvog entiteta. Krdinalnost "više-jedan" prikazuje se oznakom M:1 i znači da jedna instanca prvog entiteta može biti povezana sa najviše jednom instancom drugog entiteta, dok jedna instanca drugog entiteta može biti povezana sa više instanci prvog entiteta. Kardinalnost "više-više" prikazuje se oznakom M:M i govori da jedna instanca prvog entiteta može biti povezana sa više instanci drugog entiteta, isto tako i jedna instanca drugog entiteta može biti povezana sa više instanci prvog entiteta.

Opcionalnost je svojstvo koje pokazuje koliki broj instanci jednog entiteta može ili mora biti povezano sa instancom drugog entiteta. Oznakama 0,1 ili M označujemo da li je kardinalnost opcionalna ili obvezna. Ako je nula, znači da je kardinalnost opcionalna, a ako je jedan ili više(M), to znači da je obvezna. Dakle, ako je nula, jedna instanca prvog entiteta može, ali i ne mora biti povezana sa instancom drugog entiteta, a ako je 1 ili M, instanca prvog entiteta mora biti povezan sa jednom instancom drugog entiteta, ili više. Dodavanjem opcionalnosti dobivamo kardinalnost koja se označava u intervalima u obliku donje i gornje granice. Donja granica zove se Minimum kardinalnosti, a gornja granica je Maximum kardinalnosti. Oznaka 0,1 pokazuje da jedna instanca prvog entiteta može biti povezana sa nijednom ili najviše jednom instancom drugog entiteta. Oznaka 1,1 pokazuje da jedna instanca prvog entiteta mora biti povezana sa točno jednom instancom drugog entiteta. Oznaka 0,M pokazuje da jedna instanca prvog entiteta može biti povezana sa nijednom, jednom ili više instanci drugog entiteta. Oznaka 1,M pokazuje da jedna instanca prvog entiteta mora biti povezana sa barem jednom instancom drugog entiteta, a može i s više njih.

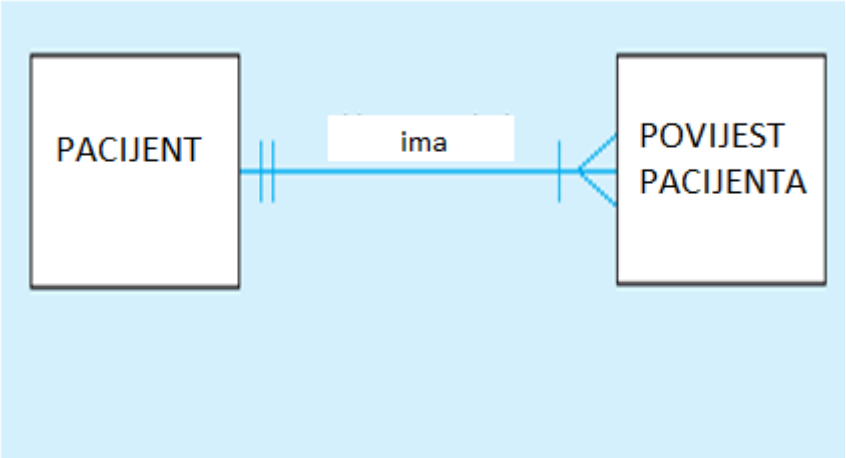
Pošto je veza između dva entiteta obostrana, određuje se kardinalnost u oba dva smjera. Tako, kardinalnost "jedan-jedan" može biti 1,1:1,1; 1,1:0,1; 0,1:1,1 i 0,1:0,1. Kardinalnost "jedan-više" može biti 1,1:0,M; 0,1:0;M; 1,1:1,M i 0,1:1,M a kardinalnost "više-više" može biti 0,M:0,M; 1,M:0,M i 1,M:1,M. **Slika 2.8.** pokazuje primjere obvezne kardinalnosti 1,1; 1,M i opcionalne kardinalnosti 0:1; 0,M.



Slika 2.8. Kardinalnosti

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

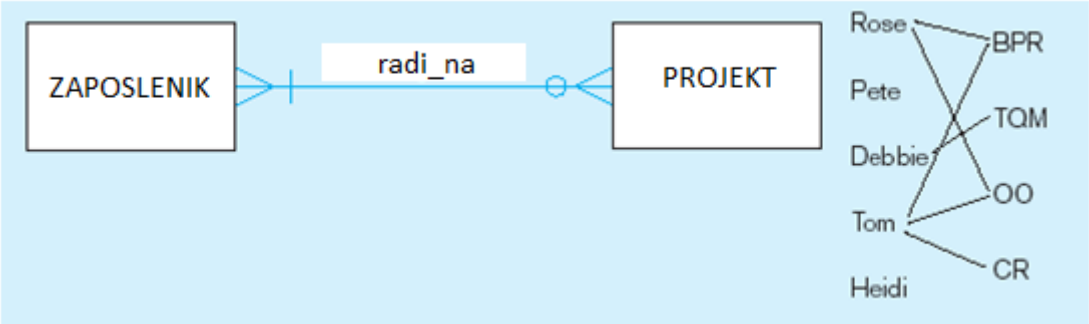
Slika 2.9. pokazuje primjer obvezne kardinalnosti 1,1:1,M u binarnoj vezi. Entiteti su PACIJENT i POVIJEST PACIJENTA. Povijest pacijenta je od jednog i samo jednog pacijenta, a pacijent mora imati povijest pacijenta, a može ih imati i više.



Slika 2.9. Obvezna kardinalnost

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

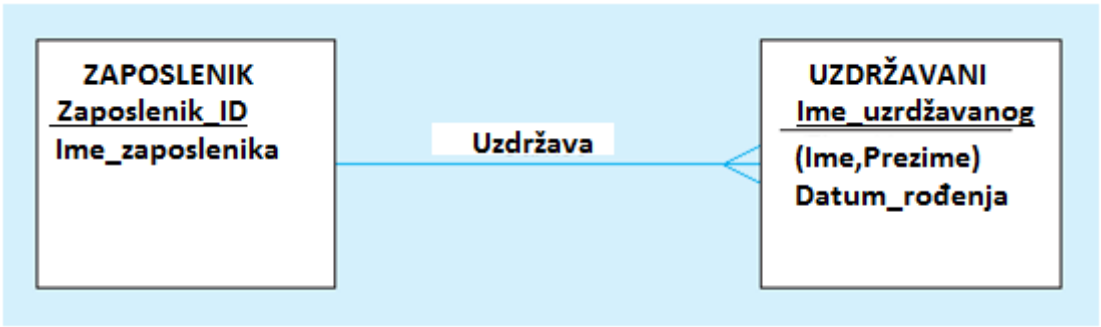
Slika 2.10. pokazuje primjer obvezne i opcionalne kardinalnosti 1,M:0,M u binarnoj vezi. Entiteti su ZAPOSLENIK i PROJEKT. Na projektu mora raditi barem jedan zaposlenik, a može i njih više. Zaposlenik može raditi na jednom ili više projekata, a može i na niti jednom.



Slika 2.10. Obvezna i opcionalna kardinalnost

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

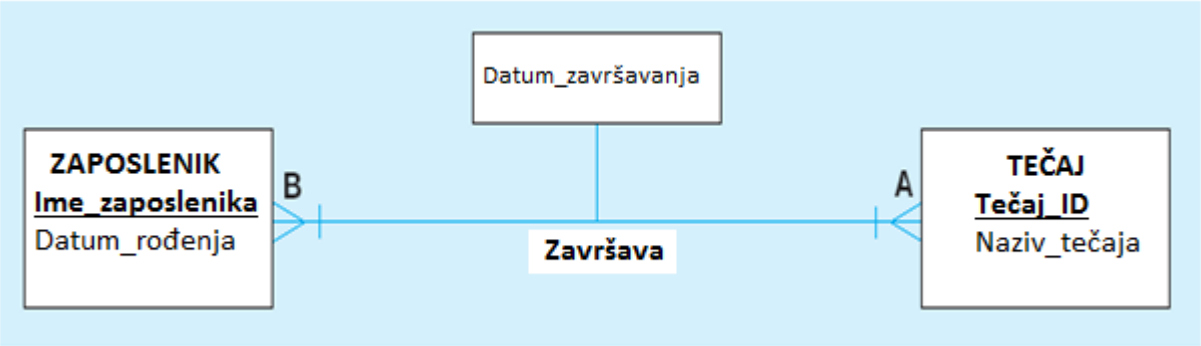
Veza koja nastaje između jakog i slabog entiteta naziva se Identifikacijska veza. Jaki entitet je entitet koji je neovisan od drugim entitetima i može postojati samostalno. Jaki entitet ima svoj identifikator- atribut koji ga jedinstveno opisuje(ključ). Slabi entitet je entitet koji nema svoj identifikator i on ne može postojati samostalno, nego je vezan za postojanje jakog entiteta. Dakle, identifikacijska veza je veza koja spaja jaki i slabi entitet. Jaki entitet može biti povezan sa više slabih entiteta, a slabi entitet mora imati samo jedan jaki entitet. **Slika 2.11.** pokazuje identifikacijsku vezu između jakog entiteta ZAPOSLENIK i slabog entiteta UZDRŽAVANI.



Slika 2.11. Identifikacijska veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Iako entiteti imaju atribute, a veza je odnos između entiteta, i veza, također, ima svoje atribute. Veza s atributom može postati asocijacijski entitet. Asocijacijski entitet može postati ako je veza "više-više", ako entitet ima značenje neovisno o drugim entitetima, kada ima identifikator i svoje atribute, kada samostalno djeluje u drugim vezama i ako se radi o ternarnoj vezi. **Slika 2.12.** prikazuje entitete ZAPOSLENIK i TEČAJ povezani vezom više-više sa svojim atributom.



Slika 2.12. Binarna veza s atributom

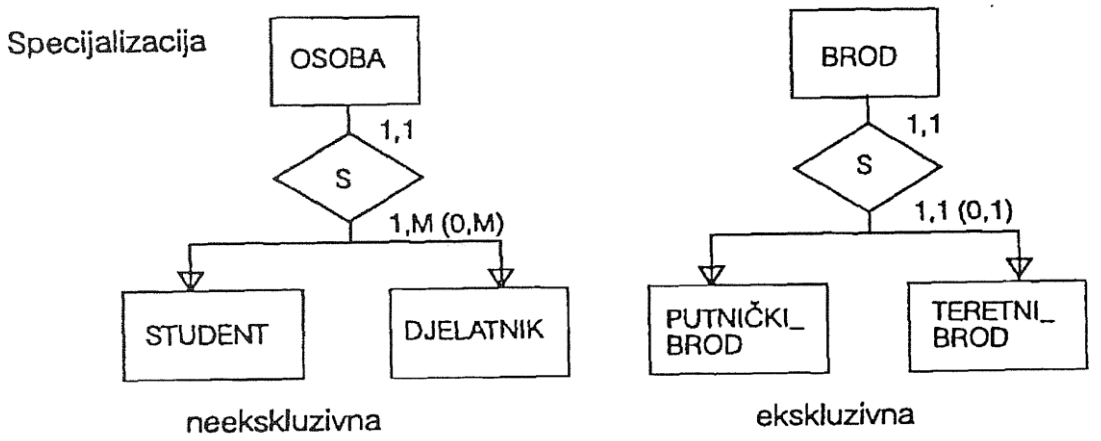
Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 2.13. prikazuje pretvaranje veze s atributom u asocijacijski entitet CERTIFIKAT. Nastale su dvije veze jedan-više.

Slika 2.13. Asocijacijski entitet

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Veza koja povezuje pod-tipove i nad-tipove entiteta, nastale postupkom specijalizacije ili generalizacije, zove se specijalizacija/generalizacija. Imamo ekskluzivnu i neekskluzivnu specijalizaciju. Ekskluzivna specijalizacija ili generalizacijska hijerarhija je veza koja nastaje kad se određeni nad-tip entiteta specijalizira samo u jedan pod-tip entiteta. Neekskluzivna specijalizacija ili podskupovna hijerarhija je veza koja nastaje kada se određeni nad-tip entiteta specijalizira istovremeno u više pod-tipova entiteta. Specijalizacija može biti, također, obvezna ili opcionalna, pa ekskluzivna specijalizacija može imati kardinalnosti 1,1 i 0,1 dok neekskluzivna može imati kardinalnost 1,M i 0,M.



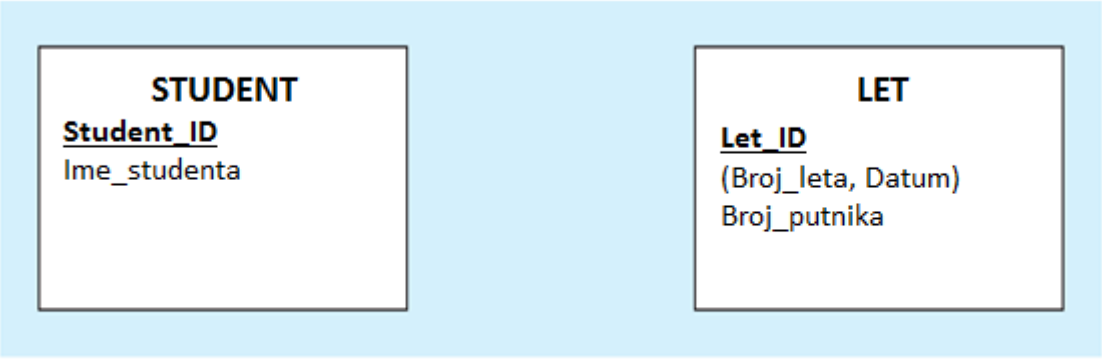
Slika 2.14. Ekskluzivna i neekskluzivna specijalizacija

Izvor: Varga,M.,1994., Baze Podataka

Slika 2.14. prikazuje primjer ekskluzivne i neekskluzivne specijalizacije. Entitet BROD može biti ili PUTNIČKI_BROD ili TERETNI_BROD pa je ta specijalizacija ekskluzivna dok OSOBA jest i STUDENT i DJELATNIK pa je tu riječ o neekskluzivnoj specijalizaciji.

2.2.3. ATRIBUTI

Atribut je obilježje entiteta. Služi kao opis entiteta i olakšava određivanje stanja tog entiteta. Postoje indikacijski i opisni atributi. I indikacijski i opisni atributi mogu imati jedan ili više atributa. Takav atribut, koji je sastavljen od više atributa, naziva se složeni atribut, a ako je sastavljen od jednog atributa, onda je taj atribut jednostavan. Opisni atribut je atribut koji služi za detaljan opis entiteta. Indikacijski atribut je atribut čije vrijednosti na jedinstven način određuju tip entiteta. Dakle, svaki entitet ima svoj atribut, sa jedinstvenim vrijednostima, koji ga identificira i ne postoje dva entiteta sa istim vrijednostima indikacijskog atributa. Indikacijski atribut nazivamo još i indikator ili ključ. (Primjer. OIB, ID_djelatnika, Broj_Bankovnog_Računa, ID_Odjela...). Ključ može biti sastavljen od više atributa. Svaki od tih atributa bi mogli biti ključ pa se oni nazivaju kandidati za ključ. U tom slučaju možemo izabrati jedan od kandidata za ključ i on tada postaje primarni ključ. **Slika 2.15.** prikazuje primjer jednostavnog indikatora ID_Studenta i složenog indikatora ID_Leta.



Slika 2.15. Složeni i jednostavni indikator

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

U a) primjeru **slike 2.15.** entitet STUDENT ima ključ koji je sastavljen od jednog atributa STUDENT_ID, pa je to jednostavan ključ. U b) primjeru entitet LET ima ključ LET_ID koji je sastavljen od dva atributa BROJ_LETA i DATUM, pa je to složeni ključ.

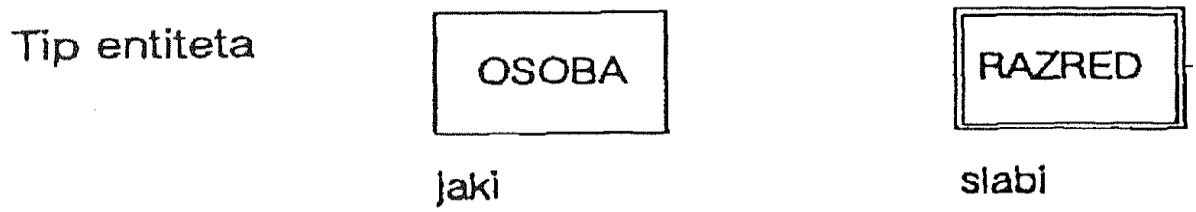
2.3 NOTACIJE E-R MODELA

E-R model omogućava izradu dobrog grafičkog prikaza. Samim time znatno je lakše razumjeti model podataka koji je prikazan. Jednostavni prikaz prikazuje tipove entiteta i veza, a u potpunom se prikazu zapisuju i atributi. Kad se crtaju složeniji modeli, potpuni prikaz sa atributima može biti nepregledan pa se češće koristi jednostavni prikaz, a atributi entiteta se zapisuju posebno.

