

Projektiranje informacijskih sustava

SDLC faza dizajna – dizajn korisničkog
sučelja

Ak. god. 2011/2012

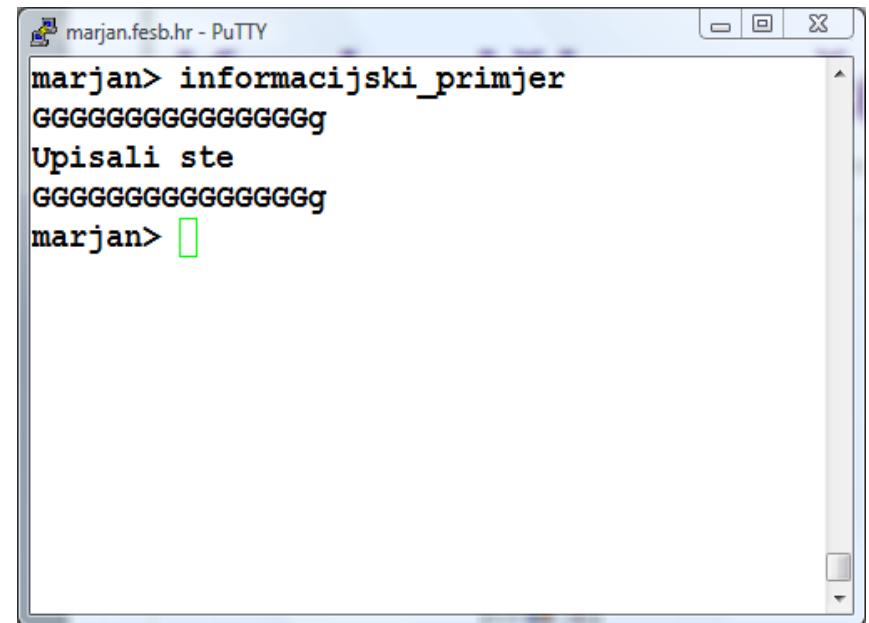
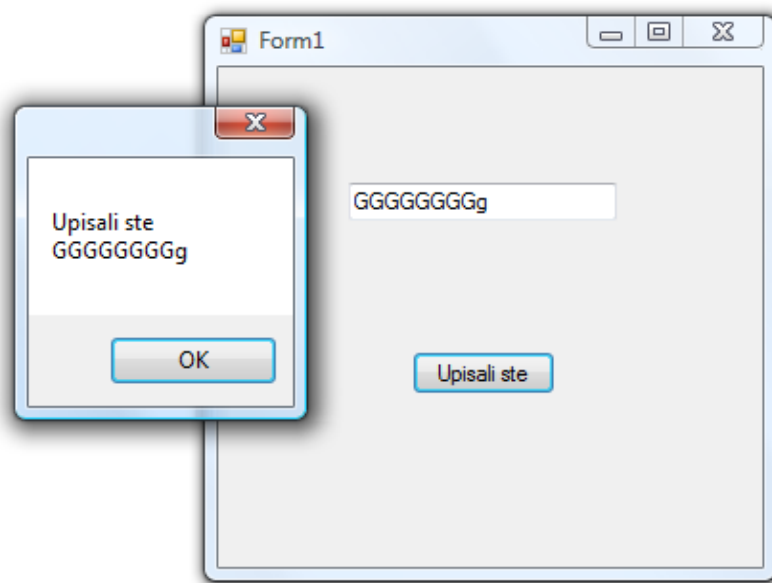
Sučelje

- Dizajn sučelja definira interakcije sustava sa vanjskim entitetima.
- Razlikujemo korisničko sučelje i sistemsko sučelje sustava.
- Korisničko sučelje je dio sustava preko kojeg korisnik komunicira sa sustavom.
- Sistemsko sučelje je dio sustava preko kojeg sustava komunicira sa drugim sustavim (npr. sustav treba eksportirati podatke za neki drugi sustav).

Korisničko sučelje

- Osnovna podjela na dva tipa korisničkog sučelja:
 - Grafičko korisničko sučelje (*Graphical user interface* (GUI)) koristi prozore, ulazni uređaj miša, izbornike itd. (WIMP – window, icon, menu, pointing device)
 - Tekstualno sučelje (*Text-based interface*).
- Danas se najčešće koristi grafičko sučelje.

Korisničko sučelje



[Primjer](#)

Korisničko sučelje

- Cilj dizajna korisničkog sučelja je dobiti sučelje koje je lako za korištenje i nije “baš previše ružno”, a u isto vrijeme omogućava korisniku obavljanje određenog posla sa minimalno napora i vremena. Npr. pokušajmo u Windows Exploreru kopirati datoteku sa jednog diska na drugi.
- Glavni problem prilikom dizajna korisničkog sučelja je prevelika količina informacija koje sučelje treba pružiti korisniku, a premalo mjesta za smještanje tih informacija ([Primjer](#)). Najčešće je potrebno koristiti više ekrana, formi, stranica (izvještaja) da bismo prikazali sve informacije što korisniku komplicira sučelje ([Primjer](#)).

Dizajn korisničkog sučelja

- Dizajn korisničkog sučelja je izuzetno važan za funkcionalnost aplikacije.
- Unaprijeđenjem razvoja korisničkih sučelja bavi se područje HCI (Human Computer Interaction).
- U ovom kolegiju ćemo samo ukratko navesti osnovne principe i smjernice razvoja korisničkog sučelja.

