

## Dizajn rozhrania GUI

- Úvod
- Koncepty návrhu
- Zásady pri tvorbe rozhrania
- Metódy návrhu GUI
  - CRAP
  - Mriežka
- Metódy validácie a hodnotenia
- Typické chyby dizajnu

Najčastejším rozhraním, s ktorým v súčasnosti komunikujeme so strojmi (PC, tablety, Mobilne telefóny) sú grafické rozhrania. Sú tak široko používané že asi každý z nás sa s nimi stretne každý deň niekoľko krát. Preto je dôležité venovať sa jeho správne mu návrhu. Tým značne zvýšime efektívnosť používania tohto rozhrania, t.j. rýchle a prehľadne vyhľadanie/ ovládanie, minimalizácia omylov, rýchle odstránenie prípadného omylu, rýchle naučenie používania rozhrania atď. Tato lekcia bude preto venovaná správnym zásadám a postupom návrhu grafického rozhrania.

## Úvod

- **Užívateľské rozhranie sa hodnotí veličinou použiteľnosť (usability)**
  - *Do akej miery môže byť produkt používaný špecifickými používateľmi efektívne, účinne a uspokojivo, aby dosiahli ciele v špecifickom kontexte použitia [ISO, 1998]*
- **Použiteľnosť v sebe zahŕňa tieto aspekty**
  - **Pochopiteľnosť (Learnability)**
    - *System by mal byť intuitívny, aby ho používateľ vedel aj prvý krát rýchlo použiť, bez zdĺhavého učenia*
    - *Ako rýchlo vie používateľ vykonať úlohu po prvý krát?*
  - **Efektívnosť (Efficiency)**
    - *Keď sa už používateľ naučil pracovať so systémom ako rýchlo je ho schopný ovládať*

Na úvod si treba zdefinovať ako sa ma hodnotiť užívateľské rozhranie a čo všetko v sebe takýto parameter musí odrážať.

## Úvod

### – Zapamätanie si (Memorability )

- *Systém by sa mal ľahko pamätať, teda používateľ by sa mal byť schopný k nemu vrátiť po nejakom čase, bez nutnosti znovu sa učiť*

### – Chyby (Errors)

- *Používatelia systému by mali produkovať málo chýb, keď sa vyskytnú, mal by sa dať ľahko opraviť*

### – Spokojnosť (Satisfaction )

- *Ak sú užívatelia subjektívne spokojní s pracou na danom rozhraní (či sa im rozhranie páči)*

## Koncepty návrhu

- Návrh možno založiť z 2 pohľadov:
  - z pohľadu systému
  - z pohľadu používateľa
- Z pohľadu systému
  - Čo ponúkajú, resp. aké sú obmedzenia dostupných technológií
    - Softwarové balíky
    - Hardware
  - Čo je v možnostiach a schopnostiach samotných programátorov

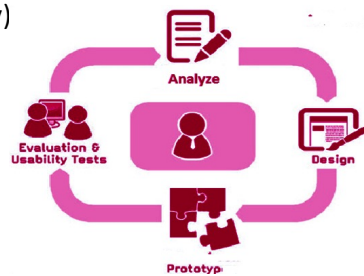
Sú 2 hlavne pohľady ako možno navrhovať rozhranie: z pohľadu užívateľa a z pohľadu systému (návrhárov). Oba majú svoje výhody a nevýhody a vedú k odlišným výsledkom. V ďalšom budú tieto dva pohľady (návrhy) rozobrane podrobnejšie.

## Koncepty návrhu

- Čo je pre programátorov a návrhárov zaujímavé a potrebné
- Z pohľadu používateľa (user centred)
  - Jeho schopnosti
  - Práca
  - Potreby
  - Spoločensko-kulturálny kontext
  - Efektívne použiteľný produkt s minimálnymi nákladmi vo fáze vývoja, prevádzky, údržby a rozšírenia.
  - Používateľ je prítomný vo fázach vývoja a nasadenia

## Koncepty návrhu

- Fázy vývoja (user centred)
  - Analýza
  - Dizajn
    - Wireframes (drôtené modely)
    - Mock-ups (napodobeniny)
  - Prototypy
  - Validácia



