

IT-systemers betydning for og påvirkning af menneskelig aktivitet

Susanne Bødker, Datalogisk Institut, Aarhus Universitet. 2011, Version 1.0

IT er skabt af mennesker, for mennesker

Når vi beskæftiger os med IT-systemer, er det først og fremmest vigtigt at forstå, at de er skabt af mennesker, for mennesker. Uanset om vi ser på et kasseterminalsystem i Føtex, Facebook-Appen på en Smartphone eller standard-skrivebordet på en PC, ligger der masser af menneskeligt arbejde bag, hvordan de ser ud og fungerer.

Dette udviklingsarbejde er foregået på tre niveauer: På det første, generelle niveau foregår der forskning og generel udvikling af alt fra effektive algoritmer og smarte nye interaktionsformer til effektiv udnyttelse af menneskers tid. På det andet niveau foregår der en konkret udviklingsproces, som fastlægger teknologiens egenskaber og brug. Endelig fortsætter udviklingsprocessen også i brug, når mennesker gør teknologien til deres egen på forskellig vis, som vi vil se på senere.

På den måde kan man sige, at udviklingsprocessen ikke holder op før den dag kasseterminalsystemet eller PC-skrivebordet ikke længere bruges. I praksis lever mange systemer så længe, at der ikke længere findes udviklere, som har bygget dem. Et godt eksempel er de store telefonselskaber, som stadig benytter systemer, der blev udviklet i 1970'erne. De er nu blandet sammen med en række nyere systemer, og selvom telefonselskabet skulle ønske sig at tage det gamle system ud af drift, er det nærmest umuligt at vide, hvilke konsekvenser det måtte have. Derfor kører de gamle systemer videre, og unge programmører efteruddannes i antikke programmeringssprog.

Udvikling til marked eller konkret brug

Den konkrete udviklingsproces er i nogle situationer rettet mod en bestemt organisation og nogle bestemte brugere, fx. når et telefonselskab udvikler et nyt system, der skal hjælpe med at sende reparatører ud til fejlene på deres net. Her er de konkrete brugere og den, der har bestilt systemet, tæt på udviklerne.

Andre udviklingsprocesser er rettet mod et marked. Det gælder Smartphone Apps såvel som fx regnskabssystemer til mindre virksomheder. I disse situationer kommer køberen først ind i billedet, når systemet er udviklet, og de konkrete brugere er ikke kendte.

Udviklingen hænger sammen og er uforudsigelig

At konkrete udviklingsprojekter spiller sammen med mange andre former for udvikling kan ses af det følgende eksempel, mobiltelefonens udvikling: Den oprindelige mobiltelefon lignede mest af alt en gammeldags telefon, blot med den forskel at den kunne flyttes, hvis man var den lykkelige ejer af en bil (for tung var den). Den næste generation af netværksteknologi muliggjorde mindre telefoner, der kunne passe i lommer og tasker.



(billeder fra Google)

SMS var i starten et relativt teknisk forsøg på at benytte ekstra kapacitet på telefonnettet til korte service-beskeder. Men til alle leverandørers og forskeres overraskelse blev SMS inden for kort tid en meget anvendt kommunikationsform på mobil-telefoner. I starten var telefonerne uden ordbøger, så brugerne måtte trykke mange knaptryk, i første omgang med pegefingeren. Ericsson og Nokia arbejdede hårdt på at udvikle ordbøger. De ihærdige brugere lærte sig, at det var hurtigere at skrive SMS med tommelfingeren og senere igen med to hænder. Telefonerne fik kamera og musikafspiller og gik for nogle brugere over til at erstatte fotografiapparat og Walkman, SONYs meget solgte musikafspiller. SMS tog i store dele af (primært) Europa over som den hyppigste brug af mobilen. Brugere blev yngre og yngre. I Japan udviklede man mange andre slags on-line services, og alt dette var med til at inspirere smartphonen. Hver andet barn i Danmark fik efterhånden en telefon, og det var ikke længere muligt at gemme sig for mor og far, eller for kammeraterne for den sags skyld. Telefon-mobning opstod, og mobilen overgik til at være en personlig ting: man ringede ikke længere til et sted (et hjem eller en arbejdsplads), men til en person.

Selvom mange mennesker i de seneste år har købt smartphones, fordi de havde brug for en ny telefon, har smartphones endnu engang revolutioneret, hvad en mobiltelefon "er": For de fleste smartphone-brugere er det primært applikationerne og internettet de bruger, mens samtaler udgør en meget lille del. Avancerede og innovative interaktionsformer er dukket op på de fleste smartphone-platforme, og mange brugere er ganske simpelt lykkelige for at eje en smartphone (hvilket næppe var tilfældet med den klumpede oprindelige mobiltelefon). iPhone har givet inspiration til iPaden, som helt bestemt ikke længere er en telefon. Men iPhone er også et relativt dårligt redskab til at sende SMS'er. Egentligt er det ganske tydeligt, at den er udviklet i en kultur, hvor man ikke SMS'er så meget som i Danmark.

Selvom mobiltelefonen er så meget andet end et stykke informationsteknologi, illustrerer dens udvikling samspillet i udvikling mellem forskellige dele. Udviklingen af netværksteknologi, hardware og brug/formål hænger sammen. Men historien illustrerer også, hvor meget mobiltelefonen har ændret vores liv og mellem-menneskelige relationer.

IT som model af menneskelig aktivitet

For at komme nærmere IT og dets udvikling og brug, vil vi starte et andet sted: Et hyppigt set perspektiv på IT i brug er at se det som en *model af den menneskelige arbejdsproces*, der foregik inden IT-systemet blev indført. Masser af brugere, ikke mindst i det offentlige, er over årene blevet bedt om at være med til at kortlægge og beskrive deres egne arbejdsgange, for at disse kan "lægges ind i" et IT-system. Med udgangspunkt i dette perspektiv kan man med andre ord se IT-systemet som en model

af en menneskelig aktivitet, og udviklingsprocessen har fokus på at modellere denne aktivitet bedst muligt.

Diskussion: Lav den bedst mulige beskrivelse af hvordan du kommer i skole om morgen. Bed en af dine venner om at følge beskrivelsen. Diskutér hvor effektiv beskrivelsen er som opskrift eller foreskrift, og overvej hvor tit du selv afviger fra beskrivelsen.

IT til rationalisering og automatisering

Siden 1960'erne, da IT-systemer først kom frem, har IT i organisatorisk sammenhæng været udviklet med henblik på *rationalisering og automatisering*. Mange manuelle arbejdsprocesser er over årene blevet afskaffet på den konto, men perspektivet er langt fra kun historisk: Hvis vi kigger på dagens kassesystemer i supermarkedet, har der været mange skridt, fra dengang købmanden langede varerne over disken og regnede prisen ud med papir og blyant. I tidernes morgen gik rationaliseringen først og fremmest ud på at lade kunderne selv samle varer fra hylderne, samtidig med at man kunne ansætte uuddannede kassedamer til at taste priser ind. I dag er vi der, hvor varerne leveres med stregkoder, som kunderne selv kan scanne. Der er både fokus på at spare arbejdskraft og på at samle information om kundernes køb, så supermarkedet automatisk kan få leveret de varer, der sælger mest, og som kommer til at mangle på hylderne. På den måde omfatter kassesystemet hele fødekæden fra leverandører til kunder og ikke kun, hvad der foregår i supermarkedet isoleret set. Gennem medlemskaber som fx hos FDB kan man gå langt videre og undersøge den enkeltes forbrugsmønstre, og det giver yderligere muligheder for at lave tilbud til den enkelte.

