

4 Prikupljanje korisničkih zahteva

1

Korisnički zahtevi:

- Prvi korak u razvoju novog sistema: **identifikovati šta treba da radi novi sistem.**
- Suštinski je značajno saznati šta korisnici zahtevaju od novog sistema, a postojeći sistem nije mogao da uradi.

2

Postojeći sistem #1:

Prvo treba proučiti postojeći sistem.

- 1) Neke od funkcija postojećeg sistema biće zahtevane u novom sistemu.
- 2) Neke od podataka treba preneti iz postojećeg u novi sistem.
- 3) Tehnička dokumentacija postojećeg računarskog sistema može da obezbedi algoritme procesiranja koji će biti potrebni novom sistemu.
- 4) Postojeći sistem može imati defekte koje treba izbeći u novom sistemu.

3

Postojeći sistem #2:

- 5) Proučavanje postojećeg sistema pomoći će razumevanju organizacije uopšte.
- 6) Delovi postojećeg sistema mogu se zadržati. Informacioni sistemi danas često predstavljaju nadogradnju i projektovanje interfejsa.
- 7) Potrebno je razumeti kako korisnici obavljaju svoj posao jer će oni raditi i na novom sistemu.
- 8) Potrebno je prikupiti informacije koje će poslužiti za merenje kvaliteta novog sistema.

4

Novi zahtevi:

U skladu sa dinamikom razvoja okoline, potrebno je stalno poboljšavati informacione sisteme unutar organizacija.

Prikupljene informacije pripadaju jednoj od tri kategorije

- funkcionalni zahtevi
- nefunkcionalni zahtevi
- upotrebnii zahtevi

5

Funkcionalni zahtevi:

Funkcionalni zahtevi opisuju šta sistem radi ili se očekuje da radi, često zvano *funkcionalnost*. Ti zahtevi uključuju:

- Opis obrade koju sistem mora da obavlja.
- Detalje ulaza u sistem sa papira i dokumenata, od interakcije između ljudi (kao što su telefonski pozivi) i od drugih sistema.
- Detalji izlaza koji se očekuju od sistema u obliku štampanih dokumenata i izveštaja, na displeju i prenosa drugim sistemima.
- Detalji o podacima koji se moraju čuvati u sistemu.

6

Nefunkcionalni zahtevi:

Nefunkcionalni zahtevi su oni koji opisuju aspekte sistema koji se razmatraju pri vrednovanju koliko dobro su zadovoljeni funkcionalni zahtevi. Ovo uključuje sledeće:

- Kriterijumi kao željeno vreme odgovora pri ažuriranju (update) podataka u sistemu ili izvlačenju podataka iz sistema.
- Predviđena količina podataka, i što se tiče protoka i što se tiče memorisanja.
- Razmatranja sigurnosti.

7

Upotrebnii zahtevi #1:

Upotrebnii zahtevi su oni koji omogućavaju da se osigura da postoji dobro slaganje između razvijenog sistema, s jedne strane, i korisnika sistema i zadataka koje će obavljati koristeći sistem, sa druge strane.

Internacionalna organizacija za standarde (ISO) definisala je *upotrebljivost* kao 'stepen do koga specifični korisnici mogu da postignu specifične ciljeve unutar posebne okoline; efektivno, efikasno, udobno i na prihvatljiv način.

8

Upotrební zahtevi #2:

Da bi se u sistem ugrađivala upotrebljivost od početka, potrebno je prikupiti sledeće tipove informacija:

- Karakteristike budućih korisnika sistema.
- Zadatke koje korisnici podrazumevaju, uključujući ciljeve koje pokušavaju da postignu.
- Situacione faktore koji opisuju situacije koje se mogu pojaviti u toku upotrebe sistema.
- Kriteriji prihvatljivosti po kojima će korisnik prosuđivati isporučen sistem.

9

Tehnike prikupljanja činjenica:

Postoje pet glavnih tehnika korišćenih za istraživanje zahteva:

- 1) čitanje pozadine,
- 2) intervjuisanje,
- 3) posmatranje,
- 4) uzorkovanje dokumenata i
- 5) ankete (upitnici).

10

1) Čitanje pozadine:

Dokumentacija koja predstavlja pogodan izvor informacija uključuje sledeće:

- izveštaji kompanije,
- organizacione karte,
- uputstva za smernice,
- opisi poslova,
- izveštaji i
- dokumentacija postojećeg sistema.