Samotný návrh zvyčajne pozostáva z viacerých fáz (analýza, návrh, prototyp, evakuácia), z ktorej každá ma svoj špecifický význam a výstup. Tieto fázy majú zabezpečiť efektívny návrh rozhrania (vo všeobecnosti nie len rozhrania)

## Koncepty návrhu

- Analýza
  - Ohodnotiť súčasný stav produktu
  - Pochopiť požiadavky používateľov
  - Možnosti súčasných technológií
  - Možnosti tímu
- Dizajn
  - Wireframes (drôtené modely)
    - Jednoduché rozmiestnenie prvkov
    - Najjednoduchšia vizualizácia ukázaná používateľovi

V prípade návrhu (vzhľadu) pomáhajú modely rôznych úrovni presnosti od najjednoduchších tzv. Wireframes cez Mockups až po prototypy. V ďalšom bude vysvetlený ich význam a rozdiely.

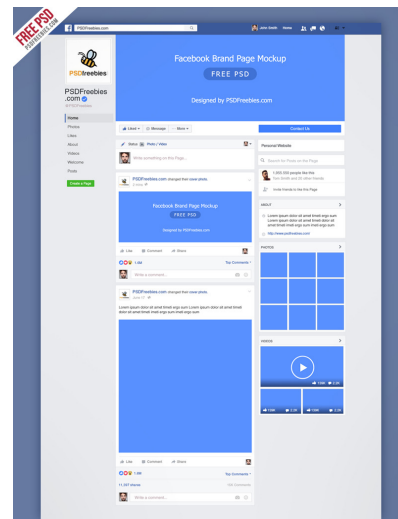




# Koncepty návrhu

– Mockups (modely, napodobeniny)

- Vizuálna stránka návrhu
- Sú ako „pokožka“ návrhu
- Kvalitné grafické prevedenie



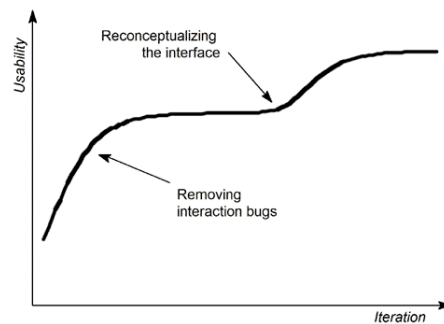
## Koncepty návrhu

### – Prototyp

- Najfunkčnejšia dokumentácia aká sa dá používateľovi vo fáze dizajnu ukázať
- Okrem vizuálnej stránky má s časti sprístupnenú aj obmedzenú funkcionálnosť
- Používateľ môže s prototypom vykonávať isté akcie
- Spätná väzba s používateľom
- Slúži na otestovanie interakcií rozhrania

## Koncepty návrhu

- Poznámka
  - Interakcia s používateľom a úprava dizajnu po jeho konzultácii pred nasadením značne zvyšuje použiteľnosť



## Všeobecné pravidlá dizajnu

- **Podľa: Jacob Nielsen**

- **Viditeľnosť stavu systému**

– *Systém by mal priebežne informovať o svojom stave (čo sa deje), pokiaľ možno v reálnom čase.*

- *Progress bar*



- *Správa potvrdzujúca úspešné vykonanie akcie*



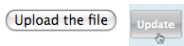
- *Chybové hlásenie*



V nasledovnej rozsiahlejšej časti budú postupne predstavené všeobecne pravidlá návrhu najmä platné pre grafické rozhrania. Spolu s nimi budú uvedené aj ich praktické príklady (kde to nie je z textu zrejme). Tieto zásady zabezpečia rýchly a prehľadný návrh platný pre širokú škálu aplikácií. Týchto sad-pravidiel je viac, tu sa vychádza z práce J. Nielsena.

