

IS 214 Needs Assessment and Evaluation of Information Systems

Usability Inspection

© Copyright 2001 Kevin McBride

Usability Inspection

- Examination and evaluation of usability aspects of a user interface
- Performed by usability specialists or domain experts, developers, writers, and/or users
- Using the knowledge and judgment of the inspectors based on their training and experience with design principles

Usability Inspection - When

- Later stages of the design cycle
- After an initial design has been created
- After details of the user interface have been identified and prototyped

Usability Inspection - Objectives

- Identify potential usability problems
- Predict problems found by actual use
- Explain causes of usability problems
- Provide suggested design changes

Usability Inspection – Methods

- Guidelines / standards inspection
- Consistency inspection
- Pluralistic walkthrough
- Cognitive walkthrough
- Heuristic evaluation

Guidelines / Standards Inspection

- Goal: assure conformance to UI standards
 - Open Source Foundation / Motif Guidelines
 - Sun Java UI Guidelines
 - Microsoft Windows UI Guidelines
- Must learn hundreds of detailed standards for use and placement of controls, capitalization of text, window management, and so on.
- Include programmers who have control over the implementation of these details

Consistency Inspection

- Goal: ensure consistent design across UIs
- Examine similar UI components side-by-side to ensure the same problem is not being solved in different ways. Choose the best method.
- Include members from each design team
- Define guidelines to ensure future consistency
- Create reusable internals to assist consistency

Pluralistic Walkthrough

- Include users, developers, and UI specialists
- Follow a scenario through hardcopy UI
- All participants take the role of user
- Before discussion, each individual writes down the actions he/she would take for the scenarios
- Discuss real users' actions first

Cognitive Walkthrough

- Goal: evaluate ease of learning by exploration
- Can be done by individual or group
- Input: sequence of actions for each task
- Examine each action and try to tell a credible story as to *why* the user would do it
- Identify where the user would not know what to do, where expectations would be violated

Heuristic Evaluation

- Goal: identify potential usability problems
- Inspectors can be usability specialists, domain specialists, developers, writers, or others
- Each individual first inspects separately
- Evaluate UI against “rules of thumb” (heuristics) for good user interface design
- Write down potential problems and consolidate with other inspectors

Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention

Heuristics (continued)

- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation

Visibility of system status

- Keep the user informed of what is going on
- Provide appropriate feedback
- Respond within a reasonable time
- Examples:
 - Norton AntiVirus Progress indicator
 - Web browser status bar

Match system to real world

- Speak the user's language
- Avoid system-oriented terms
- Follow real-world conventions
- Present info in a natural and logical order
- Examples:
 - URLs vs. Yahoo! Catalog

User Control and Freedom

- Accommodate users who make mistakes
- Provide “emergency exits” to tasks
- Support undo and redo
- Examples:
 - Stop button, back button in web browsers
 - Can't undo after save in Office products

Consistency and Standards

- Follow platform conventions
- Words should always have the same meaning
- Actions should have the same result
- Examples:
 - Enter key for the default action in web pages
 - Selection in a drop-down list
 - Accelerator keys for cut, copy, and paste

Error Prevention

- Prevent problems from occurring
- Better than good error messages
- Examples:
 - Type ahead features in browsers
 - Auto-spell and correct features in word processors
 - Preventing invalid entries in text fields

Recognition rather than recall

- Make objects, actions, and options visible
- Minimize user memory load
- Move user data automatically
- Instructions should be visible with dialogs
- Examples:
 - Help window that covers dialog being explained
 - Wizards with explanation of how to do it
 - Command line interface

Flexibility and Efficiency

- Provide short-cuts for expert users
- Enable creation of repeating sequence of actions (macros)
- Examples:
 - Menu short-cut keys
 - Command line interface
 - Recording menu / key sequence as a macro
 - Wizards that save actions as commands

Aesthetic and Minimalist Design

- Don't include rarely-needed information
- Don't include rarely-used functions
- Examples:
 - Providing every option all at once
 - Only showing some / used menus in MS Office

Help Users with Errors

- Error message should be in plain language
- Precisely indicate the problem
- Suggest possible solutions
- Example:
 - "Network connection refused" in web browser

Help and Documentation

- Solve problems in the UI first
- Easy to search
- Focused on the user's tasks
- List concrete steps to be carried out
- Not too long

Heuristic Evaluation Exercise

- Divide into groups of three or four
- Each individual separately evaluate the sample system using the provided heuristics (30 min)
- Try to identify at least 10 problems
- Consolidate problems as a group (15 min)
- Write each problem on a post-it and record the number of the heuristic violated

Consolidation

- Choose a representative from your group
- Place problems on the board by heuristic
- What are the consistently identified problems?
- Which heuristics are violated the most?
- What design ideas have arisen?

Issues

- Not a replacement for testing with users
- Identify 10-50% of usability problems
- Identify less serious problems
- Can identify problems that aren't problems

What's Important

- Learn to see usability problems as you design
- Get others to help you look for problems
- Learn the heuristics and guidelines and how to apply them to your design
- Don't rely on heuristic evaluation alone