Design Heuristics and Evaluation

Rapid Evaluation
Heuristic Evaluation

- Another method for finding usability problems in a UI design
- Validation during design - does the proposed interface …
  - Implement all variations of every user task correctly?
  - Achieve all user requirements?
- A small set of evaluators examine the interface and judge its compliance against recognized usability principles (the "heuristics")
- Use Nielsen’s Heuristics
What is a Heuristic?

• “Experience-based techniques for problem solving, learning, and discovery” Wikipedia
  – Useful when exhaustive exacting work is impractical
  – Trial-and-error
  – Self educating
  – Examples include using experiential guidelines including …
    ▪ a rule of thumb, an educated guess, an intuitive judgment, or common sense
Who is Nielsen?

- Jakob Nielsen is a Danish usability consultant
- Developed the Discount Usability Engineering (DUE) model
  - **Simplify usability design methods** to encourage wide spread adoption by the development community
- Three techniques:
  - **Scenarios** – simple focused prototypes
  - Simplified **thinking aloud** – have a small sample of real users think out loud while they perform tasks
  - **Heuristic evaluation** – evaluate designs early using 10 simple usability guidelines
    - NOTE: these are quality evaluation measures, NOT design principles
Nielsen’s Usability Goals

- Learnability
- Memorability
- Efficiency
- Minimize errors (understandability)
- Satisfaction

Fundamental measures of usability quality
Nielson’s Heuristics
10 Usability Rules of Thumb

1. Visibility of system status
   - Always keep users informed about what is going on, through appropriate feedback within reasonable time

2. Match between the system and the real world
   - 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
Nielson’s Heuristics

3. User control and freedom
   - **Support undo and redo.** 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.

4. Consistency and standards
   - **Follow platform conventions.** Users should not have to wonder whether different words, situations, or actions mean the same thing.
Nielson’s Heuristics

5. Error prevention
   – Design to prevent problems from occurring - better than good error messages
   – Either eliminate error-prone conditions or check for them ....
   – … and present users with a confirmation option before they commit to the action

6. Help users recognize, diagnose, and recover from errors
   – Error messages should be expressed in plain language (no codes), precisely indicate the problem, and suggest a solution
Nielson’s Heuristics

7. Flexibility and efficiency of use
   - Mechanisms to allow for **efficient interaction** for **inexperienced** and **experienced** users
   - Mechanisms can be hidden for novices
   - Allow users to tailor frequent actions

8. Aesthetic and minimalist design
   - Dialogues should not contain irrelevant or rarely needed information
   - Every extra unit of information in a dialogue competes with the relevant units of information and diminishes understanding
Nielson’s Heuristics

9. Recognition rather than recall

- Minimize the user's memory load by making 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
Nielson’s Heuristics

10. 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.
Heuristic Evaluation Practice

- Let’s solve an online puzzle
  http://www.jigzone.com/
- Do a pair evaluation
  - Step 1: Choose a puzzle and become familiar with it
  - Step 2: Evaluate the usability by applying Nielson’s 10 heuristics
    - Fill out a table – for each applicable heuristic, describe the interface design problem
  - Dropbox – “Web Site HE”

<table>
<thead>
<tr>
<th>Task Action</th>
<th>Heuristic Violated</th>
<th>Defect Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Heuristic Evaluation: During

- Each **individual evaluator** inspects the interface **alone** and documents problems.
- The evaluators use a set of **typical usage scenarios** for a sample set of **realistic tasks**.
- Task scenarios are evaluated against a **checklist of recognized usability principles** (the heuristics).
- The **results** of the evaluation are **recorded** either as written reports from each evaluator **OR** …
- … the evaluators **verbalize their comments** to an observer as they go through the interface.
- The **session** for an individual evaluator lasts **one or two hours**, but can last longer.
Heuristic Evaluation: Evaluators

• Evaluators should go through the interface at least twice.
  – The first pass would be intended to get a feel for the flow of the interaction and the general scope of the system
  – The second pass then allows the evaluator to focus on specific interface elements while knowing how they fit into the larger whole

• It is acceptable to perform heuristic evaluation of low fidelity (paper) interfaces
The observer (or the "experimenter"): 

- Records the evaluator's comments about the interface, but does not interpret the evaluator's actions.
- As necessary, answers evaluator questions and may provide hints on using the interface.
- The evaluators should not be given help until they are clearly in trouble and have commented on the usability problem in question.
Heuristic Evaluation: Output

- After individual evaluations, evaluators (with observers) aggregate their findings to produce …
- A list of usability problems in the interface with references to those usability principles that were violated
  - Each problem is listed separately, even if from same element
  - Sufficient detail
- Evaluators can’t just say they don’t like it
- The “not liking it” needs to have a reference to the heuristics
Heuristic Evaluation: Debriefing

- Provide some **design advice AFTER** the evaluation
- The participants should include the evaluators, the observers, and design representatives
- The session
  - Discussions (brainstorming) of **possible redesigns** to address the major usability problems and general problematic aspects of the design
  - Also discuss the **positive aspects** of the design, since heuristic evaluation does not otherwise address this
In Class Evaluation

- Each team will have two observers, two evaluators for another team’s system
- Pre:
  - Each team needs to have each HTA task(5) documented
  - The checklist to be used is Nielson’s (that’s it)
  - Have the system ready for evaluation for the next class
- During (in class)
  - **Pass 1**: The evaluator will go through the system to be familiar with it and note any overall problems using the checklist that the observers write down
  - **Pass 2**: Then go through each task and note any problems using the checklist
  - The observer will answer questions
  - Use the “Heuristic Testing Worksheet” in myCourses to document issues
  - Evaluators work independently
In Class Evaluation

- **During (continued)**
  - Following the evaluation, debrief evaluator to discuss possible fixes and positive observations

- **After**
  - Team merges individual evaluations to create one problem list
    - Assign a severity priority
  - As a team brainstorm solutions and adjust the project plan
  - Submit an evaluation report to the “Deliverable 6: Heuristic Evaluation Notes” dropbox
    - The two original heuristic testing worksheets
    - The consolidated problem list with severity ratings
    - Summary of the teams problem analysis and plan forward
References

- Jakob Nielson’s Design Heuristics
  http://www.useit.com/papers/heuristic/heuristic_list.html

- Heuristic How-to