11

Čitanje pozadine – prednosti i nedostaci:

- + Razumevanje organizacije pre susreta sa zaposlenima.
- + Priprema za druge zadatke, npr. poslovnih ciljeva u firmi.
- + Dokumentacija postojećeg sistema može da predstavlja bar deo definicija zahteva za novi sistem.
- Pisani dokumenti obično se ne slažu sa stvarnošću: mogu biti zastareli ili mogu da opisuju zvanična pravila koja se razlikuju od pravila u praksi.

12

2) Intervjuisanje:

Najšire korišćena tehnika pribavljanja činjenica.

Intervju o analizi sistema je uređeni sastanak analitičara i intervjuisanog koji je obično zaposleni u organizaciji koja se istražuje.

Intervju može biti sa fiksnim unapred pripremljenim pitanjima ili može biti slobodan razgovor koji dopušta iznošenje interesantnih činjenica.

Intervju se može koristiti za prikupljanje informacija od strane menadžmenta o ciljevima organizacije i novog informacionog sistema, od zaposlenih o postojećim poslovima i njihovim potrebama, kao i od mogućih korisnika sistema.

13

Intervjuisanje – prednosti i nedostaci:

- + Lični kontakt dopušta analitičaru da dobro shvati šta korisnik kaže. Zato se ovim postupkom dobija visokokvalitetna informacija
- + Analitičar može da proba razne metode za ostvarenje posla osobe.
- + Ako intervjuisani nema šta da kaže, intervju se prekida.
- Intervjui zahtevaju mnogo vremena i mogu biti najskuplji postupak prikupljanja informacija.
- Posle intervjua potrebna je obrada materijala.
- Intervju može biti predmet polarizacije ako intervjuisani ima specifično mišljenje.
- Ako različiti intervjui daju suprotne informacije, može biti teško da se kasnije razreši.

14

3) Posmatranje:

Posmatranjem se prati rad u prirodnim okolnostima i mogu se uočiti i neke izuzetne okolnosti koje se mogu propustiti u intervjuu zbog koncentrisanosti na normalne okolnosti.

Mogu se pri tome uočiti prekidi koje treba tretirati u novom sistemu.

Ljudi nisu dobri u kvantitativnoj proceni podataka (kao što je trajanje nekih zadataka), a posmatranjem i merenjem mogu se dobiti kompletni podaci.

15

Posmatranje – prednosti:

- + Posmatranjem ljudi na poslu dobija se informacija iz prve ruke o načinu rada postojećeg sistema.
- + Podaci se prikupljaju u stvarnom vremenu i mogu biti veoma vredni ako im je posvećena pažnja.
- + Posmatranje se može koristiti za verifikovanje informacija iz raznih izvora ili za posmatranje izuzetaka.
- + Mogu se prikupiti osnovni podaci o svojstvima postojećeg sistema i korisnicima.

16

Posmatranje – nedostaci:

- Većina ljudi ne voli da bude posmatrana i pri tome se ponaša drugačije od uobičajenog.
- Da bi posmatranje imalo efekta, posmatrač mora da bude uvežban i vešt.
- Analitičar može imati logističke probleme ako npr. osoblje koje treba posmatrati putuje na dugim rastojanjima da bi obavljalo posao.
- Mogu postojati etički problemi ako posmatrana osoba radi sa osetljivim privatnim ili ličnim podacima, ili sa javnošću.

17

4) Uzorkovanje dokumenata:

Koristi se na dva načina:

- Analitičar prikuplja kopije dokumenata formiranih na bazi intervjua i posmatranja. Na osnovu toga formira se kompletna informacija o radu ljudi, o ulazima i izlazima iz procesiranja.
- Analitičar može da napravi statističku analizu dokumenata da bi našao okvire podataka.

18

Uzorkovanje dokumenata - prednosti i nedostaci:

- + Može se koristiti za prikupljanje kvantitativnih podataka, kao npr. prosečni broj linija na račun.
- + Može se koristiti za nalaženje učestanosti grešaka u papirnim dokumentima.
- Ako se sistem menja dramatično, postojeći dokumenti neće odražavati stanje koje će biti u budućnosti.

