

IDEO.md

<https://www.youtube.com/watch?v=izjhx17NuSE&t=19s>

- har inte bestämda titlar, arbetspositioner i teamet
- Går ut i världen och studerar användarna av shopping carts
- har vilda idéer, de kan bli använda/utvecklas till något, ger nya perspektiv
- Sättger up massa idéer på väggen
- gör många prototyper/mockups
 - tog prototyperna och mergade deras bästa aspekter
 -

ideation.md

från https://canvas.kth.se/courses/52114/pages/ideation-and-brainstorming?module_item_id=1023653 | canvas

En fas i human centered design

- make sense av data, generera designs, bygga rough prototypes, konsultera användare angående prototyper, etc

Metoder inom ideation:

Download your learnings

- när man har massa data/photos, impressions, och annan input angående designproblemet måste man maken sense av dem
- Anteckna idéerna man fått, sätt de på stort papper
- visa uppmärksamhet till kollegors findings
- gör det nära in på days in the field eller efter intervjuer, när har fråscht i huvudet

Share inspiring stories

- dela med av stories som ger insikt i problemet
- Bygger på collective consciousness
- bygg in de i narrativet av arbetet
- berätta mest insightful stories
- alla i teamet borde göra detta
- kan sätta upp post its med olika kategorier

Find themes

- efter downloadat learnings och share inspiring stories
- kolla om patterns emergerar från dessa, ex någon insight som återkom ofta
- om det ex finns consistent problems som de man designar för möter
- försöka sorta ut vad themesen betyder
- översätt key themes i opportunities for design

Create insight statements

- efter tre ovanstående
- konkretisera key themes till key insight-menningar som visar vägen fram
- dessa meningar ska inte vara lösningar på problemen, snarare building blocks, som kan hjälpa i "How might we" eller "Brainstorms"
- Discarda insight statements som inte relaterar till originella designproblemet

How might we

- Reframa insight statements till frågor "how might we..", tillåter opportunities för design
- bra eftersom de föreslår att lösningar är möjliga och tillåter en variety av svar

- broadena frågan om vareityn av möjliga svar är för liten
- ska va narrow nog att visa var brainstorm kan svara, men bred nog att tillåta vilda idéer

Brainstorm + Brainstorm rules

- gör de med design team men också de man designar för samt partners
- promote openness, ideas, creativity över feasability
- bätt om alla är positiva öppna
- kan ha stor tavla med post its igen
- RULES
 - Generellt, måste vara öppen, och inte kritisera idéer för att inte stifla en generative mindset
 - 7 regler:
 - 1. Defer judgement
 - 2. Encourage wild ideas
 - Kan göra att man får insikt i vad man vill att teknologin ska göra utan att tänka på constraints av feasability
 - 3. build on the ideas of others
 - 4. stay focused on the target
 - 5. One conversation at a time
 - 6. Be visual
 - Sketcher o post its på väggen förmedlar idéer väl
 - 7. Go for quantity

Data gathering for requirements.md

#InteractionDesignKap11

för att ta reda på vilka intended users är, vad deras aktiviteter är, vad deras mål och problem är, kontekten av handlingen

- mål av data gathering session är hitta alla requirements för produkten
 - vanligt att använda interviews, questionnaires eller observation (se [Data_gathering](#)) genom hela [process of interaction design](#) för detta
-

- kan samla requirements genom läsa documentation, activity logs, manuals
 - Kan också studera liknande produkter
-

- Viktigt att använda flera metoder för data gathering för att få olika perspektiv
 - Ex interviews för fokusera på specifika user groups
 - observation för att förstå kontext
 - questionnaires för att nå wider audience
 - Focu groups to build consensus view

probes är objekt som får users att berätta mer/utveckla sina resonemang, eller så att de besvara frågor i rätt context (se . 398)

contextual inquiry

- ex intervjuaren försöker sätta sig in i den intervjuades liv, lära sig från de (= apprenticeship model)
- Kollar och deltar när users gör sagda aktiviteter
- fokusera på anledningen bakom actions
- ***fyra principer av apprenticeship model:***
 - **Context principle**
 - gå till usern och se vad de gör när gör det, ger rikare insights
 - **Partnership principle**
 - Intervjuaren är inte i kontroll, interviewee och interviewer delar makten
 - **interpretation principle**
 - omvandla observation till något som kan vara basis för design hypothesis/idea

- gör tsm med den intervjuade
- **Focus principle**
 - måste paya attention till detaljer som kommer visa sig
- Skippade cool concepts s. 401

Requirements kan komma direkt från datan som samlas men också från innovation, genom [brainstorming for innovation](#)

brainstorming for innovation.md

#InteractionDesignKap11

Generisk teknik för att generera, refina, develop ideas

- oxå för alternativa eller bättre deisngs

För att en brainstorming session ska bli bra inom ramen för [requirements](#) activity:

- participants måste känna users
- kritisera/debattera inga idéer

Från robertson & robertson; Kelley with Littman

- 1. Include participants from a wide range of disciplines with a broad range of experience*
- 2. Don't ban silly stuff.* Wild ideas often turn into really useful requirements.
- 3. Use catalysts for further inspiration.* Build one idea on top of another, jump back an earlier idea, or consider alternative interpretations when energy levels start to flag. If you get stuck, use a word pulled randomly from a dictionary to prompt ideas related to the product.
- 4. Keep records.* Capture every idea, without censoring. One suggestion is to number ideas so that they can be referred to more easily at a later stage. Cover the walls and tables in paper, and encourage participants to sketch, mind-map, and diagram ideas, including keeping the flow of ideas, as spatial memory is very strong, and this can facilitate recall. Sticky notes, each with one idea, are useful for re-arranging and grouping ideas.

11.5 BRINGING REQUIREMENTS TO LIFE: PERSONAS AND SCENARIOS

1. *5. Sharpen the focus.* Start the brainstorm with a well-honed problem. This will get the brainstorm off to a good start, and it makes it easier to pull people back to the main topic if the session wanders.
2. *6. Use warm-up exercises and make the session fun.* The group will require warming up if they haven't worked together before, most of the group doesn't brainstorm regularly, or the group is distracted by other pressures. Warm-up exercises might take the form of word games or the exploration of physical items related or unrelated to the problem at hand, such as the TechBox in Chapter 2.

1.

personas and scenarios.md

#InteractionDesignKap11

Ett sätt som [user stories](#) kan enhancas och ge realistisk detalj, låta utvecklare utforska users current activiteies, framtida användning av produkter, och guida development

Personas

- Rika, realistiska, generalla descriptions av users och deras krav

- Grundad i [Data gathering for requirements](#)
- varje persona karakteriseras av ett antal unika goals
- Personas har också beskrivning user behaviour, attitudes, environment, mål, bild etc
- Personas har två mål:
 - 1. hjälpa designer ta design decisions
 - 2. påminna att riktiga personer använder produkten
- Bra persona tillåter frågor av typen "vad hade bill gjort i denna situation med produkten?"

Scenarios

- Beskriver human activities / tasks i story
- tillåter utforskning av context, needs, requirements
- naturligt för personer att berätta stories, stakeholders kan enkelt relatera
- Leder till att kan förstå vad personer gör och vilka mål de har, samt identifiera stakeholders, vilka som kan dra nytta av potentiell produkt
- kan ex beskriva current behaviour eller hur nya produkter kan användas
 - dessa kan omsättas i [requirements](#)
- Futuristic scenarios beskriver envisioned situation i framtiden

requirements.md

#InteractionDesignKap11

statements om vad produkt/system ska göra o hur den ska göra det

- discovera requirements går ut på utforska problem space, definiera vad som ska utvecklas
 - = i interactionsdesign hur människors liv kan göras mer convenient av produkter

eftersom [process of interaction design](#) iterativ nära linked process kommer evaluation, design, requirements vara nära relaterade

- Har dock olika mål och emfas

What is the purpose of the requirements activity?

- Requirements-aktiviteten utgör två första faser i [Double diamond](#) och handlar om att utforska problem space, få insights i problemet, samt skapa beskrivning av vad som ska utvecklas
- Requirements kan discovras från targeted activities eller under ex prototyping och designfaser
- Requirements-activity är iterativ så att nyfunna insikter matas tillbaka i sig själva (?)
- aktiviteten själv revisitas ofta

How to capture requirements once they are captured?

- Hur strukturerat och formellt requirements måste vara definierade beror på komplexiteten av appen
 - ex workout monitoring app kan prototyp implicitly fånga requirements
 - Om ex mer komplext factory management system måste mer rigorös notation användas
- Kan dock vara bra att fånga requirements explicit så inte tappar något genom interationerna
- använd lämplig notation för requirements baserat på produkten

Varför ska man samla requirements?

- Att definiera requirements gör så developers vet vad de ska bygga och att users kan hjälpa
- målet med iterativ [User centered design](#) med kontinuerlig evaluation och user involvement gör så att misskommunikation inte sker och så att produkten verkligen kommer uppfylla user requirements

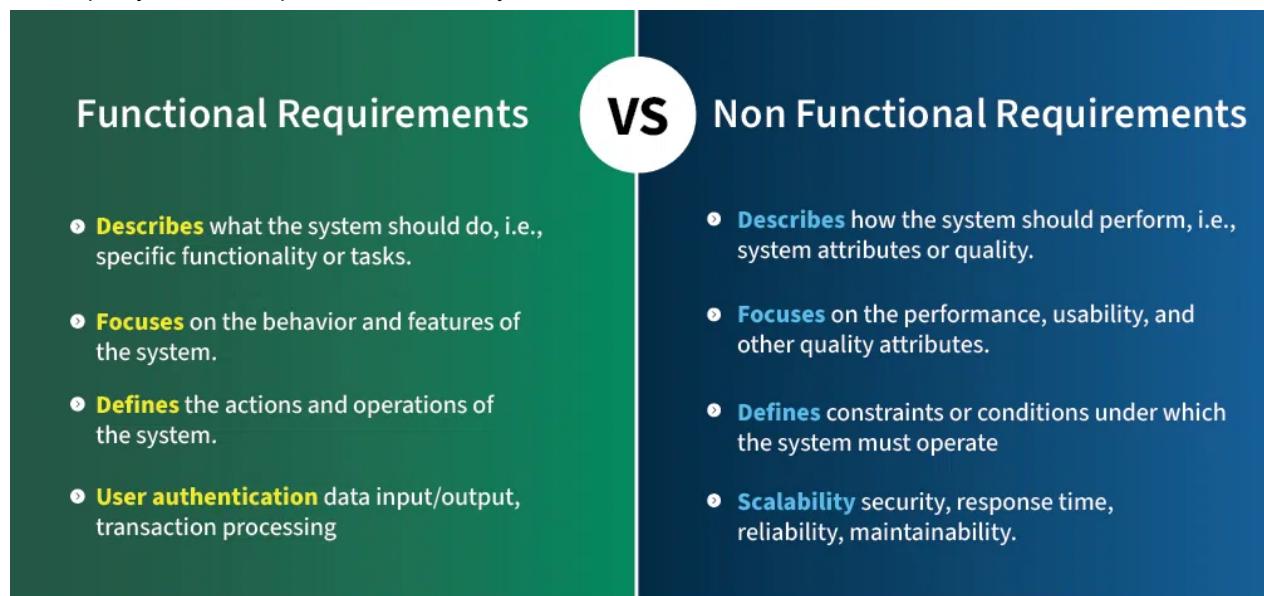
Vad är en requirement?

- statement om vad göra eller hur perform
 - om vaga requirements handlar requirements activity om att göra de tydligare, bryta ner i flera mål
 - requirements activity mål: identify, clarify, capture requirements.
 - Iteratikt

- kan också specificera kriterier som visar att målen nåtts
- **Hur skriva ner/fånga mål?**
 - Kan ha en requirements shell, som en mall man fyller i
 - **User stories**
 - kan vara meningar/beskrivningar av tänkta användare och vilket värde de vill få genom produkten
 - kan göra så att developers/stakeholders kommer koversera på ämnet
 - As a <traveler>, I want <to save my favorite airline for all my flights> so that <I will be able to collect air miles>.
 - As a <travel agent>, I want <my special discount rates to be displayed to me> so that <I can offer my clients competitive rates>.
 - vanligt i agile, ofta bas för en sprint
 - Mål, tid estimat, tester för att se att det funkar
 - user epics: väldigt stora user stories

Different kinds of requirements

- reqs Kan komma från ex user community eller business community
- *Finns ett antal typer av reqs, varav de två översta är vanligast:*
 1. **Functional requirements**
 - vad produkten ska göra
 - basic facilities som måste finnas
 2. **Nonfunctional requirements**
 - characteristics av produkten
 - ex quality constraints, performance, usability



Project Drivers	1. The Purpose of the Product 2. The Stakeholders
Project Constraints	3. Mandated Constraints 4. Naming Conventions and Terminology 5. Relevant Facts and Assumptions
Functional Requirements	6. The Scope of the Work 7. Business Data Model and Data Dictionary 8. The Scope of the Product 9. Functional Requirements

11.3 WHAT ARE REQUIREMENTS?

Nonfunctional Requirements	10. Look and Feel Requirements 11. Usability and Humanity Requirements 12. Performance Requirements 13. Operational and Environmental Requirements 14. Maintainability and Support Requirements 15. Security Requirements 16. Cultural Requirements 17. Compliance Requirements
Project Issues	18. Open Issues 19. Off-the-Shelf Solutions 20. New Problems 21. Tasks 22. Migration to the New Product 23. Risks 24. Costs 25. User Documentation and Training 26. Waiting Room 27. Ideas for Solutions

1. Data requirements

- 1. Typ, volatilitet, persistanec, size och krav på att handla data

2. Environmental requirements

- 1. Handlar om i vilken environment produkten ska operera
 - 1. finns ett antal aspekter av denna:
 - 1. physical environment
 - 1. ex använd inte speech interface för public ATM, mycket snack i bg
 - 2. Social environment
 - 1. hur samarbete inom produkten ska ske
 - 3. Organizational environment
 - 1. Ex finns träning tillgänglig, communication lättillgänglig?
 - 4. Technical environment

1. vad för technology används produkten/ kommer användas i conjunction med?

