

Week 12-2

Analytical Evaluation

SFWRENG 4HC3/6HC3 Human Computer Interfaces

** Slides adapted from previous and current instructors of COMPSCI/SFWRENG 4HC3/6HC3*

Week 12 Overview

- **Monday**
 - ~~Introduction to Evaluation~~
 - ~~Evaluation: Performance Modeling~~
- **Wednesday**
 - **Evaluation: Inspection**
 - **Evaluation: Heuristics**
- **Friday**
 - Evaluation with Users

Analytical Evaluation: Types

Performance modelling

- Fitts' Law: predict human's movements
- Hick-Hyman Law: predict human's decision time
- **Keystroke Level Modelling**

Inspections

- **Cognitive walkthrough**
- **Heuristic evaluation**

Cognitive Walkthroughs: Process

Step 1: Characteristics of typical users are identified and sample tasks are developed

- Assumptions about user population, context of use, task details; clear sequence of actions needed
- A **re-use** of user profiles, scenarios and task decompositions developed as part of the “Investigate” phase

Cognitive Walkthroughs: Process

Step 2: One or more experts use these profiles, scenarios, and tasks, to **step-by-step walk through** a task with the interface (sketch, paper prototype, working system, etc.)

At each step, ask **3 questions**

- Will the user **know the correct action**?
 - i.e., will the user know what to do in the interface to achieve the task?
- Will the user notice **how to do the correct action**?
 - Can users see the button or menu item, or be expected to know the hotkey, etc., for the next action?
 - Is it apparent when it is needed?
- Will the user **associate and interpret** the response from the action correctly?
 - Will users know from the feedback that they have made a correct or incorrect choice of action?

Cognitive Walkthroughs: Practice

Take 3-5 minutes to conduct a cognitive walkthrough for the task on **saving a document in Microsoft PowerPoint** (from the document page)

At each step, ask **3 questions**

- Will the user **know the correct action**?
 - i.e., will the user know what to do in the interface to achieve the task?
- Will the user notice **how to do the correct action**?
 - Can users see the button or menu item, or be expected to know the hotkey, etc., for the next action?
 - Is it apparent when it is needed?
- Will the user **associate and interpret** the response from the action correctly?
 - Will users know from the feedback that they have made a correct or incorrect choice of action?

Cognitive Walkthroughs: Summary

Can be integrated **throughout**
your whole design process

A simple way to keep yourself
grounded in the practicalities of
the interface

Inspection: Heuristic Evaluation

Heuristic refers to **a rule of thumb** - a principle that is a shortcut for solving a problem or making decisions

Not always right/true, but some shortcuts

Inspection: Heuristic Evaluation

Design Heuristics refer to broad **usability guidelines** that can guide a developer's design efforts

- Derived from common design problems across many systems
- Several researchers and practitioners have developed different sets of heuristics (e.g., domain specific)