19

5) Upitnici:

Tehnika prikupljanja činjenica koja se sastoji od odgovora na seriju pisanih pitanja.

20

Upitnici – prednosti i nedostaci:

- + Ekonomičan način prikupljanja podataka od velikog broja ljudi.
- + Ako je upitnik dobro napravljen, rezultati se mogu lako analizirati i pomoću računara.
- Teško je konstruisati dobre upitnike.
- Nema automatskog mehanizma za nastavak ili za dublje istraživanje, mada je moguće nastaviti intervjuom pomoću telefona ili lično ako je potrebno.
- Upitnici poslati poštom imaju malu učestanost odgovora.

21

Uključivanje korisnika u projekat #1:

Upravljanje projektom sa korisničke strane uključuje sledeće kategorije ljudi:

- stariji menadžment – sa potpunom odgovornošću za rad organizacije,
- finansijski menadžeri sa kontrolom budžeta projekta,
- menadžeri odeljenja kod korisnika i
- predstavnici korisnika.

22

Uključivanje korisnika u projekat #2:

Korisnici se mogu uključiti sa raznim ulogama u toku rada projekta, kao:

- subjekti u intervjuu da bi se ustanovili zahtevi,
- predstavnici u projektnim odborima,
- uključeni u vrednovanje prototipa,
- uključeni u testiranje,
- subjekti kurseva obuke i
- krajnji korisnici novog sistema

23

Dokumentovanje zahteva:

- Činjenice o korisničkim zahtevima i organizaciji treba odmah dokumentovati
 - u papirnoj formi ili
 - pomoću računara (koristeći projektne alate)

24

Dokumenti koji se čuvaju u biblioteci projekta:

- zapisi sa intervjua i posmatranja,
- detalji problema,
- kopije postojećih dokumenata i gde su oni korišćeni,
- detalji zahteva i
- vreme sastanaka.

25

Uređivanje dokumentacije:

- Dokumentaciju čuvati digitalno, koristeći sisteme za upravljanje dokumentacijom.
- Baze podataka povezane sa CASE alatima.
- Omogućiti praćenje od zahteva, kroz analizu i modele projekta, do implementacije i testiranja ispunjenosti zahteva.

26

Zahtevi i korisnički slučajevi #1:

- **Zahtevi** su pravila koja *određuju* ponašanje sistema.
- **Korisnički slučajevi** opisuju pojedinačno ponašanje sistema. Zato mogu da opisuju funkcionalne, a ne mogu nefunkcionalne zahteve sistema.

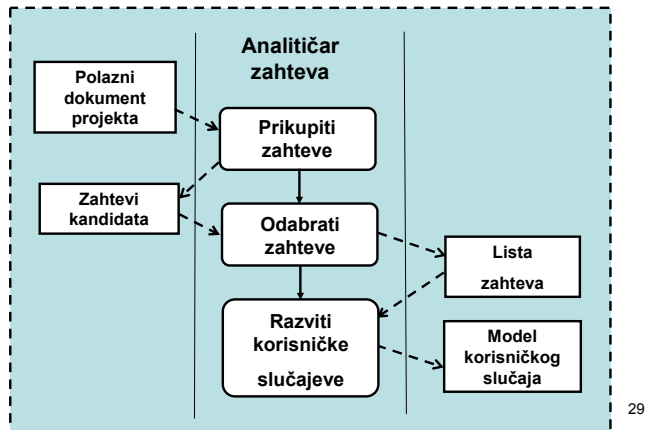
27

Zahtevi i korisnički slučajevi #2:

Dakle,
korisnički slučajevi mogu se koristiti za modelovanje funkcionalnih zahteva, ali mora se držati odvojena *lista zahteva sistema* koja sadrži sve zahteve, i funkcionalne i nefunkcionalne.

28

Dijagram aktivnosti za prikupljanje korisničkih zahteva:



29

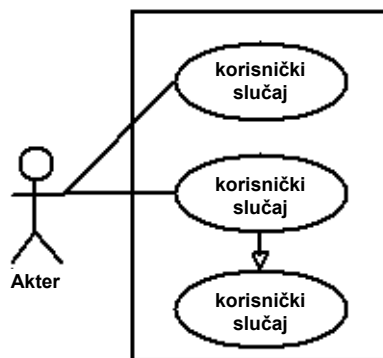
Korisnički slučajevi:

Korisnički slučajevi su opis funkcionalnosti sistema sa korisničke perspektive.

