

Uporabniški vmesniki:

1. Uvod:

- Interaktivno risanje
- Računalniško načrtovanje CAD/CAM
- Nadzor in krmiljenje procesov
- Navidezna resničnost
- Simulacije in animacije
- Pisarne in elektronsko zalozništvo
- Igre
- ...

1. Senzorni spomin – sprejema ogromno količino informacij, ki se jih ne zavedamo.
2. Kratkotrajni spomin hrani 5-9 dejstev približno 30 sekund.
3. Dolgotrajni spomin, informacije so kodirane

Look and feel – look = izgled, feel = uporaba

Zaznavanje in razumevanje -> Mentalni model ->Principi in navodila za gradnjo UV.

Dve vrsti UV:

1. GUV – grafični uporabniški vmesniki
2. OUV – objektno usmerjeni uporabniški vmesniki

Dve vrsti konceptov:

1. Akcija -> Objekt (ukazna vrstica) najprej izberemo akcijo (napišemo) in potem to izvedemo nad objektom.
2. Objekt -> Akcija (grafični vmesnik), najprej izberemo objekt in nato nad njim izvedemo akcijo.

Ni možno zgraditi enega samega optimalnega uporabniškega vmesnika za vse zahteve uporabnika.

Ni možno zgraditi enega samega optimalnega uporabniškega vmesnika za vse uporabnike.

Modeli UV:

1. Model uporabnika (konceptualni, mentalni)
2. Model programerja

### 3. Model načrtovalca (dizajnerja)

<b>Uporabnik:</b> <ul style="list-style-type: none"><li>- Izkušnje realnega sveta</li><li>- Naloge</li><li>- Procesi</li><li>- Orodja</li></ul>	<b>Načrtovalec:</b> <ul style="list-style-type: none"><li>- Upor. Koncept. Mode</li><li>- Programerjev model</li><li>- Principi načrtovanja uporabniških vmesnikov in navodila</li></ul>	<b>Programer:</b> <ul style="list-style-type: none"><li>- Platforma</li><li>- Operacijski sistem</li><li>- Lupina</li><li>- Razvojna orodja</li></ul>
---	--	---

-

FEVH = file, edit, view help, WOSH = okno, ime objekta selected (izbran objekt), help

Model uporabnika:

Je predstavitev tega kako uporabniki razumejo in komunicirajo s sistemom. Mentalni modeli obsegajo interakcijo med uporabniki in računalniki in tako formirajo temelj za principe in navodila pri načrtovanju vmesnikov.

Metafore – prenašanje lastnosti objektov ali akcij realnega sveta na druge objekte z namenom sugeriranja podobnosti ali analogije med njimi.

Odzivi uporabnika pri slabo načrtovanem UV:

- Zbeganost
- Panika
- Dolgčas
- Frustracija
- Neizkoriščenost sistema

Model Načrtovalca:

Načrtovalec sprejema želje, ideje in potrebe uporabnikov, dodal znanje in materiale, ki so na razpolago programerju in načrtuje programski produkt, ki ga je možno zgraditi in ga uporabnika lahko uporablja.

Koncept direktne manipulacije:

- Sistem je predstavljen kot podaljšek realnega sveta
- Zvezna vidljivost objektov in akcij (WYSIWYG – predstavitev realnosti, ki jo je lahko spreminjati, množica alternativ, ki jih uporabnik lahko doseže direktno ali indirektno).
- Uporablja »prepoznavanje«
- Hitri in vidni rezultati akcij
- Povleci – spusti
- Dano akcijo je možno izničiti

Slabe lastnosti:

- Težko načrtovanje
- Delovna domena je sedajnost
- Ikone niso vedno razumljive

### **Grafični uporabniški vmesniki GUI:**

Aplikativno usmerjeni uporabniški vmesniki.

Kaj je grafični uporabniški vmesnik?

Je grafična predstavitev in interakcija s programi, podatki in objekti na računalniškem zaslonu.

WIMP: okna – pogled v aplikacije, ikone – pomanjšana okna, menuji, kazalec

Lastnosti:

- Zaslon z visoko resolucijo in pripravo za kazanje (miška).
- Tekst in grafika prikazana tako kot iztiskano
- Sledi interaktivni paradigmi objekt->akcija
- Ikone in objekti (okna) so prikazani vizualno
- Omogoča direktno manipulacijo ikon in objektov (oken)
- Ima menuje in dialoge za komunikacijo
- Rezultati akcij so vizualni
- Obstoj grafični prikaz sistemskih in uporabniških akcij in načinov (menuji, palete)
- Omogoča spremembo lastnosti vmesnika in interakcij

- Omogoča vhod preko tipkovnice, grafičnih gradnikov in drugih naprav

### **Objektno usmerjeni uporabniški vmesniki OUV:**

Kaj je objektno usmerjen uporabniški vmesnik?

Uporabnik se ukvarja z objekti, ki predstavljajo realni svet in ni obremenjen z arhitekturo računalnika ali kako uporabljati aplikacije in objekte.