Dijagram E-R modela najčešće se zapisuje Chenovim, Martinovim i Barkerovim prikazom.

2.3.1. CHENOV PRIKAZ

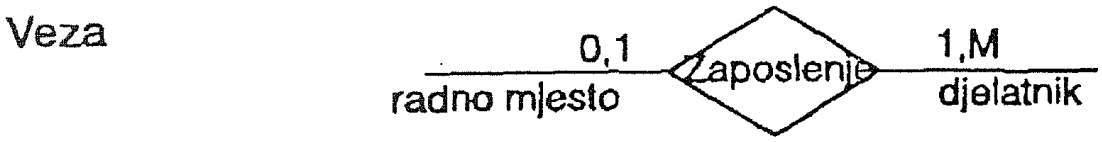
Chenov prikaz dobio je ime po Peteru Chenu koji je E-R dijagram predstavio 1967. godine. Prikazuje entitete, veze i atribute. Entiteti se u dijagramu prikazuju pravokutnikom. Ako je entitet jaki onda pravokutnik ima jednu crtu, a ako je slabi označuje se sa dvije crte. **Slika 2.16.**



Slika 2.16. Jaki i slabi entitet

Izvor: "Varga,M.,1994., Baze Podataka"

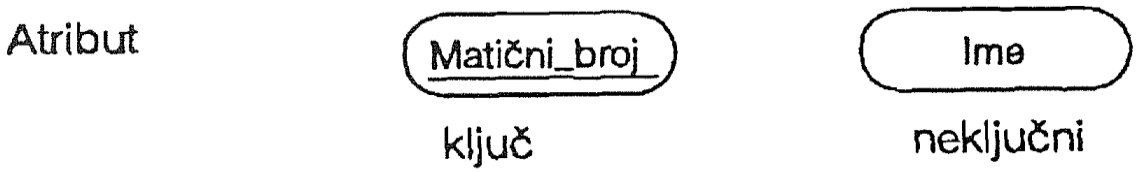
Veze se prikazuju simbolom romba koji je linijom spojen za entitete. **Slika 2.17.**



Slika 2.17. Veza

Izvor: "Varga,M.,1994., Baze Podataka"

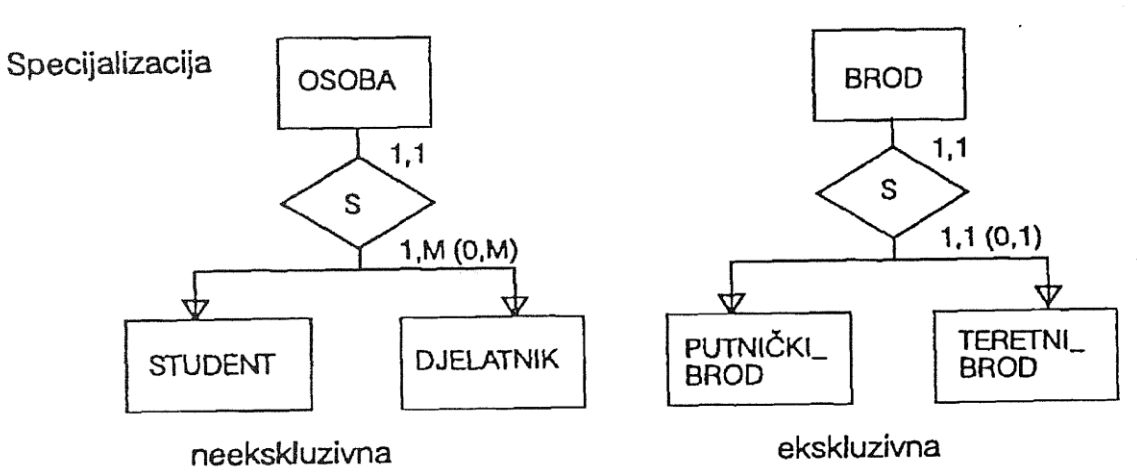
Simbol za atribut je elipsa ili krug. Oni su spojeni linijom za entitet kojega opisuju. Ako je atribut indikator(ključ) podcrtava se ravnom linijom. **Slika 2.18.**



Slika 2.18. Atribut

Izvor: "Varga,M.,1994., Baze Podataka"

Kardinalnosti "više" i "jedan" označavaju se simbolima M i 1 pokraj pravokutnika entiteta. Za označavanje kardinalnosti koristi se tzv. "Look across notation" gdje se kardinalnost prvog entiteta piše u smjeru bliže drugom entitetu. Specijalizacija se crta tako da je nad-tip entiteta povezan sa pod-tipovima entiteta vezom "s". Ako se radi o ekskluzivnoj specijalizaciji tada je kardinalnost 1:1, a ako se radi o neekskluzivnoj specijalizaciji tada je kardinalnost 1:M.

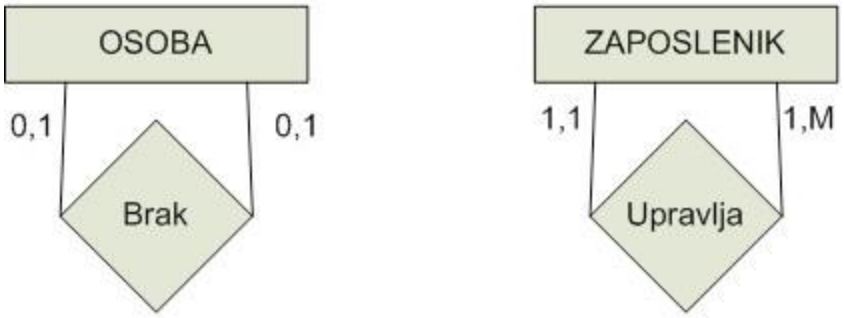


Slika 2.19. Specijalizacija

Izvor: "Varga,M.,1994., Baze Podataka"

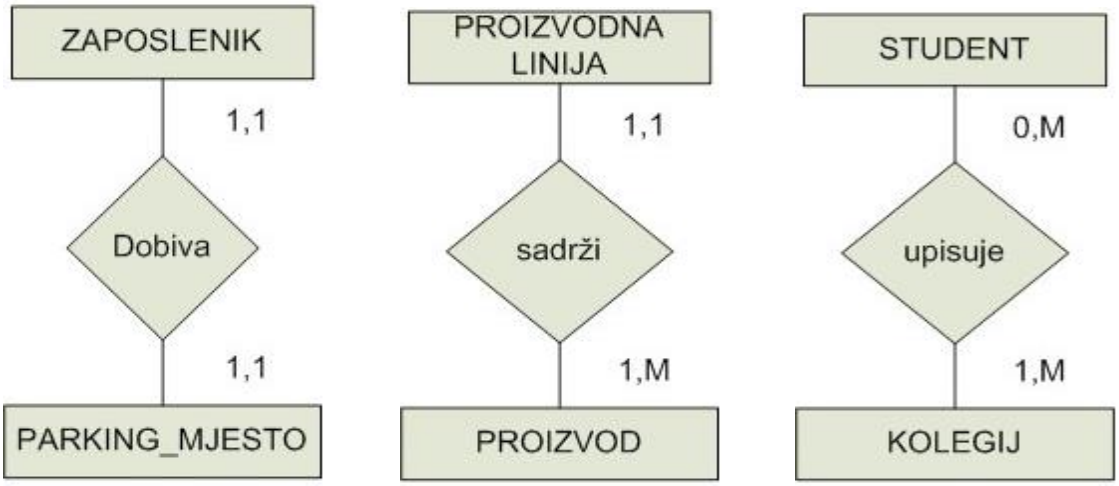
Slika 2.19. prikazuje primjer ekskluzivne i neekskluzivne specijalizacije. Entitet BROD može biti ili PUTNIČKI_BROD ili TERETNI_BROD pa je ta specijalizacija ekskluzivna dok OSOBA jest i STUDENT i DJELATNIK pa je tu riječ o neekskluzivnoj specijalizaciji.

Slika 2.20. prikazuje primjere unarnih veza u Chenovom prikazu E-R modela sa različitim kardinalnostima. Entitet OSOBA može biti u braku sa samim sobom, i OSOBA može biti u braku sa najviše jednom osobom. Entitet ZAPOSLENIK upravlja barem jednim zaposlenikom, a može upravljati sa više njih dok ZAPOSLENIK kojim se upravlja mora imati drugog zaposlenika koji upravlja njime i to može biti samo jedan zaposlenik.



Slika 2.20. Unarna veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/



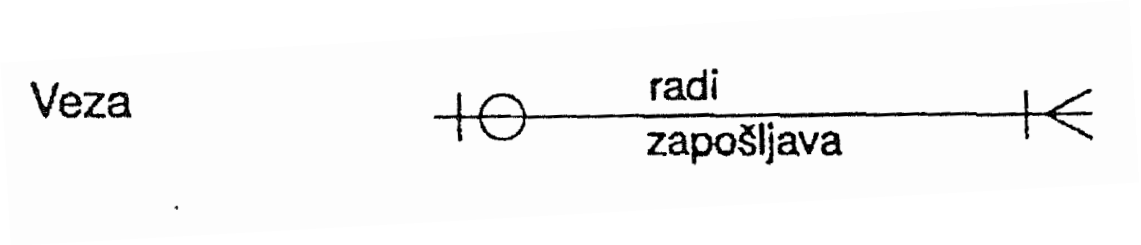
Slika 2.21. Binarna veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 2.21. prikazuje primjere binarnih veza u Chenovom prikazu E-R modela sa različitim kardinalnostima. Svaki ZAPOSLENIK dobiva svoje PARKING_MJESTO, ne može dobiti više od jednog. PROIZVODNA LINIJA sadrži najmanje jedan PROIZVOD, a može i njih više, dok PROIZVOD mora pripadati samo jednoj PROIZVODNOJ LINIJI. STUDENT upisuje barem jedan KOLEGIJ, a može upisati i njih više, dok KOLEGIJ može biti upisan od strane STUDENTA, a može i od njih više.

2.3.2. MARTINOV PRIKAZ

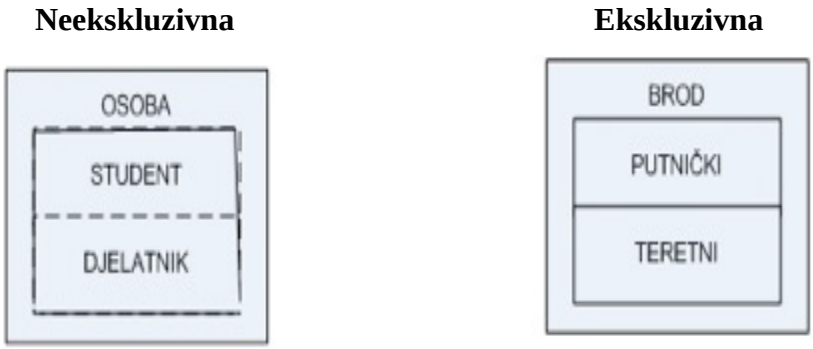
U Martinovu se prikazu E-R modela entiteti prikazuju simbolom pravokutnika, jednako kao u Chenovom prikazu. Slabi entiteti ne prikazuju se posebno. Atributi se, također, ne razlikuju od atributa u Chenovom prikazu. Dakle, prikazani su pomoću elipse ili kruga. Atribut koji je ključ se podcrtava. Veze se u Martinovom prikazu crtaju ravnom linijom između dva entiteta. Veza se ne imenuje nego se navode uloge entiteta koje veza spaja. U ovom prikazu koriste se samo unarne i binarne veze. U vodoravno nacrtanoj vezi, tekst iznad linije prikazuje ulogu lijevog entiteta, a tekst ispod linije prikazuje ulogu desnog entiteta. Kod kardinalnosti, donja granica, odnosno 0, se označava sa znakom O(kružić). Znakom |(crtica) označuje se donja ili gornja granica 1, a znakom > prikazuje se gornja granica kardinalnosti "više". **Slika 2.22.** prikazuje primjer simbola veze.



Slika 2.22. Veza

Izvor: "Varga,M.,1994., Baze Podataka"

Specijalizacija se crta tako da se unutar nad-tipa entiteta, koji je označen simbolom pravokutnika, crtaju pravokutnici entiteta koji su pod-tipovi tom entitetu. Ako je specijalizacija ekskluzivna, tada se pravokutnik pod-tipa crta punom crtom, a ako je specijalizacija neekskluzivna pravokutnik pod-tipa se crta isprekidanom linijom. **Slika 2.23.** prikazuje primjer ekskluzivne i neekskluzivne specijalizacije u Martinovom prikazu.

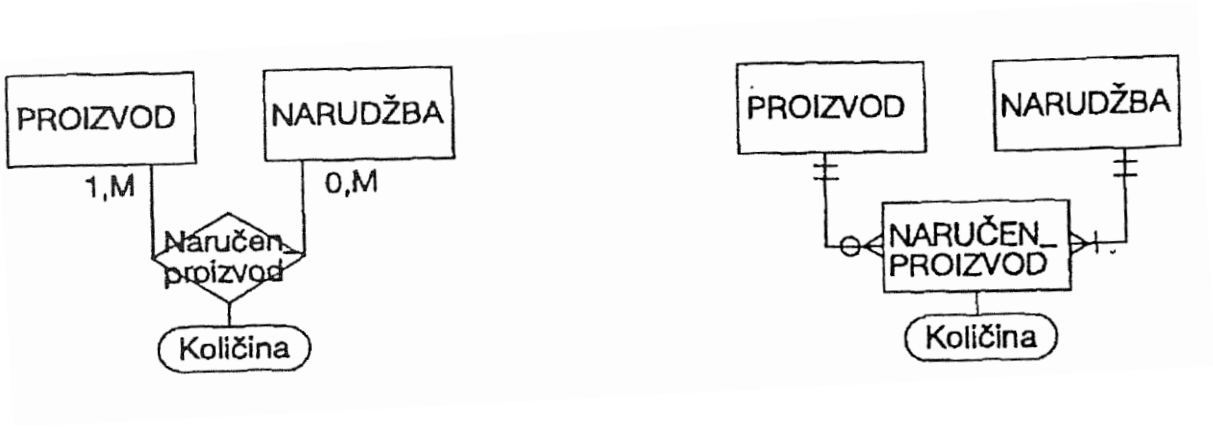


Slika 2.23. Neekskluzivna i ekskluzivna specijalizacija

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 2.23. prikazuje primjer ekskluzivne i neekskluzivne specijalizacije. Entitet BROD može biti ili PUTNIČKI_BROD ili TERETNI_BROD pa je ta specijalizacija ekskluzivna dok OSOBA jest i STUDENT i DJELATNIK pa je tu riječ o neekskluzivnoj specijalizaciji.

Veze u Martinovom prikazu ne mogu imati atribute pa se veze s atributima pretvaraju u asocijacijski entitet i tako povezuju sa ostalim entitetima.

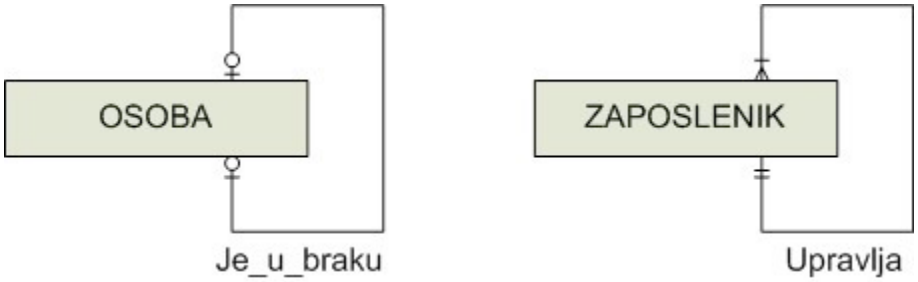


Chenov prikaz **Martinov prikaz**

Slika 2.24. Veze s atributima

Izvor: "Varga,M.,1994., Baze Podataka"

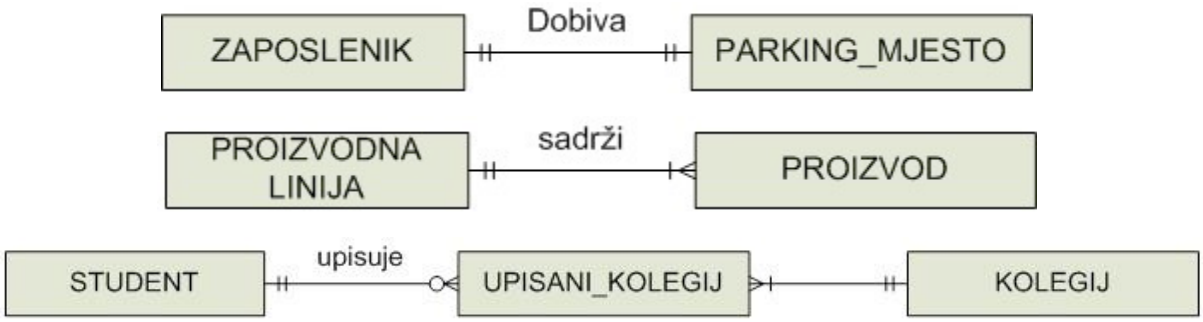
Slika 2.24. prikazuje primjer razlike u vezi s atributima u Chenovom i Martinovom prikazu gdje veza s atributima u Martinovom prikazu postaje asocijacijski entitet NARUČEN_PROIZVOD.