3. user, usability, and user experience requirements

1. diskuteras i kap 2
2. Kan använda usability engineering, välja meusers för hur mäta kvalité tidigt i processen

Olika typer av produkter kommer alltid ha olika requirements

Se [Data gathering for requirements](#)

use cases.md

#InteractionDesignKap11

dessa fokuserar på functional [requirements](#) och capturar interaction

- eftersom fokus interaction gör de så man måste tänka på interaktionen som designas, och kan också fånga [requirements](#) (genom att tänka på vad användaren måste kunna se, veta, göra)
- Är en step by step description, definierar specefic process
- kan enchanca basic [requirements](#) statement genom att beskriva de som step by step processes i form av use cases
- exempel s. 416

prototyping.md

#InteractionDesignKap12

design, prototyping och konstruktion alla del av develop i [Double diamond](#)
prototyper facilitatar iterative design-evaluation-redesign cykler

finns två aspekter av design:

- Conceptual: handlar om idén om produkten, vad den kommer göra
- Concrete: detaljer om design, ex fysiska widgets och graphics
Dessa två behöver consideras för att kunna prototypa idéer, vilka evolvar konceptet

kvaliteen på prototyper ändras och utvecklas genom designprocessen

Prototyper ger manifestation av idé vilken tillåter designers kommunicera deras idéer och användare att prova de

- prototyper kan vara lite vad som helst

Why prototype

- Tillåter kommunikation, evaluation av idéer och deras feasability, tydliggörande av vaga krav
-
- Finns skillnad product och service prototypes
 - service prototype involverar role playing och personer (vilka är integral del av prototypen och produkten själv)

Low fidelity prototypes

- limited functionality, inte samma material som slutgilltiga produkt
- enkla, billiga att producera OCH ändra/iterera efter nyupptäcka krav
- bra tidigt i development, eftersom prototyper bör vara flexibla och uppmuntra exploration och modification
- en typ LOFI prototyp är storyboarding, en vidareutveckling av [scenarios](#)
 - kan använda [sketching](#) för att göra ännu tydligare
- **andra typer av LOFI prototypes**
 - Index cards: kort med element från produkten på sig
 - Wizard of oz: någon människa sitter och kontrollerar en digital prototype som användaren interagerar med

High fidelity prototypes

- Ser mer ut som slutliga produkt och har mer ffunktionalitet

- ofta ökar fidelity av prototypes genom design stages
- att kunna använda prototypes i real context (in situ) kan ge bra insights o feedback, fördel m HIFI prototypes

Kompromissar i prototyping

prototyping innebär alltid kompromissar, ex lofi så funkar sällan produkten som tänkt

- måste alltså fokusera på de frågor man vill ha svar på genom prototypen
- **Två typer av prototyping:**
 - Horizontal prototyping: Många features men lite detail
 - Vertical prototyping: mycket djup/detaljer för några få funktioner
- Vanligt tradeoff är robusthet vs förmåga att förändra prototypen, ofta negativt korrelerade
- kompromiss av hifi-prototyp kan vara att users mindre beredda att kritisera produkt de percievear som färdig
- Behöver alltid mer rigorös testning än prototyping tillåter innan shippas
-
- Två filosofier av prototyping:
 - Evolutionary prototyping: prototype evolvas till slutgiltig produkt
 - Throwaway prototyping: Prototype är stepping stones för final design

conceptual design

- conceptual model är view av vad kan göras med produkt, o koncept som behövs för att förstå produkten
 - det senare beror på vilka users kommer vara, vilken interaction som behövs, etc
- huvuddel i conceptual model är mapping mellan concept och user experience
- sätt att generera potential conceptual models:
 - **Interface metaphors**
 - kombinera familiar och ny knowledge
 - kan göra det med metaforer
 - **Interaction types**
 - vilken typ av interaction passar produkten bäst?
 - kommer kunna använda flera typer i en produkt
 - **Interface types**
 - Bra tänka olika interfaces eftersom de supportar olika interface types
 - kan prova olika, vilken som väljs (kan) bero på vilka [requirements](#) som finns

Expanding the initial conceptual model

- måste fråga vilka funktioner användaren respektive produkten är ansvariga för
 - måste balansera så får rätt nivå av cognitive load/frihet hos användaren
- hur är funktioner relaterade; detta kan indikera suitable restrictions/task structures hos produkten
- tänk på data [requirements](#) i conceptual model

Concrete design

- göra mer detaljerade val
- kan vara att deala med konkurerande [requirements](#)
- se till att usability and user experience goals möts, ex genom att välja interface type och design considerationsen som kommer med de; dessa är desicions inom concrete design
- accesability också viktigt

Generating prototypes

- STORYBOARDS s. 448
 - ta [scenario](#) och bryt ner till steps som fokar på interaction, en scen för varje step
 - storyboards kan fokusera både på själva produkten och environment
 -
- CARD-BASED PROTOTYPES
 - för att förstå users interaction, ex exchanges mellan syst och användaren

- Fördel att kan flytta kort för att simulera vad som kommer hända
- Kan kombineras med expirence wheel / experience map (eller andra representationer), se s. 453

Construction

- när prototyping och alternativa designs progressar kommer mer tid läggas på bygga ihop komponenter och utveckla final product
- kan ta fysisk/programvara form eller båda
- ***ovanligt att behöva utveckla allt fr scratch för supporta development, finns två resurstyper för hjälpa med detta***
 - **Physical computing kits**
 - bygga prototyper med fyska material, programering, chip, elektronik etc
 - ex arduino
 - **SDKs**
 - Package av programming tools som supportar development av applikationer
 - ofta IDE ingår

Summary

This chapter explored the activities of design, prototyping, and construction. Prototyping and scenarios are used throughout the design process to test ideas for feasibility and user acceptance. We have looked at different forms of prototyping, and the activities have encouraged you to think about and apply prototyping techniques in the design process.

Key points

- Prototyping may be low fidelity (such as paper-based) or high fidelity (such as software-based).
- High-fidelity prototypes may be vertical or horizontal.
- Low-fidelity prototypes are quick and easy to produce and modify, and they are used in the early stages of design.

FURTHER READING

- Ready-made software and hardware components support the creation of prototypes.
- There are two aspects to the design activity: conceptual design and concrete design.
- Conceptual design develops an outline of what people can do with a product and what concepts are needed to understand how to interact with it, while concrete design specifies the details of the design such as layout and navigation.
- We have explored three approaches to help you develop an initial conceptual model: interface metaphors, interaction styles, and interface styles.
- An initial conceptual model may be expanded by considering which functions the product will perform (and which the user will perform), how those functions are related, and what information is required to support them.
- Scenarios and prototypes can be used effectively in design to explore ideas.
- Physical computing kits and software development kits facilitate the transition from design to construction.

Evaluation case studies.md

finns på s. 507

Computer game

- roligare att spela vs dator eller vän?
- spel måste vara engaging och challenging
 - -> Behöver evaluera dessa aspekter och behöver kriterier för att evaluera det
- mätte physiologiska faktorer, normaliseringade dessa mellan subjects, och analyserade resultat
- Tog också satisfaction questionnaires
- visar att kan gå beyond vanlig usability testing i experiment för att få djupare förståelse [User experience](#)

en till omkring s. 513

massa begrepp om evaluation på s. 514

BOX 14.3

The Language of Evaluation

Sometimes terms describing evaluation are used interchangeably and have different meanings. To avoid this confusion, we define some of these terms here in alphabetical order. (You may find that other books use different terms.)

Analytics Data analytics refers to examining large volumes of raw data with the purpose of drawing inferences about a situation or a design. Web analytics is commonly used to measure website traffic through analyzing users' click data.

Analytical evaluation This type of evaluation models and predicts user behavior. This term has been used to refer to heuristic evaluation, walk-throughs, modeling, and analytics.

Bias The results of an evaluation are distorted. This can happen for several reasons. For example, selecting a population of users who have already had experience with the new system and describing their performance as if they were new users.

Controlled experiment This is a study that is conducted to test hypotheses about some aspect of an interface or other dimension. Aspects that are controlled typically include the task that participants are asked to perform, the amount of time available to complete the tasks, and the environment in which the evaluation study occurs.

Crowdsourcing This can be done in person (as was typical in citizen science for decades) or online via the web and mobile apps. Crowdsourcing provides the opportunity for hundreds, thousands, or even millions of people to evaluate a product or take part in an experiment. The crowd may be asked to perform a particular evaluation task using a new product or to rate or comment on the product.

Ecological validity This is a particular kind of validity that concerns how the environment in which an evaluation is conducted influences or even distorts the results.

Expert review or crit This is an evaluation method in which someone (or several people) with usability expertise and knowledge of the user population reviews a product looking for potential problems.

Field study This type of evaluation study is done in a natural environment such as in a person's home or in a work or leisure place.

Formative evaluation This type of evaluation is done during design to check that the product fulfills requirements and continues to meet users' needs.

Heuristic evaluation This is an evaluation method in which knowledge of typical users is applied, often guided by heuristics, to identify usability problems.

Informed consent form This form describes what a participant in an evaluation study will be asked to do, what will happen to the data collected about them, and their rights while involved in the study.

In-the-wild study This is a type of field study in which users are observed using products or prototypes within their everyday context.

Living lab This place is configured to measure and record people's everyday activities in a natural setting, such as in the home.

Predictive evaluation This type of evaluation is where theoretically based models are used to predict user performance.

Reliability The reliability or consistency of a method is how well it produces the same results on separate occasions under the same circumstances.

Scope This refers to how much the findings from an evaluation can be generalized.

Summative evaluation This evaluation is done when the design is complete.

Usability lab This lab is specially designed for usability testing.

Usability testing This involves measuring how well a design supports users' performance on various tasks.

User studies This generic term covers a range of evaluations involving users, including field studies and experiments.

Users or participants In this context, these terms are used interchangeably to refer to the people who take part in evaluation studies.

Validity Validity is concerned with whether the evaluation method measures what it is intended to measure. ■

• **Informing participants about their rights**

- Behöver ofta berätta för institution vad för data man ska samla in och hur, skydda participants
- finns lagar som skyddar de, ex gdpr
- kan behöva anonymisera data
- kan finnas boilerplate templateS för beskriva hur participants treated
- vanligt med ndas för participants, pga inte vill konkurrenter ska få reda på

• **Making sure you take into account biases and other influences that impact how you describe your evaluation findings**

- s. 517 frammåt
- Se [reliabilitet och validitet](#)
- ecological validity: hur representativ miljön [Evaluation](#) genomförs i är av vanliga användarmiljän
 - Låg i lab settings (controlled settingS)
 - Hawthorne effect påverkar ecological validity: kunskap om att blir studerad påverkar en
- Bias: ex kan interviewers subconsciously påverka users svar med sin tone of voice, anisktsuttryck
- Scope: vissa undersökningar har för litet scope för att kunna generaliseras, eller liknande
 - Var försiktig med vilka slutsatser man drar!

Evaluation.md

Måste evaluera om ens utvecklade app/whatever actually är appealing till target audiences

- falacy att utvecklarna tror att det som appealar till de appealar till andra
- evaluation kan hända ex hemma, i labb, på workplace, utomhus
- utvärdera och observera användares performance med produkten
- Kan ist för involvera users direkt modella user behaviour för få approximation av det
- Analytics kan assesa performance av befintlig produkt, så den kan förbättras

Why evaluate?