- Je simulacija realnosti oziroma realizacija objektov realnega sveta
- Vmesnik ponuja množico objektov
- Uporabnik manipulira z objekti
- Okno je pogled v objekt
- Ikona je objekt
- Klik po ikoni pomeni odpreti objekt

Principi in navodila (postopek načrtovanja grafičnih uporabniških vmesnikov)

- Razumevanje uporabnika
- Strojna oprema in grafične sposobnosti sistema
- Razumevanje željene aplikacije
- Načrtovanje oken, menijev
- Interakcija s sistemom
- Izbor grafičnih gradnikov
- Izbor barv in ikon
- Povratna informacija: obvestila, dialogi, navodila, pomoč
- Testiranje (beta verzija, uporabnost)

### **X11: X okenski sistem:**

Cilj : poganjati aplikacije na oddaljenih sistemih

- Je neodvisen od operacijskega sistema (UNIX, MS-DOS,...)
- Je grafično okolje večjih firm (Apollo, Apple, DEC, HP, IBM, Sony, Sun,...)
- Temelji na sistemu okna (window) in konceptu odjemalec/strežnik
- Odjemalec in strežnik sta lahko na različnih strojih
- Uporablja ga več aplikacij istočasno

Sestavni deli:

- Xlib – grafična knjižnjica
- X mrežni protokol
- Xt orodja – dodatna knjižnjica za podobe
- Upravniki oken (uwm)

Prikazovalnik (display) je abstrakcija, ki predstavlja vohne in izhodne naprave (zaslon, tipkovnica, miška), enega uporabnika

Strežnik kontrolira prikazovalnik:

- Nadzoruje dostop do prikazovalnika za odjemalce
- Pošilja sporočila po mreži
- Sledi zahtevam odjemalcev in ažurira okna
- Dvodimenzionalno risanje
- Omogoča distribuirano procesiranje
- Multiprocesiranje
- Sprejema vhodne dogodke in jih posreduje odjemalcem

Odjemalci imajo dostop do prikazovalnika:

- Tečejo simultano – več oken istočasno, aplikacije so neodvisne
- Uporabljajo mrežni protokol preko Xlib
- Med seboj komunicirajo preko strežnika – delijo si podatke
- Vsak posreduje zahtevo strežniku za kreiranje vsaj enega okna, ki je naslednik osnovnega okna – zaslon (root window)

## **Motif:**

Lastnosti:

Xt je objektno usmerjen. Uporabnik kreira, upravlja in razkroji podobe. Interna implementacija podob se lahko spreminja. Uporabnik vidi aplikacijo abstraktno in splošno (lažje načrtovanje, manj napak, daljša življenska doba).

Xm določa druge izvedene razrede podob.

Lastnosti aplikacije:

- Lastnosti podob – med tekom Xt naloži podatke s sistemskih in specifičnih uporabniških datotek ter zgradi bazo lastnosti (resource database).
  - o Sistemsko določene lastnosti
  - o Konfigurabilne lastnosti
  - o Nastavitve lastnosti med tekom
- Odzivne funkcije – določene so v kodi aplikacije (callback resources).
- Interne funkcije – specifična funkcionalnost podob. Interne funkcije, akcije (actions), zagotavljajo npr. ukaze za urejanje teksta. Dogodki in akcije v prevajalni tabeli (translation table).

Razredi podob v Xt in Motif:

- Lupine (shells) – skrbijo za komunikacijo z upraviteljem oken (mwm), imajo enega naslednika z isto geometrijo, vsaka podoba, ki je naslednik osnovnega okna mora biti ovita v lupino (lupina za prvo podobo aplikacije, lupina za dialoge, lupina za menuje).
- Upravniki (managers) – imajo lahko več naslednikov, skrbijo za geometrijo svojih naslednikov in jim posredujejo podatke.
- Osnovni gradniki (primitives) – nimajo naslednikov
- Omejeni osnovni gradniki (gadgets) – nimajo naslednikov, so identični osnovnim gradnikom a z omejenimi resursi. Barve in prikazovalno matriko dedujejo od svojega predhodnika, nadzor nad dogodki ima predhodnik.

Osnovni gradniki (primitives) – nimajo naslednikov:

- Puščica (ArrowButton) – gumb
- Oznaka (Label) – tekst, ikona

- PushButton – gumb
- CascadeButton – kaskadni gumb
- ToggleButton – stikalo
- DrawnButton – gumb z ikono
- Drsni (ScrollBars) – opazovanje omejenega okna, prikaz statičnega teksta ali slike
- Separator (Separator) – ločuje gradnike vmesnika
- Tekst (Text) – prikaz in editiranje teksta
- Tekst (TextField) – prikaz in editiranje teksta ene vrstice
- Lista (List) – polje tekstualnih zapisov z možnostjo izbire

Motif programski model:

- Inicializacija Xt orodij
- Kreiranje in upravljanje podob – zgradimo objekte, povemo kako bodo zadeve/strukture izgledale. Upravljanje = razvrščanje, vidnost in nevidnost podob. Če je ne upravljamo postane nevidna.
- Konfiguracija podob s postavljanjem njihovih resursov – dogovarjanje o tem, kako so podobe med seboj razvrščene. Razvrščajo lego svojih naslednikov
- Nastavitev odzivnih funkcij glede na opravila aplikacije
- Realizacija podob in predaja nadzora nad aplikacijo Xt notranjim funkcijam – med realizacijo predamo nadzor strežniku, ki začne prikazovati na zaslon

Zgradimo lupino, povezava z X strežnikom, zgradimo bazo podatkov – preberemo vse lastnosti iz datoteke, tolmačim argumente ukazne vrstice, uporaba rezervnih lastnosti.