Slika 2.25. Unarna veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 2.25. prikazuje primjere unarnih veza u Martinovom prikazu E-R modela sa različitim kardinalnostima. Entitet OSOBA može biti u braku sa samim sobom, i OSOBA može biti u braku sa najviše jednom osobom. Entitet ZAPOSLENIK upravlja barem jednim zaposlenikom, a može upravljati sa više njih dok ZAPOSLENIK kojim se upravlja mora imati drugog zaposlenika koji upravlja njime i to može biti samo jedan zaposlenik.



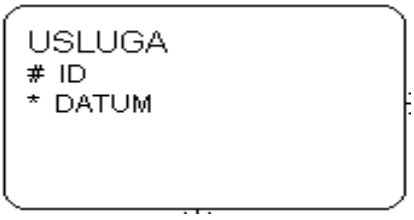
Slika 2.26. Binarna veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 2.26. prikazuje primjere binarnih veza u Martinovom prikazu E-R modela sa različitim kardinalnostima. Svaki ZAPOSLENIK dobiva svoje PARKING_MJESTO, ne može dobiti više od jednog. PROIZVODNA LINIJA sadrži najmanje jedan PROIZVOD, a može i njih više, dok PROIZVOD mora pripadati samo jednoj PROIZVODNOJ LINIJI. STUDENT upisuje barem jedan KOLEGIJ, a može upisati i njih više, dok KOLEGIJ može biti upisan od strane STUDENTA, a može i od njih više. Zbog relacije M:M stvara se asocijacijski entitet UPISANI KOLEGIJ između druga dva entiteta.

2.3.3. BARKEROV PRIKAZ

Barkerov prikaz nazivamo još i Oracle Case method. Entiteti se u prikazu označavaju pravokutnikom. Jaki se entiteti ne izdvajaju posebno od slabih entiteta. Atributi se upisuju u pravokutnik entiteta tako da budu navedeni ispod naziva entiteta. Obavezni atribut označuje se znakom *(zvjezdica), dok se neobavezni atribut prikazuje znakom °(prazni kružić). Indikator entiteta prikazuje se znakom #(ljestve). **Slika 2.28.** prikazuje primjer entiteta USLUGA sa ključem ID i obveznim atributom DATUM



Slika 2.28.

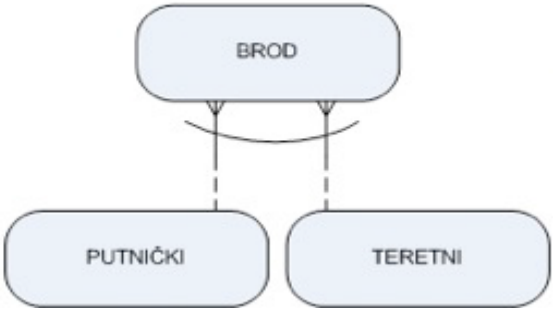
Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Veza se u Barkerovu prikazu prikazuje kao linija između dva entiteta. Kardinalnost veze se označava tako da ako je kardinalnost "više" crta se znak > pokraj entiteta. Opcionalnost se označava tako da ako je veza opcionalna onda se crta isprekidana linija kod suprotnog entiteta, a ako je opcionalnost obvezna, tada se crta puna linija. Dakle, za označavanje kardinalnosti koristi se tzv. "Look her notation", a za prikazivanje opcionalnosti tzv. "Look across notation". **Slika 2.29.** prikazuje primjer veze u Barkerovom prikazu.

Slika 2.29. Veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

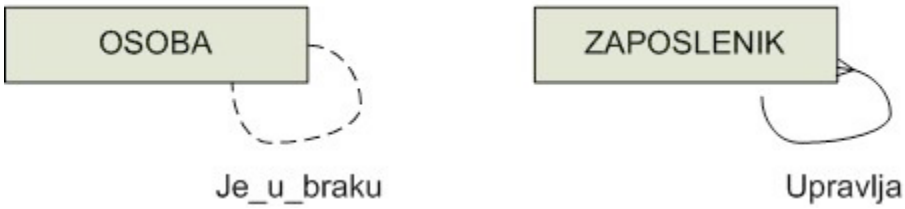
U ovom prikazu koriste se samo unarne i binarne veze. Za rješavanje veze "više-više", u Barkerovim prikazu, stvara se asocijacijski entitet koji povezuje druga dva entiteta. Specijalizacija veze se crta tako da se nad-tip entiteta povezuje linijom sa pod-tipom entiteta, a kod nad-tipa entiteta crta se luk. Luk može obuhvatiti samo veze koje su ili sve ekskluzivne ili sve neekskluzivne. Može obuhvatiti najmanje dvije veze, a najviše četiri veze. Lukovi se gotovo uvijek crtaju na M strani veze i oni ne mogu obuhvatiti veze koje su povezane sa drugim entitetima.



Slika 2.30. Ekskluzivna specijalizacija

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

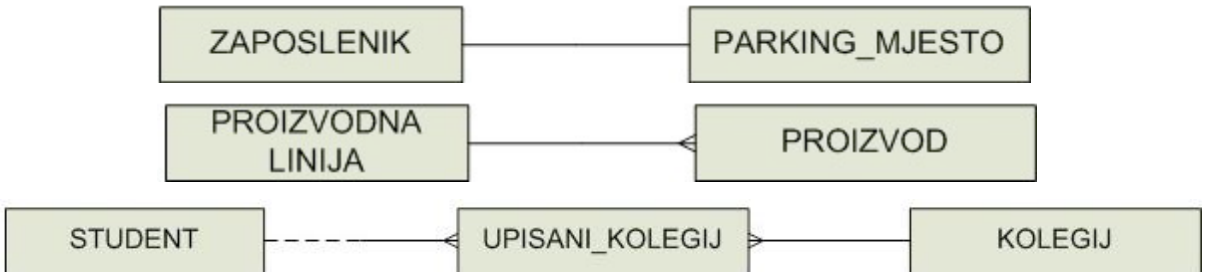
Slika 2.30. prikazuje primjer ekskluzivne specijalizacije u Barkerovom prikazu. Entitet BROD može biti ili PUTNIČKI_BROD ili TERETNI_BROD pa je ta specijalizacija ekskluzivna .



Slika 2.31. Unarna veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 2.31. prikazuje primjere unarnih veza u Barkerovom prikazu E-R modela sa različitim kardinalnostima. Entitet OSOBA može biti u braku sa samim sobom, i OSOBA može biti u braku sa najviše jednom osobom. Entitet ZAPOSLENIK upravlja barem jednim zaposlenikom, a može upravljati sa više njih dok ZAPOSLENIK kojim se upravlja mora imati drugog zaposlenika koji upravlja njime i to može biti samo jedan zaposlenik.



Slika 2.32. Binarna veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

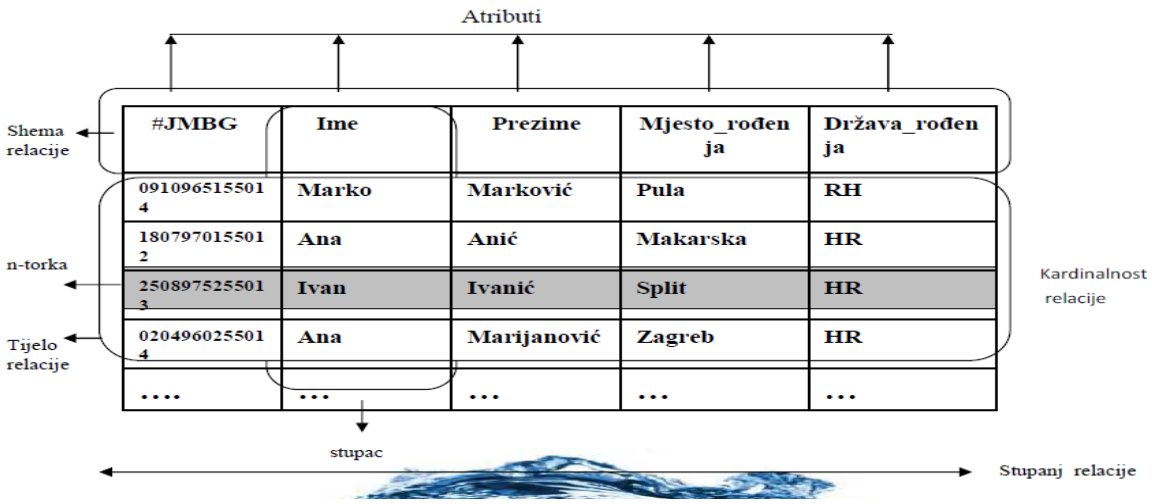
Slika 2.32. prikazuje primjere binarnih veza u Barkerovom prikazu E-R modela sa različitim kardinalnostima. Svaki ZAPOSLENIK dobiva svoje PARKING_MJESTO, ne može dobiti više od jednog. PROIZVODNA LINIJA sadrži najmanje jedan PROIZVOD, a može i njih više, dok PROIZVOD mora pripadati samo jednoj PROIZVODNOJ LINIJI. STUDENT upisuje barem jedan KOLEGIJ, a može upisati i njih više, dok KOLEGIJ može biti upisan od strane STUDENTA, a može i od njih više. Zbog relacije M:M stvara se asocijacijski entitet UPISANI KOLEGIJ između druga dva entiteta.

3. LOGIČKO MODELIRANJE PODATAKA

Logičko modeliranje podataka je druga faza u modeliranju baze podataka. U ovoj se fazi konceptualni model pretvara u relacijski model podataka. Da bi se baza podataka mogla implementirati, potrebno je napraviti relacijsku shemu jer se većina današnjih sustava za upravljanje bazom podataka koristi relacijskim modelom.

U relacijskom se modelu entiteti i veze između entiteta pretvaraju u relacije tj. tablice. Relacijski je model, dakle, model u kojem se baza podataka sastoji od skupa relacija. Relacija je tablica u modelu sa stupcima i redovima koja predstavlja entitet ili vezu. Vrijednost atributa prikazuje se stupcem u relaciji. Vrijednosti atributa su podaci iste vrste, i takav skup vrijednosti jednog ili više atributa zove se domena. Broj atributa u relaciji nazva se stupanj relacije. Redak u relaciji naziva se n-torka i ona uglavnom predstavlja vrijednosti jednog entiteta. Relacija je skup n-torki i ne smiju postojati dvije jednake n-torke u jednoj relaciji. Svaka n-torka sadrži samo jednu vrijednost za svaki od svojih atributa. Broj n-torki u jednoj relaciji je kardinalnost relacije.

Relacija se sastoji od zaglavlja i tijela. Zaglavlje relacije je zaglavlje n-torke. Relacija ima iste atribute i isti stupanj kao zaglavlje. Zaglavlje se još naziva i relacijska shema. Tijelo relacije je skup n-torki koje imaju isto zaglavlje, a kardinalnost n-torki jest i kardinalnost relacije.



Slika 3.1. Relacija u logičkom modelu

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 3.1. prikazuje i opisuje relaciju u logičkom modelu podataka. Atribut je stupac u relaciji. N-torka je redak u relaciji. Shema relacije predstavlja zaglavlje relacije i n-torke. Tijelo relacije je skup n-torki sa istim zaglavljem. Kardinalnost je broj n-torki u relaciji. Stupanj relacije je broj atributa u relaciji.

Relacijska shema je skup svih atributa, odnosno svih imena atributa, pa je relacija tada relacija nad relacijskom shemom. Shema baze podataka je skup svih relacijskih shema svih relacija u bazi podataka, a baza podataka je skup relacija. Različite relacije mogu imati ista imena atributa, ali u bazi ne mogu postojati dvije relacije s istim imenom.

Super ključ je atribut ili skup atributa koji jedinstveno određuje n-torku u relaciji. Kandidat ključ je super ključ koji nema nijedan odgovarajući podskup koji bi bio super ključ u relaciji. Kandidat ključ mora zadovoljiti 2 uvjeta: 1. jedinstvenost - na relacijskoj shemi ne mogu postojati dvije n-torke s jednakim vrijednostima skupa atributa. 2. minimalnost - izbacivanjem bilo kojeg atributa, iz skupa atributa, narušava se svojstvo jedinstvenosti. Kada relacija ima više kandidata ključeva, izabire se jedan od njih, te on postaje primarni ključ. Primarni ključ je, dakle, kandidat ključ koji je odabran da jedinstveno odredi n-torku u relaciji.

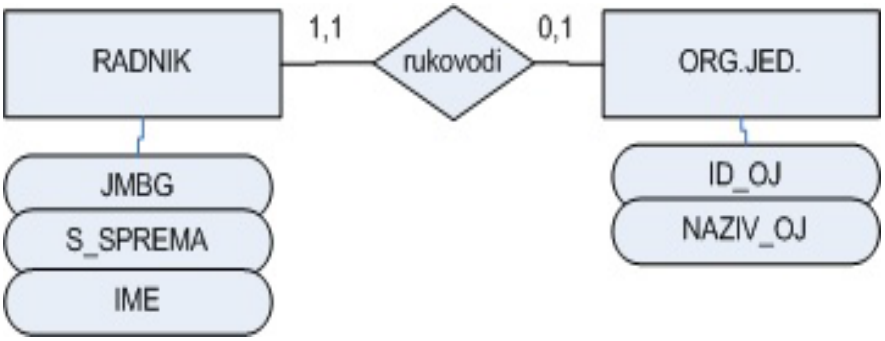
Vanjski ključ je atribut ili skup atributa jedne relacije koji odgovara kandidat ključu neke druge relacije. Pomoću vanjskih ključeva može se povezivati n-torke iz različitih tablica tj. povezuju se tablice u relacijskom modelu. Atribut ili skup atributa jedne relacije može biti vanjski ključ ako nije primarni ključ te relacije i ako ima zajedničku domenu sa primarnim ključem druge relacije.

3.1. PRETVARANJE KONCEPTUALNOG MODELA U RELACIJSKI MODEL PODATAKA

U relacijskom se modelu, za prikazivanje tipa entiteta koristi relacijska shema. Za prikaz skupa entiteta koristi se relacija, odnosno tablica. Entitet se prikazuje jednom n-torkom, odnosno jednim retkom u tablici. Atributi koji opisuju entitet postaju atributi relacije. Primarni ključ entiteta postaje primarni ključ relacije. Kod povezivanja jakog i slabog entiteta, jaki entitet ima jednostavni primarni ključ, dok slabi entitet ima složeni ključ sastavljen od primarnog ključa jakog i primarnog ključa slabog entiteta. Kad se radi o specijalizaciji, pod-tipovi entiteta poprimaju primarni ključ od nad-tipa entiteta pa svi imaju isti primarni ključ. Veze između entiteta nastaju povezivanjem vanjskog ključa jedne relacije

sa primarnim ključem druge relacije ili se za prikazivanje veza stvara nova relacija. Kod veza 1:1 i 1:M način povezivanja i prikaza veze između entiteta ovisi o njihovoj opcionalnosti. Ako je obvezna, upisuje se strani ključ na stranu koja obvezno sudjeluje u vezi (kod veze 1:1) i na stranu M (kod veze 1:M). Ako je opcionalna, tada se stvara nova relacija za vezu koja uzima ključ od jednog od entiteta (veza 1:1) ili uzima ključ od entiteta na strani M (veza 1:M).

Slijedeći primjer prikazuje pretvaranje konceptualnog modela u relacijski model u vezi 1:1, sa obveznom kardinalnosti.



