Heuristic Evaluation

Ananda Gunawardena

Carnegie Mellon University
Computer Science Department
Fall 2008
Background

- Heuristic evaluation is performed early in the development process to understand user requirements
- Heuristic evaluation addresses usability issues in the design
- Heuristic evaluation involves small sample of potential users of the product

Ref: http://www.useit.com/papers/heuristic/
Nielson Heuristics

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition vs. recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Error recovery
10. Help and documentation
H1: Visibility of System Status

- The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
- Simple and natural dialog
H2: Match between system and the real world

- The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
- In Mac, drag disk to trash can to eject – bad example
H3: User Control and Freedom

- Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
- Always have more than one choice available
H4: Consistency and Standards

- Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
- Use standard conventions like file, edit, view, insert (if applicable) in your toolbars.
H5: Error prevention

- Careful design which prevents a problem from occurring in the first place
H6: Recognition rather than recall

- Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
H7: Flexibility and Efficiency of Use

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
H8: Aesthetic and Minimalist design

- Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
H9: Help users recognize, diagnose, and recover from errors

- Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
H10: Help and documentation

- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.
H5: Error prevention. If users presses “No” by accident, the picture will be gone forever! Must include an “undo” action.

H2: (Good thing) Speaks the user’s language. “Yes/No” is better than “Ok/Cancel”

H8: (Good thing) Minimalist design.

This is your image. Do you want to keep it?

User clicks here
Exercises

For each of the following screens, indicate which heuristic is violated
Use one of the following

✓    ×    NA
Example

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition vs. recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Error recovery
10. Help and documentation
Example

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition vs. recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Error recovery
10. Help and documentation
Example

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition vs. recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Error recovery
10. Help and documentation
References

• How to conduct a heuristic evaluation method
Designing for a Great Mobile Experience
Things to Consider

• Be easy to use and fully functional in the absence of a keyboard and mouse
• Present information in a way that users can quickly digest it. Avoid congested screen designs
• Be able to launch and run in full screen mode
• Provide exit control when in full screen mode
• Be able to run in portrait and landscape modes
• Ensure that all user interface elements are visible and unclipped at 800x480
Things to Consider

• Ensure that dialog box height is less than 480 pixels
• Ensure that UI elements scale to accommodate range of display sizes and typical display resolutions
• Design the UI to support content consumption, not content creation
Optimizing Touch command and Control

- Make touch target at least 1cm x 1 cm
- Space targets to sufficiently ensure an accurate interaction layer
- Ensure that graphics in the target area maintains legibility
- Ensure that targets are not obscured by hand during interaction
- Provide larger targets if your application is used in a highly mobile situation
- Group targets by functionality
- Avoid placing targets at the edges of the display
Designing for keyboard Free experience

• Provide on-screen alternatives to commonly used keyboard shortcuts such as copy, paste, undo, save and open
• Provide an alternative to ctrl-alt-del
• Where text is needed offer auto complete or quick text
• Explore ways to use alternate input modalities such as speech
Designing for mouse free scenario

• Touch screen devices typically do not provide hover-like features
  – Do not rely on hover like features for critical tasks
  – Consider providing hover feedback with object focus after an item is tapped
  – If pop-ups are provided for hover feedback, then close them immediately
Replacing Right Click and more

- Minimize reliance on right click functionality
- Provide alternative ways to access a functionality that is typically right click
- Use “tapping” as a way to open context menus
- Use alternative methods to scrolling
- Ensure that drag-and-drop features are clear, quick and simple and requires no fine motor skills
Storyboarding

A first step towards making a usable computer application
What is storyboarding?

• Storyboarding is when you draw out each screen, view, or page of an application.
Why is storyboarding important?

• Your first shot at planning!
• So everyone can agree what the application will look like and how it will be structured
• Making a map of the structure will help the team divide and conquer
Each box represents a screen, and drawn inside each box is a rough design of how that screen will look.
Navigation

- Arrows drawn between boxes show how screens are connected
  Link between two pages

- Most lines represent links, or buttons a user clicks on.

- **Double-arrowed lines show a user can go BACK**
More than one path
Navigation

• What’s wrong with this storyboard’s navigation?
Navigation

- Missing link to yellow page.
- No way to go BACK

- Can a user re-start this application?
Navigation
A Photo Annotation System

• Suppose I’d like to develop a mobile application that takes pictures and allow me to annotate and share those pictures

• Here are the things I’d like to have
  – Take a picture
  – Save a picture
  – Remove a picture
  – Label/Annotate a picture
  – Find a picture
  – Send a picture
Organizing the App

• Main Menu
  – Some graphic
  – Take a Picture
  – Find a Picture
  – Remove a picture
• Take a Picture
  – Label, save, retake, do not save
• Find a picture
  – Type a keyword to search
  – Find a picture(s) matching the keyword
  – Select the picture
    • View
    • edit the label
• Remove a picture
  – Find a picture
  – Confirm to delete
  – Delete the picture