SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

User Interface Design & Implementation

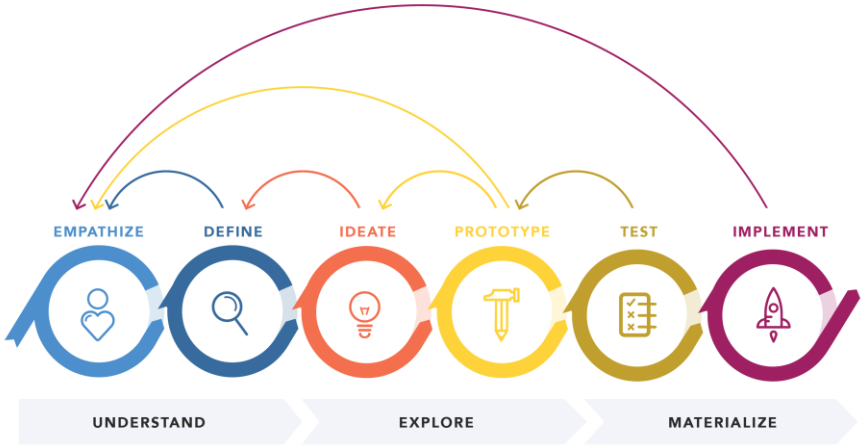
UI Evaluation

Week 4 – Lecture 9

UI Evaluation Methods

Some of the following slides adapted from HCII (CMU) course material.

- Expert Reviews
  - Heuristic Evaluation
  - Cognitive Walkthrough
- Usability Testing and Laboratories
  - Think-Aloud and Probes
- Acceptance Tests
  - Formative versus summative evaluation
- Model-based Evaluation
  - Empirical versus analytical methods
  - Model-Human Processor (MHP)
  - Goals, Operators, Methods and Selection Rules (GOMS)
  - Keystroke-Level Model (KLM)



DESIGN THINKING 101 NNGROUP.COM

Iterative Design Process

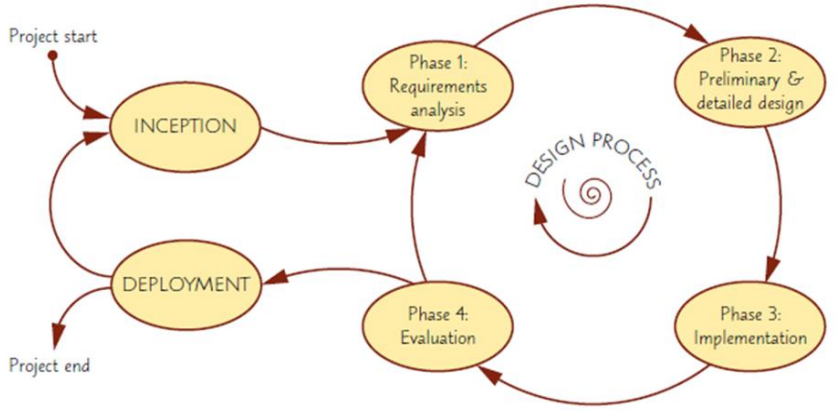


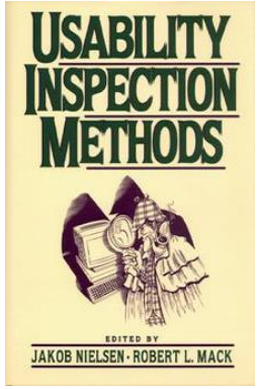
Image from *Designing the User Interface, 6th Ed.*

Why Evaluate?

- Evaluation: the act of determining the significance or worth of something, usually by careful appraisal and study
- A usability evaluation should always be done in relation to the goals of the project
- Ask:
  - What is the goal of the system?
  - What are the requirements of the system?
  - What kinds of usability are my objectives (what can I actually measure)?
  - What types of users do I expect to have?
  - Where are we in the development process?
- Why evaluate?
  - To catch “show stoppers”
  - Generate requirements
  - Discern if system meet the requirements
  - Discern if one system (or idea) is better for some task than another system
  - To accumulate knowledge of system design

Heuristic Evaluation

- One of many usability inspection methods (aka expert reviews)
- Involves a small team of evaluators to evaluate an interface based on recognized usability principles
- Developed by Jakob Nielsen and Robert Mark



