

Orliste basert på forelesninger i INF1500 ved Suhas Govind Joshi og Jo Herstad ved Univeritetet i Oslo.

Kjekke ord å kunne

Holisme (Holistisk)	<ul style="list-style-type: none">• Helheten i et system er avgjørende for hvordan hver enkelt del oppfører seg
Reduksjonisme (Reduksjonistisk)	<ul style="list-style-type: none">• Motsatt av holisme.• Et komplekst system kan forklares ved reduksjon til hver enkelt bestanddel.

Kapittel 1: Introduksjon til HCI

Ord	Forklaring
HCI (Human-Computer interaction)	Studiet om å forstå hvordan mennesker interagerer med teknologi, og hvorvidt maskinene er utviklet for vellykket interaksjon med mennesker. Enkel definisjon: kurset handler om hvorfor og hvordan. Opptatt av å forstå hvordan: <ul style="list-style-type: none">• mennesker tenker, resonnerer, forstår, planlegger, reagerer etc.• Mennesker er en del av en sosial struktur• mennesker ønsker å utføre oppgaver som hvile, leke, arbeide etc.• Teknologien fungerer.
Interaksjon	Måten vi samhandler med en maskin på. Person <- interaksjon -> Grensesnitt <-> System
Instruerende interaksjon	Gi kommandoer og/eller velge fra valgmeny (knapper) Positivt fordi «Jeg vil åpne DEN appen»
Konverserende interaksjon	En dialog med systemet. Formål: Simulere en samtale med en annen person. Kan være alt fra Siri til chat-systemet.
Manipulerende interaksjon	Manipulering av objekter i virtuellet eller fysisk rom. (flytte apper. "kaste" filer i søppelbøtta, zoom)
Eksplorerende (utforskende) interaksjon	Bevegelse eller forflytning i virtuelle eller fysiske omgivelser. (Minecraft, Kinect, (eksplorerende i fordi vi forflytter oss for å gi input til systemet.))
Grensesnitt	Kommunikasjonspunktet der to ulike systemer møtes og

Ord	Forklaring
	<p>interagerer.</p> <p>Menneske <-interaksjon->grensesnitt<->system</p> <p>grensesnitt typer:</p> <ul style="list-style-type: none"> - Talebasert - Touchbasert - Bevegelsesbasert - Tangible - Wearable - Haptic - Commandbased
Hvordan velge grensesnitt?	<ul style="list-style-type: none"> • Avhenger av flere faktorer <ul style="list-style-type: none"> ◦ Oppgave <ul style="list-style-type: none"> ▪ Hva skal gjøres, hvor lenge, hvor omfattende, hvor mange ganger? ◦ Bruker. <ul style="list-style-type: none"> ▪ Ekspert eller novise, en eller flere brukere? ◦ Kontekst <ul style="list-style-type: none"> ▪ Hvor er de? Hjemme, på farten, på ferie etc. ◦ Kostnad <ul style="list-style-type: none"> ▪ Dyrt å implementere? Hvor lenge holder det? Skalerer det bra? ◦ Robusthet <ul style="list-style-type: none"> ▪ Er det solid? Hva tåler det? Hvilket materiale?
Antakelser	<ul style="list-style-type: none"> • Å ta noe for gitt når det krever videre undersøkelse
Kontekst	<ul style="list-style-type: none"> • Omstendighetene hvor en hendelse skjer. Kan være fysiske, sosiale, kulturelle omstendigheter som omkranser en situasjon.
Problemområde	<ul style="list-style-type: none"> • Hva ønsker vi å lage? • Hvilke antakelser gjør vi? • Vil vi oppnå det vi ønsker at det skal? • En god forståelse an problemområdet informerer godt design -> hva slags grensesnitt, oppførsel, funksjonalitet som må tilbys.
De fire bølgene innen HCI I slides Side 34 ->	<ol style="list-style-type: none"> 1. 1980: Brukerstøtte <ul style="list-style-type: none"> • Maskinsentrert fokus. • Fokus på å bringe alt til skrivebordet 2. 1990: Brukbarhet <ul style="list-style-type: none"> • Kontekst-spesifikt fokus • Flyttet oss fra skrivebord til arbeidsområde. • Brukerne fikk egne behov, teknologien skulle hjelpe med å løse dette. • Partisipatory design! 3. 2000: Brukeropplevelser <ul style="list-style-type: none"> • Fokus inndelt i ulike kontekster! <ul style="list-style-type: none"> ◦ Ikke bare jobb. Kom inn i kontekster som hytta,

Ord	Forklaring
	<p>hytta etc.</p> <ul style="list-style-type: none"> • Kulturer er viktig, forskjellige typer mennesker, trenger forskjellige verktøy. • Kan følelser knyttes opp mot design? <p>4. 2010: Bruksverdier</p> <ul style="list-style-type: none"> • Fokus på verdier og større implikasjon ved designet. • Menneskene har verdier tilknyttet behov. <ul style="list-style-type: none"> ◦ Pante telefoner etc. ◦ Passe på miljøet. • Brukere får mer makt. <ul style="list-style-type: none"> ◦ Kontinuerlig produkt-innovasjon.

Kapittel 3 og 7: Bruk og brukere

Bruker	<ul style="list-style-type: none"> • En som bruker et system. Veldig vidt begrep! • Hvem/Hvor/Når/Hvorfor er brukeren? • Hvordan forstå bruker: <ul style="list-style-type: none"> ◦ Tanker ◦ Kognisjon ◦ Følelser ◦ Behov ◦ Relasjon til steder, andre mennesker, «ting» etc.
Kognisjon	<p>Deles ofte opp i hoveddeler(/bolker):</p> <ol style="list-style-type: none"> 1. Oppmerksomhet (Attention) <ul style="list-style-type: none"> • Fokuserert <ul style="list-style-type: none"> ◦ Fokuserer på små omgivelser. ◦ Lese til eksamen f.eks. • Åpen <ul style="list-style-type: none"> ◦ Må se hele omgivelsen ◦ Sykler rundt i byen. ◦ Trenger helhetsbildet • Oppmerksomhet handler om alle sansene. (Syn, hørsel, taktil (kjenne på), lukt, smak) • Implikasjoner <ul style="list-style-type: none"> ◦ Gjøre informasjon tydelig ◦ Unngå støy 2. Persepsjon og gjenkjenning <ul style="list-style-type: none"> • Hvordan informasjon blir "innhentet" fra verden og omgjort til "erfaringer" • Hvordan "kjenner vi igjen" noe vi har sett og erfart fra før. • Implikasjoner for design: <ul style="list-style-type: none"> ◦ Ikoner: minne om noe i "verden" ◦ Metaforer: noe familiært ◦ Gruppering av informasjon ◦ Tydelighet og enkelhet 3. Minne – hukommelse

	<ul style="list-style-type: none"> • Erfare – så gjenskape • Vi husker ikke alt, men filtrerer ut uviktig info. • Vi kjenner oss igjen i kontekst/omgivelse/situasjon • Gjenkjenne vs huske noe <ul style="list-style-type: none"> ◦ Lettere å gjenkjenne enn huske. ◦ Lettere med grafisk grensesnitt enn kommando-basert grensesnitt • Implikasjoner <ul style="list-style-type: none"> ◦ Ikke overbelast minne <p>4. I forhold til aktiviteter.</p>
Kognitive rammeverk	<ul style="list-style-type: none"> • Interne rammeverk <ul style="list-style-type: none"> ◦ Mentale modeller ◦ Gjennomføring og evaluering "gulf" (Donald Norman) <ul style="list-style-type: none"> ▪ Det er en stor forskjell på hvordan det var tenkt at systemet skulle gi input/output, og hvordan det faktisk responderer ◦ Intern-modell <ul style="list-style-type: none"> ▪ Se på mennesket som en informasjons-prosess. <ul style="list-style-type: none"> • Vi får input og gir en output • Eksterne rammeverk <ul style="list-style-type: none"> ◦ Vi tilegner verktøy i omgivelsene oppgaver, og så samhandler vi med disse. (Kalkulator) ◦ Distribuert kognisjon ◦ Embodied cognition (Kroppslig kognisjon)
Ekstern kognisjon	<ul style="list-style-type: none"> • Vi tilegner verktøy i omgivelsene oppgaver, og så samhandler vi med disse. (Kalkulator)
Metafor	<ul style="list-style-type: none"> • Noe vi kjenner fra før (Familiært)
Miller om hukommelse	<ul style="list-style-type: none"> • 7 pluss minus 2 • Mennesker husker ikke veldig mye. • Ikke belast brukere veldig mye!
Mental modell	<ul style="list-style-type: none"> • Hvordan en person "ser for seg" at noe fungerer. Disse varierer fra person til person.
Representert modell	<ul style="list-style-type: none"> • Hvordan vi viser frem den implementerte modellen for brukeren. Målet er å gjøre den så lik den mentale modellen som mulig. (Ingen skarpe kanter).
Implementert modell	<ul style="list-style-type: none"> • Hvordan systemet faktisk fungerer. (Koden, programmeringen av systemet).
Datainnsamling	<p>Metoder for å samle inn data. Foregår hele tiden i alle faser vi jobber i.</p> <ul style="list-style-type: none"> • Sett klare mål • Relasjon med deltakere <ul style="list-style-type: none"> ◦ Vær profesjonell. ◦ Tenk på hvor intervjuet er. ◦ Samtykke når nødvendig.


	<ul style="list-style-type: none"> • Triangulering • Pilotundersøkelser
Mål	Hvordan skal dataen brukes og analyseres?
Triangulering	<p>Bruk av flere teknikker for å samle inn data for å få en bedre oversikt over dataen. Viktig for å fjerne store feil i et av datasettene. Har fire "grunnpilarer"</p> <ol style="list-style-type: none"> 1. Triangulering betyr å hente data fra ulike kilder til forskjellig tid, et annet sted eller fra forskjellige personer. Gjerne med forskjellige sampling-teknikker. 2. Investigator triangulation er når det er tatt i bruk forskjellige observatører, intervjuere etc. 3. Triangulering av teorier er når en tar i bruk forskjellige teoretiske rammeverk for å undersøke dataene. 4. Metodisk triangulering er å bruke flere måter å samle inn data.
Pilotundersøkelser	<ul style="list-style-type: none"> • Viktig å teste på forhånd!!! (Før den faktiske undersøkelsen) • Teste kvalitet av undersøkelsen, slik den står. <ul style="list-style-type: none"> ◦ Hvordan fungerer opplegget? • Undersøke om vi får svar på det vi ønsker svar på. <ul style="list-style-type: none"> ◦ Kvalitet på dataen vi får inn.
Data-innsamlingsmetoder	<ul style="list-style-type: none"> • Intervju • Observasjon • Spørreskjema
Varsomhet innen datainnsamling	<ul style="list-style-type: none"> • Etiske spørsmål • Hvem "eier" dataene? • Informert samtykke • NSD – Norsk samfunnsvitenskaplige datatjeneste
Observasjon	<ul style="list-style-type: none"> • Observere hva som foregår i enten i felt eller kontrollert på et laboratorie. • Rammeverk or observasjon <ul style="list-style-type: none"> ◦ Stedet ◦ Aktørene ◦ Aktiviteter ◦ Objekter og ting ◦ Handlinger ◦ Hendelser ◦ Tid ◦ Mål ◦ Følelser
Direkte observasjon	<ul style="list-style-type: none"> • Når det er med en observatør, som noterer ned hva personen gjør, helst uten å hjelpe til. • Etnografi • Kan gjøres i kontrollerte og ukontrollerte omgivelser.
Indirekte observasjon	<ul style="list-style-type: none"> • Når det ikke er med en observatør. • Dagbok.