Lupine (shells) – skrbijo za komunikacijo z upraviteljem oken (mwm), imajo enega naslednika z isto geometrijo, vsaka podoba, ki je naslednik osnovnega okna mora biti ovita v lupino. Lupina je nevidna in nima dimenzij.

- Inicializacija in kreiranje lupine prve podobe aplikacije – ApplicationShell (je predhodnik vsem podobam v aplikaciji).
- Lupina za dialoge – DialogShell (ima povezavo z mwm)
- Lupina za menije – MenuShell – (nima povezave z mwm) Dialogi in meniji so sekundarna ali začasna okna aplikacije.

Upravniki (managers) – imajo lahko več naslednikov, skrbijo za geometrijo svojih naslednikov in jim posredujejo podatke

- Okno z drsniki (ScrolledWindow) – prvo okno aplikacije z drsniki
- Glavno okno (MainWindow) – prvo okno aplikacije
- Okno z vertikalno nastavljivimi nasledniki (PanedWindow) – nasledniki so iste širine
- Okvir (Frame) – obkroža naslednika
- Vrstica-kolona (RowColumn) – nasledniki ležijo v vrsticah ali kolonah (implementira menu, menuje ali polje gumbov)
- MenuBar – vrstični menu
- PulldownMenu – izvlečni menu
- CascadeMenu – kaskadni menu
- PopupMenu – dvižni menu
- OptionMenu – opcijski menu

Razvoj aplikacije v Motifu:

1. Vnesi zahtevane datoteke s prototipi
2. Inicializacija
3. Kreiranje hierarhije podob (najprej kreiraj nato upravlja)
4. Definicija odzivnih funkcij
5. Prikaz podob
6. Zanka dogodkov

Motif lastnosti aplikacije: (prioriteta pada od zgoraj navzdol)

1. Med gradnjo podob v programu
2. Z argumenti ukazne vrstice
3. V datoteki
  - a. Domači direktorij
  - b. Kjerkoli (XAPPLRESDIR/<ime razreda>)



c. Na sistemsko določenem direktoriju: (/usr/openwin/lib/X11/app-defaults/<ime razreda>)

#### 4. Rezervne lastnosti v kodi programa

Uporabnost programske opreme kvantificiramo v smilsu naslednjih ciljev:

- Lahka za naučiti
- Lahka za uporabljati
- Prijetna za uporabo

Razvijalci in uporabniki danes uporabljajo beta verzije operacijskih sistemov, razvojnih orodij, aplikacij

Kriteriji, ki vplivajo na načrtovanje:

- Izkušnje
- Razumevanje uporabnika
- Učinkovit postopek načrtovanja
- Potrebe (socialni, ekonomski prispevek)
- Lahko za naučiti
- Uporabnost
- Primernost (učinkovitost, praktičnost)
- Estetika
- Praktičnost (inštalacija, učenje, cena, podpora, nadgradnje,...)

### **Principi:**

#### 1. Zagotovi nadzor uporabnika

- Omogoči uporabo tipkovnice in miške (fleksibilnost)
- Omogoči prekinitev danih opravil (prekinljivost)
- Prikazuj obvestila in tekste (pomoč)
- Zagotovi takojšnje in ponovljive akcije ter povratno informacijo (odzivnost)

- Zagotovi značilne poti in izhod (navigacija)
- Prilagodi se uporabnikom z različnimi nivoji znanja (dostopnost)
- Zagotovi jasnost in čistost vmesnika (preglednost)
- Omogoči spreminjanje lasnosti vmesnika (želje)
- Omogoči direktno manipulacijo z grafičnimi gradniki (interaktivnost)
- Uporabljaljaj »načine« pametno (nedvoumnost)

## 2. Reduciraj obremenitev uporabnikovega spomina

- Razbremenjaj kratkotrajni spomin (pomnenje)
- Zanašaj se na razpoznavanje in ne na spoim (prepoznavanje)
- Zagotovi vizualne namige (informiranost)
- Zagotovi vgrajene akcije in razveljavitev ter ponovitev akcij (preprostost, reševanje)
- Zagotovi bližnjice (hitrost)
- Podpiraj način gradnik – akcija (intuitivnost)
- Uporabljaljaj metafore realnega sveta (prenos)
- Uporabljaljaj progresivni dostop (navigabilnost)
- Podpiraj vizualno čistost (organiziranost)

## 3. Zagotovi konsistenčnost vmesnika

- Ohranjaj kontekst uporabnikovih opravil (zveznost)
- Ohranjaj enovitost v predstavitvi informacij, obnašanju gradnikov in tehnikah interakcije (izkušnje)
- Ohranjaj enovitost rezultato interakcij (pričakovanje)
- Zagotovi estetsko privlačnost in polnost (izgled)
- Vspodbujaj preiskovanje (napovedljivost)