Summary of Usability Inspection Methods: https://www.nngroup.com/articles/summary-of-usability-inspection-methods/

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 rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation

Resource (main article, sub-article and videos) on Usability Heuristics: https://www.nngroup.com/articles/ten-usability-heuristics/

Steps in Heuristic Evaluation

- Briefing
  - Teach the heuristics and heuristic evaluation method to non-usability experts
  - Introduce the user work domain to set the context for heuristic evaluation
- Evaluation by each individual
  - Two passes through interface:
    - Inspect flow of interaction (different states of the user-interface and the flow between them)
    - Inspect each screen, one at a time against heuristics
  - Each evaluator compares design with known usability principles or heuristics
  - Writes down all problems found in Usability Aspect Reports (UARs)
- Team process
  - All evaluators combine the problems found
  - Eliminate redundancies and clarify
  - Assign severity ratings
- Write report
  - Problems and priorities for next design iteration

Resources on Expert Reviews and Usability Evaluation:

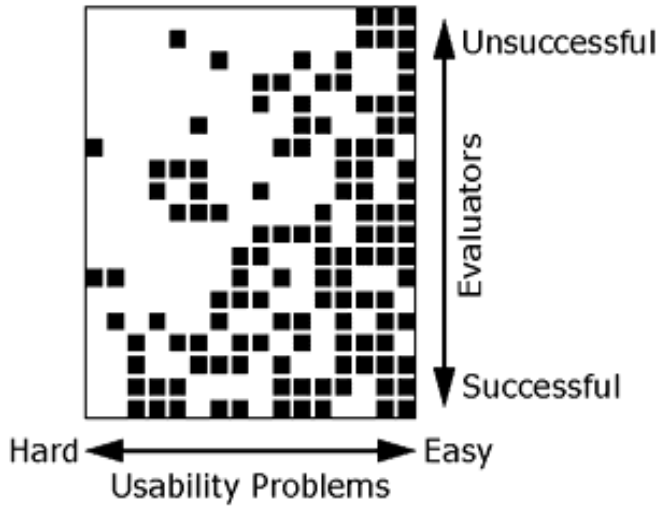
https://www.nngroup.com/articles/ux-expert-reviews/

https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/

https://www.nngroup.com/videos/heuristic-evaluation/

Team of evaluators

- Why more than one evaluator?
  - A single person is less likely to find all the usability problems
  - Different people find different usability problems
  - More successful (experienced; careful) evaluators may find both easy and hard problems
- Types of evaluators
  - Novice evaluators – knowledge of system but **no** usability expertise
  - Regular usability specialists – training/experience in usability but **not** specialized in the kind of interface being evaluated
  - Double specialists – expertise in **both** usability and the kind of interface being evaluated
- How many evaluators?
  - 4 or 5 are recommended by Nielsen



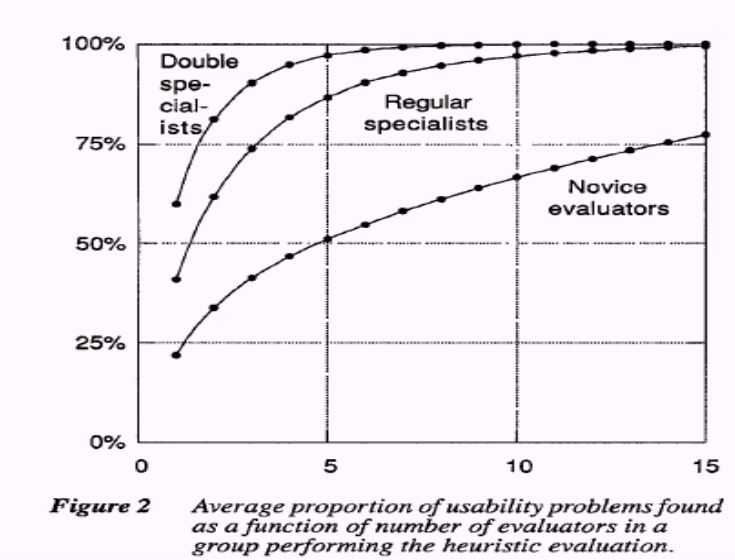


Figure 2 *Average proportion of usability problems found as a function of number of evaluators in a group performing the heuristic evaluation.*

