Evaluation*

Evaluation is the fourth main process of the interactive system design.

Evaluating means reviewing, trying out or testing a desing idea, a piece of software, a product or a service to discover whether it meets some criteria.

These criteria could be summed up by the guidelines for good design, but at other times the designer might be more interested in some other characteristic of the design, such as whether the system is fit for its purpose, enjoyable, engaging and so on.

Evaluation

Even though we say that evaluation is the fourth main process of the interactive system design, evaluation occurs throughout the interaction design process.

Evaluation is closely tied to other key activities of interactive systems design, understanding, design and envisionment.

Evaluation is also critically dependent on the form of envisionment used to represent the system: you will only be able to evaluate system that are represented in a form appropriate for the type of evaluation

Evaluation

There are two main types of evaluation:

1) Expert-based methods: involve a usability expert, or an interaction designer, reviewing some form of envisioned version of a design

2) Participants methods: Involve recruiting people to use an envisioned version of a system.

In both cases the choice of who to involve is crucial, because problems could raise due to people involved in the evaluation.

Evaluation

In expert based methods people could be chosen or not. If the last is the case, is better to used some mediation techniques, such as some additional materials to help the process.

Evaluation

In participants based evaluation the choiche of people (i.e. who to involve) is more critical: when possible, these people should be representative of the people to whom the system is aimed (“end-users”), and this is the reason why this method is the preferred where the designer has access to the target population.

Alternatively, the participants may be other people (perhaps other designers, students or whoever happen to be around) who are invited to play the part of the people who will use the system. The characteristics of the target population can be captured through personas.

Formative evaluation: what to evaluate?

When evaluation is at an early stage, it is called “formative”, because the result help to form − or shape − the design.

You may need to evaluate initial concepts, especially if the application is novel, or to decide between different design options.

In this case we could evaluate:

1) Quick papers prototypes or even software if this can be produced rapidly (sometimes a Power Point simulation is good enough);
2) Competitors products

Formative evaluation: what to evaluate?

3. Previous versions of technology

4. A prototype with (at least) the features that should be tested. For technological products we distingiush between horizontal prototype (when the whole system is not fully operational) and vertical prototype (when the whole system completely functional, but only in some parts)

Accessing the usability of a finished product (summative evaluation)

This may be:

1) To test against in-house guidelines or
2) Formal usability standard, such as ISO 9241;
3) Provide evidence of usability (often required by a customer), for example the time to complete a particualr set of operations.

Government departments and other public bodies often requires suppliers to conform to accessibility standard and health and safety registration.

Expert evaluations

A simple, relatively quick and effective method of evalution is to get an interaction design, or usability, expert, to look at the system and try using it.

This is not substitute for getting real people to use your design. In the participatory design approach stakeholders help designers set the goals for the evaluation work.

Involving stakeholders has great benefits in terms of eventual uptake and use of the technology.

Expert evaluations

In other words, it is crucial to understand WHO are the stakeholders that we want (and could) involve.

Involving stakeholders is particularly crucial for technology that is tailor-made for defined communities.

For example, in a redevelopment of a Web-based gallery showcasing the work of disadvantaged artistis, as many of the artists themselvels as possible were involved in evaluating the new design, thus preserving the sense of community which the site aimed to embody.

Expert evaluations

Expert evaluation is effective, particularly early in the design process.

First evaluations (which sometimes are the only one...) are usually done by a single expert (or in a very little team).

If not the only one, an expert evaluation is however effective, because experts will pick up common problems based on this experience and will identify factors that might otherwise interfere with an evaluation by non-experts.

Expert evaluations

To help the experts structure their evaluation, it is useful to adopt a particular approach. This will help to focus the expert's critique on the most relevant aspects for the purpose.

For this reason Heuristic Evaluation and Cognitive Walkthrough are extremely useful (and for these techniques there are also several "discounted" versions).

Expert evaluation: Heuristic Evaluation

Heuristic evaluation refers to a number of methods in which a person(s) trained in HCI and interaction design examined a proposed design to see how it measures up against a list of principles, guidelines or “heuristics” for good design.

It could resolves in a quick discussion or in a more formal process carefully documented: ideally, several people with expertise in interactive system should review the interface. Each expert notes the problems and the relevant heuristic, and suggests a solution where possible (re-design).

It is also helpful if a severity rating, say on a scale of 1-3, is added, according to the likely impact of the problem as recommended by Dumas and Fox (2007), who however also note the disappointing level of correlation amongst experts in rating severity of problems.

