

## Kapitel 1

# Problemer på brugerkurser

Stort set alle medarbejdere i danske virksomheder har prøvet at være på brugerkurser, de fleste endda rigtig mange gange. Desværre er der stadig hvert år hundredtusindvis af brugere, som efterfølgende finder ud af, at de ikke kan bruge systemet i praksis, når de kommer tilbage til deres dagligdag. Jeg var for nylig til møde med en leder af en grafisk afdeling, som netop havde været på et femdages leverandørkursus i et nyt grafiksistem, som skulle indføres i hans afdeling. Hans spontane kommentar var, at han efter de fem dage ikke var i stand til at slå en streg!

Når virksomhederne alligevel fungerer i det daglige, er det ikke nødvendigvis på grund af kurserne, men på trods. De mange brugere, som ikke får nok ud af kurserne, må møjsommeligt selv lære sig systemerne, når de er tilbage på deres egen pind, ofte med betydelig hjælp fra den lokale superbruger. Dermed har den egentlige idé med brugerkurserne spillet fallit, nemlig at accelerere brugernes læreproces.

Dette kapitel beskæftiger sig med, hvilke problemer brugere og undervisere løber ind i, og hvorfor problemerne opstår. Overordnet fokuseres der på tre faktorer:

- Pædagogikken
- Organisatoriske aspekter
- Rammerne

Når der bruges forholdsvis mange sider på problemerne, er det, fordi de løsninger, som resten af bogen beskæftiger sig med, kræver, at man virkelig har indset problemernes natur, ellers har de det med at dukke op igen, når man mindst venter det.

## Hands On modellen

I dette kapitel fokuseres der på de problemer, der traditionelt er forbundet med IT-brugerkurser. Analysen tager udgangspunkt i den såkaldte Hands On undervisning, som typisk består af to faser, som gentages igen og igen:

### Oplægget

I denne fase gennemgår underviseren skærbilleder, menuer mv. via "kanonen". Varigheden er typisk 10-20 minutter.

### Opgavefasen

Idéen er, at brugerne i denne fase skal bruge den viden, de har fået i første fase, og at underviseren skal hjælpe, hvis de har problemer.

Fig. 1.1

Den grundlæggende idé er helt rigtig, fordi den giver fremragende muligheder for vekselvirkning mellem teori og praksis. I modsætning til megen anden undervisning er der kun 10 minutter, fra man har hørt om noget, til det skal anvendes.

Modellen bygger naturligvis på, kursisterne skal bruge det, de har hørt om i oplægget, til at løse praktiske opgaver. Problemet er bare, at brugerne i masser af tilfælde netop *ikke* har lært det, som underviseren har vist dem i gennemgangen af skærbillederne via "kanonen". Derfor går mange af brugerne hurtigt i stå med opgaveløsningen, indtil de har fået hjælp af underviseren. Dette gentager sig igen og igen, og resultatet er dels frustration, dels at brugerne ofte kan alt for lidt, når de kommer hjem fra kurset. Samtidig giver det en særdeles stresset dagligdag for underviserne.

Brugeruddannelse i praksis

At brugerundervisningen ikke fungerer, har mange konsekvenser på kort og lang sigt:

- Der skal bruges flere ressourcer til brugeruddannelsen i form af antal kursusdage, undervisere mv.
- Der skal bruges mange ressourcer på at hjælpe brugerne i dagligdagen
- Brugere udnytter systemernes potentiale alt for lidt
- Brugernes generelle IT-kompetence udvikles ikke
- Risikoen for modstand mod ændringerne stiger

Hvis nu moderne systemer var meget, meget vanskelige og brugerfjendske, var der ikke noget underligt ved, at brugerne havde svært ved det. Men de fleste vil nok være enige i, at nutidens IT-systemer er særdeles brugervenlige i forhold til tidligere. Dermed ikke være sagt, at de ikke kunne være meget bedre, for det kunne de fleste.