**Komponente oken:** okvir, naslov, sistemski menu, gumbi za velikost okna, vrstični menu, delovno področje, vrstica za izpis statusa, vrstica za obvestila, vrstica za ukaze, drsniki, pregrade, polje za prikaz permanentnih ukazov ali opcij

Okna glede na vrsto opravil:

- Primarno okno . okno aplikacije (aktivnosti dostopne uporabniku v vsakem trenutku)
- Sekundarna okna za razširitev aplikacije (brez vrstičnega menuja, obravnava enega podatkovnega pobjekta)
- Dialogi (akcije omejenega konteksta, kritični in paralelni dialogi)

Kompozicija oken:

- Drug poleg drugega (eno opravilo, predvidljiva količina podatkov, omejeno število oken, lažja manipulacija)
- Prekrivajoča okna (več opravil, nepredviljiva količina podatkov, večja okna, poljubno število oken, težja manipulacija)

Splošna navodila za načrtovanje menujev glede na:

- Prikaz opcij (permanenten, na zahtevo)
- Organizacijo opcij (splošni menu, prikaz le relevantnih alternativ na danem nivoju, omejeno število nivojev, neaktivne možnosti prikazane šibko)
- Grupiranje opcij (po hierarhiji in z ozirom na logičnost, različnost in medsebojno izključevanje, uporaba separatorjev)
- Vrstni red opcij (po vrstnem redu uporabe, pogostosti, abecedi, možnem destruktivnem učinku, relativnem položaju glede na sorodne menuje)
- Najverjetnejšo opcijo (kurzor)
- Navigacijo (vnaprejšnji prikaz opcij)
- Indikatorje naslednikov
- Ekvivalente preko tipkovnice (mnemoniki, pospeševalniki)
- Izvršitev opcij (izbor, izvršitev)

**Vrste menujev:** vrstični menu (menu bar), ruletni (izvlečni) menu (pulldown menu), kaskadni menu (cascading menu), dvižni menu (pop-up menu), menu z ikonami (iconic menu), opsijski menu (option menu)

Izbor menuja:

- Glede na število opcij
- Pogostost uporabe
- Pogostost spremembe vsebine menija

1. Pogoste akcije, skupne akcije za sekundarna okna
2. Pogoste akcije, za opcije predstavljene tekstualno, redke spremembe menuja, 5-10 opcij
3. Zagotovi preprostejši in preglednejši predhodni menu; za opcije, ki se medsebojno izključujejo, 1-kaskadi
4. Pogosta uporaba akcij v danem kontekstu, redke spremembe vsebine vsebine menuja, malo prostora na zaslonu, 5-10 opcij
5. Za izbiro aplikacije, za doseg posebnih funkcij dane aplikacije
6. Pogosta raba ene akcije v danem kontekstu, druge opcije redke v istem kontekstu, malo prostora na zaslonu, 5-10 opcij

#### **Motif vrstica kolona – RowColumn:**

- Naslednike razvrsti po vrsticah in (ali) kolonah
- Primeren za implementacijo menujev
- Možno razvrščanje naslednikov po realizaciji

#### **Motif upravljanje podob:**

- Upravnik upravlja svoje naslednike z nadzorom njihove velikosti in njihovega položaja
- Upravniki so zgrajeni in razkrojeni kot vse druge podobe
- Razlike nastopijo, ko je dana podoba deklarirana kot upravljana med procesom gradnje
- Podobi se realizirata, če sta upravnik in naslednik v upravljanem stanju
- Če je naslednik v neupravljanem stanju, ga upravnik preskoči
- Če je upravnik v neupravljanem stanju, potem ne upravlja s svojimi nasledniki

**Motif stilaka:** (toggleButton) so gumbi s pridruženim tekstom ali ikono

**Motif podoba lista:** (List) je namenjena prikazu liste tekstovnih možnosti in njih izbire

**Motif podoba tekst:** omogoča urejevanje teksta

Ne omogoča:

- Istočasnega prikaza večih barv
- Istočasnega prikaza različnih oblik znakov

- Ni emulator terminala
- Ne more poganjati interaktivnih programov
- Ne more prikazovati multimedijskih objektov

### **Navodila (elementi za interakcijo preko zaslona):**

Vrste elementov:

