# INTERACTION DESIGN

## 1. DESIGN PROCESS

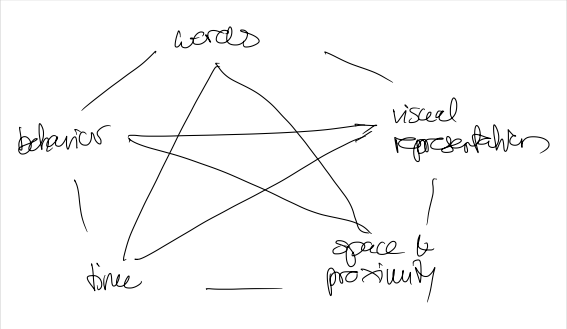
Design is:
- look and feel
- structure, ordidecture, layout
- formalised specification like diagram, prototype

When: at start, iterative, formative, not in isolation

good: generalisable features, prioritise-importance & relevance

**Design Iteractions with:**
- context
- implications
- impact (what, whom, how → affections, point of views)
- universal whenever possible → validation
- utility & pleasant
- test

**5 dimensions**:

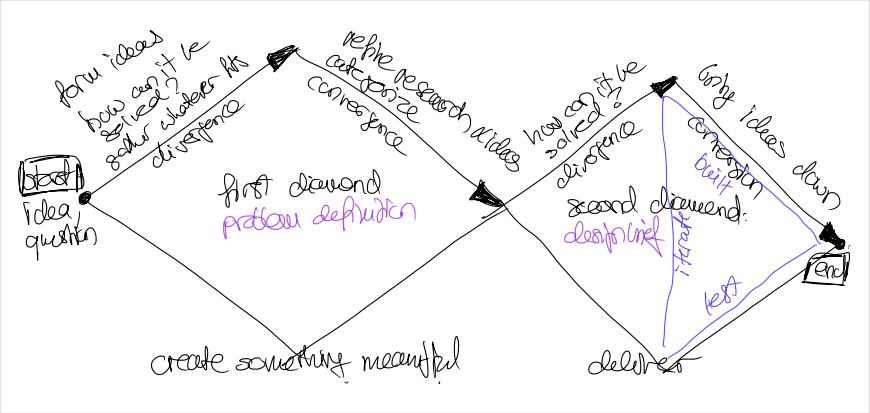


**WIMP**: Windows; Icons & signs; Menus & sub menus; Pointers

**user experience:**
usability is a subset of interaction design and it is about worldly processes and how to improve them

keep in mind: digital world is different from the real
physical world so be careful about the richness of features (there is a too much)

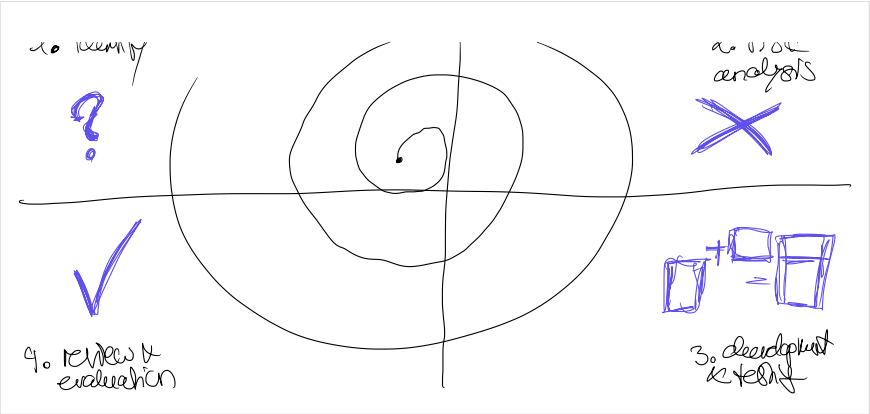
**Double diamond:**




# 2. MODELS

e.g. house
"strichmannchen"

**spiral model:**
1. . identify object
2 perform risk analysis
g. • development × testing
4. review & evaluation



**Waterfall**: step by step
- requirements
- design
- implantation
- verification
- maintenance

stages of a design: architecture, detail design, change management & verification

Model Computer: x inputs + processing power = y outputs

Model user: inputs (senses (read, listen,...)) + processing power (cognition) - output (type, move, talk....)