	<ul style="list-style-type: none"> • Logging fra systemer
Observasjon i naturlige omgivelser	<ul style="list-style-type: none"> • Når en observerer noe der det faktisk skal brukes. • En musikkspiller som skal brukes når en jogger, observeres mens brukeren faktisk jogger.
Observasjon i unaturlige omgivelser (Kontrollerte omgivelser)	<ul style="list-style-type: none"> • Når personen som observeres ikke er der de naturlig ville brukt produktet. • De befinner seg gjerne på en lab, der observatøren kan gi spesifikke oppgaver, og måle metrikker lettere.
Etnografi	<ul style="list-style-type: none"> • Reiser ut i felt, til det stedet der det du er interessert i foregår, og observerer. • Dra til en flyplass hvis du vil undersøke informasjonsflyt blant sikkerhetsvakter.
Intervju	<ul style="list-style-type: none"> • Kan være u-/semi-/Strukturert. • Kan være en til en, eller gruppeintervju • Intervju i kontekst. Mens brukeren utfører en oppgave. <ul style="list-style-type: none"> ◦ Fører til mer spesifikk informasjon. ◦ Mindre sannsynlighet for bias når du spør hvordan noe gjøres. • Intervju på "sterilt" kontor. Utenfor kontekst <ul style="list-style-type: none"> ◦ Kommer mer abstrakt informasjon • Gjennomføring av intervjuet <ul style="list-style-type: none"> ◦ Introduksjon <ul style="list-style-type: none"> ▪ Forklare mål for intervju, etiske spm., frivillighet ◦ Oppvarming <ul style="list-style-type: none"> ▪ Lette spørsmål => Bli varm i trøya ▪ «Hva heter du?» + «Hvor kommer du fra?» etc. ◦ Hoveddel <ul style="list-style-type: none"> ▪ De tunge spørsmålene ▪ Prøv å ha de tyngste spm. til slutt. ◦ Avrunding <ul style="list-style-type: none"> ▪ Spørsmål for å roe ned stemningen. ◦ Avslutning <ul style="list-style-type: none"> ▪ Påpek at intervjuet er over. ▪ Skru av evt. Lyd/billedopptaker
Ustrukturert intervju	<ul style="list-style-type: none"> • Stor åpenhet rundt intervjuet. • Kanskje noe informasjon en er interessert i å hente ut • Ikke mer struktur enn dette i intervjuet
Semi-strukturert intervju	<ul style="list-style-type: none"> • Ofte en strukturert del, og en del der du kan være litt mer åpen.
Strukturert intervju	<ul style="list-style-type: none"> • Fastlåste spørsmål • Alle testpersoner får samme spørsmål • Ikke rom for variasjon eller endring av spørsmål underveis
Hvordan stille gode spørsmål	<ul style="list-style-type: none"> • God oversikt over åpne og lukkede spørsmål. • Korte eller lange spørsmål

I boka: 10.4.1	<ul style="list-style-type: none"> • Stammer fra etnografisk datainnsamling. • Brukeren er ekspert. • Utvikler er novise. • Er en blanding av direkte observasjon i naturlige omgivelser, diskusjon og rekonstruksjon av tidligere hendelser. • Når en person drar til f.eks. en arbeidsplass, observerer mens de arbeider, og diskuterer med de hva de gjør og hva som kan endres. (f.eks. Data-arbeidet til plenumsforeleseren i Afrika) • Støtter seg på fire prinsipper: <ol style="list-style-type: none"> 1. Kontekst (Context) <ul style="list-style-type: none"> • 2. Partnerskap (Partnership) <ul style="list-style-type: none"> • Mellom bruker og utvikler 3. Tolkning/Forståelse (Interpretation) <ul style="list-style-type: none"> • Observasjonene må tolkes/forstås for å kunne brukes innen design. Tolkningen/Forståelsen skal skje som et samarbeid mellom bruker og utvikler/designer. 4. Fokus (Focus) <ul style="list-style-type: none"> • Hold fokuset ved datainnsamlingen på målet. Det kan være lett å miste fokus.
Informert samtykke	<ul style="list-style-type: none"> • Når en person samtykker til noe på bakgrunn av god informasjon om hva de samtykker til.
Utvalg	<ul style="list-style-type: none"> • Er et utvalg fra målgruppen.
«5 key issues»	<ol style="list-style-type: none"> 1. Sette mål <ul style="list-style-type: none"> • Det er viktig å sette mål før datainnsamlingen. Dette vil hjelpe med hvilken form for datainnsamling en skal bruke, for å "hente" den informasjonen en er på jakt etter. 2. Identifisere deltakere <ul style="list-style-type: none"> • Målene som settes vil si noe om HVEM jeg ønsker å hente data fra. De som passer inn her er populasjonen. • Ofte kan en ikke ha datainnsamling med alle i populasjonen, og må gjøre et utvalg => sampling. 3. Forhold til deltakere <ul style="list-style-type: none"> • Forholdet mellom personen(e) som samler, og personen(e) som "gir" data er svært viktig. • Pass på at dette er profesjonelt! • Samtykkeerklæring. 4. Triangulering 5. Pilotstudier <ul style="list-style-type: none"> •
Populasjon	<ul style="list-style-type: none"> • Personer som kan være relevante for datainnsamling
Sampling	<ul style="list-style-type: none"> • Et utvalg av personer i populasjonen. • Dersom det er umulig å gjøre et utvalg, og du kan ha

	<p>datainnsamling med alle => Saturation sampling. Dette er veldig sjeldent</p> <ul style="list-style-type: none"> • Random sampling er mulig ved å tilegne hver person i populasjonen et nummer, og trekke tilfeldig. • Convenience sampling er når du bruker de som var tilgjengelige, fremfor å velge ut de du ønsker selv.
--	---

Kapittel 9: Utviklingsprosesser og tilnærminger

Hvorfor fungerer ikke løsninger vi har designet?	<ul style="list-style-type: none"> • Fravær av en designprosess som tillater at kunnskap om brukerens behov innhentes, analyseres og brukes til å drive utviklingen. • Malplasserte prioriteringer av produkthåndtering og utviklingsteam • Ignorering av de faktiske brukerne og deres grunnleggende behov • Interessekonflikt når designere skal bygge løsningene selv
Når bør vi tenke på design?	<ul style="list-style-type: none"> • Helt fra starten av • Bør være en sentral prosess gjennom hele utviklingen av produktet
Prosess innen interaksjonsdesign (Designprosess) Vi lærer fire tilnærminger	<ul style="list-style-type: none"> • Mål-orientert problemløsning • Kreativ og skapende aktiviteter <ul style="list-style-type: none"> ◦ Ikke noe fasit-svar • Må ta beslutninger hele veien • Glir ofte over i hverandre • Viktig å skille mellom hva en selv ønsker å bruke, og ha som er best for problemløsningen
Den iterative modellen for UCD	<ol style="list-style-type: none"> 1. Identifisere brukerens behov, og etablere krav <ul style="list-style-type: none"> • Hvem er målbrukerne? • Hvilken støtte kan vår interaktive løsning tilby? 2. Utvikle alternativer til design som løser problemet <ul style="list-style-type: none"> • Foreslå løsninger som imøtekommer kravene • To subaktiviteter <ol style="list-style-type: none"> 1. Konseptuelt design 2. Fysisk design 3. Prototype ulike versjoner av løsningene <ul style="list-style-type: none"> • Prototyping. • Kan være av papp, papir, trefigur, eller et program. 4. Evaluere de ulike løsningene <ul style="list-style-type: none"> • Evaluering • Avgjøre brukbarhet og respons på produktet eller designet
Struktur (av modellen over)	<ul style="list-style-type: none"> • Hva? (Identifisere behov og etablere krav <div style="display: flex; align-items: center; justify-content: space-between;"> <div> Bakrunnsstudier Intervjuer </div> <div style="text-align: center;">  </div> <div> Use case Oppgaveanalyse </div> </div>

	<p>Spørreskjemaer Meningsmålinger</p> <ul style="list-style-type: none"> Hvordan? (Deigne alternativer) <ul style="list-style-type: none"> Ulike prototyper <ul style="list-style-type: none"> Skisser, storyboards, wireframes etc. Hvorfor? (Evaluere løsninger) <ul style="list-style-type: none"> Gjennomganger (Walkthrough) Brukbarhetstesting Feltstudier Heuristisk evaluering <p>Scenario Kravspesifikasjon</p>
Konseptuelt design	<ul style="list-style-type: none"> Produsere en konseptuell modell or produktet. <ul style="list-style-type: none"> Hva skal det gjøre, og hvordan skal det oppføre seg, se ut etc.
Fysisk design	<ul style="list-style-type: none"> Forslag til produktets detaljer <ul style="list-style-type: none"> Farger, lyder, bilder som brukes, menydesign, ikoner, grensesnitt etc. Hvordan kan vi bruke design-retningslinjer for å realisere disse prototypene
UCD (User-centered design)	<ul style="list-style-type: none"> Fokus på brukeren av løsningen Brukeren vet best <ul style="list-style-type: none"> Eneste som får lov til å styre og påvirke designet Det brukeren vil ha er vår sannhet Brukers ønsker, behov og begrensninger tas i betraktning i hver aktivitet i utviklingen Donald Norman! Aldri feil å lage noe brukeren vil ha, selv om det ser rart ut. Brukersentrert tilnærming baserer seg på <ul style="list-style-type: none"> Tidlig fokus på brukeren og oppgaven som skal løses <ul style="list-style-type: none"> Direkte studier av brukers oppførsel, samt deres kognitive, antropomorfske holdningsmessige egenskaper. Empiriske undersøkelser og resultater <ul style="list-style-type: none"> Brukers reaksjoner, prestasjoner ved bruk av scenarioer, manualer, simuleringer og prototyper blir observert, tatt opp og analysert. Iterativt design <ul style="list-style-type: none"> Repeterende prosess der problemer oppdages under brukertesting, fikses, og testes på nytt Typiske metoder og teknikker <ul style="list-style-type: none"> Etnografi Contextual inquiry Brukbarhetstesting
Antropomorfisering	<ul style="list-style-type: none"> Tillegge en datamaskin egenskaper som vi mennesker har, men egentlig ikke datamaskinen.
ACD (Activity-centered design)	<ul style="list-style-type: none"> Fokus på aktiviteten til brukeren og omgivelsene rundt

	<p>bruken an en løsningen</p> <ul style="list-style-type: none"> Ikke opptatt av brukerens behov og mål Interessert i oppførsel, væremåte og karakteristikk Typiske metoder og teknikker <ul style="list-style-type: none"> Observasjon Intervjuer Etnografi
Systems design (System design)	<ul style="list-style-type: none"> Fokus på systemet <ul style="list-style-type: none"> Menneskene kommer i andre rekke, men forsvinner ikke Strukturert, rigid, holistisk tilnærming med fokus på kontekst Siden systemet er fokus brukes tilnærmingen ofte i komplekse problemer Typiske metoder og teknikker <ul style="list-style-type: none"> Modellering (UML, ERD) Brukerhistorier Intervjuer
Genius design	<ul style="list-style-type: none"> Designeren skaper ideer som brukeren vurderer i etterkant av designprosessen. Designeren "vet" hva brukeren trenger/vil ha Bruker ikke involvert i designprosessen. Avhenger av designers mye kunnskap, erfaring og kreativitet Typisk Apple Typiske metoder og teknikker <ul style="list-style-type: none"> Observasjoner Studier av liknende produkter og tjenester Rapid prototyping
Participatory design (PD)	<ul style="list-style-type: none"> Stammer fra Skandinavia, deriblant forskning i Norge Fokus på å aktivt involvere alle brukere og aktører gjennom hele designprosessen Holder ikke å bare forstå brukeren. <ul style="list-style-type: none"> De må også være med å forme løsningene Brukerne skal være med til det ekstreme. De skal være med å foreslå løsninger, i workshops og brainstorming. «It is not the same having a voice, and having a say» <ul style="list-style-type: none"> UCD: The user has a voice PD: The user has a say Typiske metoder og teknikker <ul style="list-style-type: none"> Workshops Brainstorming Storyboard
Primærbruker, sekundærbruker, tertiærbruker	<ul style="list-style-type: none"> Primærbruker: <ul style="list-style-type: none"> De som interagerer direkte med systemet regelmessig <ul style="list-style-type: none"> Jeg låser opp døren til leiligheten hver dag

	<ul style="list-style-type: none"> • Sekundærbruker <ul style="list-style-type: none"> ◦ De som av og til, eller via noen andre, interagerer med systemet <ul style="list-style-type: none"> ▪ En gjest som kommer inn i huset ditt • Tertiærbruker (Resten av verden -> vi gjør det litt mer spesifikt) <ul style="list-style-type: none"> ◦ De som blir berørt av systemet, eller påvirker det <ul style="list-style-type: none"> ▪ Alle som bor rett ved meg. De
Livssyklusmodeller	<ul style="list-style-type: none"> • Angir aktivitetene som inngår i utviklingsprosessen og hvordan de relaterer til hverandre • Sier noe om livssyklusen til designprosessen • Forenklede modeller av virkeligheten. • Den enkle livssyklusmodellen for UCD (Ligger i mappe) <ul style="list-style-type: none"> ◦ Iterativ prosess • Fossefallsmodellen <ul style="list-style-type: none"> ◦ Første velkjente systemutviklingsmodellen ◦ Lineær modell hvor hver aktivitet først starter når forrige er fullført. ◦ Tar ikke hensyn til endringer underveis i utviklingen. ◦ Brukes lite i praksis i dag • Spiralmodellen <ul style="list-style-type: none"> ◦ Oppstod som alternativ til fossefallsmodellen ◦ Risikodrevet modell med ovedvekt på analyse av risiko og prototyping ◦ Oppfordrer til å vurdere alternativer ◦ Hver loop representerer en iterasjon (dvs. En fase i utviklingsprosessen). ◦ Brukere involveres for å identifisere og kontrollere risiko •
Interessent	<ul style="list-style-type: none"> • Alle mennesker eller systemer som påvirkes av vårt system, enten direkte eller indirekte. • Engelsk: Stakeholders
Hvordan generere alternativer	<ul style="list-style-type: none"> • Mennesker liker å forholde seg til et de vet fungerer. • Systemutviklere er opptatt av raskeste, mest effektive løsningen som tilfredsstiller en liste med krav. • Interaksjonsdesignere trent til å vurdere ulike alternativer og inspirere brukeren til å tenke nyskapende og kreativt • Hvordan skape alternativer? <ul style="list-style-type: none"> ◦ Teft og kreativitet: Undersøke og syntetisere <ul style="list-style-type: none"> ▪ Undersøke mange alternativer ▪ Lener seg mot studier og forskning ◦ Søke etter inspirasjon: Studer liknende produkter eller se på helt ulike produkter