Heuristic Evaluation

Basically, heuristic are principles for good design. There are many sets of heuristics to choose from, both general purpose and those relating to particular application domains, for example heuristics for WEB design. Here a list of the most important/ common:

1) Visibility
2) Consistency
3) Familiarity
4) Clarity
5) Navigation
6) Control
7) Feedback
8) Recovery
9) Constraints
10) Flexibility
11) Style
12) Conviviality

These 12 principles are often referred to as “the Nielsen heuristics”, based on “Usability engineering” by Nielsen (1993). Be careful, because Nielsen suggested only 10 heuristics, and on the web you can fined 8, 10, 12...

The principles reported here are elaborated by Benyon on the base of Norman, Nielsen and others.

All the principles interact in a complex way, affecting each other, sometimes conflicting with each other and sometimes enhancing each other. But they help to orientate the designer to key features of good design and sensitize the designer to important issues.

For easy of memory and use, design principles are grouped into three main categories, but this groupings are not rigid.

.

- Principles 1-4 are concerned with access, ease of learning and remembering (learnability).
- Principles 5-7 are concerned with ease of use, and principles 8 and 9 with safety (effectiveness).
- Principles 10-12 are concerned with accomodating differences between people and respecting those differences (accomodation).

Heuristic Evaluation

Usually evaluators works independently from one each other, to join thei results afterwards. They could need to test the system and to be informed on the functions of the system itself.

In this case users' scenarios and use case are extremely useful.

Learnability

1. Visibility. Try to ensure that things are visible so that people can see what functions are available and what the system is currently doing. This is an important part of the psychological principle that it is easier to recognize things than to have to recall them. If it is not possible to make it visible, make it observable. Consider make the things "visible" through the use of sounds and touch.

Learnability

2. Consistency. Be consistent in the use of design features and be consistent with similar system and standard ways of working. Both conceptual and physical consistency are important.

3. Familiarity. Use language and symbols that the intended audience will be familiar with. Where this is not possible because the concepts are quite different from those people know about, provide a suitable metaphor to help them transfer similar and related knowledge from a more familiar domain.

Learnability

4. Clarity. Design things so it is clear what they are for; for example,make buttons look like buttons so people will press them. It refers to the properties that things have (or are perceived to have)and how this relate to how the things could be used. Buttons afford pressing, chair afford sitting on, and post-it notes afford writing a message on and sticking next to something else...In other words, remember what we sayd about affordance (started as purely perceptual, now cultural, interacting with subject/with object/ with both...nested, sequential and so on)

Giving to people the sense of being in control, knowing what to do and how to do it (ease of use)

5. Navigation. Provide support to enable people to move around the parts of the system: map, directional signs and information signs.

Ease of Use

6. Control. Make it clear who or what is in control and allow people to take control. Control is enhanced if there is a clear, logical mapping between controls and the effect that they have.

   Also make clear the relationship between what the system does and what will happen in the world outside the system.

7. Feedback. Rapidly feed back information from the system to people so that they know what effect their action have had. Constant and consistent feedback will enhance the feeling of control

Safely and securely (effectiveness)

8. Recovery. Enable recovery from actions, particualrly mistakes and errors, quickly and effectively.

9. Constraints. Provide constraints so that people do not try to do things that are inappropriate. In particular, people should be prevented from making serious errors through properly constraining allowable actions and seeking confirmation of dangerous operations.

In a way that suits them (accomodation)

10. Flexibility. Allow multiple ways of doing things, so as to accomodate people with different levels of experience and interest in the system. Provide people with the opportunity to change the way things look or behave so that they can personalize the system.

Accomodation

11. Style. Design should be stylish and attractive.

12. Conviviality. Interactive system should be polite ,friendly and generally pleasant. Design for politeness.

What does it mean?

Expert Evaluation: Discount Usability Engineering

This approach to evaluation was pionereed by Jacob Nielsen (1993) and enthusiastically followed by many time-pressured evaluation practitioners. It is now used for any “quick and dirty” approach to evaluation where the aim is to get useful, informed feedback as soon as possible.

Expert Evaluation: Discount Usability Engineering

Once again a number of usability experts “walk through” concrete scenarios, preferably accompanied by personas, and inspect the design for difficulties.

