

ANALIZA ZAHTEV

Iztok Savnik

Analiza zahtev

- Pregled prvega procesa programskega inženirstva
- Podatkovne baze so praviloma del informacijskega sistema
 - Ni smiselno obravnavati podatkovn bazo posebaj
- Kasneje bomo videli nekatere specifične aspekte PB

Analiza zahtev

- Ena od težjih nalog programskega inženirja (analitik)
- Nočna mora analitika
 - Stranka pride v pisarno ga pogleda v oči in iskreno razloži:
 - »Vem, da mislite, da razumete kaj sem povedal, kar ne razumete je to, da kar sem povedal nisem zares mislil.«
 - To se zgodi po začetnih fazah projekta, ko je že bila postaljena časovnica.

Osnovna vprašanja

- Kaj delamo?
- Kdo dela?
- Zakaj je to pomembno?
- Kakšni so koraki?
- Kaj je rezultat?
- Kako zagotoviti korektnost?

Koraki analize zahtev

- Zametek
- Elicitacija
- Elaboracija
- Pogajanje
- Specifikacija
- Validacija
- Upravljanje zahtev

Zametek

- Idealno:
 - Programski inženirji in stranke definirajo skupen team
 - Analiza zahtev predstavlja komunikacijo teama
- Realnost:
 - Analitiki so v oddaljenem kraju
 - Stranke ne vedo kaj narediti
 - Imajo konfliktne ideje
 - Imajo omejeno tehnično znanje
 - Imajo omejen čas

Zametek

- Cilj je vzpostavitev temelje razumevanja programskih zahtev
- **Identifikacija udeležencev:**
 - Vsak, ki ima direktne in posredne koristi od razvitega sistema
 - Poslovni in produkcijski managerji, stranke, končni uporabniki, konsultanti, produkcijski inženirji, ...
- Seznam ljudi, ki bodo prispevali pri elicitaniji

Zametek

- **Prepoznavanje več pogledov:**
 - Potrebe bodo zbrane iz več različnih virov
 - Primeri:
 - Marketinška skupina želi imeti funkcije, ki bi pritegnila potencialen trg
 - Poslovna skupina bi želela izdelati vse ob danih omejitvah
 - Programski inženirji želijo uporabiti nove tehnologije, ki bi pritegnile trg
 - Podporni inženirji bi želeli čim bolje obvladati sistem
 - Vsi udeleženci bodo prispevali k zbranim potrebam: konflikti, smeri, ...

Zametek

- Izgradnja sodelovanja:
 - Dobimo lahko pet različnih pogledov o potrebah
 - Uporabniki morajo sodelovati pri definiciji potreb, če hočemo uspešno narediti projekt (njih to ne zanima ;)
 - Identificirati je potrebno skupne poglede in konflikte
 - Določitev »prioritetnih točk«
 - Potrebe velikokrat določi »lokalni voditelj«

Zametek

- **Prva vprašanja**
 - Identifikacija udeleženca, ki bo imel interes pri razvoju sistema:
 - Kdo je za naročilom tega dela? Kdo bo uporabil rešitev? Kakšen bo ekonomski profit uspešne rešitve? Je še kakšen vir rešitve potreben?
 - Pridobivanje boljšega razumevanja problema:
 - Kakšen bi definirali »dober« rezultat projekta? Katere probleme bo rezultat rešil? Pokaži poslovno okolje kjer bo rešitev koristno uporabljena. Bosta izdelava rešitve ali omejitve pri izdelavi vplivale na rešitev?

Zametek

- **Prva vprašanja**
 - Vprašanja glede učinkovitosti komunikacije; meta-vprašanja:
 - Si pravi naslov za podobna vprašanja? So tvoji odgovori uradni? Preveč sprašujem? Lahko kdo drug poda odgovore? Bi moral spraševati kaj drugega?

Elicitacija

- **Odkrivanje, zbiranje, opisovanje potreb.**
 - Kombinacija reševanja problemov, elaboracije, pogajanja in specifikacije.
 - Skupen, teamski pristop k reševanju problemov; udeleženci sodelujejo pri reševanju problemov

Elicitacija

- Skupno zbiranje zahtev
 - Sestankov se udeležujejo vsi udeleženci.
 - Pravila za pripravo in udeležbo na sestankih so postavljena.
 - Razpored je zadosti splošen, da pokrije vse teme, in zadosti odprt za pretok idej
 - Povezovalec (iz katerekoli skupine) vodi sestanke
 - Uporablja se mehanizem za definicijo odločitev (tabla, listki, el.table, etc.)