1. Gumbi – za ukaze, prikaz menujev, manipulacijo z oknom(i)
  - + vedno vidni, hitre akcije, razumljiv opis, 3D, omogočajo ekvivalente preko tipkovnice.
  - zasedajo veliko prostora, veliko premikov kurzorja
2. Polja za vnos – vnos informacije preko tipkovnice
  - +fleksibilnost, malo prostora
  - uporaba tipkovnice
3. Polja za izbiro (stikala, palete)
  - Stikala – postavljanje atributov, lastnosti
  - +lahak dostop, lahka primerjava, razumljivost
  - zasedajo veliko prostora, omejeno število možnosti
  - Palete – grafične alternative, ikone
  - +prijetnost, lahka primerjava, manj prostora kot tekst
  - potrebne izkušnje za atraktivno predst., omejeno število možnosti
4. Liste – izbira ene ali več opcij, stalen prikaz, (vnos preko tipkovnice)
  - +neomejeno število opcij, »prepoznavanje«, stalen prikaz
  - zasedajo veliko prostora, zamudno iskanje opcij
5. Izvlečne in dvižne liste – izbira ene opcije, prikaz na zahtevo, (vnos preko tipkovnice)
  - +neomejeno število opcij, »prepoznavanje«, ne zasedajo prostora
  - dodatni korak za prikaz liste, zamudno iskanje opcij
6. Kombinirana polja za vnos ali izbiro (krožna polja) – izbira ene opcije, možen vnos preko tipkovnice
  - +zasedajo malo prostora, fleksibilnost
  - težko primerjati opcije, le določena vrsta podatkov

7. Kombinirane liste – isto kot 4.
8. Kombinirane izvlečne in dvižne liste – isto kot 5.
9. Drsniki za parametre aplikacije – omogoča zvezno in kvalitativno nastavitve vrednosti parametra.

+prostorska predstavitev relativne nastavitve, vizualnost

-manjša preciznost od numerične nastavitve, veliko prostora

10. Drsniki – omogočajo prikaz informacije, ki zaseda več prostora od predvidenega

+prikaz neomejene količine podatkov

-prostor

Aspekti:

- število ukazov aplikacije
- kompleksnost ukazov
- frekvence (pogostost) ukazov
- ali je dani ukaz povezan z nekim drugim ukazom?

**Ukazi:**

Izbira:

7 ali več in obstoja hierarhija

Vrstični menu, menuji

6 ali manj in preprosta aplikacija

Gumbi v oknu

6 ali manj in pogosta uporaba

Gumbi v oknu

So uporabljeni z drugimi ukazi

Gumbi v dialogu

Pogosta uporaba, dve stanji

Stikala

Aspekti aranžiranja elementov za interakcijo:

- Količina informacije (perceptualna ločljivost gradnikov, estetska prijetnost, lega glede na pogostost uporabe, navigacija)
- Logična organiziranost (organizacija gradnikov naj odraža organiziranost okolja realnega sveta, ki mu služi aplikacija)
- Velikost okna (eno okno, če je le mogoče)
- Balansiranje (enakomerna distribucija elementov)
- Grupiranje

- Orientacija in poravnavanje

## GTK+/GNOME

Za razvoj aplikacij (GUV) pod Linuxom. Poenostavita razvoj aplikacij, ki temeljijo na X sistemu. Upravniki oken in okolja za razvoj aplikacij: standardizacija UV in večja uporabnost)

- GNOME uporablja GTK+ za GUV
- KDE uporablja QT za GUV

GNOME = GNU Network Object Model Environment

Upravniki:

- Dialog (BulletinBoard) – komunikacija med uporabnikom in aplikacijo. Ne spreminjajo po velikosti in ne aranžirajo svojih naslednikov
- Form – določa geometrijo in medsebojno lego svojih naslednikov v. Nasledniki so razvrščeni relativno eden na drugega
- SelectionBox – izbira v polju tekstualnih zapisov
  - o Command – ukaz
  - o fileSelectionBox – izbira datoteke
- MessageBox – sporočilo
- Nadzorni drsnik (Scale) – drsnik in izpis vrednosti neke veličine
- Grafično polje (DrawingArea) – polje za risanje (Xlib)

**Motif oglasna deska:** Upravnik oglasna deska (Bulletin Board) postavi naslednike tja, kamor načrtovalec želi. Je osnova dialogom.

**Motif upravnik obrazec:** opis relativnih povezav med nasledniki, opis relativnih povezav med lokacijo znotraj sebe in nasledniki

**Motif upravnik drsnik:** prikazuje numerično vrednost znotraj v naprej definirane intervala s spodnjo in zgornjo mejo

**Motif podoba za risanje:** Drawing Area) je namenjena uporabi funkcij knjižnice Xlib.

**Navodila ikone:** Ikone opisujejo in predstavljajo objekte in akcije. Ikona je nekaj, kar zgloda tako kot tisto kar predstavlja.

- Določi slike, ki najboljše predstavljajo objekte ali akcije
- Uporabi vsebovane ikone danega sistema
- Z neupoštevanjem navodil kreiraj ikone. Uporabi preproste metafore in minimalno množico konceptov
- Uporabi standardne ikone danega sistema ali mednarodnih organizacij (ISO)



- Kreiraj razumljive in prepoznavne oblike
- Kreiraj vizualno in konceptualno ločljive oblike
- Predstavi objekte čisto in preprosto, brez detajlov
- Ohranjaj konsistenčnost strukture in oblike pri skupinah ikon (omejeno število sprememb v kotih, debelinah črt, oblikah, količini praznega prostora)
- Ohranjaj konsistenčnost metodologije načrtovanja pri skupinah ikon (uporabi: stanja pred in po akcijah, akcije, orodja za akcije)
- Kreiraj oblike, ki vizualno izražajo relacije med ikonami
- Izražaj attribute ikon čim bolj jasno
- Po potrebi dodaj pojasnila
- Testiraj ikone z ozirom na učinkovitost, prepoznavanje in učenje

