



SVEUČILIŠTE U ZAGREBU



Fakultet
elektrotehnike i
računarstva

Razvoj komunikacijske programske podrške

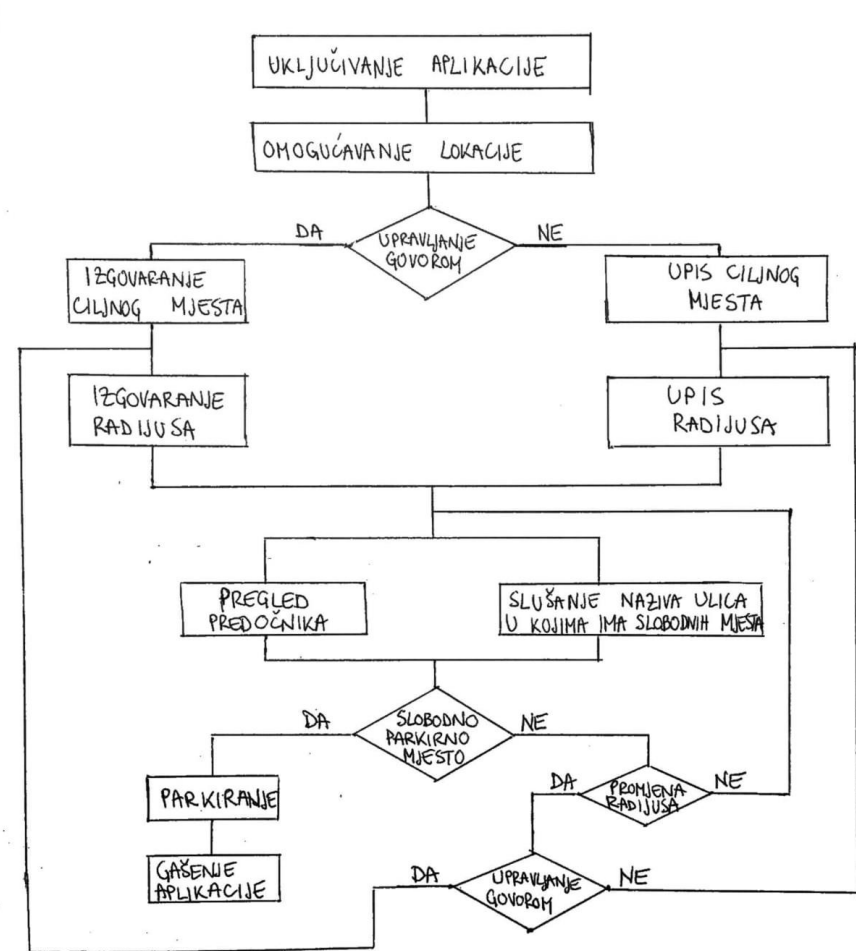
Ak. god. 2021/2022

Procesi prikupljanja i analize programskih
zahtjeva

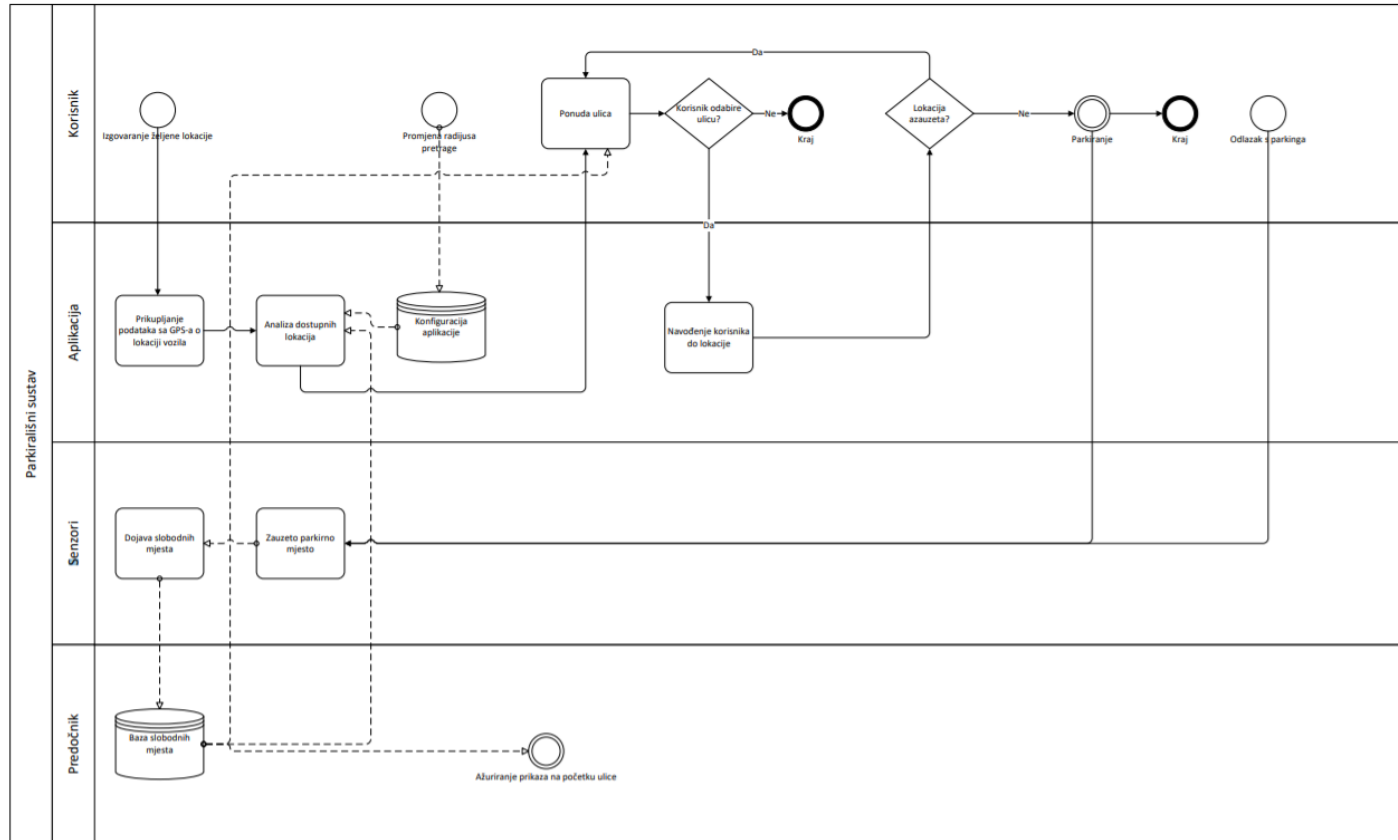
Pametni parkirališni sustav u naselju

- ◆ sadrži senzore na svakom parkirnom mjestu
- ◆ sadrži predočnike koji se nalaze na početku ulice i na kojima je označen broj slobodnih parkirnih mjesta u toj ulici
- ◆ mobilna aplikacija upravljana je govorom korisnika te temeljem prepoznavanja lokacije korisnika u pokretu izgovara nazive ulica u kojima ima slobodnih mjesta, a prema kojima se korisnik kreće
- ◆ radijus dojave slobodnih mjesta korisnik može podesiti u postavkama

