

## Chapter 3

# Usability Engineering

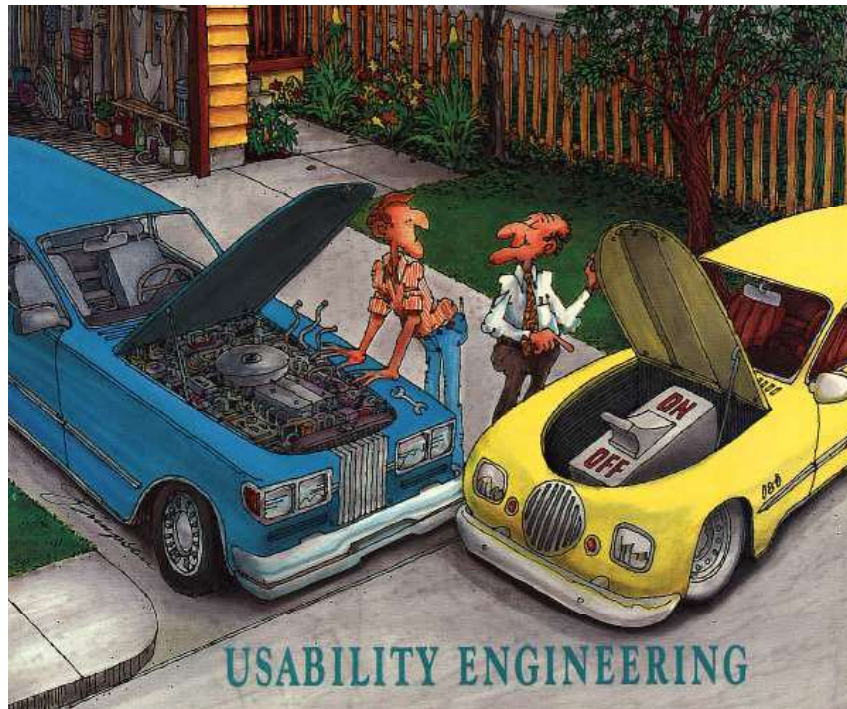
*“ When the cook tastes the soup it is formative evaluation  
and when the guest tastes the soup it is summative. ”*

[ Robert E. Stake, 1976. [Stake 1976, page 19] ]

Usability Engineering . . . *iterative process* to improve usability of a system.

### References

- ++ Alan Cooper, et al; *About Face: The Essentials of Interaction Design*; 4<sup>th</sup> Edition, Wiley, 02 Sept 2014. ISBN 1118766571 (com, uk) [Cooper et al. 2014]
- ++ Lazar, Feng, and Hochheiser; *Research Methods in Human-Computer Interaction*; 2<sup>nd</sup> Edition, Morgan Kaufmann, 2017. ISBN 0128053909 (com, uk) [Lazar et al. 2017]
- ++ Rex Hartson and Pardha Pyla; *The UX Book*; 2<sup>nd</sup> Edition, Morgan Kaufmann, 14 Dec 2018. ISBN 0128053429 (com, uk) [Hartson and Pyla 2018]
- + Chauncey Wilson (Ed.); *User Experience Re-Mastered*; Morgan Kaufmann, 2009. ISBN 0123751144 (com, uk) [C. Wilson 2009]
- + Stone, Jarrett, Woodruffe, and Minocha; *User Interface Design and Evaluation*; Morgan Kaufmann, March 2005. ISBN 0120884364 (com, uk) [Stone et al. 2005]
- Jakob Nielsen; *Usability Engineering*; Morgan Kaufmann, 1994. ISBN 0125184069 (com, uk) [Nielsen 1993b]
- Deborah Mayhew; *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design*; Morgan Kaufmann, 1999. ISBN 1558605614 (com, uk) [D. J. Mayhew 1999]
- Rosson and Carroll; *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*; Morgan Kaufmann, 2001. ISBN 1558607129 (com, uk) [Rosson and Carroll 2001]
- John Carroll; *HCI Models, Theories, and Frameworks*; Morgan Kaufmann, 2003. ISBN 1558608087 (com, uk) [Carroll 2003]
- ISO 9241-11 *Ergonomics requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability*, 1998. [ISO 1998]
- ISO 13407 *Human-centred design processes for interactive systems*, 1999. [ISO 1999]