Elicitacija

- Udeleženci začetno napišejo 1-2 lista **zahtev projekta** – vsak svoje
- Določi se sestanek, povezovalec, objavijo se zahteve projekta
- Zahteve projekta pridobljene od posameznikov bodo redundantne, nepopolne, protislovne, ...
- Udeležence se prosi za opis objektov
 - Objekti v okolju; objekti, ki so izhod; in objekti, ki predstavljajo notranje stanje

Elicitacija

- Udeležence se prosi za seznam omejitev, ki ni popoln (cena, velikost, prostor, ...)
- **Primer: SafeHome**
- Identificirani objekti:
 - kontrolni pult; senzorji za temperaturo, dim, gibanje, ...; alarm; dogodek; prikazovalnik; tel klic; ...
- Seznam ident servisov:
 - konfiguriranje; postavljanje alarma; monitoring senzorjev; klic tel številke; programiranje kontrol pulta; branje prikazovalnika, ...

Elicitacija

- **Seznam objektov se objavi**
 - Pripne na zidove prostora kjer se vodijo sestanki, vstavi v elektronsko tablo, na interne internetne strani firma, itd.
 - Vsak zapis seznama naj se da manipulirati
 - Sčasoma se prilagodijo seznamami, razširijo opisi objektov, itd. tako pridemo do skupne rešitve
 - **Konsensus** glede seznamov objektov, servisov, omejitev ter performans
 - Naredijo se **mini-specifikacije** vseh stvari na seznamih

Elicitacija

- SafeHome
- Mini-specifikacije kontrolne table

Kontrolna tabla je zaslon, ki je z wireless povezavo povezan na senzorje in PC. Kontrolna tabla prikazuje stanje senzorjev na LCD zaslonu in omogoča vpogled v podrobnosti meritev. Interakcija z uporabnikom poteka preko tipkovnice, ki ima 12 tipk.

- Mini-specifikacije se predstavijo vsem udeležencem
 - Seznam problemov (issue-list), ki se vzdržuje
 - Dodaja se lahko nove objekte, servise, ...

Elicitacija

- **Postavitev funkcije kvalitete**
 - Mehanizmi za kontrolo kvalitete
 - Kontrola kvalitete »poudari razumevanje vrednot stranke in predstavi vrednote preko prog. inženirstva«
 - Identificiramo tri vrste potreb (zahtev) uporabnika:
 - 1) Normalne zahteve
 - 2) Pričakovane zahteve
 - 3) Fancy zahteve

Elicitacija

- **Scenariji uporabe**
 - Pri zbiranju potreb se kreira pogled na funkcije in značilnosti sistema
 - Težko se je ukvarjati z bolj tehničnimi aspekti brez da razumemo kako se bodo funkcije uporabljale
 - **Primeri uporabe** lahko opišejo uporabo posameznih funkcij
 - Primeri uporabe pri analizi potreb bodo predstavljeni v naslednjem poglavju

Elicitacija

- **Elicitacija delovnih produktov**
 - Izjava o potrebi in uporabnosti
 - Namembnost produkta
 - Seznam udeležencev, ki so sodelovali pri elictaciji
 - Opis tehničnega okolja sistema
 - Seznam zahtev urejenih po funkciji
 - Omejitve domene nad funkcijami
 - Množica scenarijev uporabe, ki prikažejo delovanje in uporabo produkta med različnimi operacijami
 - Prototipi razviti za boljšo definicijo zahtev

Razvoj primerov uporabe

- **Primer uporabe** pove zgodbo o tem kako uporabnik v različnih vlogah kumunicira s sistemom
- Definira se množica akterjev
 - Karkoli kominicira s sistemom
 - Isti akterji so lahko v različnih vlogah
 - V prvi iteraciji niso definirani vsi akterji
 - Primarni in sekundarni akterji
- Potem ko so akterji definirani se razvijejo primeri uporabe