Proces pametnog parkirališta

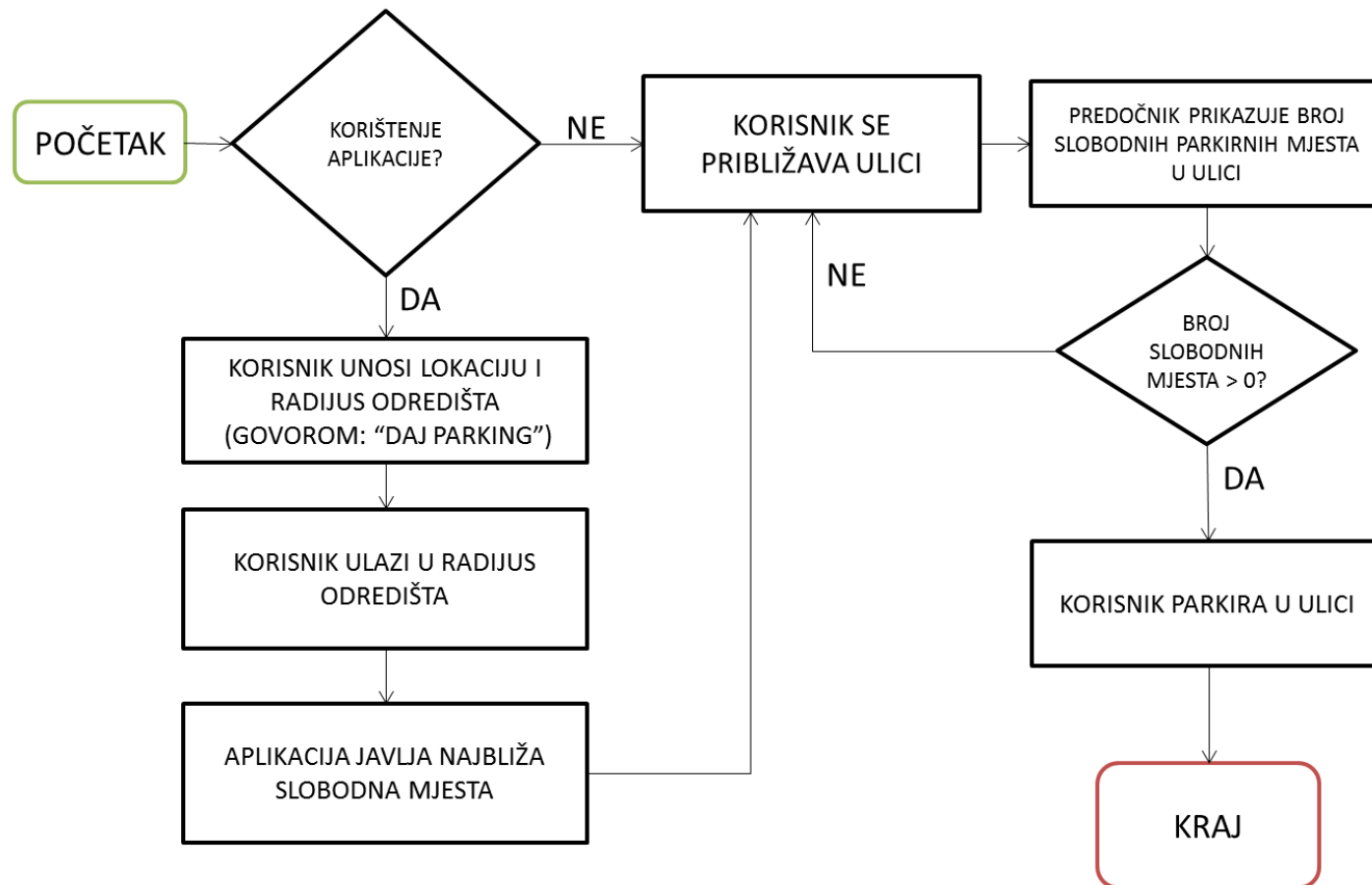


Proces pametnog parkirališta

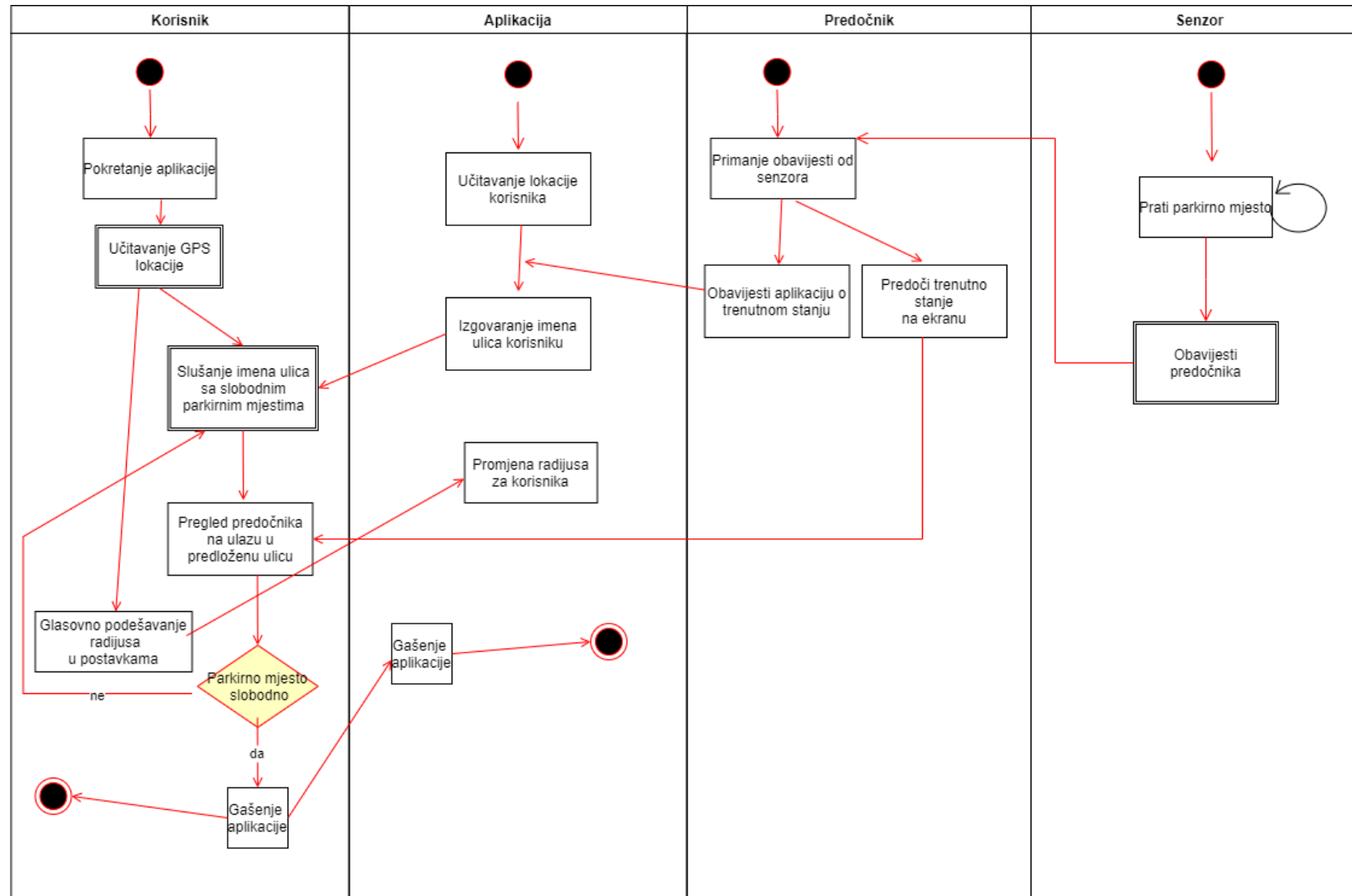


Proces pametnog parkirališta

POSLOVNI PROCES: PAMETNO PARKIRALIŠTE "DAJ PARKING"



Proces pametnog parkirališta



Zaključak

- ◆ Poslovni procesi korisnika mogu uključivati softverska pomagala, ali mogu se odvijati i bez njih
- ◆ Analizom poslovnog procesa korisnika bez primjene sw alata može dati dobar uvid što se želi postići alatom
 - veća učinkovitost
 - veća brzina izvođenja procesa
 - ukidanje nekih aktivnosti
 - itd.

Promjene programskog proizvoda

- ◆ Uzroci nastanka novih verzije prog. proizvoda
 - različita računala/OS
 - nove funkcionalnosti
 - ispravci identificiranih neispravnosti
 - prilagodba specifičnim korisničkom zahtjevima
- ◆ Promjene se odnose na promjene
 - zahtjeva
 - dizajna
 - koda
 - dokumentacije
- ◆ Odluke o promjenama ne donose pojedinci nego timovi stručnjaka!

Uvod

“Najteži dio izgradnje programskog sustava je precizno odlučiti što izgraditi. Nijedan konceptualni dio razvoja nije toliko složen kao definiranje detaljnih tehničkih zahtjeva... Nijedna faza razvoja ne kviri toliko konačni rezultat. Ništa nije tako složeno naknadno ispraviti.”

Frederick P. Brooks



Osnovni pojmovi

- ◆ Zahtjev
- ◆ Funkcija, ograničenje ili svojstvo koje se mora realizirati, zadovoljiti ili osigurati kako bi se ispunile potrebe korisnika programskog proizvoda.

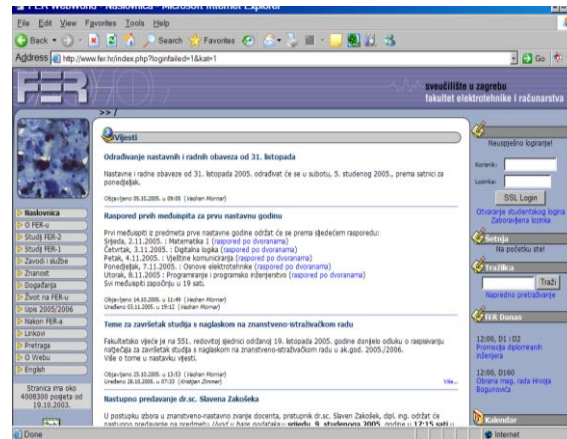
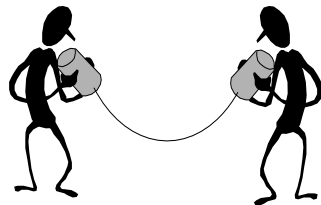


Funkcijski zahtjevi

- ◆ Grupe zahtjeva koji opisuju
 - što programski proizvod treba obaviti
 - nužne poslovne funkcije
 - akcije koje sustav mora biti sposoban izvršiti
 - pojedinačne rezultate izvršavanja programskog proizvoda
 - interakcije sustava i okoline
 - implementacijski neovisno
- ◆ odraz potreba budućih korisnika
- ◆ prikupljaju se iz različitih izvora
- ◆ najčešće se opisuju dijagramima slučajeva uporabe (use case)

Primjer usluge

- ◆ *Potrebno je implementirati uslugu govorne pošte kojoj se može pristupiti putem Web-a. Usluga treba biti dostupna isključivo studentima i djelatnicima FER-a.*



Primjeri funkcijskih zahtjeva

- ◆ *Svaki korisnik ima pridružen vlastiti pretinac govorne pošte kojem pristupa pomoću korisničkog imena i lozinke.*
- ◆ *Korisnik može pregledati popis informacija o pristiglim govornim porukama.*
- ◆ *Nove poruke koje nisu preslušane u popisu su označene na poseban način.*
- ◆ *Korisnik može izbrisati pojedinu govornu poruku.*
- ◆ *Korisnik može pohraniti lokalno pojedinu govornu poruku.*
- ◆ *Korisnik može proslijediti pojedinu govornu poruku.*
- ◆ *Korisnik može odgovoriti na pojedinu govornu poruku.*
- ◆ *Korisnik ima mogućnost pohranjivanja poruke automatskog odgovora (ako je duže vrijeme odsutan).*

Nefunkcijski zahtjevi

Grupe zahtjeva koje opisuju

- ◆ vidljivi aspekti prog. proizvoda koji nisu direktno povezano s funkcijama koje izvršava
 - pogodnost za korištenje (usability)
 - pogodnost za ponovno korištenje (reusability)
 - pouzdanost i sigurnost (L, MTBF; dependability)
 - performanse (alociranje resursa, broj konkurentnih korisnika, vrijeme odziva i sl.)
 - efikasnost (troškovi - HW, ljudi, resursi - i vrijeme realizacije ciljeva PP s uključenom dogovorenom razinom performansi).
 - “*Look and feel*” i ograničenja!