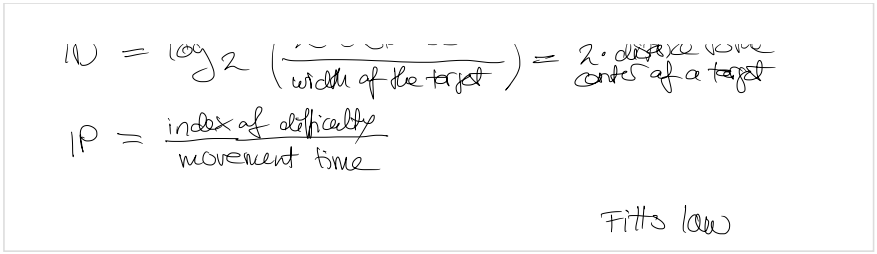
**Fitts law:**
about the way in which we model human movement in relation to objects on the screen

ID: index of difficulty

IP: index of performance per second

mt: movement time

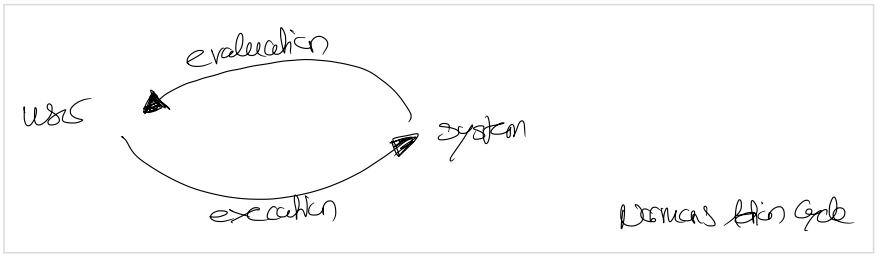
TP: throughput through a network




**GOMS**:

G: Goal, 0: Operations, M: Method , s: Selection

**Normans Action Cycle:** better understanding of the user




There is usually a gap between physical system and the goals. This gap is the execution. In this gap different states or choices are possible. The next gulf is about evaluation & corresponding feedback

**mental models:** e. g. user feels cold → wants to get warmer → increase temperature (thermostat)
Many user increase it more than needed, because they believe it will warm up quicker even though that's
physically not possible.

other examples:
blue text, underlined = link; increase, close icons: windows & linux on the right, Mac on the left

**7 stages of action:**

1. get data input
2. Think about wht to do with the input -> interpretation
3. Evaluation of the interpretation -> take a decision to get (closer) to the goal
4. Intention to act
5. Conceptual: act
6. Execution of action (e.g. 4. = get jacket, 5. get up walk to jacket, 6. get the jacket)
7. Feed back into the real world (7. holding the jacket != warm -> next circle (repeat 1-7) to wear the jacket

# 3. UNDERSTANDING USERS

What information is needed to design for particular users?
What are user experiences?
It is necessary to understand the complexity of the use case to understand how we can impact the experience.

motivations:a need or desire supported by an action to achieve a goal.

- users' need from system
- how does the interaction manifest (action)
- route a ways to achieve goal
- facilitate interaction

⇒ understand how & what & why a user tries to achieve something

Usually a task can be broken down into a series of formative steps.