Pointen er imidlertid, at det ikke kan være systemernes sværhedsgrad, der er hovedårsagen. Da vi heller ikke kan skyde skylden på brugerne, må analysen rette sig mod pædagogikken og rammerne.

## **Pædagogiske problemer**

I det følgende trækkes de typiske problemer i forbindelse med brugerundervisningen frem.

Naturligvis er det ikke alle problemerne, som dukker op hver gang, men de fleste brugere og undervisere vil nok kunne nikke genkendende til de fleste.

### **Hukommelsesproblemet**

Det mest åbenlyse problem er, at brugerne ikke kan huske de informationer i form af skærbilleder, menuer, felter mv., som underviseren netop har gennemgået. Da den grundlæggende idé er, at kursisterne skal bruge det, som de netop har hørt, er dette problem naturligvis temmelig afgørende.

Grundene til hukommelsesproblemerne er mange. For det første er der alt for meget huskestof til, at man kan huske det, blot fordi man har hørt det en gang. Men selv om man gør noget ved *stoftrængslen* og laver meget små oplæg, vil brugerne stadig have svært ved at huske detaljerne.

Hjernen er nu engang ikke en båndoptager, som blot kan "indspille" lange kæder af hukommelsesstof. Hvis vi virkelig vil have, at brugerne skal lære stoffet udenad, skal der skrappe metoder til, således som man brugte det i skolen i "gamle dage" (da jeg gik i skole). Det var noget med remser og gentagelse, gentagelse og atter gentagelse. Dette skal bestemt ikke opfattes som en anbefaling, blot en konstatering af, at det ikke er realistisk at brænde huskestof ind i hjernevindingerne ved blot at gennemgå detaljerne én gang.

Det er i øvrigt interessant, at selv øvede brugere heller ikke kan *huske* detaljerne udenad. Kommandoerne sidder i stedet "i fingrene", og da det er der, de skal bruges, er det jo meget hensigtsmæssigt.

### **Overblikproblemet**

Når man skal lære at anvende moderne systemer, er det sværeste - og vigtigste - normalt at skaffe sig et overblik over systemets forskellige dele, og ikke mindst de "ruter" man som bruger skal følge. Når undervisere bruger kanonen, føler de normalt selv, at de *giver* brugerne overblik. Problemet er imidlertid, at brugerne ikke *modtager* overblikket, tværtimod giver den sekventielle gennemgang af skærbilleder forvirring.

### **Forståelsesproblemet**

Indtil nu har vi set på de informationer og det overblik, som underviserne faktisk giver, men som kursisterne desværre ikke får. Men der er også noget, som underviserne alt for sjældent fokuserer på, nemlig at skabe en *forståelse* hos brugerne, som sætter dem i stand til at gennemskue de principper, som systemet og anvendelsen af systemet er opbygget efter, og som kan gøre dem til kompetente brugere. Når man ikke forstår principperne, drukner man nemt i detaljerne, og ens mulighed for at udnytte systemets muligheder indskrænkes kraftigt.

### **Miksede forklaringer**

Men selv om underviserne faktisk prøver at forklare principperne, mens de gennemgår skærbillederne, må de normalt bagefter konstatere, at brugerne alligevel ikke har opnået den ønskede forståelse. Årsagen er, at man benytter sig af såkaldt *miksede forklaringer*, dvs. at man på en og samme tid udsætter brugere for detaljer og principforklaringer.

Hvis man kommer med en principforklaring, mens kursisterne stirrer på et skærbillede fyldt med snesevis af detaljer, vil opmærksomheden være rettet mod detaljerne. Konsekvensen er, at brugerne ikke får fat i den principielle forklaring. Lidt efter har de glemt detaljen igen, men den har altså blokeret for, at de fik skabt en forståelse og et overblik.