Primjeri ne-funkcijskih zahtjeva

- ◆ *Vrijeme odziva < 5 s (do početka preslušavanja govorne poruke).*
- ◆ *Komunikacija klijentskog preglednika s Web poslužiteljem treba se odvijati korištenjem SSL tehnologije.*
- ◆ *Dizajn aplikacije treba omogućiti periodičko automatsko ažuriranje aplikacije.*
- ◆ *Idle time je 10 minuta.*
- ◆ *Nakon isteka idle-time perioda sustav zahtijeva ponovnu prijavu za rad korisnika (logiranje).*
- ◆ *Aplikacija se treba izvršavati na minimalnim verzijama Internet Explorer 9, Firefox 7, Chrome 14 .*

Osnovni pojmovi (2)

◆ Definicija zahtjeva

- općeniti opis osnovnih usluga koje programski proizvod mora omogućiti
- govorni jezik + dijagrami

◆ Specifikacija zahtjeva

- više detalja, precizno opisane pojedinosti
- strukturirani dokument, može se prikazati u formalnoj notaciji ili u nekom dizajn jeziku
- osnova za proces razvoja, važno za održavanje i evoluciju prog. proizvoda

Osnovni pojmovi (3)

◆ Inženjerstvo zahtjeva predstavlja skup

- metoda,
- tehnika,
- procesa,
- aktivnosti,
- modela
- pomagala

za točno utvrđivanje koje usluge, funkcionalnosti i druga svojstva korisnik zahtijeva od prog. proizvoda, te ograničenja uz koja će se razvijati i nakon isporuke izvršavati.

Osnovni pojmovi (4)

◆ Najčešći problemi vezani uz zahtjeve

- Ljudski/socijalni faktori – problemi prikupljanja, iskazivanja, pretvorbe u tehnički jezik
- Jedinstvenost, tehnička specificiranost, validnost



Razumijevanje potreba korisnika

Dizajn

Elektronički popis za kupovinu



Elektronički popis za kupovinu



Zadatak

Pronađite uljeza – zahtjev koji se ne odnosi na funkcionalnost!



Primjeri zahtjeva

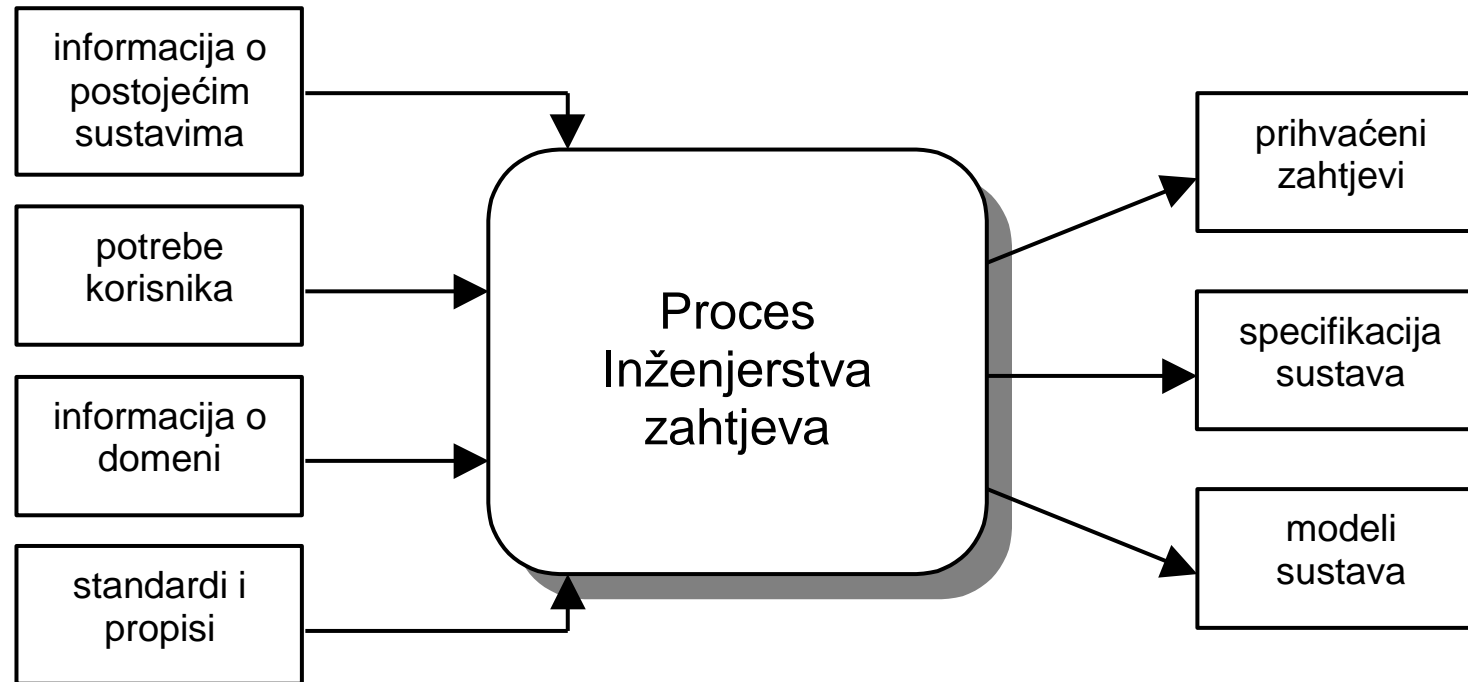
1. *Aplikacija omogućuje kreiranje liste s maksimalno 100 artikala.*
2. *Artikli se u listu dodaju odabirom iz ponuđenih prethodno definiranih kategorija.*
3. *Korisnik može sam unijeti naziv artikla u trenutnu listu.*
4. *Korisnik može izbrisati artikl koji je sam unio.*
5. *Korisnik ne može izbrisati predefinirane artikle.*
6. *Korisnički dodani artikli označeni su posebnim fontom.*
7. *Korisnik može vlastiti artikl pridijeliti određenoj kategoriji predefiniranih artikala.*
8. *Korisnik može sam definirati i brisati svoje kategorije artikala.*
9. *Korisnik može pohraniti listu artikala.*
10. *Korisnik može promijeniti pohranjenu listu dodavanjem i brisanjem artikala iz nje.*

Svojstva dobrih zahtjeva

- ◆ razumljivi krajnjim korisnicima
- ◆ ne kako? nego što?
- ◆ ispravni
- ◆ izvedivi
- ◆ potpuni
- ◆ koncizni
- ◆ precizni
- ◆ jasni
- ◆ nedvosmisleni
- ◆ konzistentni
- ◆ mogućnost praćenja kroz projekt
- ◆ jednostavno unošenje promjena



Proces inženjerstva zahtjeva



Proces inženjerstva zahtjeva (2)

- ◆ Različiti procesi koji variraju ovisno o
 - aplikacijskoj domeni
 - sudionicima
 - organizaciji koja se bavi prikupljanjem zahtjeva.
- ◆ Generički proces inženjerstva zahtjeva
 - generički skupovi aktivnosti sadržani u većini procesa



Proces prikupljanja zahtjeva



Prikupljanje zahtjeva (1)

- ◆ Nužne socijalne, komunikacijske i upravljačke vještine
- ◆ Najmanje tehnička faza razvoja SW-a
- ◆ Svrha
 - Osigurati narativni opis funkcijskih i ostalih zahtjeva za koje interesni sudionici očekuju da će biti zadovoljeni u razvijenom i implementiranom programskom proizvodu.
- ◆ Problemi prikupljanja zahtjeva
 - komunikacijski, tehnički, ljudsko ponašanje
 - najčešće kombinirani
- ◆ Različite metode za prevladavanje različitih problema

Uloga zajedničkog rječnika

- ◆ U prikupljanju zahtjeva potrebno je definirati rječnik
 - utvrđivanje semantičke interpretacije zahtjeva
 - definicija tehničkih izraza koji će se koristiti
 - definicija mjernih veličina, pravila klasifikacije i sl.
- ◆ Projektni rječnik
 - dogovoreni i prihvaćeni nazivi i opisi
- ◆ Višestruko korištenje rječnika
 - moguće unutar iste klase proizvoda koji koriste istu terminologiju
- ◆ Zahtjevi su dobro formulirani ako se svi termini mogu jedinstveno mapirati s dogovorenim i prihvaćenim rječnikom!

Primjer rječnika

ict-aac.hr/katalog/index.php/Glavna_stranica#Uvod_E2.80.93_osnovni_pojmovi.2C_definicije_i_povijest_potpomognute_komuni

- [5.5 Evaluacija kvalitete korisničkog doživljaja](#)
- [5.6 Hassenzahlov model korisničkog doživljaja](#)
- [6 Konferencije o potpomognutoj komunikaciji](#)
- [7 Literatura](#)

Uvod – osnovni pojmovi, definicije i povijest potpomognute komunikacije (PK)

Komunikacija je proces razmjene poruka prilikom kojeg se koristi dogovoreni sustav znakova. Jedno od temeljnih ljudskih prava upravo je pravo na komunikaciju, odnosno na primanje i slanje poruka. Postoje skupine ljudi koji iz različitih razloga nisu u mogućnosti komunicirati putem osnovnog sredstva komunikacije među ljudima-govorom. Navedene osobe nazivaju se osobama sa složenim komunikacijskim potrebama, a metode komunikacije koje potpomažu i/ili nadomještaju njihov govor podrazumijevaju uporabu potpomognute komunikacije.

Hrvatska prevedenica za augmentativnu i alternativnu komunikaciju (eng. **Augmentative and Alternative Communication**) je **potpomognuta komunikacija** (PK). Prevedenicom se upućuje na činjenicu da je potpomognut komunikacijski proces bilo nekim alternativnim sredstvima ili pojačavanjem (augmentacijom) postojećih. Sustav PK pomaže osobama koje ne govore da se izraze, no on se može koristiti i kao sustav potpore koji osoba koristi za razumijevanje jezika ili komunikacije u cjelini. Dakle, strategije PK koriste se s osobama koje imaju teškoća u proizvodnji i/ili razumijevanju oralnog i/ili pisanog jezika, a jednako je primjeren za korištenje kod djece i odraslih ^[1]. Uporaba PK može značajno unaprijediti kvalitetu života osoba s teškoćama, omogućiti im ravnopravije sudjelovanje u društvu kao i mogućnost boljeg obrazovanja te veću razinu samostalnosti ^[2].

