

# Testování uživatelského rozhraní a WWW stránek

Radek Mařík

CA CZ, s.r.o.

September 14, 2007



## 1 Testování uživatelského rozhraní

- Typy uživatelského rozhraní
- Chyby uživatelského rozhraní
- Dobré uživatelského rozhraní

# Výhody používání GUI <sup>[Ger97]</sup>

- Standardní "look and feel"
- Svou flexibilitou pokrývá většinu aplikací.
- Umožňuje bezproblémovou integraci zakázkových a již vytvořených aplikací.
- Výběr ovládání pomocí klávesnice a myši.
- Lepší porozumění uživatelem pomocí násobných oken.
- Mnohem vyšší řiditelnost ze strany uživatele.

# Formuláře <sup>[Ger97]</sup>

- Hierarchické uspořádání
- Typicky přístup pouze k jednomu formuláři
- Navigace:
  - **Menu:**
    - dvou-úrovňové menu s 15 položkami na menu
    - přístup k více než 200 obrazovek
  - **Příkazová řádka:** Pomocí příkazových řádek na navigaci formulář-formulář
- Pole formuláře přístupná v pomoci sekvence tabulátorů.
- Příklady
  - Telnet formuláře (Unix, MS Windows, z/OS)
  - ISPF (z/OS)

# GUI - grafické uživatelské rozhraní <sup>[Ger97]</sup>

- Práce s mnoha okny najednou.
- Ovládací prvky:
  - menu
  - toolbary
  - tlačítka
  - klávesové zkratky
- Zahrnuje funkcionality formulářů
- Přidává další objekty:
  - radio tlačítka,
  - posuvné listy,
  - check boxy
- Položky mají rovněž definované pořadí tabulace (1D)
- Uživatel může kdykoliv použít myš pro urychlení navigace (2D).

# Chyby uživatelského rozhraní - funkcionalita <sup>[KFN93]</sup>

## Funkčnost

- Program má problém s funkcí, jestliže
  - nedělá něco, co by měl dělat nebo
  - to dělá nevhodně, zmatečným způsobem či neúplně,
  - lze některé operace provést obtížně,
- Konečná definice, co se “předpokládá” od programu, žije pouze v mysli uživatele.
- Všechny programy mají problémy s funkcí vzhledem k různým uživatelům.
- Funkční problém je chybou, pokud očekávání uživatele jsou rozumná.

# Chyby uživatelského rozhraní - vstupy <sup>[KFN93]</sup>

## Komunikace

- Jak lze nalézt, jak používat daný program?
- Jaká je nápověda, pokud uživatel udělá chybu či spustí *< Help >*?

## Struktura příkazů

- Jak snadné je ztratit se v programu?
- Jaké chyby uživatel dělá, kolik je to stojí času a proč?

## Chybějící příkazy

- Co chybí?
- Nutí program uživatele přemýšlet nějakým pevným, nepřirozeným nebo neefektivním způsobem?

# Chyby uživatelského rozhraní - výstupy <sup>[KFN93]</sup>

## Výkonnost

- Rychlost je základem interaktivního softwaru.
- Cokoliv vyvolává v uživateli pocit, že program pracuje pomalu, je problém.

## Výstup

- Získá uživatel, co potřebuje?
- Mají výstupní reporty smysl?
- Může uživatel přizpůsobit výstup svým potřebám?
- Lze přesměrovat výstup podle výběru uživatele na monitor, tiskárnu, či do souboru daného formátu?



# Potíže s testováním I <sup>[Ger97]</sup>

## Software je řízen událostmi

- Existuje nepřeberné množství vstupů, které uživatel může provést.
- Každá událost se vykonává v určitém kontextu.
- Pohyb mezi jednotlivými vlastnostmi aplikace:
  - nová vlastnost se musí vyrovnat s potenciálně neznámým kontextem,
  - předchozí vlastnost může být opuštěna v částečně neuzavřeném stavu.

## Neočekávané události

- Příklady:
  - dialog oznamující, že tiskárna byla odpojena
  - zpracování zprávy z databáze, žádající obnovení prezentace dat v klientské aplikaci.
- Mohou nastat kdykoliv
- Vyžadují speciální testovací řadiče.

# Základní otázky <sup>[Pat02]</sup>

- **Použitelnost:** míra nakolik je interakce mezi uživatelem a rozhraním
  - odpovídající,
  - funkční a
  - efektivní
- **Ergonomika:** vědní odvětví zabývající se návrhem a konstrukcí předmětů tak,
  - aby se s nimi pracovalo lehce a
  - aby byly funkční
- Kdo bude potenciálním zákazníkem testovaného softwaru?
- Ačkoliv existuje řada přísných vědeckých teorií, v praxi výsledek závisí na zkušenostech softwarového týmu.

# Zásady uživatelského rozhraní <sup>[Pat02]</sup>

- Dodržuje standardy nebo zásady.
- Je intuitivní.
- Je konzistentní.
- Je flexibilní.
- Je pohodlné.
- Je správné.
- Je užitečné.

# Zásady uživatelského rozhraní I <sup>[Pat02]</sup>

## Dodržuje standardy nebo zásady

- Addison-Wesley: Macintosh Human Interface Guidelines
- Microsoft Press: Microsoft Windows User Experience
- Každá větší firma má své interní předpisy.
- Detaily o tom
  - jak má software běžící na příslušné platformě *vypadat*,
  - jak se má *chovat*.
- Princip **Look and feel**.

## Je intuitivní

- Je rozhraní zřetelné, není příliš vlezlé, není příliš zahlcené?
- Je dobře uspořádáno a vhodně rozloženo?
- Neobsahuje příliš funkcí?

# Zásady uživatelského rozhraní II <sup>[Pat02]</sup>

## Je konzistentní

- Stejná operace se provádí stejným způsobem i v jiném programu.
- Klávesové zkratky a výběry z nabídek.
- Terminologie a pojmenování.
- Projevy softwaru (stejná úroveň odezev - legrační, formální, atd.)
- Umístění tlačítek a jejich klávesové ekvivalenty
- **Základní problém velkých firem s řadou akvizicí.**

## Je flexibilní

- Může si rozhraní uživatel přizpůsobit?
- Vyšší flexibilita typicky znamená vyšší složitost:
  - podstatně komplikovanější diagram stavů a přechodů,
  - násobné možnosti, jak ukončit či přeskočit stavy,
  - množství konfigurací vstupu a výstupu dat.

# Zásady uživatelského rozhraní III <sup>[Pat02]</sup>

## Je pohodlné

Pojem pohodlí je velmi subjektivní a velmi obtížně se kvantifikuje.

- Náležitost - působení správným dojmem podle kontextu aplikace (např. volba barev)
- Software by měl být připraven na chyby uživatele a řádně je ošetřovat.
- Rychlost zpracování by měla brát zřetel na schopnosti uživatele
  - chybové hlášky zobrazovat po dostatečně dlouhou dobu.

## Je správné

Ověřujeme, zda rozhraní dělá to, co má.

- Rozdíly mezi marketingovými materiály a programem.
- Správné jazykové formulace a pravopis.
- Konzistentní média (ikony, obrázky, zvuky, videa).
- WYSIWYG (What you see is what you get) - dostanete přesně to, co vidíte.

# Literatura I



Paul Gerrard.

Testing GUI applications.

In *EuroSTAR'97*, pages 24–28, Edinburgh UK, November 1997.



Cem Kaner, Jack Falk, and Hung Quoc Nguyen.

*Testing Computer Software*.

International Thomson Computer Press, second edition, 1993.



Ron Patton.

*Testování softwaru*.

Computer Press, Praha, první edition, 2002.