CS3205: Introducing Evaluation

- Readings: from ID-book:
  - Chapter 13 in published book (Ch. 12 in eText)
  - Paper: “Supporting Elementary-Age Children’s Searching and Browsing…” http://hcil2.cs.umd.edu/trs/2008-31/2008-31.pdf

Where We Are…

- We've covered:
  - Usability goals and criteria
  - How HCI could fit into process models
- And also:
  - How to identify and describe user needs, requirements
  - Started first 2 phases of Course Project
- Now an intro to evaluation activities
  - More details later (yes, we're iterating)

Our Initials Goals

- At this point in the course, we want to:
  - Get an initial sense of how evaluation is done
- **Specifically, we'll achieve this by:**
  - Explaining concepts and terms used in discussing evaluation
  - Discuss the basics of the user-observation technique
    - Other techniques later!

What to evaluate

Iterative design & evaluation is a continuous process that examines:

- Early ideas for conceptual model
- Early prototypes of the new system
- Later, more complete prototypes
- Existing systems

Designers need to check that they understand users' requirements.

  - Evaluation may teach you about your users as well as about your current design

Types of evaluation

- **Formative evaluation**
  - done at different stages of development
  - check that the product meets users' needs.
- **Summative evaluation:** assess the quality of a finished product.
  - In this course, we stress the importance of formative evaluation
- **Diagnostic evaluation:** find problems
- **Measurement evaluation:** assess performance

Techniques for Evaluation

- User observation:
  - Observe a participant interacting with system etc.
- Inspection (heuristic inspection), Walkthrough
  - Various techniques
  - No participants usually involved
- Questionnaires, Surveys
- Observation of real users in their real world
  - For understanding users, needs, and requirements
- Other methods (later)

When to evaluate

- Throughout design
- From the first descriptions, sketches etc. of users' needs through to the final product
- Design proceeds through iterative cycles of 'design-test-redesign'

- **Evaluation is a key ingredient for a successful design**
  - Why iterate if you don't learn and revise?

Reasons for doing evaluations

- Understanding the real world
  - How a technology is employed in workplace?
  - Lead to a better fit with work environment
- Does a system or prototype meet usability requirements?
  - Checking conformance to a standard
    - screen legibility, etc.
- Engineering towards a target
  - x% of novice users should be able to print correctly on first try
- Comparing designs
  - compare with competitors or among design options

Evaluation without Formal Goals

- Examples of more targeted evaluation that is not so closely tied to a specific usability goals
  - For a system, or a prototype etc.

- Why can't a user complete a task (easily)?
- Is something for novices acceptable for experts?
  - Or related issues
- Do users like a (new) design feature?

Bruce Tognazzini tells you why you need to evaluate

"Iterative design, with its repeating cycle of design and testing, is the only validated methodology in existence that will consistently produce successful results. If you don't have user-testing as an integral part of your design process you are going to throw buckets of money down the drain."

http://www.asktog.com/columns/037TestOrElse.html

See AskTog.com for topical discussion about design and evaluation.

Some Terminology

- **Usability testing, Usability evaluation** (or just an **evaluation**):
  - A systematic “testing” of an interface. See 13.3.1 in book.
- **Evaluation session:**
  - A meeting where participants participated in “testing”
- **Participant**:
  - Old term was “subject”. A person or user who provides input and feedback to those evaluating a UI
- The evaluation team, or **evaluators**:
  - **Observer**: watches and records
  - **Facilitator:** interacts with participant, explains, guides, follows-up
  - **Interviewer:** another term

Case Studies

- Look over case studies in book.
- Also, look at evaluation done in "Children's Searching and Browsing" article:
  - http://hcil2.cs.umd.edu/trs/2008-31/2008-31.pdf

How to Do User Observation Evaluations?

- Lots of practical advice and guidelines follow. Also, see this:
  "Do's and Don'ts of Usability Testing," from UX Matters
  website: http://www.uxmatters.com/mt/archives/2010/03/dos-and-donts-of-usability-testing.php
- And see the book too.

Technique Details: User Observation

- Evaluators study users who are actively using the user interface
  - The sessions are often videotaped
  - Can be done in user's environment
- What are the goals for evaluation?
  - Could measure variables (e.g. time), or could just look for problems
    - Problems ➔ "usability defects" described later
  - Perhaps you choose to focus on efficiency or learnability
    - This effects choice of participants
  - Long tasks or several short tasks

In Industry

- Microsoft webpage and video:

https://www.microsoft.com/en-us/usability/default.aspx

- Google:
  https://www.google.com/usability/

- Search YouTube for "usability lab"
  - For example, Sony Online Entertainment:
    https://www.youtube.com/watch?v=pAZglMkQsmg

