



**ALGEBRA
BERNAYS**
SVEUČILIŠTE

**UVOD U BAZE
PODATAKA**

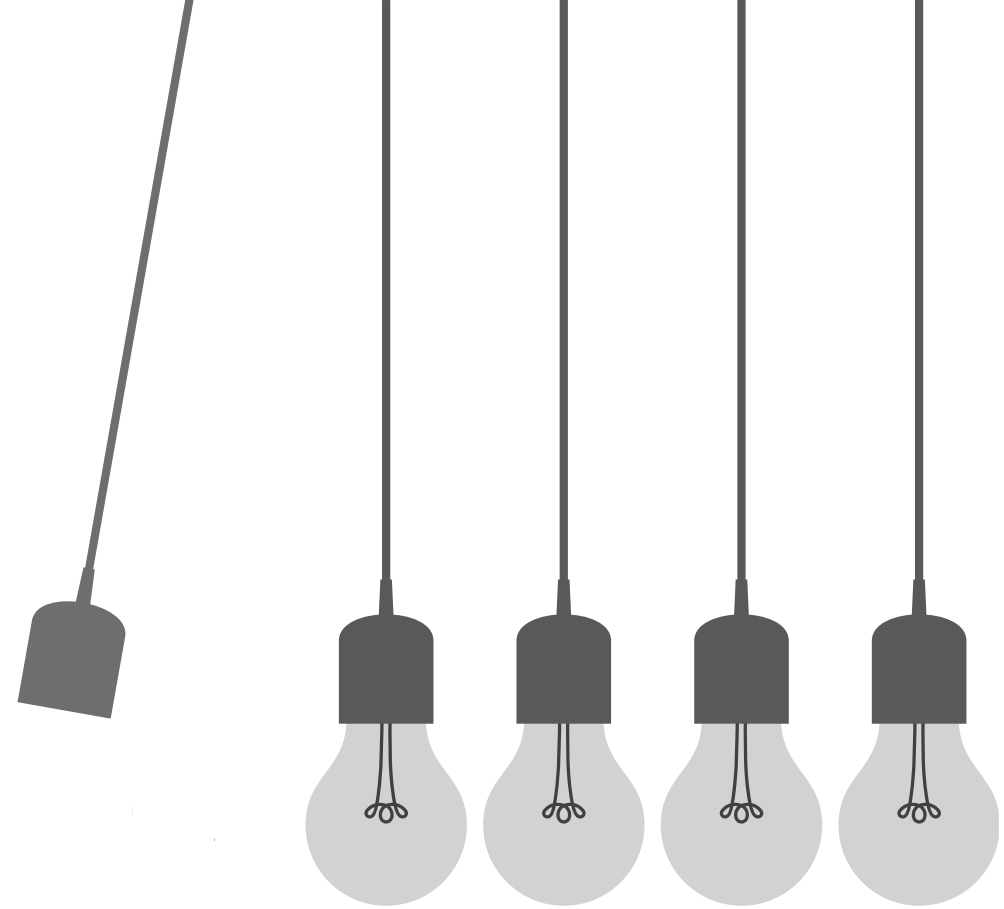
Upute za pohađanje i polaganje
kolegija
akademska godina 24/25

Organizacija predavanja i vježbi

| | | |
|----------------------------|---------------------|--|
| Nositelji kolegija: | Mario Fabijanić | mario.fabijanic@algebra.hr |
| | Mateo Sokač | mateo.sokac@algebra.hr |
| Asistenti: | Aron Thompson | aron.thompson@algebra.hr |
| | Matija Kolarić | matija.kolaric@algebra.hr |
| | Silvije Davila | silvije.davila@algebra.hr |
| Izvođenje nastave: | Predavanja | 2 sata tjedno - ukupno 30 sati |
| | Vježbe | 2 sata tjedno - ukupno 30 sati |
| | Konzultacije | Prilikom vježbi ili e-mailom |

Informacije o kolegiju

- 5 ECTS bodova = 150 sati rada studenta
 - 30 sati predavanja
 - 30 sati vježbi
 - 90 sati rada kod kuće
(15 tjedana po 6 sati)



Cilj kolegija

Naučiti dizajnirati jednostavne relacijske baze podataka.