Osnovni principi kod dizajniranja korisničkog sučelja

- Osnovni principi (*design guidelines* ili *usability guidelines*) (dobiveni heuristički) koji se koriste pri dizajnu korisničkog sučelja (i navigacije i ulaza i izlaza) su:
 1. Razmještaj (*layout*)
 2. Svijest o sadržaju (*content awareness*)
 3. Estetika
 4. Iskustvo korisnika (*user experience*)
 5. Konzistentnost
 6. Minimiziranje korisnikovog truda (*minimize user effort*)

Razmještaj

- Razmještaj određuje organiziranje određenih područja određenih funkcionalnosti na ekranu, formi, izvještaju. Informacija korisniku može biti prezentirana u različitim područjima. Ta područja se trebaju koristiti konzistentno kroz aplikaciju.
- Većina softvera danas koristi tzv. Windows i Macintosh standardni pristup razmještaju na ekranu, pri čemu je ekran podijeljen u tri područja: područje navigacije na vrhu koje korisniku pruža funkcionalnost navigacije kroz sustav, područje statusa na dnu koje sadrži statusne informacije o korisnikovim aktivnostima i radna površina na sredini koja je najveća i služi za unos informacija ili prikaz. [Primjer](#)

Razmještaj



Razmještaj

- Vrlo često se (osobito kod Web aplikacija) sučelje dijeli na više od tri područja.
- Uobičajeno je postojanje više od jednog navigacijskog područja, ali može biti i više područja za unos i prikaz informacija.
- Područja i informacije unutar područja trebaju imati intuitivni tijek od jednog područja do drugog, tj. od jedne informacije do druge. U zapadnoj civilizaciji korisnici percipiraju informacije s lijeva na desno i odozgo prema dolje pa je uobičajeno korištenje ili horizontalnog tijeka informacija s lijeva na desno ili vertikalnog tijeka informacija odozgo prema dole.

Vertikalni i horizontalni razmještaj

Patient Information

Patient Name
First Name:
Last Name:

Address:
Street:
City:
State/Province:
Zip Code/Postal Code:
Home Phone:
Office Phone:
Cell Phone:

Referring Doctor:
First Name:
Last Name:
Street:
City:
State/Province:
Zip Code/Postal Code:
Office Phone:

(a) Vertical flow

Patient Information

Patient Name
First Name: Last Name:

Street: City: State/Province: Zip Code/Postal Code:

Home Phone: Office Phone: Cell Phone:

Referring Doctor
First Name: Last Name:

Street: City: State/Province: Zip Code/Postal Code:

Office Phone:

(b) Horizontal Flow

Svijest o sadržaju

- Svijest o sadržaju je svojstvo sučelja da korisnik percipira informacije koje sučelje sadrži s minimumom truda.
- Svi dijelovi sučelja trebaju pružati visoku svijest o sadržaju.
- Svijest o sadržaju se podiže sa:
 1. Svaki ekran, forma, izvještaj trebali bi imati naslove.
 2. Izbornici bi trebali pokazivati
 - Gdje ste
 - Odakle ste došli da bi stigli tu [Primjer](#)

Svijest o sadržaju

3. Trebalo bi biti jasno koja je informacija unutar kojeg područja (npr. oznaka područja).
4. Oznake polja i oznake područja trebalo bi odabrati pažljivo. Oznake trebaju biti što kraće, a da korisnik preko njih ipak ima svijest o sadržaju područja ili polja. Poljem nazivamo pojedinačni element podataka koji se unose ili prikazuju.
5. Format podataka u polju bi trebao biti točno određen. (Npr. datum formata 10/5/2009 – u nekim zemlja je to 5.10.2009., a u nekima 10.5.2009.)
6. Upotreba datuma i verzije da bi se pomoglo korisnicima (osobito na izvještajima).

Svijest o sadržaju

[Primjer](#)

Unos podataka kupca

Osobni podaci

Ime

Prezime

JMBG

Adresa

Ulica

Kućni broj

Grad

Podaci o računu

Broj računa

Banka

Datum otvaranja (DD/MM/YYYY)

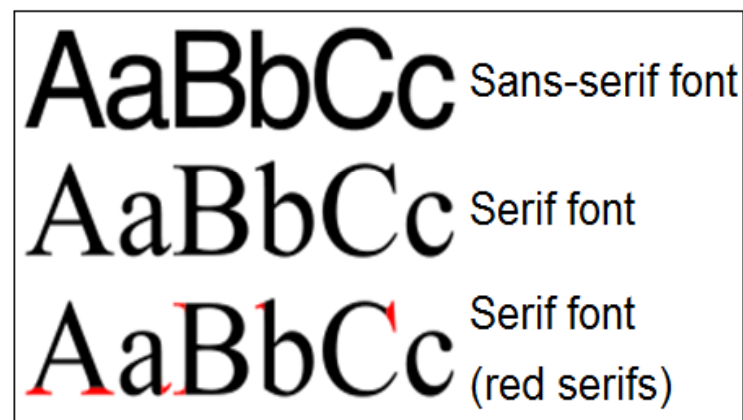
Spremi podatke o kupcu i računu

Estetika

- Estetika korisničkog sučelja se odnosi na to da sučelje bude “ugodnog” izgleda.
- Treba izbjegavati natrpavanje sadržajem, tako da je minimalistički pristup najčešće korišten.
- Početnici lakše i rađe koriste sučelja sa gustoćom informacija manjom od 50% (manje od 50% površine forme, ekrana, izvještaja je pokriveno sa informacijama).
- Iskusni korisnici daju prednost sučeljima sa gustoćom informacija i do 90% jer takva sučelja smanjuju količinu vremena koju korisnik treba utrošiti da dođe do svih informacija ili unese sve informacije.