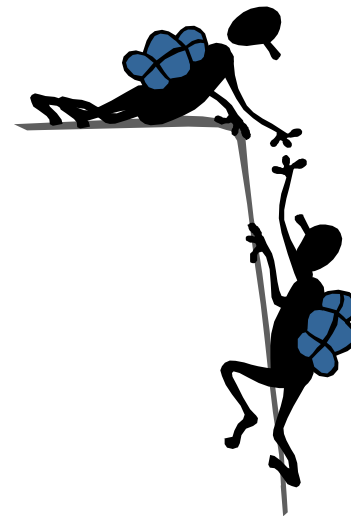
U skupinu korisnika PK s razvojnim teškoćama ubrajamo osobe koje imaju poremećaj iz autističnog spektra, cerebralnu paralizu, intelektualne teškoće i razvojnu govornu apraksiju ^[3]. Korisnici PK sa stečenim teškoćama imaju složene komunikacijske potrebe zbog različitih neuroloških bolesti ili oštećenja kao što su: progresivne neurološke bolesti, afazija, moždani udar, traumatska oštećenja mozga, demencija i amiotrofična lateralna skleroza ^[3]. Potpomognuta komunikacija uključuje sve metode komunikacije koje zamjenjuju ili potpomažu uobičajene načine komunikacije kada su oni narušeni ili kada ne odgovaraju potrebama pojedinca ^[1]. PK uključuje sve oblike komunikacije (osim oralnog govora) koji se koriste za izražavanje misli, potreba, želja i ideja. S napretkom tehnologije i povećanom svjesnosti o različitim komunikacijskim mogućnostima koje su dostupne pojedincima sa različitim vrstama teškoća, postoji veliki potencijal za pružanje boljeg načina komunikacije većem broju ljudi ^[2] ^[4].

PK sustav obuhvaća niz tehnika i tehnologije koje zajedno čine metodu totalne komunikacije za određenu osobu. Osoba može koristiti facijalnu ekspresiju, položaj tijela, pogled, vokalizaciju različitog tona i jačine ali usto i specijalizirane sustave kao što su znakovanje ili simboli i/ili elektroničke uređaje sa sintetički proizvedenim glasom ^[2]. PK sustavi su individualizirani i razlikuju se ovisno o sposobnostima, mogućnostima i potrebama pojedinca.

Važno je napomenuti da bez obzira koji se PK sustav koristi, komunikacija putem PK je sporija od tipične komunikacije ^[1].

Prikupljanje zahtjeva (2)

- ◆ Metode visoke razine
 - vrlo općenito izražavanja potreba i zahtjeva
 - definiranje okvira za detaljno prikupljanje zahtjeva
- ◆ Metode niske razine
 - više detalja, tehničke specifikacije, dulje trajanje, uključuje i pregovaranje
- ◆ “Znanost prestaje, a počinje kaos!”
- ◆ Zahtjevi se izražavaju u pisanoj formi
 - govornim jezikom
 - tablicama i crtežima
 - dijagramima i modelima



Vizija!

- ◆ Prilagođeni predložak vizije programskog proizvoda
- ◆ Cjelovita vizija programskog proizvoda

Metode prikupljanja zahtjeva

- ◆ Introspekcija
- ◆ Proučavanje tehničke dokumentacije
- ◆ Intervjui
- ◆ Ankete
- ◆ Promatranje
- ◆ Oluja ideja (Brainstorming)
- ◆ Izrada prototipa

Introspekcija

- ◆ Uživljavanje u ulogu korisnika (popularna)
- ◆ Igranje uloga (razvijatelji zamišljaju da su u ulozi budućih korisnika)
- ☺ Dobra početna tehnika za proizvode namijenjene širokom tržištu
- ☺ Jednostavna, brza, jeftina, nema nesporazuma
- ☹ Razlika “naive users” – eksperti, tehničko (pred)znanje ...
- ☹ Često neprimjenjiva (“Nemam nikakvu ideju!”)
- ☹ Može navesti na potpuno pogrešan trag
- ◆ *Mantra of usability people is: "Users are not like us!"*

Promatranje

◆ Primjenjuju se ako

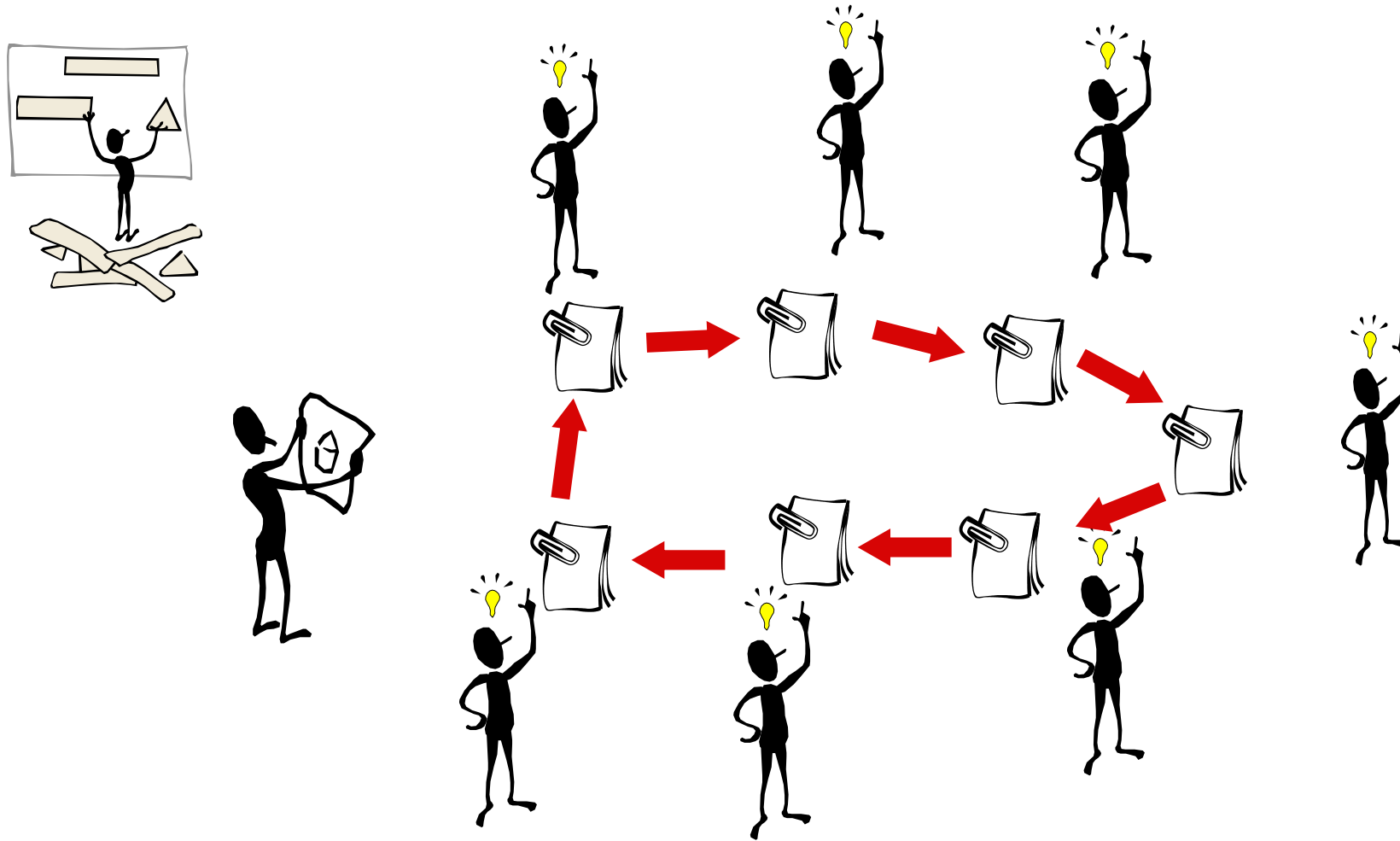
- sudionici poslovnog procesa imaju fragmentarno znanje o cijelom procesu
- previše je složeno prikupiti zahtjeve ostalim metodama.
- **Pasivno promatranje**
 - promatrač promatra kako sudionici izvršavaju posl. proces
- **Aktivno promatranje**
 - promatrač sudjeluje u izvođenju poslovnog procesa
- **Promatranje uz objašnjenje korisnika**
- Treba promatrati dulji period vremena, u različitim vremenskim intervalima i pod različitim radnim opterećenjima

☹ **Korisnici mijenjaju ponašanje ako znaju da su promatrani**

Brainstorming (Oluja ideja)

- ◆ Konferencijska tehnika za definiranje novih ideja ili pronalaženje rješenja specifičnih problema u okolini lišenoj osude, isključivanja i pravila
- ◆ Nije prikladno za analizu problema i donošenje odluka
- ◆ Nužan je dobar voditelj sesije
- ◆ *Trigger* pitanja:
 - Koje usluge trebaju biti podržane od strane budućeg PP?
 - Koji su ulazni podaci i koji su rezultati izvršavanja PP-a?
 - Koja pitanja treba prikupiti anketama?
 - Koje probleme treba riješiti?

Brainstorming



Intervjui

◆ Strukturirani intervju

- pitanja definirana unaprijed

◆ Nestrukturirani intervju

- teme ispitanika potiču na slobodno razmišljanje

◆ Kombinirani

- najčešći

◆ Intervjuiraju se

- domenski eksperti (osiguravaju domensko znanje)
- potencijalni korisnici (informacije o najčešćim slučajevima uporabe)

Intervjui (2)

- 😊 Mogućnost prikupljanja bogate kolekcije informacija i različitih pogleda na PP
- 😊 Nove ideje, zanimljivi osvrti na problem koji PP treba riješiti
- ♦ **Intervjuiranje je složena vještina kojom treba dobro ovladati te imati urođene soc. psih. predispozicije**
- 😞 Prikuplja se velika količina kvalitativnih podataka koje je teško analizirati
- 😞 Složena je usporedba različitih odgovara
- 😞 Potrebno je eliminirati utjecaj sugovornikovog stava prema problemu/rješenju

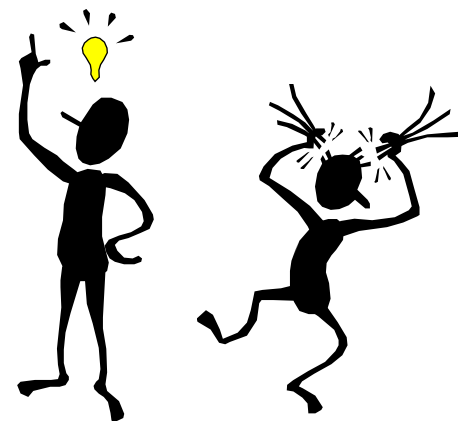
Intervjui (3)

Izbjegavati:

- Pitanja s nametanjem vlastitog mišljenja
 - “Moramo li doista primijeniti scenarij B? Nije li bolji scenarij A?”
 - “Nećete valjda implementirati to na ovaj način?”
- Retorička pitanja