Naučiti koristiti programski jezik SQL za dohvaćanje, umetanje, ažuriranje i brisanje podataka iz relacijske baze podataka.

Ishodi učenja

| Ishod | MINIMALNI ISHODI UČENJA (po uspješnom završetku kolegija, student će moći) | ŽELJENI ISHODI UČENJA (uspješan student bi trebao moći) |
|-------|---|--|
| I1 | Konstruirati relacijski model podataka počevši od korisničkih zahtjeva. | Konstruirati relacijski model podataka počevši od korisničkih zahtjeva, uključujući podskupove i involuirane odnose. |
| I2 | Izraditi bazu podataka koristeći DDL naredbe na temelju relacijskog modela. | Razlikovati učinke ograničenja na relacijskom modelu. |
| I3 | Predložiti izmjene relacijskog modela kako bi se dosegla viša normalna forma. | Predložiti izmjene kompleksnog relacijskog modela kako bi se dosegla viša normalna forma. |
| I4 | Koristiti osnovne naredbe za upravljanje podacima u relacijskoj bazi podataka. | Riješiti problem korištenjem spajanja. |
| I5 | Konstruirati rješenje korištenjem sistemskih i agregatnih funkcija te grupiranja. | Konstruirati kompleksno rješenje korištenjem sistemskih i agregatnih funkcija te grupiranja. |
| I6 | Primijeniti podupite kod izrade kompleksnijeg upita. | Riješiti problem kod izrade kompleksnijeg upita podupitom, uključujući podupit u FROM dijelu upita. |

Tematske cjeline

| Tjedan nastave | Cjelina | Tjedan nastave | Cjelina |
|----------------|--|----------------|--|
| 1. Tjedan | Entiteti i atributi Modeli podataka | 9. Tjedan | Unutarnje spajanje tablica |
| 2. Tjedan | Odnosi među entitetima Dijagram entiteta i ER model | 10. Tjedan | Vanjska spajanja i unakrsno spajanje tablica |
| 3. Tjedan | Relacijski model i ključevi | 11. Tjedan | Funkcije |
| 4. Tjedan | Pretvaranje ER sheme u relacijsku i normalizacija | 12. Tjedan | Grupiranje i agregatne funkcije |
| 5. Tjedan | Implementacija relacijskog modela | 13. Tjedan | Podupiti |
| 6. Tjedan | SELECT naredba | 14. Tjedan | Podupiti |
| 7. Tjedan | INSERT, UPDATE i DELETE naredbe | 15. Tjedan | Ponavljjanje |
| 8. Tjedan | Operatori | | |

Literatura

OBAVEZNA LITERATURA

- Harrington, J. L. (2016) *Relational Database Design and Implementation*. 4th edn. Amsterdam: Elsevier Inc.

PREPORUČENA LITERATURA

- Churcher, C. (2012) *Beginning Database Design: From Novice to Professional*. New York City: Apress.

Za potpis treba?

Za stjecanje prava na potpis potrebno je prisustvovati nastavi u postotku propisanom Pravilnikom o studijima i studiranju

| DOLAZNOST NA PREDAVANJA I VJEŽBE | |
|---|---|
| Najmanje 50% fizičke prisutnosti na predavanjima | Najmanje 60% fizičke prisutnosti na vježbama |

Tko ne dobije potpis, mora sljedeće godine ponovno upisati kolegij, platiti upis kolegija te nema pravo polaganja ispita.