**Figure 3.1:** Usability engineering. By Jay Simpson, from the cover of IEEE Computer, March 1992.

## Online Resources

- + Susan Farrell; *UX Research Cheat Sheet*; NN/g, 12 Feb 2017. <https://nngroup.com/articles/ux-research-cheat-sheet/>
- Nick Babich; *Most Common UX Design Methods and Techniques*; UX Planet, 13 Jul 2017. <https://uxplanet.org/most-common-ux-design-methods-and-techniques-c9a9fdc25a1e>
- UX Collective; *UX Design Methods & Deliverables*; 01 May 2016. <https://uxdesign.cc/ux-design-methods-deliverables-657f54ce3c7d>
- UXPA; *Usability Body of Knowledge*. [usabilitybok.org](http://usabilitybok.org) Last updated 2012.

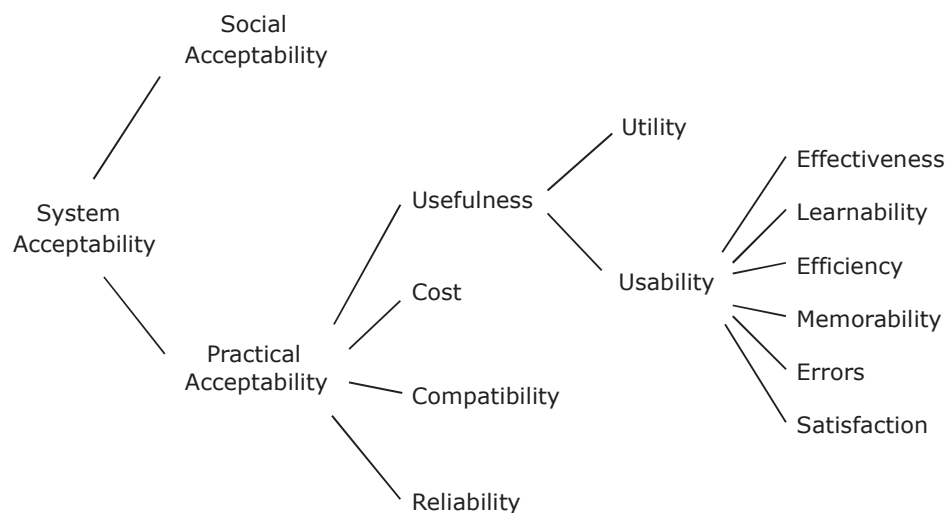
## 3.1 Defining Usability

The ISO defines usability as “the extent to which a product can be used by *specified users* to achieve *specified goals* with *effectiveness*, *efficiency* and *satisfaction* in a *specified context of use*.” [ISO 1998].

[The emphasis is mine, not part of the original definition.]

The three measurable usability attributes defined by ISO [1998] are:

- *Effectiveness*: accuracy and completeness with which users achieve specified goals.
- *Efficiency*: resources expended in relation to the accuracy and completeness with which users achieve goals.
- *Satisfaction*: freedom from discomfort, and positive attitudes towards the use of the product.



**Figure 3.2:** A model of the attributes of system acceptability, based on Figure 1 of Nielsen [1993b, page 25].

### Usability in Context

Nielsen [1993b, pages 24–25] defines usability in the context of overall system acceptability, as shown in Figure 3.2.

### Six Usability Attributes

Combining the three ISO usability attributes with Nielsen’s five usability attributes, leads to the following six usability attributes:

1. *Effectiveness*: completeness with which users achieve their goal.
2. *Learnability*: ease of learning for *novice* users.
3. *Efficiency*: steady-state performance of *expert* users.
4. *Memorability*: ease of using system intermittently for *casual* users.
5. *Errors*: error rate for minor and catastrophic errors.
6. *Satisfaction*: how satisfying a system is to use, from user’s point of view.

### Measuring Usability Attributes

- *Effectiveness*: decide on definition of success. For example, number of substitution words spotted in a text, or binary measure of success (order completed or not).
- *Learnability*: pick novice users of system, measure time to perform certain tasks. Distinguish between no/some general computer experience.
- *Efficiency*: decide definition of expertise, get sample expert users (difficult), measure time to perform typical tasks.
- *Memorability*: get sample casual users (away from system for certain time), measure time to perform typical tasks.