◆ Trebaju sadržavati pitanja vezana uz

- Specifične informacije – *Five w's: What? Who? When? Where? Why?*
- Viziju budućnosti
- Alternativne ideje
- Minimalno prihvatljivo rješenje
- Ostale izvori informacija



Ankete

- ◆ Prikupljanje informacija od velikog broja korisnika
- ◆ Pasivni način prikupljanja
- ◆ Primjena alata za anketiranje
 - Mogu se prikupiti zahtjevi “na daljinu” (pomagala)
 - Olakšana analiza
 - Prikladno za prikupljanje kvantitativnih podataka
- ◆ Specifična problematika
 - potreba za dodatnim pitanjima
 - kategorizacija ispitanika

Ankete (2)

◆ Sadržje

- Pitanja s više mogućih odgovora iz skupa ponuđenih odgovora
- Rangirajuća pitanja
- Ocjenjujuća pitanja
- Komentari

◆ Izbjegavati

- sugestivna pitanja
- dvosmislena i nerazumljiva pitanja
- rizik premalog uzorka ispitanika

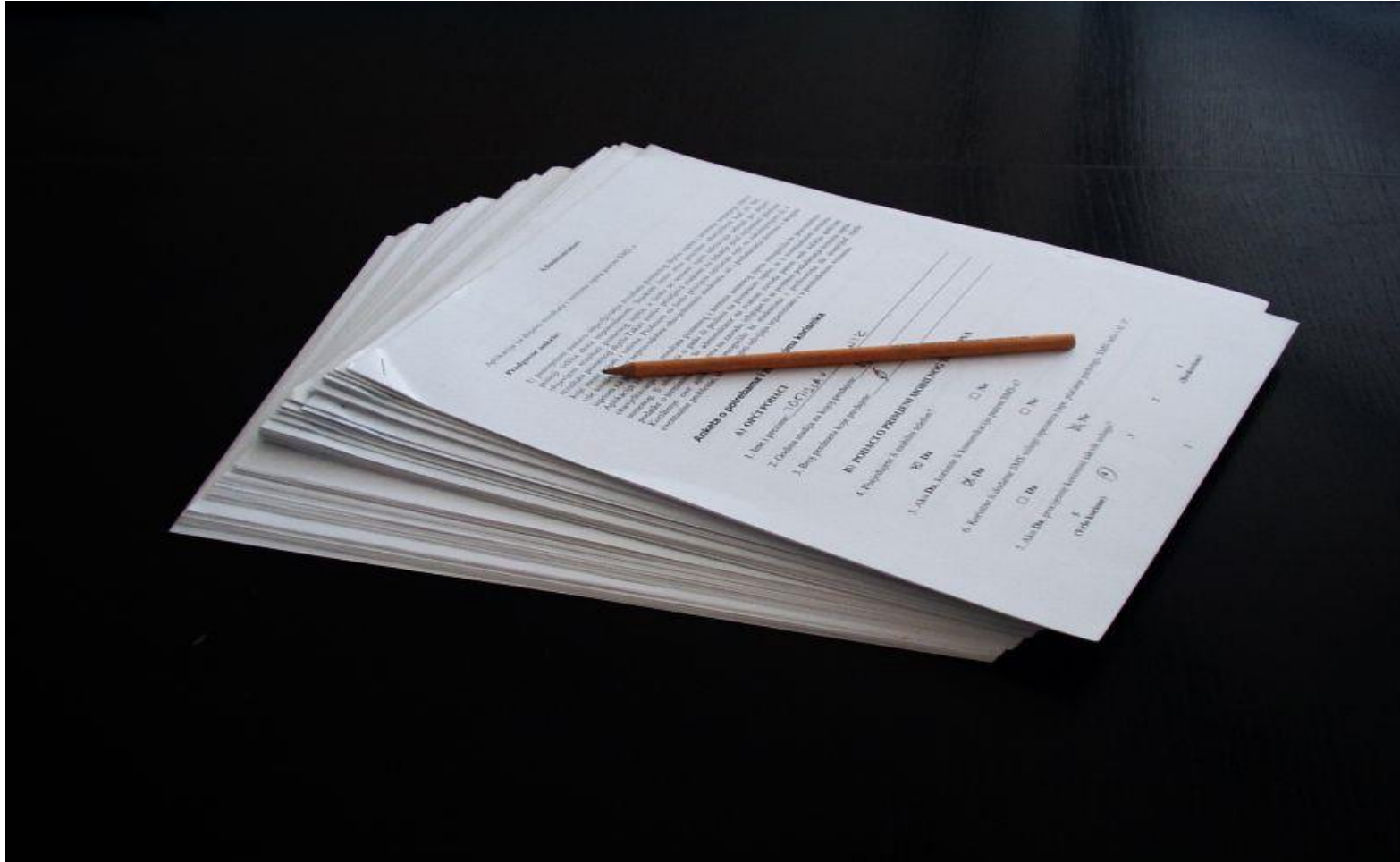
◆ U praksi važno najprije testirati anketu!

◆ [Studijski primjer anketa](#)

Ankete (3)

- ☹ Nedostatak utjecaja na proces ispunjavanja ankete (na sadržaj se ne smije utjecati!) – nemogućnost motiviranja
- ☹ Otpor prema anketiranju
- ☹ Preskakanje pitanja
- ☹ Davanje više odgovora nego je traženo
- ☹ Odgovaranje na pitanja koja nisu namijenjena određenoj kategoriji ispitanika
- ☹ Nemogućnost izražavanja stvarnih potreba korisnika

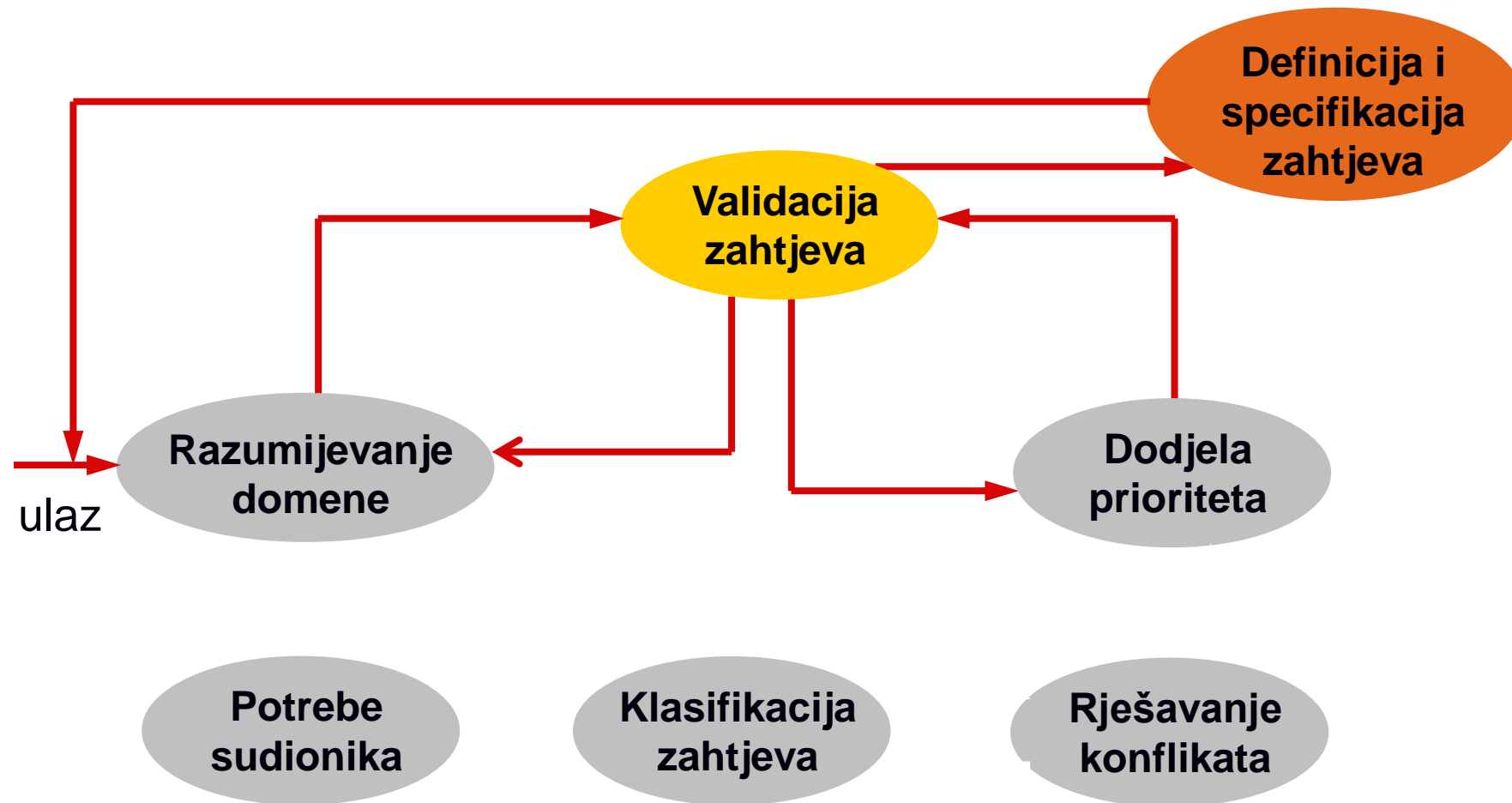
Obrada anketa...



Elektroničke ankete

- ◆ Mark Pickavance, Brian Turner. *Best survey tools for customer research in 2021*,
- ◆ <https://www.techradar.com/best/best-survey-tools>
- ◆ Google Forms

Proces analize zahtjeva



Konfliktni zahtjevi

- ◆ Dva su zahtjeva u konfliktu ako programska implementacija jednog zahtjeva onemogućuje implementaciju drugog.



Preklapajući zahtjevi

- ◆ Dva se zahtjeva preklapaju ako se odnose na istu funkcionalnost, svojstvo ili ograničenje programskog proizvoda, ali je iskazuju drugim riječima.