Polaganje kolegija

- Kolegij ima definiranih 6 ishoda učenja
- **Da bi student položio kolegij mora po svakom ishodu učenja ostvariti minimalno 50% bodova raspoloživih za taj ishod učenja**
- **Ako student ne ostvari 50% bodova iz nekog ishoda učenja, na sljedećem roku može opet polagati taj ishod učenja.**
- Metode provjeravanja skupova ishoda učenja:
 - Međuispiti
 - Školske zadaće

Kako je to raspoređeno po ishodima učenja

| Ishod | M1 | M2 | Školska zadaća | MAX |
|--------|----|----|----------------|-----|
| I1 | 21 | | 2 | 23 |
| I2 | 9 | | 2 | 11 |
| I3 | 5 | | 2 | 7 |
| I4 | | 30 | 2 | 32 |
| I5 | | 12 | 2 | 14 |
| I6 | | 13 | | 13 |
| Ukupno | 35 | 55 | 10 | 100 |

Ocjenjivanje

| Broj osvojenih bodova | Ocjena |
|-----------------------|----------------|
| 0,00 – 50,00 | 1 (nedovoljan) |
| 50,01 – 58,00 | 2 (dovoljan) |
| 58,01 – 75,00 | 3 (dobar) |
| 75,01 – 92,00 | 4 (vrlo dobar) |
| 92,01 – 100,00 | 5 (izvrstan) |

Ispiti

- Na svakom kolegiju vrijedi **pravilo 3 + 1**
 - To znači da student mora položiti ispit iz najviše 4 izlaska
 - 3 redovna izlaska – Uključena u cijenu školarine
 - 1 izvanredni izlazak – Odlukom o naknadi troškova 4. prijava ispita se naplaćuje
 - Vremenski rok za položiti ispit je **12 mjeseci** od dana upisa kolegija
 - Ako student u 12 mjeseci ne položi kolegij, **mora ponovno upisati kolegij te ponovno polagati sve ishoda učenja kako je definirano kolegijem**
- **Vodite računa o rokovima prijave i odjave ispita na IE**
 - Ako niste prijavili ispit na vrijeme, ne možete pristupiti ni pismenom niti usmenom dijelu
 - Ako je student prijavio više ispitnih rokova iz istog kolegija, pri dobivanju ocjene kojom je zadovoljan, dužan je odjaviti svaki sljedeći rok koji je iz tog kolegija prijavio. U suprotnom, studentu se u Infoeduku unosi nedovoljan (1)

Školske zadaće

- Školska zadaća provodi se na nastavi putem blic testova, a sastoje se od 1-2 kratkih pitanja
- Blic testovi se pišu na vježbama
- Na blic testovima piše se do tada obrađeno gradivo
- **Blic testovi se ne ponavljaju, nije ih moguće pisati naknadno i nije ih moguće ispravljati (ili popravljati)**
- Blic testovi se *neće najavljivati*

Akademski standard ponašanja

- U komunikaciji (pisanoj i usmenoj) pridržavati se pravila poslovne komunikacije primjerene akademskoj razini
- Potrebno je držati se jasno definiranih rokova za predaju zadataka (školske zadaće, ispiti)
 - Svaki zadatak poslan nakon definiranog roka neće se ocjenjivati
- Samo oni studenti koji mogu potvrditi svoje pohađanje, smatrat će se prisutnima
 - Potpisivanje drugih studenata ili registracija njihovom karticom nije dopušteno i može biti predmet stegovnog postupka. Nastavnik će obrisati prisustvo ako utvrdi da je student prijavljen, a da nije prisutan na nastavi

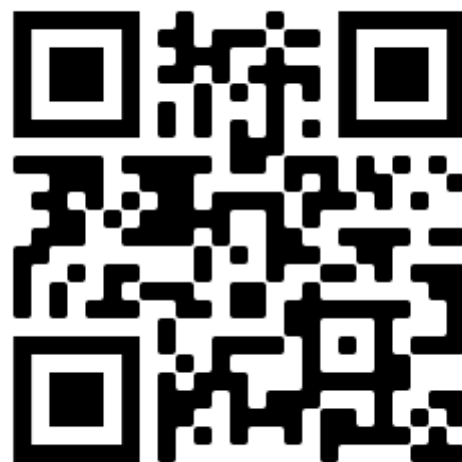
Pravila ponašanja na nastavi – fizička prisutnost

- Na nastavu se dolazi na vrijeme
- Pri ulasku u učionicu student prilazi do stola i prijavljuje se na nastavu karticom te sjeda na dostupno mjesto za rad
- Ometanje nastave i neaktivno sudjelovanje na nastavi nije dozvoljeno
 - Repetitivno kršenje ovog pravila sankcionira se prijavom Stegovnom povjerenstvu

Pitanja?