Slika 3.2. Konceptualni model veze 1:1

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

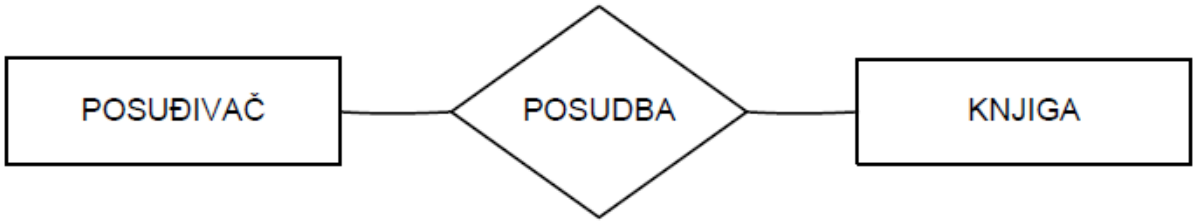
Slika 3.2. prikazuje primjer konceptualnog modela u vezi 1:1 s obveznom kardinalnosti. Entitet RADNIK može rukovoditi najviše jednom ORGANIZACIJSKOM_JEDINICOM, a može i niti jednom. Dok entitet ORGANIZACIJSKA_JEDINICA mora biti rukovođena najmanje jednim RADNIKOM, i ne može biti više od jednog.

ORG_JED(ID_OJ, NAZIV_OJ, JMBG)

RADNIK(JMBG, IME_R, S_SPREMA)

ORG_JED i RADNIK su tipovi entiteta. Entitet ORG_JED ima primarni ključ ID_OJ, a entitet RADNIK ima primarni ključ JMBG. Da bi se mogla napraviti veza između entiteta, entitetu ORG_JED dodan je vanjski ključ JMBG koji je primarni ključ entiteta RADNIK.

Slijedeći primjer prikazuje pretvaranje konceptualnog modela u relacijski model u vezi 1:M.



Slika 3.3. Konceptualni model BP u knjižnici

Izvor: "Manger, R., 2014., Baze podataka"

Slika 3.3. prikazuje posuđivanje knjige u knjižnici u konceptualnom modelu. Prikazana je veza 1:M sa opcionalnom kardinalnosti, gdje POSUĐIVAČ može posuditi jednu ili više knjiga, a KNJIGA može biti posuđena od samo jednog POSUĐIVAČA, ali i ne mora.

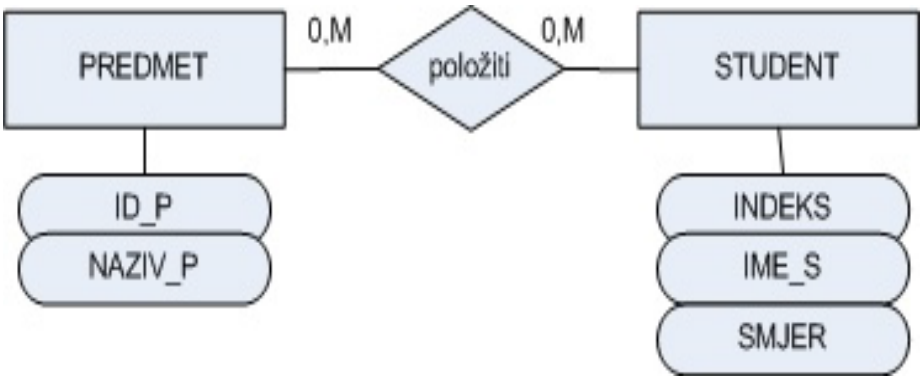
POSUĐIVAČ (BROJ ISKAZNICE, PREZIME, IME, ADRESA, …)

KNJIGA (INVENTARSKI BROJ, NASLOV, …)

PODSUDBA (INVENTARSKI BROJ, BROJ_ISKAZNICE)

POSUĐIVAČ i KNJIGA su tipovi entiteta. Entitetu POSUĐIVAČ dodan je primarni ključ BROJ _ISKAZNICE, a entitetu KNJIGA, primarni ključ INVENTARSKI_BROJ. Kako bi se napravila veza između entiteta, napravljena je nova relacija POSUDBA. Pošto jedna KNJIGA može biti posuđena samo jednom POSUĐIVAČU, primarni ključ u relaciji POSUDBA je INVENTARSKI_BROJ, isti kao u relaciji KNJIGA.

Kod veze tipa M:M, za povezivanje entiteta, uvijek se stvara nova relacija, a primarni ključ te relacije je složen od primarnih ključeva oba entiteta.



Slika 3.4. Konceptualni model veze M:M

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 3.4. prikazuje primjer konceptualnog modela u vezi M:M s opcionalnom kardinalnosti. Entitet STUDENT može položiti više PREDMETA, a ne mora niti jedan, dok entitet PREDMET može biti položen od više STUDENATA, a može i od niti jednoga.

PREDMET(ID_P, NAZIV_P)

STUDENT(INDEKS, IME_S, SMJER)

POLOŽITI(ID_P, INDEKS)

Tipovi entiteta jesu PREDMET i STUDENT. Entitet PREDMET ima primarni ključ ID_P, dok entitet STUDENT ima primarni ključ INDEKS. Kako bi se napravila veza između entiteta, stvara se nova relacija POLOŽITI. Nova relacija uzima ključeve od oba entiteta te je njen ključ složen, ID_P, INDEKS.

Kod unarne veze, ako je veza tipa 1:1 i 1:M, za povezivanje tipa entiteta sastavlja se nova relacija. Relacija preuzima primarni ključ od entiteta, a dodaje joj se i strani ključ koji se preimenuje. Pr. UPOSLENIK(JMBG, IME, PREZIME)

UPRAVLJA(JMBG, JMBG_MANAGERA)

Ako je unarna veza tipa M:M, tipovi entiteta se povezuju tako da se dodaje nova relacija čiji je ključ složen.

Slika 3.5. Konceptualni model - Unarna veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

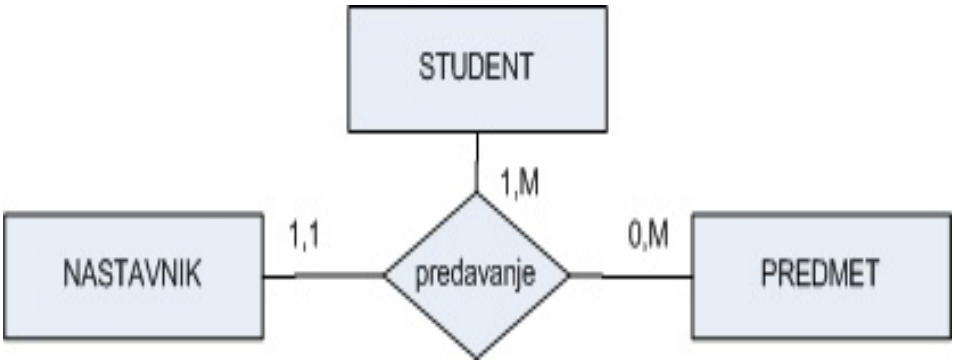
Slika 3.5. prikazuje unarnu vezu tipa M:M. Entitet PROIZVOD može imati u sastavu jedan ili više PROIZVODA, a ne mora niti jedan, dok PROIZVOD može biti u sastavu jednog ili više PROIZVODA, a ne mora od niti jednoga.

PROIZVOD(ID_PROIZ, NAZIV_PROIZ)

SASTAV(ID_PROIZ, ID_PROIZ_U_SASTAVU)

Tip entiteta PROIZVOD ima primarni ključ ID_PROIZ. Kako bi se stvorila veza, napravljena je nova relacija SASTAV koja ima složeni ključ ID_PROIZ i ID_PROIZ_U_SASTAVU. preuzet je primarni ključ iz entiteta PROIZVOD te je dodan još jedan izmijenjeni ključ.

Kod ternarne veze, za povezivanje tipova entiteta, dodaje se nova relacija koja ima složeni ključ sastavljen od svih ključeva entiteta koji su u vezi.



Slika 3.6. Konceptualni model - Ternarna veza

Izvor: "Bevanda, V., 2015.,Materijali s predavanja", http://e-ucenje.oet.unipu.hr/

Slika 3.6. prikazuje primjer ternarne veze. Entitet NASTAVNIK mora održati predavanje barem jednom STUDENTU, a može i više njima. NASTAVNIK može predavati više PREDMETA, a može i niti jedan. STUDENTU predavanje mora držati barem jedan NASTAVNIK i ne može više od jednoga, dok STUDENT može slušati više PREDMETA, a ne mora niti jedan. PREDMET mora predavati barem jedan NASTAVNIK i ne može više od jednoga, PREDMET mora slušati barem jedan STUDENT, a može i njih više.

STUDENT(ID_S, …)

NASTAVNIK(ID_N, …)

PREDMET(ID_PRED, …)

PREDAVANJE(ID_S, ID_N, ID_PRED,…)

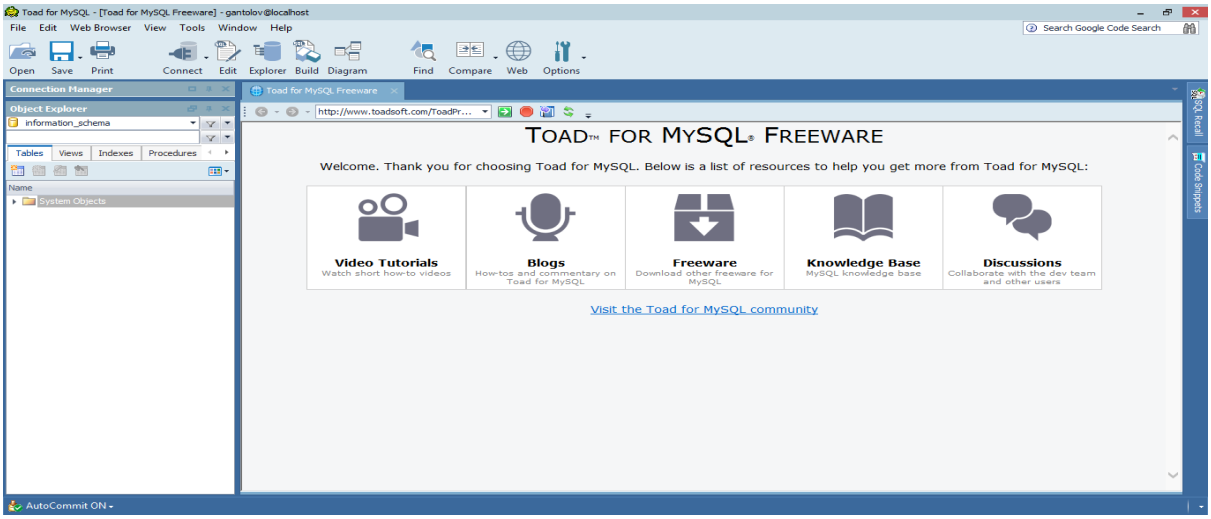
Entiteti su STUDENT, NASTAVNIK i PREDMET. Entitet STUDENT ima primarni ključ ID_S. Entitet NASTAVNIK ima primarni ključ ID_N, a entitet PREDMET ima primarni ključ ID_PRED. Kako bi se napravila veza između entiteta, stvorena je nova relacija PREDAVANJE koja ima složeni ključ sastavljen od ključeva svih entiteta, ID_S, ID_N, ID_PRED.

4. ALATI

Razvojem informacijskog sustava povećava se potreba i važnost izrade dobrog konceptualnog modela podataka. Sukladno tome, dolazi do stvaranja i razvoja alata za konceptualno modeliranje podataka sa grafičkim sučeljem koji olakšavaju izradu modela. Danas već postoji veliki broj alata za konceptualno modeliranje podataka koji se mogu spajati na razne sustave za upravljanje bazom podataka. Dalje će biti opisan rad triju alata koji se spajaju na MySQL sustav za upravljanje bazom podataka: TOAD, NAVICAT i DBFORGE STUDIO.

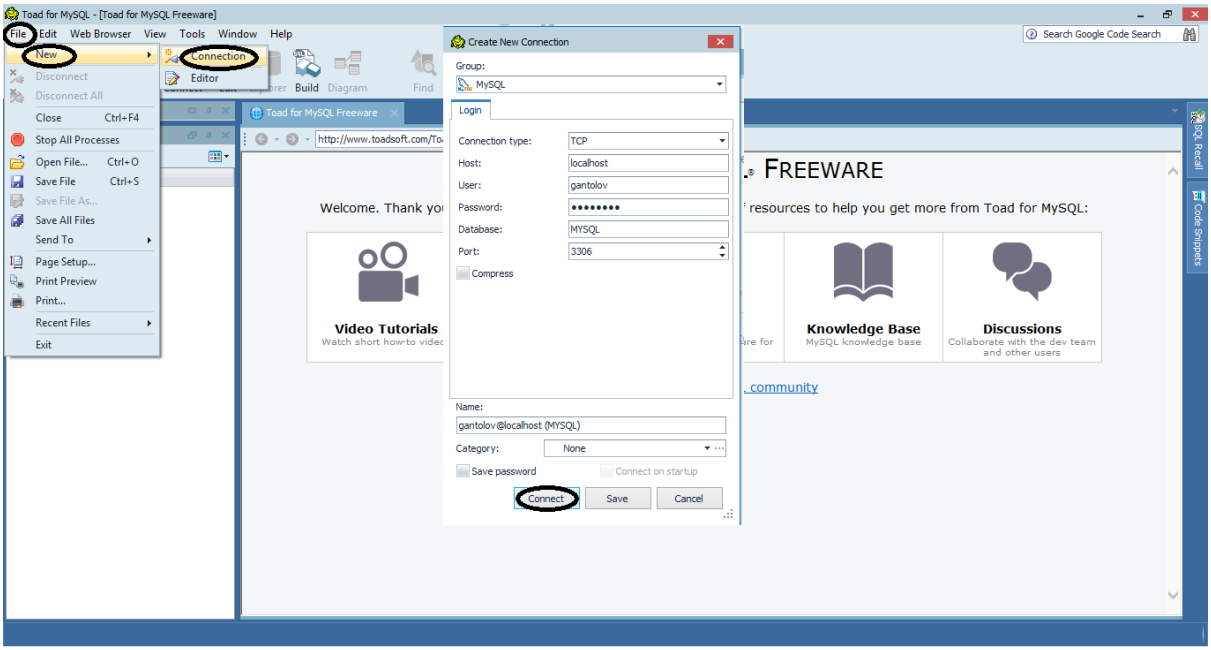
4.1. TOAD

Toad je alat za konceptualno modeliranje podataka. Može se spajati na Oracle, SQL server, MySQL, SQLite, Postgre SQL. Radi samo na Microsoft Windows platformi. Ovdje će biti prikazan Toad za MySQL sustav za upravljanje bazom podataka. Najprije treba instalirati MySQL server te onda Toad for MySQL. Kad instalacija završi, otvara se početni zaslon i može se krenuti sa radom. **Slika 4.1.** prikazuje početni zaslon Toad alata



Slika 4.1. Početni zaslon

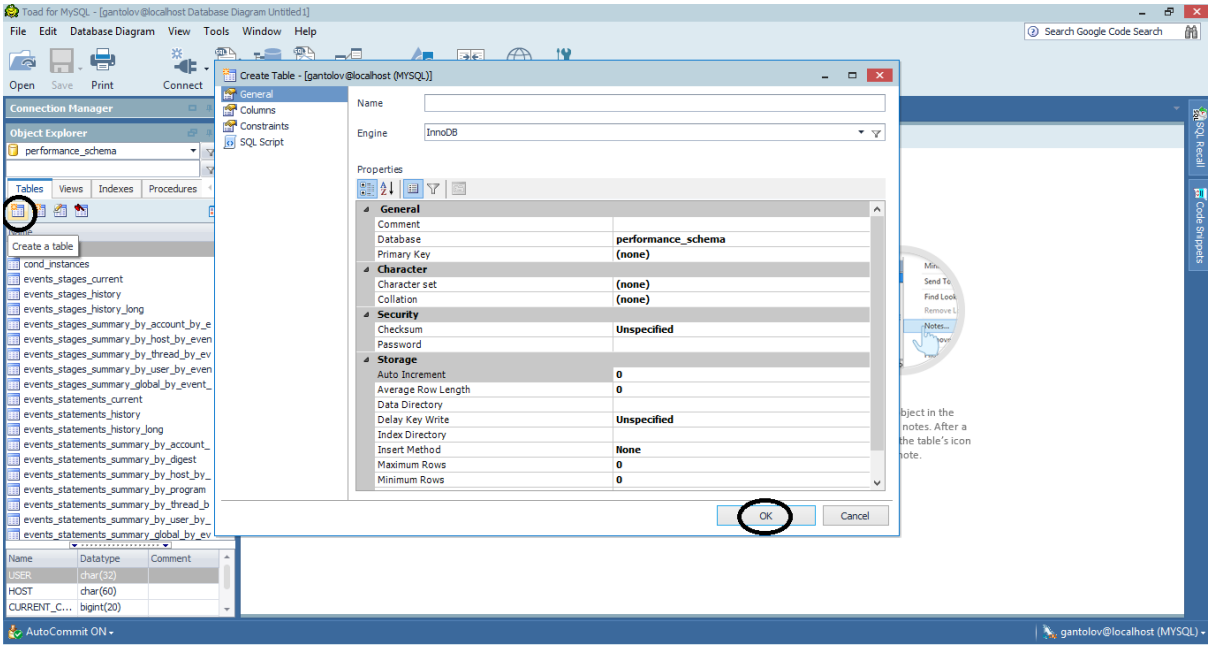
Da bi mogli raditi, prvo se alat treba spojiti na MySQL server. U gornjem, lijevom kutu alatne trake stisne se FILE→NEW→CONNECTION. Otvara se prozor u koji treba upisati sve informacije potrebne za povezivanje na bazu, te se klikne na CONNECT. Sada se konekcija nalazi sa lijeve strane početnog zaslona. Svaki put kad se ponovno otvori alat, i želi se povezati na istu konekciju, stisne se desni klik na ime konekcije i CONNECT. Kad je spojen sa serverom, ispod konekcije pojavljuje se popis baza podataka