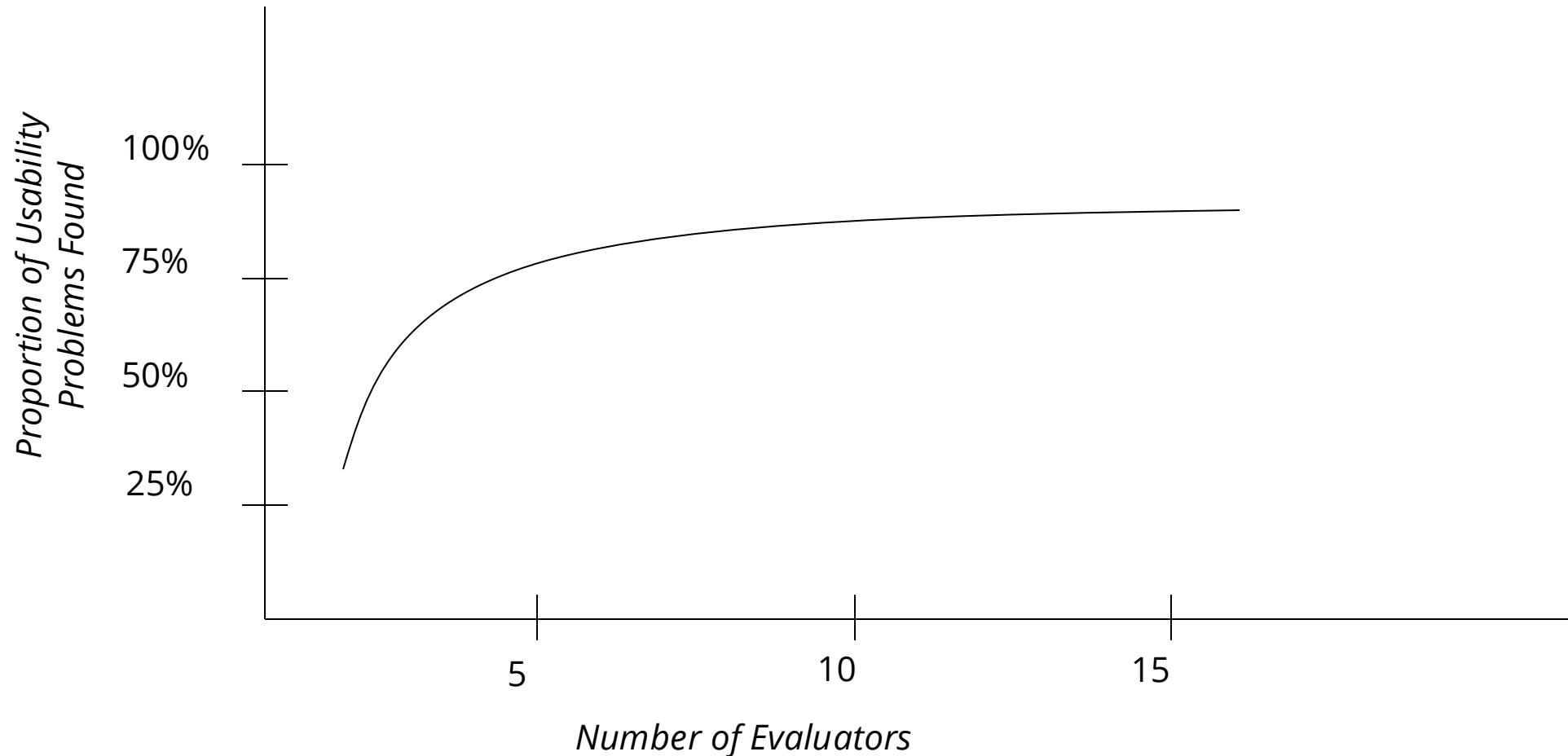
Heuristic Evaluation: Process

Systematic inspection of an interface design to see if an interface complies with a set of usability heuristics, or usability guidelines

General process

- 3-5 inspectors (usability engineers, experts...)
- Inspect interface in isolation (~1-2 hrs for simple interfaces)
- Results are aggregated afterwards
 - Single evaluator catches ~35% usability problems
 - 5 evaluators catch ~75%

Heuristic Evaluation: Process



Heuristic Evaluation: Process

Usually use scale from 0 (not a problem), to 4 (OH NO!), or something like it.

Factors to consider during evaluation:

- **Impact** of the problem
- The **persistence** of the problem
- **Frequency** of the problem

The standard severity scale:

0 = “**I don't agree** that this is a usability problem at all”

1 = “**cosmetic problem** only: need not be fixed unless **extra time** is available on the project.”

2 = “**minor usability** problem: fixing this should be given **low priority**.”

3 = “**major usability** problem: important to fix, so should be given **high priority**.”

4 = “**usability catastrophe: imperative** to fix this before product can be released.”

Nielsen's Heuristics

- Developed by Jacob Nielsen in the early 1990s
- Based on **heuristics** distilled from an empirical analysis of 249 usability problems
- These heuristics have been **revised for current technology**
- Heuristics still needed for some emerging technologies (e.g., AR, VR).
- **Design guidelines form a basis for developing heuristics**

Nielsen's Heuristics

- **Visibility of system status**
 - Are users kept informed at all times?
- **Match between system and real world**
 - Is the UI language simple?
- **User control and freedom**
 - Are there easy escapes from unexpected locations?
- **Consistency and standards**
 - Is performing similar action consistent?
- **Error prevention**
 - Help users recognize, diagnose, recover from errors

Nielsen's Heuristics

- **Recognition rather than recall**
 - Are objects, actions and options always visible?
- **Flexibility and efficiency of use**
 - Are there accelerators?
- **Aesthetic and minimalist design**
 - Is any unnecessary and irrelevant information provided?
- **Help users recognize, diagnose, recover from errors**
 - Are error messages helpful?
- **Help and documentation**
 - Is help provided that can be easily searched?

Nielsen's Heuristics

1. Visibility of system status *discoverability, feedback*
2. Match between system and real world *mapping*
3. User control and freedom
4. Consistency and standards *consistency*
5. Error prevention
6. Recognition rather than recall *conceptual model*
7. Flexibility and efficiency of use *multiple interface/interaction type*
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, recover from errors
10. Help and documentation *learnability*

Nielsen's Heuristics

Could be **too general** and might need to be tailored to the environment

Example: HOMERUN suggested for **corporate web site** evaluation

- High-quality content
- Often updated
- Minimal download time
- Ease-of-use
- Relevant to users' needs
- Unique to the online medium
- Netcentric corporate culture

Heuristics Evaluation: Advantages

- A few guidelines identify **many common usability problems**
- Fewer practical and ethical issues to deal with
 - no participants
- **Cheap and fast:** a few guidelines identify many common usability problems
- **Provides common evaluation template** (to compare approaches, systems)

Heuristics Evaluation: Problems

- Principles may be **too general**
- **Subtleties** involved in use
- Designer may not be able to overcome being defensive OR experts may disagree
- Can be hard to find experts
- **False positives:** does the rule always apply?
- **Not complete:** will miss problems

Not a replacement for user testing

Heuristic Evaluation: Practice

Take 5-7 minutes to conduct a heuristic evaluation for the McMaster CAS website (<https://www.eng.mcmaster.ca/cas/>), on the task for **finding course requirements and information** for a software engineering student.

- Familiarize yourself with the 10 heuristics before starting the task
- Walk through the website to complete the task, and take notes on the usability issues you notice

Week 12 Overview

- **Monday**
 - ~~Introduction to Evaluation~~
 - ~~Evaluation: Performance Modeling~~
- **Wednesday**
 - ~~Evaluation: Inspection~~
 - ~~Evaluation: Heuristics~~
- **Friday**
 - **Evaluation with Users**