Aranžiranje ikon:

- Ohrani število vrst ikon pod 12 in število vseh ikon pod 20
- Aranžiraj ikone tako, da odražajo organizacijo objektov v realnem svet
- Aranžiraj ikone za objekte in ikone za akcije v različne skupine
- Omogoči uporabniku aranžiranje ikon in prikaz pojasnil

**Navodila besede in teksti:** besede in teksti naj bodo preprosti, čisti in vljudni

- Besede (kratke, domače, cele, v pozitivnem smislu)
- Teksti (40-60 znakov na vrstico, levo poravnani, zamikanja, prazne vrstice med odstavki, povdarjanja (škatle, krepko, značke), alineje)
- Stil pisanja (kratki stavki, <30 besed, kratki odstavki, aktiv, primeri)

**Principi povratnih informacij:** odzivnost sistema naj je prilagojena hitrosti in toku človekovih miselnih procesov

- Človekova sedanost (psihološka zavest): 2-3 sekunde
- Kreativno delo: odzivni čas med 400ms in 900ms
- Konstantne zakasnitve
- Zahtevana zveznost razmišljanja, informacija potrebna v zavesti preko več akcij: <2sekundi
- Med zaključki opravil: 2-15sekund

Delo s časovnimi zakasnitvami:

- < 5 sekund: signal (ikona)
- >5 sekund:
  - o Indikator stanja procesa
  - o Obvestilo o odstotku opravljenega dela
  - o Obvestilo o preostalem času potrebnem za dokončanje procesa

**Navodila uporaba zvoka:** zvok uporabimo pri: napakah, nepričakovanih dogodkih, potrebni takojšnji pozornosti, po izvršitvi dolgega procesa

- Vedno v povezavi z vizualno indikacijo
- Do 6 različnih tonov
- Konstistenčna uporaba (podobni toni za podobne situacije)
- Med 500 in 1000 Hz
- Možnost izklopa ali nastavitve jakosti

### **Navodila uporaba barv:**

- Povdarja logično organizacijo okna (zaslona)
- Izboljša ločljivost komponent okna
- Naredi prikaz bolj atraktiven
- Izboljša zmogljivost
- Pri večjih oknih (tudi več barv)
- Štiri do največ 10 barv

Nepravilna uporaba povzroča: neprijaznost, vizualno utrujenost, manjšo zmogljivost.

Problemi pri uporabi barv:

- Ista barva gradnikov, ki nimajo nič skupnega (zbežnost, zmešnjava, počasnejše razpoznavanje)
- Pri večjih oknih (več barv -> večji reakcijski čas -> večja verjetnost napake)
- Oko je manj občutljivo na modro in rdečo (zgornji, spodnji del spektra) kot na rumeno in zeleno (srednji del spektra) – reakcijski čas, odstotek napak
- Uporaba samo modre in rdeče – problem fokusiranja (utrujenost)
- Barvna slepota (8%; 0,4%; rdeče-zelena, modra, rumena)
- Pomen barv glede na kulture in discipline (rdeče: stop, nevarnost)

Izbira barv:

- Načrtuj okno (zaslon) najprej črno-belo nato konzervativno dodajaj barve
- Za harmonijo in absolutno ločljivost (v danem polju ni drugih barv): 4-5 barv (rdeča/zelena, rumena/modra)
- Za harmonijo in komparativno ločevanje (v danem polju so druge barve): 6-7 barv enakomerno razporejenih po barvnem spektru
- Za povdarjanje in vzbujanje pozornosti: svetle barve (bela, rumena, zelena, cianova, modra in rdeča)
- Za povdarjanje in ločljivost: kontrastne barve (rdeča/zelena, rumena/modra)
- Za povdarjanje podobnosti: podobne barve (rdeča/rumena, zelena/modra)
- Za izražanje skupnega pomena (nujne akcije: tople barve (rdeča, oranžna, rumena); informacije o statusu: hladne barve (zelena, cianova, modra, vijolična))

Alfanumerični in grafični prikazi: izbrane barve naj zadoščajo zahtevam vidljivosti, kontrastu in harmoniji.

Barve ospredja: tople (svetle) barve, različne od barv ozadja, z med seboj istim nasičenjem in intenziteto, nenasičene barve in barve spodnjega dela spektra (bela), rumena, zelena, cianova.

Barve ozadja: hladne (temne) barve, nasičene barve in zgornjega dela spektra – (črna), modra, cianova, (modro-rdeča).

Intenziteta – količina intenzitete svetlobe (luminance), sprejeta intenziteta, svetlost (brightness). Fizični smisel: valovna dolžina, čistost, intenziteta. Fiziološki smisel: barva, nasičenje (saturation – kako daleč je barva od sive), svetlost (brightness).