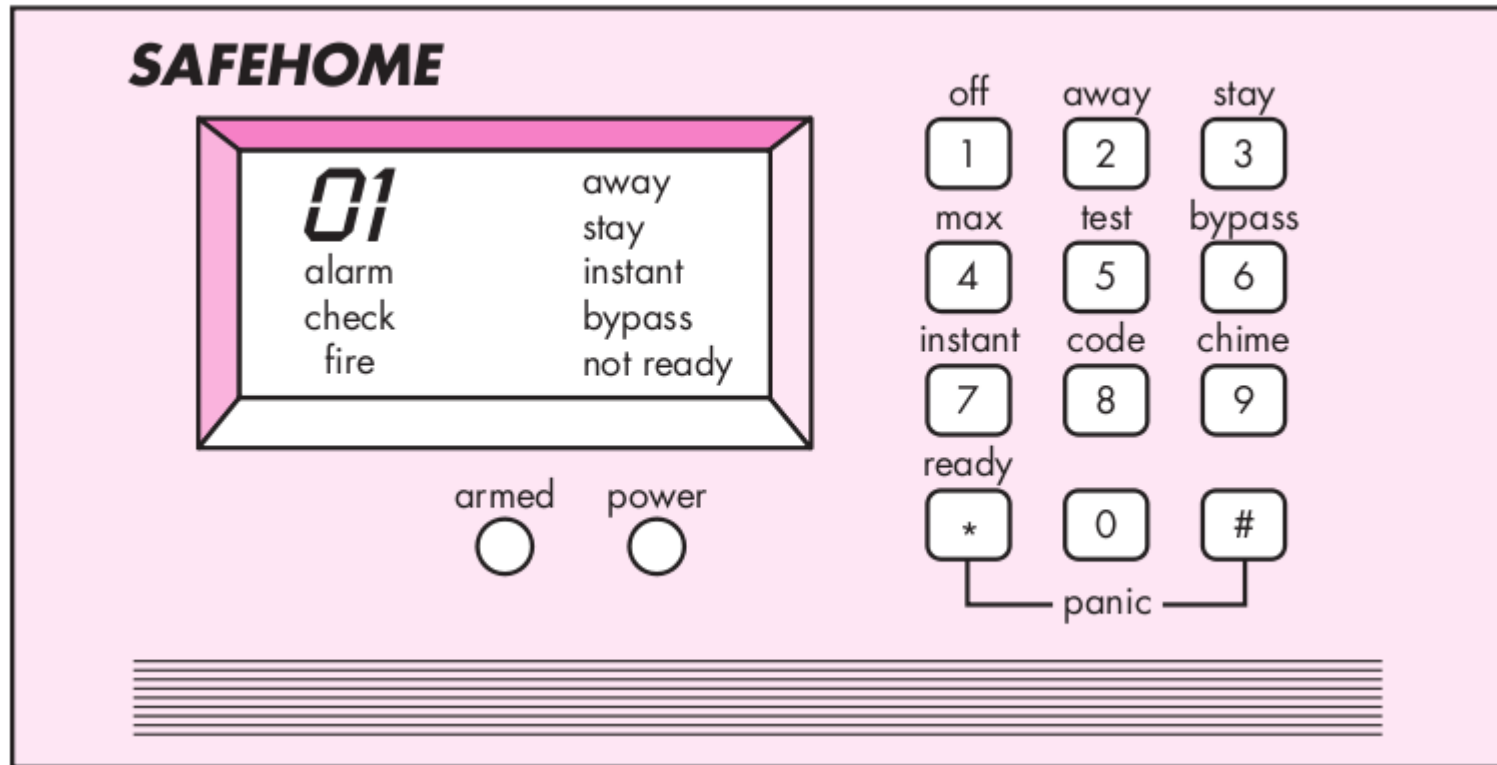
Razvoj primerov uporabe

- **Primer uporabe da odgovore na naslednja vprašanja**
 - Kdo je primarni akter, sekundarni akterji? Kaj so cilji akterjev? Kakšni so predpogoji za zgodbo? Osnovne funkcije, ki jih opravlja akter? Kakšne izjeme so opisane? Kakšne so variacije interakcije akterja? Systemske informacije, ki jih akter vnese, prebere ali spremeni? ...