## Všeobecné pravidlá dizajnu

– *Znemožnenie/ Sprístupnenie funkcií (tlačítok) podľa stavu spracovania*




- **Súlad medzi systémom a skutočným svetom**

– *Použiť jazyk (slová, výrazy) blízky používateľom (nie termínmi návrhárov).*

– *Informácie sa majú objavovať vo vhodnom poradí (logicky, chronologicky, atď.)*

## Všeobecné pravidlá dizajnu

- **Riadenie vykonávania používateľom**

- *Používatelia môžu vybrať funkciu systému omylom preto potrebujú viditeľnú možnosť návratu (opustiť nechcený stav bez prechádzania dlhého dialógu).*
- *Zaviest' funkciu undo, resp. redo (text, resp. zaužívaný piktogram  )*

## Všeobecné pravidlá dizajnu

- **Konzistencia a štandardy**
  - *Postupujte podľa konvencie*
  - *Rovnaké slová, frázy, situácie, piktogramy, majú opisovať rovnaké akcie, ktoré sa majú vykonať.*
- **Prevenia chýb**
  - *Zásada: je lepší premyslený dizajn, zabraňujúci omylom, než dobré chybové správy.*
  - *Eliminácia vzniku chýb jednoznačným dizajnom a umožnenie dodatočnej kontroly a potvrdenia používateľom pred akciou.*

## Všeobecné pravidlá dizajnu

- ***Lepšie rozpoznanie ako pamätanie***
  - *Objekty, akcie a možnosti majú byť viditeľné*
  - *Používateľ by si nemal pamätať inštrukcie v jednej časti dialógu potrebné v druhej časti.*
  - *Návody by mali byť vždy viditeľné alebo ľahko vyhľadateľné.*



## Všeobecné pravidlá dizajnu

- **Flexibilita a efektívita**
  - Neviditeľné akcelerátory pre začiatočníka môžu skúsenému používateľovi urýchliť obsluhu. Systém je tak vhodný pre oba typy používateľov (aj efektívny aj flexibilný).
  - Umožní používateľom prispôbiť si systém pre časté akcie.

## Všeobecné pravidlá dizajnu

- **Minimalistický dizajn**
  - Dialógy by mali obsahovať len relevantné a často poskytované informácie.
  - Každá nadbytočná informácia znižuje viditeľnosť tých podstatných.
- **Pomôž používateľom rozpoznať, diagnostikovať a zotaviť sa z chýb**
  - Chybové správy:
    - majú byť vyjadrené v jednoduchom jazyku
    - mali by presne indikovať problém
    - majú poskytnúť konštruktívne riešenie

## Všeobecné pravidlá dizajnu

- **Nápoveda a dokumentácia**
  - *Aj ten užívateľský najlepší systém môže niekedy potrebovať nápovedu, resp. dokumentáciu.*
  - *Preto je ju potrebné vytvoriť a poskytnúť*
    - *Informácie by mali byť ľahko vyhľadateľné.*
    - *Je vhodné uvádzať postupnosti konkrétnych krokov*
- **Odozva (Reakčná doba)**
  - 0.1 sekundy – pocit okamžitej odozvy
  - 1.0 sekundy – limit pre pocit neprerušovanej práce
  - 10 sekúnd – limit pre udržanie pozornosti
  - Označiť chybu systému ?

## Metódy návrhu GUI CRAP

- Metóda stojí na dodržiavaní nasledovných pravidiel
  - Contrast
  - Repetition
  - Alignment
  - Proximity
- Contrast
  - Odlišné veci majú vyzerat' odlišne
  - Dôraz na dôležité prvky

Sumarizácia spomenutých základných pravidiel viedla ku konštrukcii viacerých postupov návrhu, ktoré majú za cieľ navlnúť požadovane grafické rozhranie, ktoré "automaticky" zabezpečí jeho súlad aspoň so základnými pravidlami návrhu. V ďalšom si rozperieme 2 metódy: CRAP a Grid- mriežka, a to aj na praktickom príklade. Najprv začneme s metódou CRAP ktorá sleduje nasledovne aspekty: kontrast, opakovanie, zarovnanie a blízkosť. Ich význam a príklady sú vysvetlene v ďalšom.