Hvordan velge blant alternativer	<ul style="list-style-type: none"> • Evaluering med brukere eller andre designere • Teknisk gjennomførbarhetsanalyse <ul style="list-style-type: none"> ◦ Hva er mulig med teknologien som finnes? • Kvalitets-terskler <ul style="list-style-type: none"> ◦ Brukbarhetsmål kan lede til brukbarhetskrav som kan brukes som kriterier som kan kontrolleres jevnlig <ul style="list-style-type: none"> ▪ Sikkerhet -> hvor trygg er løsningen? ▪ Funksjonalitet -> Hvilke funksjoner er overflødige? ▪ Oppgaveløsning -> Er det nok support og informasjon tilgjengelig? Hvordan er oppgavedekningen? ▪ Prestasjon (Performance) -> Hvordan er prestasjonen til brukerne?
Krav vs. design	<ul style="list-style-type: none"> • Krav <ul style="list-style-type: none"> ◦ Beskrivelse av HVA systemet skal gjøre (eller kvaliteter det skal ha) ◦ Fra en kundes eller klients synsvinkel ◦ Ikke uttrykt med tanke på løsning ◦ Skiller mellom funksjonelle og ikke-funksjonelle krav (Se kapittel 10) • Design <ul style="list-style-type: none"> ◦ En beskrivelse av HVORDAN vi vil implementere en løsning ◦ En modell, prototype eller skisse som imøtekommer kravene ◦ Gjøres før implementeringen slik at de kan evalueres
Verifisering vs Validering	<ul style="list-style-type: none"> • Verifisering <ul style="list-style-type: none"> ◦ Designe produktet riktig (Kontrollerbart) • Validering (Dette er vi opptatt av) <ul style="list-style-type: none"> ◦ Designe det riktige produktet (Ikke kontrollerbart)
Formality gap	<ul style="list-style-type: none"> • Validering vil alltid avhenge av subjektive vurderinger • De fleste projekter starter med uformelle, menneskelige tanker om systemet. <ul style="list-style-type: none"> ◦ «Trykk på knappen, så skjer det noe magisk» • Lager vi en modell over systemet (f.eks. Kravspek) er vi over i et formelt domene. <ul style="list-style-type: none"> ◦ I overgangen hit, fra det uformelle domene, oppstår det en reduksjon av den virkelige verden. Den beskrives med ord, begreper og målinger. • Det vil alltid eksistere et gap mellom virkelighetens krav, og det vi klarer å fange av krav i vår modell <ul style="list-style-type: none"> ◦ Dette gapet beskrives som «Formality gap».
Abstrakte modeller	<ul style="list-style-type: none"> • Vi bruker abstrakte modeller for å hjelpe oss med å knytte beskrevet modell med virkelig modell

Stjernemodellen	<ul style="list-style-type: none"> • Stammer fra empiriske studier av interaksjonsdesigneres arbeidsvaner • Viktige egenskaper <ul style="list-style-type: none"> ◦ Evaluering står i sentrum av alle aktiviteter ◦ Ingen spesiell rekkefølge på aktivitetene, utvikling kan starte hvor som helst ◦ Muliggjør både top-down og bottom-up tilnærmingen kalles ofte for 'yo-yoing" av systemutviklere • Skal en gå fra en aktivitet til en annen, MÅ man gå via en evaluering. <ul style="list-style-type: none"> ◦ Evaluering står i sentrum
Livssyklusmodell: ISO9241-210	<ul style="list-style-type: none"> • ISO-definert modell • Viktige egenskaper <ul style="list-style-type: none"> ◦ Prosessen er drevet og målt gjennom brukersentrert evaluering ◦ Designprosessen tar for seg hele brukeropplevelsen ◦ Designteamet innehar som oftest av tverrfaglig kompetanse ◦ Iterativ prosess
Løk-modellen (Ikke viktig å kunne)	<ul style="list-style-type: none"> • En modell som viser brukere av et produkt i forskjellige lag <ul style="list-style-type: none"> ◦ Produktet ◦ Systemet <ul style="list-style-type: none"> ▪ ◦ Omfattende systemet ◦ Samfunnet
Brukerinnvolvering	<ul style="list-style-type: none"> • Involvering av bruker i designteamet <ul style="list-style-type: none"> ◦ Heltid <ul style="list-style-type: none"> ▪ Konstant input ▪ Mister kontakt med resten av brukerne ◦ Deltid <ul style="list-style-type: none"> ▪ Sporadisk input ▪ Ofte stressende pga. Annen jobb ◦ Kortsiktig <ul style="list-style-type: none"> ▪ Inkonsistent gjennom projektet ◦ Langsiktig <ul style="list-style-type: none"> ▪ Konsistent ▪ Mister kontakt med andre brukere • Bruker kontaktes via nyhetsbrev, internett forum etc. <ul style="list-style-type: none"> ◦ Når bredere publikum -> bør gi bedre grunnlag for brukergruppen ◦ Trenger toveiskommunikasjon • Brukere involvert etter at produktet er sluppet <ul style="list-style-type: none"> ◦ Forhold som ikke var til stede under beta-testing og prototyping, vises frem når løsningen slippes. <ul style="list-style-type: none"> ▪ f.eks. Antallet mennesker som kobler seg til en server. => nettverkproblemer

- Kombinasjon av disse tre tilnærmingene

Kapittel 10: Behov og krav

Behov (needs)	<ul style="list-style-type: none"> • Grunnleggende behov. <ul style="list-style-type: none"> ◦ Har ikke noe med teknologi å gjøre ◦ Ikke noe med spesifikke mennesker å gjøre. • Man kan ha behov for <ul style="list-style-type: none"> ◦ trygghet ◦ forflytning ◦ samhörighet ◦ integritet ◦ næring
Krav (requirements)	<ul style="list-style-type: none"> • Koblet til strategier/løsninger/teknologier/plattformer. • Eksempler på krav: <ul style="list-style-type: none"> ◦ Størrelse ◦ Krav om oppdatering for å få gjennomført en oppgave. ◦ Kunne koble til et head-set med AUX-kabel •
Funksjonelle krav	<ul style="list-style-type: none"> • Hva skal systemet gjøre/kunne? • Funksjoner • Krav rettet direkte mot systemet. <ul style="list-style-type: none"> ◦ AUX-input ◦ Størrelse
Ikke-funksjonelle krav	<ul style="list-style-type: none"> • Krav som ikke er om systemet direkte, men hvordan det oppfører seg • f.eks. Responstid, min/maks tid for kvittering.
Omgivelseskrav	<ul style="list-style-type: none"> • Hvordan omgivelsene ser ut • Lys, fuktighet, risting
Sosiale krav	<ul style="list-style-type: none"> • Hvordan gruppen bruker noe • Kommunikasjon, deling, gruppe
Organisatoriske krav	<ul style="list-style-type: none"> • Krav til et system sett fra en organisasjons synspunkt • Kalibrering, idriftsetting
Hvorfor ha datainnsamling?	<ul style="list-style-type: none"> • Forstå så mye som mulig om brukerne <ul style="list-style-type: none"> ◦ Oppgaver ◦ Situasjoner ◦ Mål ◦ Behov • Etablere stabile krav som kan brukes under utviklingen
Hvordan ha datainnsamling?	<ul style="list-style-type: none"> • Intervju, observasjon, spørreskjema studere dokumentasjon, studere tilsvarende tjenester/produkter • Dataanalyse • Uttrykke krav man finner i analysen • Iterativ prosess – med brukerne/aktørene

Hvorfor etablere krav	<ul style="list-style-type: none"> • Krav etableres ved å forstå brukerens behov i sammenheng • Kravene settes i lys av dataene som er samlet inn
Hvem er brukere?	<ul style="list-style-type: none"> • Personer som påvirkes av systemet • Karakteristika, bakgrunn, holdninger til datamasjoner • Novise eller ekspert? • Sanser – fungerer alle optimalt? • Frekvens av brukbarhet <ul style="list-style-type: none"> ◦ Daglig ◦ Ukentlig ◦ En gang året? • Aktører (Stakeholders) <ul style="list-style-type: none"> ◦ Sluttbruker ◦ Kunder ◦ Innkjøper ◦ De som drifter systemet ◦ Videreutviklere ◦ Reparerer ◦ Vedlikeholdere ◦ Andre aktører?
Persona	<ul style="list-style-type: none"> • Beskrive brukerkarakteristikker <ul style="list-style-type: none"> ◦ Kan gjerne beskrive ekstremer <ul style="list-style-type: none"> ▪ Vil være alene ▪ Veldig sosial ▪ Veldig strukturert, bruker tid effektivt • Fiktive brukere, skapt av reelle brukere • Ikke ideelle • En typisk bruker • Gi liv til personaen <ul style="list-style-type: none"> ◦ Navn ◦ Mål ◦ Bakgrunn ◦ Vær konkret • Ha flere personas
Utfordringer ved datainnsamling	<ul style="list-style-type: none"> • Identifisere aktører • Engasjere aktørene • Reelle brukere vs andre (representanter, "proxy" brukere) • Håndtering av krav <ul style="list-style-type: none"> ◦ behandling av nye krav ◦ Hvem eier kravene • Kommunikasjon mellom partnere <ul style="list-style-type: none"> ◦ Innen utviklingsteam ◦ Mellom utvikler og brukeren ◦ Mellom bruker • Domenekunnskap <ul style="list-style-type: none"> ◦ Trenger du kunnskap om fagområdet du designer en

	<p>løsning for? Må du kunne kirurgi for å kunne håndtere kravene de har til en ny løsning?</p> <ul style="list-style-type: none"> • Politisk innen organisasjon/gruppen • Økonomiske endringer • Teknologiske endringer
Retningslinjer for datainnsamling (Smørbrøddliste)	<ul style="list-style-type: none"> • Fokuserer på aktørens behov • Involvere mer enn én person fra hver gruppe. • Bruke triangulering! • Støtt aktiviteten med props – prototyper og oppgavebeskrivelser • Kjør pilotundersøkelser • Finn ut hvordan analysere dataene som kommer inn • Vurder nøye hvordan data tas opp og lagres <ul style="list-style-type: none"> ◦ Format etc.
Scenario	<ul style="list-style-type: none"> • En uformell historie for å beskrive en oppgave • Spesifikk for en bruker • Ikke generaliser
Use cases	<ul style="list-style-type: none"> • Formelt • Beskriver ulike måter å bruke det spesifikke systemet på. • Forutsetter <ul style="list-style-type: none"> ◦ Interaksjon med systemer ◦ Detaljert forståelse av interaksjon
Oppgaveanalyse	<ul style="list-style-type: none"> • Brukes for å se for seg nye systemer/tjenester • Gjøres for å undersøke eksisterende oppgaver for å informere "nye" • Fokuserer på viktige punkter <ul style="list-style-type: none"> ◦ Hva ønsker brukeren å oppnå? ◦ Hvorfor gjør de det? ◦ Hva gjør de for å få det til? • Hierarkisk oppgaveanalyse er den viktigste teknikken (HTA)
Hierarkisk oppgaveanalyse (HTA)	<ul style="list-style-type: none"> • Bryte ned helhet i deloppgaver • Fokuserer på fysiske og observerbare aktiviteter <ul style="list-style-type: none"> ◦ Også aktiviteter som ikke angår "datamaskinen" • Starte med brukerens mål, og oppgavene som gjøres for å komme dit.

Kapittel 8: Kvalitativ analyse

Analyse	<ul style="list-style-type: none"> • Å systematisk gå inn i datamaterialet for å undersøke og identifisere bruddstykker, relasjoner, mønstre. Ut i fra dette finne ut av hvordan systemet oppfører seg. • Handler om å organisere og presentere de viktigste detaljene i dataen. • Ikke gjenta dataen, analyser den! • Analysedataen må organiseres på en slik måte at vi kan bruke den i ny eller videre analyse. Ellers må vi trekke
---------	--

	slutninger om fenomenet vi undersøker. <ul style="list-style-type: none"> Gjennom analyse kan vi hevde funn i dataen vår -> viktig med systematisk og vitenskapelig gyldig tilnærming. Vi skiller mellom kvalitativ og kvantitativ analyse <ul style="list-style-type: none"> I inf1500 er det fokus på kvantitativ 	
Hvorfor ønsker vi å analysere?	<ul style="list-style-type: none"> Innenfor HCI er vi opptatt av å forstå brukere og deres interaksjon med teknologi For å designe gode løsninger trenger vi ofte innsikt <ul style="list-style-type: none"> Handlinger Aktiviteter Opplevelser Ferdigheter Omgivelser Verdier <p>Data ----- Analyse -----> Funn</p>	
Når kan vi analysere	<ul style="list-style-type: none"> Vi kan analysere når som helst, så lenge vi har tilgang til data som kan analyseres Det kan være lurt å analysere tidlig i en prosess for å oppdage <ul style="list-style-type: none"> Fenomener Forholde Relasjoner etc. Analyse kan brukes til å finne ne problemstillinger, samt svare på eksisterende problemstillinger Analyse gjøres ofte i forbindelse med evaluering. Men kan også gjøres i forbindelse med de aller fleste aktiviteter 	
Hva kan vi analysere?	<ul style="list-style-type: none"> Det meste kan analyseres, men det betyr ikke at dataen er god. Eks på analyserbar data <ul style="list-style-type: none"> Intervjudata (Transkribering) Observasjonsnotater Film og bilder Evalueringsdata Dokumenter, dagbøker, aviser Oppgavebeskrivelse 	
Hvor kommer analysemetoder fra?	<ul style="list-style-type: none"> Stammer fra andre vitenskapelige disipliner. HCI er et såpass ungt felt, at det ikke finnes noen analysemetoder som er spesielle for HCI. 	
Kvalitative vs kvantitative studier	Kvalitative studier	Kvantitative studier
	Subjektiv tolkning	Objektiv sannhet
	Holistisk (helhetlig)	Reduksjonistisk (Identifisere et sett med variabler)
	Deskriptiv	Eksperimentell (setter opp hypotese, og tester den)