**Motif dialogi:** aplikacije večkrat rabijo prehodna okna – dialoge.

1. Dialogi za obvestila: kratko obvestilo, gumbi (OK, Cancel, Help), modalnost, takojšnja uporaba kateri sledi razkroj, (ErrorDialog, InformationDialog, MessageDialog, QuestionDialog, WarningDialog,...)
2. Dialogi za izbiro: lista alternativ, gumbi: Ok, Aplly, Cancel, Help, periodična uporaba, (SelectionDialog, PromptDialog, FileSelectionDialog, CommandDialog,...)
3. Nemodalni način, okzno zamrzne, aplikacija zamrzne, sistemski modalni način

Navodila dialogi: dvižno okno, tekst, ikona, gumbi za akcije

Stil in ton obvestil:

- Direktni stavki
- Trdilni in ne nikalni stavki
- Aktiv
- Kronologija besed

Izogibaj se:

- Autoritarnosti
- Negativnemu tonu
- Antropomorfnosti
- Patroniziranju
- Humorju in kazni

### **Standardi in testiranje uporabnosti:**

**Standardi** definirajo karakteristike objektov in sistemov, ki jih uporabljamo vsakodnevno (npr.: QWERTY). Omogočajo »komunikacijo« načrtovalca s katerim koli drugim načrtovalcem. Se stalno ažurirajo, sicer zavirajo razvoj tehnologije in inovativnost.

- ISO – International Standard Organization
- ANSI – American National Standard Institute

Zakaj standardi?

- Lažja izbira in uporaba informacijsko procesne opreme
- Eliminacija nekonstitenčnosti in razlik

**Navodila** za načrtovanje uporabniških vmesnikov se nanašajo na:

- Predstavitev grafičnih gradnikov vmesnika uporabniku in estetiko
- Intearkicijo uporabnika preko naprav za interakcijo in preko grafičnih gradnikov za interakcijo
- Lastnosti gradnikov (objektov) in njihove medsebojne relacije

Problem pri novih tehnologijah (npr: zvok, virtualna realnost): navodila lahko nastanejo šele po določenem času uporabe teh novih tehnologij

**Testiranje uporabnosti** vmesnika lahko nakaže možnost za odstopanje od navodil. Kaj je test uporabnosti? ISO: Test uporabnosti je postopek testiranja uporabnikov, nalog in okolja z namenom vrednotenja zmogljivosti vmesnika z ozirom na uporabnikovo učinkovitost in zadovoljstvo.

Mere zmogljivosti (kvantitativne mere): število nalog, čas opravlja naloge, napake, pomoč.

Subjektivna merila (kvalitativna merila): Ocena glede na razpoznavanje, mnenja , želje in zadovoljstvo.

Faktorji uporabnosti:

- Primernost – mere zmogljivosti (Stopnja do katere vmesnik uporabniku omogoča doseganje dane naloge ? Motivacija za uporabo ?)
- Učinkovitost – mere zmogljivosti (kako uspešno uporabnik opravlja opravila?)
- Naučljivost – mere zmogljivosti (ali se uporabniki naučijo uporabe vmesnika v danem času?)
- Odnos – subjektivna merila (uporabnikova ocena glede na razpoznavanje, mnenja, občutke o učenju in uporabi)

Faktor uporabnosti	Cilj
Primernost:	Uporabnik bo uporabljal nov vmesnik
Učinkovitost:	Vmesnik bo izboljšal uporabnikovo uspešnost
Naučljivost:	Manj potrebnega učenja
Odnos:	Uporabnik bo zadovoljen

Merljivi cilji pri testiranju uporabnosti:

Faktor uporabnosti	Kriterij	Zmogljivost	Pogoj
Primernost	N% uporabnikov	Uspešno izvrši	Po N danih nalogah

		nalogo	
Učinkovitost	N% uporabnikov v T minutah	Uspešno izvrši nalogo	Po N danih nalogah
Naučljivost	Vsi uporabniki z danim predznanjem	Uspešno osvojijo znanje	Po M urah učenja
Odnos	N% uporabnikov srednja ocena O	Ocena prijaznosti	Po N nalogah

### **Objektno usmerjeni uporabniški vmesniki**

Razlike med aplikativno in objektno usmerjenimi uporabniškimi vmesniki (GUV, OUUV)

G: Aplikacijo sestavljajo ikona, primarno in sekundarna okna

O: Vmesnik sestavlja množica sodelujočih objektov in pogledov na njih

G: Ikone predstavljajo aplikacije

O: Ikone predstavljajo objekte, ki jih lahko direktno manipuliramo

- G: Uporabnik požene aplikacijo preden dela z njo.
- O: Uporabnik odpre objekte v poglede preden dela z njimi
- G: Pozornost nad osnovno nalogo kot je ta določena z aplikacijo
- O: Pozornost nad vhodom in izhodom na objekte ter nalogami
- G: Učenje je koncentrirano na aplikacijo in njene funkcije
- O: Učenje je koncentrirano na skupne pristope in izgled objektov
- G: Vsebina je prikazana s tekstovnimi listami ali dialogi
- O: Vsebina je prikazana z vsebovalniki (mape – folder, in beležke – notebook)
- G: Ena aplikacija je eno opravilo
- O: En objekt lahko uporabimo za različna opravila