**environment**:
- system has to fit in
- we don't work in isolation ← → context impacts interactions systems can be designed for specific environments (e.g. sun, night)
- physics have an impact (e. g. mobile phone & rain)
- consider standards & legal aspects

**types of data:**

- quantitative: numeric, figures, orders, representation
- qualitative: descriptive, context-driven, experience, scenario → is more dynamic, more difficult to generalise
- both are useful, depending on the task one can be more useful than the other

**interviews**:
- is a conversation
- interviewer listens, keeps own opinion to oneself
- get detailed rich data
- adapt to what a user wants to tell you so you can discover the important parts
- types:
    - open-ended (topic/area to talk about)
    - semi-structured (list of questions, possible to dig deeper
    - structured (ordered questions, nothing else)
- traps:
    - keep quiet & listen
    - not to correct the interviewee, or clarify something, or add your own opinion
- plan your interviews
- use simple & clear questions
- only pose important questions
- iterative, learn from interciew to interview, reflect yourself
- find participants (think about experience, what to learn, reimbursement for their time)
- running the intoliew:
    - everyone comfortable
    - ask if allowed to record
    - take notes
    - decide on questions (keep time in mind)
    - give an estimated duration
    - tell participant they can stop introiew any time
- tools:
    - artefacts
    - sketch paper
    - ice Creakers (e. g. small talk questions)

**focus group**:
- It's sort of an interview with many people, and it is
usually seem structured.
- it's less intimitating for users and they can share their ideas. Additionally, they are

were efficient then 1: 1 interviews, and more ideas can be generated.
- groupthink, power dynamic, only consider loudest person, or hang issues voiary out if believing nobody else shares the same are the regatie aspects.
- Communicate the arrangement early, ask for recording- permission, have questions prepared and take over the conversation lead, and might hire a facilitator to run the focus group.

**questionnaires**:
- find important questions with interviews
- closed questions: often gravitate towards odd numbers, not really valid when always neutral option gets defen
- shot questions & process → only important questions

**observations**:
- see what use does (but: natural behaviour? →
  Hawthorne effect: people change behaviour when they are under observation}
- ethnography
- introspective: observe yourself

**using existing data:**
- log data (like analytics)
- reviews

# 4. STARTING DESIGN

guidelines - design systems -design patterns

**Guidelines**

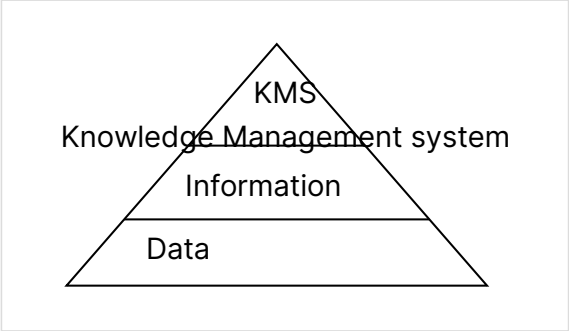
**Ben Shneiderman: 8 golden rules** about Interface Design
1. Consistency: use the known
2. usability: for all (newbies × experts)
3. Feedback: about events
4. Dialogue: let the user know and not guessing
5. Error handling: explain what went wrong
6. Reversal of actions
7. Internal location of control, give users options

8. Reduce short term memory load

**Design standards / design systems:**
- good for safety, accessibility,....
- can be rigid & not very granular or detailed
- less abstract than guidelines, less ready made then design patterns

**Design patterns**
- comes from architecture
- iterative
- focus more on what then why
- easy to follow

**Information architecture**: how data is organized & structured
- take some data & do something to give it some meaning
- store it in a specific way
- structure ← → accessibility
- categorize information in a meaningful way
- LATCH: Location, Alphabet, Time, Category, Hierarchy
- card sorting: components on cards, let user arrange the cards
- collating data has three levels




**requirements:**
- needs, wants, how, fundamentals, considerations
- the process of formality requirements can be done in different ways (UML, use cases, cost diagram)
- Formal requirements: can be a checklist
- functional requirements: focus on Input/output