Anketa

- <https://bit.ly/37ZuJaa>



Realni svijet

Entiteti i instance entiteta

- **Entitet** je bilo što o čemu možemo razmišljati i prikupljati informacije
 - Čovjek, Student, Mačka, Životinja, Seminar, Vrsta polaznika, Račun, Usluga na računu, Način plaćanja računa
- **Instance entiteta** su točno određeni objekti u realnom svijetu
 - Točno taj Mirko je instanca entiteta Čovjek
 - AMEX kreditna kartica je instanca entiteta Način plaćanja računa
 - Četvrti semestar je instanca entiteta Semestar
- Entitete možemo smatrati strukturama, a instance entiteta možemo smatrati objektima

Atributi

- Entiteti posjeduju **niz atributa** koji ih opisuju
 - Entitet Čovjek posjeduje atribut Datum rođenja
 - Entitet Mačka posjeduje atribut Boja dlake
- **Svaka instanca entiteta sadržava svoje vrijednosti atributa**
 - Instanca Točno taj Mirko ima vrijednost atributa Datum rođenja jednaku 17.12.1953.
 - Instanca Garfield posjeduje vrijednost atributa Boja dlake jednaku Narančasta
- Attribute možemo smatrati članovima strukture
 - Vrijednost je ono što je dodijeljeno atributu na nekoj instanci

Ovisnost atributa

- Vrijednosti atributa mogu ovisiti o nekom parametru, najčešće o vremenu:
 - Neka instanca entiteta Čovjek ima vrijednost atributa Starost jednaku 9125 dana, sutra će biti 9126 dana, prekosutra 9127 dana, ...
- Takve attribute bismo trebali izbjegavati u bazama podataka
 - Umjesto atributa Starost bolje je čuvati atribut Datum rođenja
 - Starost se izračunava za željeni trenutak

Informacije i podaci

- Elementarnu informaciju čine instanca entiteta, atribut i njegova vrijednost
- Primjer: "Mirko je visok 178 cm"
 - Entitet: Čovjek
 - Instanca entiteta: Točno taj Mirko
 - Atribut: Visina
 - Vrijednost atributa: 178 cm
- Podatak je zapisana informacija
 - Zapisana na papiru
 - U memoriji računala
 - Na CD-u, DVD-u, ...

