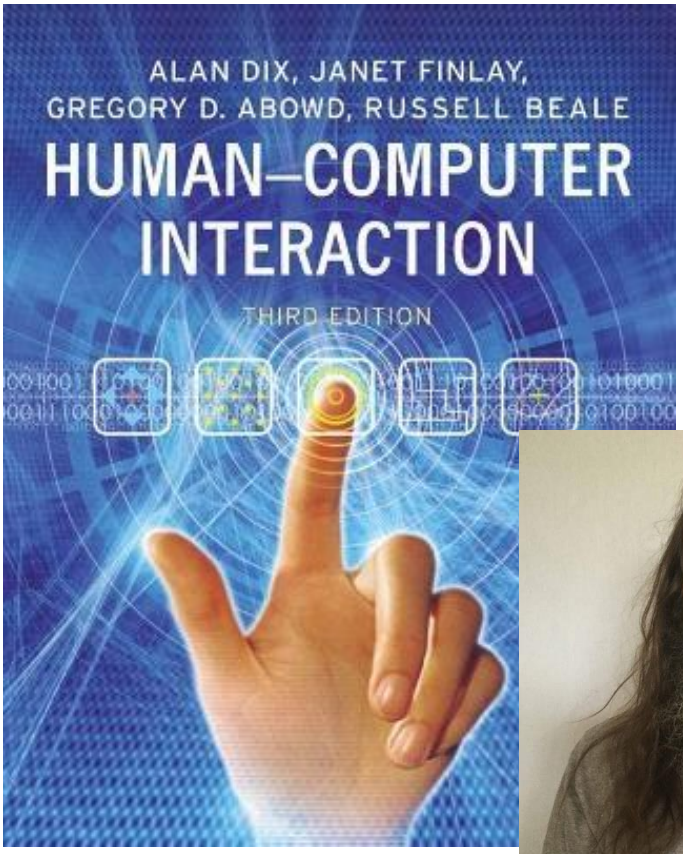
Chapter 4:

Principles to support Usability by Dix et al.

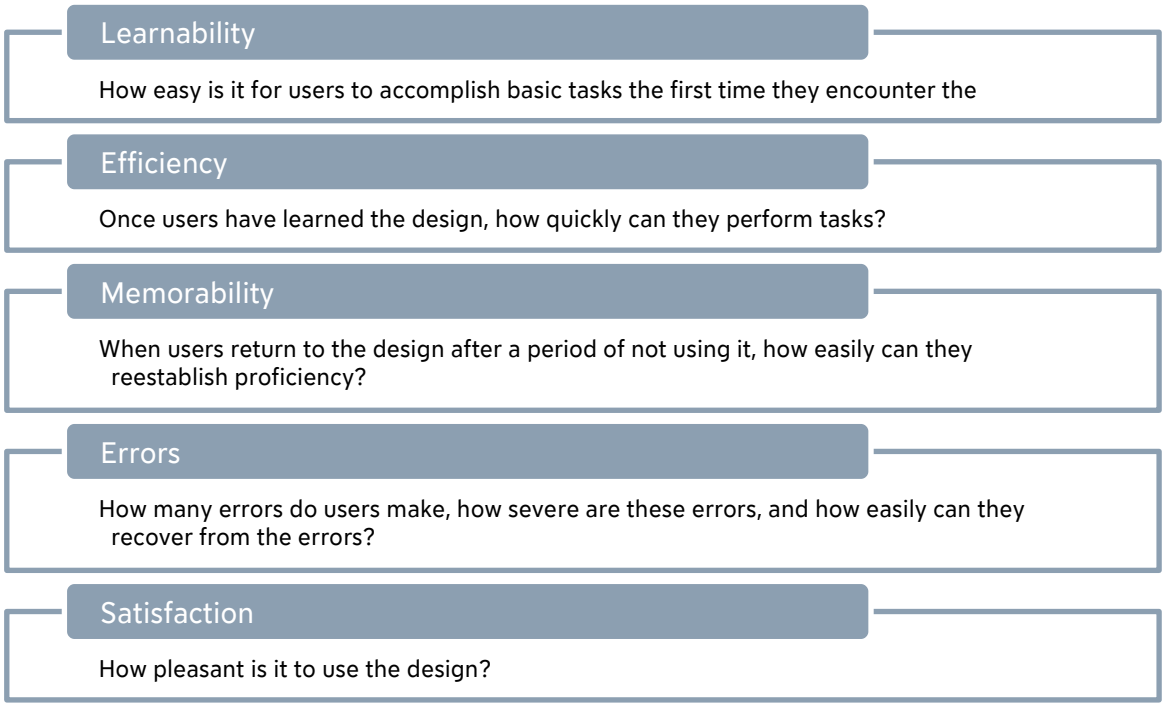
Overview



Usability 101 by Jakob Nielson

Usability is a quality attribute that assesses how easy user interfaces are to use. The word "usability" also refers to methods for improving ease-of-use during the design process.