Estetika

- Općenito pravilo za dizajn teksta je da sav tekst treba biti u istom tipu slova i slične veličine.
- Slova trebaju biti velika barem 8 piksela, ali preporučena minimalna veličina je 10 piksela.
- Promjena veličine ili tipa slova bi trebala ukazivati na neku promjenu u informaciji (npr. naslovi područja mogu biti veći).
- Tipovi slova poput Times New Roman (tzv. serif fonts) najvidljiviji su u tiskanim izvještajima, osobito ako su slova mala, dok su Arial, Helvetica (tzv. sans serif fonts) vidljivija na ekranu.
- Izbjegavati korištenje velikih slova (PREZIME -> Prezime).



Estetika

- Boje i uzorci se koriste rijetko i to sa nekom informacijskom svrhom i pažljivo da ne narušavaju kvalitetu korisničko sučelja.
- Oko 10% muške populacije su daltonisti pa informacije prenesene kroz boju ne mogu percipirati.
- Boja se može koristiti za naglašavanje bitne informacije, za naglašavanje naslova u odnosu na ostatak teksta, itd.
- Boje utječu i na raspoloženje, tako da crvena boja podiže agresivnost, dok plava boja smiruje.

Iskustvo korisnika

- Iskustvo korisnika se odnosi na dizajniranje sučelja u skladu sa korisnikovim iskustvom.
- Korisnicima početnicima je bitno koliko je vremena potrebno da se nauče služiti programom (*ease of learning*).
- Iskusni korisnici su zainteresiraniji za jednostavnost uporabe programa (*ease of use*).
- Često se ova dva svojstva sučelja postižu na isti način, ali ne uvijek.


Iskustvo korisnika

- Početnici rijetko koriste statusne informacije za razliku od iskusnih korisnika.
- Početnici očekuju da su sve funkcionalnosti predstavljene u izborniku jer to olakšava učenje korištenja aplikacije, dok iskusni korisnici očekuju da su samo često korištene funkcionalnosti predstavljene u izborniku jer to olakšava korištenje aplikacije.
- Ovisno o tipu korisnika (početnici ili iskusni korisnici) treba dizajnirati sučelje.
- Npr. za problem dohvaćanja svih funkcionalnosti ili samo često korištenih funkcionalnosti stari Microsoft Office ima rješenje u vidu “show me” funkcija.

Konzistentnost

- Konzistentnost je jedan od najvažnijih faktora dizajniranja jednostavnog sučelja jer omogućuje korisnicima da predvide što će se dogoditi.
- Kada je sučelje konzistentno korisnik u interakciji sa jednim dijelom sučelja (npr. jednom formom) nauči kako koristiti i druge dijelove sučelja jer su konzistentni.
- Konzistentnost se implementira na različitim razinama sučelja od navigacije, terminologije do izvještaja i formi.

Konzistentnost

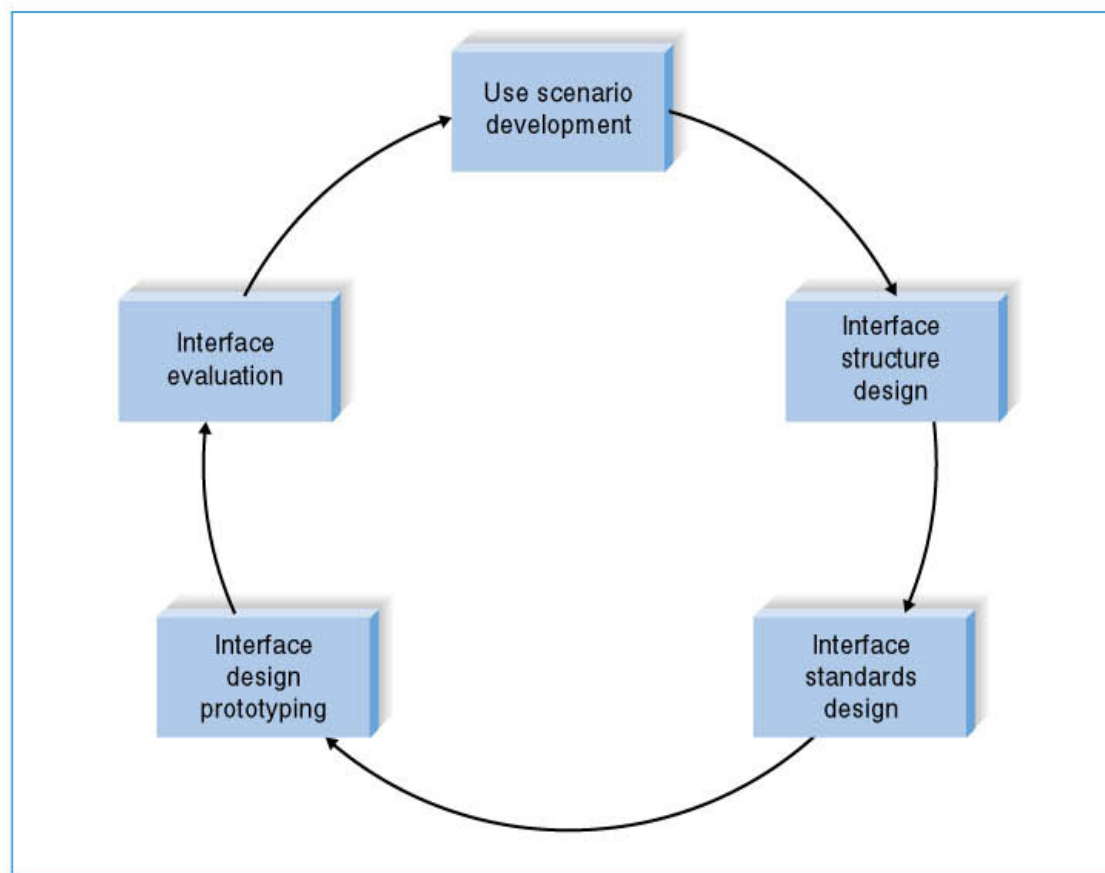
- Npr. korištenje iste ikone ili iste naredbe sa određenom funkcijom kroz cijeli sustav (npr. undo, ).
- Korištenje iste terminologije je također važno. Npr. nije poželjno da na jednoj formi piše klijent, a na drugoj kupac.
- Konzistentnost u izgledu formi i izvještaja olakšava korisniku njihovo korištenje. Naravno forme i izvještaji ne mogu biti identični, ali poželjno je da su razlike u dizajnu minimalne, a da se informacija prenese.

