

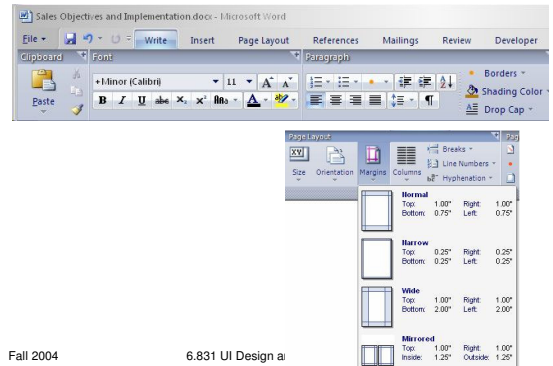
Lecture 14: Heuristic Evaluation

Fall 2004

6.831 UI Design and Implementation

1

UI Hall of Fame or Shame?



Fall 2004

6.831 UI Design a

2

Nielsen's Heuristics

- **Meet expectations**
 1. Match the real world
 2. Consistency & standards
 3. Help & documentation
- **User is boss**
 4. User control & freedom
 5. Visibility of system status
 6. Flexibility & efficiency
- **Errors**
 7. Error prevention
 8. Recognition, not recall
 9. Error reporting, diagnosis, and recovery
- **Keep it simple**
 10. Aesthetic & minimalist design

Fall 2004

6.831 UI Design and Implementation

3

Heuristic Evaluation

- Performed by an expert
- Steps
 - Inspect UI thoroughly
 - Compare UI against heuristics
 - List usability problems
 - Explain & justify each problem with heuristics

Fall 2004

6.831 UI Design and Implementation

4

How To Do Heuristic Evaluation

- Justify every problem with a heuristic
 - “Too many choices on the home page (Aesthetic & Minimalist Design)”
 - Can’t just say “I don’t like the colors”
- List every problem
 - Even if an interface element has multiple problems
- Go through the interface at least twice
 - Once to get the feel of the system
 - Again to focus on particular interface elements
- Don’t limit yourself to the 10 heuristics
 - We’ve seen others: affordances, visibility, Fitts’s Law, perceptual fusion, color principles
 - But the 10 heuristics are easier to compare against

Fall 2004

6.831 UI Design and Implementation

5

Example

Shopping Cart Contents

Welcome, Ben Bitdiddle.

You have 2 item(s) in your shopping cart.
To remove an item, check "Remove" box & click "Recalculate".
Shipping Calculator below

There is a problem with your order.

Product #	Description	Quantity	Unit Price	Ext Price
323022	Finalice_Clean_Plus_Version_4.0_Retail ""(Free_2nd_Day)""	<input type="checkbox"/>	\$61.00	\$61.00
60099-21	Corsair_VS1GBx400_1GB_x4_DDR400_PC3200_Value_Select Memory Retail (out of stock) View Product	<input type="checkbox"/>	\$179.00	\$179.00

Subtotal: **\$240.00**
[View Cart](#) [Clear Cart](#)

For more information about tax, please [click here](#).

Shipping Promotion details: Please read. [Check Out](#)

Note: Discount will be applied during check out
Coupon Code: [Apply](#) [Calculate shipping charge](#)

Ship to Zip Code: [Calculate shipping charge](#)

Have not made up your mind? Save all the items in your shopping cart.
Cart Title: [Save Shopping Cart](#)

Return to old shopping cart.
Cart Name: [Load Shopping Cart](#)

Fall 2004

6.831 UI Design and Implementation

6

Heuristic Evaluation Is Not User Testing

- Evaluator is not the user either
 - Maybe closer to being a typical user than you are, though
- Analogy: code inspection vs. testing
- HE finds problems that UT often misses
 - Inconsistent fonts
 - Fitts’s Law problems
- But UT is the gold standard for usability

Fall 2004

6.831 UI Design and Implementation

7

Hints for Better Heuristic Evaluation

- Use multiple evaluators
 - Different evaluators find different problems
 - The more the better, but diminishing returns
 - Nielsen recommends 3-5 evaluators
- Alternate heuristic evaluation with user testing
 - Each method finds different problems
 - Heuristic evaluation is cheaper
- It’s OK for observer to help evaluator
 - As long as the problem has already been noted
 - This wouldn’t be OK in a user test

Fall 2004

6.831 UI Design and Implementation

8

Formal Evaluation Process

1. Training
 - Meeting for design team & evaluators
 - Introduce application
 - Explain user population, domain, scenarios
2. Evaluation
 - Evaluators work separately
 - Generate written report, or oral comments recorded by an observer
 - Focus on generating problems, not on ranking their severity yet
 - 1-2 hours per evaluator
3. Severity Rating
 - Evaluators prioritize all problems found (not just their own)
 - Take the mean of the evaluators' ratings
4. Debriefing
 - Evaluators & design team discuss results, brainstorm solutions

Fall 2004

6.831 UI Design and Implementation

9

Severity Ratings

- Contributing factors
 - Frequency: how common?
 - Impact: how hard to overcome?
 - Persistence: how often to overcome?
- Severity scale
 1. Cosmetic: need not be fixed
 2. Minor: needs fixing but low priority
 3. Major: needs fixing and high priority
 4. Catastrophic: imperative to fix

Fall 2004

6.831 UI Design and Implementation

10

Evaluating Prototypes

- Heuristic evaluation works on:
 - Sketches
 - Paper prototypes
 - Unstable prototypes
- "Missing-element" problems are harder to find on sketches
 - Because you're not actually using the interface, you aren't blocked by feature's absence
 - Look harder for them

Fall 2004

6.831 UI Design and Implementation

11

Writing Good Heuristic Evaluations

- Heuristic evaluations must communicate well to developers and managers
- Include positive comments as well as criticisms
 - "Good: Toolbar icons are simple, with good contrast and few colors (minimalist design)"
- Be tactful
 - Not: "the menu organization is a complete mess"
 - Better: "menus are not organized by function"
- Be specific
 - Not: "text is unreadable"
 - Better: "text is too small, and has poor contrast (black text on dark green background)"

Fall 2004

6.831 UI Design and Implementation

12

Suggested Report Format

- What to include:
 - Problem
 - Heuristic
 - Description
 - Severity
 - Recommendation (if any)
 - Screenshot (if helpful)

12. Severe: **User may close window without saving data** (error prevention)

If the user has made changes without saving, and then closes the window using the Close button, rather than File >> Exit, no confirmation dialog appears.

Recommendation: show a confirmation dialog or save automatically

