

Systematický vývoj UI

Josef Pavlíček

Software eXperience desing team

Sun Microsystems

Obsah

- Úvod
- Použití případů užití a scénářů
- Designová specifikace
- Tvorba funkčních prototypů
- Testování

Úvod

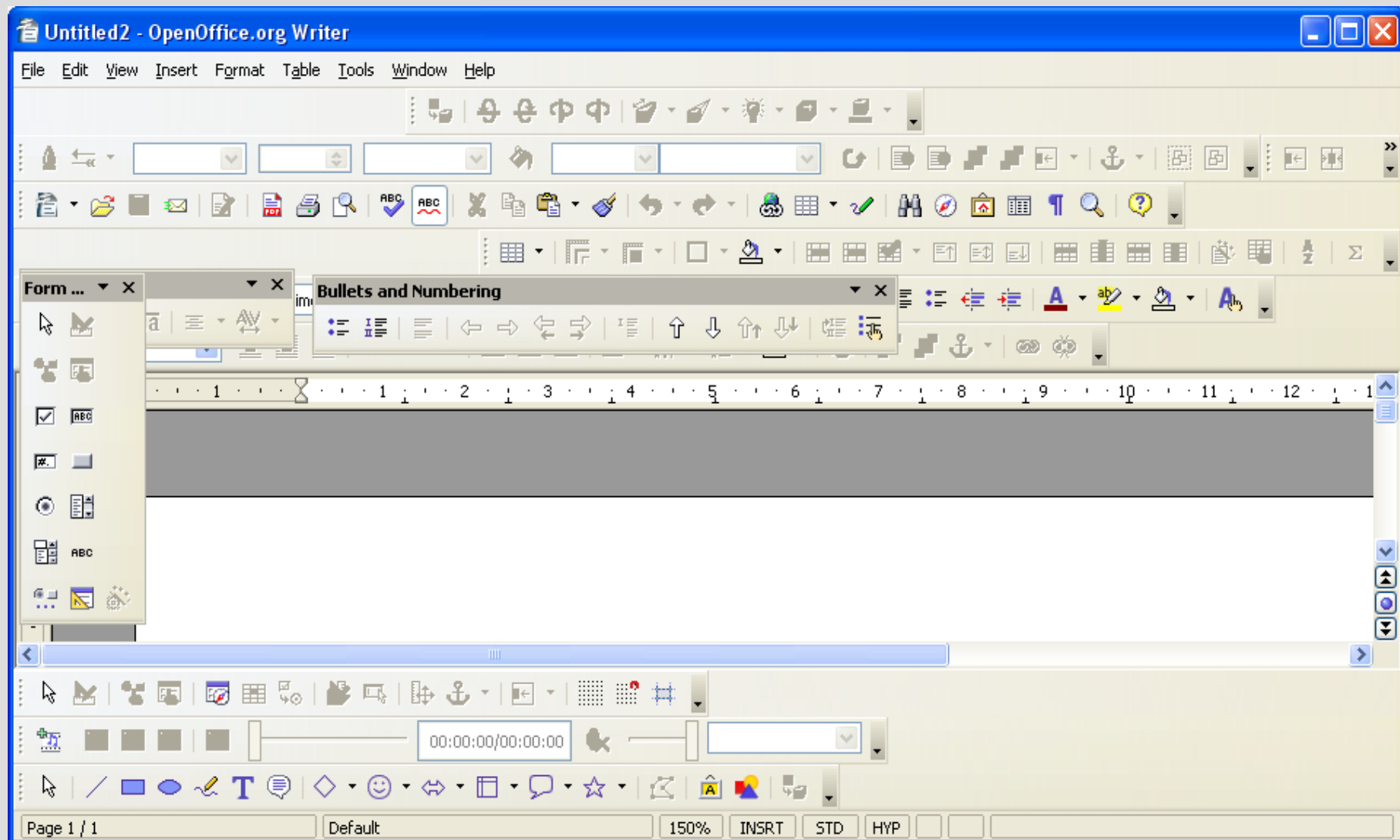
- Proč je dobré UI důležité
- Jak zjistit „to“ důležité
- Základní chyby při návrhu UI
- Jak správně UI navrhovat

Proč je dobré UI důležité

- Protože pomáhá uživateli používat produkt
- Uživatel se při používání produktu necítí hloupě
- Uživatel si snadno zapamatuje, jak produkt ovládat

Proč je dobré UI důležité

...příklad



Jak zjistit „to“ důležité

- Nalézt hlavní případ užití a jeho scénář
- Snažit se UI navrhovat pro konkrétního uživatele podle jeho konkrétních případů užití
- UI návrh má pomoci realizovat
 - Nalezené případy užití
 - Ty případy užití, které jsou nejčastější
 - Nejlépe ne všechny

Základní chyby při návrhu

- Kopírování konkurence
 - Přebereme UI i s chybami
 - Nevíme kde jsou UI chyby
 - Nemáme Know How těch, které kopírujeme
- Snaha realizovat pokud možno všechny požadavky uživatelů

Základní chyby při návrhu

- AHOJ HOMERE, tak co nová kára ?!
 - Jde to
 - Ale sem tam
 - Něco chybí
 - A někdy
 - Něco
 - Je navíc



Jak správně UI navrhovat

- Nalezením hlavních případů užití
- Za pomoci scénářů
- Za pomoci vhodných nástrojů
 - Výzkum (research)
 - Modelů a specifikací
 - Testování použitelnosti
- Je potřeba mít zkušenosti

Použití případů užití a scénářů

- Příklad užití popisuje chování uživatele se systémem
 - Je psán z jeho pohledu
 - Co uživatel od systému očekává
- Příklad užití neobsahuje řešení problému

Použití případů užití a scénářů

- Scénář popisuje chování systému směrem k uživateli
 - Je psán z pohledu systému
 - Popisuje jak na daný podmět systém reaguje
- Obsahuje nástin řešení problému

Designová specifikace

- Na základě případu užití a scénářů designer vytváří „designové specifikace“
 - Specifikace obsahuje detailní popis UI
 - Popisuje pomocí schématických modelů navržené rozhraní
 - Design na papíře a slovní popis
 - ASCII design
 - Grafický design
 - Dynamické modely

Tvorba funkčních prototypů

- Právě tvorba „funkčních prototypů“ je ve skutečnosti cesta trnitá
 - Prototyp stojí mnoho času
 - Vede často k přesvědčení, že z funkčního prototypu je jen krůček k finálnímu řešení
 - Tvůrce prototypu musí umět programovat
 - Problém se zaběhlými stereotypy programátora
 - Intuitivní tíhnutí k nepřehlednému řešení
- `for (int i=10;i<0;i++){while (x_word==true) System.out.print("pisu x_word"); System.out.printl....`

Tvorba funkčních prototypů

- Přesto výhoda „funkčních prototypů“ je zásadní
 - Ideální jsou nástroje umožňující modelovat chování systému, aniž by vyžadovali programátorské zkušenosti
 - Ideální jsou takové, jenž umožní rozpohybovat návrhy
 - Obrázovek
 - Tlačítek
 - Atd .

Tvorba funkčních prototypů

- A proto nyní ukážeme tvorbu prototypu pomocí nástroje Macromedia Flash

Závěr

- Děkuji za Váš čas a hurá do volné diskuse



Josef Pavlíček

Software eXperience desing team

Sun Microsystems