Dijagrami korisničkih slučajeva koriste se za prikazivanje funkcionalnosti koju će posedovati sistem, i da pokažu koji korisnici će komunicirati sa sistemom na neki način da bi se koristila ta funkcionalnost.

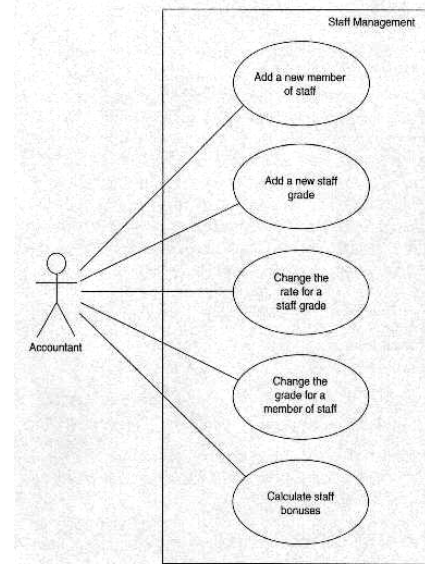
30

Primer dijagrama korisničkog slučaja:



31

Primer dijagrama korisničkog slučaja:



Svrha:

Model korisničkog slučaja je deo modela zahteva.

Model zahteva uključuje i objektni model domena problema i opis korisničkog interfejsa.

Korisnički slučajevi su podržani *specifikacijama ponašanja* (UML dijagrami ili tekstualni opisi korisničkih slučajeva).

33

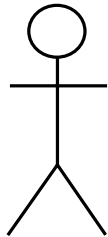
Notacija:

Dijagrami korisničkih slučajeva pokazuju tri aspekta sistema:

- akteri (učesnici),
- korisnički slučajevi i
- granica sistema ili podsistema.

34

Akteri:



Akteri (učesnici) predstavljaju uloge koje ljudi, drugi sistemi ili uređaji preuzimaju kada komuniciraju sa posebnim korisničkim slučajem u sistemu.

35

Akteri:

- Akteri nešto **predaju** sistemu ili nešto **primaju** od sistema.
- Akteri komuniciraju sa sistemom slanjem poruka sistemu i primanjem poruka od sistema kako ostvaruju korisničke slučajeve.
- Akteri modeluju bilo šta što zahteva interakciju sa sistemom da bi izmenjali informacije – čoveka-korisnika, računarski sistem, itd.
- Fizički korisnik može da igra ulogu jednog od nekoliko aktera jer ima interakciju sa sistemom, a nekoliko individualnih korisnika mogu da deluju kao različiti uzorci istog aktera.)

36

Identifikovanje korisničkog slučaja:

Korisnički slučajevi opisuju:

- Funkcije koje korisnik očekuje od sistema
- Operacije koje kreiraju, čitaju, obnavljaju i brišu informacije.
- Kako se akteri obavestavaju o promenama unutrašnjih stanja sistema i kako oni obavestavaju sistem o spoljašnjim događajima.

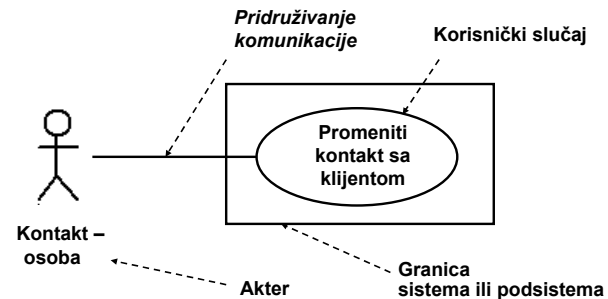
37

Identifikovanje aktera:

- Da bismo identifikovali ko je akter, postavljamo sledeća pitanja:
- Ko koristi sistem?
- Ko dobija informacije od sistema?
- Ko daje informacije sistemu?
- Ko instalira, pokreće ili održava sistem?