Slika 4.2. Spajanje na MySQL server

Slika 4.2. prikazuje postupak spajanja na server. FILE→NEW→CONNECTION, i nakon što su svi podaci upisani, CONNECT

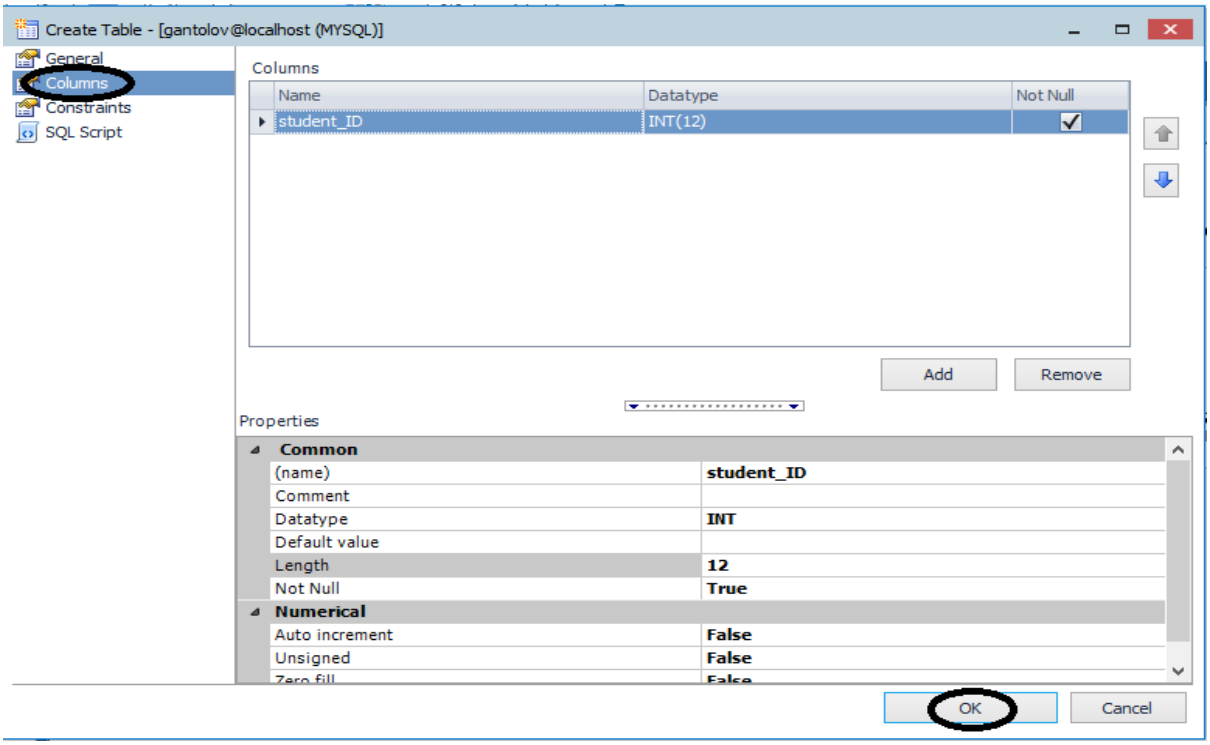
Kako bi kreirali novi model klikne se na alatnoj traci TOOLS→DATABASE DIAGRAM. Nakon što je napravljen novi model, kreće se sa crtanjem dijagrama, tj. kreiranjem entiteta, atributa i veza. Kako bi se kreirao entitet, prvo treba napraviti tablicu i zatim tu tablicu povući u model, ili se može povući već postojeću tablicu. Za kreiranje tablice na alatnoj traci klikne se VIEW→OBJECT EXPLORER, odabere bazu, klik na TABLES, te na znak tabele na skroz lijevoj strani. Otvara se novi prozor sa informacijama o tablici koju se želi napraviti. Moguće je odmah entitetu pridodati atribute, te napraviti željene restrikcije. Kada je oblikovanje tablice gotovo, klik na OK i tablica je kreirana i smještena lijevo na popis tablica u bazi podataka. Uzme se napravljenu tablicu i povuče u model. **Slika 4.3.** prikazuje postupak kreiranja nove tablice



Slika 4.3. Kreiranje tablice

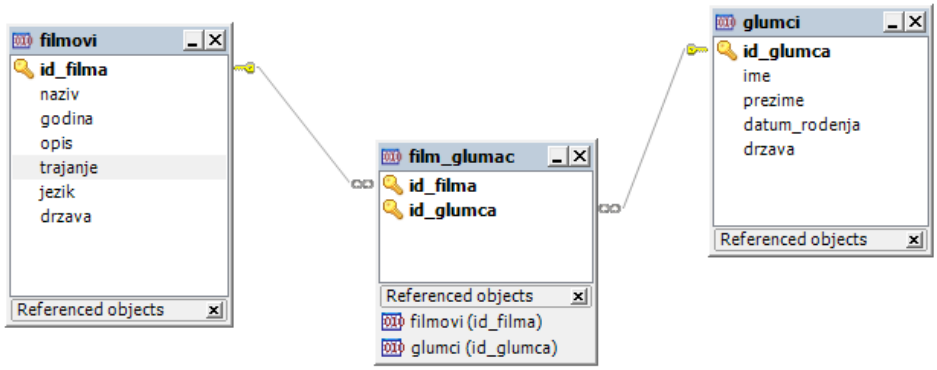
Ako se želi dodati atribute entitetu, prilikom stvaranja tablice, na prozoru tablice klikne se COLUMNS. Nakon toga upišu se sve informacije o atributu koji se želi kreirati. Primarni ključ entiteta upisuje se u GENERAL. Kad se završi, klik na OK. **Slika 4.4.** prikazuje kako se upisuje atribut u Toad alatu.

Ukoliko se sve tablice koje su potrebne za izradu modela nalaze već u bazi podataka i nije ih potrebno kreirati, klikne se na alatnu traku TOOLS→ DATABASE DIAGRAM. Nakon otvaranja modela, na traci modela klik na opciju ALL TABLES AND VIEWS FROM DATABASE ONLY, i sve se tablice iz baze podataka prebace u model.



Slika 4.4. Kreiranje atributa

Povlačenjem atributa iz jedne tablice na atribut druge tablice, stvara se veza između tablica. Dakle, povezuju se primarni i vanjski ključevi. Oznaka ključa na vezi pokazuje primarni ključ, a oznaka beskonačno pokazuje vanjski ključ. Klikne li se desni klik na tablicu, mogu se vidjeti i mijenjati postavke te tablice.

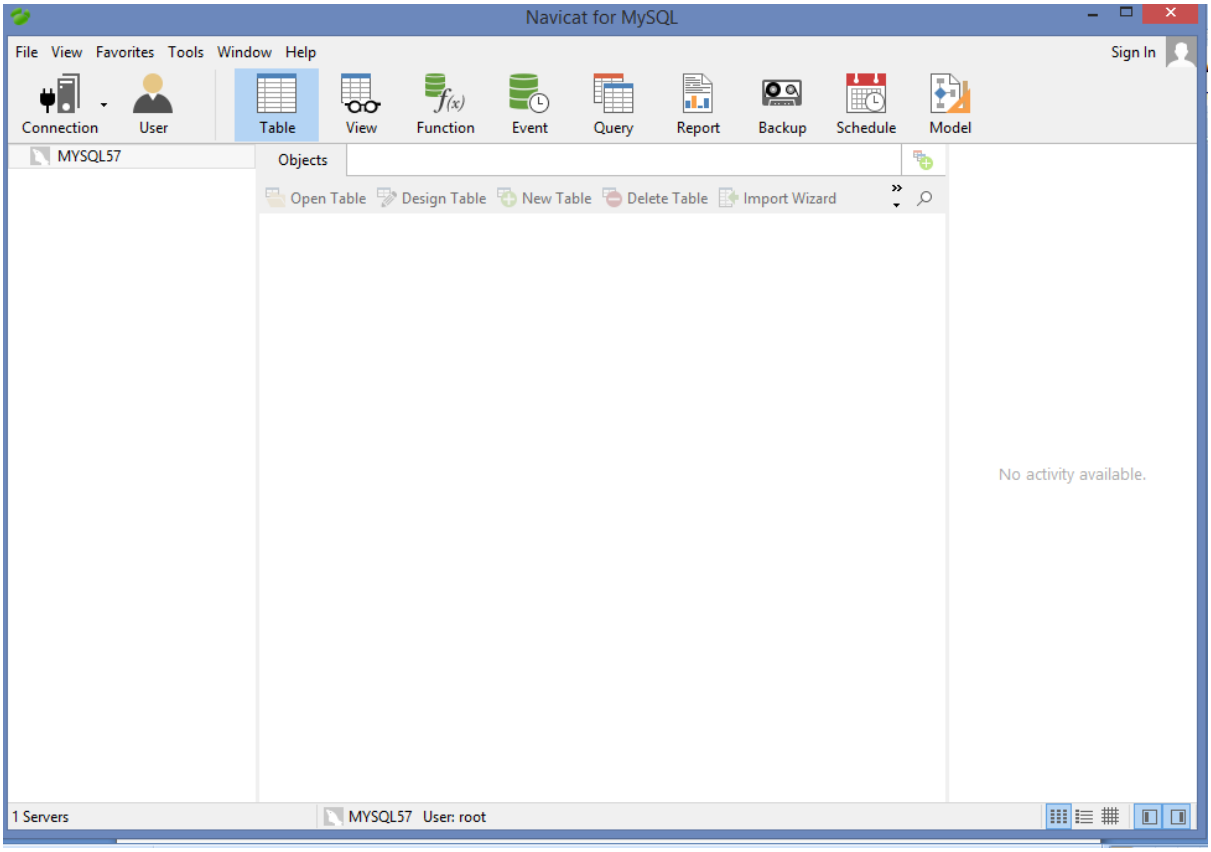


Slika 4.5. E-R model

Slika 4.5. prikazuje primjer E-R modela u alatu Toad. Entiteti FILMOVI, FILM_GLUMAC i GLUMCI prikazani su tablicama i oni se povezuju definiranjem primarnih i vanjskih ključeva.

4.2. NAVICAT

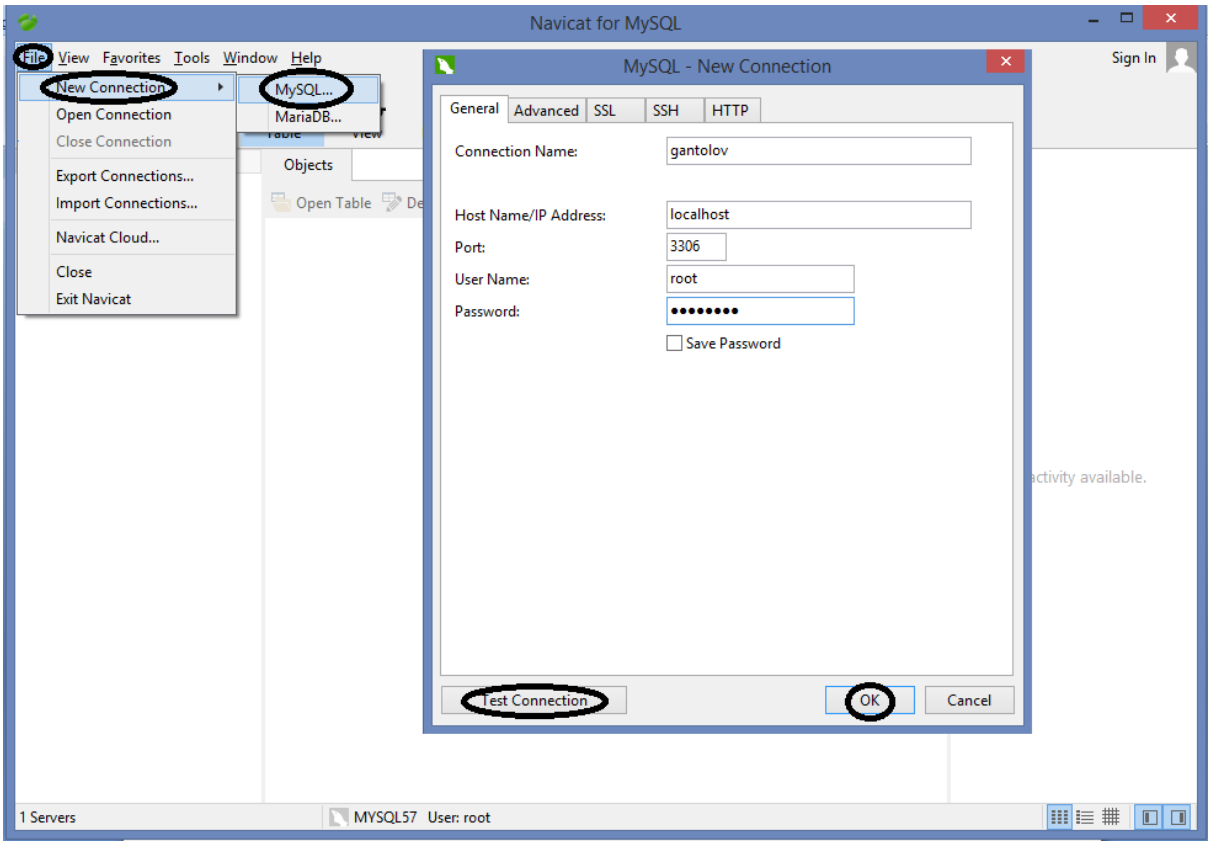
Navicat je alat za konceptualno modeliranje podataka sa grafičkim sučeljem koji se spaja na Oracle, SQL Server, MySQL, SQLite, PostgreSQL, MariaDB sustave za upravljanje bazom podataka Radi na Microsoft Windows, MAC OS i Linux platformi. Ovdje će biti prikazan Navicat za MySQL sustav za upravljanje bazom podataka. Najprije treba instalirati MySQL server te onda Navicat for MySQL. Kad instalacija završi, otvara se početni zaslon i može se krenuti sa radom. **Slika 4.6.** početni zaslon u alatu Navicat.