- utöver fungera ska produkt vara easy och pleasing to use, simpel, elegant
- Bättre design / upplevelse säljer bättre
- kan fixa problem innan säljer produkt

What to evaluate

- designers vill veta olika saker baserat på produkten i fråga
 - främst om den gör det den är avsedd att göra
 - ex om designar browser: hittar saker snabbare?
- vad som evalueras beror på om det skapar värde för designers och användare
- Huvudsakliga kriteriet som är viktigt är om användarna gillar produkten och kommer använda den

Where to evaluate

- beror på vad som evalueras
- om systematiskt vill prova om alla requirements möt är labb vanligt
- design choices kan utvärderas i labb
- [User experience](#) utvärderas ofta bättre i naturlig setting, kallas *in-the-wild studies*
- kan remortly evaluta ex om man testar socialt nätverk, kan bli det mest naturliga

When to evaluate

- beror ex på om produkt ny eller uppgradering
- om ny ofta mycket evaluation
 - testa om requirements av users blev interpreted av designers correctly genom evaluation, ändra according till resultat
- *formative evaluations* är när man kollar om produkt fortsätter möta user needs
- *summative evaluations* är när för kolla success av finished product
 - ex om ska uppgradera och vill veta vad som behöver improvas

Se [Types of Evaluation](#)

Types of Evaluation.md

#InteractionDesignKap14

Sätt att genomföra [Evaluation](#)

Tre typer baserat på setting, user involvement, level of control:

1. Controlled settings directly involving users
 1. ex labb för observera beteende
 2. huvudmetoder usability testing, experiments
2. Natural settings involving users
 1. lite kontroll över användares aktiviteter, vill se hur skulle användas i riktiga världen
 2. main method = field studies
3. Any settings not directly involving users
 1. experter/researches modellerar/predictar most obvious usability problems
 2. ex inspections, heuristics, walk-throughs, models, analytics

Finns för och nackdelar med alla typer

- lab-based kan visa usability problems enkelt men missa context of use
- field studies kan visa hur teknologi används i intended setting men kan vara svåra att genomföra
- modeling kan vara snabba men missa unpredictable/subtle aspekter av [User experience](#)

vad som används beror på produkt, vad man vill testa, resurser tillgängliga

Controlled settings involving users

- designade för kontrollera vad och hur users gör saker med produkten
Vanlig approach:

Usability testing

- Usability testing involverar datainsamling i kontrollerad setting
 - datainsamling genom observationer, intervju, questionnaires
 - Huvudsakligen mål: **Determine om intended users kan använda interfacen för att göra intended tasks**
 - Ta bort externa faktorer, testa specifikt några funktioner för att endast se **deras** påverkan
 - Logga ofta actions med video eller logger
 - questionnaires för samla info om upplevelsen
 - all kvantitativ/kvalitativ data används i conjunction för att avgöra hur väl usability mål nås
 - Findings summariseras ofta till *usability specification* som framtida prototyper kan testas mot, notera nivåer av performance. nya ändringar kan implementeras och trackas (om det blev bättre/sämre ????)
 - exempel: kan testa text input med olika devices för att avgöra om något signifikant bättre än någon annan

Natural settings involving users

- huvudmål evaluera produkter med users i natural setting
HUVUDSAKLIG METOD: FIELD STUDIES
- huvudmål:
 - Help identify opportunities for new technology
 - Establish the requirements for a new design
 - Facilitate the introduction of technology or inform deployment of existing technology in new contexts
- vanliga metoder: observation, interviews, interaction logging
- ska försöka vara unobtrusive, men svårt att vara det helt
- in-the-wild studies: hur nya produkter har blivit deployade och används
 - tappar lite kontroll över vad som observeras, mäts, när är in the wild
 - kan ex behöva rekrytera några användare, säga till de att använda den men inte ge mer instruktioner
- Nackdel: Svårt anticipata om/när något kommer hänta, kan bli svårare att vara där och kontrollera/observera jämfört med i *usability testing*

- kan göra virtual field studies, ex i spel

Any Settings Not Involving Users

- när researcher själv måste modelera och anticipata ex problem utofrån knowledge av usability, behaviour och context of use
- Exempel på metoder:
 - *Heuristic evaluation:* Knowledge av typical users, rules of thumb, walkthroughs som steppar genom scenario eller answers a set of questions
 - problem kan vara att heuristics inte är så accurate som de verklade vara
 - *cognitive walk-throughs:* simulera users problem-solving process med applikationen. fokus evalvera designs ease of learning
 - analytics kan också användas
 - samla in data från kunden
- metoder används primarily för att jämföra effectiveness av olika interfaces för samma application

Selecting and combinig methods

- olika metoder används ofta i kombination för att få djupare insikt
- fördelar o nackdelar finns med alla metoder:
 - i controlled kan kontrollera hypotheses och generalisera
 - i uncontrolled kan få unexpected insights som är bra

Opportunistic evaluations/explorations

- = generally när man tidigt i designprocess provide designers med feedback
- bra för man kan veta om värt fortsätta

Conducting experiments.md

#InteractionDesignKap15

Testar ofta specifika hypoteser om hur users använder en interface för att rigoröst kunna dra slutsatser om att en interface kommer vara bättre än en annan

- hypoteser ofta baserade på teorier eller tidigare findings

Hypotheses testing

- Undersöker samband mellan dependent och independent variables (vilka varieras av användaren)
- nollhypotes och alternativ hypotes
 - Two-tailed: Tvåsidigt begärsad nollhypotes, föredragen över ensidig om ingen anledning att tro att det är mer sannolikt med en skillnad åt ena hålet än andra
- Måste hålla variabler man inte vill ändra konstanta för att inte påverka ens findings
 - detta kallas the *experimental design*
 - många variabler gör det svårare att avgöra vad som orsakar resultat

Experimental design

- måste veta vilka participants att involvera för vilka conditions i ett experiment
- exposure till ett test gör att man reagerar annorlunda på ett annat till test (de blir tränade)
 - Vissa experimental designs kan dock använda alla participants för alla conditions utan att träningseffekter biasar resultat
- Det finns olika typer av designs:
 - **Different-participant design**
 - randomly allokerar alla participants till olika conditions
 - alla gör bara testerna under ett sätt av conditions -> inga träningseffekter
 - disadvantage: behöver många participants för att motverka slumpeffekter av att några participants sticker ut, ex redan har expertise
 - Kan slumpa, eller förtesta participants för att minska detta
 - **same-participant design**
 - Alla participants gör test under alla conditions

- kräver färre participants
- måste se till att välja ordningen av tester för att motverka träningseffekter
- ex inte alla gör A sen B, hälften hälften bättre (kallas *counterbalancing*)
- ***matched-pairs design***
 - grupperar subjects som har liknande characteristics i par, sedan randomly allocated till varje experimental condition
 - kan mitigera order effects och individual differences
 - kan dock finnas variables som inte considerats som kan influensa results; kan aldrig vara säker att det inte finns avgörande skillnader mellan par
- Gör tester på dessa sätt, samla in data, sen ex göra t-test för se om signifikant

field studies.md

#InteractionDesignKap15
mer o mer [Evaluation](#) studies gör i natural settings med lite kontroll över participants

- eftersom mer teknologi används utanför jobbet o i vardagen/hela tiden idag

stor skillnad mellan hur produkter används i labb och hur de kommer användas i riktiga världen

- Mer distractions etc, mindre order i naturlig setting
- att göra field studies kommer ge mer representativa resultat för om produkten är bra/användarvänlig i kontexten den kommer användas i
- nackdel: svårare än ex i [usability testing](#) att undersöka hypoteser eftersom finns många fler okontrollerbara faktorer som påverkar användandet
 - får använda mer kvalitativa mått för att bedömma hur users reagerar på en interface
- kan samla in data genom video, bilder etc, eller questionnaires
 - Kan interrupta under tiden de gör en action och kolla dess performance (?)

concerns angående berätta subjects att de är med i study

- Kunskap om är med i study påverkar beteende
- måste ensura privacy

findings från field studies oft apresenterade som vignettes excerpts, critical incidents, behaviour patterns, narratives

- för vis hur produkt används och adoptas *in i natural surroundings*

speciell typ field studies för testa ny teknologi: [in the wild studies](#)

vidare

- kan vara bra med field studies eftersom users kan få familiariza sig över lång tid i sin setting naturliga setting med teknologin
- om vill testa hur expert interagerar med komplicerad applikation över lång tid är det svårt att göra i controlled lab environment, bättre med field study

Andra mål med field studies:

- 1) Help identify opportunities for new technology
- 2) Establish Requirements
- 3) Inform the deployment of existing technologies

in the wild studies.md

#InteractionDesignKap15

vanligt använda dessa för kolla hur folk interagerar med nya teknologier eller prototyper in situ

Dessa utforskar ofta nya teknologier i natural settings istället för att developa solutions som passar in med existerande practices

- vill kolla hur users interagerar med och integrerar nya teknologin i deras liv
- ofta lättare att se hur teknologin faktiskt kommer användas en i controlled settings, ex [usability testing](#)

case study s. 538

usability testing.md

#InteractionDesignKap15

vanligt att testa usability av application, desktop program i controlled environment (ex labb) där man har kontroll över vad användare ska göra och kan kontrollera environmental/social aspekter som kanske annars hade påverkat användaren

- **Huvudmål att upptäcka om intended users kan använda produkten för att utföra dess intended use**
- också ibland om den är enjoyable

Methods, tasks and users

- Måste collecta data för att göra usability testing och kunna dra slutsatser
- Delta kan vara logging, video recording, questionnaires, intervjuer
- Vanliga taska kan vara söka information, läsa olika typefaces
- performance measures är ofta performance typer och antal interactions
- **Performance measures som är baseline för user performance data:**
 - Number of users completing a task successfully
 - Time to complete a task
 - Time to complete a task after a specified time away from the product
 - Number and type of errors per task
 - Number of errors per unit of time
 - Number of navigations to online help or manuals
 - Number of users making a particular error
- Hur många test subjects ska vi ha?
 - kan vara bra med många för blir mer representativ av målpopulationen, större reliabilitet
 - om bara vill få snabb feedback för tidiga val i designprocessen kan färre fungera

settings

- Kan ha custom usability labb med mätutrustning och waiting area, ex google gör det
- Ibland får de att se ut som miljön produkterna ska användas i, ex office area
- Skärmar vanligen bort yttrare distractions för att bara testa usabiliteten
 - Tänk dock på att dessa finns i verkligheten; där produkten ska användas
- har ofta mikrofoner och kameror och annan utrustning för göra observationer
- Kan också göra *remote testing* med många participants; fördelar natural setting och data recordas och kompileras automatiskt
 - Convenient, mer participants

case study om ipad s. 528

analytics and A/B testing.md

#InteractionDesignKap16

Automatisk datainsamling för [Evaluation](#) kan vara bra för att unobtrusive
kan dock inte ignorera ethics av datainsamling
kan analysera ex visuellt och på liknande vis

Web analytics

- på websida kan tracka varifrån folk kommer, vilka sidor de visar, hur länge de stannar
- Kan ge big picture overview av [Interaction design](#) av en hemsida och UX problem som måstes fixas
- Eftersom strävar optimisera users usage av hemsida är detta valued av marketers, businesses
 - kan ex utvärdera marketing campaigns effectiveness
- I inte transnational sammanhang, ex utbildning och entertainment, kan detta även användas
 - learning analytics i utbildning, ex analysera varför folk droppar ut ur courses

Using web analytics

- två typer: On och off site analytics
 - on site är visitor behaviour på sidan
 - Off site är hur synlig sidan är
- Har blurrats ihop över åren
- kan använda komplementerande data, ex email, sales och history data

- Google analytics är bra och populärt för on-site analytics

A/B Testing

- annat sätt att utvärdera applikation/websida/etc
 - Har två stora grupper, en som får befintlig design (kontroll) och en som får en ny experimentell design
- Är ett controlled experiment
- har ofta 100s/1000s of participants
- Skillnad från det diskuterat i Kap 15 genom detta större i skala och vanligtvis gjort online
- **FÖR GÖRA A/B:**
 - Identifiera variable of interest, ex designa av advertisement
 - Ge grupp A existerand design, grupp B experimentell design
 - Mät några dependent measures, ex hur många gånger advertisementen klickades på
 - Räkna ut konfidensintervall att det finns skillnad?
- Online controlled experiments används idag till stor grad för att göra data driven decisions
- Kan först göra ett A/A test för att verifiera att de två delpopulationerna är lika, viktigt eftersom internet kan vara komplext
- Måste vara väldigt careful så att det man testar endast påverkar de independent variablerna man bryr sig om

heuristic evaluation.md

#InteractionDesignKap16

!!!! MÅSTE LÄSA <https://www.nngroup.com/articles/ten-usability-heuristics/>

Metod för Evaluation som inte involverar users

Denna går ut på att testa om en interface och dess element möter ett antal tested and tried principer (*heuristics*) för [Interaction design](#)

- dessa heuristics är goda generella designprinciper
- Revised version av original heuristics:

Visibility of System Status The system should always keep users informed about what is going on, through appropriate feedback and within reasonable time.

