More Prototyping / Intro To Evaluation

* Material from: [Klemmer, Stanford]

Quantity vs. Quality

- Is it better to produce a large quantity of designs, or to focus on creating the best one design?

Quantity vs. Quality

- Bayles and Orland put this to the test.

http://jamesclear.com/repetitions

Quantity vs. Quality

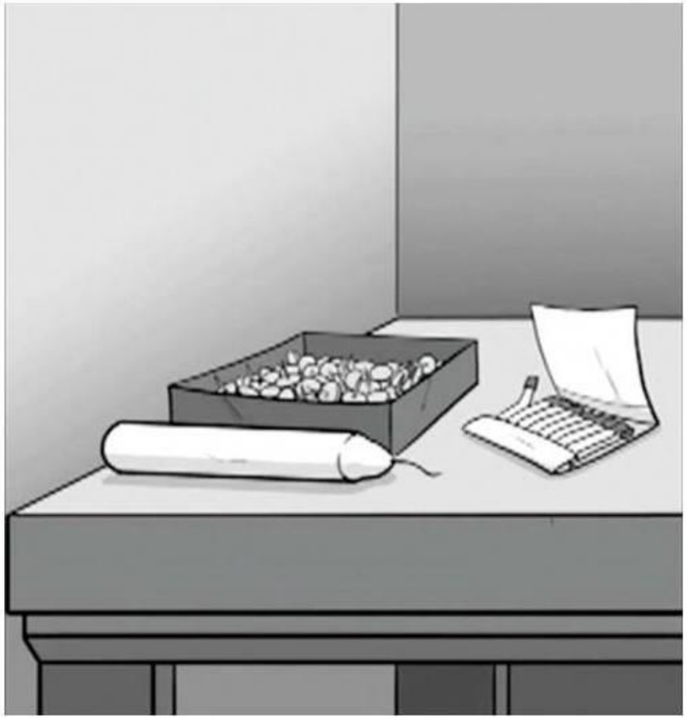
- ***Well, come grading time and a curious fact emerged: the works of highest quality were all produced by the group being graded for quantity. It seems that while the "quantity" group was busily churning out piles of work - and learning from their mistakes -- the "quality" group had sat theorizing about perfection, and in the end had little more to show for their efforts than grandiose theories and a pile of dead clay***. [Bayles, Orland]

Duncker's candle problem

The test presents the participant with the following task: how to fix and light a candle on a wall (a cork board) in a way so the candle wax won't drip onto the table below.

- To do so, one may only use the following along with the candle:
    - a book of matches
    - a box of thumbtacks