Slika 4.6. Početni zaslon

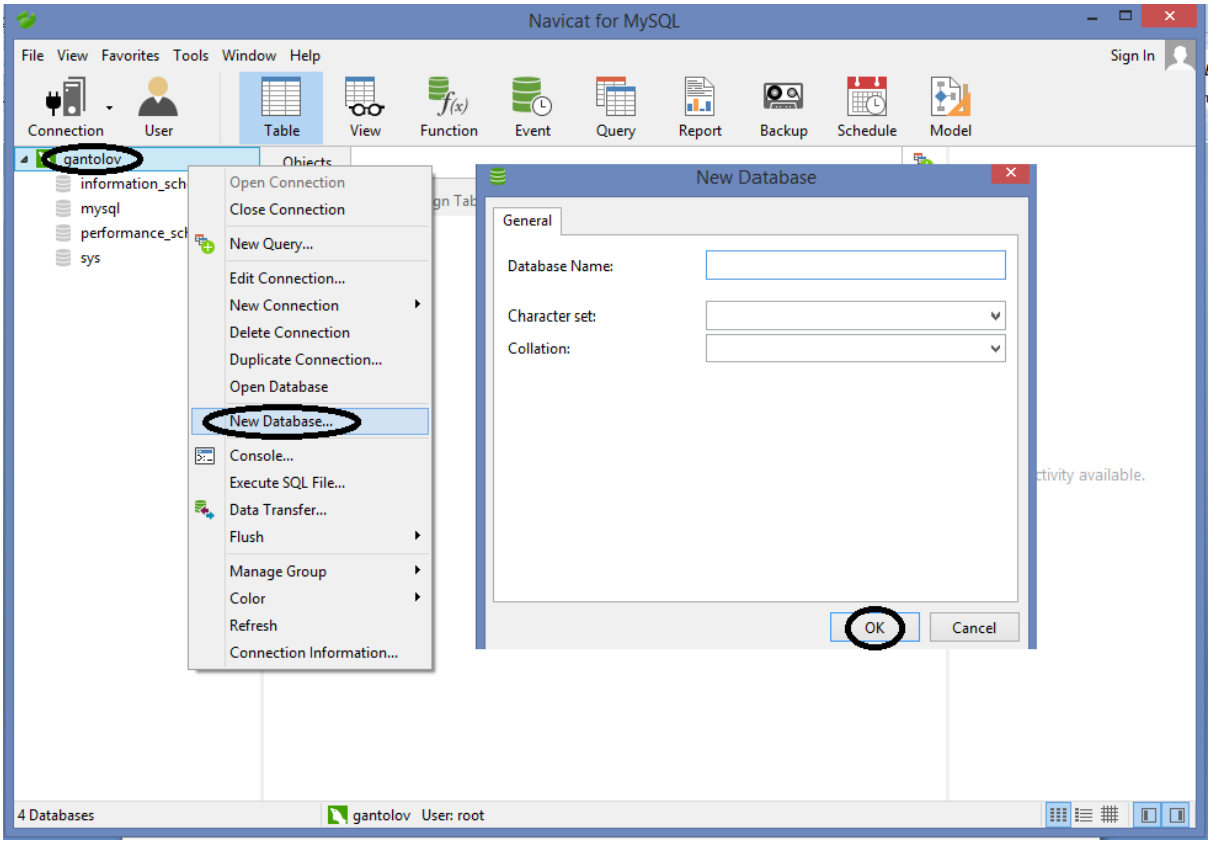
Za daljnji rad, potrebno je spojiti se na MySQL server. U gornjem lijevom kutu alatne trake klikne se na FILE→NEW CONNECTION→MySQL. Otvori se prozor sa podacima koje treba ispuniti kako bi konekcija na server bila uspješna. Traži se ime konekcije, ime računala koji je spojen na MySQL server, lozinka, user name... Nakon što su svi podaci upisani, klikne se na TEST CONNECTION kako bi provjerili da li je sve dobro upisano. Ako je sve uredu, klik na OK. Ime konekcije se sada nalazi na lijevoj strani početnog zaslona. Svaki put kad se Navicat pokrene, desni klik na ime konekcije, stisne se OPEN

CONNECTION, upiše password, te klik na OK. Sada je spojen na server. **Slika 4.6.** prikazuje postupak stvaranja nove konekcije sa MySQL serverom.



Slika 4.6. Spajanje na MySQL server

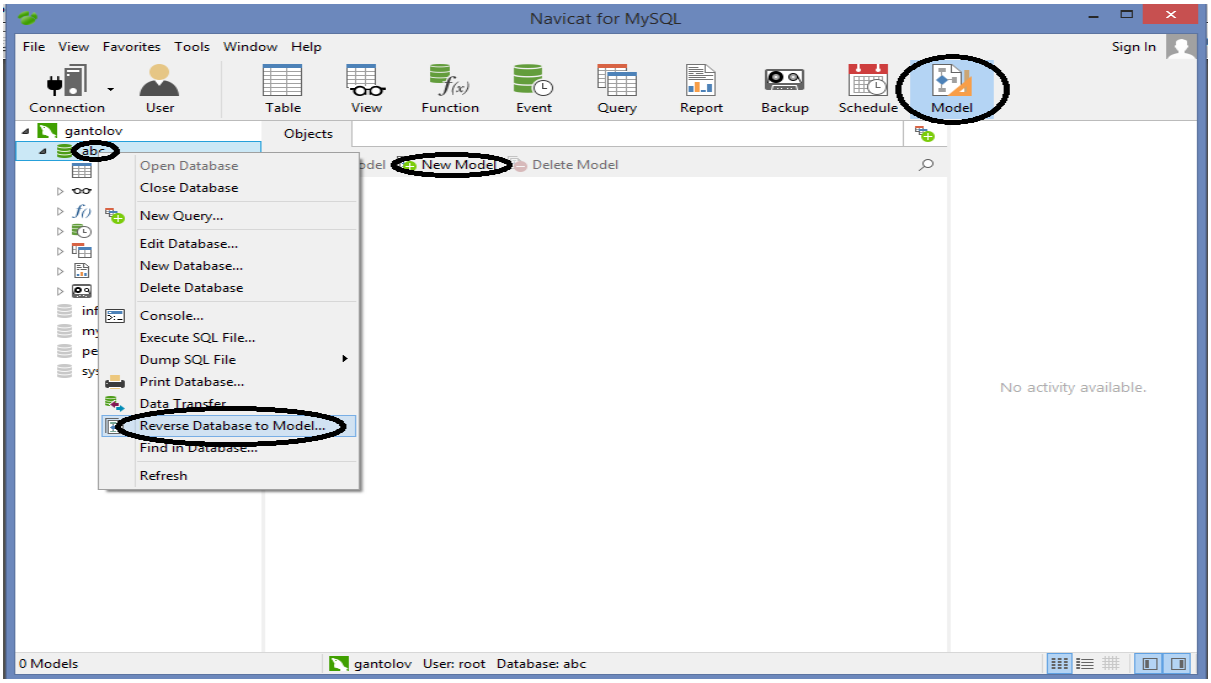
Nakon što je uspostavljena konekcija sa MySQL serverom, da bi mogli dalje raditi, potrebno je kreirati bazu podataka. Desni klik na ime konekcije koja se nalazi na lijevoj strani zaslona i odabere se NEW DATABASE. Otvara se novi prozor sa podacima o bazi koje treba upisati(ime baze, set znakova, te određivanje pravila). Nakon što su upisani svi podaci, klikne se OK, te je baza kreirana i smještena na lijevu stranicu zaslona.



Slika 4.7. Kreiranje nove baze

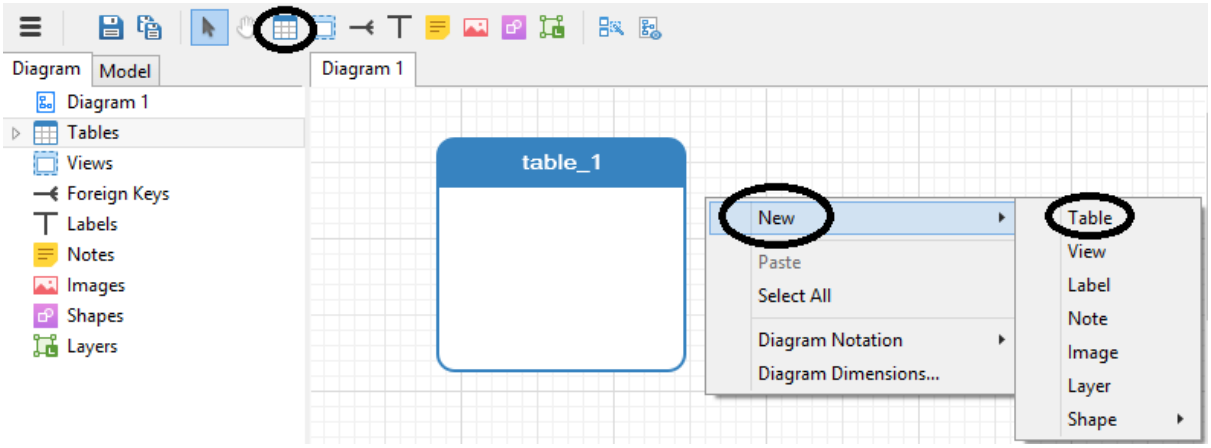
Slika 4.7. prikazuje postupak kreiranja nove baze. Klikne se na ime baze→NEW DATABASE, te kada se upišu podaci, stisne se OK.

Da bi kreirali konceptualni model, u alatnoj traci klikne se na MODEL. Ako se želi kreirati potpuno novi model, klik na NEW MODEL i novi se model otvori na zaslonu. Ako se želi uređivati neki model koji već postoji u bazi podataka, desni klik na ime baze, koje se nalazi sa lijeve strane, te zatim klik na REVERSE DATABASE TO MODEL. Nakon toga, model se pojavi na zaslonu. **Slika 4.8.** prikazuje postupak kreiranja modela u alatu Navicat.



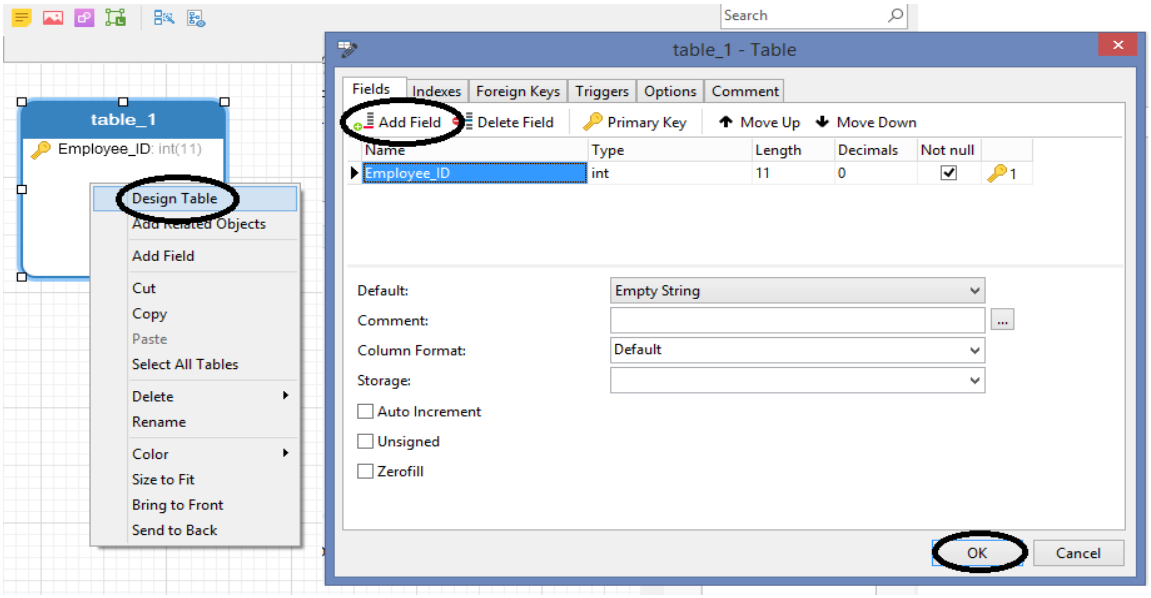
Slika 4.8. Kreiranje modela

Nakon što je konceptualni model kreiran, može se početi sa dodavanjem entiteta, veza i atributa. Ako se želi kreirati entitet, klikne se na simbol tablice, u alatnoj traci iznad modela, i dovuče se ju u model. Isto tako može se kliknuti desnim klikom na model i odabrati NEW→TABLE. **Slika 4.9.** prikazuje postupak kreiranja entiteta.



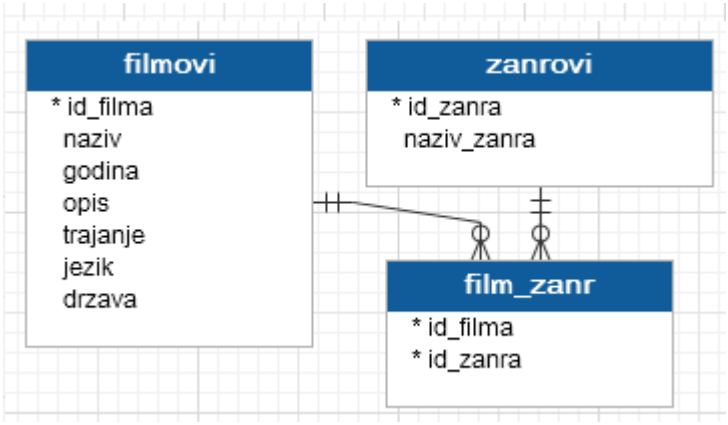
Slika 4.9. Kreiranje entiteta

Nakon što je entitet kreiran, može mu se dodavati atribute koji ga opisuju. Klikne se desni klik na entitet i odabere DESIGN TABLE. Otvori se prozor za uređivanje entiteta. U opciji FIELDS, klikom na ADD FIELD, dodaju se novi atributi. Nakon što su unijeti svi podaci o atributu, klik na OK, i atribut je dodan. **Slika 4.10.** prikazuje dodavanje atributa entitetu.



Slika 4.10. Dodavanje atributa

Veze između tablica kreiraju se pomoću primarnog i vanjskog ključa koje treba odrediti kod dodavanja atributa u tablicu. Isto tako, vanjske ključeve moguće je uređivati desnim klikom na vezu i opcija DESIGN RELATION. Kardinalnost veze može se odrediti tako da se klikne desnim klikom na vezu, zatim odabere CARDINALITY ON "naziv entiteta1" i CARDINALITY ON "naziv eniteta2", ovisno o tome za koju se tablicu mijenja kardinalnost. Opcije kardinalnosti koje su ponuđene jesu: "jedan", "više", "jedan ili više", "nula ili jedan!, "nula ili više".

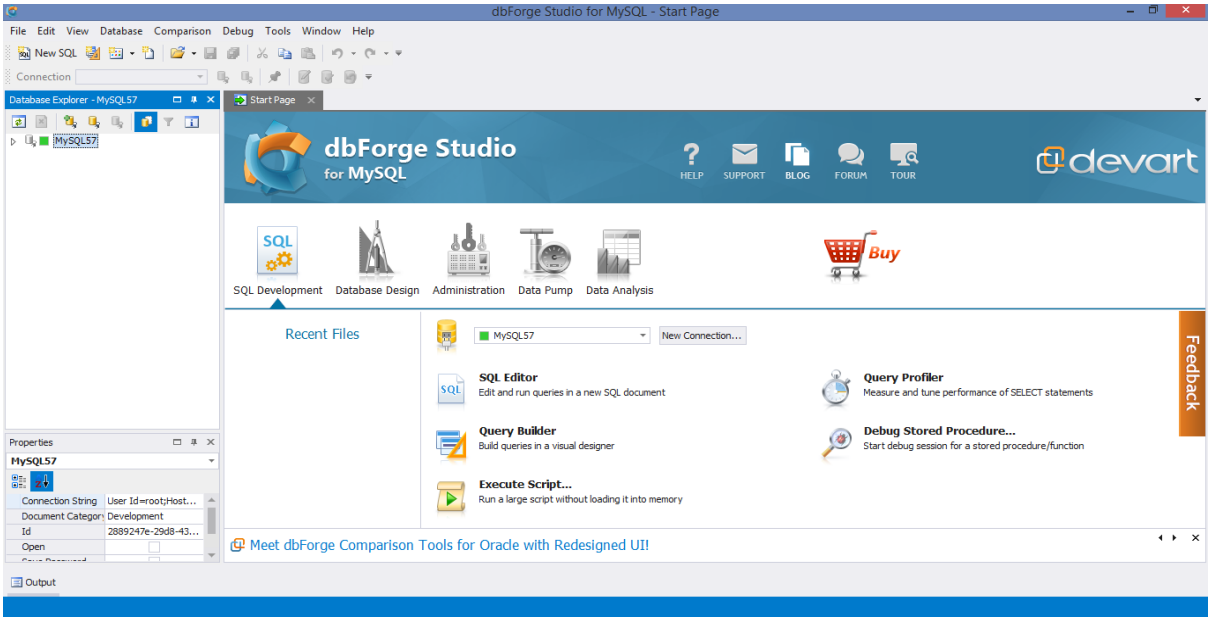


Slika 4.11. E-R model

Slika 4.11. prikazuje primjer E-R modela u Navicat alatu. Entiteti FILMOVI, FILM_ZANR i ZANROVI prikazani su tablicama i oni se povezuju definiranjem primarnih i vanjskih ključeva.