Minimiziranje korisnikovog truda

- Korisničko sučelje treba dizajnirati na način da se minimizira trud koji korisnik treba uložiti za obavljanje neke akcije.
- Trud u obavljanju akcije preko korisničkog sučelja se sastoji u korištenju ulaznih uređaja preko kojih korisnik komunicira sa aplikacijom, tj. smanjenje klika mišem i pritisaka tipki tipkovnice.
- Većina dizajner primjenjuje pravilo “tri klika” po kojem bi korisnici trebali biti u mogućnosti otići od glavnog menija ili početka sustava do informacije ili akcije koju žele u ne više od tri klika mišem ili tri pritiska na tipku.

Proces dizajna korisničkog sučelja

- Dizajna korisničkog sučelja je iterativni proces sa pet koraka:
 1. razvoj scenarija korištenja
 2. dizajn strukture sučelja
 3. dizajn standarda sučelja
 4. prototipiranje dizajna sučelja
 5. evaluacija sučelja
- Često se prilikom evaluacije postavljenog prototipa uvidi da sučelje nije adekvatno pa se postupak provodi iterativno do zadovoljavajućeg rješenja .



Razvoj scenarija korištenja

- Prvi korak u dizajnu sučelja je razvoj scenarija korištenja na osnovu DFD-ova i slučajeva korištenja iz faze analize.
- Scenarij korištenja je opći pregled koraka koji korisnik napravi da bi izvršio određeni zadatak. To je ustvari jedan uobičajeno korišteni put kroz slučaj korištenja.
- Izdvajaju se uobičajeni (najčešće korišteni) putovi korištenja aplikacije jer je za njih najbitnije ispoštovati smjernice za dizajn sučelja.
- Scenariji korištenja se zapisuju u tekstualnom obliku.

Dizajn strukture sučelja

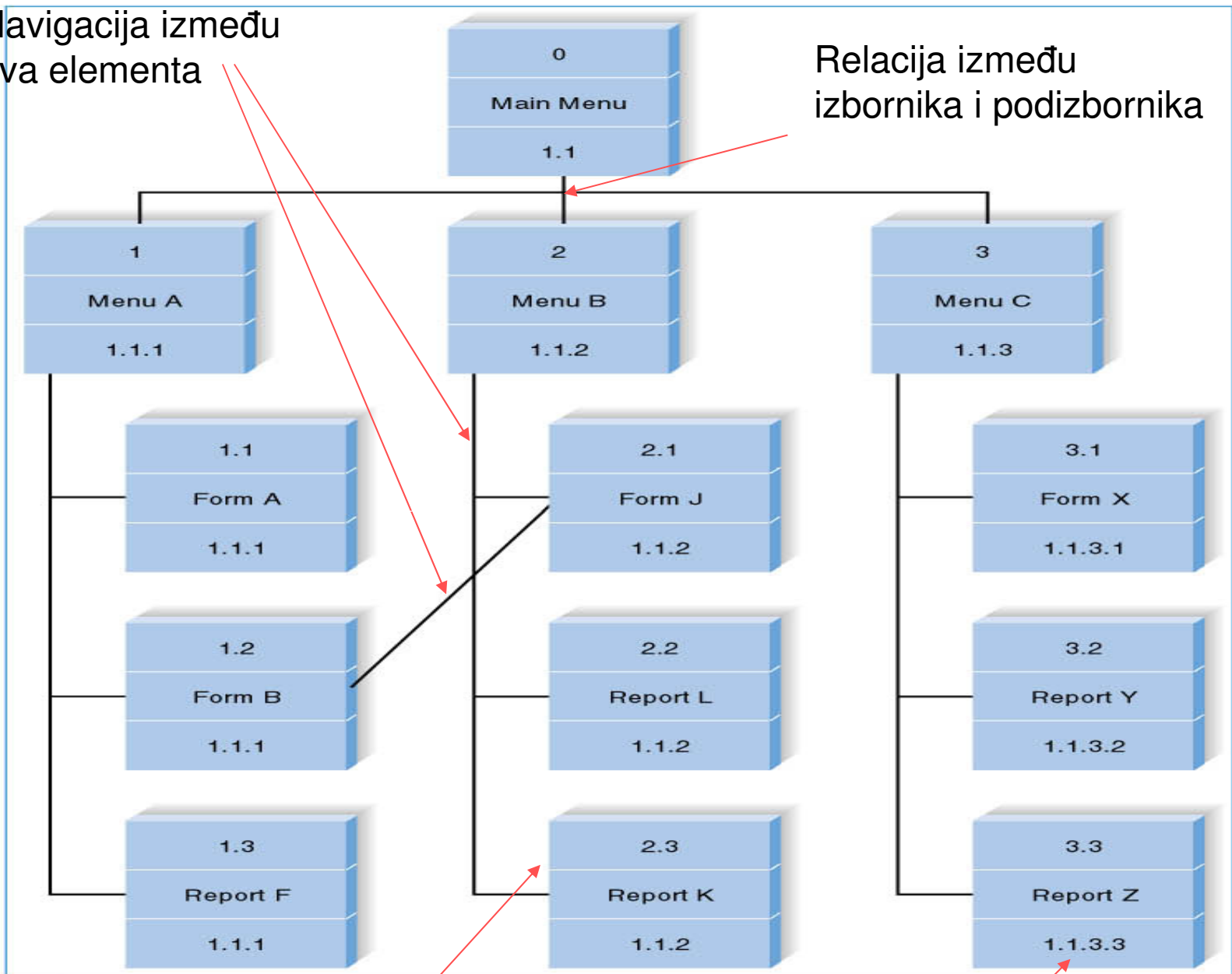
- Dizajn strukture sučelja definira osnovnu strukturu sučelja na način da definira sve osnovne komponente ili elemente sučelja (ekrane, forme i sl.) i način na koji su komponente sučelja međusobno povezane.
- Dijagram strukture sučelja (interface structure diagram – ISD) prikazuje sve ekrane, forme i izvještaje, njihovu povezanost te način na koji se korisnik kreće između pojedinih ekrana, formi i izvještaja.
- ISD dijagrami nalikuju DFD (kvadrati i linije), ali bez standardnih pravila.