Primer: SafeHome

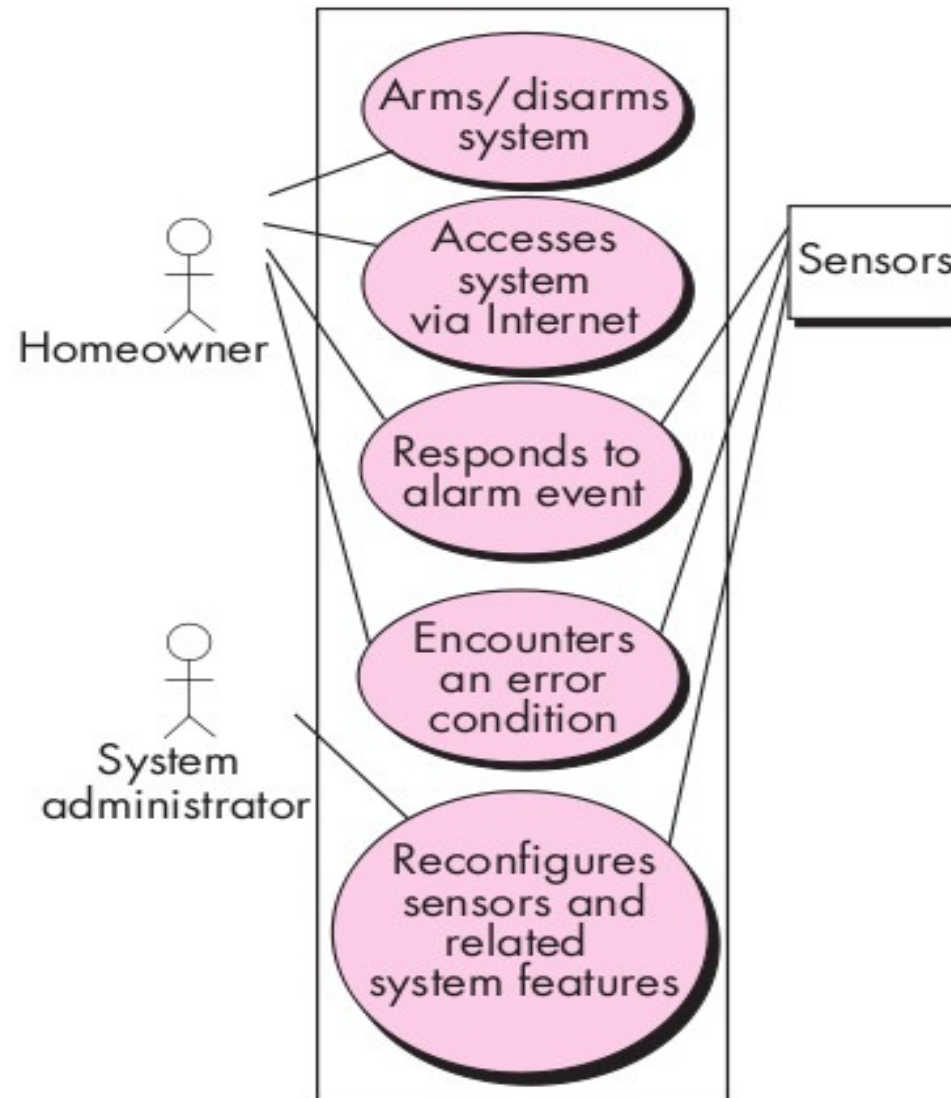
- Akterji:
 - Lastnik, upravitelj-nastavitev, senzorji, monitorski-sistem
- Akter **Lastnik**:
 - Lastnik kumunicira s sistemom preko PC-ja ali panela.
 - Vstavi geslo, gleda status področja ali senzorja, vključi alarm, aktivira/deaktivira sistem.

Primer: SafeHome



Primer: SafeHome

Primeri
uporabe
Lastnika



Bolj podroben opis primera uporabe

Use case:	<i>InitiateMonitoring</i>
Primary actor:	Homeowner.
Goal in context:	To set the system to monitor sensors when the homeowner leaves the house or remains inside.
Preconditions:	System has been programmed for a password and to recognize various sensors.
Trigger:	The homeowner decides to “set” the system, i.e., to turn on the alarm functions.

Scenario:

1. Homeowner: observes control panel
2. Homeowner: enters password
3. Homeowner: selects “stay” or “away”
4. Homeowner: observes red alarm light to indicate that *SafeHome* has been armed

Exceptions:

1. Control panel is *not ready*: homeowner checks all sensors to determine which are open; closes them.
2. Password is incorrect (control panel beeps once): homeowner reenters correct password.
3. Password not recognized: monitoring and response subsystem must be contacted to reprogram password.
4. *Stay* is selected: control panel beeps twice and a *stay* light is lit; perimeter sensors are activated.
5. *Away* is selected: control panel beeps three times and an *away* light is lit; all sensors are activated.