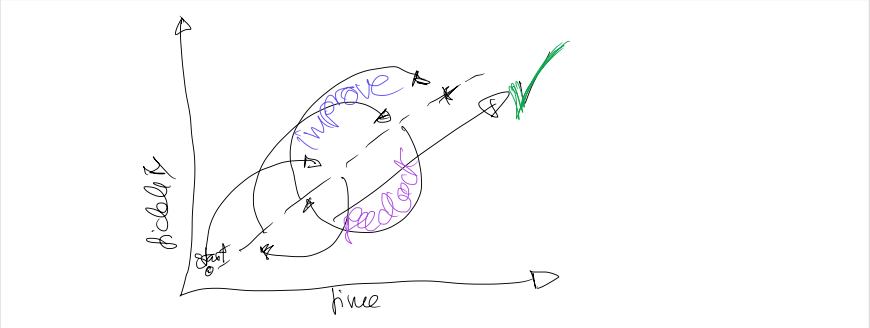
- non-functional requirements specify the underlying architecture
- specify to be able to verify 8 validate requirements so that they can be measured

**Personas & scenarios:**
A persona is a generalised user. Then describe their behaviour scenario based and with contextual behaviour.
The scenario helps to better understand the various effects.

# 5. PROTOTYPING

from very basic sketches to sophisticated complex interactive prototypes = from low to high fidelity consider feedback, users, therefore iterate & develop

usually starts with some basic scribbling of components, e.g. buttons, storyboard. Then navigation, structure gets added.




Always consider user, context, goal in each phase.

# 6. ACCESSIBLE AND INCLUSIVE DESIGN

**Accessibility:** Making a system usable by as many as we can
- consider users possibilities e. g. internet connection, disabilities
- usability: test if the system is usable (not the user)
- poor design
- contest is king
- over county problems (bridging)

- avoid unnecessary components (like animation)
- terminology
- language
- talk & listen to users → adapt their terminology & language
- be careful about discrimination & being offensive
- there is a disability language style guide
- WCAG: Web Content Accessibility Guidelines (from DNS to render content)

types of accessibility:
- temporary: vary from case-to-case
    - earphones
    - driving & listening to music → road signs become "more "important

**requirements:**
- law
- cases
- community
- demographics
- pragmatism

"Better system are those that encapsulate things we equality & equity. Universal design help to build quality into the system."

**4 principles of WCAG:**
1. perceivable
2. operable
3. understandable
4. robust

**Inclusive design & assistive technology:**
- exclusive: persistence in long term, short term (injuries), isolated incidents (in a foreign county & not understanding)
- inclusive: more than one case (consider all circumstances) I humans are diverse, embed all good things
- extending criteria: not just 3 cases, wide spectrum (beyond min, max, percentile)
- If possible build-in else bolt on
- challenge is the norm
- involve users, be empathic

- assistive technology: helps to support people in interactive processes and consists of a range of tools e. g. screen readers)
  - visual
  - hearing
  - mobility
  - motor control
  - contemporary leg-BC1-Brain computer toffees)

Cultural & universal accessibility: important if designing for different markets

# 7. HUMAN ERROR

- many different types (description, testing, mental models,.. l
- user-centred issues
- can't be avoided entirely but partly mitigated
  - importance
  - type
  - classification
  - when do they occur? (e. g. context with memory?)
  - reduce potential errors (→ prediction)

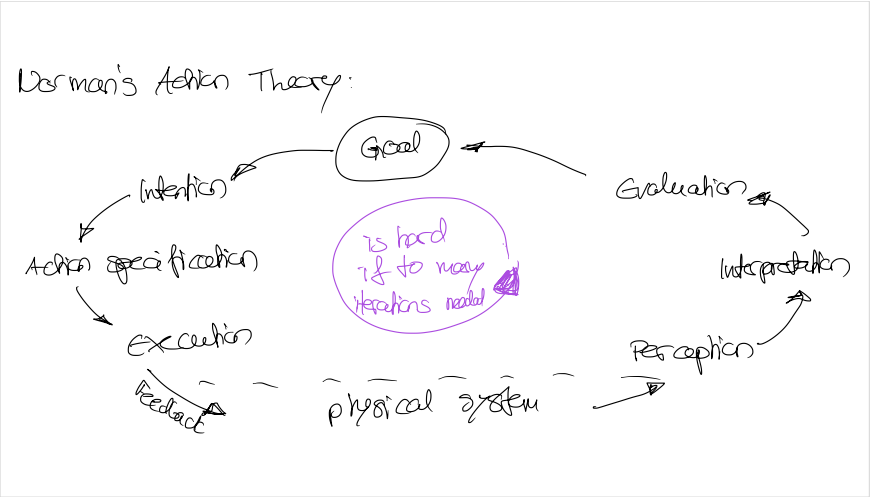
**error:** In error is essentially a system failing to achieve some intended outcome from some plan sequence of activities.