Dizajn strukture sučelja

- Svaka komponenta sučelja (forma, ekran, izvještaj) se predstavlja kvadratom sa jedinstvenim identifikatorom (brojem) i jedinstvenim imenom.
- Obično dijagram ima strukturu stabla, s time da hijerarhija stabla ne znači da pojedina komponenta pripada nekoj komponenti na višoj razini ili da je njen dio, već označava mogućnost navigacije od jedne komponente do druge ili relaciju između izbornika i podizbornika.
- U ISD kvadratima se obično navodi i oznaka procesa iz DFD-a koji je vezan uz određeni element sučelja. Određeni procesi mogu biti vezani uz nekoliko elemenata sučelja.

Navigacija između
dva elementa

Relacija između
izbornika i podizbornika



Oznaka elementa sučelja

Oznaka proces iz DFD dijagrama²⁷

Dizajn standarda sučelja

- Standardi sučelja su temeljni dizajnerski elementi zajednički pojedinim ekranima, formama, i izvještajima unutar aplikacije. Npr. može se postaviti standarda izvještaja po kojem svaki izvještaj u gornjem desnom uglu mora imati datum, informacije moraju biti prezentirane u dvije kolone i sl.
- Identificiranje dizajnerskih elemenata se radi kroz:
 1. Metafore sučelja – to je koncept iz stvarnog svijeta koji se koristi kao model u sučelju. Npr. checkbox element je koncept iz stvarnog svijeta unosa podataka prenesen u korisničko sučelje. Jednostavne metafore ne treba opisivati. Primjeri metafora sučelja su i Desktop, kolica za kupnju itd.

Dizajn standarda sučelja

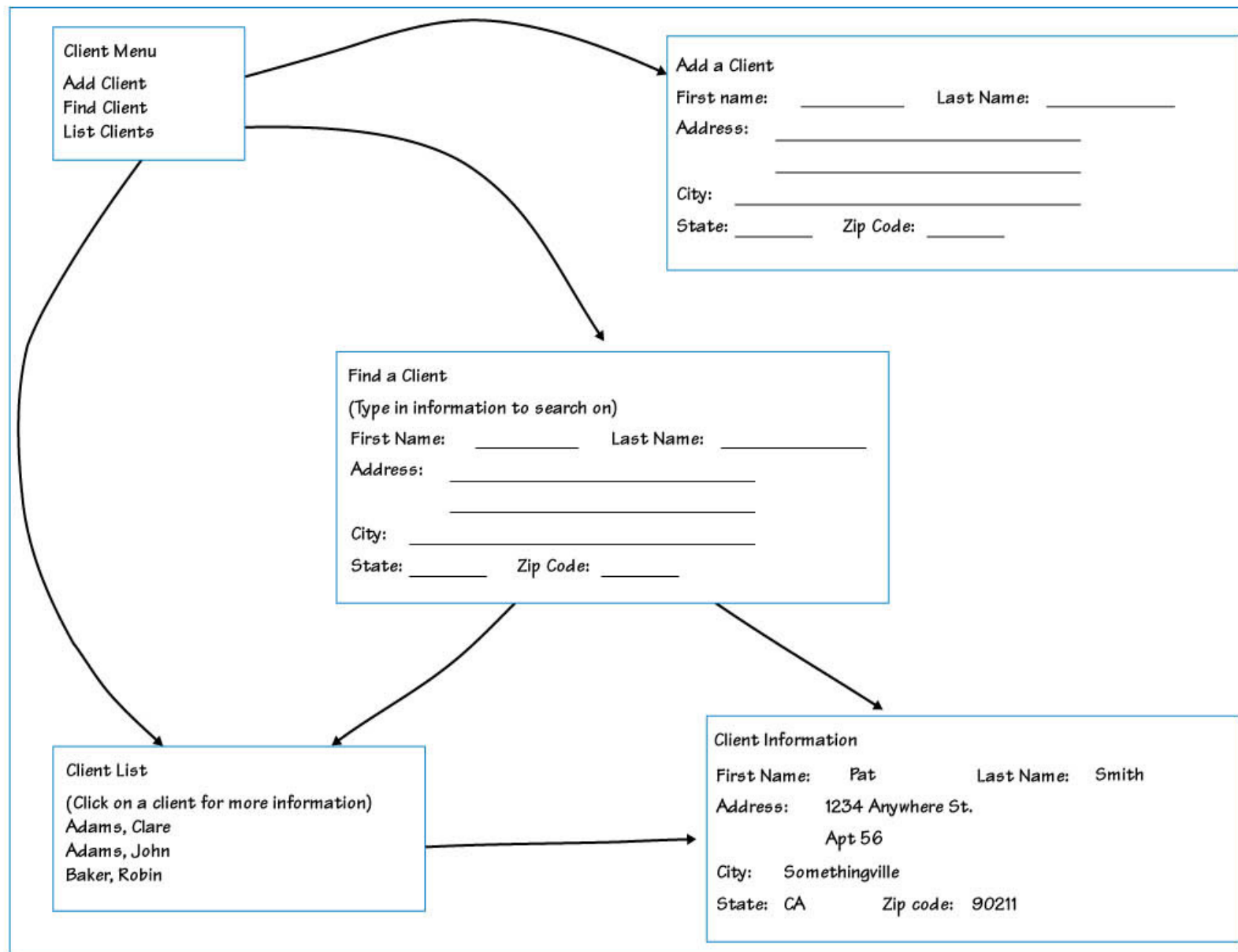
2. Predlošci sučelja (interface template) definiraju općeniti izgled svih ekrana i izvještaja, poput definiranja gdje se nalazi navigacija, gdje statusni podaci, koje boje će se koristiti u sučelju i sl. (npr. kod web aplikacije dio predloška je CSS koji se uključuje u sve stranice, ili u ASP.NET master page, Joomla predlošci).
3. Objekti sučelja su osnovni građevni blokovi sučelja. Izvode se i iz metafora sučelja tako da se metafore prevode u instance definiranog imena (npr. [lista kolegija po semestru studija](#)).
4. Akcije sučelja su definirane preko izbornika ili na neke druge načine (npr. akcija “Spremi podatke o kupcu i računu”).
5. Ikone sučelja – objekti sučelja, akcije sučelja kao i stanje objekta ili akcije može biti predstavljeno ikonom.