Priority: Essential, must be implemented

When available: First increment

Frequency of use: Many times per day

Channel to actor: Via control panel interface

Secondary actors: Support technician, sensors

Channels to secondary actors:

Support technician: phone line

Sensors: hardwired and radio frequency interfaces

Open issues:

1. Should there be a way to activate the system without the use of a password or with an abbreviated password?
2. Should the control panel display additional text messages?
3. How much time does the homeowner have to enter the password from the time the first key is pressed?
4. Is there a way to deactivate the system before it actually activates?

Izgradnja modela zahtev

- Namen modela analize je predstavitev informacijskih in funkcijskih zahtev ter zahtev po obnašanju sistema.
- Načrtovalci se spoznajo s sistemom; udeleženci natančno vidijo kakšne zahteve imajo.
- Sčasoma se nekatere zahteve ustalijo, za druge pa se vidi, da niso usklajene oz. domišljene.

Elementi modela potreb

- Imamo različne šole
- Nekateri predlagajo, da se en model uporablja za predstavitev modela potreb
- Drugi uporabljajo več različnih modelov
 - Več različnih pogledov na komponente sistema
 - Več možnosti za odkrivanje napak in izpuščenih delov in aspektov

Elementi modela potreb

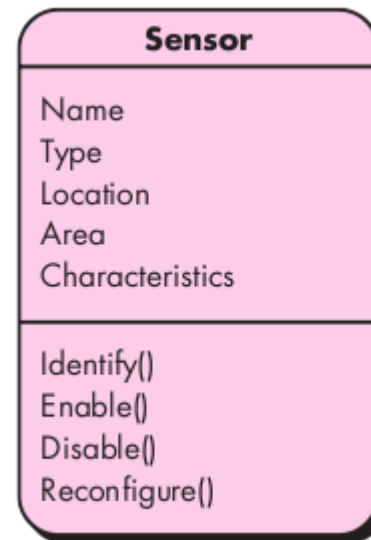
- Gradniki za predstavitev potreb
 - Scenariji
 - Razredi
 - Opis obnašanja
 - Opis tokov podatkov

Scenariji

- Primeri uporabe preidejo v stabilne vzorce
- Primeri uporabe so velikokrat prvi jezik specifikacije
- Ostali modeli se naslanjajo na primere uporabe

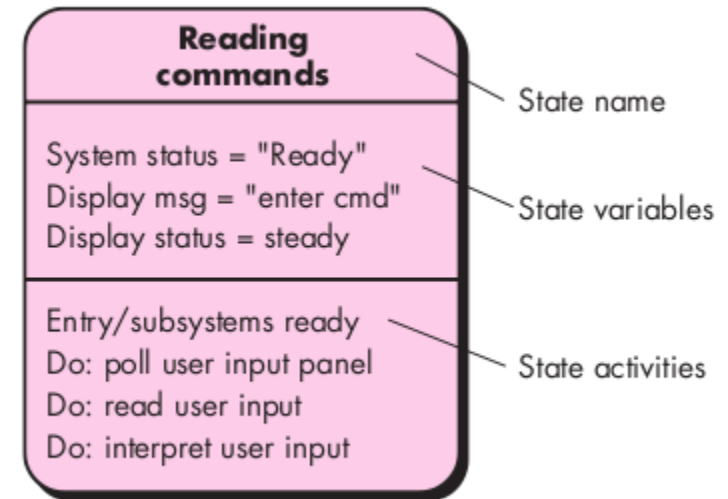
Razredi

- Vsak scenarij implicira množico objektov
- Objekte se klasificira in opiše z lastnostmi
- Zveze med razredi



Opis obnašanja

- Obnašanje objektov ima lahko zelo velik vpliv na design in implementacijo objekta
 - Zato je potreben opis obnašanja
- Diagramske tehnike
 - Diagrami razredov
 - Diagram stanj
 - Diagrami sodelovanja
 - Diagram sekvenc



Tokovi podatkov

- Data flow diagrami (DFD)
- Diagrami aktivnosti

Načrtovalski vzorci

- Informacijski sistemi vsebujejo dele, ki se ponavljajo
- Identifikacija komponente, ki predstavlja načrtovalski vzorec
 - Učinkovita rešitev in implementacija
 - Ponovna uporaba kode
- Firme zbirajo načrtovalske vzorce v knjižnicah

Pogajanje glede zahtev

- Idealno imamo po fazah zametka, elicitanije in elaboracije model zahtev za razvoj sistema
- V praksi se je potrebno pogajati z enim ali večimi udeleženci
 - Udeleženci balansirajo funkcionalnost, performanse, ter druge lastnosti sistema, nasproti cene in časa izdelave.