# Metódy návrhu GUI

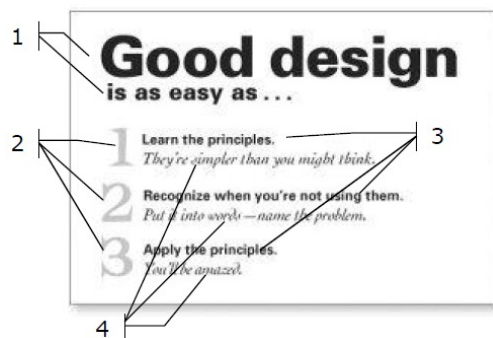
## CRAP

– Vytvárat dynamiku



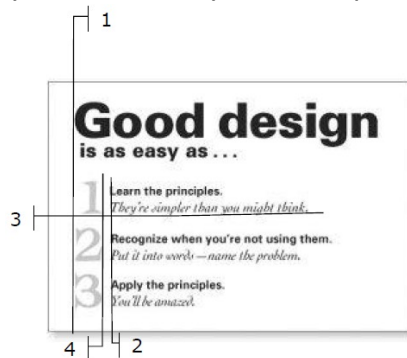
## Metódy návrhu GUI CRAP

- Repetition (opakovanie)
  - V celom rozhraní opakuj rovnaký vzhľad
  - Dodržovať konzistenciu
  - Vytvárať jednotu



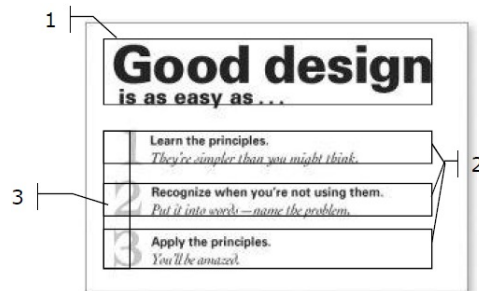
## Metódy návrhu GUI CRAP

- Alignment (zarovnanie)
  - Vizualne prepojovať / oddeľovať jednotlivé prvky
  - Vytvoriť vizuálny tok (ako prechádzať textom)



## Metódy návrhu GUI CRAP

- Proximity (blízkosť)
  - Zblížiť / združiť súvisiace prvky
  - Oddeliť nesúvisiace prvky






# Metódy návrhu GUI

## CRAP

- Príklad uplatňovania pravidiel - originál

**GroupLab**  
The University of Calgary

[Saul Greenberg](#) [GroupLab](#) [Dept Computer Science](#) [University of Calgary](#)



Saul Greenberg, Professor  
Human-Computer Interaction &  
Computer Supported Cooperative Work  
Dept. of Computer Science  
University of Calgary  
Calgary, Alberta  
CANADA T2N 1N4  
Phone: +1 403 220 4087  
Fax: +1 403 284 4707  
Email: [saul@gpsc.ucalgary.ca](mailto:saul@gpsc.ucalgary.ca)

**Research**

[GroupLab project](#) describes research by my group

[Publications](#) by our group; most available in HTML, PDF, and postscript

[Project snapshots](#) describes select projects done in GroupLab

[GroupLab software repository](#)

[GroupLab people](#)

**Graduate Students**

I have a few openings for M.Sc. and PhD students who are interested in Human Computer Interaction and/or Computer Supported Cooperative Work. [Some research and project ideas, honors and graduate students](#)

*Courses offered this year*

CPSC 481: Foundations and Principles of Human Computer Interaction

**Originál**

Príklad postupného uplatňovania kontrastu, blízkosti, opakovania a zarovnaní na zle navrhutej web- stránke.

# Metódy návrhu GUI CRAP

- Príklad uplatňovania pravidiel- blízkosť

**GroupLab**  
The University of Calgary

Saul Greenberg GroupLab Dept. Computer Science University of Calgary

**Saul Greenberg, Professor**  
Human-Computer Interaction &  
Computer Supported Cooperative Work

Dept. of Computer Science  
University of Calgary  
Calgary, Alberta  
CANADA T2N 1N4

Phone: +1 403 220-4087  
Fax: +1 403 234-4107  
Email: [saul@cpsc.ucalgary.ca](mailto:saul@cpsc.ucalgary.ca)



#### Research

[GroupLab.ca/research](http://GroupLab.ca/research) describes research by my group  
Publications by our group, most available in HTML, PDF, and postscript  
[Project.stanpaloha.com](http://Project.stanpaloha.com) describes select projects done in GroupLab  
[GroupLab software repository](http://GroupLab.ca/software/repository/)  
[GroupLab people](http://GroupLab.ca/people)

#### Graduate Students

I have a few openings for M.Sc. and PhD students who are interested in Human Computer Interaction and / or Computer Supported Cooperative Work. Some research and project ideas [GroupLab.ca](http://GroupLab.ca) honours and graduate students