Matrica ovisnosti zahtjeva

ZAHTJEVI	R2	R3	R4
R1	✓	✓	Konflikt
R2		Preklapanje	✓
R3	Preklapanje		Konflikt
R4	✓	Konflikt	

Dodjela prioriteta

- ◆ Korisnici/kupci: "Svi zahtjevi su kritični i treba ih implementirati!"
- ◆ Potrebno je odrediti razine prioriteta svakog zahtjeva
- ◆ Viši prioritet
 - Funkcijski zahtjevi koji uključuju poslovne zahtjeve
 - Zahtjevi čiji slučajevi uporabe imaju veliku frekvenciju korištenja
 - Jezgreni poslovni procesi
- ◆ Treba odrediti verzije prog. proizvoda s obzirom na implementaciju različitih prioriteta zahtjeva

Dodjela prioriteta (2)

Primjer 1

PRIORITET

Opis

Visoki

Implementacija zahtjeva je neophodna; zahtjev će se pojavljivati u svim narednim verzijama progr. proizvoda.

Srednji

Važan zahtjev, implementacija je moguća u narednim verzijama bez velikog utjecaja na funkcionalnost aktualnog prog. proizvoda.

Niski

Poboljšanja, implementacija u jednoj od budućih verzija ako resursi dopuste.

Dodjela prioriteta (2)

PRIORITET	Opis
Osnovni	Prog. proizvod nije prihvatljiv ako zahtjev nije implementiran
Uvjetni	Zahtjev unosi poboljšanja, prog. proizvod je prihvatljiv ako zahtjev nije implementiran.
Korisni	Poboljšanja, implementacija u jednoj od budućih verzija ako resursi dopuste

Modeliranje programskih zahtjeva

◆ Ciljevi i svrha modeliranja

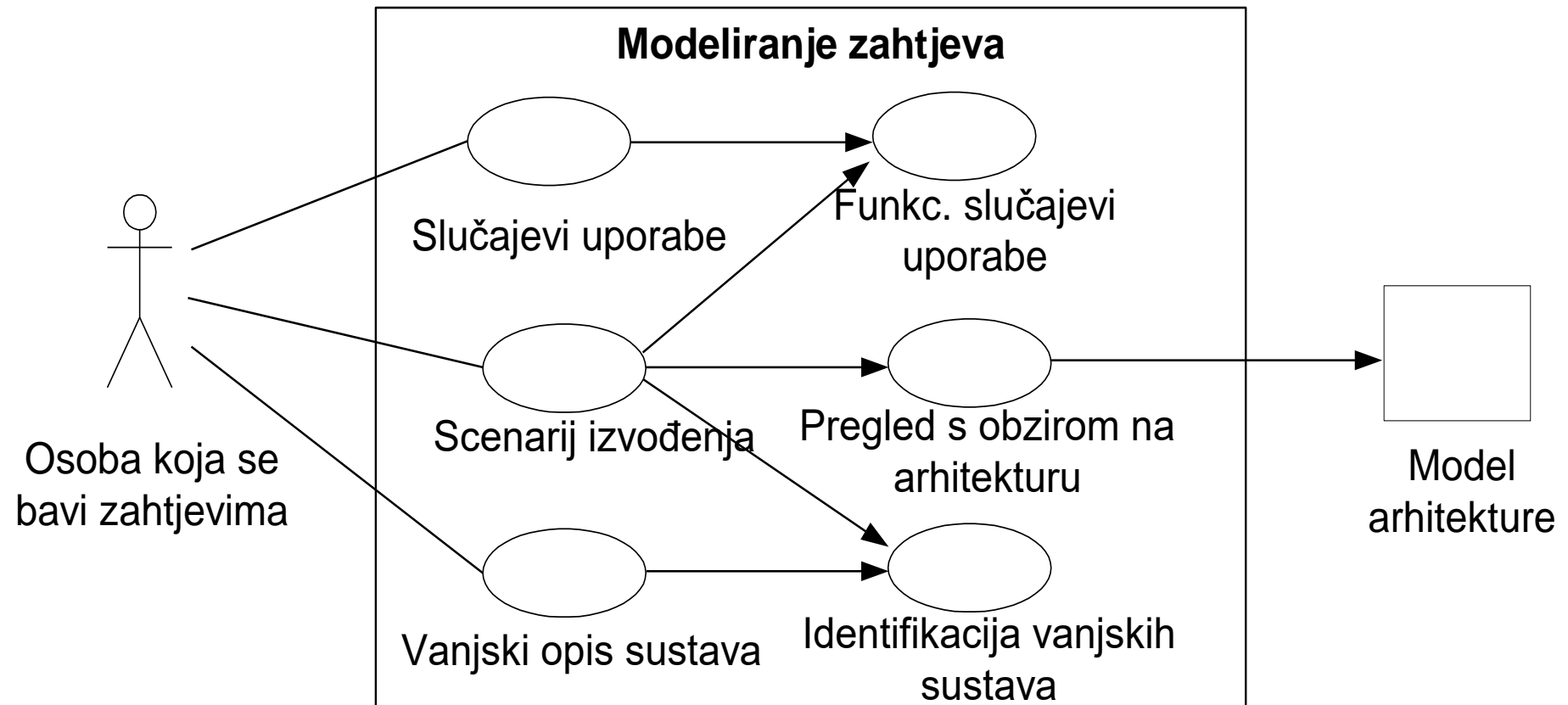
- bolje razumijevanje zahtjeva sustava sa stajališta korisnika
- bolja komunikacija korisnik – razvijatelji
- poboljšano preslikavanje poslovne domene u programsku domenu
- različite grupe korisnika – različite perspektive
- povezivanje logičke i fizičke arhitekture
- omogućava se stvaranje komponenti i njihovo višestruko korištenje
- svaki model ima vlastitu perspektivu programskog proizvoda, odabir ovisi o problemu koji se želi riješiti

Modeliranje programskih zahtjeva (2)

- ◆ Slučaj uporabe (Use Case) - UML dijagram
 - Opisuje što korisnik zahtijeva od sustava i koristi se za aktivno pregovaranje s korisnikom
- ◆ Osnovna zadaća
 - Odrediti granice sustava (ključno)
 - Odrediti ponašanje sustava
 - Prikazati odnos sustav – okolina
- ◆ Osnovni dijelovi
 - Akter
 - Slučaj uporabe
 - Sustav
- ◆ Scenarij (prolaz kroz slučaj uporabe): primarni, sekundarni

Modeliranje zahtjeva

Primjena slučajeva uporabe



Dijagram slučajeve uporabe

◆ Odrediti aktere sustava

- vanjski entiteti koji su u interakciji s programskim sustavom
- akteri mogu biti osobe, uređaji/sklopovlje, drugi programi.
- akteri se nalaze IZVAN SUSTAVA!
- primarni akteri iniciraju interakciju
- sekundarni akteri odgovaraju na akcije

◆ Svaki slučaj uporabe treba imati jasno definirani cilj i svrhu

- definirane akcije koje poduzima primarni akter kako bi ostvario predviđeni cilj

Dijagram slučajeve uporabe

2. Odrediti slučajeve uporabe

- opisuju ponašanje sustava kako bi se odgovorilo na određene potrebe korisnika
- ◆ Pri određivanju mogu pomoći odgovori na sljedeća pitanja:
 - Tko koristi sustav?
 - Tko instalira sustav?
 - Tko pokreće sustav?
 - Tko opskrbljuje sustav ulaznim informacijama?
 - Tko koristi informacije nastale kao rezultat izvršavanja programskog sustava?
 - Treba li sustav u zadanom periodu poduzeti neke automatske akcije?

Dijagram slučajeve uporabe

3. **Odrediti preuvjete i post-uvjete**
 - identificirati stanje sustava na početku i na kraju izvršavanja određenog slučaja uporabe.
4. **Odrediti korake u svakom slučaju uporabe**
 - identificirati primarne i sekundarne scenarije kako bi primarni akter ostvario cilj slučaja uporabe.
5. **Nacrtati dijagram slučaja uporabe**
 - strelica pokazuju na inicijatora pojedine aktivnosti:
 - primarni akter → slučaj uporabe → sekundarni akter

Verifikacija i validacija programskih zahtjeva

◆ Verifikacija

- provjera jesu li zahtjevi ispravni

Validacija

- provjera opisuju li zahtjevi dobro željeni PP

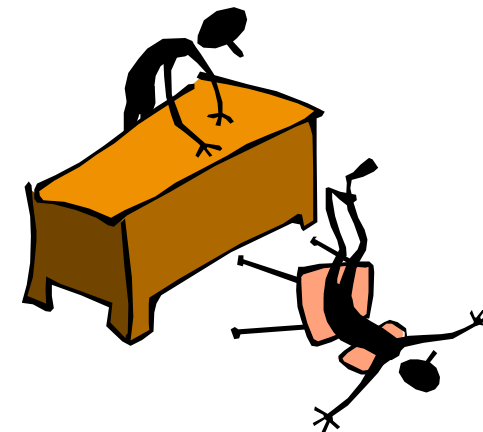
◆ Neadekvatni V&V

- greške propagiraju do dizajna i implementacije (troškovi)

◆ Provjerava se

- ◆ izvedivost zahtjeva
- ◆ potpunost zahtjeva
- ◆ konzistentnost zahtjeva
- ◆ ispravnost zahtjeva

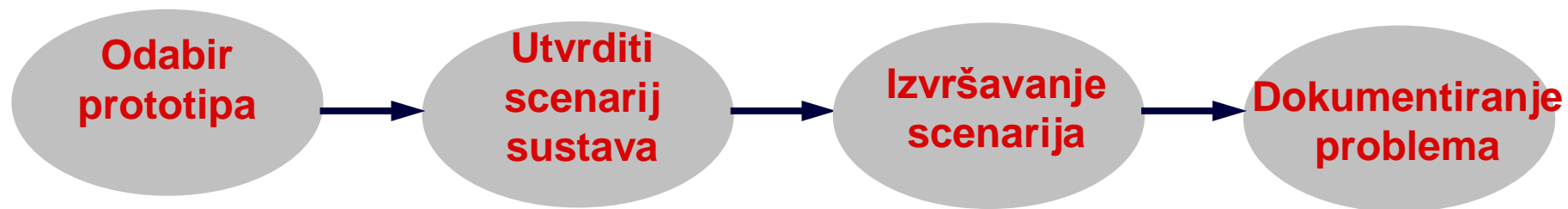
■



Tehnike verifikacije i validacije zahtjeva

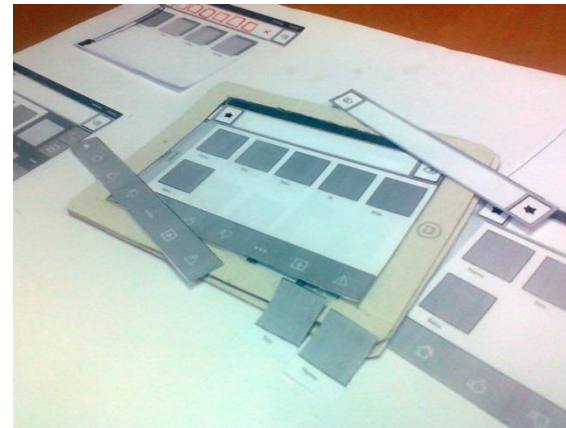
1. Izrada prototipa (*throw-away prototype*)

