Evaluating Interfaces with Users

Why evaluation is crucial to interface design

General approaches and tradeoffs in evaluation

The role of ethics

Why Bother?

Tied to the usability engineering lifecycle

• Pre-design
  – investing in new expensive systems requires proof of viability

• Initial design stages
  – develop and evaluate initial design ideas with the user

• Iterative design
  – does system behaviour match the user’s task requirements?
  – are there specific problems with the design?
  – can users provide feedback to modify design?

• Acceptance testing
  – verify that human/computer system meets expected performance criteria
  • ease of learning, usability, user’s attitude, performance criteria
  • e.g., a first time user will take 1-3 minutes to learn how to withdraw $50 from the automatic teller

Recall the iterative approach:
Can you evaluate the following navigation bar…

Can you evaluate the following map…
**What Defines Success?**

We want a “usable” system. What are some metrics that can be used to measure whether a system or feature within one is usable?

– Time to learn
– Speed of performance
– Rate of errors by users
– Retention over time
– Subjective Satisfaction

Often, there will be tradeoffs between these goals.

---

**Approaches: Naturalistic/Qualitative**

**Naturalistic:**

• describes an ongoing process as it evolves over time
• observation occurs in realistic setting
  – ecologically valid
• “real life”

**External validity**

• degree to which research results applies to real situations
Approaches: Experimental/Quantitative

Experimental
• study relations by manipulating one or more independent variables
  – experimenter controls all environmental factors
• observe effect on one or more dependent variables

Internal validity
• confidence that we have in our explanation of experimental results

Trade-off: Natural vs Experimental
  precision and direct control over experimental design
  versus
  desire for maximum generalizability in real life situations

Validity Concerns
Does the test measure something of relevance to usability of real products in real use outside of lab?
Some typical validity problems of testing vs real use:
  – non-typical users tested
  – tasks are not typical tasks
  – physical environment different
    quiet lab -vs- very noisy open offices vs interruptions
  – social influences different
    motivation towards experimenter vs motivation towards boss

A partial solution involves using real users, using representative tasks from task-centered system design, and testing in an environment similar to real situation…
**Introspection Method: Designer**

Typically used with interface design. A design team member tries the system (or prototype) out (doing a walkthrough of the systems screens and features).

- They are looking to determine whether the system “feels right” when being used.
- Is probably still the most common evaluation method…

Potential problems are reliability issues since:
- it is completely subjective
- the “introspector” is a non-typical user
- being so close to the project your intuitions and introspection are often biased and thus wrong…

---

**Introspection Method: User**

Typically done as a user-centered walkthrough of a system. The idea here is typically one of conceptual model extraction by showing representative users prototypes or even screenshots of a mock-up.

- Can ask the user to explain what each screen element does or represents as well as how they would attempt to perform individual tasks.

This can allow us to gain insight as to a user’s initial perception of our interface and the mental model they might be constructing as they begin to use a system.

NOTE: Since we’re walking them through specific parts as their guide, we won’t really see how a user might explore the system on their own or their learning processes.
**Direct Observation**  
The evaluator(s) observe and record users interacting with a design/system, either in a lab setting or “field” setting.  
- When done in a lab the user is typically asked to complete a set of pre-determined tasks and it might be done in a special instrumented usability lab to facilitate recording.  
- When done “in the field” the user might be asked to go through their normal routine, or if they are asked to complete a set of tasks, they are at least doing it in natural setting.  

While this can be excellent at identifying gross design/interface problems, the validity and reliability depends on how controlled and/or contrived the situation is...

---

**Direct Observation Approaches**  
Typically utilized in software design, and there are three general approaches that can be used for direct observations:  
- simple observation  
- think-aloud  
- constructive interaction
**Direct observation: Simple Observation Method**

The user is given the task(s) to perform and the evaluator(s) simply watch (and possibly record) what the user does.

**Potential problem**

– it is quite possible this does not provide any insight into the user’s decision process or their attitude/feelings while performing the tasks

**Direct observation: Think Aloud Method**

A similar setup to simple observation, but the users are asked to say what they are thinking/doing during the tasks.

– what they believe is happening
– what they are trying to do
– why they took an action

This can give insights into what the user is thinking, but there are potential problems

– can be awkward/uncomfortable for subject (thinking aloud is not natural when working alone)
– “thinking” about why they are doing things could alter the way people perform their task
– hard to talk when they are concentrating on problem

Generally seems to be the most widely used evaluation method in industry
**Direct observation: Constructive Interaction Method**

Similar to the other two, but here two people work together on the task(s).

- This can lead to a normal conversation between the two users which can then be monitored.
- It should remove the awkwardness of think-aloud but might be less realistic depending on the tasks.

![Image of two people working together]

Now, why did it do that?

Oh, I think you clicked on the wrong icon

---

**Co-Discovery**

A *variant* of constructive interaction is to have co-discovery learning take place, where the pair working together are:

- a semi-knowledgeable “coach”
- a beginner (who is actually using the system)

**Ideally, this results in**
- the “naïve” beginner participant asking questions
- the semi-knowledgeable “coach” responding
- insights into thinking process of both beginner and intermediate users
What might the future hold?

We live in a time where the use of AI is on the rise and chat bots are in the thick of things. We went from the Universal McCann agency’s Jill020306 to Microsoft’s Tay in the span of a decade. Where could they take us in the future? What are the ethical issues that could come out of this?

When you write a new library or program module, you can use unit testing tools to automatically assess the accuracy of various things. The tools continue to expand, even into the ability to automatically test GUI elements. Where might they go in the future?

Possible direction: “Chatbots: Your Ultimate Prototyping Tool”

https://medium.com/ideo-stories/chatbots-ultimate-prototyping-tool-e4c2831967f3#.74c9jy2n