### **Afhængighedsproblemet**

Hvis undervisningen løber ind i ovennævnte problemer, er resultatet, som allerede beskrevet, at de ikke selv kan finde ud af at løse opgaverne, hvorfor de skal have hjælp. Det er naturligvis frustrerende for kursisterne og stressende for underviserne, men nok så vigtigt er det medvirkende til, at mange brugere bliver afhængige af hjælp. Det kan være slemt nok på selve kurset, men

ofte vil det betyde, at de også i dagligdagen vænner sig til, at de i tide og utide skal have hjælp af kollegaer, superbrugere og supportere. Dette er ikke blot ressourcekrævende, men det betyder også, at der er en risiko for, at brugerne ikke i tilstrækkelig grad udvikler deres selvstændige kompetence.

### **Opsamling: Pædagogiske problemer**

Den pædagogiske metode bygger på, at underviseren gennemgår skærbilledsekvenser på kanonen med henblik på, at brugerne efterfølgende skal bruge informationerne, når de selv skal løse opgaver. Problemet er imidlertid, at kursisterne ikke får det, som underviserne giver. Underviseren giver en masse informationer, men kursisterne har dem ikke bagefter, når de skal bruge dem. Underviseren giver overblik via kanongennemgangen, men brugerne savner netop overblik bagefter.

### **Organisatoriske problemer**

Når en stor virksomhed indfører et nyt system, vil det som nævnt i introduktionen næsten altid medføre organisatoriske ændringer i form af ændrede roller, opgaver og forretningsgange. Den klassiske metode med gennemgang af skærbilleder stammer fra en tid, hvor man underviste i Word og Excel, og den tager slet ikke højde for de målsætninger, der er med et moderne brugerkursus, som skal geare medarbejderne til en ny dagligdag.

### **Perspektivproblemet**

Når man underviser skærbilledorienteret, er det selvsagt IT-systemet, som er i centrum. Og der er nok ikke megen tvivl om, at både de ansvarlige, underviserne og kursisterne intuitivt definerer brugerkurser som IT-kurser. Naturligvis skal et moderne brugerkursus også hjælpe brugerne i gang med at anvende det

## Brugeruddannelse i praksis

system, som skal implementeres, men det er mindst lige så vigtigt at fokusere på den *dagligdag*, som brugerne skal tilbage til. Dvs. at omdrejningspunktet i langt højere grad skal være de ændringer af roller og arbejdsgange, som er årsagen til, at man i det hele taget vil implementere et nyt system.

Hvis man f.eks. i en kommune vil indføre et elektronisk dokumenthåndteringssystem, så skal medarbejderne kunne håndtere systemet, men det er da nok så vigtigt, at de forstår baggrunden for, at ledelsen vil indføre det nye system og de nye muligheder for samarbejde og borgerbetjening, som systemet giver.

### **Modstandsproblemet**

Indførelsen af nye systemer og de ændringer i organisering og forretningsgange, som nye systemer ofte medfører, vil ofte udløse modstand hos medarbejderne. Dette tema behandles senere i bogen, pointen lige her er, at den undervisningsform, som her er beskrevet, i sig selv kan udløse den latente modstand. Dels på grund af den irritation og frustration, som kursisten føler, dels fordi det er meget lettere at brokke sig over en af de mange detaljer end over ændringerne.

### **Rammeproblemer**

Det er ikke kun det pædagogiske koncept, som kan medføre, at man får for lidt ud af brugeruddannelsen. De rammer, som de implementeringsansvarlige stiller op for brugeruddannelsen, har også stor betydning.

### **Timing af brugerkurserne**

Den enkeltfaktor, som nok har allerstørst betydning for effekten af brugeruddannelsen, er timingen af brugerkurset i forhold til ibrugtagningen af systemet. Alt for ofte vælger man af en række

praktiske og logistiske grunde at sende brugerne på kursus lang tid før, de skal i gang med at bruge systemet. Hvis brugerne er på kursus mange uger før ibrugtagningen, har det imidlertid flere negative konsekvenser.