Microsoft SQL Server

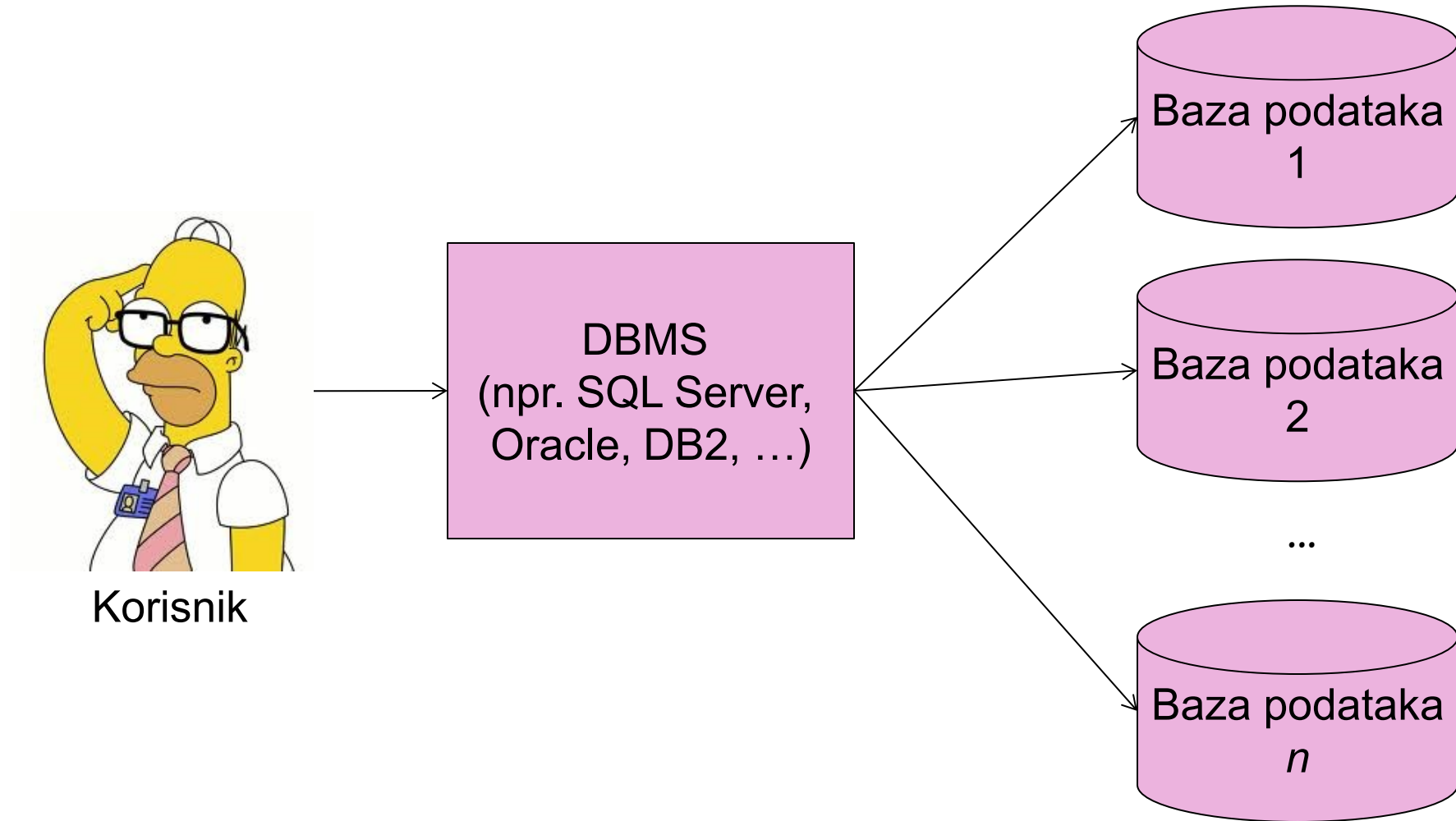
- Za ovaj kolegij nam je potreban Microsoft SQL Server
- Sastoji se od serverskog i klijentskog dijela
 - Klijentski dio: MS SQL Server Management Studio
 - Služi za komunikaciju i zadavanje naredbi serverskom dijelu
- **Za naše potrebe, verzije 2005, 2008 (R2), 2012, 2014, 2016, 2017 i 2019 su dovoljno slične**
- Potrebno ga je instalirati i koristiti za vježbanje
 - Dostupan i kao besplatna inačica SQL Server 2017 Express
 - <https://www.microsoft.com/en-us/download/details.aspx?id=55994>

Baze podataka

DBMS i baze podataka

- Baza podataka je **skup međusobno povezanih podataka**
- Sustav za upravljanje bazom podataka se kraće naziva DBMS (engl. DataBase Management System)
 - Sadržava jednu ili više baza podataka
 - Koriste ga korisnici i aplikacije za rad s bazama podataka
 - Brine se o fizičkom smještaju podataka iz baze
 - Pruža niz usluga, između ostalog i usluge pristupa, sigurnosti i administriranja baze podataka
- Ako je DBMS entitet, onda su SQL Server, PostgreSQL, Oracle i DB/2 instance tog entiteta

DBMS i baze podataka – grafički prikaz



Modeli podataka

- Model podataka je **skup pravila** koja određuju kako mora izgledati logička struktura baze podataka
- Modeli podataka:
 - Mrežni model
 - Hijerarhijski model
 - **Relacijski model**
 - Objektni model
 - Prostorno-vremenski model

Prostorno-vremenski model

- Sadržavaju prostorne, vremenske ili prostorno-vremenske podatke
 - Prostorni: teritorij Austrije je <niz_GPS_koordinata>
 - Vremenski: Mirko je od 1.7.2002. do 1.7.2003. radio u Nabavi
 - Prostorno-vremenski: Zrakoplov LH890 je 26.02.2013. u 11:22 UTC bio na lokaciji 51,6002 / 11,2643
- Nude jezike za izražavanje upita:
 - Prostorni: kroz koje države protječe Dunav?
 - Vremenski: gdje Mirko trenutno radi?
 - Prostorno-vremenski: koji zrakoplovi će u idućih 5 minuta biti međusobno udaljeni manje od 2 km?