What Participants Do

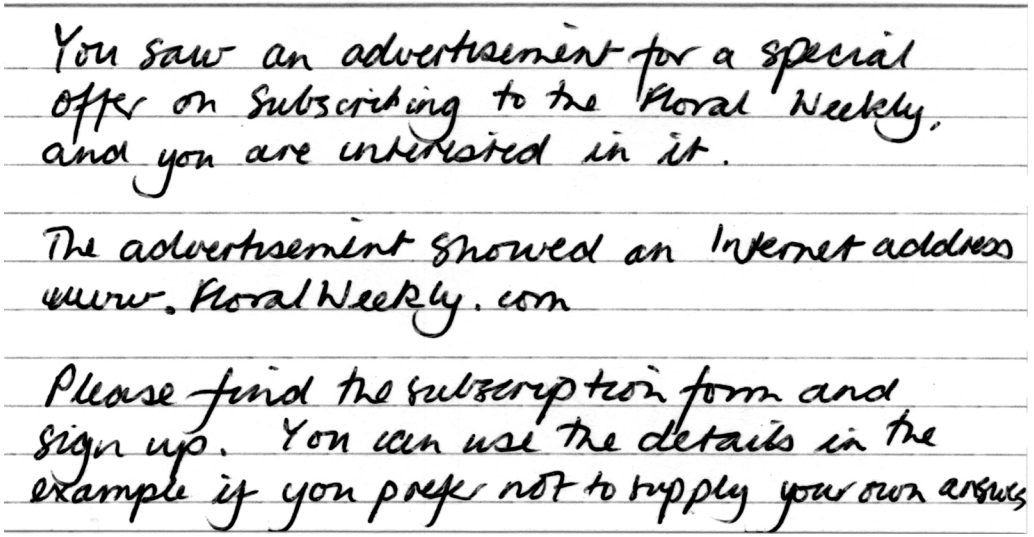
- Activities of the participant:
  - Performs pre-defined tasks
    - With or without detailed instructions on how to perform them
    - Decide to what level you'll help them
  - Preferably talks to herself as if alone in a room
    - This is called a 'think-aloud protocol'
    - This process is called co-operative evaluation when the evaluator and user talk to each other
    - http://en.wikipedia.org/wiki/Think_aloud_protocol

Tasks for Participants

- If they don't know the system, they need "warm up"
  - Plan to give an introduction, perhaps basic training
  - Give simple tasks first, perhaps with step by step guidance
- Decide what tasks matter for your evaluation. Choose among:
  - Core tasks that are done frequently
  - Tasks that are very important to users, the business
  - Tasks with new features or functionality
  - Critical tasks even if done infrequently
  - A task you think needs validation, exploration, understanding (by the design team)

Writing Task Descriptions

- Write tasks (AKA "scripts") on paper or index cards
  - State objective, final goal
  - Might sequence tasks that build on each other
  - Be prepared for user to fail. How will you continue?
  - Have extra tasks just in case…
- Example of "task cards"



Writing Task Descriptions (2)

- Why write these out?
  - Also, why plan, practice in advance, decide what info you'll collect?
- Answers:
  - You want to do this for more than one participant
  - You want consistent results
  - You want this to be as much like an experiment as it can be.

Details: Evaluation by Observation

- The importance of video:
  - Without it, 'you see what you want to see'
    - You interpret what you see based on your mental model
  - In the 'heat of the moment' you miss many things
  - Minor details (e.g. body language) captured
  - You can repeatedly analyze, looking for different problems
- Tips for using video:
  - Several cameras are useful
  - Software is available to help analyse video by dividing into segments and labeling the segments
  - Evaluation can be time consuming so plan it carefully

Steps for Evaluation by Observation

1. Select some representative users per user class (3 to 5?)
   - E.g. client, salesperson, manager, accounts receivable
2. Invite them to individual sessions
   - Sessions should last 30-90 minutes
   - Schedule 4-6 per day
3. If system involves user's clients in the interaction:
   1. Have users bring important clients
   2. or have staff pretend to be clients
4. Select facilitators/observers and note-takers
5. Prepare tasks
6. Prepare notebook or form for organizing notes

Steps for Evaluation by Observation

7. Set up and test equipment
   1. Hardware on which to run system
   2. Audio or video recorder
   3. Software logs
8. Do a dry run (pilot study)!
9. At the Start of an Observation Session
   - Facilitator explains:
     - nature of project
     - anticipated user contributions
     - why user's views are important
     - focus is on evaluating the user interface, not evaluating the user
     - all notes, logs, etc., are confidential
     - user can withdraw at any time
     - usage of devices
     - relax!
   - Participants asked to sign informed consent form:
     - very important

Steps for Evaluation by Observation

10. Start user verbalizing as they perform each task (thinking aloud)
    - For co-operative evaluation, the facilitator also verbalizes
    - Appropriate questions to be posed by the facilitator :

Question	Defect if
What do you want to do?	They do not know; the system cannot do what they want
What do you think would happen if ...?	They do not know; they give wrong answer.
What do you think the system has done?	They do not know; they give wrong answer.
What do you think is this information telling you?	They do not know; they give wrong answer.
Why did the system do that?	They do not know; they give wrong answer.
What were you expecting to happen?	They had no expectation; they were expecting something else.