Usability has **five quality components**:



On the Web, usability is a necessary condition for survival. If a website is difficult to use, people leave. If the homepage fails to clearly state what a company offers and what users can do on the site, people leave. If users get lost on a website, they leave. If a website's information is hard to read or doesn't answer users' key questions, they leave. Note a pattern here?

Users do not spend much time over reading a website manual or trying to figure out an interface (unless it is obligatory). If there are plenty of alternatives available, leaving is the first line of defence when users encounter a difficulty.

There are several evaluation methods for getting users' feedback to improve usability. Some are based on evaluation by UX experts, but probably the most common is usability testing with users, i.e., the actual end users carry out typical tasks with a complete product or a prototype. Together with a usability test, other methods can be used to collect complementary data. Those are, for example, questionnaires, interviews and focus group discussions.

Types of Design Rules

There is a terminology to describe the different types of Design rules.

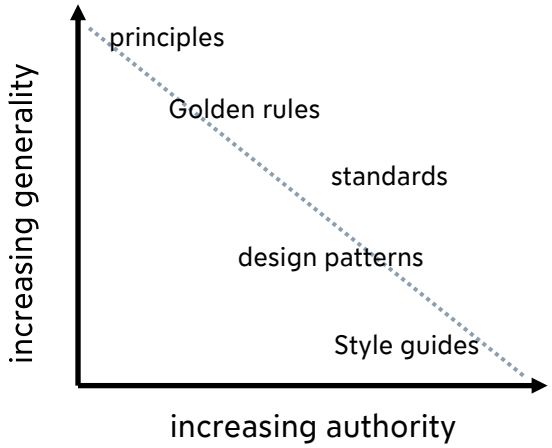
Principles:
Abstract design rules

Golden rules and heuristics:
More concrete than principles

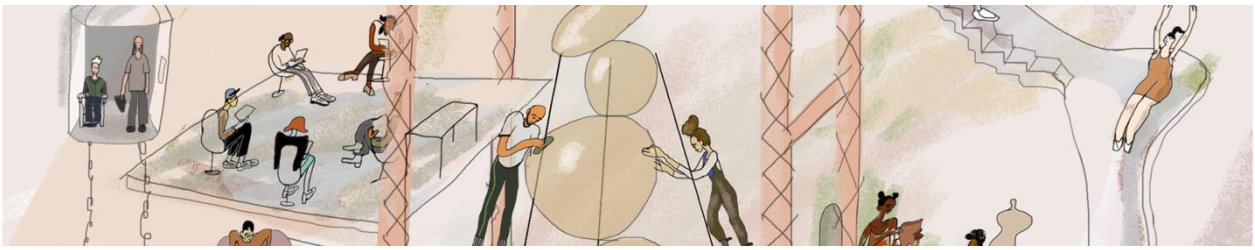
Standards:
(Very) detailed design rules

Design patterns:
Generic solution for a specific problem

Style guides:
Provided for devices, operating systems, widget libraries



Authority: whether a rule must be followed or whether it is just suggested
Generality: applied to many design situations or focused on specific application situation



For an example of a style guide, visit: https://design.google

A style guide helps to ensure a continuous product experience. It means that no matter how, when or where a customer experiences a brand or a product, they are experiencing the same underlying traits. It's this consistency across every touchpoint that helps to create an association with a brand or a product. The usage experiences feels 'complete' and consistent and helps building habits with the product.

Principle 1: Learnability

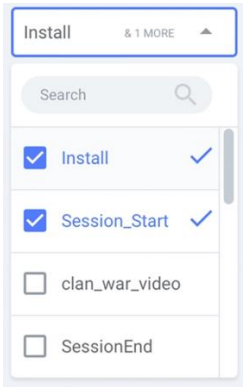
Learnability: the ease with which new users can begin effective interaction and achieve maximal performance.

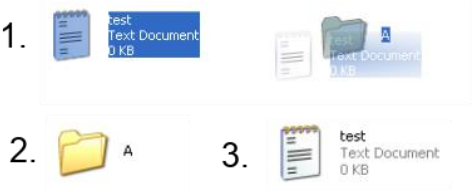
Learnability captures how well the user can start using the new system and which prior knowledge is required for this.