Match Between System and the Real World The system should speak the users' language, with words, phrases, and concepts familiar to the user, rather than system-oriented terms. It should follow real-world conventions, making information appear in a natural and logical order.

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 dialog. The system should support undo and redo.

16.2 INSPECTIONS: HEURISTIC EVALUATION AND WALK-THROUGHS

5:

Consistency and Standards Users should not have to wonder whether different words, situations, or actions mean the same thing. The system should follow platform conventions.

Error Prevention Rather than just good error messages, the system should incorporate careful design that 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.

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 dialog to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

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.

Aesthetic and Minimalist Design Dialogs should not contain information that is irrelevant or rarely needed. Every extra unit of information in a dialog competes with the relevant units of information and diminishes their relative visibility.

Help Users Recognize, Diagnose, and Recover from Errors Error messages should be expressed in plain language (not codes), precisely indicate the problem, and constructively suggest a solution.

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.

•

Om ska prova en del av interfacen:

1. Identifiera de heuristics som är relevanta för elementen
2. gå igenom elementen flera gånger och se om de följer heuristicsen
3. Åtgärda usability problems som hittas
4. (gör iterativt)

Concerns för heuristic evaluation

- De ovanstående heuristics kan vara för generella för vissa produkter, kan utveckla sina egna if
- Vanligt med 5-10 heuristics per produkt för att inte vara för svårt utföra, inte för lite information/testning av produkten
- Ungefär 3-5 researchers kan hitta upp t 75% av designproblem, att använda fler kan bli resource innefficient

Några heuristics för web design: De generella fungerade inte för webben som konstant utvecklades

Clarity

Make the system as clear, concise, and meaningful as possible for the intended audience.

- Write clear, concise copy.
- Only use technical language for a technical audience.
- Write clear and meaningful labels.
- Use meaningful icons.

Minimize Unnecessary Complexity and Cognitive Load

Make the system as simple as possible for users to accomplish their tasks.

- Remove unnecessary functionality, process steps, and visual clutter.
- Use progressive disclosure to hide advanced features.
- Break down complicated processes into multiple steps.
- Prioritize using size, shape, color, alignment, and proximity.

Provide Users with Context

Interfaces should provide users with a sense of context in time and space.

- Provide a clear site name and purpose.
- Highlight the current section in the navigation.

16 EVALUATION: INSPECTIONS, ANALYTICS, AND MODELS

- Provide a breadcrumb trail (that is, show where the user has been in a website).
- Use appropriate feedback messages.
- Show the number of steps in a process.
- Reduce perception of latency by providing visual cues (for instance, a progress indicator) or by allowing users to complete other tasks while waiting.

Promote a Pleasurable and Positive User Experience

The user should be treated with respect, and the design should be aesthetically pleasing and promote a pleasurable and rewarding experience.

- Create a pleasurable and attractive design.
- Provide easily attainable goals.
- Provide rewards for usage and progression. ■

Doing heuristic evaluations

Kan brytas ned i tre main stages:

1. **A briefing session:** Researchers briefas om mål med evaluation

2. **the evaluation period:** Researches independently undersöker produkten o använder heuristics för guidance

1. Ofta två passes, en för generell feel av produkt, en för att utvärdera alla heuristics
2. Om evaluatör färdig produkt kan man kanske föreslå tasks som ska göras, i vissa fall kan researches föreslå egna tasls

- 3. Måste adapta evaluation till circumstances

3. Debriefing session:

- 1. Alla kommer tillsammans, visar findings, prioriterar problem och föreslår lösningar

Vilka heuristics man använder bestämmer var researchers fokus kommer ligga -> måste välja de noggrant utifrån vad man vill evaluta

Ibland viktigt att tänka på availability issues

- Kan skapa heuristics specifikt för availability, disabilities

Turning Design Guidelines, Principles, and Golden Rules into Heuristics

- Ibland används guidelines direkt som heuristics
- Mer principledd att omvandla principles till frågor
 - Ex *hedonic heuristics*, heuristics (i form av frågor) som undersöker om users njöt av sin tid med produkten (ex "Is the service a lively place where it is enjoyable to spend time?")
 - Toni Granollers föreslår att omvandla heuristics till principer till frågor så att de blir användbara och kan användas för [Evaluation](#)

Ben Shneiderman föreslog design guidelines som ofta används som heuristics:

1. Strive for consistency.
2. Seek universal usability.
3. Offer informative feedback.
4. Design dialogs to yield closure.
5. Prevent errors.
6. Permit easy reversal of actions.
7. Keep users in control.
8. Reduce short-term memory load

inspections.md

#InteractionDesignKap16

Om inte kan involvera users pga otillgängliga, inte lagom tid och pengar, kan experter behöva ge feedback istället

- Istället får dessa, baserat på sin kunskap om [Interaction design](#) och beteende hos användare för att ge feedback
- Kan använda metoder som [heuristic evaluation](#) och [walkthroughs](#), involverar roleplaya user och gå igenom ex en interface, identifiera potentiella problem
- Kan göras genom alla steg i designprojekt

predictive models.md

#InteractionDesignKap16

en metod för [Evaluation](#) och [inspections](#) där varken users eller user researchers måste vara present

- Denna använder predictive modeling och formulars för att mäta user performance
- ger estimates till efficiency av olika val för olika tasks

Fitts' law

- Formel som beskriver hur lång tid det tar att flytta en pointer till en knapp på skärmen
- Kan användas för att optimera layouts på physical/digital buttons
- anger att $T = k \cdot \log_2(D/S + 1.0)$
- Större och närmare => hittar snabbare
- Kan använda i digitala och fysiska sammanhang
-

walkthroughs.md

#InteractionDesignKap16

Ett sätt att genomföra [Evaluation](#) för ex usability/nöjdhet av en produkt

Involverar inte direkt användare

är ett alternativ till [heuristic evaluation](#)

- Handlar om att steg för steg gå igenom process av att göra något och identifiera usability problems

Cognitive walkthroughs

- simulerar hur users går igenom problem solving process i vare steg av human-computer interaction
- **Fokuserar främst på att evaluate om designs har bra ease of learning**
 - underbyggt av observationer att man lär sig genom exploration
- **Huvudsteg i cognitive walkthroughs:**

Cognitive Walk-Throughs

Cognitive walk-throughs involve simulating how users go about problem-solving at each step in a human-computer interaction. A cognitive walk-through, as the name implies, takes a cognitive perspective in which the focus is on evaluating designs for ease of learning—a focus that is motivated by observations that users learn by exploration. This well-established method (Wharton et al., 1994) is now often integrated with a range of other evaluation and design processes. See, for example, the Jared Spool blog at <https://medium.com/@jmspool>, (Spool 2018).

The main steps involved in cognitive walk-throughs are as follows:

1. The characteristics of typical users are identified and documented, and sample tasks are developed that focus on the aspects of the design to be evaluated. A description, mock-up, or prototype of the interface to be developed is also produced, along with a clear sequence of the actions needed for the users to complete the task.
2. A designer and one or more UX researchers come together to do the analysis.
3. The UX researchers walk through the action sequences for each task, placing it within the context of a typical scenario. As they do this, they try to answer the following questions:
 - a. Will the correct action be sufficiently evident to the user?
(Will the user know what to do to achieve the task?)
 - b. Will the user notice that the correct action is available?
(Can users see the button or menu item that they should use for the next action? Is it apparent when it is needed?)

16 EVALUATION: INSPECTIONS, ANALYTICS, AND MODELS

- c. Will the user associate and interpret the response from the action correctly?
(Will users know from the feedback that they have made a correct or incorrect choice of action?)
In other words, will users know what to do, see how to do it, and understand from feedback whether the action was completed correctly or not?
4. As the walk-through is being done, a record of critical information is compiled.
 - a. The assumptions about what would cause problems and why are identified.
 - b. Notes about side issues and design changes are made.
 - c. A summary of the results is compiled.
5. The design is then revised to fix the problems presented. Before making the fix, insights derived from the walk-through are often checked by testing them with real users.

- Bra att tydligt dokumentera processen, kan ex ha standardiserade mallar med frågor
 - Notera extra tydligt negativa svar
 - Dokumentera information om produkten (dess ID etc), datum, severity av problem

[walkthroughs](#) fokuserar mer på att hitta specifika usabilityproblem än [heuristic evaluation](#), som mer kollar om generellt goda tumregler följs (?)

- Ibland kan designers bli defensiva, diskussion ta lång tid, allt detta kan kosta och skada social relationships
 - Rick Spencer föreslår att ska ha lägre antal frågor, vilket tar kortare tid, identifiera ledare och banna defending designs och komma på designs on the fly
- Kan göra det mer agilt också (s. 566)

Pluralistic walkthroughs

- Developers, usability testers och users jobbar tillsammans och steppar genom en task scenario

- Alla involverade assumerar roll som user, får prototype screen och uppgift att göra något/gå till annan skärm, skriver ner sekvens av actions de skulle ta för att göra det (enskilt), diskuterar tillsammans sen, och går sedan vidare till nästa skörm/uppgift
- **Benefits:**
 - Stark fokus users tasks, kollar steps taken
 - extra bra om security critical, ett enda steg kan vara avgörande
 - Bringar tog heter expertise av folk med många områden
- **Problem:**
 - Kan vara svårt få ihop alla researchers/utvecklare samtidigt
 - Behöver vänta på den längsammaste ska bli klar för att fortsätta, kan vara kostsamt
 - Kan bara testa limited mängd scenarion pga time constraints
 -

Good design.md

#InteractionDesignKap1

- Måste tänka på vem/var som ska använda systemet när designar det

Interaction design.md

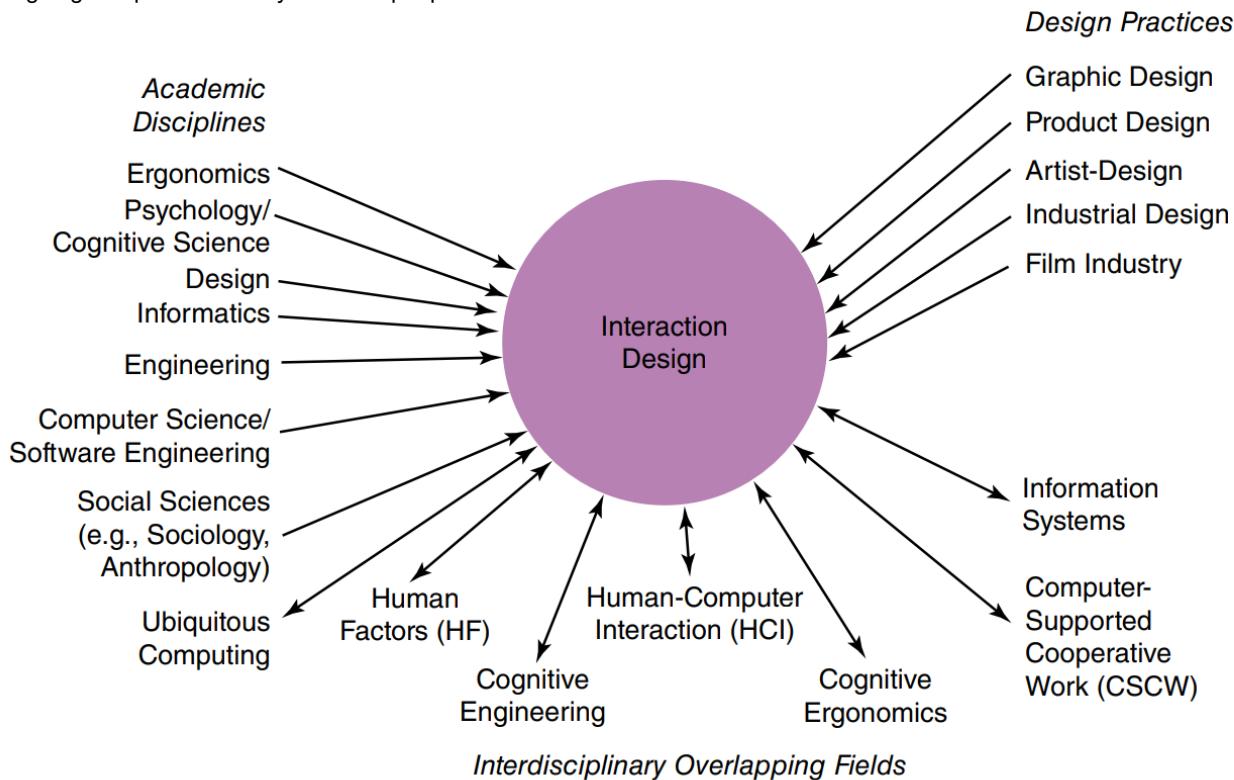
#InteractionDesignKap1

By interaction design, we mean the following: Designing interactive products to support the way people communicate and interact in their everyday and working lives

Interaction design är overarching term för UI, UX, [User centered design](#)

UX används i branschen för ex. beskriva jobbet som interaction designer

We view interaction design as fundamental to many disciplines, fields, and approaches that are concerned with researching and designing computer-based systems for people.