- repetition of action
- loss of activation (forget goal)
- description applied to the wrong task
- crosstalk (sequential): switch steps between currant and previous task
- crosstalk (concurrent): switch between two tasks
- post-completion error: goal achieved but then something went wrong

**slips:** A ship is an incorrect execution of a correct action sequence

- capture slip: e. g. autopilot-well learned routine overrides current indented activity
- perceptual confusion slip: incorrect mental schema applied to similar looking objects (e. g. same design for chili & eye drops)
- omission slip: omitting one step of a task

**mistakes**: A mistake is the correct execution off an incorrect sequence of action.




**queues**: set of steps performed → ask if something shall really be" overstepped".

**lockouts:** used to ensure people do the right things the right way

# 8. CONTROLLED EXPERIMENTS

An **experiment is** a scientific way to explore something.

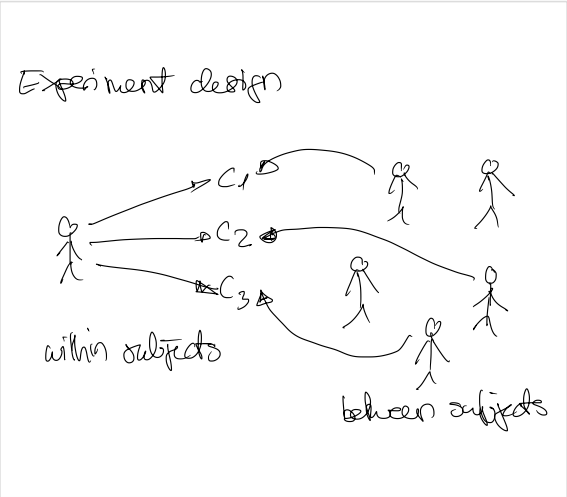
- can be described as a test
- can be build around  provingsomething
- can be based around discovery
- engages with a rigorous set of processes
- should be systematic, repeatable
- control for confounding variables
- measure things

**hypothesis testing:**
- null hypothesis: nothing happens
- alternative hypothesis: opposite of null
- many more
- measure units depend a lot on context

independent variable fits changed by scientist,
dependent variable depends from the independent variable.

Choosing the right **participants** is important:

- right user group?
- why do they represent the user group?
- random sample
- convenience sampling (pick those around you)
- purposive soapley: more specific than convenience sampling
- quotes, stratification, snowball,... sampling




**Descriptive statistics**: e. g. mean, median, mode, standard deviation, range, difference between lowest & highest value,....

**measuring data**: nominal (numeric), ordinal (positional), interval (distance relevance), ratio (scalable)

**Inferential statistics**: figure out things to explore, compare groups with multiple trajectories, regression, probability, go above data we have

# 9. LATE STAGE EVALUATION

We evaluate early, often, iteratively; late stage evaluation is when getting closer to the result & requirements & expectations

- important:
  - reflection to improve
  - systematically reason with science
  - keeprovements are the target of interaction design
  - evaluate different things/states (how you feel/ achieve something/limited, think about positive and negative aspects
- rigour:
  - evidence
  - visualize
  - use different teechiques
- process: many tools are available for evaluation,
  important to reflect on complete versions and compare them
- value
- a good place to start are the requirements
- prepare questions & answers
- reflect the procedure to be able to improve
- at this stage there is less room for changes