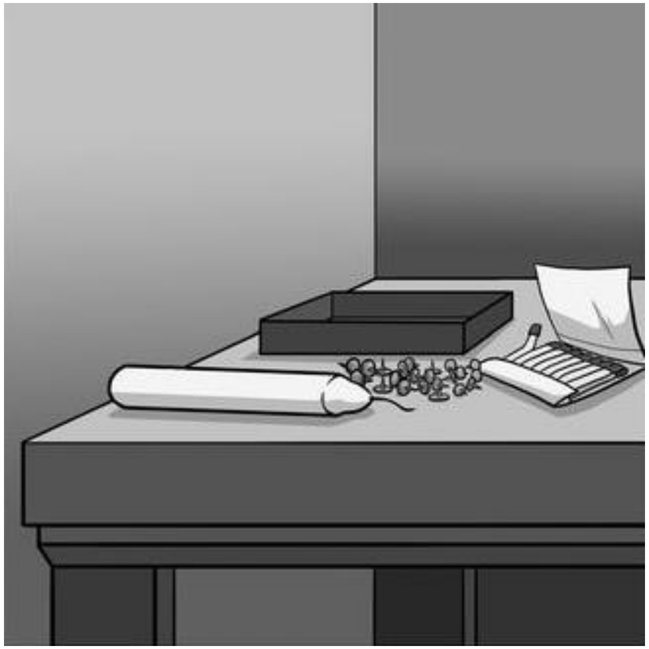
Functional Fixation

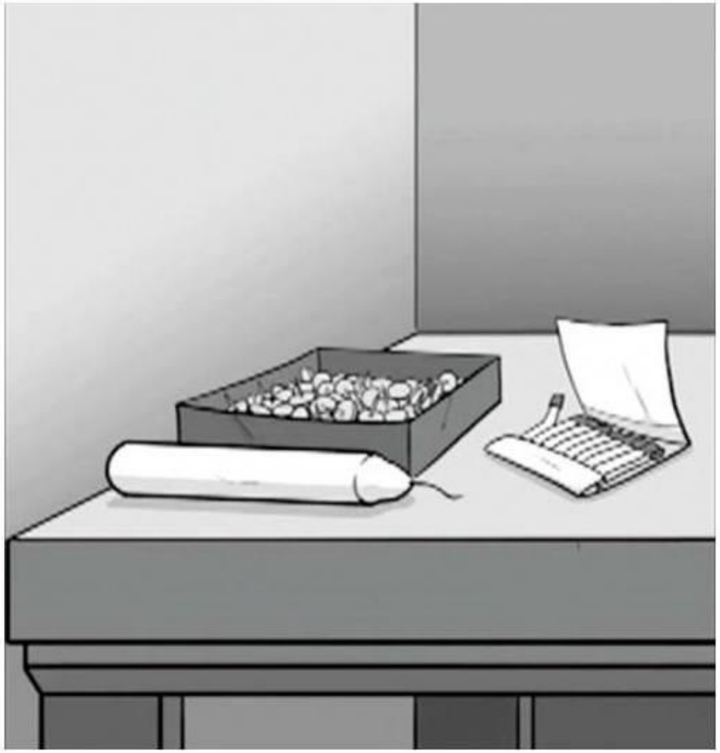


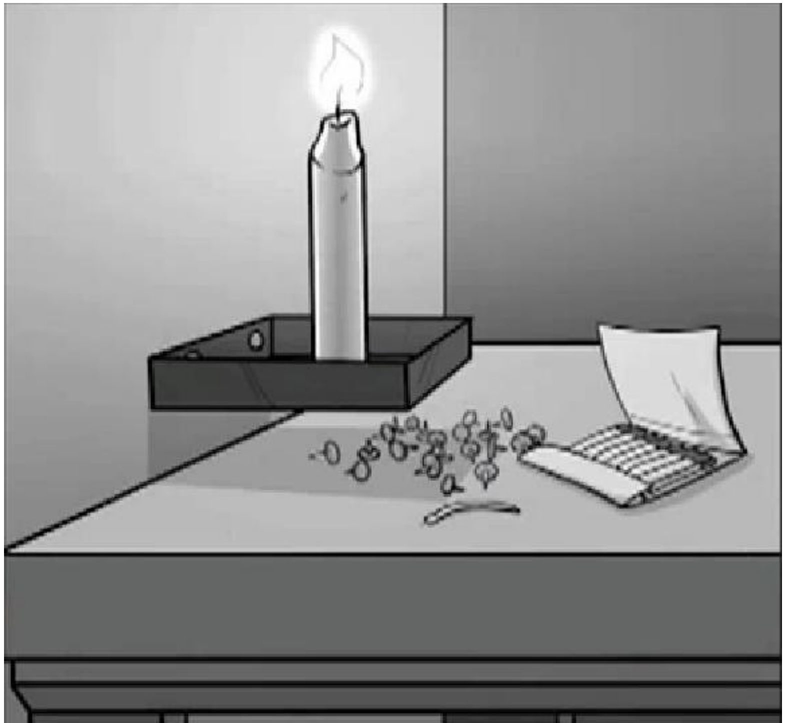
The concept of functional fixedness predicts that the participant will only see the box as a device to hold the thumbtacks and not immediately perceive it as a separate and functional component available to be used in solving the task.

[Duncker, 1945]

Functional Fixation







[Duncker, 1945]

Funny Video
“Functional Fixedness” or not……

- https://www.youtube.com/watch?v=ksgaup4zqz0

Better Approach?

- Parallel Prototyping:
  - Making multiple prototypes in parallel
  - Studies show that this leads to better designs!
    - i.e., designs led to statistically higher values in quantifiable variables of interest (more on this later).
    - *Klemmer, Gentner, Loewenstein, Thomson, etc.