Diskussion: Hvad er mon næste skridt i automatiseringen af supermarkeder. Hvad kan stadig rationaliseres og hvordan? Hvordan kan udgifterne skæres ned? Har det konsekvenser for privatlivets fred, at supermarkederne kender vore indkøbsmønstre i stor detalje?

IT som værktøj til kvalitetsbrug

I 1980'erne opstod en modbevægelse til både modellerings- og automatiseringstænkningen: Ville det så ikke være bedre at *forstå og udvikle IT-systemer som værktøj* på linje med de værktøjer, som kunsthåndværkeren eller den faglærte brugte? Med dette perspektiv kom mange forskellige værktøjer til at producere kvalitetsprodukter i højsædet: Værktøjet hjælper brugeren med at opnå resultater, som vedkommende ellers ikke kunne. I stedet for at fuld-automatisere fx bog- og avis-produktion sætter dette perspektiv fokus på, hvordan brugeren kan have kontrol over materialet og styre kvaliteten af produktet. Velkendte IT-systemer som Microsoft WORD er et resultat af denne slags tænkning.

Udviklingen tog yderligere fart, da IT flyttede fra arbejdspladsen ud i hjemmet, hverdagslivet og hvor som helst. Det er lettere at forstå de fleste Apps som værktøj end som modeller af menneskelig aktivitet.

IT som dynamisk infrastruktur

I organisatorisk sammenhæng har det, som antydnet ovenfor, også været nødvendigt at forstå og gentænke IT, som noget der er mere dynamisk end et system, der udvikles og indføres et ad gangen. Mange teknologiske løsninger, der kommer og går og griber ind i hinanden, giver en *dynamisk infrastruktur*. På samme måde som andre infrastrukturer, såsom veje og lignende, giver IT-infrastrukturen grobund for

forskellige nye aktiviteter og handlemuligheder. Samtidig er arbejdsopgaver under stadig udvikling, og arbejde og fritid er filtret sammen: Folk bruger elektronisk post, chat/instant messaging, SMS'er og Facebook, både når de arbejder og ikke arbejder. Elektronisk post (email) er et godt eksempel på en teknologi, som startede på arbejdspladsen og derfra flyttede ud og blev brugt i andre dele af menneskers liv. Instant messaging har i høj grad haft den modsatte udvikling. Fra at være noget som (unge) mennesker benyttede med vennerne, bruges det i dag til diskussioner i projektgrupper, fx i IT-udviklingsprojekter, og chat-funktioner findes i alt fra Facebook til projektstyringsværktøjer.

De fire beskrevne perspektiver (model, automatisering, værktøj og dynamisk infrastruktur) på IT-systemer i brug kan delvist forstås som en historisk udvikling. Ikke desto mindre kan alle perspektiverne genfindes i den IT, vi omgiver os med i det daglige.

Diskussion: Find nogle gode eksempler på, hvordan de fire perspektiver kan genfindes i dag og diskutér dem.

Mennesket i fællesskaber

Mennesket er grundlæggende socialt, og menneskelig aktivitet er præget af *samarbejdet* mellem os, hvad enten det er i en familie for at sætte en flok børn i vej, eller det er på arbejdet, hvor vi løser bestemte opgaver sammen med andre, i *forskellige fællesskaber*. Samtidig kan vi ikke alle sammen al ting – konkret har vi forskellige evner, og på det samfundsmæssige niveau kører nogen tog, mens andre bygger dem, og endnu andre sælger billetter og betaler regningerne. Der er en *arbejdsdeling* på mange niveauer, og fællesskaber såvel som arbejdsdeling understøttes af IT, som vi vender tilbage til.

Diskussion: Hvilken arbejdsdeling er der på gymnasiet, som i sidste ende gør, at I står med et eksamensbevis i hånden?

Menneskelig viden

Forskellen mellem system og værktøj relaterer sig til forskellig slags forståelse af, hvad viden er:

Systemperspektivet baserer sig på en forestilling om, at *menneskelig viden* er eller kan modelleres og gøres *eksplicit*, for derefter at blive lagt ind i systemerne (hvorved systemerne så også kan bruges af folk, som ikke besidder den pågældende viden – automatisering i en nøddeskal). Det samler altså model og automatisering.

Værktøjsperspektivet fokuserer på kvalitet og dynamisk infrastruktur og hylder en forestilling om, at visse former for viden ikke er eksplicite, og at de taber i kvalitet, når de forsøges ekspliciteret. Klassiske eksempler herpå er *færdigheder* som at cykle eller spille et instrument. Ingen af disse færdigheder kan læres ved at læse en beskrivelse eller manual. Yderligere er det vanskeligt af beskrive, hvornår nogen er en dygtig guitarist, eller hvorfor Ritchie Blackmores guitarriff på ”Smoke on the Water” er verdens bedste. Det er en viden, som folk, der selv mestrer håndværket, kan dele mellem sig. Vi snakker i disse tilfælde om *tavs viden*, som står i modsætning til eksplicit viden, så som hvad prisen er på en busbillet, og hvor man køber den. Man kan kort fortalt sige, at eksplicit viden er let at lægge ind i IT-systemer, mens tavs viden må aktiveres på andre måder.

Diskussion: Giv eksempler på, hvad der er tavs og eksplicit viden, når det kommer til at holde et mundtligt oplæg i klassen. Hvilke krav skal vi stille, hvis vi skal bygge en IT-løsning der understøtter præsentationsprocessen? Gør det en forskel, om vi benytter værktøjs- eller system-perspektivet?

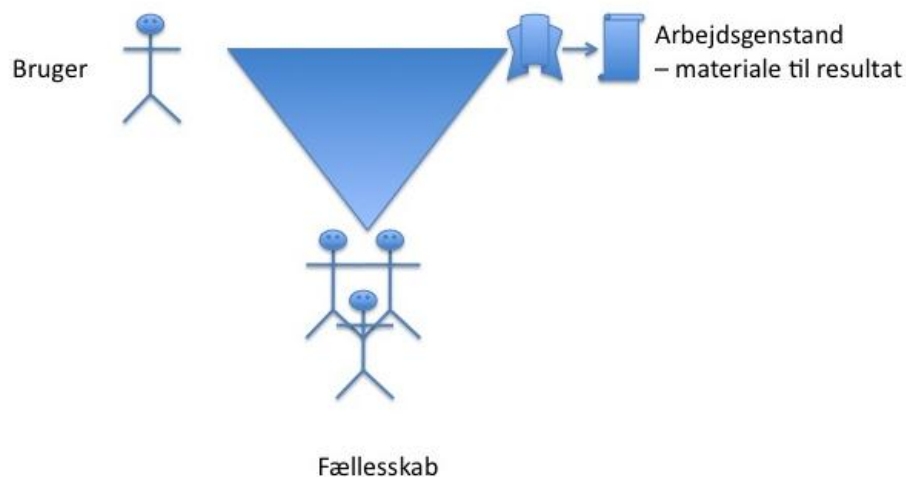
Diskussion: Hvad vil vi lægge vægt på, hvis vi skal forstå skolens skema-applikation som hhv. system og værktøj? Giver de to perspektiver anledning til, at man kan foreslå forbedringer?

Læring og fællesskab

I alle sådanne fællesskaber foregår der en *læring*: Den enkelte starter som ny tømmerlærling, elevrådsmedlem osv. og lærer gennem deltagelse de rutiner og opgaver, der hører til: at bruge de nødvendige redskaber, samarbejdsformer m.m., hvad enten de er understøttede af IT eller ej. Nybegynderen er *perifer deltager* i fællesskabet, men har ret til og behov for at være der for at kunne lære både den eksplicite og den tavse viden i fællesskabet. Efterhånden som nybegynderen lærer, flytter vedkommende sig fra periferien af læringsfællesskabet til centrum, hvor vedkommende er et fuldgældigt medlem.