11. Observers watch and take notes.

12. Post-mortem, debriefing
    - Express appreciation. Answer any questions the participant might have.
    - Ask: What was most difficult to learn?
    - Ask: What were the most significant problems?
    - Any general thoughts the participant has about the evaluation or the software
    - You ask for clarifications of things you thought you observed
    - Etc.
13. Analysis of results by evaluation team

In-Class Exercise

- Form group of 2 or 3
  - Choose a complex task for one of your cell phones E.g. add a new contact, edit a contact,…
  - Hopefully one that might be easy to do wrong
  - Write out a task description
- Partner with group next to you
  - Ask one of them to carry out the task
  - One person “moderates” and you practice the “talk aloud” method
  - Others make notes of what they observe
- Total time: 15 minutes

Usability Defects

- You might observe usability defects during an evaluation
- An important idea:

  "There are no good user interfaces ... just user interfaces that fit"
  - A truly bad user interface never fits
  - But among the 'good' ones, some will suit one task/user; some with suit another

- To maximize fitness, we must minimize the occurrence of usability defects in the context of the expected use of the system

What Is a Usability Defect?

- Note: There is not a strict definition for this. But we'll use it to mean the following:
- A usability defect could be:
  - A mismatch between what the user wants, needs or expects and what the system provides
  - A breakdown in usability
  - An obstacle to performing a desired task
  - A problem in the smooth functioning of the user/computer system

Caveats

- You should know that:
  - Occasional breakdowns are normal
  - Systematic or frequent defects need to be fixed
- Users may not be aware of many defects
  - the defects may only be located through careful analysis
  - they may be subtle

Analyzing Usability Defects

- You may choose (or be asked) to do a more detailed analysis of individual usability defects that you find
  - This may lead to a more complete understanding of a system-wide issues, or user characteristics

- Things you could document
  1. In what part of the (large) system did this occur? What subsystem or component?
  2. A what stage did this occur as the user tried to carry out a task?
  3. At which “level” in the system is there a problem?
  4. How was the problem revealed?
  5. What can be said about the cause?

  (Details follow)

“When” Did Things Go Wrong?

- Consider the user’s goal/decide/execute cycle
  - The user now wants to achieve some goal.
    - Perhaps she doesn’t know what to do now.
    - Perhaps she chooses the wrong goal.
  - She decides what to do with the system to achieve. She decides upon a particular action.
    - Perhaps she can’t figure out how to do what she wants.
    - Perhaps she choose the wrong action.
  - She carries out the action and the system does something in response.
    - Perhaps she cannot do the action in this interface.
    - Perhaps failure in system functionality (a bug).
  - She interprets the results and responses
    - Perhaps she can’t interpret it or misinterprets it

How Was the Defect Revealed?

- Detected by the system
  - Easy to see! System catches invalid use.
- Detected by the user
  - I can't find the command. I don't know what to do next. This message makes no sense. Etc.
- Undetected Defects
  - The user doesn't see the problem, at least not right away
  - May do unnecessary work later
- Inefficiencies
  - Complex, long operations or steps.
  - Too much time needed to think, choose, or execute

Defects Occur at Different Levels

- Task and Goal level
  - E.g. system doesn't provide needed functionality
- Conceptual level
  - E.g. user has wrong mental model or doesn't understand system's mental model
  - Misinterprets system actions, features
- Interaction Style level
  - System-wide issues with how the general mode of interaction is defined
- Interaction Element level
  - Some specific component in the UI has a problem
- Physical Element level
  - E.g problem typing, clicking, etc.

What is the Cause?

- You know your usability principles etc.
- Consider:
  - Defect in functionality (AKA a bug)
  - Poor feedback, labeling, or message
  - Failure to indicate important distinctions (e.g. color)
  - Distraction or failure to focus attention when needed
  - Feature interaction
  - Failure to remember (recall vs. recognition)
  - Lack of knowledge to achieve goal, do task
  - Misleading information, help, or training
  - Overly complex or confusing screen design
  - Overly complex or confusing task support
  - Physical difficulties (key-combinations, mouse movements, button use)

Wrap-up on Defect Analysis

- You could say a lot about a usability defect
- Consider why you're evaluating and what level of reporting you need
  - Perhaps save detailed analysis for most serious problems
- When will you do this analysis?
  - Not while observing users, probably!
  - Afterwards, reviewing notes
  - Afterwards, reviewing video
  - Note: post-session evaluation of video may take 3-5 times as much time as the session itself