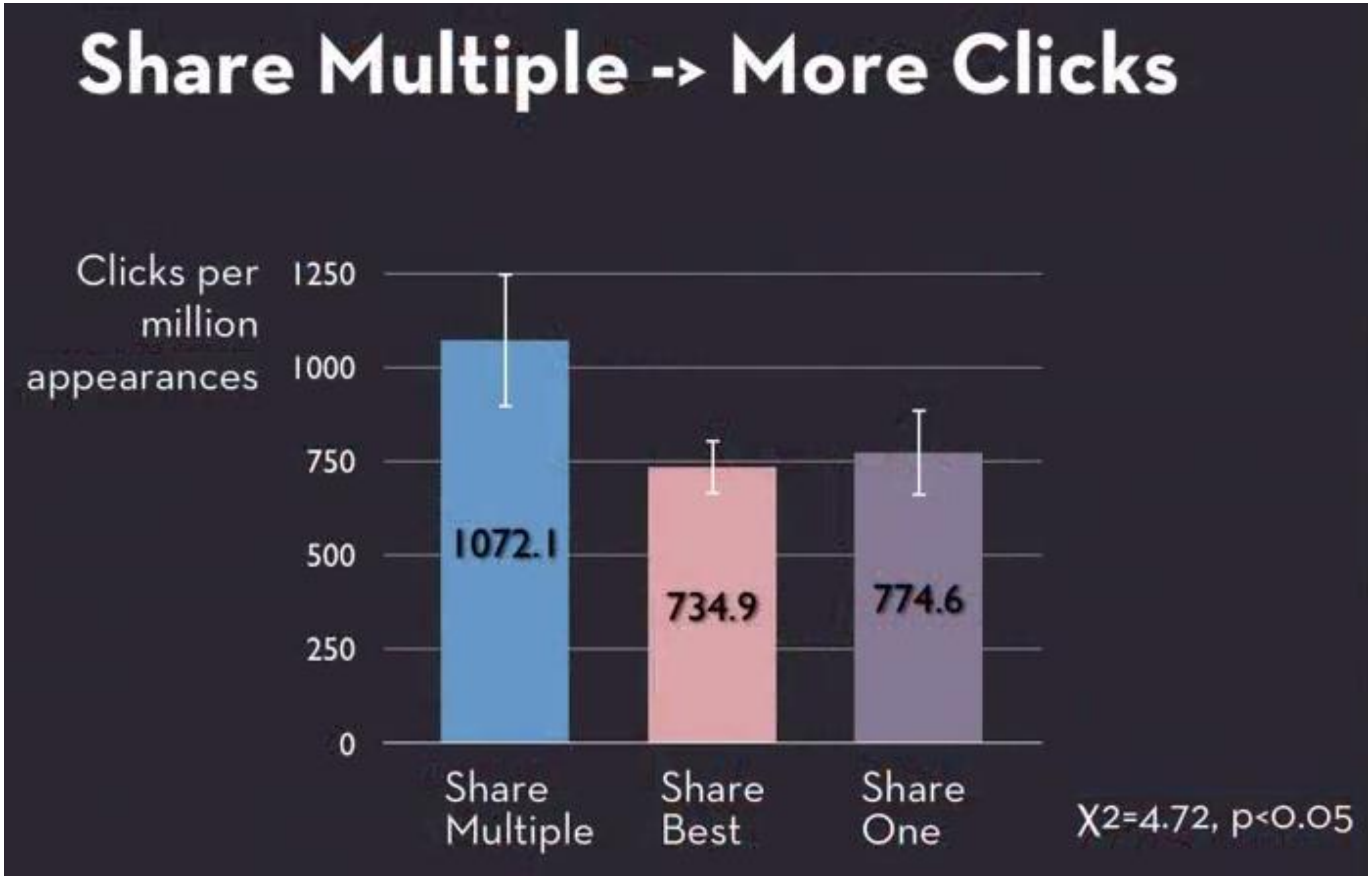
- Separates Ego from Artifact
  - i.e., a criticism of one design is NOT a criticism towards the designer.

Parallel Prototyping

- Another advantage:
  - Supports TRANSFER of positive attributes across designs.

- Study:
  - Three groups (n=84)
  - 1: each member made multiple prototypes and shared them among team.
  - 2: each team member shared only their best prototype.
  - 3: each team (as a group) created a single prototype together (only one design).