Samtidig påvirker det enkelte medlem af læringsfællesskabet de rutiner og opgaver, der udføres, og er med til at forandre fællesskabets måder at gribe dets gøremål an på. Nye redskaber kommer til, og andre forsvinder, og igen er IT på den måde med til at ændre rutiner og opgaver.

Den enkelte person eller *bruger*, som vi ofte siger, når IT er involveret, deltager altså i et fællesskab, som i samarbejde skaber et resultat eller et produkt vha. diverse materialer og teknologier (Figur 1).



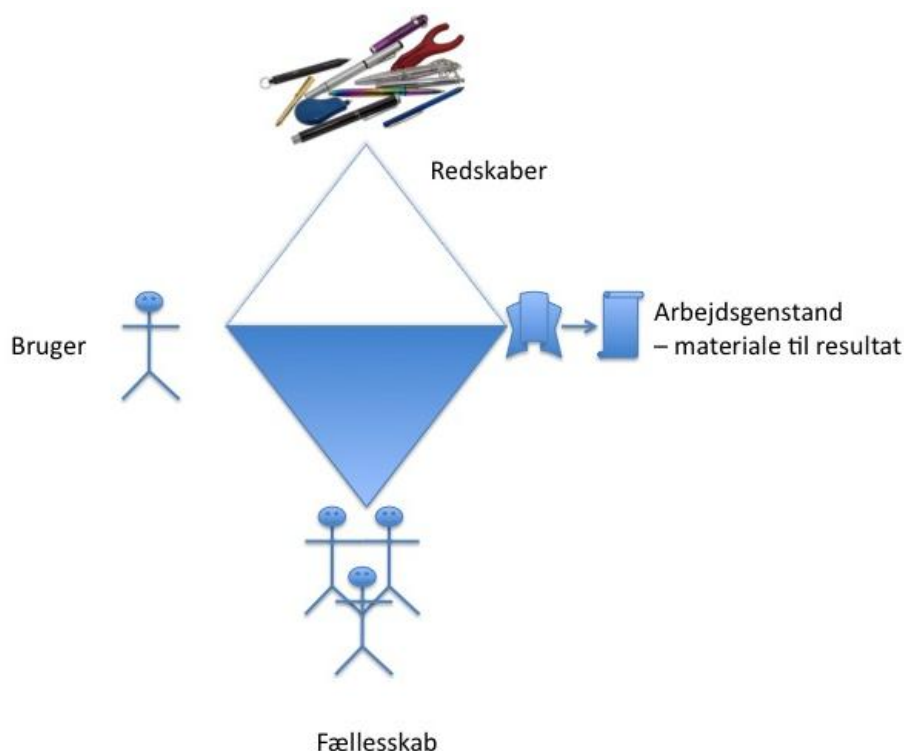
Figur 1. Bruger og fællesskab

Mediering, arbejdsgenstande og redskaber

Vi vender tilbage til menneskets relation til de *genstande*, vi bearbejder, de *materialer* som forvandles til et eller andet *resultat eller produkt* gennem brugen af redskaber. *Redskaber* øger menneskets handlemuligheder fx ved at forøge vores styrke eller præcision. Vi skriver breve vha. et tekstbehandlingsprogram; forvandler nogle

udgifter og indtægter til et regnskab vha. et regneark; og vi bygger huse af mursten og bjælker vha. sav, hammer og murer-ske. Redskabet står imellem, eller *medierer*, brugeren og arbejdsgenstanden (materialet som transformeres til resultat eller produkt, figur 2). Det støtter nogle erfaringer og handlemuligheder, og forhindrer andre. Når redskabet fungerer, er brugerens opmærksomhed rettet mod arbejdsgenstanden, mens opmærksomheden rettes mod redskabet som sådan i *sammenbrudssituationer*, hvor redskabet på en eller anden måde holder op med at virke. Sådanne sammenbrud er der, hvor læring sker, og nye rutiner udvikles.

Diskussion: Tænk tilbage på sidste gang du fik ny telefon. Hvilke rutiner kunne du umiddelbart tage med videre til den nye telefon? Sender man fx en SMS på præcis samme måde? Hvornår opdagede du, at noget skulle gøres anderledes, end du plejede? Beskriv sammenbruddet/sammenbruddene, og hvad du lærte af det.



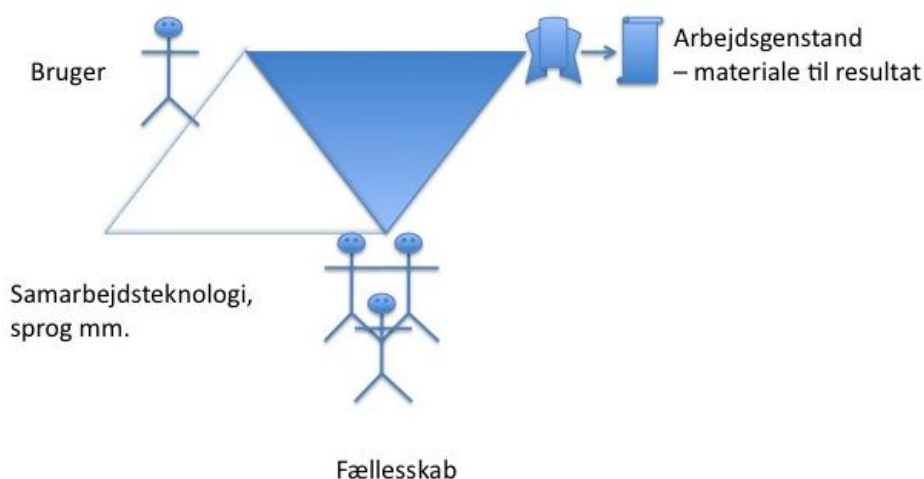
Figur 2. Redskaber

Hvor er arbejdsgenstanden?

Arbejdsgenstanden findes i nogle tilfælde *uden for* computeren, i andre *inde i*, mens den i endnu andre tilfælde har både en *digital eksistens* og en *fysisk*. Rejseplanen.dk kan fx ses som et redskab til at finde en konkret bus. Bussen findes jo som sådan ikke inde i computeren eller smartphonen, i modsætning til regnearket, som ikke rigtigt er noget i en udskrift. Et Word-dokument har til gengæld begge eksistenser, og det er et kvalitetskriterium, at de to ligner hinanden. Det er essensen af What-You-See-Is-What-You-Get-tankegangen.

Samarbejde

Som nævnt er menneskelig aktivitet i bund og grund social. Vi *samarbejder*, når vi bygger huse, skriver projektopgaver eller laver IT-systemer. Sproget er den grundlæggende mediering af menneskers samarbejde, men også elektronisk post og gruppeteknologier som First Class og google docs har denne funktion. For at blive i begrebsrammen, *medierer sådanne teknologier relationen mellem den enkelte og det fællesskab* hvori en eller anden opgave udføres (Figur 3), og på samme måde som redskabet tilkalder hverken sproget eller andre samarbejdsteknologier sig brugerens opmærksomhed, så længe brugen går godt.