Basically they based on a summarized version of the list of design principles reported for the “heuristic evaluation”, which is summarized by the three overarching usability principles of learnability (1-4), effectiveness (5-9) and accomodation (10-12)

Discount Usability Engineering: a quick exercise

Carry out a quick review of the control for a domestic device (e.g. washing machine, microwave) for learnability, effectiveness and accomodation.

Discount Usability Engineering

Unless ther is no alternative, you should not evaluate your own design. It is extremely difficult: it is extremely difficult to ignore your knowledge of how the system worrks, the meaning of icons or menu names and so on, and you arelikely to give the design the “benefit of the doubt” or to find obscure flaws which few users will ever happen upon.

Discount Usability Engineering: better not use it?

Woolrich and Cokton (2000) conducted a large-scale trial of heuristic evaluations. Evaluators were trained in use the tecnique, then evaluated the interface to a drawing editor. The editor was then trialled by customers.

Comparison of findings showed that many of the issues identified by the experts were not experienced by people (false positive), while some severe difficulties were missed by the inspection against heuristics.

Why?

Many false positive stemmed from a tendency by the experts to assume that people had no intelligence of even common sense.

As for “missing” problem, this tended to result from a series of mistakes and misconceptions, often related to set of linked items, rather than isolated misunderstandings.

Woolrich and Cokton conclude that the heuristic add little advantage to an expert evaluation and the result of applying them may be counter-productive.

At the very least, the technique must be used together with careful consideration of people and their real-life skills and participant evaluation is required to get a realistic picture of the success of a system.

Discount Usability Engineering: better not use it?

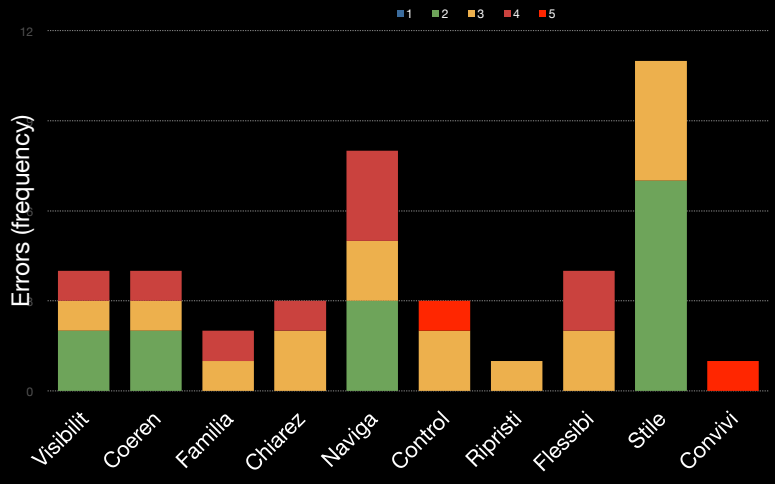
Discount Heuristic Evaluation therefore is valuable as a formative evaluation. to help the designer improve the interaction at an early stage.

It should not be used as a summative assesment, to make claims about the usability and other characteristics of a finished product. If that is what we need to do, then we must carry out properly designed and controlled experiments with a much greater number of particpants.

However, the more controlled the tewsting situation becomes, the less it is likely to resemble the real world, which lead us to the question of ECOLOGICAL VALIDITY

Expert Heuristic Evaluation

33 violations have been identified.



Cognitive walkthrough

Cognitive walkthrough is a rigorous paper-based technique for checking through the detailed design and logic of steps in an interaction.

It is derived from the human interaction processor view of cognition and closely related to task analysis.

In essence, the cognitive walkthrough entails a usability analyst stepping stepping through the cognitive task that must be carried out in interacting with technology.

Originally developed by Lewis et al., (1990) for application where people browse and explore information, it has been extended to interactive systems in general.

Cognitive walkthrough

Aside from its systematic approach, the great strength of CW is that it is based on well established theory rather than the trial and error or a heuristical based approach

Inputs to the process are:

1) An understanding of the people who are expected to use the system;
2) A set of concrete scenarios representing both (a) very common and (b)uncommon but critical sequences of activities
3) A complete description of the interface to the system, including also the correct sequences of actions for achieving the scenario tasks, usually as a hierarchical task analysis (HTA).

Cognitive walkthrough

Having gatherd these materials together, the analyst asks the following four questions for each individual step in the interaction:

1) Will the people using the systeem try to achieve the right effect?
2) Will they notice that the correct action is available?
3) Will they associate the correct action with the effect that they are trying to achieve?
4) If the correct action is performed, will people see the progress is being made toward the goal of their activity?