**A/B testing:**
- controlled experiment with 2 versions
- evidence based reasoning
- good for small granular things

**Think aloud:** a user talks about their process (what they want to do)
- complex → so it is challenging (e.g. code & talk)
- tell us what people say and do
- record what someone does, then talk about it

**Cognitive walkthroughs**: Examine learn ability and memorability without user testing. the evaluator assumes or emulates the role of the user.
- start with something to evaluate, think carefully
- resources: analytics, personas, involve stakeholders, reflect on best practices, different perspectives
- focus: logical set of tasks, sequential and take notes
- look for: matches between expectations and reality
- evaluate: standardise your feedback, reflect

**heuristic evaluations:** A guiding set of principles for design that is evaluated against an

actual design

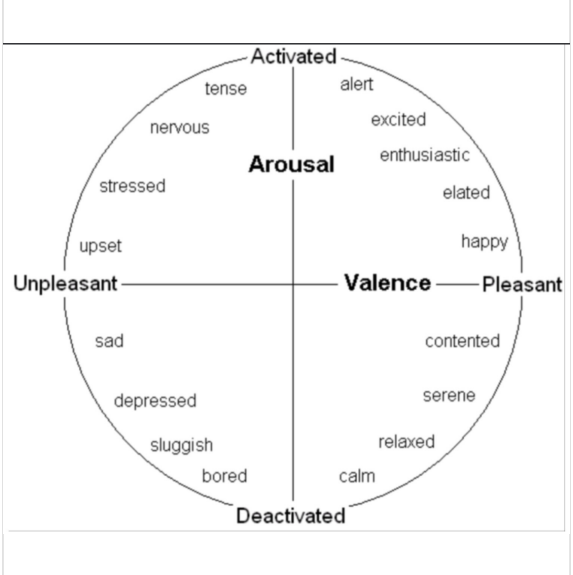
- Jakob Nielson
- Ben Shneidrman
- Donald Norman

All have their advantages & disadvantages-keep that in mind

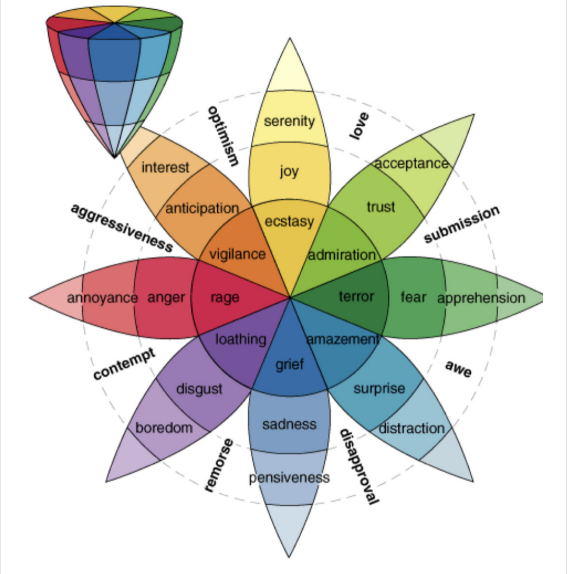
# 10. AFFECTIVE COMPUTING AND SOCIAL INTERACTON AND THE FUTURE OF INTERACTION DESIGN

**Affective computing** is about how computer and emotions relate to each other. This involves measuring emotions. These can be expressed with words or body language or a tone in the voice.

**valence arousal diagram**.




**Plutchicks wheel of emotions**




both images from research gate

user emotion shall be detected and captured, then be fed back into the system. E. g. fear

can lead to fight or flight which should be considered in interaction design. Depending on context different options should be available.

Emotional intelligence is very important for a successful interaction. Here artificial intelligence comes in.

**social computing:**

- involves conversational aspects & communities
- different ways of communication are possible (text, video, bot chat,...)
- technology can separate & bring together → CMC Computer Mediated Communication
    - subfield of social computing
    - focus on how humans communicate
    - different modalities available
    - asynchronous (time is not important)
    - tone of a massage can be difficult to transmit
    - like: agree or enjoy?
    - new forms of expressions: abbreviations (LOL), emojis,...