#### Courses offered this year

[CPSC 481](http://CPSC.481): Foundations and Principles of Human Computer Interact  
[CPSC 481](http://CPSC.481): Human Computer Interaction II: Interaction Design  
[CPSC 601.13](http://CPSC.601.13): Computer Supported Cooperative Work

**Blízkosť**

# Metódy návrhu GUI

## CRAP

- Príklad uplatňovania pravidiel- zarovnanie

Saul Greenberg GroupLab Dept. Computer Science University of Calgary  The University of Calgary

Saul Greenberg, Professor  
Human-Computer Interaction &  
Computer Supported Cooperative Work

Dept. of Computer Science  
University of Calgary  
Calgary, Alberta  
CANADA T2N 1N4

Phone: +1 403 220-6087  
Fax: +1 403 254-4707  
Email: saul@cpsc.ucalgary.ca



**Research** [GroupLab project](#) describes research by my group  
[Publications](#) by our group; most available in HTML, PDF, and postscript  
[Project assignments](#) describes select projects done in GroupLab  
[GroupLab software repository](#)  
[GroupLab people](#)

**Graduate Students:** I have a few openings for MSc and PhD students who are interested in Human-Computer Interaction and/or Computer-Supported Cooperative Work. Some research and project ideas honors and graduate students.

**Courses offered this year**  
Cpsc 351: Foundations and Principles of Human-Computer Interaction  
Cpsc 351: Human-Computer Interaction II: Interaction Design  
Cpsc 551.12: Computer-Supported Cooperative Work

**Previous Terms:**  
Cpsc 681: Research Methodologies in Human-Computer Interaction  
Cpsc 659: Research Methodology for Computer Science (old!)  
Cpsc 601.48: Social Trends: Heuristic Evaluation

**Zarovnání**

# Metódy návrhu GUI CRAP

- Príklad uplatňovania pravidiel- kontrast



The screenshot shows a personal website for Saul Greenberg, a professor at the University of Calgary. The page includes a navigation menu, a header with the 'CrapLab' logo, a profile picture of Saul Greenberg, and sections for 'Graduate Students', 'Courses offered this year', 'Previous Years', and 'Administration'. The 'Courses offered this year' section lists several courses: CPSC 481, CPSC 581, CPSC 601.13, CPSC 681, CPSC 699, CPSC 601.48, CPSC 601.56, SENG 609.05, and SENG 609.06. The 'Administration' section mentions his role on the 'Ethics Committee'.

**Graduate Students** **Research Ideas.** I have a few openings for MSc and PhD students who are interested in Human Computer Interaction and / or Computer-Supported Cooperative Work.

**Courses offered this year**  
**CPSC 481:** Foundations and Principles of Human Computer Interaction  
**CPSC 581:** Human Computer Interaction II: Interaction Design  
**CPSC 601.13:** Computer-Supported Cooperative Work

**Previous Years**  
**CPSC 681:** Research Methodologies in Human Computer Interaction  
**CPSC 699:** Research Methodology for Computer Science (old)  
**CPSC 601.48:** Special Topics: Heuristic Evaluation  
**CPSC 601.56:** Advanced Topics in HCI: Media Spaces and Casual Interaction  
**SENG 609.05:** Graphical User Interfaces: Design and Usability  
**SENG 609.06:** Special Topics in Human Computer Interaction  
**Ego alert:** My entry on U Calgary's "Great Teachers" Web Site

**Administration** **Ethics Committee** for research with human subjects; I am the chair

Last updated: March 20, 1987

**Kontrast**

# Metódy návrhu GUI CRAP

- Príklad uplatňovania pravidiel- opakovanie

Saul Greenberg GroupLab Dept. Computer Science University of Calgary  
**Saul Greenberg**  
**Professor**  
Human-Computer Interaction & Computer-Supported Cooperative Work  
Dept. of Computer Science  
University of Calgary  
Calgary, Alberta  
CANADA T2N 1N4  
Phone: +1 403 220-0007  
Fax: +1 403 244-0707  
Email: saul@cs.ucalgary.ca



**Graduate Students** **Research Ideas** I have a few openings for MSc and PhD students who are interested in Human Computer Interaction and / or Computer-Supported Cooperative Work.