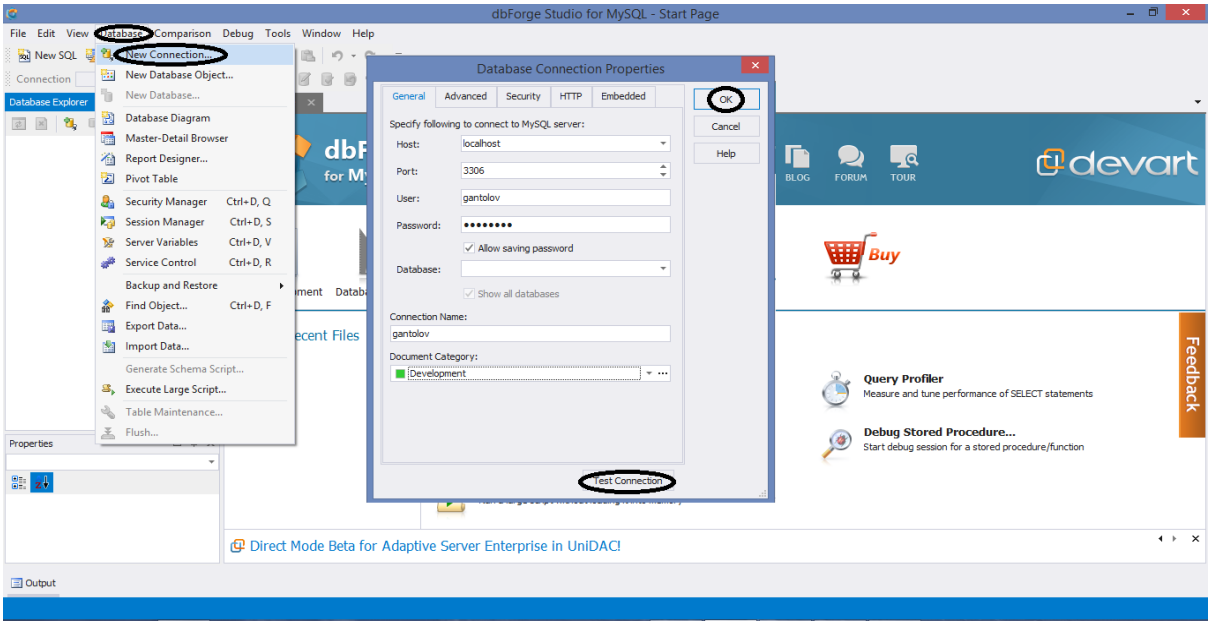
4.3. DBFORGE STUDIO

DbForge Studio je alat za konceptualno modeliranje podataka sa grafičkim sučeljem. Spaja se na SQL Server, Postgre SQL, MySQL, Oracle sustave za upravljanje bazom podataka. Radi samo na Microsoft Windows platformi. Ovdje će biti prikazan dbForge Studio za MySQL sustav za upravljanje bazom podataka. Najprije treba instalirati MySQL server te onda dbForge Studio for MySQL. Kad instalacija završi, otvara se početni zaslon i može se krenuti sa radom. Na početnom zaslonu, odmah se pojavljuju skupine funkcija kojima se može upravljati u dbForge Studio alatu: SQL Development, Database design, Administration, Data pump i Data analysis. **Slika 4.12.** prikazuje početni zaslon alata dbForge Studio.



Slika 4.12. Početni zaslon

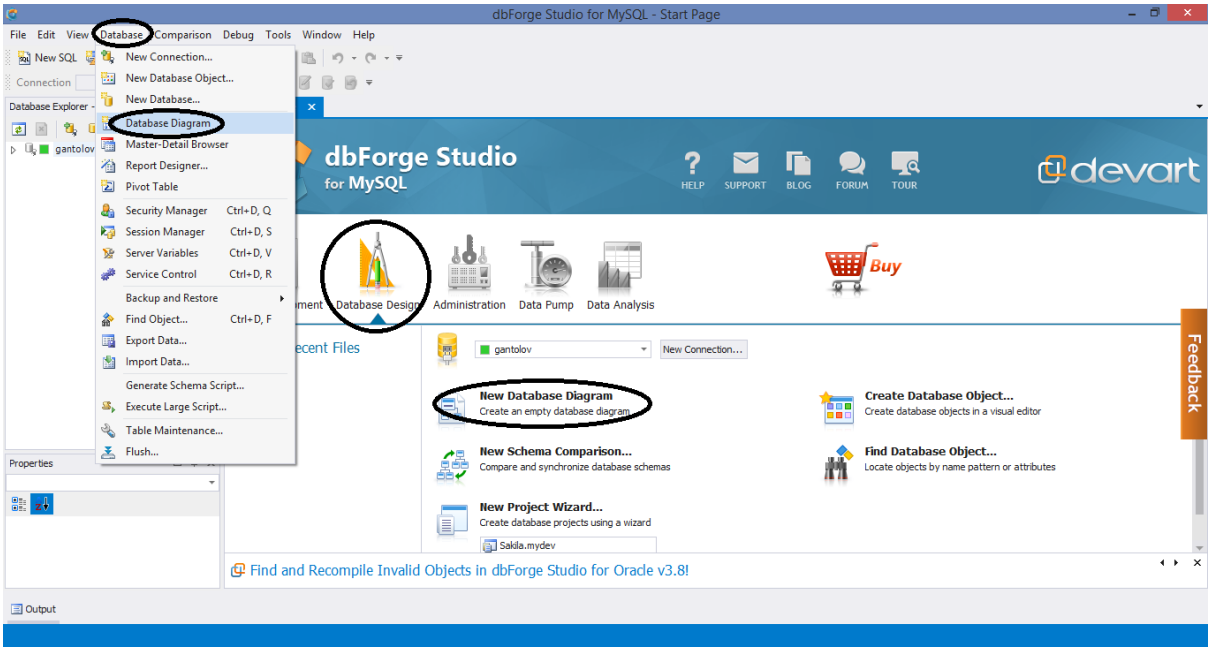
Prije početka rada u alatu, potrebno je povezati se na MySQL server. Ide se u alatnu traku, klik na DATABASE→NEW CONNECTION. Zatim se otvara prozor u koju se upisuju podaci o konekciji koju se želi napraviti. Treba upisati ime računala s kojega se pristupa, ime korisnika, lozinku, server na koji se povezuje... Nakon što se upišu svi podaci, klikne se na TEST CONNECTION da bi se provjerilo je li sve napisano kako treba, i ako je sve uredu, klik na OK i alat je spojen na MySQL server. **Slika 4.13.** prikazuje postupak spajanja na MySQL server



Slika 4.13. Spajanje na MySQL server

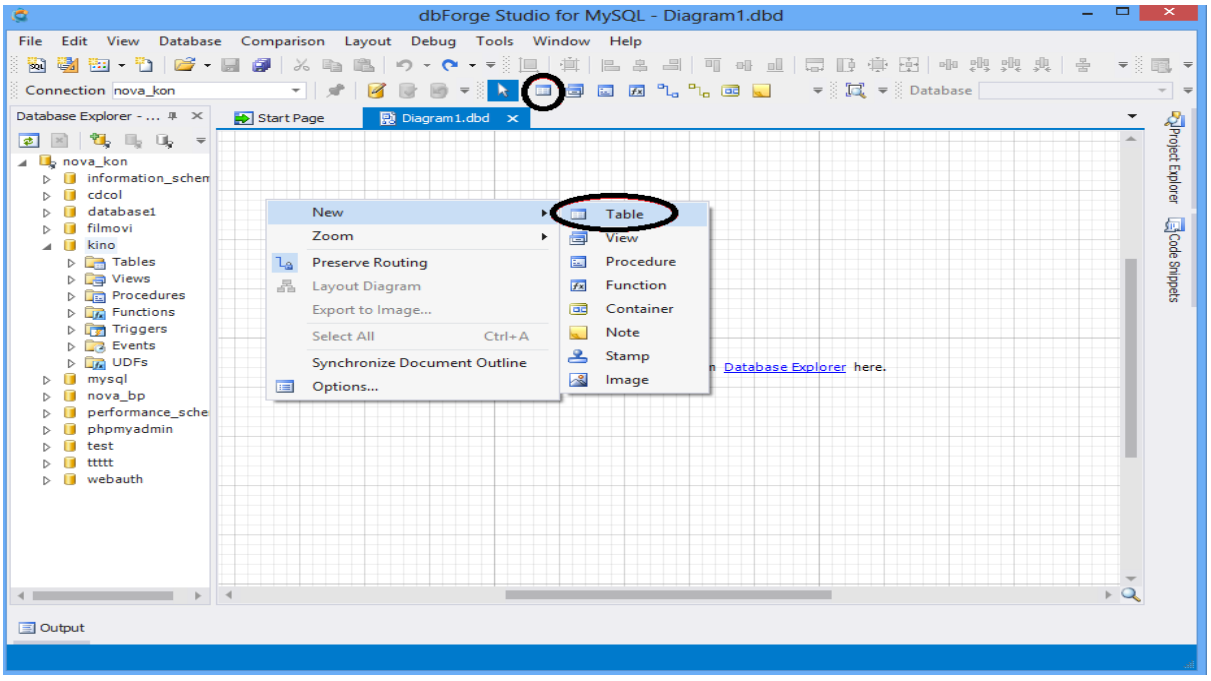
Nakon uspostavljanja veze sa MySQL serverom, da bi mogli dalje raditi, potrebno je kreirati bazu podataka. Desni klik na ime konekcije koja se nalazi na lijevoj strani zaslona i odabere se NEW DATABASE. Otvara se novi prozor sa podacima o bazi koje treba upisati(ime baze, set znakova, te određivanje pravila). Nakon što su upisani svi podaci, klikne se UPDATE DATABASE, te je baza kreirana.

Za kreiranje novog modela, u alatnoj traci klikne se DATABASE→DATABASE DIAGRAM. Novi model može se, također kreirati iz izbornika. Klik na DATABASE DESIGN u izborniku te se zatim odabere NEW DATABASE DIAGRAM. Koji god način odabrali, otvoriti će se model u koji se mogu dodavati entiteti, atributi i veze. **Slika 4.14.** prikazuje kreiranje modela u alatu dbForge Studio.



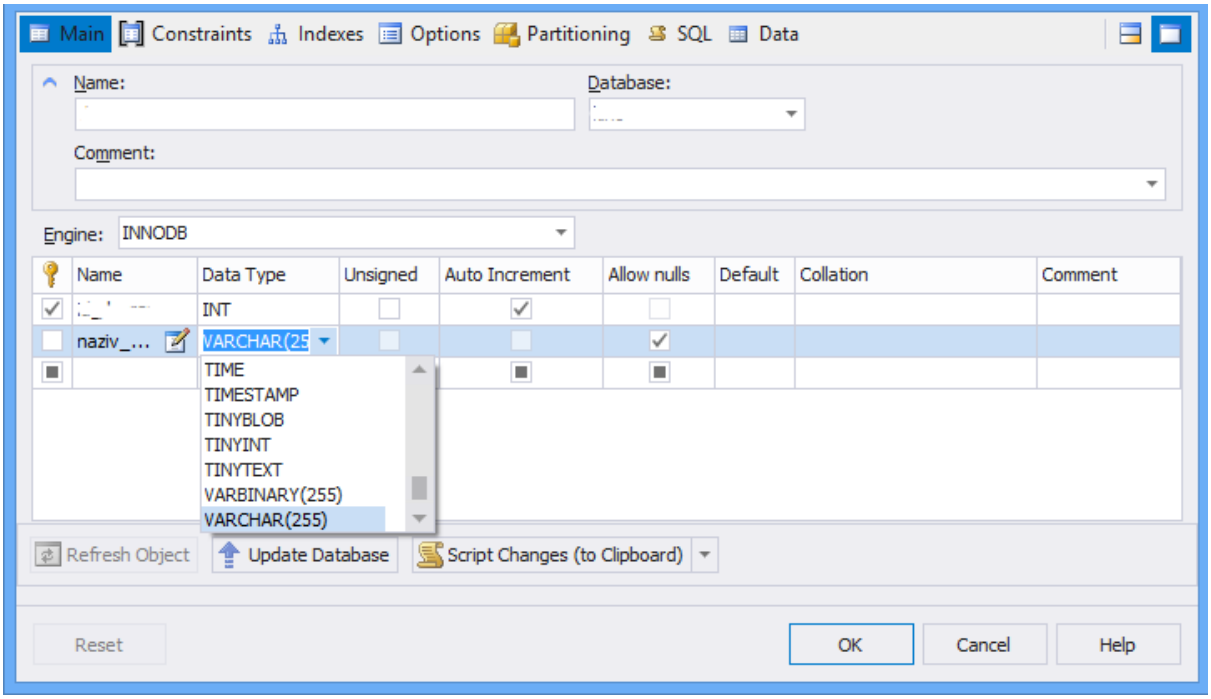
Slika 4.14. Kreiranje modela

Nakon što je model kreiran, može se krenuti sa crtanjem dijagrama, odnosno dodavanjem entiteta, veza i atributa. Entitet se može kreirati tako da se na alatnoj traci klikne znak tablice. Osim toga načina, entitet se može kreirati i tako da se klikne desni klik na površinu modela, i stisne NEW→ TABLE. **Slika 4.15.** prikazuje postupak kreiranja entiteta u dbFroge Studio alatu.



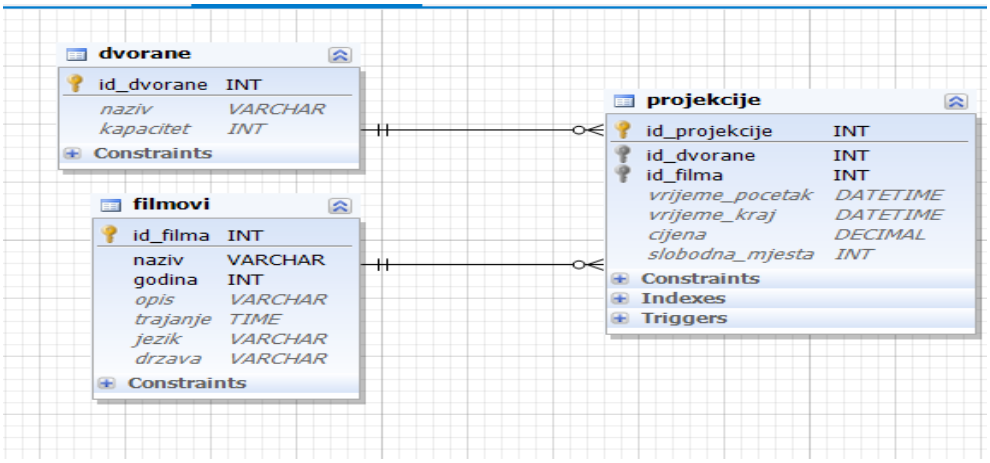
Slika 4.15. Kreiranje entiteta

Ako se kreira entitet preko NEW TABLE, otvara se prozor u kojem se upisuju svi podaci o novom entitetu. Ovdje se mogu kreirati i upisati atributi koji opisuju entitet. Treba upisati naziv atributa, koji je tip podatka te definirati primarni ključ. Nakon što su svi atributi upisani, mogu se dodati i vanjski ključevi, ako je to potrebno. Klikne se na CONSTRAINTS te zatim NEW FOREIGN KEY. Otvori se prozor u kojeg se upisuje naziv vanjskog ključa, odabere se koji će atribut biti vanjski ključ, te tablicu i atribut na koje će se odnositi taj vanjski ključ. Klikne se na OK i stvara se veza između tablica. Kada je upisivanje atributa gotovo, klikne se OK na glavnom prozoru, i novi entitet sa atributima je kreiran. **Slika 4.16.** prikazuje prozor za definiranje entiteta i atributa u dbForge Studio alatu.



Slika 4.16. Definiranje entiteta i atributa.

Definiranjem primarnih i vanjskih ključeva, određuju se i veze između entiteta. Kardinalnost veze se ne može mijenjati. Za mijenjanje i uređivanje vanjskih ključeva klikne se desni klik na liniju veze i odabere EDIT FOREIGN KEY. **Slika 4.17.** prikazuje primjer E-R modela u dbForge Studio alatu.



Slika 4.16. E-R model

Slika 4.17. prikazuje primjer E-R modela u dbForge Studio alatu. Entiteti DVORANE, FILMOVI i PROJEKCIJE prikazani su tablicama i oni se povezuju definiranjem primarnih i vanjskih ključeva.

4.4. USPOREDBA ALATA

U prošlom je dijelu općenito bio opisan rad triju alata za konceptualno modeliranje podataka sa grafičkim sučeljem: Toad, Navicat i dbForge Studio. Mora se naglasiti da se, za sva tri alata, radi o besplatnim verzijama sa smanjenim funkcijama. Pa će te verzije i biti uspoređene. Ako se želi nadograditi, odnosno raditi u verziji sa poboljšanom funkcionalnošću, tu verziju treba platiti.