Parallel Prototyping



*Klemmer

Benefits of Sharing Multiple Prototypes

- More individual exploration
- More feature sharing
- More conversational turns
- Better consensus
- Increase in group rapport

Ok fine…so how do we compare prototypes?

- We perform an evaluation!
- An ***evaluation*** is an experiment (or set of experiments) meant to provide answers to at least one design question.

- Prototyping Exercise – see Collab, Assignment, In-class prototyping exercise

Starting An Evaluation

- MUST have a research question!
  - Typically, these research questions are related to your usability requirements!
  - ‘Which design accomplishes usability requirement X best!”
- It is very common to ask things like “Do you like my interface?”
  - Anything wrong with just asking people this?

Leading Questions

- Firstly, ‘Do you like my interface?’ is a leading question!

- A ***Leading Question*** is a question that suggests the answer the examiner is looking for or contains the information the examiner is looking to have confirmed.

- Don’t ask users leading questions!

Please the Experimenter Bias

- People want to make you feel good about your work (because it’s assumed that you worked hard).
- So users will tend to say ‘Yes’ to this question.
    - How do we get around this?

Getting Beyond “Do you like my interface?”

- Ways to get around please the experimenter bias
  - 1) Double-blind studies
    - i.e., Both user and facilitator don’t know which experimental group the user is in.
  - 2) Don’t let the user know what you are measuring / what you care about (until study is over).
    - E.g., Don’t tell the user that the number of mistakes while typing is being measured.
  - 3) Ask questions that cancel each other out
    - E.g., Ask about how useful the interface was AND how frustrating it is. User can’t tell which you care about.

Getting Beyond “Do you like my interface?”

- If possible, evaluation measures (quantifiable variables) should ALWAYS have a baserate.

- Baserates: How often does ‘Y’ occur in the current setting (if one exists)?
  - Very reasonable for some of your projects, if there is a competing product that exists.

Getting Beyond “Do you like my interface?”

- Baserates: How often does ‘Y’ occur in the current setting (if one exists)?

- Example: “User will make less than 3 mistakes while performing task X”
  - Where did the 3 come from?
  - If 3 is the average that users make on some competing system, than that is a good baserate!
  - If 3 was made up, then it is not. It provides little context.
  - * Need to define what ‘mistake’ means clearly…but that’s another story.

Getting Beyond “Do you like my interface?”

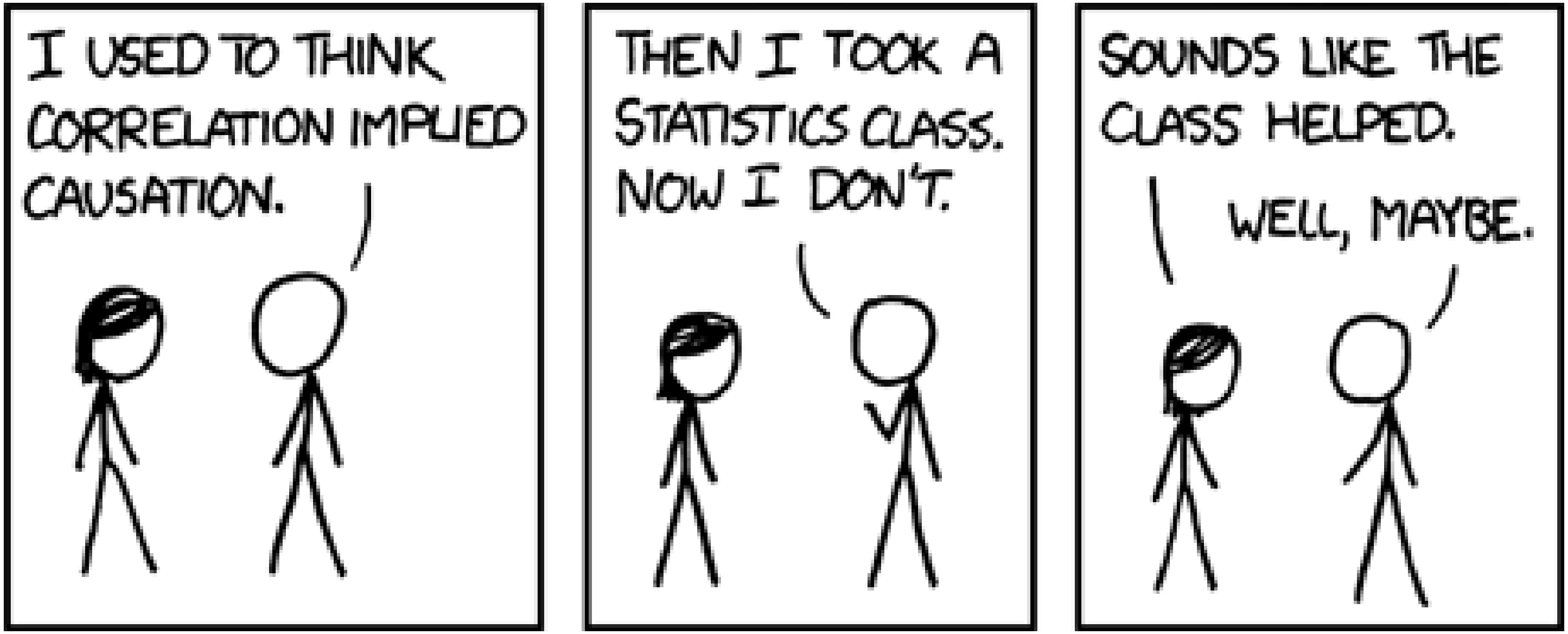
- Correlations: Do X and Y co-vary?
  - Requires measuring X and Y.
  - Probably need two prototypes OR two versions of a prototype (each with different X).