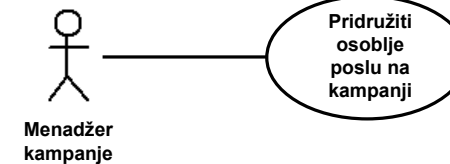
38

Dijagram korisničkog slučaja #1:



39

Dijagram korisničkog slučaja #2:



40

Neophodni i stvarni korisnički slučajevi:

- **Neophodni (essential)** korisnički slučajevi opisuju 'suštinu' korisničkog slučaja u smislu da ne uzimaju u obzir detalje tehnologije ili implementacije.
- **Stvarni (real)** korisnički slučajevi opisuju konkretne detalje korisničkog slučaja u odnosu na projekat.

(U fazi analize svi korisnički slučajevi su suštinski jer se tek odlučuje o projektu.)

41

Alternativni kurs:

- Svaki korisnički slučaj opisuje uobičajeni način prolaska aktera kroz svaku transakciju ili funkciju s kraja na kraj. Moguća alternativna trasa naziva se **alternativni kurs**.
- **Scenario** je korisnički slučaj koji detaljno opisuje alternativni kurs.

42

Šablon za dokumentaciju:

- Jedan način dokumentovanja korisničkih slučajeva je korišćenje šablona (prazne forme ili word- dokumenta koji se popunjava).

Treba da obuhvati:

- ime korisničkog slučaja,
- preduslove (šta treba ispuniti pre nego se desi korisnički slučaj),
- postuslove (šta treba ispuniti pošto se desi korisnički slučaj),
- cilj (šta korisn. slučaj treba da postigne) i
- opis (ukratko ili u gornjem formatu).

43

Zavisnost:

Zavisnost je odnos između dva elementa koja se modeliraju gde promena kod jednog izaziva promenu kod drugog jer je jedan na neki način zavisn od drugog.

Zavisnost se prikazuje isprekidanom linijom sa otvorenom strelicom koja ukazuje na element od koga je drugi zavisn.

44

Stereotip:

Stereotip je specijalna upotreba elementa modela koja je ograničena na ponašanje na poseban način.

Stereotipi se mogu prikazati koristeći ključne reči, na primer 'extend' ili 'include', smeštene u zagrade oblika

<< >>

kao na primer <<actor>>.

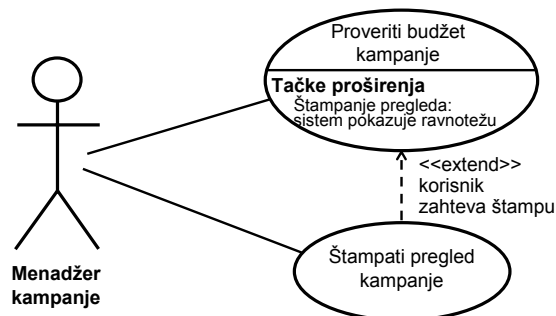
45

Relacija *extend*:

<<extend>> se koristi kada se pokazuje da korisnički slučaj omogućava dodatnu funkcionalnost koja se može zahtevati u drugom korisničkom slučaju.

46

Dijagram korisničkog slučaja koji prikazuje <<extend>>:



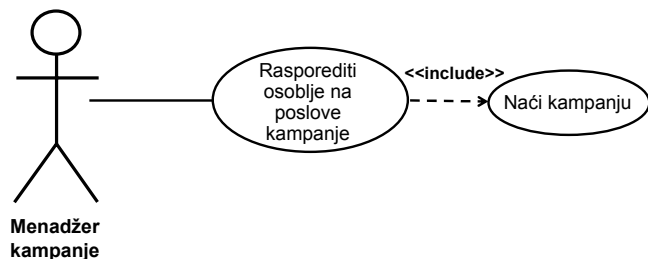
47

Relacija *include*:

<<include>> se primenjuje kada postoji niz ponašanja koji se često koristi kod korisničkih dijagrama, čime se izbegava kopiranje istog opisa svaki put kada se koristi.