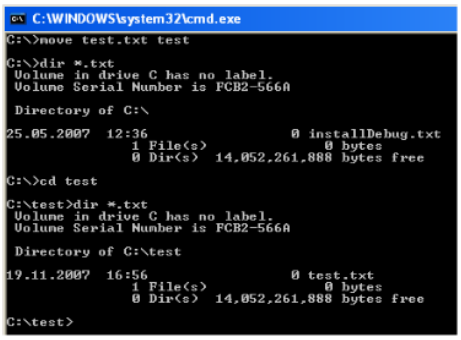
Therefore, several aspects of learnability need to be considered, which include:

Predictability Determining effect of future actions based on past interaction history and the visibility of operations

Synthesizability Ability of the user to assess the effect of past operations on the current state – this means that the user should see the changes of an operation given through immediate vs. eventual feedback







Familiarity To which extent can the user apply prior knowledge to new system – remember: affordance (guessability).

Generalizability Can specific interaction knowledge be extended to new situations

Image from the movie Star Trek IV:
The Voyage Home

Consistency Likeness in input/output behavior arising from similar situations or task objectives

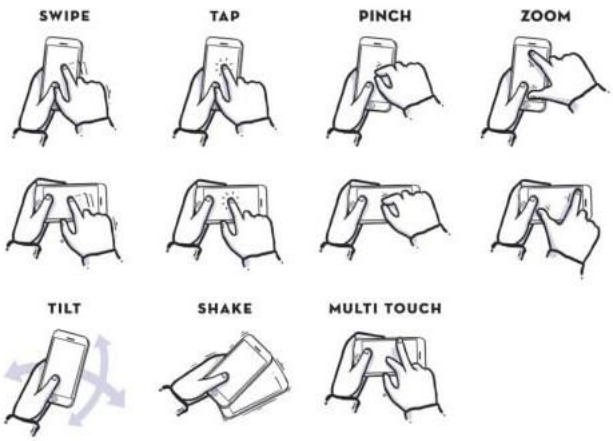


Image source: Julian Burford

Excurse: The power of gestures

Gestures allow direct changes to UI elements using touch and help users perform tasks rapidly and intuitively. Through the ubiquity of mobile smartphones, some gestures have become common synonyms for a specific action (e.g., zoom in through touching the surface with two fingers and moving them apart.)

This previous knowledge can be used in new applications and there are common guidelines from big software companies how to implement these gestures.

E.g.:
Google Gesture Guide
Microsoft Gesture Guidelines
Apple Gesture Guidelines
Android Gesture Guidelines

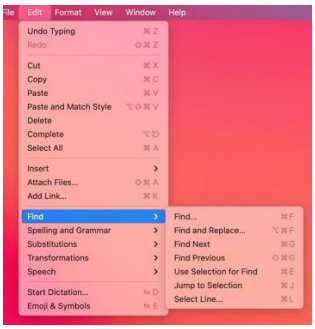
Principle 2: Flexibility

Flexibility: the multiplicity of ways the user and system exchange information.

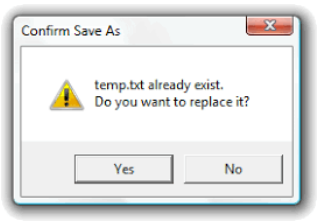
The flexibility of the interaction with a system is determined by several components, which will be introduced in this section.

Dialogue initiative

The dialogue initiative includes the freedom from system-imposed constraints on input dialog. There are two types of dialogue initiatives, which are **user preemptiveness** (the user initiates a dialog) and **systems preemptiveness** (the system initiates dialog)



User preemptiveness



System preemptiveness

Multithreading

Describes the ability of system to support user interaction for several tasks at a time. Two types of Multithreading in UX design are **concurrent multimodality** and **interleaving multimodality**.