Images from *Usability Inspection Methods*

Usability Aspect Report (UAR)

- What is a UAR and why?
  - A “better problem report to communicate with developers” (from *Usability Inspection Methods*)
  - Developers question the validity of recommendations
  - Evaluators go into solutions without describing the problem
    - Solutions may be too narrow
    - Interface design involves trade-offs

- UAR can be used to document findings in other evaluation methods besides heuristic evaluation

UAR fields to fill in

- **Identifier**
  - Initials of evaluator and UAR index number
  - Mark whether this UAR reports a problem or good aspect
- **Name**: succinct description of the usability aspect
- **Evidence**: supporting material for the aspect
  - E.g. which heuristic, what happened (for other evaluation methods), where in the user-interface?
- **Explanation**: your own interpretation
- **Severity**: your reasoning about the importance of this aspect
- **Solution**: if the aspect is a problem, include a possible solution and potential trade-offs
- **Relationships**: to other usability aspects (if any)

UAR explanation and severity rating fields

- Explanation
  - Two parts: the situation and why the situation is bad (e.g., how it violated heuristic)
  - Make sure you have evidence (in evidence field)
  - Explanation can include possible causes of the problem
  - Imagine that someone would disagree, what would you say?
- Severity
  - Combination of frequency, impact and persistence
  - To prioritize and allocate resources to fix problems
  - Should be done independently by all evaluators
  - Averaged across all evaluators in combined report

Article on UAR severity rating:

https://www.nngroup.com/articles/how-to-rate-the-severity-of-usability-problems/

Combined Aspect Report

Combined Aspect Statement	Heuristic (s) Violated	Individuals Referenced
Apply button-Users may use apply button in different ways. They can think it means “enter” or “save.”	Match, Consistency	jd-HE-01, sf-HE-11, el-HE-23
Hidden info-Window doesn’t scroll to show new information when it is entered. Some users then re-enter the information, and they are not told the information is entered twice.	Visibility	jd-HE-10, mk-HE-02
Etc…		

Cognitive Walkthrough

- Another usability inspection method
  - Evaluate user interfaces, especially their “first-time” use.
  - Good in early design spirals, from design specification, UI sketches, etc.
- Base on theory of learning by exploration (by Clayton Lewis, Peter Polson, Cathleen Wharton & John Rieman, University of Colorado):
  1. Assume user has a goal
  2. User searches for currently available controls to perform an action
  3. User selects the control that seems likely to make progress toward the goal
  4. User performs the action using the control
  5. User evaluates the result for evidence of progress
- Same basic method as other design or code walkthroughs
  - Present the proposed interface to reviewer(s) in a form that can be critiqued
  - Reviewer(s) evaluate the proposed interface in the context of one or more specific tasks from scenario(s)

Cognitive Walkthrough

- Preparation
  - Identify users & background knowledge
    - From contextual inquiries, interviews, surveys and other activities to understand your users
  - Create sample tasks
    - Choose tasks on the basis of frequency, importance, coverage of proposed user-interface
  - Describe interface in detail
    - E.g. paper prototype: sketches of screens and controls (layout, icons, labels, etc.)
  - Define correct action sequences for tasks
    - The interaction (sequence of actions and transitions between screens) to correctly perform the task
- Analysis
  - Can be done individually or by a group of reviewers (group is recommended)
  - For every action in an action sequence:
    - Tell credible success and/or failure stories
    - Record problem(s) with interface

Credible story for each action

- Four questions to evaluate the user's cognitive process as the user interacts with an interface

For each correct action to perform a task, answer the following 4 questions:

1. Will the user try to achieve the right effect?
2. Will the user notice that the correct action is available?
3. Will the user associate the correct action with the effect he or she is trying to achieve?
4. If the correct action is performed, will the user see that progress is being made toward solution of the task?

- Question 1 is about the user's next ***goal*** at any point in the task
- Question 2 is about ***perception*** of the correct action
- Question 3 is about ***comprehensibility*** of the correct action
- Question 4 is about ***perception*** and ***comprehensibility*** of the feedback

- Success requires passing all four criteria; failure can be in just one criterion.