- *Errors*: count minor and catastrophic errors made by users while performing some specified task. For example, number of deviations from optimal click path.
- *Satisfaction*: ask users' subjective opinion (questionnaire), *after* trying system for real task.

## 3.2 Usability Evaluation

There are four types of evaluation, according to the purpose of the evaluation:

- *Exploratory* - how is it (or will it be) used?
- *Predictive* - estimating how good it will be.
- *Formative* - how can it be made better?
- *Summative* - how good is it?

My definitions [Andrews 2008] are adapted from those of several authors [Stone et al. 2005; Rubin 1994; Lockee et al. 2002; Ellis and Dix 2006].

### Exploratory Evaluation

Explores current usage and the potential design space for new designs:

- Done *before* interface development.
- Learn *which* software is used, *how often*, and *what for*.
- Collect *usage data* – statistical summaries and observations of usage.

### Predictive Evaluation

Estimates the overall quality of an interface:

- Done once a design has been done, but before implementation proceeds.
- Like a summative evaluation, but a prediction made in advance.

### Formative Evaluation

Informs the design process and helps improve an interface during design;

- Done *during* interface development.
- Learn *why* something went wrong, not just that it went wrong.
- Collect *process data* – qualitative *observations* of what happened and why.

Formative evaluation methods are “find and fix” methods, and typically produce a list of problems found as output.

### Summative Evaluation

Assesses the overall quality of an interface:

- Done once an interface is (more or less) *finished*.
- Either compare alternative designs, or test specific performance requirements.
- Collect *bottom-line data* – quantitative *measurements* of performance: how long did users take, were they successful, how many errors did they make.

### Modified Soup Analogy

Extending Robert Stake's soup analogy [Stake 1976, page 19] to usability evaluation gives:

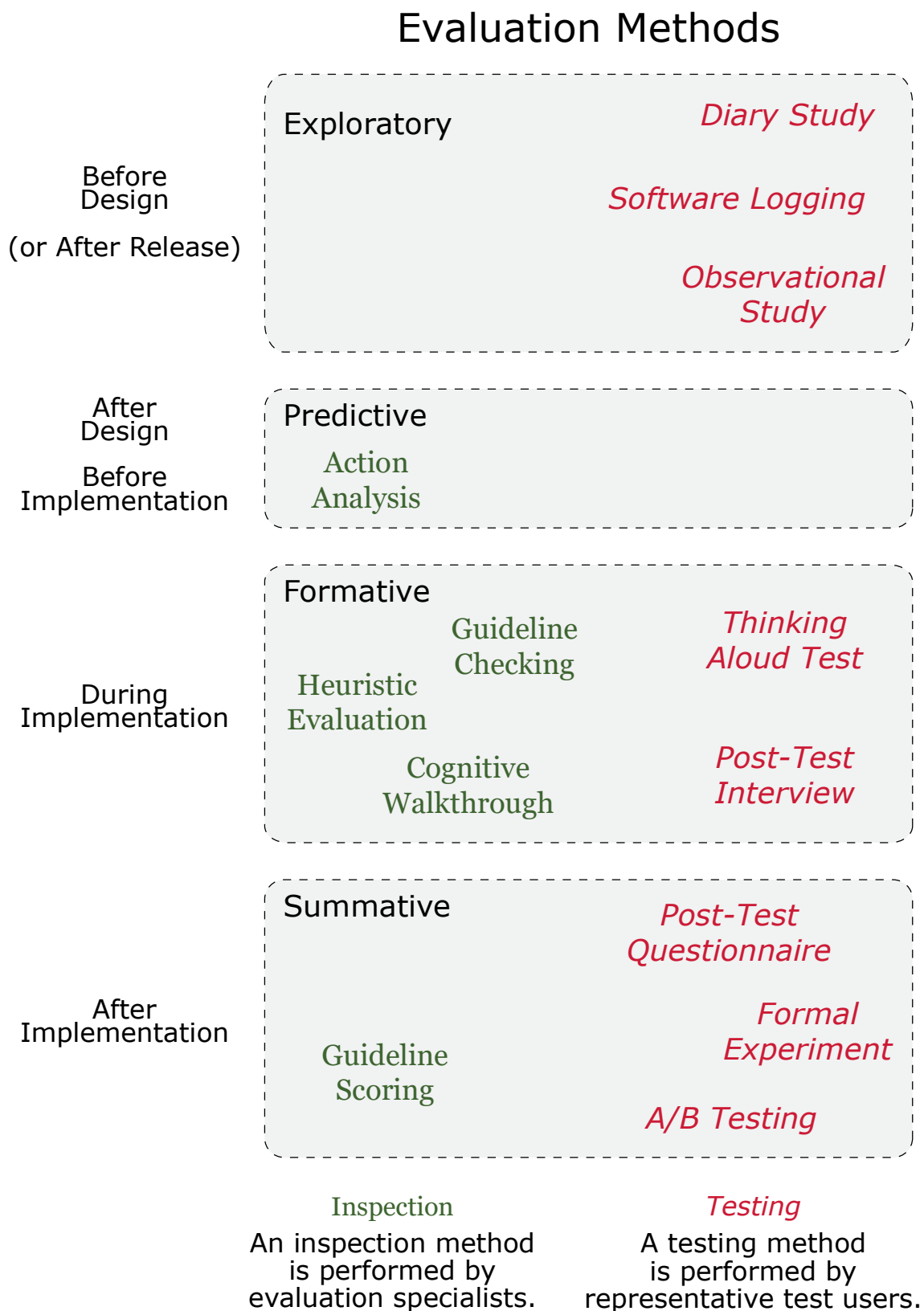
“When the cook tastes other cooks' soups, that's *exploratory*.  
When the cook assesses a certain recipe, that's *predictive*.  
When the cook tastes the soup while making it, that's *formative*.  
When the guests (or food critics) taste the soup, that's *summative*.”