Prototipiranje dizajna sučelja

- Prototip dizajna je nacrt izgleda ekrana (formi, izvještaja,...) ili simulacija ekrana, formi ili izvještaja sučelja.
- Prototip dizajna se razvija za svaku komponentu sučelja (svaku formu, izvještaj, ...).
- Najčešće metode prototipiranja sučelja su:
 1. Storyboard – najčešće se koristi papir za prikaz izgleda sučelja, ali postoji i elektronički storyboard. Elementi sučelja se nacrtaju i onda se kao kod crtanih filmova slažu slike sučelja da bi se pokazalo što se dešava u navigacija ili drugim aktivnostima korisnika. (npr. [stpsoft StoryboardingTM](#))

Prototipiranje dizajna sučelja

2. HTML prototip – najčešće se koristi. Sučelje se prikazuje kao HTML stranice. Ovakav prototip omogućava korisniku “stvarno” testiranje sučelja jer prototipovi sadrže i navigaciju i polja za unos i botune. Jedino što nije implementirana funkcionalnost iza sučelja.
3. Jezični prototip (language prototype) – prototip sučelja se razvija slično kao i HTML prototip, ali u onom programskom jeziku u kojem će se razviti i “pravo” sučelje. Razvoj traje duže no kod HTML prototipova, ali korisnik dobiva točan izgled budućeg sučelja.
- Storyboard je najjeftinija i najbrža metoda, ali korisnik nema mogućnost “testiranja” sučelja, dok HTML i jezični prototipovi omogućavaju korisniku testiranje sučelja.



Storyboard primjer

Evaluacija sučelja

- Evaluacija sučelja identificira moguća poboljšanja predloženog dizajna sučelja.
- Metode evaluacije korisničkog sučelja:
 1. Heuristička evaluacija – članovi tima uspoređuju razvijeni dizajn sa principima dizajna.
 2. Walk-through evaluacija - tim simulira kretanje kroz sučelje korisniku (korištenjem storyboarda ili HTML ili jezičnih prototipova) koji identificira moguća poboljšanja sučelja.
 3. Interaktivna evaluacija - korisnici sami iskušavaju sustav (za to su potrebni ili HTML ili jezični prototipovi ili gotovi sustav).
 4. Formalno testiranje uporabljivosti – je vrlo skupo jer se provodi u uvjetima specijalnog laboratorijskog testiranja.

Korisničko sučelje

- Korisničko sučelje uključuje tri osnovna dijela:
 1. Mehanizam navigacije (*navigation mechanism*) određuje na koji način korisnik postavlja zahtjeve prema sustavu tj. kako korisnik kaže sustavu što da radi (npr. korisnik klikne na izbornik da dobije prozor za unos informacija o kupcu i onda klikne na botun da bi spremio te informacije u bazu)
 2. Mehanizam ulaza (*input mechanism*) definira način na koji će korisnik unijeti informaciju u sustav (npr. postoji forma sa 5 polja preko koje korisnik unosi podatke o kupcu)
 3. Mehanizam izlaza (*output mechanism*) definira način na koji sustav isporučuje informaciju korisnicima (*printani izvještaj, graf,...*)

Dizajn navigacije

- Komponente navigacije sučelja omogućavaju korisniku unos naredbi za navigaciju kroz sustav i za izvođenje aktivnosti na unosu ili dohvat informacija iz sustava.
- Komponente navigacije su i poruke koje sustav generira korisniku.
- Dobra navigacija korisniku pojednostavljuje korištenje sustava.

Osnovni principi dizajna navigacije

- Prilikom dizajna navigacije treba pretpostaviti da korisnici:
 - Nisu pročitali upute
 - Nisu pohađali edukaciju
 - Nemaju vanjsku pomoć pri ruci
- Sve kontrole bi trebale biti jasne i razumljive i smještene na intuitivnu lokaciju na ekranu.
- Idealne kontrole bi trebale predvidjeti korisnikov cilj i olakšati mu dolaženje do tog cilja. Npr. većina instalacijskih paketa funkcionira po principu da nudi neku tipičnu instalaciju sa navigacijom “Next” koja korisniku olakšava dolaženje do cilja, a to je instalirana aplikacija.