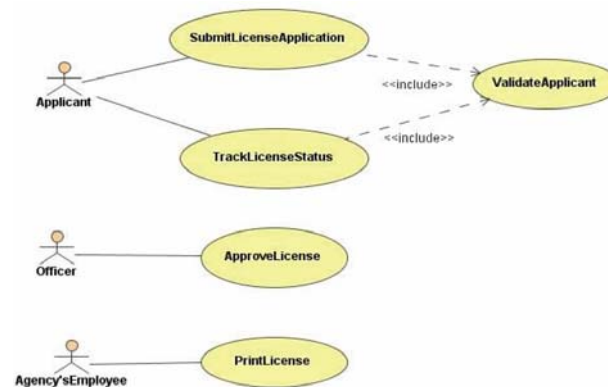
48

Dijagram korisničkog slučaja #1 koji prikazuje <<include>>:



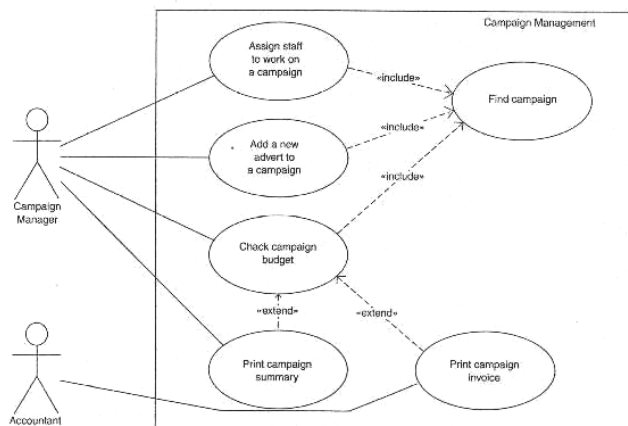
49

Dijagram korisničkog slučaja #2 koji prikazuje <<include>>:



50

Dijagram korisničkog slučaja koji prikazuje <<extend>> i <<include>>:



51

Prototip kao podrška korisničkih slučajeva:

Prototip – radni model dela sistema

(npr. program sa ograničenom funkcionalnošću koji se pravi da bi se proverili neki aspekti onoga što će sistem da radi.

Svrha: korisnici bolje sagledavaju sistem i mogu bolje da definišu korisničke slučajeve.

52

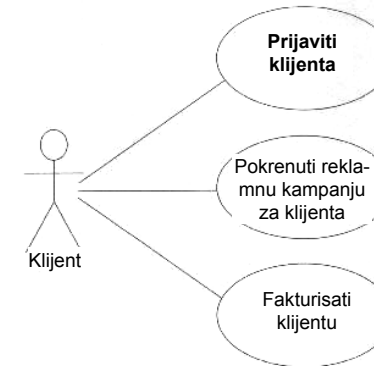
Modelovanje poslova dijagramima korisničkih slučajeva:

Pored modelovanja zahteva za sistem, dijagrami korisničkih slučajeva mogu se koristiti za modelovanje organizacije i posla (business). Akteri su tada ljudi i organizacije izvan kompanije koji saraduju sa funkcijama unutar kompanije.

Svrha: sagledavanje novih poslova ili reorganizacija postojećih i bolje razumevanje korisničkih zahteva.

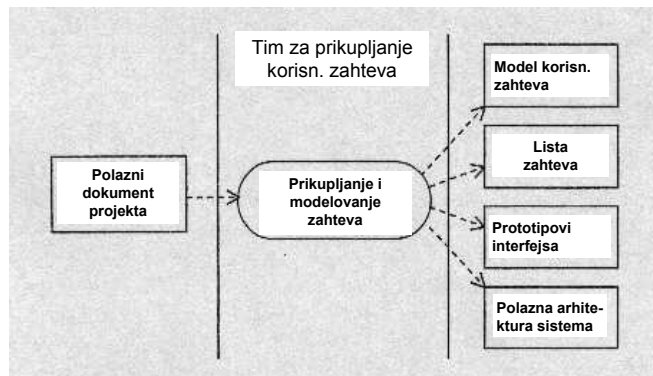
53

Primer modelovanja posla korisničkim slučajevima:



54

Dijagram aktivnosti za prikupljanje i modelovanje zahteva:



55

Kratak pregled:

- Pribavljanje korisničkih zahteva koristi pet tehnika: **čitanje pozadine, intervjuisanje, posmatranje, uzorkovanje dokumenata i anketiranje.**
- Korišćenje ugovorenih **standarda za dokumentovanje zahteva** omogućava komunikaciju analitičara sa ostalim profesionalcima i korisnicima
- **Dijagrami korisničkih slučajeva** predstavljaju tehniku za prikaz korisničkih funkcionalnih zahteva kao prikaza (na visokom nivou) načina korišćenja sistema.

56