- prototip niske vjernosti prikaza
- prototip visoke vjernosti prikaza
- funkcijski prototip



Primjer - Izrada prototipa

Prototip niske vjernosti prikaza



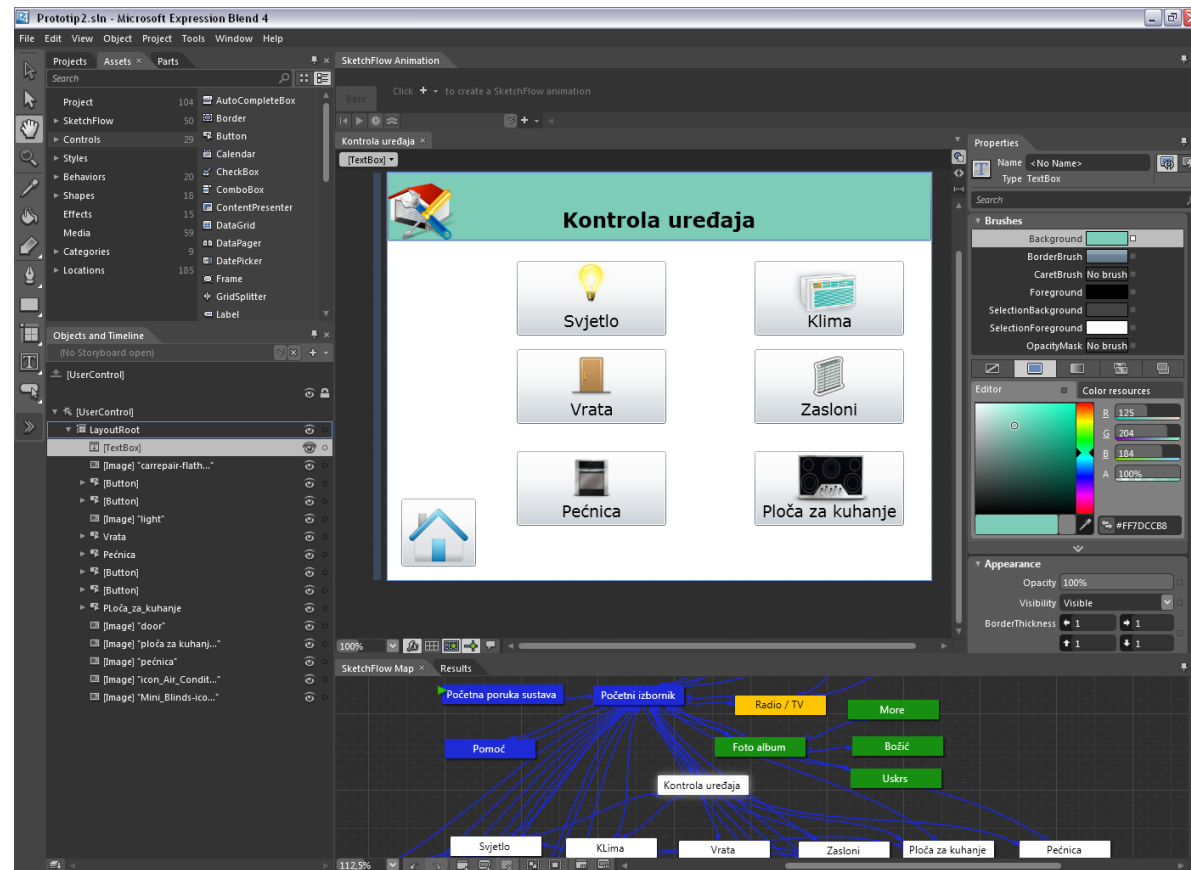
Primjer – Izrada video prototipa



Prototip visoke vjernosti prikaza



Primjer – Izrada prototipa



Alati za izradu prototipa

◆ Proto.io <https://www.youtube.com/watch?v=6t2YjjvKZR8>

◆ Invision

<http://www.invisionapp.com/enterprise>

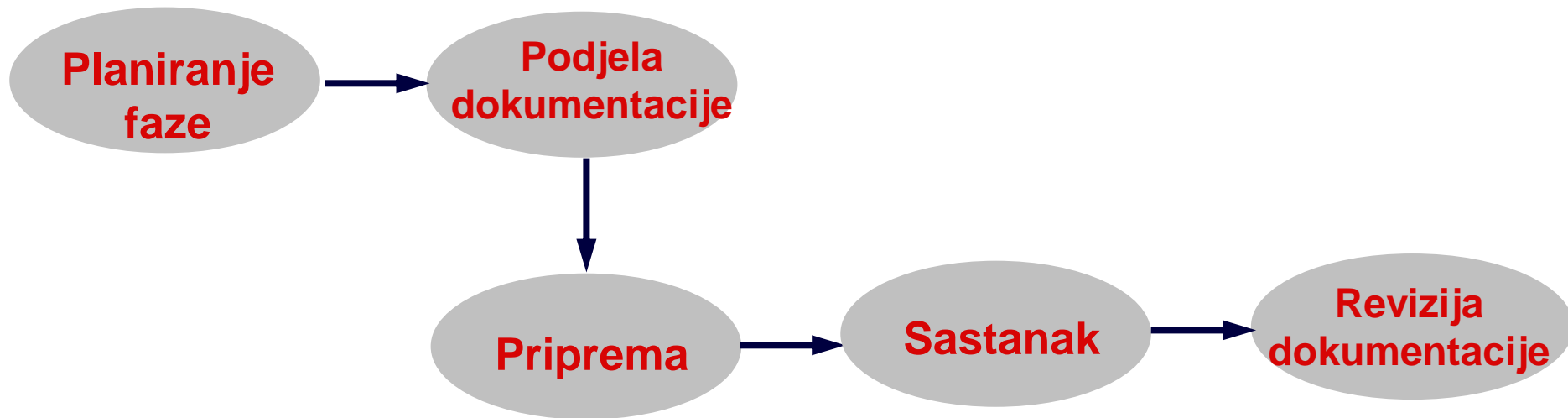
◆ Pencil Project (open source)

<http://pencil.evolus.vn/>

Tehnike verifikacije i validacije zahtjeva (2)

2. Pregledi zahtjeva

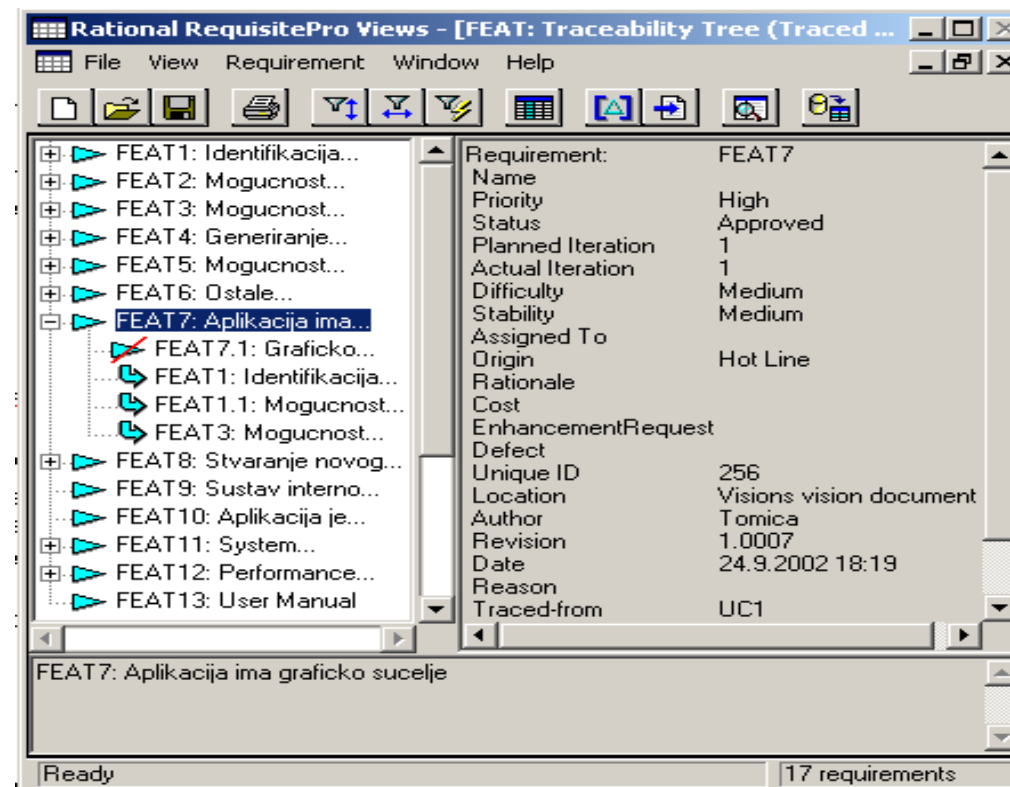
- drugi naziv: inspekcije
- formalni, neformalni
- traže se konflikti, kontradikcije, neispravnosti.