Concurrent multimodality:

- Multi-modal dialog
- Editing text and beep (incoming mail) at the same time

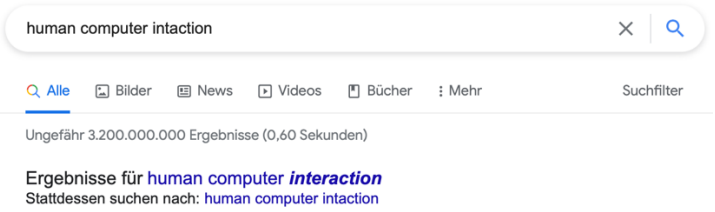
Interleaving multimodality:

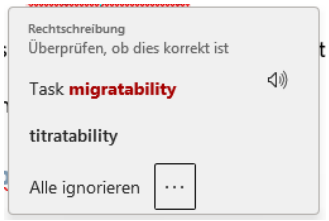
Permits temporal overlap between separate tasks, dialog is restricted to a single task

- Window system, window = task
- Modal dialogs
- Interaction with just one window at a given time

Task migrability

Some responsibilities for a given task can be passed from a user to the system, e.g., spell check in a word processing program





Substituitivity

A system needs to allow equivalent values of input and output, that can be substituted for each other. → Representation of multiplicity

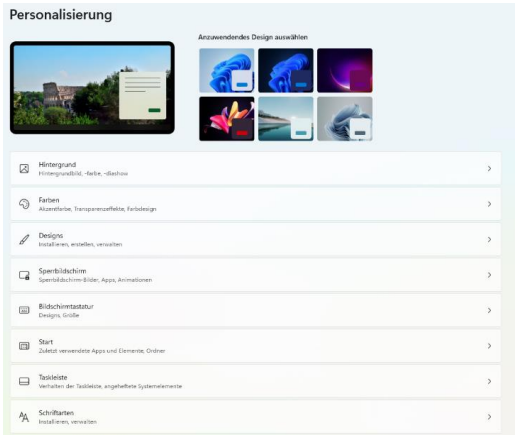
e.g.: different currencies, cm or inch

Customizability

The user interface needs to be modifiable by the user (adaptability) or the system (adaptivity). Both terms describe "how" an intelligent mechanism used to describe the "how" an intelligent mechanism can achieve the goal of tailoring a UI to a specific user, e.g., through combinations of components and attributes

Adaptability

users' ability to adjust the form of input and output.→ The user is actively and continuously involved in the adoption process of the UI



Adaptivity

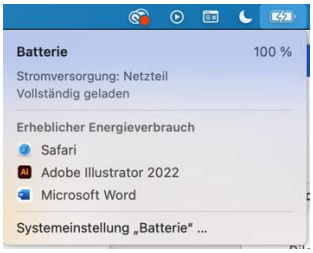
automatic customization of the user interface by the system.
→ The system collects user information based on his or her activity.

Principle 3: Robustness

Robustness: the level of support provided to the user in determining successful achievement and assessment of goal-directed behaviour.

Observability

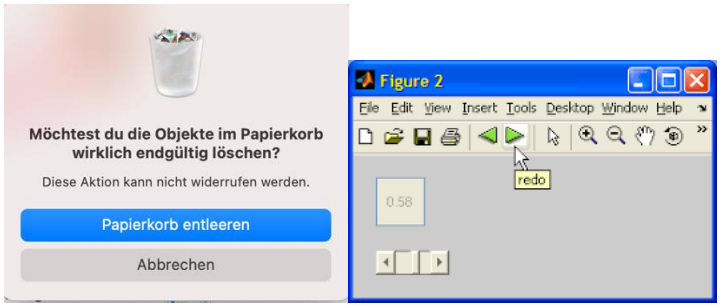
Ability of the user to evaluate the internal state of the system from its perceivable representation



A system is considered “observable” if the current state can be estimated by only using information from outputs – in a visual interface, this is must be displayed information that is accessible for the user.

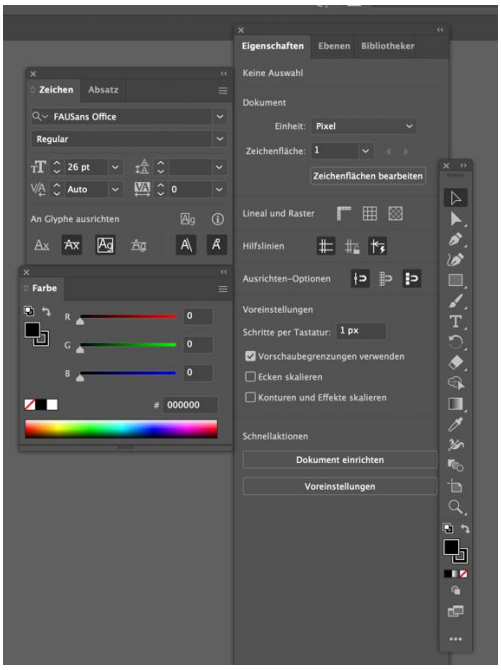
Recoverability

Ability of the user to correct a recognized error:
Reachability (states): forward (redo) / backward (undo) recovery
The effort for a given task should be adequate to the importance or consequences of it: e.g., more effort or steps should be necessary to deleting a file then just to move them.