### **Osvnovni koncepti pri OUUV:**

- Skrivanje arhitekture sistema pred uporabnikom (bližnjice – shortcuts, shadows)
- Objekti in pogledi v objekte (pogled je predstavitev informacije o objektu)
- Direktna manipulacija
- Ikone so objekti, ki imajo določene relacije z drugimi objekti

### Arhitektura OUUV:

- Sekvenca objekt-akcija (akcija-objekt : ukazna vrstica, menuji)
- Vrstični – WOSH (FEVH) menu, WOSH: W – window, O – object (objekt gledan v oknu), S – selected (objeti izbrani v pogledu), H – help
- Podatkovni objekti (tekst, preglednice, slike, grafikoni, glasba, posnet govor, video, animacija, sestavljeni objekti)
- Objektni vsebovalniki – orodja za organizacijo, shranjevanje, grupiranje, dodajanje, odstranjevanje, sortiranje (različni ključni) objektov in vsebovalnikov ter opravil (osnovno okno – zaslon (workspace, desktop), splošni vsebovalniki – mape (folders), posebni vsebovalniki – delovna področja (workareas))
- Objekti za naprave – za komunikacijo in interakcijo z objekti povezanimi z računalnikom (fax, tiskalnik, miška,...)
- Tipi pogledov na objekte:
  - o Sestavljen objekt – podatkovni objekti (okna, informacija in objekti vsebovani v objektu)



- o Vsebinski pogled – podatkovni objekti, objekti vsebovalniki, objekti za naprave (mape, komponente ali vsebine objektov)
- o Pogled nalastnosti – podatkovni objekti, objekti vsebovalniki, objekti za naprave (beleške, lasnosti objektov)
- o Pomoč – objekti, individualni elementi (kontekstualna pomoč), (vrstični menu ali dvižni menu)

OOUV in obremenitev uporabnikovega spomina:

Poleg vizualnih izbir, list, grafičnih gradnikov, ikon, pomoči, menujev, izključevanja opcij, statusne vrstice, vrstice aplikacije in obvestil uporabljajo OOUV za razpoznavanje še:

- Dolga imena datotek
- Imena oken: ime objekta in ime pogleda (Clock – Properties)
- Objekte vsebovalnike (sortiranje po ključih, mape, beleške, odpiranje/zapiranje objektov znotraj delovnih področij)
- Kontekstualno pomoč
- Bližnjice – linke
- Vzorce (templates)
- Povdaranje objekta ob izbiri ali uporabi

### **Uporabniški vmesniki za internet:**

Internet in svetovni splet (world wide web)

Omogočata iskanje, pregledovanje (brskanje) in gledanje teksta ter grafičnih informacij shranjenih na računalnikih po celem svetu. Bariere med lokalno in oddaljeno shranjeno informacijo ni več.

- Internet je množica neodvisnih računalnikov, ki so povezani v mrežo in si med seboj delijo računalniške resurse (strežniki: HTTP, FTP, SMTP).
- Intranet je področje interneta za požarnim zidom (firewall), običajno neke organizacije.
- Svetovni splet (WWW, W3, Web) je množica tistih računalniških datotek (HTML – hyper text markup language, GIF – graphics interchange file), ki so dosegljive preko internetam povezane med sabo (tags) in dosegljive preko vmesnikov za internet (browser).

Etika in morala:

- Domače strani so zaščitene

- Izogibaj se tekstom, ki škodijo ugledu
- Izogibaj se neprimernim materialom
- Sklenjeni dogovori so veljavni

Aspekti:

- Ljudje nimajo potrpljenja s slabo načrtanimi domačimi stranmi
- Ljudje nočejo pregledovati dolgih strani z drsniki
- Ljudje nočejo brati

Navodila:

- Predvidevaj profil obiskovalcev (novi, z izkušnjami)
- Definiraj uporabnikove naloge (npr: najdi informacije, učenje, naročilo, kontaktirati predstavnika, tehnična pomoč, povratna informacija, kopiranje,...)
- Pomoč uporabniku pri navigaciji (čista in intuitivna informacija, razumljiva vizualna navigacija (gumbi, tekst, slike), vključuj ključne besede, omogoči skok na vsako stran, označi povezane slike, omogoči vrnitev na prvo stran, domači elementi za navigacijo, standardne barve in poudarjanja)
- Omogoči uporabniku hitro delo (reduciraj odzivne čase, čase nalaganja slik, izogibaj se drnikom, omejeno število opcij, logično grupiranje opcij, uporablaj načelo »razpoznavanja«, preprost kontekst – ena tema na stran, važnejša vsebina na začetku, primerno ozadje, veliko belih presledkov, standardne palete in fonti, tekst dovolj velik in dovolj kontrasten z ozadjem)
- Zagotovi konsistentnost (fonti, barve, slike, oblika strani, elementi za navigacijo, izrasoslovje, povezave in oznake)

Kakšna je razlika med standardom, principom, navodili?

- Standard je potrebno upoštevati, za lažjo komunikacijo
- Navodila nekaj vmes
- Princip je najbolj ohlapen, da se najbolj spreminjati