Alati imaju poprilično sličan način rada. Razlike su uglavnom vizualne. Što se tiče vizualne preglednosti, najpregledniji i najjednostavniji, po mom mišljenju, je alat Navicat, iako dbForge Studio estetski najljepše izgleda. Navicat alat može raditi u Microsoft Windows, MAC DOS i Linux platformama, dok Toad i dbForge Studio podržavaju rad samo u Microsoft Windows platformi. Sva tri alata mogu se spojiti na poveći broj sustava za upravljanje bazom podataka. Navicat je jedini od alata na kojem je rad moguć i offline. Entiteti i atributi se u svim dijagramima kreiraju pomoću tablica. Što se tiče povezivanja entiteta u dijagramu, jedino Navicat ima ponuđen veći broj funkcija kod određivanja kardinalnosti veze. Toad se povezuje tako da je na jednom kraju veze znak ključa, a na drugom znak beskonačno, dok kod dbForge Studio alata jednom određene kardinalnosti se ne mogu mijenjati. U svim alatima se veze određuju definiranjem primarnih i vanjskih ključeva.

	OS	SUBP	OFFLINE NAČIN RADA
TOAD	**MICROSOFT WINDOWS**	**ORACLE,** **MYSQL** **POSTGREE SQL** **SQL SERVER** **SQLITE**	**NE**
NAVICAT	**MICROSOFT WINDOWS** **MAC OS** **LINUX**	**ORACLE** **MYSQL** **POSTGREE SQL** **SQL SERVER** **SQLITE** **MARIADB**	**DA**
DBFORGE	**MICROSOFT WINDOWS**	**ORACLE** **MYSQL** **POSTGREE SQL** **SQL SERVER**	**NE**

Tablica 4.1. Usporedba alata

Tablica 4.1. prikazuje razlike između alata koje se odnose na podržavane platforme, podržavane sustave za upravljanje bazom podataka te podržavaju li alati rad offline. Iz tablice je vidljivo da Navicat alat može raditi u Microsoft Windows, MAC DOS i Linux platformama, dok Toad i dbForge Studio podržavaju rad samo u Microsoft Windows platformi. Navicat se može spojiti na: ORACLE, MYSQL, POST GREE SQL, SQL SERVER, SQLITE, MARIADB sustave za upravljanje bazom podataka i podržava offline način rada, što znači da za rad na alatu nije potrebno biti spojen na internet. Toad se može spojiti na: ORACLE, MYSQL, POST GREE SQL, SQL SERVER, SQLITE i ne podržava offline način rada, što znači da je za rad na alatu potrebna veza na internet. DbForge Studio se može spojiti na: ORACLE, MYSQL, POST GREE SQL, SQL SERVER i, također, ne podržava offline način rada, kao ni Toad.

5. ZAKLJUČAK

Dobar konceptualni model je bitan, ako ne i najbitniji, u razvoju baze podataka. Pošto izrada informacijskog sustava započinje modeliranjem baze podataka, možemo reći da dobar konceptualni model ima ključnu ulogu u izradi informacijskog sustava. Nakon pisanja ovog završnog rada i proučavanja, još sam više uvjeren u tu činjenicu. Jako je važno da se dovoljno vremena posveti izradi konceptualnog modela jer o njemu ovisi daljnji razvoj i rad baze podataka. Treba paziti da se upišu svi bitni podaci kako kasnije ne bi došlo do izdavanja krivih informacija ili nedostatak istih.

Pošto dolazi do sve većeg razvoja informatičkog sustava, potrebni su sve složeniji i detaljniji konceptualni modeli, a samim time i precizniji. Razvojem sve većeg broja alata za konceptualno modeliranje podataka sa grafičkim sučeljem, izrada takvih konceptualnih modela je uvelike pojednostavljena. Alati koji su opisani u ovom radu pokazali su se poprilično pregledni i jednostavni za rad. Osim što omogućavaju kreiranje konceptualnog modela i oblikovanje podataka u modelu, alati omogućavaju, također, pretvorbu konceptualnog modela u logički model, te omogućavaju i izradu baze podataka.

6. LITERATURA

Manger, R. (2014.), *Baze podataka*. Zagreb: Element

Varga, M. (1994.), *Baze podataka: Konceptualno, logičko i fizičko modeliranje podataka*. Zagreb: DRIP

Song, I., Evans, M. i Park, E. (1994.), *A Comparative Analysis of Entity-Relationship Diagrams.* Vol.3 (No.4). str. 427.-459.

Bevanda, V. (2015.), *Materijali s predavanja*, http://e-ucenje.oet.unipu.hr/

http://www.databaseanswers.org/modelling_tools.htm

7. SAŽETAK

Napretkom i razvojem informacijske tehnologije javlja se potreba za sve složenijim i naprednijim informacijskim sustavima. Informacijski sustav je model poslovnog sustava. To je skup elemenata koji prikazuju dijelove stvarnog svijeta. Pod-tipovi informacijskog sustava su: Model procesa, Model podataka i Model resursa odnosno izvršitelja.

Izrada informacijskog sustava započinje modeliranjem baze podataka. Podaci su simbolički prikaz elemenata stvarnog svijeta. Oni predstavljaju jedno obilježje nekog subjekta ili događaja. Ako nekom podatku dodamo kontekst i predstavimo na određeni način, tada on postaje informacija. Informacija je skup podataka koji imaju neko značenje. Modeliranje informacijskog sustava započinje utvrđivanjem i analizom zahtjeva korisnika za informacijama, a završava izgradnjom stabilne i prilagodljive baze podataka. Model podataka pokazuje koji su podaci kojima raspolaže organizacija. Podaci informacijskog sustava spremaju se u bazu podataka. Baza podataka je model podataka poslovnog sustava, koji je segment stvarnog svijeta. To je skup podataka koji su obrađeni u jednoj organizaciji. Da bi se baza podataka mogla implementirati, njen razvoj prije mora proći kroz različite faze. Oblikovanje podataka je vrlo bitno i ključno u razvoju baze podataka. Cilj modeliranja podataka je da se odredi pogodna struktura baze za određene podatke i da ti podaci budu međusobno nezavisni.

Konceptualno modeliranje podataka bitan je i osnovni dio modeliranja podataka. Konceptualnim modeliranjem dobivamo konceptualni model podataka. Konceptualni model opisuje podatke koji su dio strukture cijelog informacijskog sustava, a izgrađuju ga projektanti koji surađuju sa korisnicima kako bi bolje razumjeli koji su podaci i koje informacije važne i potrebne. Za modeliranje konceptualnog modela koriste se koncepti: entiteti, veze i atributi, a za prikaz modela upotrebljavaju se dijagrami.

E-R model proučava, opisuje i promatra stvarni svijet preko entiteta, odnosa između entiteta, i atributa entiteta ili njihovog odnosa. Za promatranje stvarnog svijeta koristi se apstrakcija pomoću koje se lakše uočava i odvaja ono što je bitno od onoga što je nebitno ili manje bitno. Za konceptualno modeliranje podataka u E-R modelu koriste se tri vrste ili postupka apstrakcije. To su klasifikacija, generalizacija i agregacija. E-R je skraćenica od

eng. Entity-Relationship pa se ovaj model zove još i model entiteti-veze gdje veze pokazuju odnose između entiteta, a atributi opisuju entitete.

Entitet je nešto što postoji i što se u stvarnom svijetu može identificirati. Može biti objekt koji ima mnogo instanci u bazi podataka, može biti sastavljen od više atributa i može biti objekt kojeg sami modeliramo. Veza opisuje odnos između entiteta. Ovisno o broju entiteta može biti unarna, binarna, ternarna ili k-narna veza i takva podjela zove se podjela po stupnju veze. Veze između entiteta dijele se još po kardinalnosti i opcionalnosti. Atribut je obilježje entiteta. Služi kao opis entiteta i olakšava određivanje stanja tog entiteta. Postoje indikacijski i opisni atributi.

E-R model omogućava izradu dobrog grafičkog prikaza. Samim time znatno je lakše razumjeti model podataka koji je prikazan. Dijagram E-R modela najčešće se zapisuje Chenovim, Martinovim i Barkerovim prikazom. Chenov prikaz dobio je ime po Peteru Chenu koji je E-R dijagram predstavio 1967. godine. Prikazuje entitete, veze i atribute. Entiteti se u dijagramu prikazuju pravokutnikom. Veze se prikazuju simbolom romba koji je linijom spojen za entitete. Simbol za atribut je elipsa ili krug. Oni su spojeni linijom za entitet kojega opisuju. U Martinovu se prikazu E-R modela entiteti prikazuju simbolom pravokutnika. Atributi se prikazuju pomoću elipse ili kruga. Veze se u Martinovom prikazu crtaju ravnom linijom između dva entiteta. Veza se ne imenuje nego se navode uloge entiteta koje veza spaja. U ovom prikazu koriste se samo unarne i binarne veze. Barkerov prikaz nazivamo još i Oracle Case method. Entiteti se u prikazu označavaju pravokutnikom. Atribute se upisuje u pravokutnik entiteta tako da budu navedeni ispod naziva entiteta. Veza se u Barkerovu prikazu prikazuje kao linija između dva entiteta.

Logičko modeliranje podataka je druga faza u modeliranju baze podataka. U ovoj se fazi konceptualni model pretvara u relacijski model podataka. Da bi se baza podataka mogla implementirati potrebno je napraviti relacijsku shemu, jer se većina današnjih sustava za upravljanje bazom podataka koristi relacijskim modelom. U relacijskom se modelu entiteti i veze između entiteta pretvaraju u relacije tj. tablice. Relacijski je model, dakle, model u kojem se baza podataka sastoji od skupa relacija. Relacija je tablica u modelu sa stupcima i redovima koja predstavlja entitet ili vezu. Vrijednost atributa prikazuje se stupcem u relaciji. U relacijskom se modelu, za prikazivanje tipa entiteta, koristi relacijska shema. Za prikaz skupa entiteta koristi se relacija, odnosno tablica. Entitet se prikazuje jednom n-torkom, odnosno jednim retkom u tablici. Atributi koji opisuju entitet postaju atributi relacije. Primarni ključ entiteta postaje primarni ključ relacije. Veze između entiteta nastaju

povezivanjem vanjskog ključa jedne relacije sa primarnim ključem druge relacije ili se za prikazivanje veza stvara nova relacija.

Razvojem informacijskog sustava povećava se potreba i važnost izrade dobrog konceptualnog modela podataka. Sukladno tome, dolazi do stvaranja i razvoja alata za konceptualno modeliranje podataka sa grafičkim sučeljem koji olakšavaju izradu modela. Danas već postoji veliki broj alata za konceptualno modeliranje podataka koji se mogu spajati na razne sustave za upravljanje bazom podataka. Toad je alat za konceptualno modeliranje podataka. Može se spajati na Oracle, SQL server, MySQL, SQLite, Postgre SQL.Radi samo na Microsoft Windows platformi. Navicat je alat za konceptualno modeliranje podataka sa grafičkim sučeljem koji se spaja na Oracle, SQL Server, MySQL, SQLite, PostgreSQL, MariaDB sustave za upravljanje bazom podataka Radi na Microsoft Windows, MAC OS i Linux platformi. DbForge Studio je alat za konceptualno modeliranje podataka sa grafičkim sučeljem. Spaja se na SQL Server, Postgre SQL, MySQL, Oracle sustave za upravljanje bazom podataka Radi samo na Microsoft Windows platformi. Alati imaju poprilično sličan način rada. Razlike su uglavnom vizualne. Pregledni su i jednostavni za rad. Osim što omogućavaju kreiranje konceptualnog modela i oblikovanje podataka u modelu, alati omogućavaju, također, pretvorbu konceptualnog modela u logički model, te omogućavaju i izradu baze podataka.

SUMMARY

With the progress and development of information technologies there is a necessity for more complex and advanced information system. The Information system is a model of the business system. That is a set of elements that shows parts of real world. The sub-types of the information systems are: process model, data model and resources or executor model.

The construction of the IT system begins with database modeling. Data is a symbolic representation of real world elements. They represent a characteristic of some subject or event. If we add a contest to a data and represent it in a certain way, then it becomes an information. An information is a set of data that has some meaning. The modeling of the IT system begins with the identification and analysis of user requirements for information, and ends with the construction of a stable and flexible database. The data model shows what data are available to the organization. The data of the IT system are stored in the database. The database is a data model of the business system, which is a segment of the real world. It is a set of data that are processed in an organization. Before the database could be implemented, its development must go through different phases. Data modeling is very important and crucial for database development. The goal of data modeling is to determine the appropriate database structure for specific data and that the data are independent of each other.

Conceptual data modeling is an important and essential part of data modeling. With the conceptual data modeling we raise a conceptual data model. The conceptual model describes data that are part of the structure of the whole IT system, and it's built by designers who collaborate with customers to better understand what data and what information are important and necessary. Concepts: entities, relationships and attributes are used for modeling a conceptual model, and diagrams are used for model view.

The E-R model studies, describes and observes the real world through entities, relationships between entities and entity attributes and their relationships. For the observation of the real world there is an abstraction used to help to notice and separate what is important from what is irrelevant or less important. For the conceptual data modeling in the E-R model there are three types or process of abstraction. These types are called classification, generalization and aggregation. E-R stands for Entity-Relationship, so in this model, links shows relationships between entities and attributes describes entities.

An entity is something that exists and can be identify in the real world. It can be an object that has many instances in the database, can be composed of multiple attributes and it can be an object that we can model ourselves. The bound describes the relationship between the entities. such a division is called the division by the degree of connection. Depending on the number of entities, the bound can be unary, binary, ternary or N-ary bound and such a split is called the connection degree split. The bounds between entities are divided also by cardinality and optionality. An attribute is a characteristic of the entity. It serves as a description of entities and facilitates the determination of the status of that entity. There are indication and descriptive attributes.

The E-R model enables the creation of a good graphic display. Therefore, it's much easier to understand what is the data model showing. To write an E-R model diagram, we usually use Chen, Martin, and Oracle case method notation. Chen notation was named after Peter Chen who presented his diagram in 1967. It shows entities, relationships and attributes. Entities are represented by rectangles. Relationships are represented by a turbot symbol which is connected to entities. The symbol for the attribute is an ellipse or a circle. They are connected to the entity that describe with lines. In Martin notation of E-R model entities are represented by rectangles. Attributes are represented by ellipses or circles. Relationships are draw by a straight line between two entities. We don't name the relationship, but we do indicate the roles of the entities that the relationship bonds. There are only unary and binary relationships in this notation. In the Oracle case method notation entities are represented by rectangles. The attributes are inserted in the entity rectangle, in the way that they are listed below the name of the entity. Relationships are represented by a line between two entities.

Logical data modeling is the second phase of database modeling. At this stage, a conceptual model turns into a relational data model. In order to be able to implement the database it is necessary to make a relational schema, since most of today's database management systems uses a relational model. In the relational model, entities and bounds between entities converts into relations/tables. So, a relational model is a model where the database consists of a set of relations. A relation is a table in the model with columns and rows that represents an entity or a bound. The value of an attribute is displayed by a column in the table. A relational schema is used to display the type of the entity in the relational model. A relation, or table is used to show a set of entities. An entity is shown by one row in the table. The attributes that describe the entity become attributes of relation. The primary key of the entity becomes the primary key for the relation. Bounds between the entities are

formed by connecting an external key of one relation with the primary key of another relation, or they can be displayed with the creation of a new relation.

With the development of the IT system, the necessity and the importance of making a good conceptual data model increase. Accordingly to that, there comes the generation and development of conceptual data modeling tools with a graphical interface that facilitate the development of a model. Today there is already a large number of conceptual data modeling tools that can be connected to a variety of database management systems. Toad is a conceptual data modeling tools. It can be connected with Oracle, SQL server, MySQL, SQLite, Postgre SQL DBMS. It works only on the Microsoft Windows platform. Navicat is a conceptual data modeling tool with a graphical interface that connects to Oracle, SQL Server, MySQL, SQLite, PostgreSQL, MariaDB DBMS. It works on Microsoft Windows, MAC OS and Linux platform. DbForge Studio is a conceptual data modeling tool with a graphical interface. It connects to SQL Server, Postgre SQL, MySQL and Oracle database management system. Works onl on Microsoft Windows platform. The tools have a quite similar working mode. The diferences are mostly visual. They are transparent and easy to work on. In addition to enabling the creation of a conceptual model and data modeling in the model, the tools allow, also, the conversion of a conceptual model into a logical model, and allow the creation of a new database.