	Naturlig (fra virkeligheten)	Påtatt (skapt, reproduisert virkelighet)
	Små datautvalg (intervjuer, observasjoner etc.)	Representative utvalg
	Som oftest ingen ønsker å generalisere	Som oftest sterke ønsker om å generalisere
Viktige underliggende antakelser ved kvalitative studier	<ul style="list-style-type: none"> • Involverer flere perspektiver, inkludert deltakernes egen stemme • Flere virkeligheter i alle type studier, blant annet forskerens, deltakerens og leserens. <ul style="list-style-type: none"> ◦ Disse kan se på de samme dataene, men tolke de på en annen måte => forskjellige funn4 <ul style="list-style-type: none"> ▪ Ser ikke på det samme som viktigste data • Studien er bundet i kontekst. • Jobber nedenfra og opp, og snakker om spesifikke observasjoner • Formålet er å oppdage mønsteret eller teorier som kan hjelpe med å forklare et fenomen • Nøyaktighet kommer fra verifisering gjennom å <ul style="list-style-type: none"> ◦ Bekrefte med deltakere at informasjonen er korrekt ◦ Triangulere ulike former for metoder eller datakilder • 	
Kvalitativ analyse	<ul style="list-style-type: none"> • 	
Syntese	<ul style="list-style-type: none"> • Å sette sammen bruddstykker til en helhet <ul style="list-style-type: none"> ◦ Lager mange muligheter som oppfyller kravene, og finner den beste. ◦ Eks: Lage mange applikasjoner, og gi de til en rekke brukere. De forteller så hvilken de likte best. Så ser de hvordan de brukte den, og lærer ut i fra det. 	
Utfordringer med kvalitativ analyse	<ul style="list-style-type: none"> • Vanskelig å generalisere funn -> ikke noe mål • Ofte flere "sannheter". Disse er subjektive, og ikke objektive. Kan ikke måles nøyaktig • Bærer ofte preg av analytikers subjektivitet eller andre former for bias. 	
Bias	<ul style="list-style-type: none"> • Skjevhet i dataen som gjør at den ikke stemmer med virkeligheten <ul style="list-style-type: none"> ◦ Kan medføre feil med registrering, tolkning, forståelse 	
Inter- og intra-coder reliability	<ul style="list-style-type: none"> • Intra-coder reliability <ul style="list-style-type: none"> ◦ Vil samme person som analysere samme stykke data likt to ganger etter hverandre? • Inter-coder reliability <ul style="list-style-type: none"> ◦ Vil en annen uavhengig person kunne komme frem til samme analyse som deg? • Disse to sporene hjelper til med å bygge sporbarhet og troverdighet 	

	<ul style="list-style-type: none"> ○ Øker sjansen for å forstå brukeren <ul style="list-style-type: none"> ▪ Øker sjansen for å designe en brukbar løsning
Hawthorne-effekten	<ul style="list-style-type: none"> • Observasjonsbias • Det å bli undersøkt i seg selv frembringer atferdsendringer hos den som blir undersøkt • Enklere definisjon: Mennesker oppfører seg annerledes dersom de vet de blir observert. • Kan skyldes flere forhold, men som regel er det fordi den som undersøkes har et ønske om å komme bedre ut av undersøkelsen. • Kan være vanskelig for observatøren å oppdage.
Hva kan vi undersøke med kvalitativ analyse?	<ul style="list-style-type: none"> • Hvordan vi gjør analysen avhenger som regel av typen svar vi er på jakt etter • Eksempler på fenomener eller forhold vi kan undersøke <ul style="list-style-type: none"> ○ Hvorfor mennesker oppfører seg som de gjør ○ Hvordan menneskers meninger og holdninger formes ○ Hvordan mennesker påvirkes av sine omstendigheter ○ Hvordan og hvorfor menneskers kultur utvikler seg ○ Hvilke forskjeller som finnes i mellommenneskelige sosiale grupperinger ○ Utlede beov og krav ○ Utforske og identifisere ulike konsepter eller perspektiver ○ Utforske hvor godt noe passer inn i systemet ○ Forstå ekte fenomener <ul style="list-style-type: none"> ▪ Virkelige, unike omstendigheter som ikke lar seg reproducere i en lab ○ Sensitive forhold der det er behov for fleksibilitet for å ung ukomfortable situasjoner.
Tilnærminger til kvalitativ analyse	<ul style="list-style-type: none"> • Det finnes mange ulike tilnærminger <ul style="list-style-type: none"> ○ Innholdsanalyse <ul style="list-style-type: none"> ▪ Denne skal vi fokusere mest på ○ Diskursanalyse ○ Konversasjons-analyse ○ Narrativ analyse • Eksempelene over forteller ingenting om metoder, teknikker eller verktøy som blir anvendt under analysen • Viktig å være presis i beskrivelse av prosedyre og gjennomføring
Innholdsanalyse , grunnleggende og fortolkende	<ul style="list-style-type: none"> • Kan ha to betydninger <ul style="list-style-type: none"> ○ Et generelt begrep som brukes for å angi at man analyserer innholdet i en transkripsjon ○ Et konkret begrep som beskriver en teknikk som involverer av vi teller forekomsten av noe <ul style="list-style-type: none"> ▪ ord ▪ uttrykk ▪ konsepter etc.

	<ul style="list-style-type: none"> • To nivåer av innholdsanalyse <ul style="list-style-type: none"> ◦ Grunnleggende <ul style="list-style-type: none"> ▪ Deskriptiv og manifesterende gjengivelse av det som ble sagt eller gjort ◦ Fortolkende <ul style="list-style-type: none"> ▪ Latent og relasjonell tolkning av det bakenforliggende for det som ble sagt eller gjort • Inkluderer kategorisering av data • Involverer som regel at vi koder og klassifiserer dataen • Enkel fremgangsmåte: <ol style="list-style-type: none"> 1. Les transkribering og marker ut interessant og relevant informasjon (utsagn, observasjoner) 2. Lag en liste over de ulike typene informasjon du har markert 3. Kategoriser elementene i listen 4. Identifiser relasjoner, mønstre, sammenhenger (temaer) 5. Sammenlign likheter og ulikheter mellom kategoriene
Koding	<ul style="list-style-type: none"> • Prosess som kombinerer ulike deler av dataen • Kombinere data basert på tema, ideer, nøkkelord og kategorier • Fargekoding • Som regel brukt på tekstutdrag, men kan også brukes på lyd, video eller bildeutdrag
Åpen koding	<ul style="list-style-type: none"> • Første nivå av koding <ul style="list-style-type: none"> ◦ Linje for linje etter distinkte kategorier • Hva handler dette egentlig om? • Hva refereres til her? • Involverer typisk å bryte opp dataen i generelle hovedkategorier med ett nivå av underkategorier • Resultater ofte i 3-5 ulike farger som viser til hver sin hovedkategori
Induktiv analyse	<ul style="list-style-type: none"> • Vi går fra spesifikke og konkrete observasjoner til brede generaliseringer <ul style="list-style-type: none"> ◦ Praksis danner grunnlag for teori • Praksis danner grunnlag for teorier
Aksiell koding	<ul style="list-style-type: none"> • Andre nivå av koding som gjøres etter man har utledet kategorier eller konsepter gjennom åpen koding. • Sentrale spørsmål <ul style="list-style-type: none"> ◦ Hvordan henger de ulike kategoriene sammen? ◦ Hva forårsaket kategoriene og hva er konsekvensene? • Tar utgangspunkt i kategoriene som ble utledet gjennom åpen koding, og leser gjennom teksten på nytt, én akse (kategori) om gangen • kategoriene undersøkes enda nærmere for å avdekke <ul style="list-style-type: none"> ◦ Relasjoner mellom ulike kategorier ◦ Relasjoner mellom hovedkategorier og underkategorier

	<ul style="list-style-type: none"> ◦ Andre viktige forhold for å forstå kategorien <ul style="list-style-type: none"> ▪ Betingelser, kontekster, handlinger, strategier, konsekvenser
Deduktiv analyse	<ul style="list-style-type: none"> • Undersøker om våre generelle antakelser også gjelder for konkrete observasjoner. <ul style="list-style-type: none"> ◦ Teori sjekkes mot praksis
Tematisk analyse	<ul style="list-style-type: none"> • Metode som identifiserer, analyserer og rapporterer mønstre (temaer) i et datasett • Krever at vi har en idé om hvilket fenomen eller forhold vi ønsker å undersøke. • Et tema kan defineres som momenter i dataen som <ol style="list-style-type: none"> 1. Er viktige i forhold til problemstillingen 2. Representerer en grad av mønster i datasettet • Forhåndsdefinert prosedyre bestående av seks faser <ol style="list-style-type: none"> 1. Bli kjent med datasettet 2. Generer innledende koder 3. Søk etter temaer 4. Gjennomgå temaene 5. Definer og navngi temaene 6. Produser en rapport
Affinity diagram	<ul style="list-style-type: none"> • Teknikk for å systematisere et mangfold av innsikter og ideer • Individuelle innsikter og ideer organiseres i et hierarki som viser felles strukturer og temaer • Standard prosedyre for å gjøre affinity diagramming <ol style="list-style-type: none"> 1. Går ut på å notere alle innsikter og ideer på kort eller lapper (post-it) 2. Ser etter ideer som ser ut til å høre sammen. 3. Sorterer kortene/lappene i grupper til alle lappene er brukt
Funn	<ul style="list-style-type: none"> • Relasjoner, mønstre, sammenhenger (temaer) vi har oppdaget gjennom analysen • VIKTIG: å kalle noe for et funn krever at vi har gjort en analyse. Holder ikke å synse, vi må faktisk undersøke. • Kan presenteres som <ul style="list-style-type: none"> ◦ Liste over funn ◦ Tabell ◦ Grafisk fremstilling av data. ◦ Diagrammer (f.eks. Gigamaps, customer journey maps etc.) • Funn kan forsterkes ved å vise til <ul style="list-style-type: none"> ◦ Bilder, videoer lyd og andre opptak ◦ Sitater og utsagn ◦ Historier og eksempler

Kapittel 6: Tjenestedesign (Service design)

Tjenestedesign (service	<ul style="list-style-type: none"> • Handler om å skape nye opplevelser
-------------------------	--

design)

- for å gi **bedre kundeopplevelser**
- resulterer i **høyere avkastning**
- kanskje **reduserte kostnader**
- Eksempel om kaffe
 - (Rå)vare – Kaffebønner
 - Produkt – Ferdig kvernet kaffe
 - Tjeneste – nytraktet kaffe
 - Opplevelse – Kaféstemning
- Markedet har beveget seg fra kjøp av produkter til kjøp av tjenester
 - Månedss-abonnementer
- **Handler om design av hele tjenesten**
 - Designet en flytur, ikke kun fra du går på flyet, men fra du drar hjemme fra. Kanskje fra du kjøper billetten? Kanskje fra du ser en reklame for tjenesten?
 - Når slutter tjenesten? Når du lander, eller LENGE etter du kommer hjem?
- Utvidelse av designprosessen
 1. Vi kan bygge en pen og smart telefon
 2. Vi kan designe en telefon som løser en rekke oppgaver (/prosesser) og som tilpasser seg omringende systemer.
 3. Vi kan lage en miljøvennlig og resirkulerbar telefon, og vi kan samtidig velge en løsning basert på åpen programvare.
 4. Disse tre punktene er tre forskjellige nivåer av en tjeneste vi kan designe. Hvor stor opplevelse ønsker vi å skape?
- **Hva er en tjeneste?**
 - **Kundereise (Customer journey)**
 - Flere sammenhengende aktiviteter som former en prosess og har verdi for brukerens
 - **Brukeropplevelse (User experience)**
 - Tjenester påvirker oss i det daglige liv
 - Tjenesteskisse (Service blueprint)
 - Elementer fra andre designdisipliner
 - Design av systemer
 - **Touchpoints**
 - Fokuserer på hele systemet i bruk (tenke på hele kundereisen, ikke bare et lite sted der det er interaksjon)
- **Egenskaper ved tjenester**
 - Tjenester mer "intangible" enn produkter
 - Vanskelig å måle, forstå, vanskelig å telle ant. Interaksjoner
 - Opplevelser
 - Handler om følelser
 - Kombineres ofte med eget eid utstyr
 - For å bruke Netflix må du ha egen telefoner
 - Internett på flyplassen krever eget "device".
 - Vanskelig å prototype
 - Manifesteres ikke før noen bruker de.
 - Krever at vi lager **scenarioer** som utføres f.eks. Med