Kan få fler idéer om har många personer i team för interaction design, och kan få insikt i många fält eftersom behöver förstå ex ekonomi, interaktionsdesign, personer, organisation, computer sceience för att göra bra och feasible designs

- Kan dock vara problem med att de har olika mentala modeller, kostnader högre och organisering svårare

Usability and User Experience Goals.md

#InteractionDesignKap1

För förstå users måste vara tydligt vilka huvudsakliga objectives är när utvecklar produkt för dem

To help identify the objectives, we suggest classifying them in terms of usability and user experience goals. Traditionally, usability goals are concerned with meeting specific usability criteria, such as efficiency, whereas user experience goals are concerned with explicating the nature of the user experience, for instance, to be aesthetically pleasing.

Usability goals

Usability refers to ensuring that interactive products are easy to learn, effective to use, and enjoyable from the user's perspective

- Sex mål:
 - Effective to use (effectiveness)
 - Efficient to use (efficiency)
 - Safe to use (safety)
 - Having good utility (utility)
 - Easy to learn (learnability)
 - Easy to remember how to use (memorability)
- Massa info om varje mål s 19

User Experience Goals

A diversity of user experience goals has been articulated in interaction design, which covers a range of emotions and felt experiences. These include desirable and undesirable ones,

Desirable aspects		
Satisfying	Helpful	Fun
Enjoyable	Motivating	Provocative
Engaging	Challenging	Surprising
Pleasurable	Enhancing sociability	Rewarding
Exciting	Supporting creativity	Emotionally fulfilling
Entertaining	Cognitively stimulating	Experiencing flow
Undesirable aspects		
Boring	Unpleasant	
Frustrating	Patronizing	
Making one feel guilty	Making one feel stupid	
Annoying	Cutesy	
Childish	Gimmicky	

- mer info s. 23

User experience.md

#InteractionDesignKap1

The user experience refers to how a product behaves and is used by people in the real world.

Allt som används har en user experience

Kan inte designa en user experience, kan designa *för* en user experience

- Ex sleek vs klonig telefondesign

Mark hassenzahl model user experience: **Pragmatic och hedonic aspects** av user experience

- Pragmatic: Hur simple, praktiskt, obvious hur användaren ska uppnå sina mål med produkten
- Hedonic: hur evocative och stimulating experience är

accesability and inclusiveness.md

#InteractionDesignKap1

Accessibility refers to the extent to which an interactive product is accessible by as many people as possible.

Inclusive design is an overarching approach where designers strive to make their products and services accommodate the widest possible number of people.

designa för accesibility gör inherently inclusive deisgn för alla

Accessibility can be achieved in two ways: first, through the inclusive design of technology, and second, through the design of assistive technology. When designing for accessibility, it is essential to understand the types of impairments that can lead to disability as they come in many forms. They are often classified by the type of impairment, for example:

- Sensory impairment (such as loss of vision or hearing)
 - Physical impairment (having loss of functions to one or more parts of the body, for example, after a stroke or spinal cord injury)
 - Cognitive (for instance, learning impairment or loss of memory/cognitive function due to old age or a condition such as Alzheimer's disease)
- Måste göra val som funkar för så många som möjligt

Double diamond.md

#InteractionDesignKap2

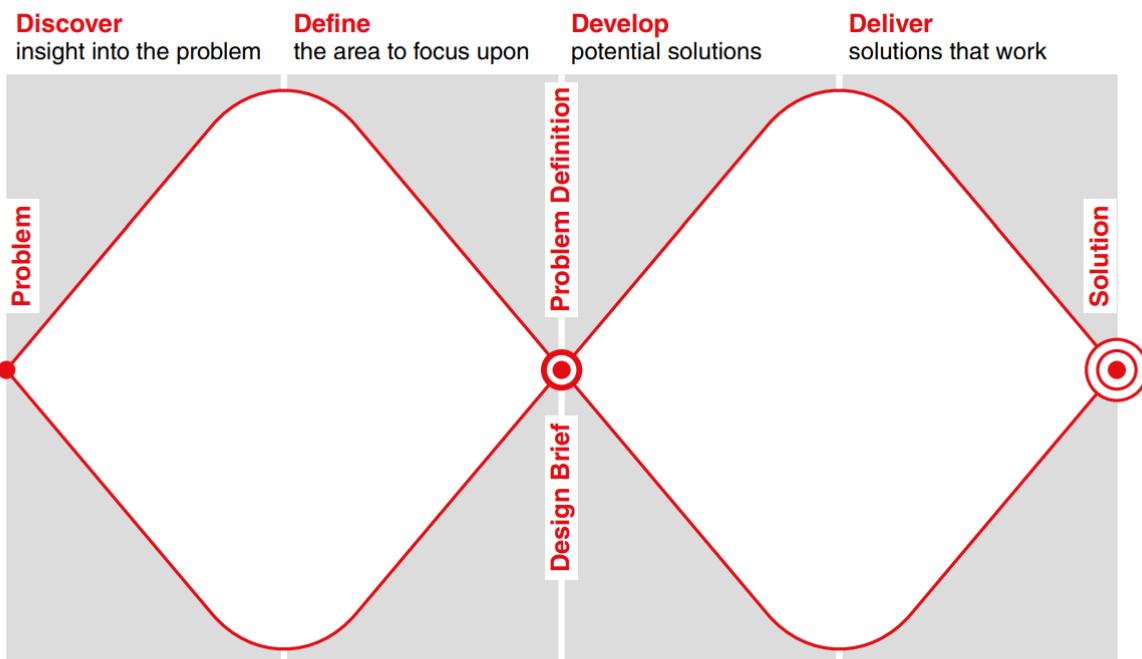


Figure 2.1 The double diamond of design

Source: Adapted from <https://www.designcouncil.org.uk/news-opinion/design-process-what-double-diamond>

double diamond of design of design capturar commonalities mellan olika typer av design

Har fyra faser:

- Discover: Designers try to gather insights about the problem.
- Define: Designers develop a clear briefthat frames the design challenge.
- Develop: Solutions or concepts are created, prototyped, tested, and iterated.
- Deliver: The resulting project is finalized, produced, and launched.

Dessa faser följs också i [Interaction design](#)

Kärnan av dessa faser fångas i filosofin av [User centered design](#), att involvera användare

- user centered approach gör att user concerns directar utvecklingen, inte bara tekniska concerns

Första steg [understanding the problem space](#)

kan appliceras på [brainstorming for innovation](#)

User centered design.md

#InteractionDesignKap2

The importance of involving users

- Bästa sättet se att användare kommer gilla produkten
- Experter kommer inte ha samma needs/perspektiv som användare -> måste använda users

vanligt med **Produkt owners**

- Någon som filtrerar in användkrav in i development cycler, prioriterar features
- ofta inte mycket kunskap interaction design
- Behöver fortfarnande involvera users

Utöver functionality är två aspekter som måste beaktas:

- **Expectation management**
 - se till att users expectations inte skilljer från vad produkten kan göra
 - ex genom representativ marketing
 - kan göras genom involva users i design process, eller träna de tidigt i prerelease
- **Ownership**
 - Om users involverade i design process känner de starkare sense av ownership mot produkten

Degrees of user involvement

- Kan ha olika nivåer
- **Full time involvement av några users**
 - De kan dock bli out of touch med sina communities och kontexter för användande
 - Fördel: continually available input
- **Part time involvement av users**
 - Kan bli hög workload för de
- **Kan involvera users i specifika aktiviteter**
 - Blir dock limited feedback till bara den aktiviteten

Kan ex använda crowdsourcing

- Ger mer perspektiv, mer satisfaction t användning av slutprodukt, starkare sense av ownership hos användare

Participatory design

- När slutanvändare bler central actors i creation activities, blir mer aktivt involverade

projektet och intended market påverkar vad som är realitstiskt

- Om kund är ett företag kan man lättare involvera de i design process jämfört med om det är öppen marknad

User centered approach

- att slutanvändare, inte bara det tekniska, är driving force behind product development
 - -> Slutprodukt kommer vara lämplig för avsedda aktiviteten, ta tillvara på human skill/judgement, inte constraina user
- **Tre huvudprinciper för user-centered approach**
 - *1. Early focus on users and tasks*
 - Måste förstå vilka användare är, studera deras kognitiva, antropomorphic och behavioural characteristics
 - För detta, måste studera de när de gör sina tasks, studera taskens natur, involvera users i designprocessen
 - *2. Empirical measurement*

- Mät reactions and performance av intended users för ex printade manualer, scenarion
- När har prototyper/simulations, mät igen
- analysera resultat
- 3. *Iterative design*
 - När hittar issues, fixar de och testar igen nya observations på samma sätt för att se effekterna av fixandet
 - Repetera dessa cykler often as necessary'

Mer detalj om varje delprincip s. 48

Early focus on users and tasks

1. Users tasks and goals are the driving force behind development
2. Users behaviour and context of use are studied, and the system is designed to support them
3. Users characteristics are captured and designed for
4. Users are consulted throughout development from earliest phases to the latest
5. All design decisions are taken within the context of the users, their activities, and their environment

Empirical measurement

om möjligt definiera [User experience](#) mål, dokumentera de tydligt. Kan då välja alternativa designs eller checka progress tidigt, och empiriskt evaluera

Iterative design

- refine designs based on feedback
 - nya perspektiv idéer kommer fram när man jobbar, users testar
 - -> måste iterera, revisa
 - speciellt när man innoverar
-

The goal of UCD is to make products with a high degree of [usability](#) (i.e., convenience of use, manageability, effectiveness, and meeting the user's requirements). The general phases of the UCD process are:

6. Specify context of use: Identify the primary users of the product and their reasons, requirements and environment for product use.
7. Specify requirements: Identified the detailed technical requirements of the product. This can aid designers in planning development and setting goals.
8. Create design solutions and development: Based on product goals and requirements, create an [iterative](#) cycle of product testing and refinement.
9. Evaluate product: Perform usability testing and collect user feedback at every design stage.

The above procedure is repeated to further refine the product. These phases are general approaches and factors such as design goals, team and their timeline, and environment in which the product is developed, determine the appropriate phases for a project and their order. Practical models include the [waterfall model](#), [agile](#) model or any other [software engineering](#) practice.

Gör det här iterativt till produkten blivit riktigt bra

Additionally, UCD ensures that design decisions prioritize users' needs rather than technical possibilities. It also encourages continuous user involvement throughout the development process to avoid incorrect assumptions and improve usability.

För tänka på users:

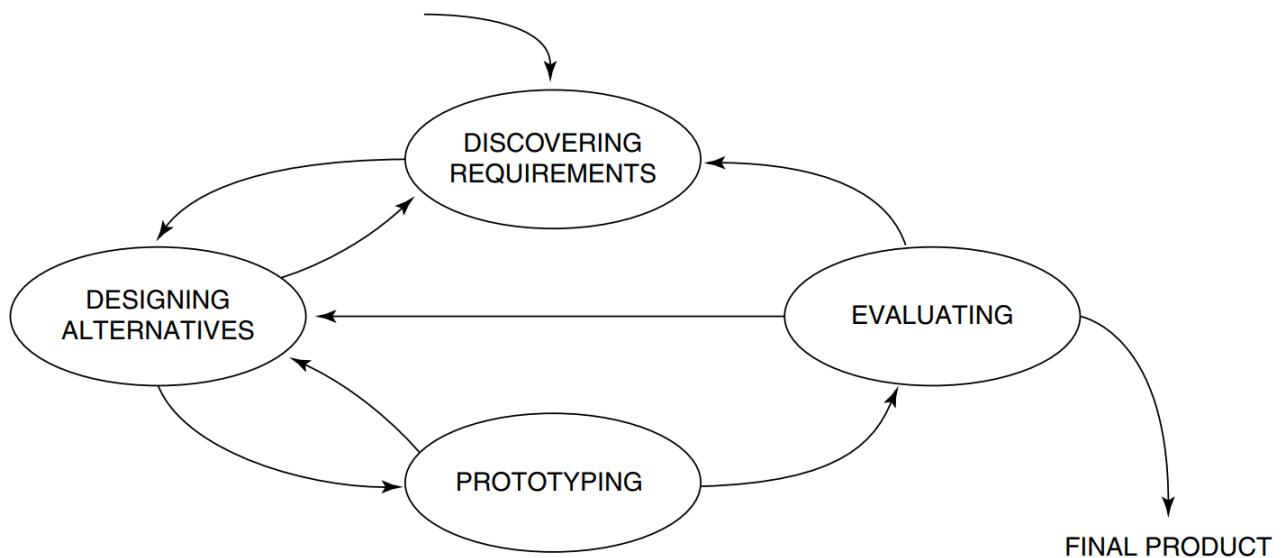
One can be more principled in deciding which choices to make by basing them on an understanding of the users. This involves the following:

- Considering what people are good and bad at
- Considering what might help people with the way they currently do things
- Thinking through what might provide quality user experiences
- Listening to what people want and getting them involved in the design
- Using user-centered techniques during the design process

Måste förstå att olika users har olika behov

- ex hur barn vs vuxna vill lära sig
- tänk på folks sensitivities, ex äldre personer vill inte känna sig lacking i kognitiv

- kulturella skillnader



process of interaction design.md

#InteractionDesignKap2

Gör ofta iterativt - se [Double diamond](#)

Deisgn handlar mycket om tardeoffs

- ex enkelhet vs mängd valmöjligheter

Four basic activities of interaction design

1. Discovering requirements for the interactive product.
2. Designing alternatives that meet those requirements.
3. Prototyping the alternative designs so that they can be communicated and assessed.
4. Evaluating the product and the user experience it offers throughout the process.