**Courses offered this year** **CPSC 481** Foundations and Principles of Human Computer Interaction  
**CPSC 581** Human Computer Interaction II: Interaction Design  
**CPSC 601.13** Computer-Supported Cooperative Work

**Previous Years** **CPSC 681** Research Methodologies in Human Computer Interaction  
**CPSC 699** Research Methodology for Computer Science (old)  
**CPSC 601.48** Special Topics: Heuristic Evaluation  
**CPSC 601.59** Advanced Topics in HCI: Media Spaces and Casual Interaction  
**SENG 609.02** Graphical User Interfaces: Design and Usability  
**SENG 609.03** Special Topics in Human Computer Interaction  
**Ego alert** My entry on U.Calgary's "Great Teachers" Web Site

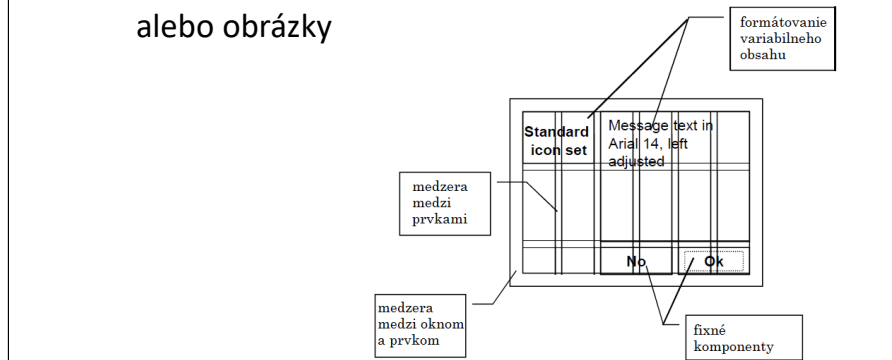
**Administration** **Ethics Committee** for research with human subjects

Last updated: March 20, 1997

**Opakování**

## Metódy návrhu GUI Mriežka

- Rozdelenie okna na mriežku
  - Rôzne rozstupy delenia, ale konzistentne
  - Do nich sa umiestňujú ovládacie prvky, texty, alebo obrázky



Ďalšou metódou návrhu grafického rozhrania je mriežka. Ako z názvu vyplýva displej je rozdelený do mriežky, ktorá môže mať nerovnaké bunky preto, aby mohla podporovať viaceré úrovne významnosti prezentovaných informácií. V ďalšom je tento koncept návrhu v stručnosti naznačený.

## Metódy návrhu GUI

### Mriežka

- Organizácia usporiadania
  - Združovať súvisiace prvky blízkosťou
  - Kontrast medzi významnými prvkami
  - Zarovnanie
- Konzistencia
  - Opakovať prvky
  - Umiestnenie
  - Formátovanie

## Metódy návrhu GUI Mriežka

- Pri návrhu treba dodržiavať
  - Vizuálnu konzistenciu
    - Vnútornú
      - Pre prvky používať rovnaké pravidla, dodržiavať mriežku
    - Vonkajšiu
      - Daná platformou nasadenia

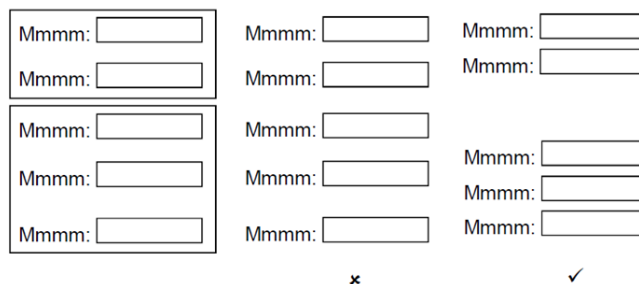


V jednotlivých grafických príkladoch správneho použitia toho- ktoreho pravidla sú nesprávne návrhy označené X a správne "fajkou".