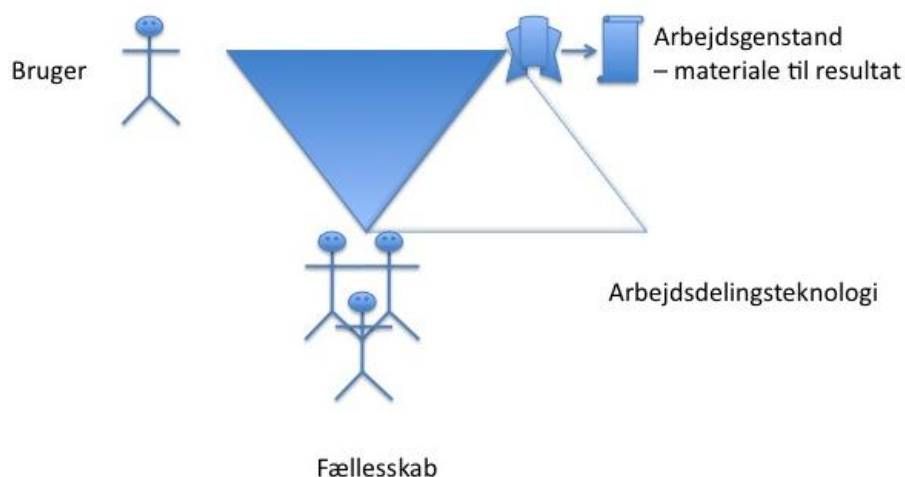


Figur 3. Samarbejde

Arbejdsdeling

Mange klassiske IT-systemer er primært udviklede med fokus på en anden mediering, nemlig *arbejdsdelingen hvorunder fællesskabet deler adgangen til arbejdsgenstand og resultat* (Figur 4). Planlægningssystemer medierer adskillelsen af planlægningsprocessen fra udførelsen af arbejdet. Samlebåndet sikrer at arbejdsopgaverne udføres i en bestemt rækkefølge med størst mulig præcision, på samme måde som IT fx kan benyttes til at foreskrive rækkefølgen af administrative processer, så som arbejdsgangen i forbindelse med ansøgning om SU. IT-systemer, der sørger for, at bestemte arbejdsprocesser udføres i bestemte rækkefølger og inden for givne tidsrammer, kaldes ofte for workflow-systemer.

Som med alle andre IT-systemer er brugerne ikke opmærksomme på systemerne som sådan, så længe brugen går godt. Problemet med mange workflow-systemer er dog fx, at de ofte ikke fungerer særlig godt, og tværtimod betyder, at brugerne skal lave ekstra arbejde, *fordi* de bruger systemet.



Figur 4. Arbejdsdeling

Workflow, et eksempel

Et eksempel på denne problematik er følgende: 2.u skal lave et projekt i fysik, og læreren har stærke ideer om, hvordan dette skal gøres, fordi hun dermed vil sikre kvalitet og progression. Hun stiller en række regler op, som skal følges, og bygger dem ind i øvelsesvejledning og rapportskabeloner:

1. Der skal være præcist tre elever i hver gruppe.
2. Gruppen skal arbejde 30 minutter ved hver af tre forskellige apparater.
3. Den må først begynde at skrive rapporten, når alle opgaver er udført.
4. Gruppen skal holde et møde, hvorefter elev A skal lave første udkast på den første øvelse, derefter overgive teksten til elev B. Elev B skriver første udkast på anden øvelse og overgive teksten til elev C, osv.
5. Rapporten skal afleveres præcist en uge efter, at projektet er startet.

Da projektet starter, er Andreas, Bilal og Camilla i en gruppe. De går i gang med den første del af øvelsen. Da de er ved at være færdige, kan de se, at gruppen ved det andet instrument har haft problemer, mens der tilsyneladende ikke er nogen på vej til at gå i gang med den tredje øvelse. Tilsyneladende vil de gøre alle en tjeneste ved at starte på den tredje øvelse, men det var jo ikke det, læreren sagde. De venter, og hele processen bliver stærkt forsinket af fumlegruppen. De overvejer, om de kan begynde at skrive rapporten, mens de venter, men heller ikke det er iflg. kravene. De kommer sent ud af fysiklokalet, og næste dag, da de skal skrive, er Bilal syg. De to andre har travlt på dag 3, så de får lyst til at starte på Bilals del i stedet for at gøre, som læreren siger. Bilal kan jo bare lave noget andet dag 3, tænker de, men læreren siger nej. Nu må alle vente til Bilal er rask igen, og til sidst har gruppen bare meget travlt med at blive færdige, og rapporten bliver ikke så god, som de havde tænkt sig, desværre.

Diskussion: Hvad vindes der, og hvad tabes der ved denne stramme strukturering af arbejdsprocessen? Hvorfor er det at "arbejde efter reglerne" en effektiv måde ataktionere på en arbejdsplads (eller for den sags skyld et gymnasium)?

Måske er det svært at se for sig, at en fysiklærer er så ufleksibel, som det er beskrevet. I dette eksempel kunne lærer og elever godt have været mere fleksible, men det er straks sværere, hvis reglerne er lagt ind i IT-systemet, så eleverne først kan komme til

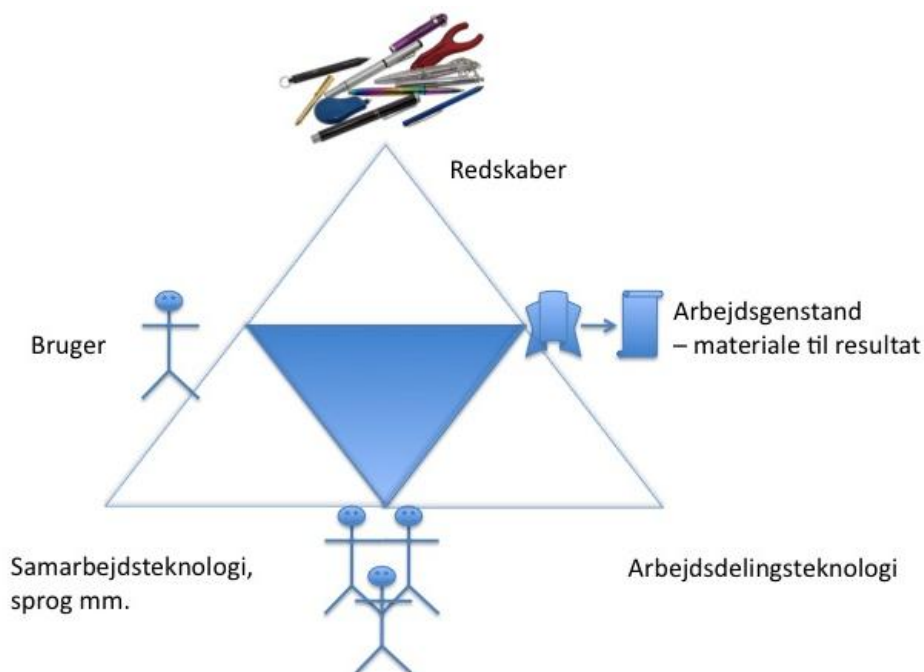
øvelsesvejledningen for det andet eksperiment, når de er færdige med det første, eller elev B først se teksten, når elev A har skrevet og leveret den videre.

Mange arbejdspladser har indført workflow-systemer, som, på trods af at formålet er effektivitet, gør noget, der svarer præcist til det beskrevne: Styrer arbejdsprocesserne i forhold til hinanden og gør det svært for de ansatte at improvisere eller bytte om på rækkefølgen af opgaver, selvom det ser ud til at være mest fornuftigt.

Summen af ovenstående er, at et enkelt IT-system oftest medierer som redskab, samarbejdsstøtte og arbejdsdeler på samme gang. I Figur 5 samler vi alle elementerne fra de forrige del-analyser i, hvad vi vil kalde *trekantsmodellen*.