	<p>rollespill</p> <ul style="list-style-type: none"> ◦ Involverer mye menneskelig aktivitet => vanskelig å skalere • Tjenstedesign fokuserer på en brukers interaksjon ved flere touchpoints over en periode med tid
Touchpoints	<ul style="list-style-type: none"> • En situasjon der en tjeneste tilbys eller informasjon utveksles <ul style="list-style-type: none"> ◦ Steder, produkter, prosesser og mennesker ◦ Objekter, fysiske steder, nettsider, kundeservice etc. • Er et bindeledd mellom bruker og selskapet. • Sted <ul style="list-style-type: none"> ◦ Stedet (settingen) der tjenesten leveres ◦ Viktig å tenke på hvor (miljø og omgivelser) ◦ Brukeren trenger hint for å henge med -> Skilt, informasjonsskranker, tavler etc. ◦ Eksempler: <ul style="list-style-type: none"> ▪ Telefonkiosk ▪ Skilting ▪ Bergogdalbane • Produkter <ul style="list-style-type: none"> ◦ Artefakten som tilbyr interaksjonen (f.eks den fysiske gjenstanden) ◦ Befinner seg i og utfyller omgivelsene (sted) ◦ Gir brukeren en mulighet til å delta (engasjement) ◦ Eksempler <ul style="list-style-type: none"> ▪ Informasjonspaneler på festivaler (Må stå inne på området) ▪ Vibrasjon/lyd når maten er klar (må være på restauranten) • Prosesser <ul style="list-style-type: none"> ◦ Beskriver hvordan en tjeneste bestilles, lages og leveres til brukeren ◦ Ofte mange veier (prosesser) gjennom en tjeneste. ◦ Gir holistisk overblikk over hele tjenesten med alle involverte elementer. ◦ En prosess kan være et touchpoint <ul style="list-style-type: none"> ▪ MEN det kan også deles opp dersom det er såpass komplisert, slik at det er hensiktsmessig å dele det opp. ◦ Eksempler <ul style="list-style-type: none"> ▪ Starbucks og navn ▪ Sporing av pakker • Mennesker <ul style="list-style-type: none"> ◦ Mennesker kommer til live gjennom tjenester og det er de som er koreografene ◦ Brukere og ansatte utfører ulike deler av tjenester for å oppnå et ønsket resultat. <ul style="list-style-type: none"> ▪ Sitter det en person å skriker i en kinosal, ødelegger det vår opplevelse av tjenesten ◦ Sitter igjen med personlige opplevelser og inntrykk av tjenesten

	<ul style="list-style-type: none"> ◦ Eksempler <ul style="list-style-type: none"> ▪ Foretrekker å gå til kjeder der ansatte har god kunnskap om produktene ▪ Dersom toget ikke kommer, viktig å ha noen som kan vise deg vei.
Hvordan jobber man med Tjenstedesign?	<ul style="list-style-type: none"> • Skiller seg fra andre disipliner => Tjenesteorientert • Visualisering av hele tjenestens levetid <ul style="list-style-type: none"> ◦ Rollespill (Utspilling av scenário, forventede situasjoner) ◦ Improvisasjon (oppdage nye utfordringer) ◦ Stakeholder map ◦ Customer Journey Map • Prototyping av løsninger må inkludere mer enn bare produkter <ul style="list-style-type: none"> ◦ Evidencing ◦ Experience prototyping ◦ Videoprototyping • Øvrige metoder <ul style="list-style-type: none"> ◦ Storytelling ◦ Workshops ◦ Observasjon ◦ Cultural probes
Experience prototyping	<ul style="list-style-type: none"> • En måte å prototype på <ul style="list-style-type: none"> ◦ Opptatt av opplevelsen rundt bruk. • Setter oss inn i rollen som brukeren <ul style="list-style-type: none"> ◦ Dette kan være vanskelig. <ul style="list-style-type: none"> ▪ Få inn rollespillere ▪ Bruk brukere fra målgruppen <ul style="list-style-type: none"> • Dersom vi ikke kan sette oss inn i situasjonen deres • Hvilke kantområder eller ekstremiteter kan brukeren havne i? • Gir noen en opplevelse, og undersøker hvordan de oppfører seg, og hvordan de har det. (følelser) • Teknikker <ul style="list-style-type: none"> ◦ Rollespill ◦ Service safari ◦ Assosiasjonsøvelser ◦ Dommedagsscenario etc. • Eksempler på oppgaver <ul style="list-style-type: none"> ◦ Gi noen på et tog oppgaver «Vær sulten, finn mat».
"Fremtidsarkeologi"	<ul style="list-style-type: none"> • Når vi prøver å se hva som kan være nyttig for fremtiden. • Eksempel <ul style="list-style-type: none"> ◦ App for å finne reise ved å swipe (Tidner liknende) ◦ App for å finne leger
Customer Journey Map (Brukerreise – ikke mye brukt)	<ul style="list-style-type: none"> • Ingen standard for hvordan dette skal gjøres, men greit å være oversiktlig. Hensikten er å orientere om kundens opplevelse av tjenesten • Metode for å visualisere kundenes reise gjennom en tjeneste. <ul style="list-style-type: none"> ◦ Visualiserer touchpointene (stegene) som inngår i tjenesten.

	<ul style="list-style-type: none"> ○ Abstrakt, mindre detaljert modell enn systemskisser ○ Inkluderer andre aspekter (humør, forventninger, objekter etc.) • Tidsaspekt viktig <ul style="list-style-type: none"> ○ Begynner/slutter ofte lenge før/etter bruk av tjenesten.
Utfordringer ved tjenestededesign anno 2016	<ol style="list-style-type: none"> 1. Kunder er mer opptatt av merkenavnet enn mediet tjenesten leveres i (Apple) 2. Touchpoint design vil ikke være nok til å gjøre en forskjell. <ul style="list-style-type: none"> • UX er blitt for allment – langsiktige og stabile kundeforhold er forskjellen 3. Holistiske opplevelser krever holistiske organisasjoner <ul style="list-style-type: none"> • Conways lov «Organisasjoner som designer systemer ... er begrenset til å produsere systemer som er kopier av den kommunikasjonsstrategien de selv innehar i sin bedrift» 4. Kunder vil tilegne seg mer flytende forventninger <ul style="list-style-type: none"> • Kunder vil forflytte seg fra en app til en annen, MEN de tar med seg forventningene på tvers av systemene 5. Design av opplevelser for ansatte vil bli en prioritet <ul style="list-style-type: none"> • Økt fokus på hvordan tjenester leveres internt <ul style="list-style-type: none"> ○ Ansatte kan betjene kunder enda bedre. 6. Bedrifter vil bygge ut kapasitet for tjenestededesign <ul style="list-style-type: none"> • Eks. Gi kundeservice tilgang til å se nettet i huset ditt. • Gi ansatte verktøy, metoder og praksis, for å kunne yte best mulig service.
Responsive Web Design	<ul style="list-style-type: none"> • Internett er tilgjengelig på mange plattformer • Handler om å utforme nettsider slik at de gir best mulig brukeropplevelse – uansett enhet. • Nettsiden skal utformes slik at bruker ikke trenger å zoome, rotere, skalere etc. Unødvendig. • Innebygget støtte i HTML og CSS for å tilpasse utformingen • Nå designes nettsider først for mobil, så tablet så pc.
Attention span	<ul style="list-style-type: none"> • Vi er forventningsfull og utålmodige • Vet hva vi er på jakt etter, og ønsker ikke å lete • Det er mye vi ikke "gidder" å lese
Steve Krugs 10 prinsipper for godt design	<ol style="list-style-type: none"> 1. Brukbarhet betyr at det fungerer godt og ikke frustrerer. 2. Nettsider bør være selvforklarende og åpenbare 3. Ikke få meg til å tenke. 4. Ikke søls med tiden min. 5. Brukere bruker fortsatt backspace 6. Vi er vanedyr – holder oss til det vi får til. 7. Ikke tid til smalltalk – ta vekk støy og koseprat 8. Ikke ta vekk søkeboksen 9. Vi former mentale kart over nettsider 10. Gjør det lett å gå hjem
Web 2.0	<ul style="list-style-type: none"> • Markerer en overgang i webdesign • Gikk fra statisk enveiskommunikasjon til nettsider med dialog <ul style="list-style-type: none"> ○ Mellom de som skaper og leser innhold ○ Mellom ulike brukere.

	<ul style="list-style-type: none"> ◦ Diskusjoner. • Overgang til mer oppfordret deling, brukerdeltakelse, samarbeid etc. • Trenden la grobunn for blogging og sosiale medier.
Identitet	<ul style="list-style-type: none"> • Personvern og datalagring har blitt aktuelle temaer.

Kapittel 11: Design, prototyping og konstruksjon

Prototype	<ul style="list-style-type: none"> • Hva er en prototype? <ul style="list-style-type: none"> ◦ Hjelper oss med å forstå egenskaper og kvaliteter ved den endelige løsninger • Hvordan prototype? <ul style="list-style-type: none"> ◦ To muligheter <ul style="list-style-type: none"> ▪ Begynne fra scratch ▪ Modifisere en eksisterende idé ◦ Serie med skisser ◦ Storyboard ◦ Lysark ◦ Video – som simulerer bruk av systemet ◦ Fysisk modell. ◦ Pappmodell ◦ Programkode som viser en begrenset funksjonalitet ◦ m.m. • Hvorfor prototype? <ul style="list-style-type: none"> ◦ Gi mulighet til å vise frem og teste ideer ◦ Evaluering og tilbakemelding fra bruker ◦ Stimulerer til refleksjon ◦ Prototyper svarer på spørsmål som støtter designere i å velge mellom alternativer. ◦ Utviklere og ikke-tekniske aktører kan snakke samme språk ◦ Utvikler og interaksjonsdesigner tenker ikke likt. Prototyping kan føre til bedre kommunikasjon, også her. • Hva kan prototypes? <ul style="list-style-type: none"> ◦ Tekniske forhold ◦ Arbeidsflyt, oppgavedesign ◦ Skjermlayout, informasjonsvisning ◦ Interaksjonsforhold ◦ Vanskelige, kontroversielle, kritiske områder
Konseptuelt design	<ul style="list-style-type: none"> • Ta kravene vi har etablert etter datainnsamling, og overføre de til en beskrivelse av et foreslått system. <ul style="list-style-type: none"> ◦ Gjøres ved hjelp av integrerte ideer og konsepter om <ul style="list-style-type: none"> ▪ Hva systemet skal gjøre ▪ Hvordan det skal oppføre seg ▪ Hvordan det skal se ut • krav -> konseptuelt design -> løsning (Det er lett å hoppe rett til løsningen) <ul style="list-style-type: none"> ◦ Gjenta, gjenta, gjenta.

	<ul style="list-style-type: none"> ▪ Pass på å få med brukeren • Er en fase før man begynner å bygge noe. Dette er kun idéer. • Utvikling av den konseptuelle modellen <ul style="list-style-type: none"> ◦ Hvilke funksjoner skal produktet gi? <ul style="list-style-type: none"> ▪ Hva skal produktet og brukerne gjøre? (Oppgaveallokering) ◦ Hvordan er funksjonene relatert til hverandre? <ul style="list-style-type: none"> ▪ Sekvensiell eller parallell? ▪ Kategorisering ◦ Hva slags informasjon skal være tilgjengelig? <ul style="list-style-type: none"> ▪ Hvilke data trengs for å gjennomføre oppgavene? ▪ Hvordan blir data transformert i systemet? • Bruk av scenarier <ul style="list-style-type: none"> ◦ Uttrykke foreslåtte eller tenkte situasjoner ◦ Benytt "ekstreme" scenarier for å utvide designområdet ◦ Brukes på ulike måter i design <ul style="list-style-type: none"> ▪ Script som kan brukes under evaluering av prototyper ▪ Konkrete eksempler på oppgaver ▪ Negative scenarier (minus-scenarier) for å utlede ikke-funksjonelle krav.
Grensesnittmetaforer	<ul style="list-style-type: none"> • Kombinerer kjent kunnskap med ny kunnskap på en slik måte at det hjelper brukeren med å forstå løsningen. <ul style="list-style-type: none"> ◦ Min def: <ul style="list-style-type: none"> ▪ Grensesnitt som designes på en slik måte at brukeren kjenner seg i gjen, og vet hvordan det fungerer, kun ved å se på det. • Tre steg for å velge metafor <ol style="list-style-type: none"> 1. Forstå systemets funksjonalitet <ul style="list-style-type: none"> • Hva skal systemet gjøre? 2. Identifisere potensielle problemområder 3. Generer metafor på bakgrunn av dette • Fem spørsmål for å evaluere metaforer <ol style="list-style-type: none"> 1. Hvilken struktur tilhører metaforen? 2. Hvor relevant er metaforen for problemet som skal løses? 3. Er metaforen enkel å representere? 4. Vil brukeren forstå metaforen? 5. Hvor skalerbar er metaforen? • Eksempel: <ul style="list-style-type: none"> ◦ Bokhylle – Bøkene vises på skjermen med fronten frem, stående på hyller. Her kjenner personen seg igjen og skjønner at det er en bokhylle (forhåpentligvis).
Dimensjoner (ved prototyping)	<ul style="list-style-type: none"> • Et aspekt vi ønsker å fokusere på når vi prototyper <ul style="list-style-type: none"> ◦ Utseende – Størrelse, Farge, form, fasong, tekstur, proposjon, hardhet, gjennomsiktighet, haptisk, lyd ◦ Data – Størrelse, type, bruk, personvern, hierarki, organisering ◦ Funksjonalitet – Systemets funksjoner og brukerens behov ◦ Interaktivitet – Input, output, feedback, informasjon ◦ Romlig struktur – Sammensetning av grensesnitt og informasjonssystemer, relasjon mellom elementer, 2D/3D, tangible eller intangible