For det første er det i sagens natur sværere at skabe en oplevelse af relevans, hvilket går ud over motivationen og dermed indlæringen. Hvis kurset på trods af dette skaber en lyst hos brugerne til at komme hjem og arbejde med systemet, så er der ret stor risiko for, at denne energi er fordampet inden ibrugtagningen. Men det største problem er, at det, som brugerne trods alt har fået ud af brugerkurset, med stor sandsynlighed går i glemmebogen, inden det når at blive brugt. Alt i alt er der således en risiko for, at selv et godt brugerkursus får en lav effekt, hvis timingen ikke er i orden.

### **Timing af udvikling**

Jeg bliver ofte kontaktet lige før ibrugtagningen, og spurgt om ikke vi kan lave et IT-pædagogisk kursus en uge eller to, før brugerkurserne skal afvikles. Intentionen er, at klæde underviserne på til at skabe god pædagogik i undervisningssituationen. Men forudsætningen for et godt brugerkursus er, at *kursusudviklingen* sørger for, at 90 % af kvaliteten er på plads, inden selve kurset skal afholdes. Man skal ikke forvente af underviserne, at de har overskud til at udtænke pædagogiske løsninger, når først de træder ind i undervisningslokalet. Løsninger kreeres i udviklingsfasen.

Ofte får underviserne alt for lidt tid til at udvikle kurserne, hvorfor den pædagogiske kvalitet alene af den grund bliver for dårlig. Dette er ærgerligt, for den tid, der investeres i udviklingen, sparer rigtig meget tid for de ofte hundredvis af brugere, som får accelereret deres læreproces.

Et andet problem er, at man ofte kommer for sent i gang med kursusudviklingen, således at denne skal finde sted, samtidig med at

## Brugeruddannelse i praksis

alle involverede har travlt med de andre dele af implementeringen. Dette gør, at udviklingen bliver præget af stress, hvilket ikke ligefrem er med til at forbedre situationen for undervisere og brugere.

### **Fysiske rammer**

Ofte bruger virksomhederne mange penge på at indrette lokaler til brugerkurserne, men desværre uden at der ligger en pædagogisk idé bag indretningen. I langt de fleste kursuslokaler sidder kursisterne ved deres skærme, også i den fase hvor underviseren skal formidle stoffet til dem. Det er naturligvis fristende for kursisterne at koncentrere sig om deres egen skærm frem for at fokusere på underviseren, og selv om man indfører et system, hvor man kan låse kursisters skærm eller tastatur, er en sådan opstilling ikke ligefrem optimal for kontakten og samspillet.

Samtidig er mulighederne for at bruge de klassiske medier såsom tavlen effektivt ofte begrænsede, fordi det lærred, som bruges til kanon-undervisningen spærrer for anvendelse, hvis der da overhovedet er andre medier i lokalet. Alt i alt betyder det, at der i det typiske kursuslokale er meget dårlige rammer for både underviser og kursister.

### **Uddannelsesmodel**

Med uddannelsesmodel menes en total plan for, hvordan brugerens kompetenceudvikling skal forløbe med de forskellige virkemidler, der nu engang er til rådighed. Hvis man skal designe en optimal uddannelsesmodel, kræver det, at man har fokus på læring. Naturligvis er det problematisk at afsætte for få ressourcer til uddannelse og support, men faktisk er der ofte afsat rigeligt med ressourcer; de skal blot anvendes, så de får en større effekt.

F.eks. er brugerkurser ofte for lange, i forhold til hvad brugerne kan kapere på en gang. De færreste brugere kan rumme de myriader af informationer, som et brugerkursus, som varer tre

dage eller flere, indeholder. Så hellere nøjes med to dage plus en opfølgning. På samme måde er brugerdokumentationen ofte så omfattende, at brugerne på forhånd opgiver at bruge den. Support er endnu et eksempel på, at der naturligvis skal være mulighed for at trække på superbrugere eller supportere, men hvis det ender med sidemandsoplæring, er det bestemt ikke hensigtsmæssigt (se kapitel 3).