Pogajanje glede zahtev

- Najboljši pogajalci poskušajo dobiti win-win rezultat
 - Obe strani naj zmagata
- Pogajalske aktivnosti:
 - Poišči vse udeležence, ki držijo pod-sistem ali sistem
 - Določi zmagovalne pogoje teh udeležencev
 - Pogajanja z udeleženci, da bi dosegli win-win rezultat

Umetnost pogajanja

- Pogajanje ni tekmovanje.
 - Obe strani imata profit ali zmago. Nujen kompromis.
- Določi strategijo.
 - Kaj želiš doseči? Kaj želijo doseči ostali? Kako to doseči?
- Dobro poslušaj.
 - Večino rešitev povejo sami udeleženci. Dobimo znanje za pogajanje.

Umetnost pogajanja

- Fokusiraj na interese nasprotnega tabora
 - Ne postavljaljaj težke pogoje
- Ne jemlji stvari osebno. Fokusiraj na problem.
- Bodi kreativen.
 - Razmišljaj fleksibilno. Prilagodi se.
- Bodi pripravljen sprejeti rešitev.
 - Ko pride do rešitve sprejmi, da se lahko premakne naprej.

Validacija zahtev

- Model potreb se preveri glede konsistentnost, majkajoče dele in dvoumnost
- Pregled modela zahtev:
 - Je vsaka zahteva konsistentna z globalnim ciljem projekta?
 - So bile vse zahteve obravnavane na primernem nivoju abstrakcije? Preveč tehnične?
 - So zahteve res potrebne ali predstavljajo dodatne lastnosti, ki niso nujne za sistem?

Validacija zahtev

- Pregled modela zahtev:
 - So omejitve vezane na konkreten del sistema in so nedvoumne?
 - Je izvor zahteve predstavljen z omejitvijo?
 - So zahteve med sabo v konfliktu?
 - Je zahteve možno realizirati v danem okolju aplikacije?
 - Se da zahteva testirati?
 - Ali zahteve odsevajo informacije, funkcije in obnašanje sistema, ki ga gradimo?

Validacija zahtev

- Pregled modela zahtev:
 - So bile zahteve porazdeljene tako, da porazdelitev prispeva k izgradnji sistema?
 - So bili uporabljeni vzorci zahtev za poenostavitev modela zahtev? So bili vzorci pravilno validirani? So vzorci konsistentni z zahtevami uporabnika?

Analiza zahtev za podatkovne baze

- Poglejmo še analizo zahtev iz vidika podatkovne baze
- Zbiramo podrobnosti glede podatkovnih baz

Analiza zahtev

- Analiza zahtev je pogosto najbolj zahtevna faza
- Načrtovalec izve zakaj bo baza uporabljena in kakšne podatke bo vsebovala

Analiza zahtev

- 1) Določi zahteve po podatkih v obliki osnovnih podatkovnih elementov
- 2) Opis podatkov o osnovnih podatkovnih elementih in razmerji med njimi
- 3) Določitev tipov transakcij, ki bodo izvajane na dani podatkovni bazi in interakcije med transakcijo in podatkovnimi elementi
- 4) Definicija performančnih, integritetnih, varnostnih ali administrativnih omejitev

Analiza zahtev

- 5) Specifikacija omejitev kot so posebne tehnologije, strojna in programska oprema, programski jeziki, pravila in standardi.
- 6) Podrobna dokumentacija zgornjega v analizi zahtev
- 7) Podatkovni elementi so del podatkovnega slovarja

Analiza zahtev

Identificiraj realnost zadosti podrobno za definicijo komponent podatkovne baze.

Zberi dve vrsti podatkov:

- 1) naravni podatki modeliranega okolja (vhod podatkovne baze) in
- 2) obdelani podatki (izhod iz podatkovne baze).