Discovering requirements

- vänstra sidan [Double diamond](#)
- upptäcka något om värld eller användare, förstå användare, bestäm vad man ska utveckla

Designing alternatives

- del av develop phase i [Double diamond](#)
- proposa idéer för möta requirements
- för interaction design kan indelas i conceptual design och concrete design
 - Conceptual design = vilka conceptual models kommer användaren använda?

Prototyping

- oxå del av develop i [Double diamond](#)
- mest effektiva sätt utveckla look feel av produkt är prototypa och testa med users

Evaluating

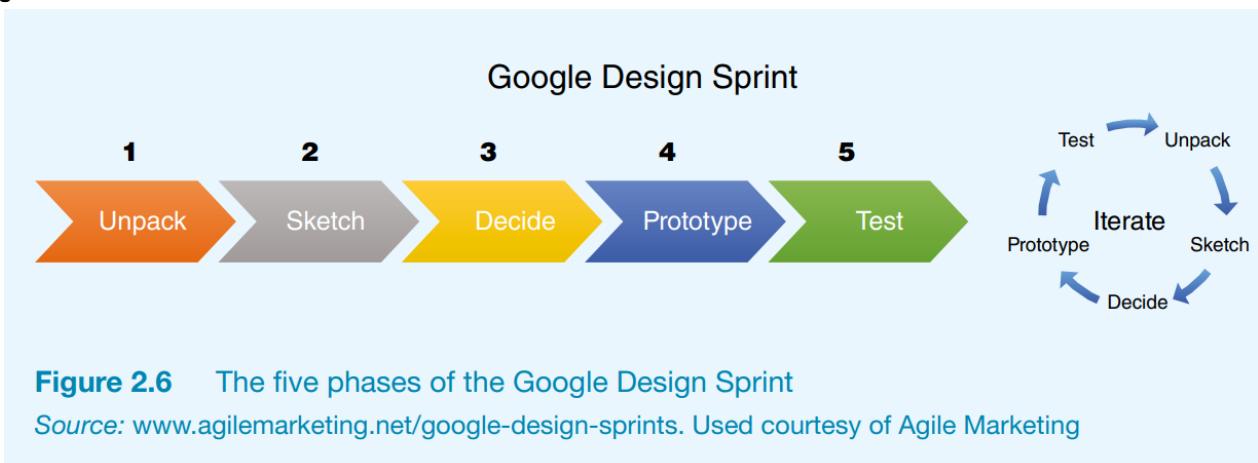
- oxå del av develop phase i [Double diamond](#)
- determine usability and acceptability

Alla dessa faser är intertwined

Models för process av interaction design

- Om komplext, många parter och utvecklare behöver modell för management

- Dessa är ofta strukturerade som lifecycles, iterativa, ibland med många alternativ
- **Google:**



- en vecka
- Info s. 53
- **RITW (research in the wild)**
 - Fokus nya teknologier som kan ändra behaviour snarare än passar conventions
 - In terms of the four activities introduced earlier, this framework focuses on designing, prototyping, and evaluating technology and ideas and is one way in which requirements may be discovered

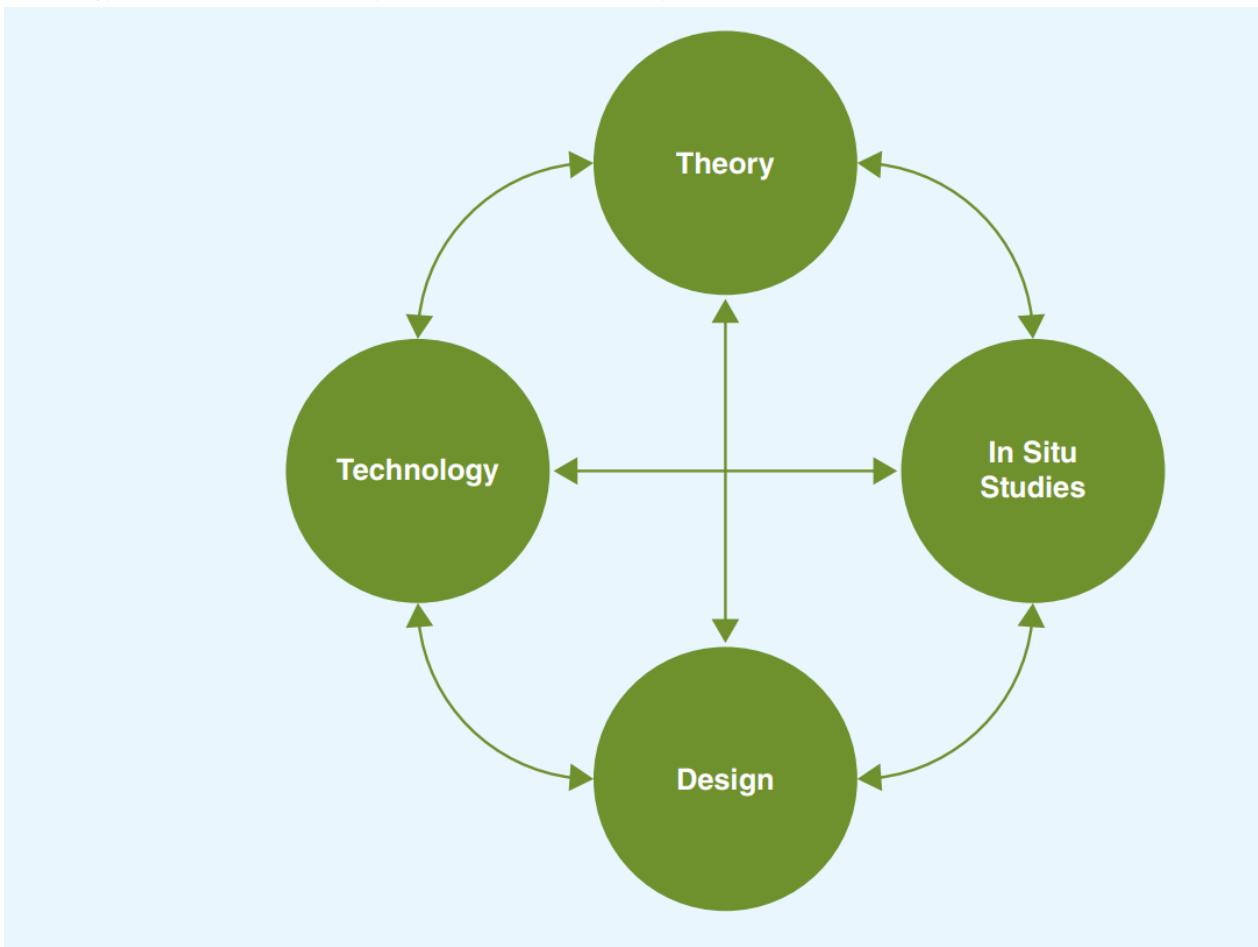


Figure 2.7 A framework for research in the wild studies

- Source: Rogers and Marshall (2017), p. 6. Used courtesy of Morgan & Claypool
- mer info s. 54

Practical issues

saker man praktiskt måste bestämma med interaction design

- Who are the users?
- What are the users' needs?
- How to generate alternative designs
- How to choose among alternatives
- How to integrate interaction design activities with other lifecycle models

finns mer info än nedan om alla dessa på s. 55 frammåt

Who Are the Users?

- Kan finnas väldigt många mer user groups än anticipated
- Stakeholders är alla som kan affecta / affectas av projektet
 - Det är inte bara users som är dessa, ta hänsyn till stakeholders

What are the users needs?

- Är inte bara att fråga vad users behöver och sen skapa det, eftersom de inte alltid själva vet att de behöver det/vad som är möjligt
- Blir istället att [understanding the problem space](#), kolla users acrivites, vad som kan förbättras, prova idéer med potentiella users
- Prova med users för helt nya innovationer
- ofta är inte det developers behöver/vill ha det som end users vill ha

How to generate alternative designs

- ofta svårt pga tendency att stick with soloution that works
- kan ofta kombinera eller vidareutveckla befintiliga idéer
- söök nya perspektiv, kolla andra designs

How to Choose Among Alternative Designs

- handlar mest om att göra många design desicions, ex om vad man ska inkludera
 - Dessa beslut påverkas av insamlad information om users och deras tasks, och teknisk feasibility av data
 - i [Interaction design](#) är faktorer som är synliga, kan uppskattas av users mer centrala än tekniska aspekter
 - prototyping för att göra det lättare för users att utvärdera designs
 - kan välja på basis av kvalitée, alla har dock olika uppfattningar om vad kvalite är

How to Integrate Interaction Design Activities Within Other Lifecycle Models

- Finns många andra lifecyclemodeller i andra discipliner som är sammankopplade med interaction design, ex för software development
 - försök att sammanföra: agile, scrum etc

understanding the problem space.md

#InteractionDesignKap2

första steg i double [Double diamond](#) och [process of interaction design](#)

ist för att börja med ex fysisk design och blueprints måste man börja med att förstå vilka användarna är, deras requirements och context av användning

- bättre att börja med tekniska aspekter efter artikulerat problem spacens natur
 - = vad problemet är, vilken [User experience](#) är i nuläget, vad dess problem är, vad ändring måste bli

Bra att involvera personer med olika kompetenser (ex software, projektledare) i projektgrupp med det här för att inte missa något

- ex inga unfounded/dåliga assumptions

Kan översiktligt presentera varför produkten behövs

Techniques for designing UX.md

#InteractionDesignKap7

- Måste kunna se det de ska interagera med
- Måste veta vad som är available
- Har en mental model över vad servicen gör och hur den fungerar
- När folk lär sig interface formar de sina intentions i kontexten av hur programvaran fungerar
- GUIs ofta based på anringen
 - Apple
 - Windows
 - Android
- GUI gör att man kan recogniza vad man vill göra istället för att behöva komma ihåg kommandot

Horton's icon checklist

William Horton (of William Horton Consulting, Inc.) has produced a detailed checklist (1991) designed to help the icon designer avoid a whole raft of common mistakes. We reproduce his top-level headings here, together with a sample question for each issue.