### Usability Evaluation Methods

The methods of usability evaluation can also be classified according to who performs them:

- *Usability Inspection Methods*  
Inspection of interface design by usability specialists using *heuristics* and *judgement* (no test users).
- *Usability Testing Methods*  
*Empirical* testing of interface design with real users.

Figure 3.3 illustrates some of the different inspection and testing methods, grouped by purpose and by who performs them.



**Figure 3.3:** Twelve common evaluation methods grouped by purpose and by who performs them.

### 3.3 The Usability Engineering Lifecycle

1. Exploratory Evaluation
2. User Research
3. Usability Benchmarking
4. Interaction Design
5. Iterative Design Cycle:
  - (a) Prototyping
  - (b) Formative Evaluation (Inspection and/or Testing)
6. Summative Evaluation
7. Follow-up Studies

The usability engineering lifecycle above is based on the discussion in Nielsen [1993b, Chapter 4]. It is illustrated in Figure 3.4.

#### 3.3.1 User Research

Get to know the target user group:

- Qualitative research: observation of users and interviews.
- Classify users according to their characteristics.
- Draw up a *user profile* for each (potential) class of user, based on behavioural and demographic variables.
- Identify user goals and attitudes.
- Analyse workflow and context of work.
- Exploratory evaluation: which software is used, how is it used, and what is it used for.
- Draw up a set of typical *user scenarios*.

See Chapter 4.