	<ul style="list-style-type: none"> ◦ Materiale – Fysisk medium (papir, tre, plastikk), utstyr for å forme fysisk medium (kniv, saks, penn, sandpapir), programvareutstyr (Adobe Flash, Visual basic), fysisk utstyr (Phigets, Arduino). ◦ Oppløsning (fidelity) – Nøyaktighet, utseende, interaktive detaljer, realistisk data vs dummy data ◦ Scope – Grad av kontekstualisering
Low-fidelity prototyper	<ul style="list-style-type: none"> • Lavoppløslige prototyper => Prototyper med få detaljer (lite virkelighetsnær) • Fokus på hovedinteraksjoner • Bruker et annet medium enn det endelige produktet • Innbyr til utforskning • Nesten all interaksjon kan fakes • «Quick and dirty» • Eksempler på low-fi prototyper <ul style="list-style-type: none"> ◦ Storyboard ◦ Skisser ◦ Kortbaserte prototyper ◦ "Wizard of Oz" ◦ Wireframing
Storyboards	<ul style="list-style-type: none"> • Serie med skisser som viser hvordan brukeren går gjennom ulike steg i oppgaven. • "Tegneserie" med 3 – 5 bilder som beskriver oppgaven en person løser • Veldig lite detaljert • Fokus på de viktigste interaksjonene • Ofte brukt sammen med scenarier
Skisse	<ul style="list-style-type: none"> • Tegning av grensesnittets utseende, følelse eller funksjonalitet • Raskt og enkelt å igangsette => Billig og greit • Bruk enkle symboler -> tegneferdigheter ikke pålagt.
Kortbaserte prototyper	<ul style="list-style-type: none"> • Indexkort (ca: 7,5 x 12,5 cm) • Hvert kort representerer en skjerm, eller del av en skjerm • Ofte brukt for utvikling av nettsider
Wizard of Oz	<ul style="list-style-type: none"> • En måte å simulere deler av systemet for å gi brukeren en mer realistisk opplevelse • Brukeren tror det er ekte, men vi sitter å styrer systemet uten at de vet det
High-fidelity	<ul style="list-style-type: none"> • Høyoppløslig prototype. Nesten helt ferdig prototype (som alpha og beta-testing innen software) • Fokus på helheten • Er såpass høyoppløslig at den lar oss teste tekniske forhold • FARE: Bruker kan tro de har et "ferdig system" -> forventningskontroll • Bruker materiale som kan forventes av endelig produkt. • Kostbar og tidkrevende affære • konkrete ideer.

	<ul style="list-style-type: none"> Selvforklarende hva løsningen skal gjøre 		
Low-fi vs high-fi prototyper	Fidelity Low	Fordeler <ul style="list-style-type: none"> Lavere kostnad Kan evaluere flere designalternativ Nyttig kommunikasjonskanal 	Ulemper <ul style="list-style-type: none"> Begrenset feilsjekking Drevet av tilrettelegginger Begrenset bruk i senere faser av design
	High	<ul style="list-style-type: none"> Fullt funksjonell og interaktiv Brukerdrevet Ser og føles ut som endelig produkt 	<ul style="list-style-type: none"> Dyr og tidkrevende å utvikle Ikke effektiv for kravinnsamling Sensitiv for feil under testing
Kompromisser	<ul style="list-style-type: none"> Må ikke ignoreres Horisontale og vertikale prototyper 		
Horisontal prototype	<ul style="list-style-type: none"> Tilfører bredt spekter av funksjoner Hver funksjon har lite detalj/dybde 		
Vertikal prototype	<ul style="list-style-type: none"> Tilfører få funksjoner Hver funksjon har mye detalj/dybde 		
Konstruksjon	<ul style="list-style-type: none"> Lære fra evalueringen av prototypene -> skape en helhet Oppmerksomhet rettes mot <ul style="list-style-type: none"> Brukbarhet Robusthet Vedlikeholdbarhet Portabilitet Effektivitet (Efficiency, effectiveness) 		
Evolusjonær prototype	<ul style="list-style-type: none"> Prototyper som utvikles mot et endelig produkt 		
"lag og kast" prototype	<ul style="list-style-type: none"> Prototypen kastes og endelig produkt bygges fra scratch 		

Kapittel 2: Designprinsipper

Persepsjon	<ul style="list-style-type: none"> Vår oppfatning av omverden slik vi forstår den når sanseorganer kommer i kontakt med omgivelsene Dette er opplevelsen av det vi tror skjer. <ul style="list-style-type: none"> Det vi tror vi ser/hører etc. Er ikke sikkert det er sant. Vår persepsjon påvirkes av <ul style="list-style-type: none"> Fortiden -> erfaring Nåtiden -> Gjeldene omgivelser Fremtiden -> våre mål Vi kan «prime» vårt perseptuelle system. Dette gjør at noe kan bli mer fremtredende enn annet. Idrettsutøvere kan være gode på å få med seg noe ingen andre ser/hører/føler. <ul style="list-style-type: none"> f.eks rød bil eller ball. Da kan det hende du ikke ser gorillaen.
------------	---

Perseptuell påvirkning: fortid	<ul style="list-style-type: none"> • Vi kan utvikle perseptuelle mønstre. Dette skjer gjennom gjentatt eksponering for en situasjon i kjente omgivelser. <ul style="list-style-type: none"> ◦ f.eks. Se kontorpulten lik over lengre tid. Da kan vi få problemer når noe endres. ◦ Vi kan tro feil om en situasjon, fordi vi kan blande virkelighet med tidligere opplevde, liknende opplevelser. <ul style="list-style-type: none"> ▪ Du han låst døren så mange ganger at du er usikker på om du gjorde det i dag, når du tenker tilbake. ◦ Dersom vi designer noe som ikke er kjent for brukeren kan dette skape trøbbel.
Habituation	<ul style="list-style-type: none"> • Fortidspåvirkning av vårt perseptuelle system. • Eksponering for like persepsjoner gjentatte ganger, reduserer vår perseptuelle sensitivitet ovenfor dem. <ul style="list-style-type: none"> ◦ Dersom du skal slette 500 bilder, og du må gjøre det ett etter ett. Den 360'ende gangen dataen spør om du er sikker på om du vil slette bildet, tenker du ikke like mye gjennom om du vil slette det, som de 10 første. • Foregår på et lavt nivå i nervesystemet vårt (nevrontalt nivå)
Attentional blink	<ul style="list-style-type: none"> • Lavnivå-eksemplen på hvordan fortid påvirker persepsjon • Umiddelbart etter vi ser/hører noe (0.15 – 0.45 sek) er vi blinde for annen stimuli (selv om syn og hørsel er operativt) • Hjernen jobber med å knytte persepsjonen mot tidligere erfaringer og er derfor "opptatt"
McGurk-effekten	<ul style="list-style-type: none"> • Når flere sanser er involvert persepsjon samtidig får vi en multimodal persepsjon. <ul style="list-style-type: none"> ◦ Når informasjonen fra disse sansene ikke er i samsvar med hverandre vil den ene overstyre den andre. <ul style="list-style-type: none"> ▪ Video av mann som sier «Baaa» og ikke «Faaa». Munnbevegelsen er annerledes, men lyden er den samme. Plutselig hører vi noe helt annet.
Perseptuell påvirkning: Fremtid	<ul style="list-style-type: none"> • Guiding og filtrering • Vi kan «Prime» persepsjonen vår til å være på utkikk etter en spesifikk gjenstand. <ul style="list-style-type: none"> ◦ f.eks rød bil eller ball. Da kan det hende du ikke ser gorillaen.
Guiding	<ul style="list-style-type: none"> • Våre målsettinger veileder vårt perseptuelle apparat, så vi velger det vi trenger fra verden rundt oss for å nå målene våre. <ul style="list-style-type: none"> ◦ Vi velger det vi trenger for å nå målene, men vi filtrerer også vekk det vi ikke har behov for. Unødvendig informasjon forsvinner.
Filtrering	<ul style="list-style-type: none"> • Målsettingene våre filtrerer oppfatningene våre: ting som ikke er relatert til målene våre blir ubevisst filtrert ut => vårt bevisste sinn registrerer det aldri
Oppmerksomhet	<ul style="list-style-type: none"> • Bevegelse, spesielt i nærheten av oss <ul style="list-style-type: none"> ◦ Bil i motsatt kjøreretning, noen som hopper mot oss • Trusler <ul style="list-style-type: none"> ◦ Alt som signaliserer fare for oss får vår oppmerksomhet • Ansikter <ul style="list-style-type: none"> ◦ Fra fødsel er vi vant til å legge merke til ansikter

	<ul style="list-style-type: none"> • Behov <ul style="list-style-type: none"> ◦ Sex og mat (selv om behovene er møtt)
Prinsipper	<ul style="list-style-type: none"> • Abstrakte designregler
Retningslinjer	<ul style="list-style-type: none"> • Råd om hvordan man kan oppnå prinsipper • Ikke et must å følge, men kan lønne seg.
Standarder	<ul style="list-style-type: none"> • Konkrete, målbare regler. Gjerne ISO definerte
Designprinsipper	<ul style="list-style-type: none"> • Beskrivelse av hvordan vi bør utforme et design <ul style="list-style-type: none"> ◦ Gir en idé om hva som bør gjøres og ikke • Hjelper med å se løsningen fra ulike perspektiver • Bidrar til økt brukervennlighet
Visibility	<ul style="list-style-type: none"> • Kan jeg se det? • Synlighet • «Dersom brukeren ikke kan se det, finnes det heller ikke» • Bruker kan se status og alle mulige valg videre • Bruker ikke overveldet av informasjon • Eksempler <ul style="list-style-type: none"> ◦ Lysbryter ◦ Knapper i heisen ◦ Sensorstyrt håndvask
Feedback	<ul style="list-style-type: none"> • Hva gjør det nå? • Bruker opplyses om hva som er blitt gjort og oppnådd hittil slik at han kan fortsette. • Forstå at input hadde en effekt på systemet. • Bruker ser progresjon eller status hvis funksjoner er utilgjengelig eller begrenset • Eksempler <ul style="list-style-type: none"> ◦ Heisknapper piper (og lyser) ◦ Datamaskiner animerer timeglass mens den "tenker" ◦ Word highlighter et ord som er feilstavet ◦ Feedback gis ved hjelp av lyd, lys, animasjon, highlighting eller kombinasjon av flere
Constraints	<ul style="list-style-type: none"> • Hva kan jeg ikke gjøre? • Passe på at brukeren ikke kan gjøre feil. • Handler om å avgrense valgmulighetene til brukeren • Eksempler <ul style="list-style-type: none"> ◦ Fysisk: saks, puslespill, USB-kabel ◦ Logisk: Høyreklikk-meny, knapper som blir grå/deaktivert ◦ Kulturelt: rød trekant for varsel, scrollefelt til høyre
Consistency	<ul style="list-style-type: none"> • Har jeg sett det før? • Bruker ser og anvender stemmer overens med det de er vant til fra før • Gammel kunnskap kan enkelt overføres til ny kontekst • Eksempler <ul style="list-style-type: none"> ◦ Estetisk: Biler (Mercedes Benz – merket forran) ◦ Funksjonelt: trafikklys ◦ Internt: Turskilt i marka

	<ul style="list-style-type: none"> ◦ Eksternt: Menyknapper oppfører seg likt i alle programmer.
Affordance	<ul style="list-style-type: none"> • Hvordan bruker jeg det? • Bruker forstår hvordan noe fungerer ved hjelp av objektets attributter • Bruker får indirekte hint om hvordan noe brukes • Brukeren forstår hvordan noe skal brukes ved hjelp av hvordan det ser ut. • Eksempler <ul style="list-style-type: none"> ◦ Knapper ser ut til å kunne klikkes ◦ «Vridere» på stekeovn ser ut til å kunne vris. ◦ Vi skjønner at vi kan løfte i et håndtak.
Mapping	<ul style="list-style-type: none"> • Hvor er jeg og hvor kan jeg gå? • Forstår sammenhengen mellom en funksjon og effekten den har på verden • Brukeren skjønner hvordan en handling vil påvirke omverdenen • Eksempler <ul style="list-style-type: none"> ◦ Plassering av vridere på komfyren ◦ Dokumentet scroller oppover når vi trykker på tastaturet
Donald Normans 7 prinsipper (Ikke viktig)	<ol style="list-style-type: none"> 1. Use both knowledge in the world and knowledge in the head 2. Simplify the strukture of tasks 3. Make things visible 4. Get the mappings right 5. Exploit the power of constraints, both natural and artificial 6. Design for error 7. When all else fails, standardize
Ben Shneidermans 8 gylne regler for god usability (Ikke viktig)	<ol style="list-style-type: none"> 1. Strive for consistency 2. Enable frequent users to use shortcuts 3. Offer error prevention and simple error handling 4. Design dialogs to yield closure 5. Offer error prevention and simple error handling 6. Permit easy reversal of actions 7. Support internal locus of control 8. Reduce short-term momory load
Jakob Nielsens ti heuristikker (som spesielt gjelder nettsider) (Ikke viktig)	<ol style="list-style-type: none"> 1. Visibility of system status 2. Match between system and the real world 3. User control and freedom 4. Consistency and standards 5. Error prevention 6. Recognition rather than recall 7. Flexibility and efficiency of use 8. Aesthetic and minimalist design 9. help users recognize, diagnose, and recover from errors 10. Help and documentation
Nærhet (proximity)	<ul style="list-style-type: none"> • Den relative avstanden påvirker vår oppfatning av hvorvidt og hvordan objektene er organisert i undergrupper • Objekter nært hverandre fremstår som gruppert
Likhet (Similarity)	<ul style="list-style-type: none"> • Gjenstanders likhet kan benyttes til å gi oss en idé om at enkelte objekter er adskilt fra resten