Alt i alt er der således behov for fokus på læring og kvalitet, og dårlig kvalitet fjernes ikke ved flere ressourcer, tværtimod. Et af IT-pædagogikkens bærende læresætninger er da også: *Less is more*.

### **Opsamling på problemer**

Konklusionen er, at den klassiske form for brugeruddannelse ikke giver de resultater, som en moderne virksomhed har brug for, når den investerer i en kostbar implementering, som skal give afgørende forbedringer i produktion og service. Der er mange store problemer, de vigtigste er:

- Den grundlæggende pædagogiske idé er at fokusere på detaljer, som brugerne alligevel ikke kan huske
- Brugere får ikke skabt overblik og forståelse
- Selve konceptet for brugeruddannelsen er forældet, fordi omdrejningspunktet er IT og ikke den daglige praksis
- De logistiske rammer for udvikling og afvikling af kurserne gør det svært for underviserne at skabe optimale læreprocesser
- Ressourcerne anvendes ikke altid, hvor de giver størst effekt

Sagt på en anden måde benytter 80-90 % af brugerkurserne i danske virksomheder en pædagogisk model, som opstod i midt i 1980'erne, da systemerne var meget lidt brugervenlige, og brugerne var helt uden IT-kompetence. Modellen har hele tiden været problematisk, men står man foran at skulle rulle et stort

Brugeruddannelse i praksis

system ud, som mange brugere skal anvende i deres dagligdag, og som fordrer nye rollefordelinger mv., er den i hvert fald uaktuel og gør mere skade end gavn.

## Det IT-pædagogiske koncept

I det følgende gives en kort beskrivelse af det IT-pædagogiske koncept, som jeg har udviklet gennem de sidste tyve år, som er beskrevet i en række tidligere bøger og artikler, og som i dag benyttes af en række store offentlige og private virksomheder, når de skal uddanne deres brugere. I resten af bogen uddybes konceptet naturligvis, men her beskrives kerneprincipperne.

### Brugerkursets formål

Når brugerkurserne traditionelt lider af stoftrængsel, skyldes det bl.a. en antagelse om, at brugerne skal kunne systemet, når kurset er færdigt. Hvis dette reelt skulle kunne lade sig gøre, ville det kræve nogle særdeles lange kurser, for det at kunne et system på fingrene kræver træning og atter træning.

Udgangspunktet for det IT-pædagogiske koncept er derfor et andet: Et brugerkursus skal give brugerne *et fundament for en læreproces*, som begynder på kurset, men som skal fortsætte i lang tid efter kurset. Med dette udgangspunkt bliver formålet med kurset at understøtte brugerens egen læreproces efter kurset, dvs. at skabe selvhjulpne brugere. Det drejer sig derfor ikke om at nå så meget som muligt, men om at opnå nogle specifikke målsætninger, som understøtter den autonome bruger:

#### 1. Brugeren skal have en forståelse for formålet med implementeringen

Når formålet med implementeringen er omlægning af roller og forretningsgange, er det helt afgørende, at brugerne forstår og

køber den generelle strategiske baggrund for implementeringen og de specifikke ændringer, som de selv skal være en del af.

**2. Brugeren skal have en forståelse af principperne**

Jo bedre brugerne forstår de principper, som programmet er opbygget efter og den principielle baggrund for de nye roller og forretningsgange, jo bedre vil de være i stand til at fungere som selvstændige brugere og trække på samme hammel.

**3. Brugeren skal have et overblik over funktioner og muligheder**

Når en stor del af læreprocessen ligger efter brugerkurset, og brugerne skal være ansvarlige for egen læring, kræver det, at de har et godt overblik over, hvad systemet indeholder af funktioner og muligheder. Overblikket skal give brugerne inspiration i den fortsatte proces og må endelig ikke drukne i detaljer, så de ikke kan se skoven for bare træer.