IT som aktiv eller passiv mediering

Nogle IT-systemer er *aktive* udvidelser af menneskets færdigheder og handlemuligheder, i den forstand at de overtager og automatiserer noget, som før var menneskelige rutiner. Robotter og en del maskiner er gode eksempler på dette. Andre er *passive*, idet de primært understøtter og udvider handlemuligheder, som vi havde før: E-mail har gjort det muligt at kommunikere hurtigere over afstand, Facebook understøtter andre måder at have kontakt med vore venner, osv.



Figur 5. Den samlede trekantsmodel

Diskussion: Argumentér for følgende: Facebook er også en aktiv udvidelse, idet den samler og præsenterer vore venners gøren og laden på måder, som gør det muligt for os at have daglig kontakt med mange flere, end hvis vi fx skulle ringe eller SMS'e til dem alle. Hvad taler imod argumentet?

Modsætninger og sammenbrud som forståelsværktøj

Oftest er det endda *modsætningerne* mellem medieringen som redskab, arbejdsdeling hhv. samarbejde, som er interessante, hvis vi skal forstå, hvorfor systemerne ikke bare bruges glat og uden *sammenbrud*.

Et eksempel på dette er brugen af chat og et tekstbehandlingsprogram som MS Word i fælles rapportskrivning:

Word er *redskabet*, hvori gruppemedlemmerne skriver deres bidrag til rapporten, og så længe vi kun fokuserer på den enkeltes skrivning, går det nogenlunde problemfrit. Word og lignende tekstbehandlingsredskaber har de fleste lært og kan bruge, i alt fald til simple noter og rapporter. Men mange vil opleve *sammenbrud*, hvis de skal sætte billeder eller tabeller ind. Pludseligt er man ikke længere i gang med at nedfælde sine tanker eller data, men med at kæmpe med et billede, der hopper rundt i dokumentet.

Chat-programmer som MS messenger, Skype eller Facebooks chat, understøtter *samarbejdet* på den måde, at man kan se, hvilke andre gruppemedlemmer der er aktive på et givet tidspunkt, og skrive med dem, hvis man har spørgsmål mm. Man kan sende rapporten frem og tilbage, men alle gruppemedlemmer kan ikke arbejde på hele rapporten på én gang. De diskussioner, man har på chat, kommer heller ikke ind i rapporten, hvilket kunne være en hjælp i nogle situationer. Derudover opfører selve dokumentet (i Word) sig sommetider forskelligt på forskellige computere, hvilket gør at fx sideombrydning kan være forskellig. Derfor kan det være svært at kommunikere om bestemte dele af dokumentet, i stil med ”se nederst på side 4”) (et *sammenbrud*).

Fordi Word grundlæggende kun tillader, at én bruger skriver ad gangen, er det nødvendigt at aftale, hvem der gør hvad, hvornår (*arbejdsdeling*), og evt. dele dokumentet op i dele. I denne slags situationer er det kolossalt svært at holde styr på, hvad der er den nyeste udgave af dokumentet, og hvem der har lige netop den på sin computer. Kriser (*sammenbrud*) opstår, når gruppemedlemmer kommer til at arbejde i gamle versioner, og chat er ikke altid tilstrækkeligt til at sikre sig mod disse problemer.

Som vi kan se, er det ikke let at sige om Word og chat er redskaber, samarbejdsteknologier eller arbejdsdelingsteknologier. De er faktisk begge lidt af alle tre, men *de tre roller forvaltes ikke lige godt og spænder ben for hinanden*.

Diskussion: Hvilke af disse problemer løses med fx Google docs, og hvilke nye problemer kan I se i stedet?

Aktivitetsnetværk

Som det fremgår af eksemplet, foregår en aktivitet som rapport-skrivning ikke i isolation. Den er knyttet sammen med andre aktiviteter, som omgiver rapportskrivningen, hvad vi kalder *aktivitetsnetværk*. Fx har læreren måske lavet en rapportskabelon i Word, som eleverne bruger. Vi står over for et nyt *fællesskab*, en ny *bruger*, men samme *redskaber*. Lærerens *resultat*, skabelonen, bliver samtidig i sig selv et *redskab* for eleverne i deres rapportskrivning.

Læreren kan også bestemme, at der skal være en bestemt *arbejdsdeling* i gruppen, eller at gruppen fx skal *samarbejde* ved at skrive dagbog (som illustreret i workflow-eksemplet). Måske er rapporten en fysikrapport, og eleverne har arbejdet med et eksperiment vha. andre *redskaber*. De har produceret nogle data (*resultat*), som er en del af *materialerne* for rapportskrivningen osv. Nogle elever har måske ligefrem været på kursus i at bruge MS Word, mens andre har lært sig det undervejs i samspil med andre og gennem simpel ”trial and error”. Desværre kan kurset være lavet forkert, så eleverne ikke har lært sig at sætte tabeller ind, hvilket stiller kursusaktiviteten i *modsatning* til rapportskrivningsaktiviteten. Måske kan fysik-

instrumentet ikke levere data, der kan sættes direkte ind i rapporten - endnu et eksempel på en modsætning.

Generelt må vi forståelsesmæssigt se på det netværk af *andre aktiviteter*, som omgiver den, der er i fokus. Dette netværk medvirker til aktiviteten i fokus ved at *producere* nye redskaber, samarbejdsteknologier, arbejdsdelingsteknologier, materialer, kvalificere brugerne fx ved at producere manualer eller lave kurser for dem i de konkrete IT-systemer (nye redskaber, samarbejdsteknologier, arbejdsdelingsteknologier). Vi må også undersøge, når samspillet ikke er glat. Forestillingerne om modsætninger og sammenbrud kan bruges til at forstå problemerne.

Vi kan vælge analytisk perspektiv og fx analysere elevernes opgaveskrivning alene eller det fællesskab, der også omfatter læreren og skabelonerne. Vi kan evt. sammenligne de to analyser og se, hvilket perspektiv der giver den mest brugbare indsigt.

Samspillet mellem IT-systemer og menneskelig adfærd er komplekst og dynamisk

Opsummeret er samspillet mellem IT-systemer og menneskelig adfærd komplekst og dynamisk. Denne tekst har givet nogle begreber, som kan hjælpe med at pille den kompleksitet fra hinanden og forstå dynamikken. Det er vigtigt at indse, at det faktisk er svært at forudse, hvordan samspillet vil være, før IT-systemerne tages i brug. Dette er en klar udfordring til dem, der designer og udvikler IT-systemer. Men samspillet ændrer sig også hele tiden, både for den enkelte og på et mere overordnet plan. Facebook blev oprindeligt lavet, for at nogle studerende, der sad isolerede på deres studenter-værelser på et enkelt universitet, kunne udveksle venskaber. Nu er det på godt og ondt stedet, hvor vi checker politikernes og de kendtes gøren og laden ud; hvor unge piger kommer i uføre, fordi de snakker med grimme ældre mænd, og hvor forelskede par inviterer vennerne til bryllup, og familier planlægger deres fælles sommerferier: Uforudseeligt og dynamisk, men også noget, som vi alle sammen er med til at forandre gennem vores daglige brug.

Den første del af denne note har fokuseret på, hvordan man kan forstå samspillet mellem IT-systemer og menneskelig aktivitet.