UARs from Cognitive Walkthrough

- Evidence
  - Refer to success or failure stories, by step number and question number.
  - The evidence is that the reviewers came to consensus on this story.
- Explanation
  - A lot is already in the success or failure stories
  - Can add information about the system's rationale for doing it the way it is currently done.

Article and video on Cognitive Walkthrough:

https://www.interaction-design.org/literature/article/how-to-conduct-a-cognitive-walkthrough

https://www.youtube.com/watch?v=Edqjao4mmxM

Usability Testing and Laboratories

- A “traditional” usability lab would have two areas separated by a half-silvered mirror
  - for participants to do their work with the system under test
  - for testers and observers
- Participants should be chosen to represent the intended user groups, with attention to:
  - background in computing and experience with the task, and
  - motivation, education, and ability with the natural language used in the interface.
- Emphasize to the participant: *“We are testing the system/interface, we are not testing you.”*
- Participation should always be voluntary, and informed consent should be obtained
- Institutional Review Boards (IRB) often governs human subject tests

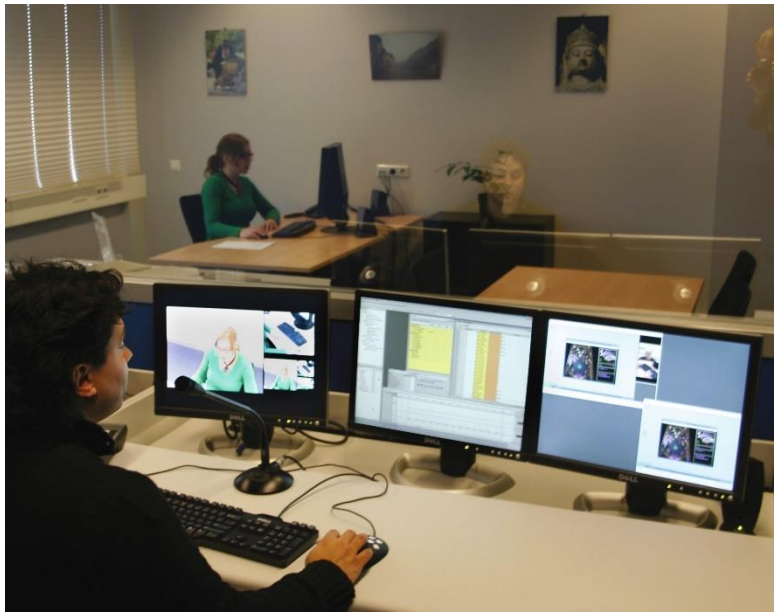


Image from *Designing the User Interface, 6th Ed.*

Usability testing in the field

Home or workplace with eye-tracker

Mobile application with eye-tracker

Interactive kiosk with eye-tracker

Mobile application with camera to record interaction

Images from *Designing the User Interface, 6th Ed.*

Think-Aloud and Probes

- Recording participants performing tasks
  - Invaluable for later review and for showing project team or management the problems that users encountered
  - Use caution to minimize interfering with participants' natural interaction
  - Invite users to think aloud about what they are doing as they are performing the task
- Concurrent Think-Aloud
  - Participants verbalise their thoughts as they work
- Retrospective Think-Aloud
  - Participants retrace their thoughts and actions after the work session
  - Watch video replay of their actions and/or eye-tracking videos
- Concurrent Probing
  - Participants verbalise their thoughts as they work
  - Researcher asks follow-up questions when something interesting was verbalised or done
- Retrospective Probing
  - Participants retrace their thoughts and actions after the work session
  - Watch video replay of their actions and/or eye-tracking videos
  - Researcher asks follow-up questions when something interesting was verbalised or done

Articles and videos on usability testing, think-alouds and probes:

https://www.usability.gov/how-to-and-tools/methods/running-usability-tests.html

https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/

https://www.nngroup.com/articles/thinking-aloud-demo-video/

https://www.youtube.com/watch?v=pxsJkAk_eo0

Formative versus Summative Evaluation

- Formative helps you "form" the system towards better usability
  - Emphasis on finding UI flaws to be addressed in subsequent iterations.
  - Usually "rich descriptive" data, more informal, relatively quick and not statistically generalizable.
  - E.g. Heuristic Evaluation, Cognitive Walkthrough, Think-aloud usability test.