Cognitive walkthrough

If any of the questions is answered in the negative, then a usability problem has been identified and is recorded, but redesign suggestions are not made at this point.

CW should be carried out as a group exercise by analysts and designers together. The analysts stepthrough usage scenarios and the design team are required to explain how the user would identify, carry out and monitorthe correct sequence of actions.

Participant based evaluation

While expert-base evaluation is a reasonable first step, it will not find all problems, particularly those that result from a chain of “wrong” actions or are linked to fundamental misconceptions.

So it’s really important to complete the picture with some real people trying out the interaction design.

Furthermore, from a pollitical point of view, it is easier to convince designers of the need for changes if the evidence is not simply one “expert” view.

Participatory Heuristic Evaluation

The procedure for the use of Participatory Heuristic Evaluation is just as for the expert version, but the participant are involved and must be briefed about what is required (providing the list of heuristic, sometimes assigning small tasks).

It could be useful to run both the evaluation and to compare the results with participants.

Cooperative Evaluation

The technique is cooperative because participants are not passive subjects but work as “co-evaluator”.

It was devised by Monk et al. (1993), who gives also a detailed table of ToBeDone.

Cooperative evaluation (Monk et al., 1993)

Step:	Notes:
1. Using the scenarios prepared earlier, write a draft list of tasks	Tasks must be realistic, do-able with the software, and explore the system thoroughly.
2. Try out the task and estimate how long they will take the participant to complete	Allow 50% longer than the total task time for each session
3. Prepare a task sheet for the participant	Be specific and explain the tasks so that anyone can understand

Step:	Notes:
4. Get ready for the test session	Have the prototype ready is a suitable environment with a list of prompt questions, notebooks and pens ready. A video or audio recorder would be very useful here.
5. Tell the participants that it is the systeem that is under test, not them; explain and introduce the tasks.	Participants should work individually.

Step:	Notes:
6. Participants start the task: Have them give you running commentary on what they are doing, why they are doing it and difficulties or uncertainties they encounter.	Take notes of where participants find problems or do something unexpected, and their comments.
7. Encourage participants to keep talking	See list of questions
8. When participants have finished, interview them briefly about the usability of the prototype and the session itself.	See list of questions

Step:	Notes:
9. Write your impression as soon as possible and insert a report on usability	

Sample questions *during* the evaluation:

- What do you want to do?
- What were you expecting to happen?
- What is the systm telling you?
- Why has the system done that?
- What are you doing now?

Sample questions *after* the session:

- What was the best/worst thing about the prototype(system/product)?
- What most needs changing?
- How easy were the tasks?
- How realistic were the tasks?
- Did giving a commentary distract you?

Co-discovery

Co-discovery is a naturalistic, informal technique that is particularly good for capturing first impressions.

The standard approach is varied by having participants to explore new technology (or systems, products) in pair (better if friends or a couple), and to ask them to "think aloud".

Controlled experiment

Is always participant based evaluation...but remember

ECOLOGICAL VALIDITY

Usability: Measures and unity (of measure)

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Overall Usability	Percentage of tasks successfully completed. Percentage of users successfully completing tasks	Time to complete a task. Time spent on non-productive actions.	Rating scale for satisfaction. Frequency of use if this is voluntary (after system is implemented)

Iso 9241-11:1998

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Meets needs of trained or experienced users	Percentage of advanced tasks completed. Percentage of relevant functions used.	Time taken to complete tasks relative to minimum realistic time.	Rating scale for satisfaction with advanced features
Meets needs for walk-up and use	Percentage of tasks completed successfully at a first attempt.	Time taken on first attempt to complete task. Time spent on help functions	Rate of voluntary use (after system is implemented.

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Meets needs for infrequent or intermit-tent use	Percentage of tasks completed successfully after a specific period of non-use.	Time spent re-learning functions. Number of persistent errors	Frequency of reuse (after system is implemented)
Learnability	Number of functions learned. Percentage of users who manage to learn to a pre-specified criterion.	Time spent on help functions Time to learn to criterion	Rating scale for ease of learning.

Good ways to have good metrics: usability methods

- Observation, interviews and questionnaires
- Personas e scenarios
- Thinking aloud, probes, card sorting
- Task analysis
- Heuristic evaluation
- Cognitive walkthrough
- Participants based evaluation