Når du er kommet hertil, kan vi opsummere følgende:

1. Du vil være i stand til at give eksempler på, hvordan IT-systemer har betydning for og påvirker menneskelig aktivitet.
2. Du har lært nogle begreber, som du kan benytte til at forstå disse sammenhænge – aktivitet, mediering, arbejdsgenstand.
3. Du har opnået en nuanceret forståelse af sammenhængene i forhold til de forskellige måder, hvorpå IT medierer menneskelig aktivitet, og i forhold til kompleksiteten og dynamikken i dette samspil – sammenbrud og netværk.
4. Du har ligeledes set eksempler på, hvordan brugen af IT-systemer er uforudsigelig.
5. Du forstår, hvordan IT-systemer er bygget ud fra nogle forskellige grundlæggende antagelser om automatisering, kvalitet, model og dynamik.

At analysere og vurdere samspil mellem IT-systemer og menneskelig adfærd

I det ovenstående har vi haft fokus på nogle eksempler, der illustrerer samspillet mellem IT-systemer og menneskelig adfærd. En model og nogle begreber til at forstå dette samspil er blevet introduceret, og vi går videre med egentlige analyser og vurderinger i det følgende. Som påpeget er samspillet dynamisk, og det fordres derfor også, at analyser og vurderinger indfanger denne dynamik. *Vi skal kunne forstå og sammenligne for at kunne forandre.*

I det foregående har vi primært fokuseret på den generelle forandring, som drives af ny forskning og generel samfundsmæssig udvikling. Vi har også set på, hvordan forandring foregår i brug, når mennesker gør teknologien til deres egen, som enkeltbrugere og som fællesskaber. Vi har derudover set på forandring i form af konkrete udviklingsprocesser, uden dog at arbejde med metoder til at inddrage brug, og brugere i udviklingsprocessen.

I det følgende vi fokuserer på samspillet mellem menneskers brug af IT og de udviklingsprocesser, hvori IT-systemerne skabes.

Brug og udvikling

I trekantsmodellen er brug og udvikling eksempler på to forskellige (slags) aktiviteter. I brug er IT-systemet *det medierende* redskab, som bearbejder materiale eller den medierende understøttelse af samarbejde eller arbejdsdeling. Udvikling er derimod en aktivitet, som har IT-systemet som *arbejdsgenstand*. IT er materialet, og resultatet. Afhængig af behov, kan vi definere udvikling og brug ved denne forskel: Udvikling har IT-systemet som arbejdsgenstand, i brug er IT-systemets rolle mediering. Eller, vi kan se på konkrete aktiviteter, hvor IT-systemer indgår, for at forstå i hvilket omfang, de kan ses som brug hhv. udvikling. Fx: Når Ditte bruger Excel til at holde styr på klassens ekskursionsregnskab, er det måske først og fremmest brug, selvom hun i starten faktisk satte regnearket op, så hun har de rækker og søjler hun skal bruge. Når Esben, Frida og Gunhild laver et regneark, som hele klassen kan bruge til et bestemt fysikforsøg, er det først og fremmest med regnearket som arbejdsgenstand, og dermed udvikling.

Diskussion: Man kan sagtens argumentere for, at Excel medierer begge aktiviteter. Diskutér hvilken forskel det gør om man stiller krav til Excel med den ene hhv. den anden brugssituation for øje.

Brugeren

De mange mennesker, der indgår i brugsaktiviteter, kaldes ofte *brugere*, selvom de i hovedsagen er gymnasieelever, lærere, reklamefolk, postbude eller tele-teknikere. Hoveddefinitionen af *brugeren* er, at han eller hun benytter IT i sit daglige virke uden primært at udvikle den. Det på trods af, at alle brugere medvirker til udviklingen af IT gennem brug.

Det er en vigtig erkendelse for udviklerne, at brugerne ikke er ligesom dem selv. Brugere har *andre forudsætninger*, de har lært noget andet i andre fællesskaber. De har i den sammenhæng *tavs og eksplicit viden* om de arbejdsgenstande, de omgiver sig med, og kvaliteten af de typer af resultat eller produkt, de skaber. De har måske

andre erfaringer med IT, ligesom de har erfaringer med andre redskaber og samarbejdsformer, som udviklerne ikke umiddelbart kender.

For at kunne lave god IT-udvikling er det nødvendigt, at udviklerne får adgang til denne slags forudsætninger, og det vil vi fokusere på nedenfor, når vi ser på brugerorienterede udviklings-metoder.

Sammenligning af brugsaktiviteter

Analytisk er det nyttigt at se på forskellige brugs-fællesskaber og forskellige forudsætninger for brug. Fx kan det være nyttigt at analysere forskellen mellem brugen af Facebook mellem teenagere i 8. klasse og en gruppe pensionister, som i fællesskab er ved at arrangere et flod-krydstogt. Eller forskellen mellem brugen af e-mail på et advokatkontor og i en foreningsbestyrelse. Trekantmodellen (figur 5) kan hjælpe os med at se på forskelle og ligheder.

Med udgangspunkt i trekantmodellen kan vi

1. Undersøge brugernes forskellige baggrundsrutiner og fællesskaber, herunder hvor brugerne har de erfaringer fra, som benyttes i den brugsaktivitet, vi ser på;
2. Undersøge hvilke andre redskaber, brugerne benytter sig af, hvilken arbejdsdeling og samarbejdsformer, der benyttes (se yderligere næste).

Hands-on erfaring

Hvis brugere skal forholde sig til nye teknologier og til, om disse kan hjælpe dem i en eller anden aktivitet, må både deres eksplicite og tavse viden i brug. Som diskuteret tidligere, er en beskrivelse af den fremtidige brug ikke tilstrækkelig til, at brugerne kan aktivere deres tavse viden. Det er nødvendigt at *prøve sig frem*, aktivere kroppen eller få hands-on erfaring, som man siger på engelsk. Hvis vi skal analysere brug, er det derfor nødvendigt at *se på brugen, mens den sker*. Brugere kan ikke sidde i et mødelokale og forestille sig, hvordan de i fremtiden skal arbejde, medieret af den nye IT. De må have noget mellem hænderne, som de kan prøve i brug eller brugslignende situationer.

Hvis vi designer nye informationsteknologier er *prototyper* af forskellig art gode hjælpemidler til at lade brugerne afprøve den *fremtidige brug* gennem brugslignende situationer.

Partnerskab og læringsrelation

Den anerkendte analyse- og design metode, Contextual design, taler om, at det er nødvendigt for den, der analyserer og designer IT, at indgå et partnerskab med brugerne. I det partnerskab er brugeren læremester, og den, der analyserer brugsaktiviteten, er lærling eller perifer deltager i praksisfællesskabet.

Prototyper

Prototyper spiller en væsentlig rolle i at fastholde en fremtidig brug både for brugerne og for udviklerne. Brugere har behov for at kunne afprøve prototyperne i brugslignende situationer. Udviklerne har behov for nogle *konkrete mellemprodukter*, som de kan dele på tværs af de mange kompetencer, der indgår i et udviklingsprojekt, og på tværs af den arbejdsdeling, der er.

Prototyper kan være tidlige *versioner* af det endelige IT-system, men er det ikke nødvendigvis.

Man kan skelne mellem prototyper, der går i *bredden*, ved at de dækker en stor del af det fremtidige IT-system, men uden at gå i dybden med detaljerne, og prototyper, hvor man har valgt en lille del ud for at gå i *dybden* med den. Udviklerne vælger altså undervejs præcist, hvad de har brug for at undersøge med en prototype. En prototype, der viser et skærmbillede, men hvor der intet sker, når man trykker på menuer og knapper, er en bred prototype, mens en prototype, der fokuserer på den bedste måde at lave en søge-funktion (uanset resten af IT-systemets funktion og udseende), er et eksempel på en dyb prototype.