Task Conformance

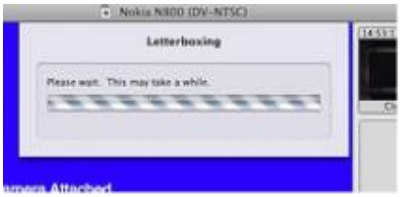
Degree to which system services support **all tasks** of a user.



However, one should keep in mind, that by adding more functionalities to a interface, this will increase complexity and the ease of use. Thus, the balance between supporting tasks but not overloading users should be kept.

Responsiveness

Describes how the user perceives the rate of communication with the system. The preferred perception should be short and contain instant responses.

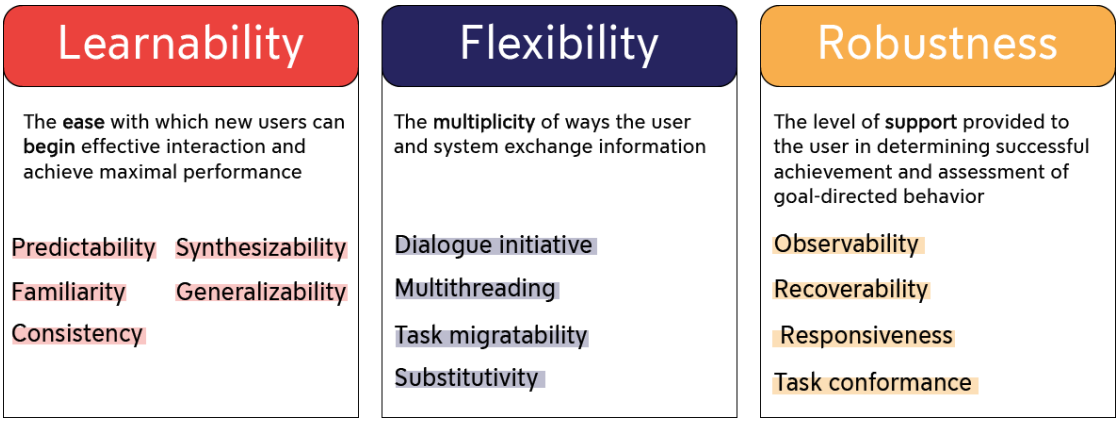


Letterboxing: Please wait.
This may take a while.

What happens if a system crashes and does not react anymore?
Which consequences does this have on the UX/UI?

Summary

Poor design criteria are responsible for wasting computer users time and are a hindrance to effective interaction with human centred systems. It is important that before any design of an interface is attempted an in-depth analysis of task and user needs must be undertaken. Therefore, Alan Dix has proposed a taxonomy of three design principles which help to guide developers in the design process of user-friendly systems. The three principles comprise **Learnability**, **Flexibility**, and **Robustness** with respective sub-categories.



References

MacKenzie, I. S., Sellen, A., & Buxton, W. (1991). A comparison of input devices in elemental pointing and dragging tasks. Proceedings of the CHI `91 Conference on Human Factors in Computing Systems, pp. 161-166. New York: ACM.

From: John, Bonnie and Kieras, David E., The GOMS Family of User Interface Analysis Techniques: Comparison and Contrast, ACM Transactions on Computer-Human Interaction 3,4 (December 1996b), 320-351

Harold Thimbleby. User Interface Design With Matrix Algebra. ACM Transactions on Computer-Human Interaction, Vol. 11, No. 2, June 2004, Pages 181–236

Card, S. K., Moran, T. P., and Newell, A. 1980. The keystroke-level model for user performance time with interactive systems. Commun. ACM 23, 7 (Jul. 1980), 396-410.

Kay, A. User Interface: A Personal View. In Brenda Laurel (ed.), The Art of Human-Computer Interface Design. New York: Addision-Wesley, 1990, pp.191-207. Cited accroding to Virtual Reality and Abstract Data: Virtualizing Information. By Michael B. Spring and Michael C. Jennings. University of Pittsburgh. (http://www2.sis.pitt.edu/~spring/papers/abstdat_vr.pdf)

Urban Stress: Experiments on Noise and Social Stressors. DC Glass, JE Singer - 1972 - Academic Press Wilfred J. Hansen, User Engineering Principles for Interactive Systems, 1971 Jef Raskin, The Humane Interface, ACM Press 2000