- Causes: Does X cause Y?
  - Requires measuring X and Y (establishing correlation).
  - Requires establishing time precedence.
  - Requires controlling for all confounding variables.

Examples of Correlation != Causation

http://www.buzzfeed.com/kjh2110/the-10-most-bizarre-correlations

Correlation != Causation


I USED TO THINK CORRELATION IMPLIED CAUSATION.
THEN I TOOK A STATISTICS CLASS. NOW I DON'T.
SOUNDS LIKE THE CLASS HELPED.
WELL, MAYBE.

*from http://xkcd.com/552/

A few terms:

- Independent Variables
  - Variable that is being manipulated to study an effect via a change.
- Dependent Variables
  - Variable that is measured for change after IV is altered.
  - Time to complete a task
  - Emotion

Example

There is no difference in user performance (time and error rate) and preference (5 point likert scale) when typing on **two sizes** of an **alphabetic, qwerty and random** on-screen keyboard using **a touch-based large screen, a mouse-based monitor, or a stylus-based PDA**.

IV in bold

DV in red

Internal Validity

- Can you reproduce the experiment multiple times yourself!?
  - Same prototypes
  - Different users
  - Same experimental setup, conditions, etc.

External Validity

- Does your experiment apply generally to other ‘outside’ settings?
    - Different users selected from a different “pool”.
    - Different prototypes with same general independent and dependent variables.
    - Different designers running the experiments.
    - Etc…
- In short, EV means your results apply generally to experiments with the same abstract characteristics as yours.

Questions?

Daily Movie

- http://www.youtube.com/watch?v=kCSzjExvbTQ

iPhone Study

- Study meant to measure the usability of typing on the iPhone.
- Bring in two groups:
  - Numeric Keypad users
  - QWERTY users
- Measure typing speed (wpm) for each group on the new iPhone.

iPhone Study

- Manipulation: Input Style
- Measure: Words per minute

- Internal Validity: Yea…probably!
- External Validity: Not so much, why?

- Results: Both groups did the same (and did poorly compared to their own phones).

- *Pc world, User-Centric

Better Version

- Comparison of:
  - iPhone users (after a month at least)
  - QWERTY users
  - Numeric users

- This time, measure BOTH speed and error rate while typing!

Results

- iPhone and QWERTY users type about the same speed (numeric much slower).
- But…iPhone users make MANY more errors!
- *Note: Any problems with this study? Think about the users in each group?

Self Selection

- “Self selection occurs when the experimental groups are chosen by the participants in some manner.”

- Even better study?
  - Select participants for each phone group at random.
  - Train them on the phone for a while.
  - Run the experiment as before.

Strategies For Fair Comparison

- Insert new approach into an actual production setting.
- Re-create the production approach in your new setting.
- Scale things down so you’re looking at a piece of a larger system (most relevant for you!).
- When expertise is relevant, train people BEFORE running your study.