Gennem *iteration* kan udviklerne afsøge forskellige løsningsmuligheder, og brugerne kan få lejlighed til at prøve sig frem. Iteration betyder, at processen gentager sig med det formål at komme tættere og tættere på et færdigt IT-system. Iteration står i modsætning til en design-proces, hvor man forestiller sig, at man starter med at afdække alle krav til det fremtidige system eller modellere brugssituationen fra ende til anden (som vi så på først i denne note). Partnerskabet mellem udviklere og brugere går således to veje: Udviklerne lærer af brugerne og omvendt, som vi vil se i eksemplet nedenfor.

Mange moderne systemudviklingsmetoder, inklusiv dem der kaldes *agile*, baserer sig på, at alt udviklingsarbejdet foregår i hurtige iterationer, som hver tilføjer i bredden og i dybden til det, der allerede er bygget. Ved at arbejde iterativt, i stedet for at lave en stor model af hele det fremtidige system først, er tanken, at man får elimineret fejl og uhensigtsmæssigheder med det samme.

Konkrete brugerorienterede metoder

I det følgende præsenteres nogle metoder, som man kan bruge, når man arbejder med at forstå IT i brug, hvad enten det er for at *analysere* en brugsaktivitet, som allerede eksisterer, eller det er for at *designe* og fokusere på fremtidig brug.

Kontekstuelle interviews

I modsætning til traditionelle interviews foregår kontekstuelle interviews i brugsaktiviteten, som en del af det partnerskab og den læringsrelation, der er mellem bruger og udvikler. Som udvikler/analytiker analyserer man den konkrete brugsaktivitet ved at følge brugeren i hendes daglige aktivitet som perifer deltager. Man tager notater undervejs, og interviewene består i, at man *spørger ind til, hvad der foregår, når der sker noget*, eller når brugeren gør noget, som man ikke forstår eller ikke kan følge med i. Det er altså analytikerens fornemste opgave at se og lytte kombineret med, at man spørger, når der er noget, man ikke forstår. For at kunne bearbejde indtrykkene senere og lave den egentlige analyse er det vigtigt at tage notater undervejs.

Man kan også tage billeder undervejs, som kan bruges i analysen. Er man en gruppe, der arbejder sammen, fordeler man arbejdet ved, at hver deltager følger sin eller sine brugere, afhængig af hvor lang tid man har til rådighed.

Helle, Isabella og Jonas arbejder på et projekt, hvor de undersøger skolens regnskabssystem. De starter med at aftale, at Helle skal tilbringe nogle timer sammen med rektor for at finde ud af, hvilken rolle regnskabet og regnskabssystemet egentlig spiller for hende. Isabella og Jonas har hver en dag med skolens

regnskabsmedarbejder, Karin. De aftaler, at det er OK, at de tager billeder undervejs, og at disse billeder kun bruges i den efterfølgende analyse.

Jonas møder op på kontoret den aftalte morgen, rustet med notesblok og kamera. Han fortæller Karin, at hun bare skal gøre, som hun plejer, og at han vil se på, hvad hun laver, og stille spørgsmål undervejs. Allerede da Karin starter sin computer, bliver Jonas nysgerrig – hvad er det for en mærkelig opsætning, hun har på sit skrivebord? Han spørger og noterer, og Karin giver svar. Karin har nogle bilag, der skal indberettes, og hun går i gang. Jonas opdager, at Karin har nogle helt bestemte rutiner, som hun benytter til dette, og han spørger, hvor de kommer fra. Karin fortæller, hvordan de altid har arbejdet på denne måde på skolen, bortset fra at da de fik nyt regnskabssystem for et par år siden, blev de nødt til at begynde at scanne bilagene ind i computeren, så nu må hun også hen til scanneren. Derfor er hun begyndt at samle flere bilag sammen ad gangen, før hun går i gang. Jonas synes, at han lærer mange ting om regnskab, som dagen går. Han ser nogle ting, som han synes er underlige, mens der også er mange ting, som Karin har en god forklaring på. Da dagen er ved at være omme, kommer de igen forbi skrivebordsopsætningen, og Jonas får lyst til at spørge Karin, om det ikke var lettere, hvis hun gjorde det på en anden måde. Han viser, hvad han mener. Karin bliver begejstret og siger, at det da var en meget bedre måde. På den måde fik de begge noget ud af dagen, udvikleren har lært meget af brugeren, men brugeren har også fået noget igen.

Analysearbejdet består i næste skridt i, at man gennemser sine noter og skriver de væsentligste observationer over på gule post-it notes. Én observation pr seddel. Med væsentlig menes her både mest typisk, mest hyppig, mest overraskende, og hvad der ellers gør, at udvikleren finder observationen interessant. Herefter følger et sorteringsarbejde, som oftest gøres i en gruppe sammen med andre, der også har fulgt en lignende brugsaktivitet:

Hvis man er flere, skiftes man til (kort) at præsentere en observation for hinanden, mens man sætter den gule seddel op på en tavle. Efterhånden som man sætter sedlerne op, må man godt begynde at gruppere dem, men tanken er, at når alle sedler er på tavlen, begynder man at samle emner, der ligner hinanden. Man prøver sig frem, flytter sedler, argumenterer, indtil der tegner sig nogle grupper, der virker stabile. Denne sortering kan også laves af én analytiker ad gangen. Har man billeder, kan de benyttes på samme måde.

Næste skridt er at navngive disse grupper. Også her prøver man sig frem, indtil man er tilfreds. Når grupperne er blevet passende navngivet, er det op til én (eller et par) af deltagerne at sammenskrive en rapport, hvor analyserne sammenfattes under de valgte overskrifter.

Helle, Isabella og Jonas har hver for sig bearbejdet deres noter ved at skrive de vigtigste ting, de så, op på gule sedler. De har også printet de billeder, de tog, og nu mødes alle tre ved en tavle. Isabella lægger ud med at fortælle om en situation, hvor Karin skulle tilbage i regnskabet og finde en bestemt postering. Det var overraskende besværligt, synes Isabella. Jonas melder ind med, hvordan Karin samler bilag sammen – typisk for den dag, han var med hende. Helle fortæller, hvordan rektor forsøgte at få styr på nogle tal om skolens bogindkøb, og hvordan rektor og Karin havde hjulpet hinanden med at få det overblik. Osv. Osv. Hver gang sætter de sedler og billeder på tavlen.

Efter en tid løber de lidt tør for flere observationer, og de går over til at sortere: Helle flytter 3-4 sedler sammen og siger, at her er noget om overblik. Jonas ser på to billeder og et par sedler og forsøger at formulere en overskrift på ”det her er noget, der har at gøre med, hvor meget Karin skal ind og ud af systemet”. De finder en gruppe, der handler om samarbejde med andre. Undervejs opgiver de nogle af grupperne igen, mens andre får nye navne og tilføjet flere sedler.

Efter et par timer synes de, at de har skabt sig et meget godt fælles overblik over, hvordan man laver regnskab på deres gymnasium.

Idé-prototyper

Baseret på resultaterne af bl.a. kontekstuelle interviews går man i gang med at brainstorme om den fremtidige IT. I denne slags idé-generering er det vigtigt på den ene side at holde tankerne åbne og på den anden side at arbejde konkret. Hertil er både video og papir nyttigt:

Papir-atrapper kan bruges til at fastholde skærbilleder, og man kan afprøve simpel interaktion fx ved at lade én af deltagerne ”lege computer,” så man lader som om, der sker noget, når der trykkes på knapper osv.