## Metódy návrhu GUI Mriežka

– Vzťahy medzi prvkami



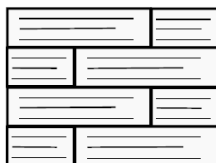
Súvisiace prvky treba dať logicky bližšie k sebe

## Metódy návrhu GUI

### Mriežka

#### – Navigačné vodítka

- Upútať pozornosť na začiatok
- Následná navigácia na menej významné prvky
- Tok navigácie má zodpovedať bežnému postupu užívateľa pre aplikáciu



x



✓

## Metódy návrhu GUI Mriežka

– Úspora vzhľadu

- Minimalizovať počet prvkov
- Aby nerozptyľovali pozornosť
- Radšej zoskupiť do ďalšieho okna

A diagram showing a cluttered GUI layout. It features multiple overlapping text labels and input fields. Labels include 'MMM', 'NNN', and 'xxx:'. There are several empty input fields and buttons scattered across the space, creating a busy and unorganized appearance.

✘

A diagram showing a clean GUI layout. It features a single window with a title bar labeled 'NNNN'. Inside the window, there is a label 'MMMM' and three 'xxx: \_\_\_\_\_' input fields. Two buttons are located at the bottom right of the window. This layout is organized and easy to read.

✔

## Metódy návrhu GUI Mriežka

### – Čitateľnosť

- Text musí byť prirodzene čitateľný
- Správna (vhodná) typografia
  - 1 až 3 typy písma
  - Zvýrazňovanie (bold, italic)
  - Max 3 veľkosti
- Uloženie textu
  - Zarovnanie, medzery, odsadenie, farba, dĺžka riadku

~~Whenever your local SMS Administrator sends you an actual software Package, the SMS Package Command Manager will appear (usually in network login time) displaying the available Package(s). The following screenshots display scenes similar to what you will see when you receive an actual SMS Package.~~

~~To start the Administration, click the "Get Package" button.~~

## Metódy návrhu GUI Mriežka

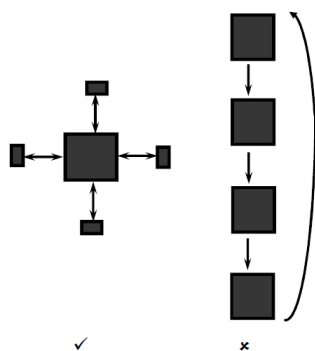
– Výstižné a viditeľné obrázky / ikony

- Dizajn obrázkov a ikon je veľmi náročný vzhľadom na sociálne, kultúrne a vekové pozadie užívateľov
- Bezpečnejšie je vždy doplniť vysvetľujúci text
- Dodržať konzistenciu vzhľadu aj pre obrázky
- Podobne obrázky-akcie zoskupovať



## Metódy návrhu GUI Mriežka

- Jednoduchá a rýchla navigácia/ ovládanie
  - Vyhnúť sa dlhým prechodom medzi úrovňami



## Metódy validácie a hodnotenia

- **Validácia- spätná väzba veľmi dôležitá:**
  - “Your Best Guess Is Not Good Enough”, “The User Is Always Right”, “Users Are Not Designers”, “Designers Are Not Users”,...
- **Metódy**
  - Dotazníky
  - Interview
  - Inšpekcia
  - Priame porovnanie
  - Spojité vyhodnocovanie (používateľská spätná väzba)

Okrem dodržania pravidiel návrhu pomocou niektorej zo známych metód je dôležité vybrať správny postup validácie návrhu. Skoro vždy sa vyskytnú menšie či väčšie chyby v návrhu, ktoré je potrebné včas odhaliť, analyzovať a korigovať. Pri hľadaní týchto nedostatkov až chýb je dôležitá aj kooperácia s užívateľmi, a to v rôznych fázach návrhu. V ďalšom teda budú opísané základné metódy validácie návrhu so svojimi, výhodami, nevýhodami a možnosťami vhodného použitia.

## Metódy validácie a hodnotenia

- Dotazníky
  - Pokrytie veľkej skupiny používateľov
  - Ľahšie hromadne spracovanie
  - Výsledok sa ľahšie kvantifikuje
  - Nevyžaduje prítomnosť pozorovateľa
  - Nákladná príprava
  - Neumožňuje adaptívne meniť a dopĺňať otázky

Poznámka: Zeleným sú uvedené výhody a červenou nevýhody tej -ktorej, to platí aj pre ďalšie metódy validácie