Primjer pomagala za zahtjeve

◆ *Rational RequisitePro*

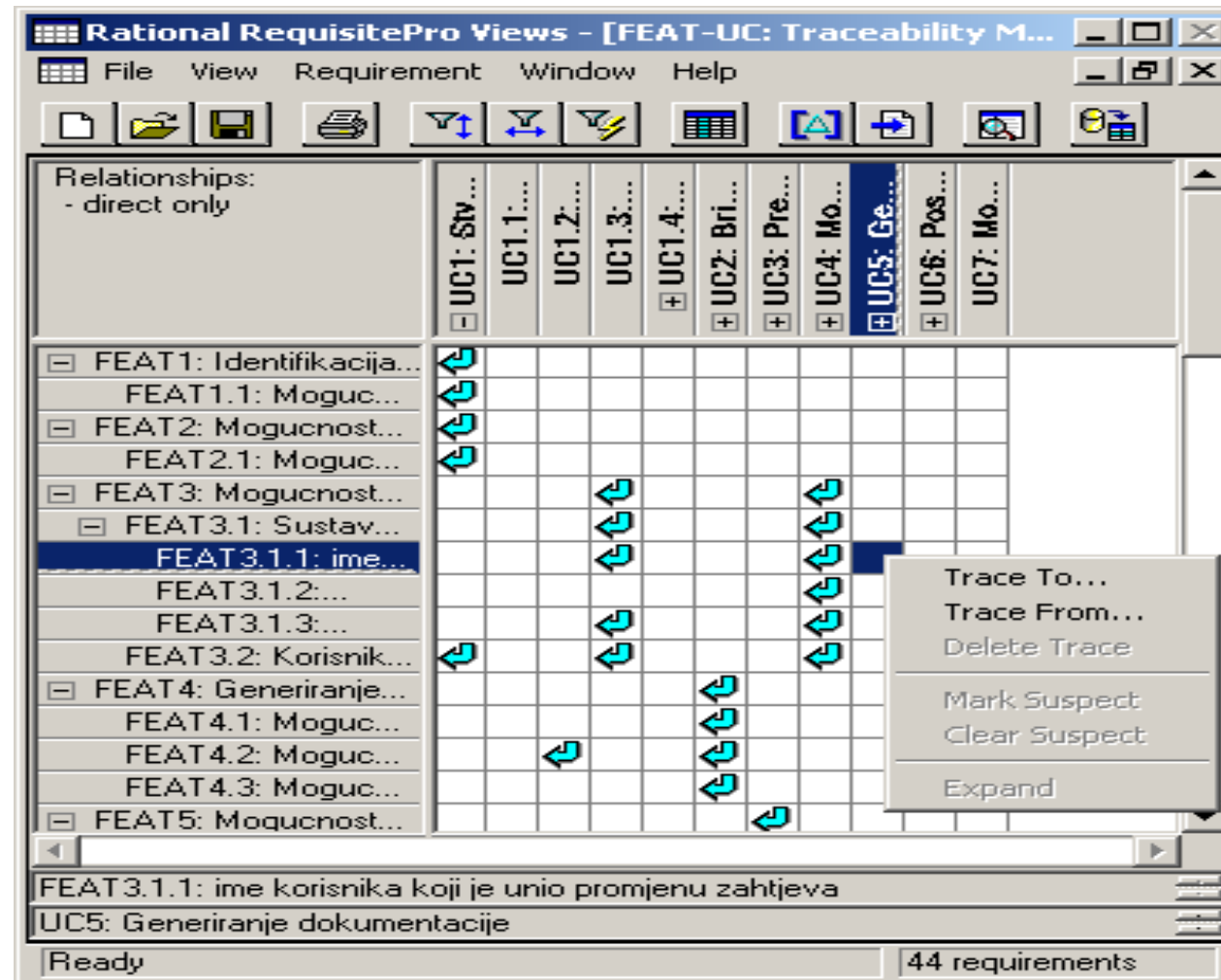
- podrška prikupljanju i upravljanja zahtjevima
- mogućnost praćenja - matrica, stablo



Primjer pomagala za zahtjeve (2)

Requirements:	Priority	Status	Planned Iterati	Actual Iteratio	Stability	Difficulty
<input type="checkbox"/> FEAT1: Identifikacija zahtjeva	High	Approved	1	1	Medium	Medium
FEAT1.1: Mogucnost stvaranja i pohrane novog zahtjeva	High	Approved	1	1	Medium	Medium
<input type="checkbox"/> FEAT2: Mogucnost klasifikacije zahtjeva	High	Approved	1	1	Medium	Medium
FEAT2.1: Mogucnost stvaranja novog tipa zahtjeva	Medium	Approved	1	1	Medium	Medium
<input type="checkbox"/> FEAT3: Mogucnost upravljanja promjenom zahtjeva	Medium	Approved	1	1	Medium	Medium
<input type="checkbox"/> FEAT3.1: Sustav pamti informacije o promjeni zahtjeva	Medium	Approved	1	1	Medium	Medium
FEAT3.1.1: ime korisnika koji je unio promjenu zahtjeva	Medium	Approved	1	1	Medium	Medium
FEAT3.1.2: datum i vrijeme kada e promjena zahtjeva...	Medium	Approved	1	1	Medium	Medium
FEAT3.1.3: razlog promjene zahtjeva	Medium	Approved	1	1	Medium	Medium
FEAT3.2: Korisnik može uspostaviti veze medu zahtjevima	Medium	Approved	1	1	Medium	Medium
<input type="checkbox"/> FEAT4: Generiranje dokumenata i metrika	Medium	Proposed	1	1	Medium	Medium
FEAT4.1: Mogucnost generiranja specifikacije zahtjeva u...	Medium	Approved	1	1	Medium	Medium
FEAT4.2: Mogucnost generiranja specifikacije u doc...	Low	Proposed	3	1	Medium	Medium
FEAT4.3: Mogucnost generiranja specifikacije zahtjeva u...	Medium	Approved	1	1	Medium	Medium
<input type="checkbox"/> FEAT5: Mogucnost pretrazivanja zahtjeva	High	Approved	1	1	Medium	Medium
FEAT5.1: Mogucnost pretraživanja zahtjeva po postojećim.	High	Approved	1	1	Medium	Medium
FEAT5.2: Mogucnost pretraživanja zahtjeva po...	Medium	Approved	1	1	Medium	Medium
FEAT5.3: Mogucnost ispisa liste svih postojećih zahtjeva...	High	Approved	1	1	Medium	Medium
<input type="checkbox"/> FEAT6: Ostale funkcionalnosti	High	Approved	1	1	Medium	Medium
FEAT6.1: Mogucnost brisanja zahtjeva	High	Approved	1	1	Medium	Medium
FEAT6.2: Mogucnost pregleda postojećeg zahtjeva	High	Approved	1	1	Medium	Medium
FEAT6.3: Mogucnost promjene zahtjeva	High	Approved	1	1	Medium	Medium

Primjer pomagala za zahtjeve (3)



Tehnike verifikacije i validacije zahtjeva (3)

3. Izražavanje zahtjeva formalnim jezicima

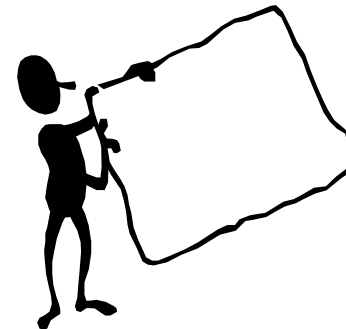


3. Simulacija sustava

- najčešće za nefunkcijske zahtjeve

Zadatak

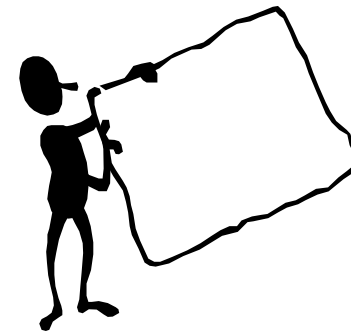
- ◆ Artikli se u listu dodaju odabirom iz ponuđenih prethodno definiranih kategorija.
- ◆ *Odredite jedan konfliktan zahtjev!*



Zadatak

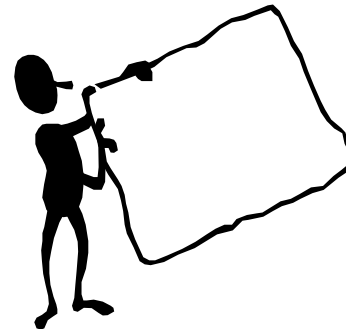
- ◆ Korisnik može vlastiti artikl pridijeliti određenoj kategoriji predefiniranih artikala.

- ◆ *Odredite jedan preklapajući zahtjev!*



Zadatok

- ◆ Dodíte priority navedením zahtevima:
 - Osnovní
 - Uvjetní
 - Korisní



Skup funkcijskih zahtjeva

1. Aplikacija omogućuje kreiranje liste s maksimalno 100 artikala.
2. Artikli se u listu dodaju odabirom iz ponuđenih prethodno definiranih kategorija.
3. Korisnik može sam unijeti naziv artikla u trenutnu listu.
4. Korisnik može izbrisati artikl koji je sam unio.
5. Korisnik ne može izbrisati predefinirane artikle.
6. Korisnik može uz artikl učitati vlastitu sliku koja ga predstavlja (npr. omiljeno piće).
7. Korisnik može pohraniti listu artikala.
8. Korisnik može izbrisati pohranjenu listu artikla.
9. Korisnik može promijeniti pohranjenu listu dodavanjem i brisanjem artikala iz nje.
10. Korisnik uz svaki artikl može unijeti cijenu.
11. Korisnik može proslijediti kreiranu listu elektroničkom poštom na željenu adresu.

Dodatna literatura

- ◆ V.Bosilj, A. Kovačić. Upravljanje poslovnim procesima. Sinergija, 2004.
- ◆ Car, Željka. Utjecaj implementacije sustava za podršku poslovnom procesu na preoblikovanje procesa. BUSINESS PROCESS CONFERENCE 2004, Zagreb, Hrvatska, 20-21. 10. 2004. (u repozitoriju na stranicama predmeta)
- ◆ Car, Željka; Belani, Hrvoje; Carić, Antun. A Conceptual Framework for Business Process Engineering: A Case Study. 2006 International Conference on Software Engineering Advances, Los Alamitos, CA, IEEE Computer Society Press, 2006. (u repozitoriju na stranicama predmeta)
- ◆ Gauta, Luka. Model poslovnog procesa razvoja arhive dokumentacijskih jedinica. Diplomski rad, FER, 2007.
- ◆ Bisera Bujić. Bujić. Definiranje programskih zahtjeva zasnovano na oblikovanju modela novih poslovnih procesa, 2011., magistarski rad, Fakultet elektrotehnike i računarstva, Zagreb