Osnovni principi dizajna navigacije

1. Spriječiti pogreške korisnika:

- Limitirati izbore
- Nikad ne prikazati naredbe koje se ne mogu koristiti (ili ih “posiviti”)
- Potvrđivati naredbe koju su komplicirane ili je nemoguće napraviti “undo”

2. Omogućiti “oporavaka” od grešaka – bez obzira na dizajn, korisnik će napraviti neku grešku. Dizajn navigacije treba omogućiti “oporavak” od greške. Npr. undo funkcionalnost, ili Back botun kod web aplikacije.

Osnovni principi dizajna navigacije

3. Upotrebljavati konzistentan gramatički poredak (use consistent grammar order) – dva su osnovna gramatička poretka kod dizajna sučelja. To su objekt-akcija poredak i akcija-objekt poredak. Češće se koristi objekt-akcija poredak. Većina naredbi očekuje od korisnika specifikaciju objekta na kojem se izvodi određena akcija, pa zatim specifikaciju akcije koja će se izvršiti na objektu. Sučelje može ograničiti korisnika na način da za izvođenje komande treba najprije definirati objekt na kojem se izvodi akcija, pa onda izvođenje akcije. Npr. kopiranje teksta u Wordu.

Tipovi navigacijskih kontrola

- Osnovni hardverski uređaji za rad sa sučeljem su tipkovnica i miš, i u novije vrijeme sustavi za prepoznavanje govora.
- Definiranje korisničkih naredbi (tj. navigacija kroz aplikaciju) može se implementirati na tri načina:

1. Jezik

- Korištenjem naredbenog jezika korisnik naredbu unosi posebnim jezikom (Linux shell, SQL,...). Ovaj se pristup danas rijetko koristi osim kada veliki broj mogućih naredbi i njihovih kombinacija čini praktički nemogućim stvaranje izbornika (SQL).
- Kada se sučelje oslanja na prirodni jezik, koristi interpreter prirodnog jezika (primjer Office Assistant) da iz korisnikove naredbe zaključi koju komandu korisnik želi izvršiti.

Tipovi navigacijskih kontrola

2. Izbornici - najčešći tip navigacijske kontrole. Izbornik korisniku pruža listu mogućih komandi od kojih korisnik izabire. Lakši su za korištenje no komandni jezici. Podizbornici su zatvoreni dok ih korisnik ne odabere tako da ne pružaju uvid u funkcije koje nude pa je bolje koristiti prostrani plitki izbornik. Bilo bi dobro da izbornik/podizbornik ima do 8 elemenata. Međutim to nije uvijek moguće pa se to kompenzira grupiranjem elemenata i odvajanjem npr. horizontalnom linijom kako bi se korisnik što lakše snašao. Korištenje short cut ili hot key tipki je također poželjno. Grupiranje sličnih elemenata izbornika može ići ili po objektu sučelja (npr. klijenti, transakcije,...) ili akciji (novo, izmjena,...).

Tipovi navigacijskih kontrola

3. Direktna manipulacija omogućava korisniku direktno pozivanje naredbe na objektu sučelja. Npr. u Windows Exploreru kada povlačimo datoteku sa jednog mjesta na drugo direktno na datoteci izvršavamo naredbu copy, move, delete. Nedostatak direktne manipulacije je što nije uvijek intuitivna tj. korisnik ne može intuitivno zaključiti kojom manipulacijom izvršiti neku naredbu.

Poruke

- Sustav porukama odgovara korisniku i obavještava ga o statusu njegovih interakcija sa sustavom preko sučelja.
- Postoji 5 tipova poruka:
 1. poruke greške (*error message*) koje obavještavaju korisnika da je pokušao napraviti neku akciju koju sustav ne prihvaća (“You can not access system directory”),
 2. poruke koje zahtjevaju potvrdu korisnik (*confirmation message*) (“Are you sure you want to delete file?”),
 3. poruke kojima sustav obavještava korisnika da je napravio neku akciju (*acknowledgment messages*) (“Your change has been saved”)
 4. poruke pomoći (*help message*),
 5. poruke o kašnjenju (*delay message*) se koriste kada neka aktivnost sustava traje određeno vrijeme (obično više od 7 sekundi).

Poruke

- Poruke bi trebale zahtjevati potvrdu od korisnika da je poruku vidio, osim poruka o kašnjenju.
- Poruke su u pravilu tekstualne, ali mogu biti i u drugom obliku (npr. pješčani sat na windowsima kao *delay* poruka).
- Poruke trebaju biti jasne, sažete i kompletne (kontradiktorni zahtjevi).
- Trebaju biti gramatički točne, bez žargona i lokalizama.
- Izbjegavanje negativnosti (“Are you sure you do not want to continue?” -> “Do you want to quit?”) i humora.

Dizajn ulaza

- Mehanizam ulaza omogućava unos podataka u sustav.
- Dizajn ulaza uključuje dizajn ekrana, formi preko kojih se informacije unose u sustav.
- Cilj dizajna ulaza je jednostavno i lako uhvatiti točnu informaciju za sustav.
- Dva osnovna formata unosa podataka u računalni sustav su:
 - Online procesiranje
 - Pozadinsko (batch) procesiranje

Dizajn ulaza

- Osnovni principi dizajna ulaza su:
 1. Korišćenje online i pozadinskog procesiranja u skladu sa situacijom
 2. Prikupljanje podataka što bliže izvoru
 3. Minimiziranje korišćenja tipkovnice

Dizajn ulaza

- Online procesiranje trenutačno prema podatke u odgovarajuću bazu podataka. Koristi se kada postoji potreba za prikupljanje podataka u realnom vremenu.
- Pozadinsko procesiranje skuplja ulazne podatke tijekom vremena i unosi ih u sustav kao kolekciju podataka.