	<ul style="list-style-type: none"> ◦ Disse vil få mest oppmerksomhet • Vi bruker farger, størrelser, former etc. For å skille objekter fra hverandre
Kontinuitet (Continuity)	<ul style="list-style-type: none"> • Vårt visuelle system har en tendens til å rette opp i tetydighet eller fylle inn manglende data. <ul style="list-style-type: none"> ◦ Vi forsøker å oppfatte kontinuerlige former fremfor oppdelte bruddstykker <p>Eksempel</p> <ul style="list-style-type: none"> • IBM logoen – Masse hvite streker over bokstavene
Kompletthet (closure)	<ul style="list-style-type: none"> • Vårt visuelle system forsøker å lukke figurer slik at de kan oppfattes som hele objekter • Muliggjør bruken av "negative space" ?> tolke blanke/tomme områder i objekter. • Kan få hjernen vår til å se ting som egentlig ikke er der. • Eksempel <ul style="list-style-type: none"> ◦ Hjernen vår tror den ser en sirkel, men det flere buede linjer som ligger ved hverandre
Symmetri	<ul style="list-style-type: none"> • Vi forsøker å tolke komplekse elementer på en måte som reduserer kompleksiteten • Muliggjør 3D-illusjoner i en flate 2D-tegninger
Fire strategier for enkelhet	<ul style="list-style-type: none"> • Fokuserer på enkle brukervennlige løsninger • Fjern <ul style="list-style-type: none"> ◦ Finn ut hva brukerens mål er ◦ Fjern alt brukeren "ikke trenger" (distraksjoner), og heller fokuser på det essensielle. (Constraints, visibility) • Organiser <ul style="list-style-type: none"> ◦ Plasser det som hører sammen, sammen i blokker. ◦ Istedenfor 50 menyvalg => 5 menyer med 10 undervalg ◦ Kartlegg oppførsel og organiser med utgangspunkt i handlingsmønsteret • Skjul <ul style="list-style-type: none"> ◦ Skjul det som ikke trengs nå, og heller ha muligheten for å vise det frem dersom det trengs/ønskes. • Forflytt <ul style="list-style-type: none"> ◦ Flytt roller over på andre enheter ◦ Forflytt det mest individuelle over til brukerens hode <ul style="list-style-type: none"> ▪ Identifiser hva brukere gjør bedre enn maskiner og motsatt ▪ Forsterk en brukeropplevelse ved å la datamaskinen ta over det "frustrerende" ▪ La brukeren dirigere, og maskinen "guide" ◦ Design løsninger for «åpne-opplevelser»

Kapittel 5: Brukeropplevelser (UX)

Brukeropplevelse (UX)	<ul style="list-style-type: none"> • En brukers helhetsopplevelse ved bruken av et system eller et produkt • «Man kan ikke designe en brukeropplevelse, man kan kun designe for brukeropplevelser»
-----------------------	---

	<ul style="list-style-type: none"> • Ulike brukere har ulike preferanser, mål og forventninger • «Dette er en enkel måte å gjøre det på» <ul style="list-style-type: none"> ◦ Hvem har sagt dette? <ul style="list-style-type: none"> ▪ IKKE lov å si dette uten at brukere har sagt det tidligere. ◦ Dersom brukere ikke syntes det er en enkel løsning, så er den heller ikke enkel • Brukervennlighet handler om grensesnittets brukervennlighet og hvor raskt eller bra man kan løse oppgaver (Effectiveness & efficiency) • Brukeropplevelser handler om følelser, kan ikke telles. Brukervennlighet er mer kvalitativt. Hvor mange feil gjør bruker? Etc. • Vi må tenke på alle faktorer som påvirker brukeropplevelsen når vi designer. <ul style="list-style-type: none"> ◦ Hva mer enn brukervennlighet påvirker brukeropplevelser? ◦ Hvordan kan vi forstå situasjonen fra brukerens side? • Hva påvirker brukeropplevelser? <ul style="list-style-type: none"> ◦ Brukbarhet ◦ HCI ◦ Utstyr ◦ Design ◦ Markedsføring ◦ Systemets ytelse ◦ Ergonomi ◦ Tilgjengelighet ◦ Menneskelige faktorer • 3 hovedkomponenter innen brukeropplevelser <ul style="list-style-type: none"> ◦ Opplevelser grunnet brukbarhetsfaktorer (Usability) <ul style="list-style-type: none"> ▪ effectiveness og efficiency ◦ Opplevelser grunnet nytteighetsfaktorer (Usefulness) <ul style="list-style-type: none"> ▪ Snakker om hva noe er iforhold til hva du ønsker å gjøre => mål ◦ Opplevelser grunnet emosjonelle faktorer <ul style="list-style-type: none"> ▪ Involverer et følelsesaspekt ▪ Emosjonell design
Flow	<ul style="list-style-type: none"> • En tilstand der en bruker er så engasjert og involvert i en aktivitet av tiden forsvinner • Typisk i spillbransjen • Jo mer komplisert arbeidskompleksiteten er, jo mer komplisert blir interaksjonen => Det blir vanskelig å skape flow hos bruker
Domene	<ul style="list-style-type: none"> • Vi designer ofte for spesifikke brukskontekster hvor god innsikt krever domenekunnskap • For å skape gode brukeropplevelse, må vi sette oss inn i domene eller hente inn domeneeksperter • Domene <ul style="list-style-type: none"> ◦ Felt eller omgang av en bestemt kunnskap eller aktivitet • Domenekunnskap <ul style="list-style-type: none"> ◦ Kunnskap innenfor et bestemt domene • Domeneekspert <ul style="list-style-type: none"> ◦ Ekspert innenfor et bestemt domene

Gamification	<ul style="list-style-type: none"> • Teknikk som utnytter menneskers naturlige konkurranseinstinkt • Handler om å trigge ønsket om å være best • Vi ønsker å basere oss på konkurranseelementer <ul style="list-style-type: none"> ◦ Oppnåelse, status, ledere, medaljer, poeng, nivåer etc. • Brukes mye i reklame og annonser for å drive markedsføring <ul style="list-style-type: none"> ◦ Skattejakt, finn-og-vinn, besøk alle våre butikker for rabatter etc.
Representasjon	<ul style="list-style-type: none"> • Vi bør alltid velge en representasjon som ligger tett på brukers mentale modell
Mental modell	<ul style="list-style-type: none"> • Hvordan en bruker tror et system fungerer • Bruker tar i bruk systemer med utgangspunkt i sin mentale modell • Både bruker og designer tar med sin mentale modell i interaksjonen <ul style="list-style-type: none"> ◦ Pass på at ikke du overtar, men baser systemet på brukeren • En konseptuell modell presenterer designerens mentale modell • Viktig å tenke på kontekst for å forstå brukerens mentale modell
Implementert modell	<ul style="list-style-type: none"> • Måten et system faktisk fungerer på <ul style="list-style-type: none"> ◦ programkode, hardware etc.
Representert modell	<ul style="list-style-type: none"> • Måten vi presenterer en løsning på ovenfor en bruker • Systemet snakker samme språk som menneskene
Brukeropplevelsesmål	<ul style="list-style-type: none"> • Handler om hvordan en bruker opplever bruken av et system fra deres perspektiv • Konflikt med brukbarhetsmål? • Mål (Ikke fullstendig liste) <ul style="list-style-type: none"> ◦ Positivt (Ønsket) <ul style="list-style-type: none"> ▪ Tilfredshet, underholdende, engasjement, behagelighet, fornøyelighet, spennende, utfordrende, hjelpsomt, motiverende. ◦ Negativt (Ikke ønsket) <ul style="list-style-type: none"> ▪ Kjedsomhet, barnslighet, forvirrende, irriterende, frustrerende, fornærmende, nedlatende • Hvordan «fange» brukeropplevelser <ul style="list-style-type: none"> ◦ Samle artefakter <ul style="list-style-type: none"> ▪ Bilder og oppbevaring av slike artefakter kan være nyttig for å forstå oppgaveflyt og rutiner ◦ Intervju eller observasjon etter brukeropplevelse ◦ Se etter opplevelse i dataen både før og etter analysen
Brukbarhetsmål	<ul style="list-style-type: none"> • Angir hvor produktivt et system er i et eget perspektiv • Sier ingenting om følelser! • Mål <ul style="list-style-type: none"> ◦ Effektivitet (effectiveness) ◦ Flittighet (efficiency) ◦ Trygghet ◦ Nyttighet ◦ Lærbarhet ◦ Memorerbarhet
Emosjonell design	<ul style="list-style-type: none"> • Handler om frembringelse og oppfattelse av emosjonelle sider gjennom designet og brukeropplevelsen. <ul style="list-style-type: none"> ◦ Brukervennlig, komfortabel, tilfreds

Visceral	<ul style="list-style-type: none"> • Biologiske instinktive reflekser • Raske beslutninger basert på hva som er bra/dårlig eller trygt/farlig • Utseende og attraktivitet => appellerer til "magefølelsen"
Behavioral	<ul style="list-style-type: none"> • Her det er meste av menneskelig oppførsel • Ikke et bevisst plan (Kjøre il og tenke på noe annet samtidig) • Glede og effektivitet => appellerer til vår jakt etter brukbarhet og prestasjon
Reflective	<ul style="list-style-type: none"> • Evnen til å jobbe med tanker og refleksjoner • Handler ikke om direkte kontroll over sensorisk input eller oppførsel, men å reflektere, overvåke og påvirke oppførsel over tid. • Selvbilde og utvikling => appellerer til vårt selvbilde, minner, identitet, personlig glede

Kapittel 13 og 14: Evaluering

Evaluering	<ul style="list-style-type: none"> • Generelle typer evalueringsstudier <ul style="list-style-type: none"> ◦ Kvantitativt => numeriske studer, statistikk ◦ Kvalitativt => studier av mening, tekst, samtale <ul style="list-style-type: none"> ▪ Case study => Lavt antall, men høyt detaljerte beskrivelser ▪ Etnografi => Studier i naturlig omgivelser • Krever en artefakt => simulering, prototype, ferdig produkt, funn eller analyse. • Hva ønsker vi å evaluere? <ul style="list-style-type: none"> ◦ I begynnelsen er det f.eks. Nyttig å vite om alle funksjonene man vil ha med er det ◦ Mot slutten av designprosessen er det mer nyttig å evaluere for å finne ut av om fargekombinasjoene er gode • En aktivitet som gir oss tilbakemelding og innspill, basert på bruk av en rekke artefakter • Tre tilnærminger for evaluering (Fra bok dette semester) <ul style="list-style-type: none"> ◦ Kontrollerte omgivelser ◦ Naturlige omgivelser ◦ Uten brukere. • Tre tilnærminger for evaluering (tidligere brukt bok) <ul style="list-style-type: none"> ◦ Brukbarhetstesting ◦ Feltstudier ◦ Analytisk evaluering • Hvorfor evaluere? <ul style="list-style-type: none"> ◦ Prosess ikke hendelse ◦ Identifisere + og – egenskaper for å drive design videre ◦ Avgjøre hvor brukbar en løsning er for ulike brukere/brukergrupper. <ul style="list-style-type: none"> ▪ Møter vi brukerens behov? • Hva kan vi evaluere? <ul style="list-style-type: none"> ◦ Konseptuelle modeller, low- og high-fi prototyper
------------	---

	<ul style="list-style-type: none"> ◦ Alt som kan gi brukeren en ide om endelig løsning • Hvor kan vi evaluere? <ul style="list-style-type: none"> ◦ Laboratorium eller «naturlige omgivelser» • Når kan vi evaluere? <ul style="list-style-type: none"> ◦ Når som helst, så ofte som mulig
Oppertunistisk evaluering	<ul style="list-style-type: none"> • Når man kombinerer brukbarhetstesting, feltstudier og/eller analytisk. • Vi benytter «Anledning» til å velge tilnærming
Brukbarhetstesting	<ul style="list-style-type: none"> • Brukere gjør definerte oppgaver i kontrollerte omgivelser <ul style="list-style-type: none"> ◦ Laboratorium ◦ Evaluering av en bruker ISOLERT • Observasjoner, målinger og feil rapporteres • Spør brukere • Måler følgende metrikker <ul style="list-style-type: none"> ◦ Tid. ◦ Fullføringsrate. ◦ Antall feil. ◦ Antall forsøk. ◦ Tilfredshet.
Formativ testing	<ul style="list-style-type: none"> • Utføres i en tidlig fase • Typisk low-fi prototypene • Fokus på hvordan grensesnitt oppfattes (fremfor oppgaveløsning) <ul style="list-style-type: none"> ◦ Forstår brukere dette grensesnittet? • Billig, raskt og kan utføres • Lettere å være kritisk når de ser papirprototype, fremfor ferdig app
Summativ testing	<ul style="list-style-type: none"> • Senere fase • Formell prototype er klar • Høy-nivåbeslutninger er allerede tatt <ul style="list-style-type: none"> ◦ Grensesnitt ferdig • f.eks. Evaluering av brukbarheten ved et designvalg
Validering (ikke viktig)	<ul style="list-style-type: none"> • Grensesnittet eller funksjonalitet måles opp mot andre tilsvarende løsninger • Brukes sjeldent
Effektivitet, flittighet og tilfredshet	<ul style="list-style-type: none"> • Er tre ISO-definerte dimensjoner ved brukbarhet • Effektivitet (Effectiveness) <ul style="list-style-type: none"> ◦ Hvor godt vi klarer å løse oppgaver <ul style="list-style-type: none"> ▪ Presisjon, antall feil • Flittighet (Efficiency) <ul style="list-style-type: none"> ◦ Hvor raskt vi klarer å løse oppgaver • Tilfredshet (Satisfaction) <ul style="list-style-type: none"> ◦ Hvor tilfreds vi er med å løse oppgaver
Lærbarhet og memorerbarhet	<ul style="list-style-type: none"> • Lærbarhet (Learnability) <ul style="list-style-type: none"> ◦ Hvor lett kan vi lære oss å løse oppgaver • Memorabilitet (Memorability)