Understandable	<i>Does the image spontaneously suggest the intended concept to the viewer?</i>
Familiar	<i>Are the objects in the icon ones that are familiar to the user?</i>
Unambiguous	<i>Are additional cues (label, other icon documentation) available to resolve any ambiguity?</i>
Memorable	<i>Where possible, does the icon feature concrete objects in action? Are actions shown as operations on concrete objects?</i>
Informative	<i>Why is the concept important?</i>
Few	<i>Is the number of arbitrary symbols less than 20?</i>
Distinct	<i>Is every icon distinct from all others?</i>
Attractive	<i>Does the image use smooth edges and lines?</i>
Legible	<i>Have you tested all combinations of colour and size in which the icon will be displayed?</i>
Compact	<i>Is every object, every line, every pixel in the icon necessary?</i>
Coherent	<i>Is it clear where one icon ends and another begins?</i>
Extensible	<i>Can I draw the image smaller? Will people still recognize it?</i>

Några guidelines för interface design:

- Relaterade knappar tillsammans, logiska groupings
- Consistency

några element

- Radio buttons - mutually exclusive checkboxes
- Checkboxes

- Toolbars
- List boxes - lista med contents
- Sliders
- Form fill
- Wizards - komplex task step by step
- Alerts
 - Se till att inte för distraherande, speciellt om man gör något viktigt
 - kan bli / inte bli ignorerade när lämpligt
 - inte mer information än man kan responda till

interface types.md

#InteractionDesignKap7

interface types kan fokusera på funktion, tex att vara adaptiva, eller med interaction style used (ex command line, graphical), eller input/output device som används eller platformen man designar för

en produkt kan gå in under flera typer av interfaces

Några exempel:

1. Command
 2. Graphical
 3. Multimedia
 4. Virtual reality
 5. Web
 6. Mobile
 7. Appliance
 8. Voice
 9. Pen
 10. Touch
 11. Gesture
 12. Haptic
 13. Multimodal
 14. Shareable
- NOTE This chapter is not meant to be read from beginning to end; rather, it should be dipped into as needed to find out about a particular type of interface.
- 7.2 I Typ 195
 15. Tangible
 16. Augmented reality
 17. Wearables
 18. Robots and drones
 19. Brain-computer interaction
 20. Smart

Command line interfaces

- skriva in abbreviations i en konsol
- superceded av GUI
- Kan fö ha advantage om users tycker de är snabbare

Graphical user interfaces

- nära information presenteras med grafisk interface
- kan använda images, color, typography
- originella hetera wimp:
 - Windows
 - Icons - symboler för ex appar, objekt
 - Menus - list of options
 - Pointer (pointing device) - ex mus
- idag är vanligaste sättet att kontrollera touch (via mobiler)
- Många av elementen från wimp finns kvar men är elaborated på

considerations för basic building blocks i WIMP:

Windows

- gör att man kan ha fler tasks o saker per skärm
- behöver inte byta skärm för att byta vad man gör, kan kolla på samtidigt
- scrolling för mer information kan visas
- kan visa alla fönster side by side, översiktligt
- dialog boxes vanligt

Menu design

- vanligt med headers för options, vilka blir visible när man trycker på dem (ex photoshop)
- drop down menus ofta ordered top bottom mest frequently used
- *Flat menus* bra när lite options, eller om skärm liten
 - Kan dock behöva nöta saker, kan ta många klick komma till desired plats
- *Expanding menus* kan visa alternativ som expanderar i samma fönster

- Exempelvis cascading menus, "ex dropdown i en dropdown"
 - dock lätt göra misstag, overshootar eller väljer fel exempelvis
- *mega menu* en meny som har allt, ex shoppingkategorier
- *Collapsible menus* är alternativ till expanding menus, visar meny när klickar på header
- *Contextual menus* kan vara ex högerklicka på bild för få alternativ associerade med den bilden

Icon design

- assumption: lättare att lära och komma ihåg än text labels
 - kompakta, easily visible
- Har ofta bra mapping (se [Norman Everyday things](#)), speciellt om isomorphic: direkt mapping mellan vad som representeras och hur det representeras = "direct mapping"
- kan visa actions genom associations: ex soptunna'
- metafor: transfiörra knowlegde från en domän till en annan
- convention: saker blir standard över tid

Multimedia

- många typer av media som engas, kan vara mer interaktivt och mer pleasant/intuitivt att visa information så
- kan visa olika represenattioner av information snabbt
- kan ha interaktiva simulations

Virtual reality

- kan ha många modalities för att öka känslan av att det är riktigt
- många tänker mer realistikst = bättre träningssimulationer

Website design

- I början handlade mest om strukturera information i form av text
- nu måste vara usable och aesthetically pleasing
- menar att man numera inte payar mycket attention till text utan klickar/navigerar till exakt dit man vill komma så snabbt som möjligt
- breadcrumbs: för visa vilken väg man tog (hans o greta)
- måste tänka annorlunda på PC och mobil

Mobile devices

- används i alla aspekter av everyday och working lives
- har många sensorer
- barcodes o qr koder kan ge info

Appliances

- maskiner för everyday at home (ex diskmaskin)
- många vill ofta få saker gjorda snabbt med dessa, vill inte explora en interface

Voice user interfaces

- när user pratar in sin instruktion
- kan göra det on the go
- ibland mer effektivt
- ex om visually impaired är det bra
- applicerat i ex call routing
- vidare ex siri, google translate, smart speakers
- ex svårare att recognisa speech från barn

Data gathering.md

tree vanliga typer av data gathering:

1. [Interviews](#)
 1. vanligtvis synchronous: gjorda i presence av investigator (linjärt?)
2. [Questionnaires](#)
 2. asynchronous
 3. kan vara online
3. [Observation](#)
 1. kan vara Direkt, studera och notera aktiviteter när de händer, eller indirket, samla records av aktiviteter och sedan reviewa dem at a later date

Använd gärna flera för minska inherent bias fr varje

Five key issues:

Five key issues require attention for any data gathering session to be successful:

- goal setting,
- identifying participants,
- the relationship between the data collector and the data provider,
- triangulation,
- pilot studies

goal setting

- huvudsaklig reason få info om users behaviour och reaktion till teknologi

Identifying participants

- Måste ofta sampla - välja - participants
- ovanligt att kunna få saturation sampling (alla i populationen)
- välja mellan probability (ex random) sampling eller nonprobability (ex convenience) sampling

Relationship with participants

- måste ha bra o proffessionell relation m subjects
- ex consent forms som förklarar hur datan ska lagras

Triangulation

- att investigata fenomen från flera synvinklar
- fyra typer:
 - Triangulation of data means that data is drawn from different sources at different times, in different places, or from different people (possibly by using a different sampling technique).
 - Investigator triangulation means that different researchers (observers, interviewers, and so on) have been involved in collecting and interpreting the data.
 - Triangulation of theories means the use of different theoretical frameworks through which to view the data or findings.
 - Methodological triangulation means to employ different data gathering techniques.
- methodological är vanligast
- bra att få flera perspektiv dock, men kan vara svårt

Pilot studies

- liten trial run av main study
- ska se så viable innan börjar helt
- kan slösa tid att inte göra det
- kan fråga ex kollegor om nödvändigt

Interviews.md

#InteractionDesignKap8

tre typer:

• **Unstructured**

- ofta djupa, exploratory
- öppna frågor
- probing = "kan du berätta mer om?"
- *fördelar:*
 - interrelaterad rik komplex data genereras
 - intervieuves kan nämna saker man inte tänkt på
- *Nackdelar*
 - Kan vara time consuming att analysera

• **Structured**

- förbestämmda frågor som i questionnaires
- samma frågor för alla participants
- ofta closed questions
- bara bra om vet mål med undersökning samt kan definiera specifika frågor
- kan funka bra om range of possible answers är känt, participants har lite tid

• **Semi-structured interviews**

- kombinerar structured och unstructured
- open/closed ended frågor
- har script för guidance
- börjar med nya frågor, probar kontrinuerligt tills ingen mer interessant information fås
- mindre viktigt med ordning av frågor
-
- ställ inte frågor som assumar ett svar

focus group = sortera personer efter deras användarkrav och intervju de tillsammans

Från canvas

contextual interviews: intervjun sker där användaren faktiskt använder produkten

- fr design space till requirements
 - Understanda users för att göra detta
 - ex via [field studies](#), och inom de interviews
- Intervieees berättar vad de gör, ger värdefulla insights

Skillnad intervju och konversation

- Interview har definite purpose
- interviewer har responsibility directa interaction

varför intervjuer?

- Kan visa olika sätt att se eller förstå en topic
- Kan follow up previous questions

- interjuare mycket power att shapa intervju, ex vilka frågor som ställs och info som fås

WHAT IS AN INTERVIEW GOOD FOR?

PROS

- Detailed and rich
- Personal perspective
- Dynamic
- Allows following an interesting line of thought
- People want to tell their story
- Hard work
- Bias from researcher
- Non-standardised
- Requires preparation
- Time consuming to conduct, to transcribe, to analyse
- Can go wrong

bygg rapport med den som blir intervjuad

Läs mer elr skippat interaction design.md

s. 43 four approaches to interaction design

Andra av normans saker

s.445 och tidigare, fattade inte riktigt skillnad conceptual och concrete design

Prototyping for tiny fingers tog inte med det som stod i tidigare material eller sånt som inte verkade superviktigt

Norman Everyday things.md

<https://medium.com/@DEJIDOPE/summary-of-design-of-everyday-things-by-don-norman-f06f023fdc95>

Denna om 7 frågor man måste kunna svara på:

- <https://uxdesign.cc/ux-psychology-principles-seven-fundamental-design-principles-39c420a05f84>
- **Discoverability**
 - Whenever we engage with an everyday thing such as a TV remote control, or a product like a website or an application, we figure out where and how to perform various functions. Through good discoverability, we can consider the different options and choose the one that should work to meet our goal. However, we cannot do this if the actions are not discoverable.
 - Norman describes good discoverability as: "it is possible to determine what actions are possible and the current state of the device." Clear focal points (calls to action, images, and headers); visual hierarchy (content structured in order of priority); and transparent navigation systems all constitute good discoverability and understanding within a design.
 - Other design principles, particularly signifiers, feedback, and constraints (we'll come across these soon), can also increase discoverability and reduce possible errors.
- **Feedback**
 - Norman describes feedback as "some way of letting you know that the system is working on your request." He also explains that feedback must be immediate, informative, discreetly planned, and prioritised.

- He also explains that feedback must be immediate, informative, discreetly planned, and prioritised.
- Subtle feedback, however, can inform us that our action has worked without interrupting our flow. For example, the Twitter mobile app displays a small, animated message at the bottom of the screen to confirm successfully executed tweets.

• **Conceptual model**

- A conceptual model is a simple and helpful explanation of how something works. For example, a website or application onboarding experience demonstrates how to use the product or service. Norman explains, "the design projects all the information needed to create a good conceptual model of the system, leading to an understanding and a feeling of control."
- ex iconography, soptunna

• **Affordance**

- Affordance is an object's perceived action and properties that help us determine its operation. Norman states, "The term affordance refers to the relationship between a physical object and a person." For example, door handles and pull chain designs relate to their intended action; hence, form follows function.
- Let's take a chair, for example; its shape allows sitting. Then some chairs come with casters that afford to push and pull. And, in the extreme case of a fire, we might even perceive the chair's size and weight as an opportunity to break a window.
- Vad föremålet "Tillåter/affordar" en att göra. Dörhandtag affordar nedtryckning
- kan vara physical (intrinsic properties) eller percieved (vad usern tror att den kan göra)

• **Signifiers**

- A signifier communicates to us where the action should take place. Norman explains, "The term signifier refers to any mark or sound, any perceivable indicator that communicates appropriate behaviour to a person." For example, a button label tells us precisely what action the corresponding control takes.
- While signifiers clarify where to act, they also rely on cultural understanding. For instance, when Amazon launched its website in India, users mistook the search icon for a ping pong bat. This story proves that signifiers must hold relevant meaning to be recognised by the target user.

• **Mapping**

- Mapping is the relationship between controls and their effect on the world. For example, the arrow buttons on an arcade claw machine correspond to the movements of the toy grabbing mechanical arm. Norman explains, "when mapping uses spatial correspondence between the layout of the controls and the devices being controlled, it is easy to determine how to use them."
- Natural mapping enables us to subconsciously change the effects of our world, such as flicking the appropriate light switches. We can particularly find good mapping behaviours within the iOS control centre, which utilises sliding bars to communicate the effect of our actions.
- Conversely, unnatural mapping increases cognitive and physical effort. For example, an unclear cooker layout makes it difficult to determine which knobs control the corresponding hobs.
- Please note that if a natural mapping is impossible, use signifiers to clarify where the relationships exist.

• **Constraints**

- Long forms, complicated layouts, and heavily written descriptions can increase distraction and cognitive load (working memory).
- Constraints, however, can restrict the kind of interactions that can take place and therefore help reduce the amount of information we process. Norman explains, "physical, logical, semantic, and cultural constraints guide actions and ease interpretation."
- There are different forms of constraints, including physical, semantic, cultural, and logical. Each can help focus our attention on an important task and help reduce the chance of human error.
- Physical constraints restrict the possible operations; for example, we cannot move a mouse cursor outside the screen. Semantic limitations are clues to where we can perform actions; for instance, a socket has meaning to where we can insert the plug pins. Cultural restrictions are social conventions, such as waiting in a queue to be served. And finally, logical constraints help determine the alternatives; for example, scrolling is how we know how to see the rest of the screen.

Consistency

Consistency – This refers to designing interfaces to have similar operations and use similar elements for achieving similar tasks. In particular, a consistent interface is one that follows rules, such as using the same operation to select all objects. For example, a consistent operation is using the same input action to highlight any graphical object at the interface, such as always clicking the left mouse button. Inconsistent interfaces, on the other hand, allow exceptions to a rule.