#### 3.3.2 Usability Benchmarking

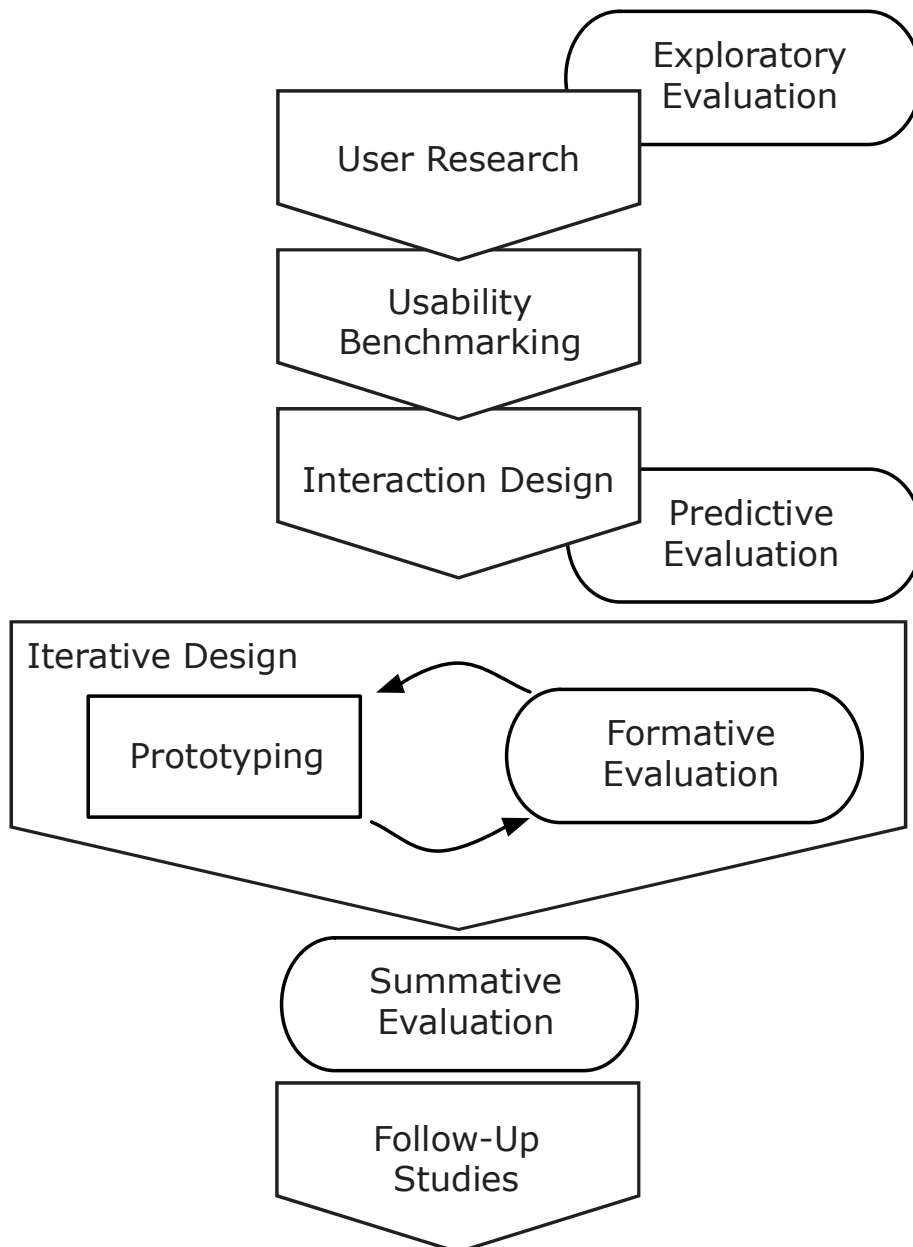
- Analyse competing products or interfaces heuristically and empirically.
- Set measurable usability targets for your own interface.

See Chapter 5.

#### 3.3.3 Interaction Design

Goal-oriented initial design of interface.

See Chapter 6.



**Figure 3.4:** The usability engineering lifecycle. Adapted from a figure kindly provided by Martin Loitzl.



### 3.3.4 Iterative Design

“Design, Test, Redesign.”

Build and evaluate prototype interface, then:

- *Severity ratings* of usability problems discovered.
- Fix problems → new version of interface.
- Capture *design rationale*: record reasons why changes were made.
- Evaluate new version of interface.

until time and/or money runs out.

A cycle of continuous improvement.

### Building Prototypes

- Verbal description.
- Paper prototype.
- Interactive prototype.
- Working prototype.
- Implementation of final design.

See Chapter 7.

### Formative and Summative Usability Evaluation

The usability evaluation methods are grouped according to who performs them:

- *Usability Inspection Methods*: Chapter 8.
- *Usability Testing Methods*: Chapter 9.

Both chapters include both formative and summative methods.

### 3.3.5 Follow-Up Studies

Important usability data can be gathered after the release of a product for the next version:

- Specific field studies (interviews, questionnaires, observation).
- Standard marketing studies (what people are saying in the newsgroups and mailing lists, reviews and tests in magazines, etc.).
- Analyse user complaints to hotline, modification requests, bug reports.
- Usage studies of longer-term use of product (see Section 9.9):
  - Diary studies.
  - Software logging: instrumented versions of software → log data.

- Observational studies.

### 3.4 Planning Usability Activities

1. Prioritise activities.
2. Write down explicit plan for each activity.
3. Subject plan to independent review (e.g. colleague from different project).
4. Perform pilot activity with about 10% of total resources, then revise plan for remaining 90%. [*Always* perform a pilot study!]