	<ul style="list-style-type: none"> ◦ Hvor mye som huskes hvis vi eter en stund skal løse oppgaver
Eksperimentell evaluering	<ul style="list-style-type: none"> • Tester en hypotese for å generere ny kunnskap • Brukbarhetstesting er anvendt eksperimentering • Relevante begreper <ul style="list-style-type: none"> ◦ Hypotese <ul style="list-style-type: none"> ◦ Avhengige, uavhengige og kontrollerte verdier ◦ Presisjon og generaliserbarhet • Eksempel på adferdsforskning • Eksempel <ul style="list-style-type: none"> ◦ Hypotese <ul style="list-style-type: none"> ▪ Det er ingen forskjell i mengden gjødsel sin påvirkning på plantens vekst ◦ IV <ul style="list-style-type: none"> ▪ Mengde gjødsel ◦ DV <ul style="list-style-type: none"> ▪ Høyde, vekt, bær ◦ CV <ul style="list-style-type: none"> ▪ Potta, vann, temperatur, sollys ◦ Utføring <ul style="list-style-type: none"> ▪ Ni planter skal testes med 3 mengder gjødsel ▪ plante 1-3: <ul style="list-style-type: none"> • Minst gjødsel ▪ plante 4-6: <ul style="list-style-type: none"> • Mellomstor mengde gjødsel ▪ Plante 7-9: <ul style="list-style-type: none"> • Mest gjødsel ▪ Måler avhengige variabler for å si noe om den uavhengige variabelen har påvirkning <ul style="list-style-type: none"> • Dette brukes for å bekrefte/avkrefte hypotesen
Hypotese	<ul style="list-style-type: none"> • En utestet antakelse av virkeligheten • Må være etterprøvable for å være gyldig
Uavhengig variabel (Independent variable(IV))	<ul style="list-style-type: none"> • Hva er det du manipulerer/endrer for å undersøke om det påvirker sluttresultatet? => Som regel en del av grensesnittet eller interaksjonen. • Eksempel: <ul style="list-style-type: none"> ◦ Ulike måter å presentere en løsning på (oblig 2) • Uavhengig av deltakers adferd • Denne variabelen endres for å undersøke om den faktisk har en påvirkning på f.eks. Vekst av planten
Avhengig variabel (Dependant variable(DV))	<ul style="list-style-type: none"> • Hva skal du måle? => avhenger av oppsettet, og faktorene du manipulerer. • Eksempler <ul style="list-style-type: none"> ◦ Nøyaktighet, emosjonell, feil antall etc. • Avhenger av deltakers adferd (Eller IV) • Her måler vi prestasjon, tilfredshet, flittighet, ved å måle metrikker.

	<ul style="list-style-type: none"> • Vi bruker metrikker for å måle utfallet gjennom denne variablen • Dette er en faste variabler. Vi gir samme mengde vann, sollys, samme potte og temp.
Metrikk	<ul style="list-style-type: none"> • Målbart eller kvantifiserbart forhold vi kan bruke til sammenlikning • Kvantitative data • Eksempler <ul style="list-style-type: none"> ◦ Flittighet => Tid => Sekunder ◦ Vekt => Gram ◦ Presisjon => Antall feil
Kontrollert variabel (Controlled variables (CV))	<ul style="list-style-type: none"> • Forholdene vi bevisst holder like mellom alle rundene i eksperimentene. • Faktorer som kontrolleres gjennom eksperimentet for å unngå IVs effekt på DV
Presisjon	<ul style="list-style-type: none"> • Intern gyldighet • Hvis vi gjør eksperimentet en gang til, vil jeg få de samme resultatene? <ul style="list-style-type: none"> ◦ Hvis dette stemmer, har vi god presisjon. • Fokus på konsistens, fjerne alle forstyrrende faktorer, dokumentasjon etc. <ul style="list-style-type: none"> ◦ Vil været være det samme (Hvis det kan ha betydning)
Generaliserbarhet	<ul style="list-style-type: none"> • Ekstern gyldighet • Er det vi måler, noe man kan finne igjen i den virkelige verden? • «Iboende funn» (intrinsic validity) gjør oss ikke automatisk i stand til å kunne generalisere
Between-subject testing	<ul style="list-style-type: none"> • Vi deler inn i grupper som tester hver sin tilstand (like mange grupper som tilstander å undersøke)
Within-subject testing	<ul style="list-style-type: none"> • Alle deltakere tester alle tilstander <ul style="list-style-type: none"> ◦ Kan være en fordel å randomisere rekkefølgen, slik at den ikke er lik hos alle (randomisering) <ul style="list-style-type: none"> ▪ Dette er fordi det kan oppstå læringseffekt (bias) <ul style="list-style-type: none"> • Dette betyr at brukeren kjenner systemet, slik at de har mer kunnskap om systemet etter hver tilstand
Feltstudier	<ul style="list-style-type: none"> • Evaluering av prototype i brukssituasjonens naturlige omgivelser • Fåreta i senere fase når: <ul style="list-style-type: none"> ◦ Prototypen er mer utviklet ◦ Ikke bare horisontale, men vertikale funksjoner. ◦ Prototypen allerede er evaluert i kontrollerte omgivelser • Fordeler: <ul style="list-style-type: none"> ◦ Tilstander som er umulig å ha i lab, kontekst og beliggenhet (omgivelsene stemmer)

	<ul style="list-style-type: none"> ○ Bruker oppfatter bruken mer naturlig ○ Mer realistisk (støy, andre personer etc.) ○ Kan utvides til langtidsstudier • Ulemper <ul style="list-style-type: none"> ○ Omfattende (ressurskrevende) ○ Hawthorne-effekten ○ Vanskelig å reproducere <ul style="list-style-type: none"> ▪ Variabler som ikke er konstante (vær og vind)
Etnografisk evaluering	<ul style="list-style-type: none"> • Brukes for å forstå brukere og brukskontekst • Kvalitativ forskningsmetode • Typiske observasjonsteknikker: skygging, flue-på-vegg • Kan også bruke intervju
Analytisk evaluering	<ul style="list-style-type: none"> • Brukere holdes utenfor evaluering • Kan gjøres når som helst • Tar ofte utgangspunkt i ikke-bruk • Analytisk: (eksperter) <ul style="list-style-type: none"> ○ Teori, modeller, retningslinjer • Empirisk: (brukere) <ul style="list-style-type: none"> ○ Observasjoner, spørreundersøkelser, litteratur • gjennomgang <ul style="list-style-type: none"> ○ Kognitiv eller pluralistisk gjennomgang ○ Inkludere domeneeksperter ○ Grupper av HCI-eksperter og domeneeksperter • Heuristisk evaluering <ul style="list-style-type: none"> ○ Gjennomgang etter retningslinjer ○ Brukes mye i næringslivet og industrien ○ Nielsen mente at kun med 3-5 brukere kan man finne 80% feil • Problemer og tilbakemelding rapporteres • Spørre eksperter (HCI eller domene)
Heuristikk	<ul style="list-style-type: none"> • Bygger på «nedfelte» erfaringer og prinsipper • «tommefingerregel»-vurdering
Kognitiv gjennomgang (Cognitive walkthrough)	<ul style="list-style-type: none"> • Designet evalueres av eksperter • Formål: undersøke hvor godt designet støtter bruken i oppgaveløsning • Ekspertene følges steg for steg for å identifisere potensielle problemer. • For hvert steg <ul style="list-style-type: none"> ○ Hvilken påvirkning vil interaksjonen få? ○ Hvilke kognitive prosesser kreves fra brukeren? ○ Hvilke læringsproblemer kan oppstå? • Fokus på mål og kunnskap • Ofte brukt tidlig i designprosessen
Ekspertevaluering	<ul style="list-style-type: none"> • Analytisk evaluering • En eller flere eksperter evaluerer en løsning og påpeker problemer og svakheter

	<ul style="list-style-type: none"> • Ekspertene setter seg i brukerens rolle • Kan bruke HCI-eksperter eller domeneeksperter
--	--

Kapittel 15: DECIDE

DECIDE	<ul style="list-style-type: none"> • Seks punkters sjekkliste • DECIDE kan kun brukes for planlegging av en evaluering • Består av seks steg <ul style="list-style-type: none"> ◦ Determine the goals ◦ Explore the questions ◦ Choose the evaluation methods ◦ Identify practical issues ◦ Decide how to deal with ethical issues ◦ Evaluate, analyze, interpret and present the data
Bestem mål (Determine the goals)	<ul style="list-style-type: none"> • Hva er målene med evalueringen? <ul style="list-style-type: none"> ◦ Disse påvirker metoder og evalueringen • Hvem vil ha resultatene og hvorfor? • Mål varierer. Noen eksempler <ul style="list-style-type: none"> ◦ Identifisere metoden for et gitt design. ◦ Sjekke at grensesnittet er konsekvent ◦ Sjekke om vi dekker brukerens behov
Utforsk spørsmålene (Explore the questions)	<ul style="list-style-type: none"> • Spørsmål bidrar til å lede evalueringen • Eksempel fra tidligere bok <ul style="list-style-type: none"> ◦ Mål: Finne ut hvorfor reisende foretrekker papirbilletter fremfor elektroniske billetter når de skal ut å fly. <ul style="list-style-type: none"> ▪ Spørsmålet deles opp <ul style="list-style-type: none"> • Hva er brukerens holdning til elektroniske billetter? • Er brukerne bekymret eller opptatt av sikkerhet? • Hvordan oppfattes brukergrensesnittet?
Velg tilnærming og metode for evaluering (Choose the evaluation methods)	<ul style="list-style-type: none"> • Valg av tilnærming og valg av metode påvirker hvordan data samles inn, analyseres og presenteres. • Hva slags data vil du ha? Kvantitative eller kvalitative data? • Valget avhenger også av <ul style="list-style-type: none"> ◦ Hvilken type data som er nødvendig for å svare på spørsmålene ◦ Hvilke teorier eller rammeverk som er passende for konteksten • Eksempel for feltstudie: <ul style="list-style-type: none"> ◦ Observasjoner og intervjuer ◦ Involvere brukere i naturlige omgivelser • Metoder og tilnærminger kan kombineres
Identifisere praktiske forhold (Identify practical issues)	<ul style="list-style-type: none"> • Velge deltakere <ul style="list-style-type: none"> ◦ Antall, bakgrunn, utvalg (kjønn, alder, kultur, holdning), signifikans • Finne de som skal evaluere <ul style="list-style-type: none"> ◦ HCI-eksperter, medhjelpere, domeneeksperter • Velge utstyr <ul style="list-style-type: none"> ◦ Type opptaksutstyr, antall, plassering, ansvarlig • Forholde seg til budsjett

	<ul style="list-style-type: none"> ◦ Betaling til deltakere, tilgang til utstyr • Forholde seg til tid <ul style="list-style-type: none"> ◦ Tidsbegrenset tilgang til utstyr/deltakere, frister, timeboxing
Beslutning av etiske forhold skal håndteres (Decide how to deal with ethical issues)	<ul style="list-style-type: none"> • Utvikle samtykkeerklæring • Deltakere har rett til. <ul style="list-style-type: none"> ◦ Informasjon om formål og målsettinger for studien ◦ Hva som skal gjøres med funn, og hvordan dataene skal behandles ◦ Vern av personopplysninger ◦ Frihet til å trekke seg ◦ Høflig behandling
Evaluerer, analysere, tolke og presentere data.	<ul style="list-style-type: none"> • Tilnærming og metode påvirker hvordan data evalueres, tolkes og presenteres • Forhold som må vurderes <ul style="list-style-type: none"> ◦ Pålitelighet (Reliability) <ul style="list-style-type: none"> ▪ Kan studiet reproduseres ◦ Gyldighet (Validity) <ul style="list-style-type: none"> ▪ Måles det man forventet å måle? ◦ Bias <ul style="list-style-type: none"> ▪ Medfører prosessen noen form for bias? (Halo-effekten) ◦ Omfang <ul style="list-style-type: none"> ▪ Kan funnene generaliseres ◦ Økologisk gyldighet <ul style="list-style-type: none"> ▪ Påvirker miljøet og omgivelsene funnene? (Hawthorne-effekt)