Objektni model

- Podaci u obliku objekata kao u objektno orijentiranim programskim jezicima
- Prednost:
 - Čuva objekte i kolekcije
- Nedostaci:
 - Nedostatak fleksibilnog jezika za postavljanje upita
 - Nepovezanost sa standardnim sustavima za izvještavanje, OLAP, povezivanje s drugim bazama, ...
 - Nedostatak formalnog matematičkog temelja
- Primjeri: db4o, ORION, DTS/S1, Perst, ...

Relacijski model

- **Najčešće korišteni model današnjice**
- Najsporiji model, ali najfleksibilniji
- Zasnovan na matematičkom pojmu relacije (tablice)
 - Podaci se prikazuju tablicama (tablični model)
 - Primjer: tablica Grad

| IDGrad | Naziv | BrojStanovnika | ZupanijaID |
|--------|---------|----------------|------------|
| 1 | Zagreb | 792875 | 1 |
| 2 | Split | 178192 | 4 |
| 3 | Rijeka | 128735 | 3 |
| 4 | Osijek | 107784 | 6 |
| 5 | Kaštela | 38474 | 4 |

Sustavi za upravljanje relacijskim bazama podataka

- Relacijski DBMS-ovi se kraće zovu RDBMS-ovi (engl. Relational DataBase Management Systems)
- Među poznatijim RDBMS-ovima su:
 - Oracle – od 1979. Oracle v2
 - Microsoft SQL Server
 - od 1989., preuzet sustav od Sybasea na OS/2
 - od 1993., na Windows NT, SQL Server 4.21
 - IBM DB2 – od 1987.
 - PostgreSQL – University of California, Berkeley, od 1989.
 - MySQL – Švedska, MySQL AB, od 1995.

Jezici za rad s bazama podataka

- Služe za komunikaciju korisnika i DMBS-a
- Govore DBMS-u što treba napraviti s bazom podataka i podacima u njoj
- Kod relacijskih baza se koristi **Structured Query Language (SQL)**
- Prvotno **Structured English Query Language (SEQUEL)** - promijenjen naziv radi *Hawker Siddeley airlines (UK)*
- Postoji puno verzija SQL jezika:
 - PL/SQL (Oracle)
 - T-SQL (Microsoft/Sybase)
 - SQL/PSM (MySQL) ...

Životni ciklus baze podataka

Životni ciklus baze podataka

- Životni ciklus baze podataka započinje idejom o bazi podataka i završava prestankom korištenja baze podataka
- Postoji puno podjela životnog ciklusa, jedna od mogućih je:
 - Analiza potreba
 - Modeliranje podataka
 - Implementacija
 - Testiranje
 - Održavanje
- Često je životni ciklus baze podataka vezan uz životni ciklus jedne ili više aplikacija

Analiza potreba i modeliranje podataka

- **Analiza potreba**

- Analiza poslovnih procesa
- Prikupljanje korisničkih zahtjeva
- Prepoznavanje bitnih entiteta, njihovih atributa i veza među njima
- Prepoznavanje operacija nad entitetima

- **Modeliranje podataka**

- Definira se relacijska shema
- Definiraju se tipovi podataka
- Shema se normalizira

Implementacija i testiranje

- **Implementacija**

- Uz pomoć RDBMS-a izrađuje se baza
- Definira se fizički smještaj baze
- Baza se puni inicijalnim podacima koji predstavljaju "nulto stanje" (popis država, gradova, valuta, ...)