Is Interface X better than interface Y?

- What does better mean?
  - Need explicit measures when possible.

- Nearly always, your answer will be that ‘It Depends’.
  - So, more interestingly, what does it depend on?
  - Figure that out! And then describe your results!

Controlled Comparison Enables Causal Inference

- When possible, control as many external variables as possible.
  - Users placed in groups randomly.
  - Users perform experiment in exact same environment.
  - Users are given the exact same instructions, training time, etc.
  - Any outside variable that could effect the result of the study should be equivalent for ALL test subjects.
  - If done well, then the third requirement (confounding factors) for causal inference is satisfied.

Types of Evaluation

- Heuristic Evaluation
  - Experts look at a system and analyze carefully.
  - Produce a report of usability problems.

Types of Evaluation

- Quantitative Analysis
  - Perform an experiment that involves the collection of quantitative data (numeric data or data that can be translated into numeric data).
  - Run statistical tests to evaluate differences across prototypes.

Types of Evaluation

- Qualitative Analysis
  - Collect non-numerical data.
  - Conversation transcripts, general observations, etc.
  - Analyze for broad consistent patterns.

Usability Test

- **Guerilla Testing with Usability Cafe**

What’s Coming Up!

- Let’s look at each type of evaluation in more detail!
    - Heuristic Evaluation
    - Quantitative Evaluation
    - Qualitative Evaluation

Heuristic Evaluation

The main goal of heuristic evaluations is to identify any problems associated with the design of user interfaces.

The simplicity of heuristic evaluation is beneficial at the early stages of design. This usability inspection method does not require user testing which can be burdensome due to the need for users, a place to test them and possibly a payment for their time.

Heuristic Evaluation

Small group of people trained in heuristic evaluation examine a user interface.

Each evaluator list the problems with respect to established usability principles (heuristics) (Nielsen 1995)

Each evaluator ranks problems by severity.

1. List problem with interface.
2. Give severity score.
3. Associate each problem with the heuristic

Heuristic Evaluation

Go through the interface at least twice

1. Explore the interface
2. Find problems

Generally exploring and not performing a real task

Output

List of severity rating, heuristic, problem detail

Remember: not just a critique but a **reasoned critique** that references a heuristic (site which heuristic that has been violated)

Heuristic Evaluation

Severity
Rating Heuristic Detail of problem

____________________ ______________________________

____________________ ______________________________

____________________ ______________________________

____________________ ______________________________

____________________ ______________________________

____________________ ______________________________

Heuristic Evaluation

- Idea of heuristic evaluation is independent of list of heuristics.
- Could be a list of any list of established usability principles to find problems.
- Could be domain specific (e.g. designing for children or older adults)

Most commonly used: Jakob Nielsen's list of 10 heuristics are probably the most-used usability heuristics for user interface design. Nielsen developed the heuristics based on work together with Rolf Molich in 1990.

Jakob Nielsen's list of 10 heuristics

1. **Visibility of system status**:
   The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
2. **Match between system and the real world**:
   The system should speak the user's language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
3. **User control and freedom**:
   Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
4. **Consistency and standards**:
   Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
5. **Error prevention**:
   Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

Jakob Nielsen's list of 10 heuristics

5. **Recognition rather than recall**:
Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

6. **Flexibility and efficiency of use**:
Accelerators—unseen by the novice user—may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

7. **Aesthetic and minimalist design**:
Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

8. **Help users recognize, diagnose, and recover from errors**:
Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

9. **Help and documentation**:
Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Heuristic Evaluation Severity Assessment

0 = This is not a problem at all

1 = cosmetic problem only – fix if extra time is available

2=minor usability problem – fixing should be given low priority

3=major usability problem – important to fix – given high priority

4=usability catastrophe- imperative to fix before product can be released

Use average of all evaluator’s scores

Heuristic Evaluation

- Heuristic Evaluation can be done throughout design process (prototypes, live systems)
- Heuristic Evaluation is a complement to not a replacement to user testing.
- Low cost however, it will not uncover all problems. User testing is critical

Heuristic Evaluation

- Focus is not on problem solving, but on problem identification.
- Easier to find minor problems using heuristic evaluation than user testing

Sources

How to Conduct a Heuristic Evaluation - https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/