Visibility

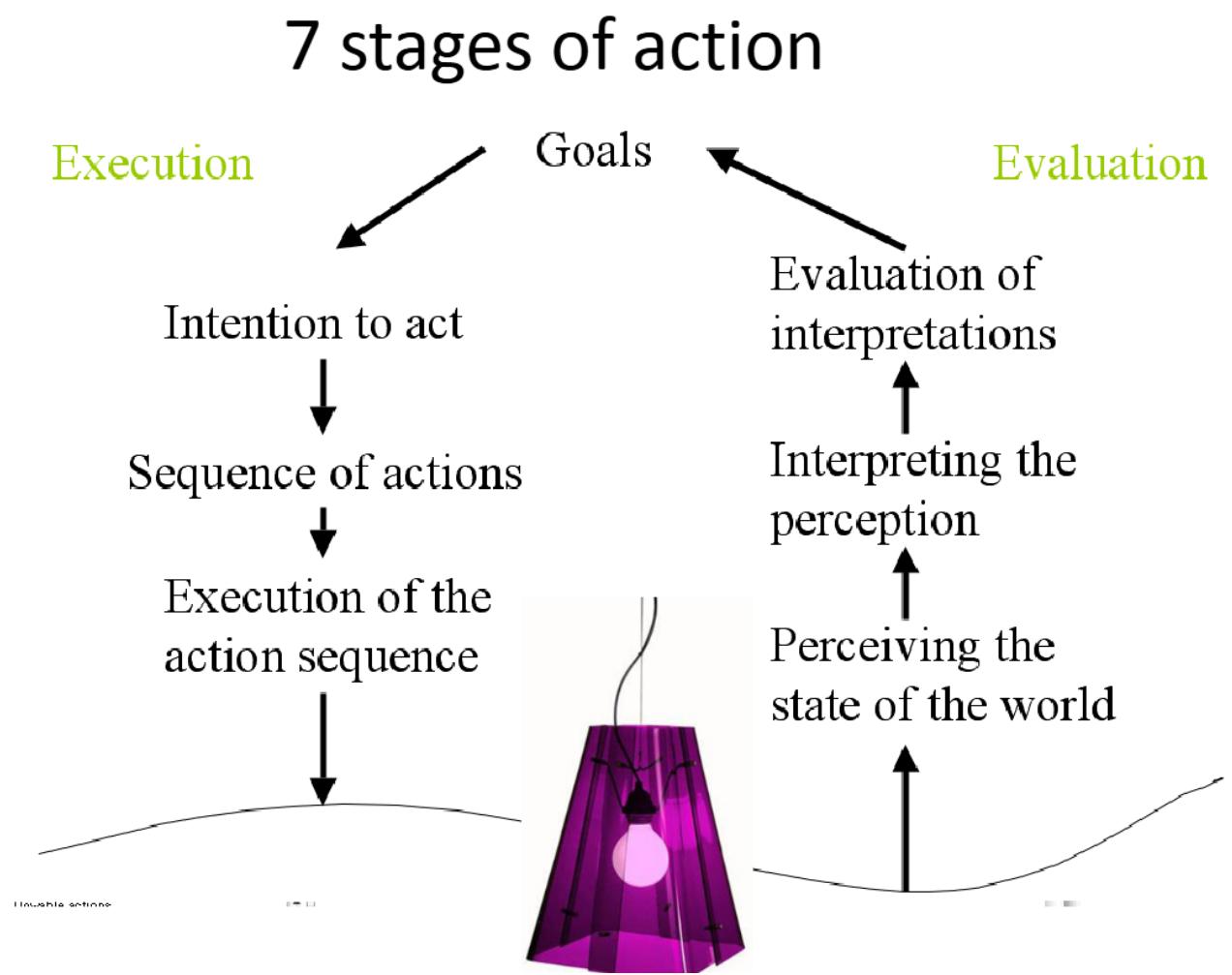
The more visible functions are, the more likely users will be able to know what to do next. In contrast, when functions are "out of sight," it makes them more difficult to find and know how to use.

Other Norman things

Mistakes

- slips: sådant man gör av misstag, man kan egentligen bättre, ex man är tankspridd
- mistakes: när man genuint har missförstått
- slips är när intenderar göra rätt, gör fel, mistakes är när intenderar göra fel, gör fel
- **Design for errors:**
 - make it possible to make mistakes: gör att konsekvenserna inte blir devastating
 - undo function
 - Ask the user often
 - forcing functions, måste svara ja för fortsätta med något

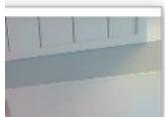
7 stages of action



- model för hur man interagerar med any type of object in surroundings
- först **execution:**
 - måste ha mål att göra något
 - Har intention att göra något för uppnå målet
 - Formulerar sequence of actions som ska göra att man kan uppnå sina actions
 - Utför denna sequence
 - måste perceive state av värld
 - Då kan man interpreta om något hände som förde en närmare ens mål
 - evaluera om det förde en närmare sitt mål

Feed forward and backward

- Feedforward – information we need in order to execute an action
- Feedback – understand the result of the action

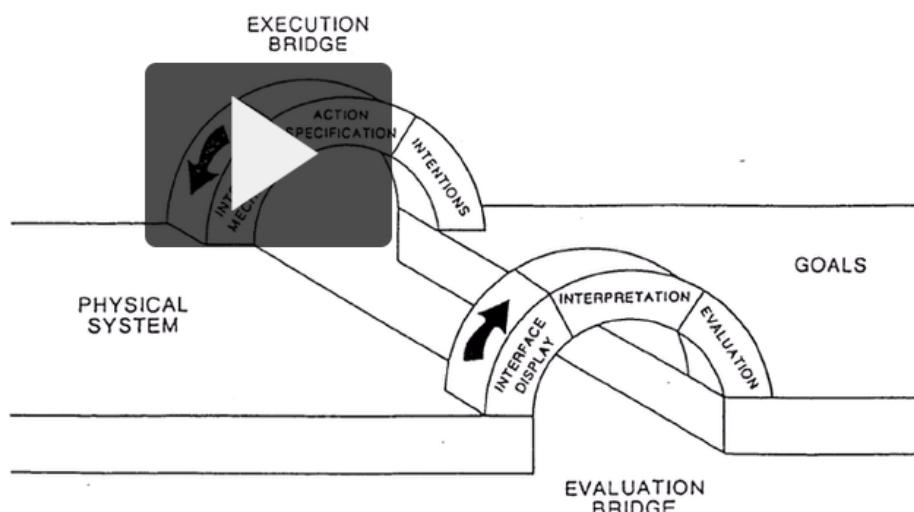
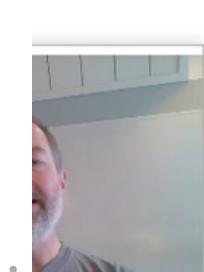


- ex kulturell information är feedforward - man har elljus överallt
- feedback - förstå att man gjorde något, att man gjorde rätt eller fel

Gulf

The gulf!

- Gulf of execution
- Gulf of evaluation



- i alla system där någon når mål genom interagera med system finns differences i understanding
- måste minimera gulf i execution och evaluation bridge
 - gör det lättare att använda, lättare att evaluera om man hade rätt

Prototyping for tiny fingers.md

problem med hifi prototypes:

1. Hi-fi prototypes take too long to build and change
 - viktigt att gå igenom många iterationer för att förbättra designs eftersom det förbättrar designs, detta långsamt med hifi prototypes
 - snabbt att lära sig
 - lätt ändra
2. Lättare att tappa bort sig i detaljer
 1. Tänker mer på content i lofi

3. Developers resist changes
 1. Eftersom tar lång tid implementera
4. A prototype can set expectations that will be hard to change
 1. blir ibland väldigt bra med prototyping tools, svårt leva upp t expectations
5. En enda bugg i hifi prototyp kan halta tester

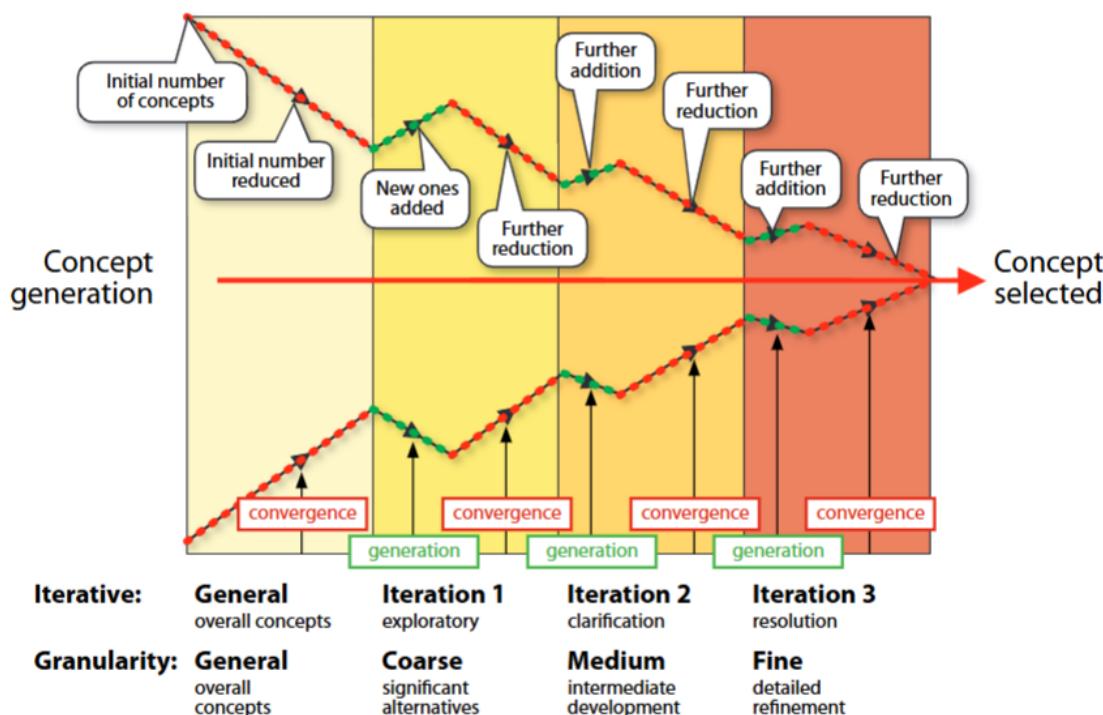
lofi bra eftersom fokuserar på users, formative evaluation, och maximerar antal iterationer kan göra innan måste börja koda

Några steg i förberedande för test

1. gör lagom mycket research innan test för att få uppfattning om vilka users är
 - kan dock vara svårt att alltid hita bona fide users att göra underöskning med
2. Förbered test scenariors
 1. Ex vad man ska göra
3. practice
 1. förbered så att testerna går bra till
- 4.

Design funnel.md

- Att alternata mellan convergance och generation av idéer
- Generera och refina solutions till når resolution
 - granularity av idéer blir fina desto längre fram man kommer



- Kan lägga till explicit design funnel innan greenlightar softwareprojekt för att utveckla idéer i parallell, filtrera fram de bästa lösningarna innan börjar

The 10 plus 10 method hjälper en att descend i the design funnel

Steg:

1. State your design challenge
2. Generate 10 or more different design concepts of a system that addresses this challenge
 1. ska bara generera många diverse lösningar, tänk inte på kvaliteten
 2. kan använda [sketching](#)
3. Reduce the number of design concepts
 1. ta bort de som inte verkar bra

2. använd sketchbook för att förklara resterande för andra
4. Choose the most promising design concept(s) as a starting point.
 1. Kommer veta vilka dessa är genom sin egna intuition, eller andras reaktion
5. Produce 10 details and/or variations of a particular design concept.
 1. skapa 10 sätt att realisera det konceptet
 2. gå sen lite djupare i konceptet och flesha ut details av idén
6. Present your best idea(s) to a group.
 1. Be om feedback om possible redesigns
7. As your ideas change, sketch them out
 1. Continue to refine and generate your concept as needed

finns bra case study av detta i sista texten här https://canvas.kth.se/courses/52114/pages/sketching?module_item_id=1031793

sketching.md

- viktig del i [process of interaction design](#)
- att sketchar ut idéer och interfaecs gör kommunikation och design snabbare
- ska bara ge en outline/essence av produkt, inte många detaljer
- kan vara evocative på sätt som tekniska ritningar inte kan vara, bra för interaction
- bra kommunikationsverktyg
- (ofta) independent av ritningsförmåga
-
- kan snabbt och iterativt prova många och förbättra lösningar
- är inte ritningar över final produkt, istället kommunikation mellan designer, developer och customer
- users sketcher kan visa hur de vill använda systemet

Varför sketcha?

- Sketching handlar om design ist för att rita
- Hjälper designers ta fram och kommunicera idéer
- Kritisk del i processen -> idea generation ([ideation](#)) -> design elaboration -> design choices -> engineering
- hjälper en testa väldigt många olika idéer och hitta ett globalt maxima för bästa designen (kan inte bara prova att utveckla vidare på en idé)

Getting the right design

- generera många idéer
- reflektera på alla
- välj de som är mest promising, utveckla de i parallel
- lägg till nya idéer när de påkoms

Elaboration and reduction

- Sätt att komma fram till bra designlösning
- *Elaborate* är att generera lösningar, *Reduction* är att välja mellan vilka som är värda att persua

The design funnel