- **Testiranje**

- Jako bitno i izazovno za izvesti
- Korisnici rade s bazom i provjeravaju zadovoljava li ona poslovne procese
- Ispravljaju se uočene pogreške nastale u prethodnim fazama

Održavanje

- **Održavanje**

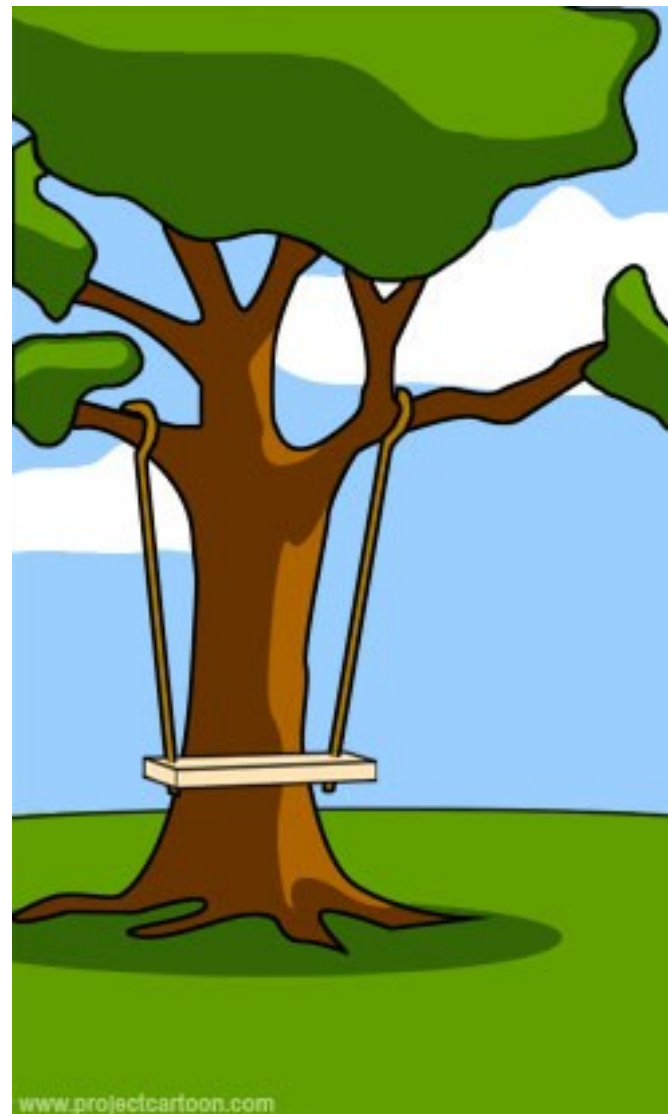
- Baza je u svakodnevnoj upotrebi
- U manjim slučajevima popravak pogrešaka koje nisu prije otkrivene
- Implementacija novih zahtjeva korisnika
- Podešavanje fizičkog smještaja
- Podešavanje brzine rada baze (indeksi)
- Izmjene modela podataka kako bi se zadovoljile performanse
- Kreiranje zaštitnih kopija
- Aktivno definiranje sigurnosnog konteksta