Prikupljanje podataka na izvoru

- Prikupljanje podataka u elektroničkom obliku na samom izvoru podataka ili što bliže izvoru podataka jedan je od osnovnih principa dizajna ulaza. (npr. papirnata prijavnica koju u sustav unosi djelatnica referade ili elektronička prijava ispita).
- Automatizacija unosa podataka sa samog izvora može biti napravljena korištenjem sljedećih tehnologija:
 - Čitači bar kôda
 - Prepoznavanje optičkih znakova
 - Čitači magnetskih vrpce
 - Pametne kartice (smart cards)
 - RFID (radio frequency identification tags)

Prikupljanje podataka na izvoru

- Primjer sustava za identifikaciju korisnika na fakultetu sa smart karticama.
- Prikupljanje podataka na samo izvoru reducira vrijeme procesiranja, smanjuje trošak i smanjuje mogućnost pogreške.

Minimiziranje pritisaka na tipke

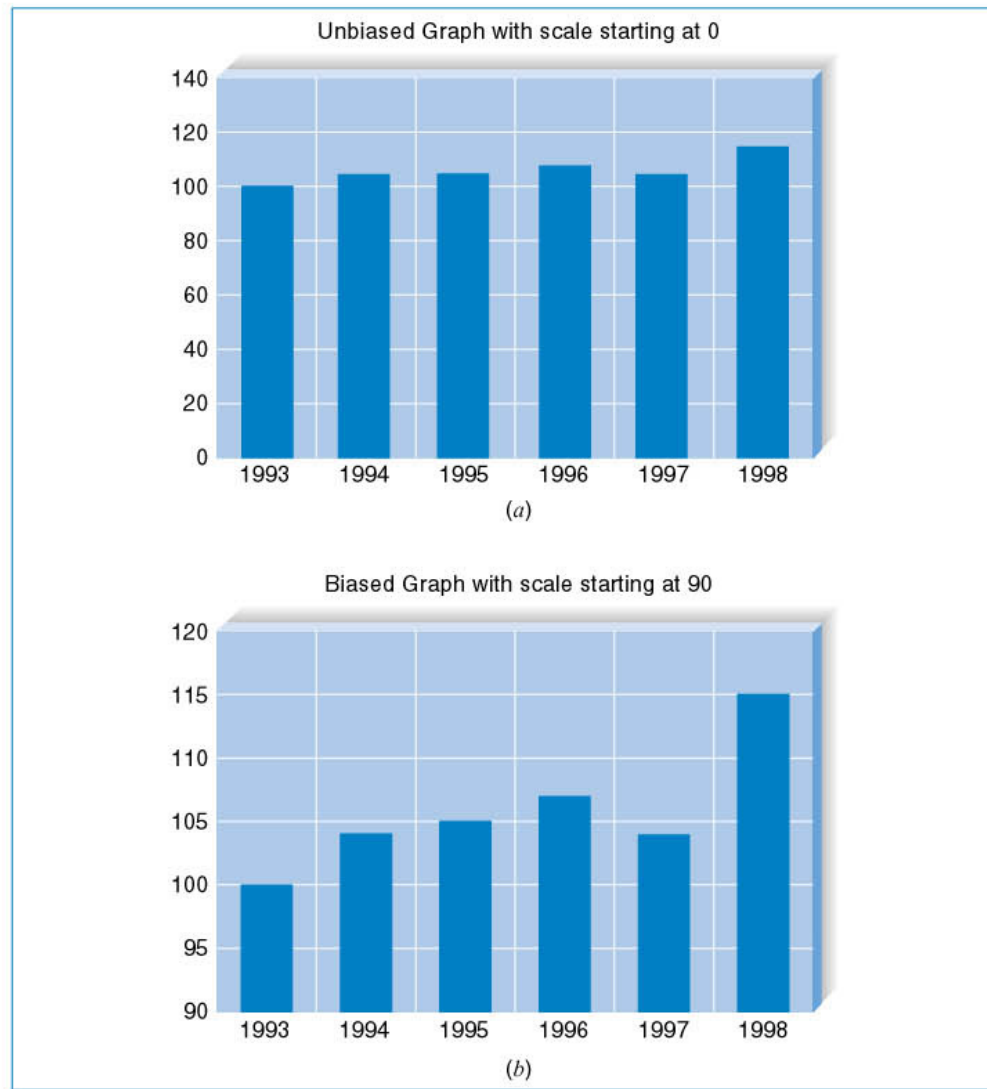
- Još jedan važan princip dizajna ulaza je minimiziranje pritisaka na tipke prilikom unošenja podataka.
- Nikad ne pitati za informaciju koja može biti dobivena drugim putem
 - Pregledom tablice
 - Padajuće liste
 - Standardne vrijednosti

Dizajn izlaza

- Izlazi sustava su izvještaji koje sustav generira bilo u papirnatom obliku bilo na ekranu bilo u nekom drugom mediju.
- Cilj mehanizma izlaza je prezentirati informacije korisniku na način da ih korisnik razumije uz što manje truda.
- Tipovi izvještaja – pregledni izvještaj, detaljni izvještaj, grafički izvještaj,...

Dizajn izlaza

- Osnovni principi dizajna izlaza su:
 - Razumijevanje upotrebe izvještaja
 - Referencirajući ili potpuni (cover-to-cover)
 - U realnom vremenu ili batch izvještaji
 - Racionalno korištenje informacija
 - Samo potrebne informacije, ništa više
 - Minimiziranje pristranosti (npr. izvještaj sortirani po abecedi stavlja naglasak na podatke s obzirom na abecedu)



Primjer pristranosti