Naravne zahteve po podatkih

- Kaj gre v podatkovno bazo?

1. Organizacijski cilji

- Prodaj več avtomobilov letos
- Pomik v rekreacijski trg avtomobilov

2. Cilji informacijskega sistema

- Sledenje konkurenci glede produktov in cen
- Izboljšava kvalitete in časovnih zahtev po podatkih v vodstvu glede produkcije, zakasnitev pri izvajanju, itd.
- Sledenje vitalnim virom potrebnih za produkcijo in marketing

Naravne zahteve po podatkih

3. Organizacijska struktura / načrt

4. Administrativna in operacijska politika

- Letno preverjanje zaposlenih
- Tedensko poročilo o napredku
- Mesečen pregled inventarja
- Pregled potnih nalogov

5. Podatkovni elementi, razmerja, omejitve, računsko okolje

Računske zahteve

- Kaj pride iz podatkovne baze?
- 1. Obstoječe „fizične“ aplikacije se informatizirajo
- 2. Planirane nove aplikacije
 - Ocenitev uporabe podatkov za aplikacije
 - Morale bi uporabljati podmnožico že identificiranih podatkov (lahko tudi ni zaradi nepredvidenih aplikacij)
 - Problem – veliko prihodnjih aplikacij ni znanih

Podatkovni in procesni zapisi slovarja

- Rezultati analize zahtev v življenjskem ciklu načrtovanja podatkovnih baz.
- Podani bodo primeri zapisov, ki predstavljajo rezultate analize zahtev.

Opis entitete (podatkovni slovar)

Name	customer
Reference-no	4201
Cardinality	10,000
Growth rate	100 per month
Synonyms	user, buyer
Role (or description)	someone who purchases or rents a product made by the company.
Security level	0 (customer list is public)
Subtypes	adults, minors
Key attribute(s)	cust-no
Non-key attribute(s)	cust-name, addr, phone, payment-status
Relationship to other entities	salesperson, order, product
Used in which applications	billing, advertising

Opis atributov (podatkovni slovar)

Name	cust-no
Reference-no	4202
Range of legal values	1 to 999,999
Synonyms	cno, customer-number
Data type	integer
Description	customer id number set by the company.
Key or nonkey	key
Source of data	table of allowable id numbers
Used in applications	billing
Attribute trigger	/*describes actions that occur when a data element is queried or updated*/

Opis razmerij (podatkovni slovar)

Name	purchase
Reference-no	511037
Degree	binary
Entities and connectivity	customer(0,n), product(1,n)
Synonyms	buy
Attributes (of the relationship)	quantity, order-no
Assertions	quantity, order-no
Process (application) description	a customer must have purchased at least one product, but some products may not have been purchased as yet by any customers.

Izračun plač (procesni zapis)

Name	payroll
Reference-no	163
Frequency	Frequency
Priority	Priority
Deadline	noon Fridays
Data elements used	emp-name, emp-salary
Entities used	Data volume (how many entities)
Employee	implicit from entity cardinality

Administracija naročil (procesni zapis)

Function: Take customer orders and either fill them or make adjustments.

Frequency: daily

Task Def	Volume	Data Elements
1. Create order	2000	A, B, E, H
2. Validate order	2000	A, B, G, H, J
3. Fill out error form	25	A, C
4. Reserve item/price	6000	A, D, H
5. Request alternate items	75	A, E, I, K, M
6. Enter unit price	5925	A, F, J, N

Intervjuji na različnih nivojih

Vrh upravljanja – poslovne definicije, plani/stanje, plan za prihodnost

Sredina upravljanja – funkcije v področjih dela, tehnična področja, delovni nazivi, delovne funkcije

Zaposleni – posamezna dela, potrebni podatki, izhodni podatki

Specifični končni uporabniki SUPB – aplikacije in potrebni podatki

Osnovna pravila za intervju

1. Razišči poslovni del najprej
2. Dogovori se s stranko glede oblike dokumentacije (ERD, DFD, itd.)
3. Definiraj človeška opravila in poznane računalniške aplikacije
4. Načrtaj in preveri diagrame toka podatkov in ER diagrame
5. Primerjaj aplikacije s podatki (to pomaga programerjem)

Literatura

- Toby Teorey, Sam Lightstone, Tom Nadeau, Database Modeling and Design: Logical Design, Elsevier, 2006.
- Roger S.Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill, 2010.