## Metódy validácie a hodnotenia

- Interview
  - Otázky sú kladené používateľovi
  - Zamerané na konkrétne aspekty
  - Je možné meniť a dopĺňať otázky podľa odpovedí
  - Časovo náročné
  - Nepresnosti:
    - Kvôli používateľovi (nemusí vedieť objektívne posúdiť a hodnotiť)
    - Kvôli pýtajúcemu (nesprávne otázky, resp. sugestívne-  
navádzajúce na odpoveď)

## Metódy validácie a hodnotenia

- Inšpekcia
  - Systém je skúšaný (používaný) zo strany návrhárov, nie používateľov
  - Odstránenie funkčných a zjavných chýb
  - Menej nákladná
  - Vývojári nie sú cieľová skupina, majú inú úroveň aj pohľady
  - Takáto metóda nie je veľmi presná

## Metódy validácie a hodnotenia

### – Priame porovnanie

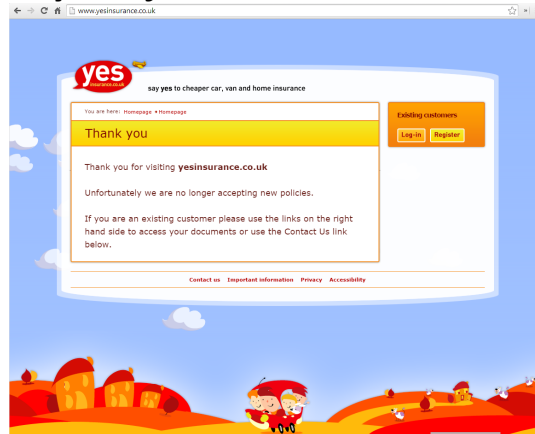
- Hodnotiteľ sleduje používateľa pri ovládaní systému
- Dobré na odstránenie častých a hrubých chýb návrhu
- Vylepšenia: „hlasné“ premýšľanie používateľa pri riešení úlohy (nie vždy je to možné)
- Prítomnosť hodnotiteľa
- Nie je vhodné pre veľa používateľov

### – Spojité vyhodnocovanie

- Používateľská spätná väzba v jednotlivých krokoch
- Odhalenie koncepčných chýb od začiatku
- Časovo náročné

## Typické chyby dizajnu

- **Neprimeraný dizajn**



Zdroj: Jana Melišová, World Usability Days 2012

Na nasledovných príkladoch sú uvedené a popísané niektoré z očividných chýb návrhu reálnych internetových stránok.

## Typické chyby dizajnu

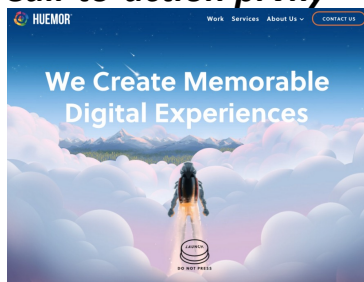
- **Zlá, resp. žiadna navigácia**
- **Nevhodné technológie**
  - Niektoré technológie nemusia byť vždy dostupné, a nie je poskytnutá alternatíva ak nie sú k dispozícii
- **Preplnený a neadekvátny obsah**



Zdroj: Jana Melišová, World Usability Days 2012

## Typické chyby dizajnu

- **Dlhé texty**
  - *Len málo kto ich dočíta do konca*
  - *Strata podstaty*
  - *Otrávený používateľ*
- **Chýbajúce Call-to-action prvky**



## Typické chyby dizajnu

- **Neštandardné alebo nevýrazné linky**
  - *Kliknite tu!!!!!!*
  - *Najčastejšia linka na SK : **tu***
- **Povinná (zbytočná) registrácia**
- **Úmorné formuláre**
- **Mobilný web z nutnosti len aby bol / alebo žiaden**

## Referencie/zdroje

- I.Kolingerová, Grafický návrh obrazovky
- I.Kolingerová, Rozhraní člověk – počítač (HCI) – koncepce
- Mullet & Sano, Designing Visual Interfaces, Prentice Hall
- Robin Williams, Non-Designers Design Book, Peachpit Press
- Vanda Benešová , Human computer interaction
- *Jana Melišová, World Usability Days 2012*
- *Jakob Nielsen, Usability Engineering, published by Morgan Kaufmann, 1993*
- *Alan Dix, Human–Computer Interaction. Third Edition.*