- Summative helps you “summarize” the usability of the system
  - Assess the level of usability that has been achieved by the design.
  - Compare the level of usability achieved versus the previous iteration or competing design.
  - Tend towards formal, controlled experiments.
  - Accurate results that are statistically generalizable involve substantial time and effort.
  - UI design practitioners rarely do these, but if required, it’s important to do them right.
  - Customer might require these and are specified in the contract as acceptance tests.
  - E.g. time to learn specific functions, speed of task performance, rate of errors by users, retention of commands over time, subjective user satisfaction.

Empirical versus Analytical Evaluation

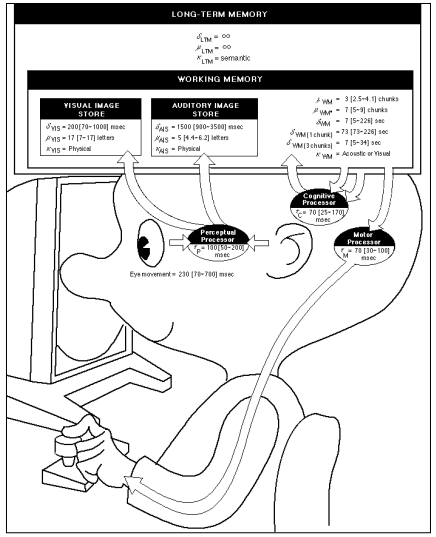
- Empirical
  - Collecting data from people.
  - From observations and experiments.
  - In laboratories and field studies.
  - E.g. Think-aloud usability studies, keystroke/clicks logging, surveys and questionnaires.

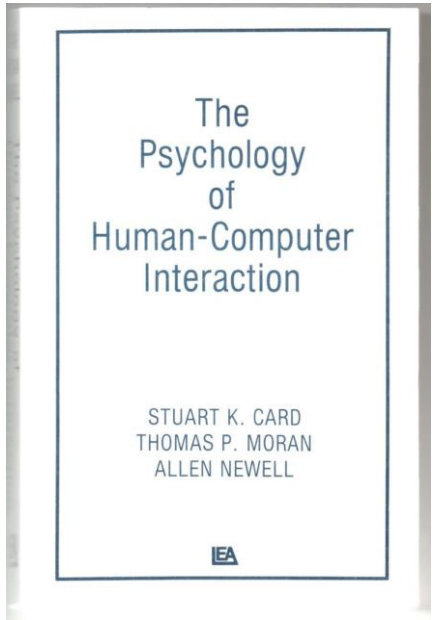
- Analytical
  - Working from theory/insight.
  - Theoretical analysis based on physical, psychological, or sociological theories.
  - Heuristics derived from experience.
  - E.g. MHP, GOMS, KLM, Cognitive Walkthrough, Heuristic Evaluation.

Model Human Processor

- Model Human Processor (MHP)
  - Helps us remember human constraints and make approximate predictions of user behaviour
  - Informs design decisions when empirical data is not available
  - Helps explain empirical data when it is available

- MHP is the foundation for follow-on models of human-computer interaction by Card, Moran and Newell
  - Goals, Operators, Methods and Selection Rules (GOMS)
  - Keystroke-Level Model (KLM)





MHP, GOMS and KLM

- Goals, Operators, Methods and Selection Rules (GOMS)
  - Family of modelling methods.
  - Consist of a set of **Goals**, a set of **Operators**, a set of **Methods** (sequence of operators) for achieving the goals, and a set of **Selections rules** for choosing among competing methods for goals.
- Keystroke-Level Model (KLM)
  - Simplest of GOMS-family member.
  - Pre-defined level of detail: keystroke level.
  - No representation of goals, methods or selection rules, just a sequence of operators that perform a task.
  - Input: a suite of benchmark tasks that are important to your design or evaluation.
  - Output: the time it would take a skilled user to perform these benchmark tasks.

Paper on KLM and software tool to perform GOMS/KLM analysis:

https://www.researchgate.net/publication/2848715_Using_the_Keystroke-Level_Model_to_Estimate_Execution_Times

http://cogulator.io/