**Kako to u praksi često izgleda
(i što bi trebalo izbjeći)**

Kako je naručitelj objasnio što želi



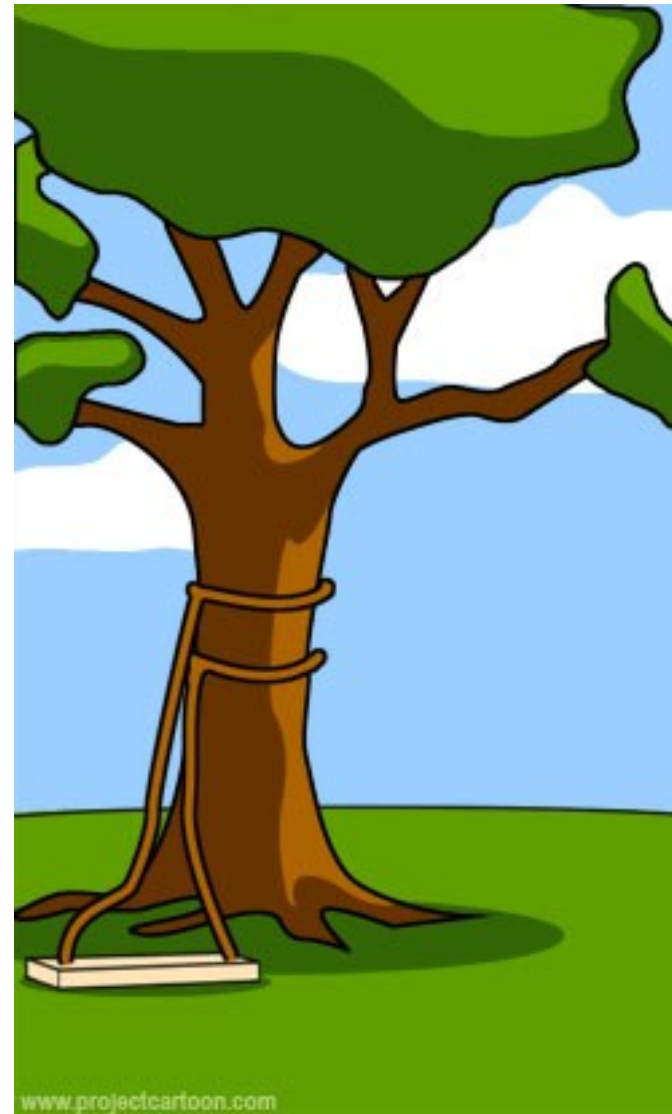
Kako ga je naš poslovni
ekspert shvatio



Kakvo je rješenje dizajnirao
glavni programer



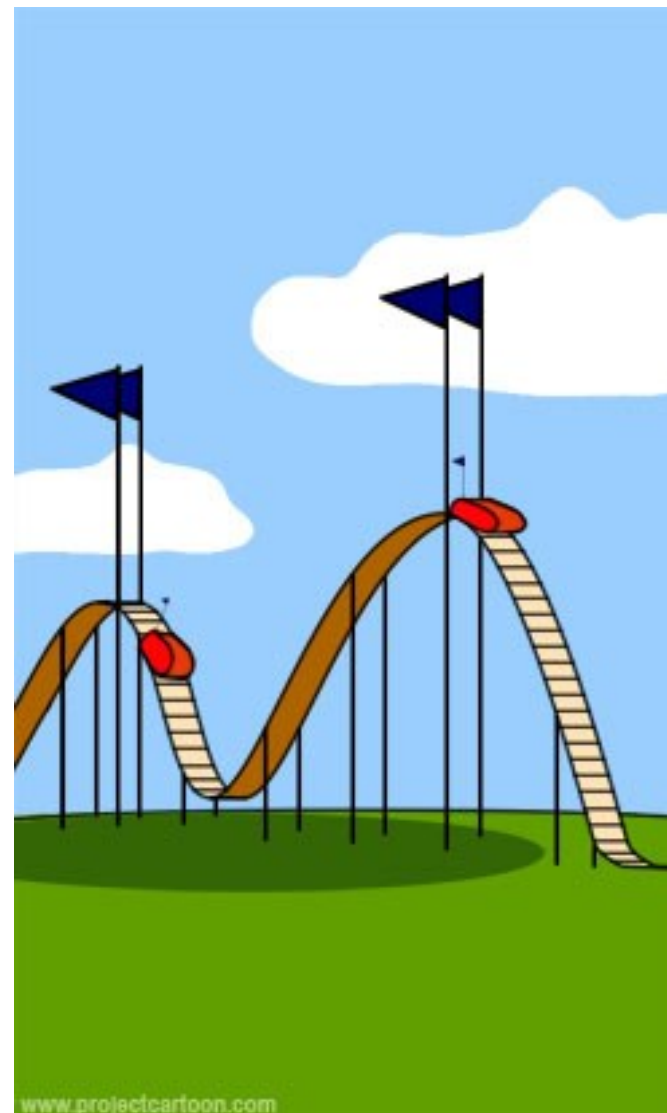
Kako ga je programer izradio



Kako ga je prodaja izreklamirala



Kako je naplaćen klijentu



Što je klijentu u stvari trebalo