Man kan også benytte video, ved at man skridt for skridt (såkaldt stop-motion) optager en idé og afspiller den for hinanden. Både papirattrapper og videoprototyper kan udvikles sammen med brugerne og benyttes til at illustrere over for brugerne, hvordan den fremtidige brug vil være.

Tilbage hos Helle, Isabella og Jonas er de kommet til idé-genereringen. De brainstormer lidt om de skærbilleder, som Karin benytter, og tegner lidt på papir – kunne man lave noget bedre? Pludselig siger Helle: ”Hvad nu hvis rektor og andre selv skannede bilag ind?” Isabella tager sin telefon frem: ”Man kunne måske bruge telefonens kamera?” Hun holder telefonen op foran et stykke papir. De bestemmer sig til at prøve ideen. De forbereder noget, der ligner en faktura. Helle tager sin telefon frem og agerer kameramand. Isabella leger rektor. Hun beder Helle om at gøre klar og begynder at tale, mens hun tager telefonen og holder den op foran fakturaen. Undervejs siger Jonas: ”Stop, hvad nu hvis du i stedet...?” Helle synes, at det er en bedre ide, og de tager situationen om. Senere ser de begge ideerne igennem og bestemmer sig til at holde fast i den første idé.

Videoscenarier

Fra de kontekstuelle interviews har man i udviklergruppen materialet til at forstå, hvad der er en realistisk brugssituation. Fra idé-prototyperne, og evt. andre slags prototyping, har man en teknologiske løsningsidé, som gerne skal formidles til brugerne på en måde, hvor brugerne får en mulighed for at opleve fremtiden.

Det handler om at fortælle en brugshistorie på en måde, så tanken bag ideen eller prototypen fremstår klarest muligt for brugerne. Ideen er at give brugerne mulighed for at diskutere løsningen, evt. ved at man problematiserer den. Man kan gå så vidt som at fortælle den samme historie som god og glad og som problematisk og trist. Det handler under ingen omstændigheder om at lave salgsmateriale for sin idé.

Helle, Isabella og Jonas arbejder videre med, hvordan de kan præsentere deres designidé for rektor og Karin. De diskuterer brugssituationen grundigt igennem og laver en struktur, så de ved hvordan de vil fortælle historien. Så laver de en kort video, hvor Isabella og Jonas gennemspiller situationen, mens Helle filmer.

Den teknologiske verdenshistorie indeholder berømte eksempler på sådanne videoscenarier, som er lavet i relativt dyre produktioner. For de fleste projekter kan billigere hjemme-redigerede videoer sagtens gøre det. Det mest kendte eksempel på et videoscenarie, Apples "Knowledge Navigator" fra 1987, har på det seneste fået ny aktualitet, fordi Apple har introduceret SIRI på deres iPhone 4s. Den originale video kan findes på YouTube eller gennem Wikipedia.

Afprøvning i brug: Højtænkning

Når man har en prototype, som er teknisk stabil, kan man lade brugerne afprøve den i en realistisk brugssituation. Materialet til at forstå, hvad der er en realistisk brugssituation kan fås fra de kontekstuelle interviews. Hoved-idéen er, at man lader en bruger, eller to sammen, gennemgå en brugsaktivitet eller løse en opgave, mens de taler sammen og *tænker højt*. Det er ikke så let, som det lyder, at tænke højt, og derfor er det ofte en fordel at sætte to brugere sammen, så de kan tale sammen. Men uanset er det vigtigt, at brugerne får at vide, at de skal tænke højt og snakke sammen så meget som muligt.

Udvikleren/analytikerens er tilstede, men griber mindst muligt ind i det, brugerne foretager sig. Hun må gerne hjælpe, hvis der opstår tekniske problemer, som gør det umuligt at komme videre, men generelt skal brugerne klare sig selv, for højtænkning tjener præcist til at få indblik i, hvordan brugerne forholder sig til teknologien. Det er klart, at der sagtens kan være mange udviklere i et projekt, og det er næppe hensigtsmæssigt, at de alle er tilstede og kigger på én eller to brugere. Derfor vil der oftest kun være én analytiker tilstede, som først og fremmest er fluen på væggen og dokumenterer højtænkningen uden at gribe ind.

Som tilstedeværende analytiker dokumenterer man processen ved fx at optage brugen på video, gerne med flere kameraer, så man både kan se skærm, tastatur og brugernes ansigter og krop. Der tages noter undervejs, og en væsentlig del af analysen er at gennemse videomaterialet eller anden dokumentation, ofte mange gange. Udvalgt materiale kan bruges både som udgangspunkt for diskussion i udviklergruppen og for flere spørgsmål til brugerne.

Diskussion: Da det i udgangspunktet er meningen, at den prototype, man prøver, skal køre af sig selv, kan man også benytte højtænkning til at skaffe information om den daglige brug af en færdigudviklet teknologi. Planlæg hvordan I vil bruge højtænkning til at forstå mere om, hvordan skolens lærere benytter FirstClass. Hvad vil I kunne lære af en sådan evaluering?

Diskussion: Planlæg, hvordan I vil bruge kontekstuelle interviews hhv. højtænkning til at forstå mere om, hvordan jeres kammerater bruger smartphones. Hvor er styrkerne og svaghederne med de to metoder? Hvilken metode vil I benytte til hvad? Hvad vil I kunne lære af en sådan evaluering?

Den anden del af denne note har bidraget til følgende:

1. Du har fået introduceret begreber og metoder til at analysere og vurdere IT-systemers betydning for menneskelig aktivitet.
2. Du er blevet præsenteret for begreber, der hjælper med at forstå dynamikken i samspillet mellem IT-systemer og menneskelig aktivitet.
3. Specifikt er du blevet introduceret for begreber til at forstå og vurdere samspillet mellem udvikling og brug.
4. Du kan forklare hvorfor iteration, prototyper, hands-on erfaring og læring er vigtigt for udviklingsprocesserne, og hermed hvorfor brugerorienterede teknikker er væsentlige.
5. Du har set eksempler på og kan anvende sådanne brugerorienterede teknikker.

Design-projekt

I denne note er der mange eksempler på projekter, som man kan gå i gang med eller finde inspiration fra.

Af andre eksempler på projekter kan nævnes:

1. Bestilling af billetter til skolefester vha. mobiltelefonen
2. Ny billetautomat til DSB
3. Nyt, forbedret mitSU

Hvis man skal benytte teknikkerne til et design-projekt for i sidste ende at konstruere eller om-konstruere et IT-system eller produkt er her nogle anbefalinger:

1. Arbejdet laves i grupper på 3-4 elever.
2. Projektet laves i iteration, så der undervejs bygges mindst 2 prototyper (evt. den ene som videoscenarie og den anden som software-prototype).
3. Der laves yderligere kontekstuelle interviews; mindst 2 idé-prototyper; og hvis muligt en højtænkningsevaluering.
4. I et B-niveau-forløb er dette hovedparten af undervisningen i 2-4 uger (afhængig af de tekniske krav, og om der laves højtænkning).
5. Læreren er metodemæssig vejleder, og der laves en projektplan som det første.
6. Som afslutning viser eleverne projekterne frem for hinanden.

Læs mere:

Benyon, D. (2010). Designing Interactive Systems

Ciborra, C (2002). The labyrinths of information: challenging the wisdom of systems

Dourish, P. & Bell, G. (2011). Divining the digital future

Lave, J. & Wenger, E. (2003). Situeret læring og andre tekster

Mackay, W (2002). Videomateriale om video-prototyper (på engelsk):

<http://www.lri.fr/~mbl/VideoForDesign/>

Morgan, G. (2006). Images of organisation