**4. Brugeren skal være fortrolig med systemet**

For mange brugere er tryghed en forudsætning for, at de tør prøve noget nyt, når de kommer hjem. Derfor er det vigtigt, at de får så mange praktiske øvelser på kurset, at de føler sig rimeligt fortrolige med brugergrænseflade, manøvrering mv.

**5. Brugeren skal selv kunne lære nye funktioner**

Det sidste delmål er selve essensen af den selvhjulpne bruger: Brugeren skal selv være i stand til at sætte sig ind i de funktioner og dele af systemet, som *ikke* er gennemgået på kurset. Dette delmål er på den ene side konsekvensen af, at de andre delmål er nået, men det er også vigtigt, at kurset aktivt medvirker til at dette mål opnås.

### **Perspektivet**

Som påpeget tidligere bliver brugerundervisningen let systemorienteret. Det IT-pædagogiske koncept går ud på, at man konsekvent arbejder med et *situationelt perspektiv*. Dvs. at der hele tiden tages udgangspunkt i typiske situationer fra brugerens dagligdag, således at brugeren kan se, hvad systemet skal bruges til, og hvordan det hænger sammen med de overordnede mål med implementeringen.

Situationsorienteringen gælder på alle niveauer. På kursusniveau betyder det, at kurset ikke skal starte med, hvordan man f.eks. logger på systemet, men tværtimod med nogle af de centrale funktioner, som vil gøre en forskel i fremtiden. Den enkelte lektion tager ligeledes udgangspunkt i dagligdags situationer, og på samme måde tager den skriftlige dokumentation, som anvendes på og efter kurset, udgangspunkt i situationer og ikke i systemets logik.

### **Eliminering af hukommelsesproblemer**

Når deltagerne på traditionelle brugerkurser overrisles med huskestof, som de vel at mærke ikke kan huske, bidrager pædagogikken i hvert fald ikke til at udvikle autonome brugere, tværtimod bliver de dybt afhængige af hjælp. Det IT-pædagogiske koncepts radikale løsning på denne problemstilling er ganske enkelt at *fjerne alt huskestof* fra underviserens gennemgang. Dette betyder, at også al skærbilledgennemgang forsvinder. Hvis man er vant til traditionel brugerundervisning, er det måske svært at forstå, at det kan lade sig gøre. Men konceptet er afprøvet på titusindvis af brugere, og alle erfaringer viser, at brugerne bliver begejstrede, samtidig med at effekten stiger kraftigt.

Fordelen er bl.a., at underviseren og deltagerne kan fokusere på principper og overblik. Men spørgsmålet er naturligvis, hvordan kursisterne så finder ud af at betjene systemet.

Betjeningen af et system består i virkeligheden primært af to ting:

- **Korrekte valg**

Trykke på de rigtige knapper, vælge den relevante funktion, udfylde felter korrekt etc. Det svære i moderne systemer er ikke at trykke på de rigtige knapper. Brugergrænsefladen er blevet forbedret markant i løbet af de sidste 10 år, og i de fleste tilfælde kan man overføre sine erfaringer fra system til system. I forretningssystemer vil brugerne endda kende (det meste af) terminologien i forvejen.

- **Navigation**

Bevæge sig rundt i systemet, komme det rigtige sted hen, have en fornemmelse af, hvor man har været, og hvor man er på vej hen. Da moderne systemer er blevet større og mere komplekse, og da det er afgørende, at man forstår sammenhænge mellem de forskellige dele af systemet, er navigationen blevet noget af det vigtigste og samtidig det sværeste ved at lære et nyt system.

Det IT-pædagogiske koncept eliminerer hukommelsesproblemet og understøtter brugerens valg og manøvrering ved hjælp af det, vi igennem de sidste 15 år har kaldt Huskesedler, men som i dag rettelig bør hedde *Navigationssedler*, fordi den primære funktion er at hjælpe brugeren med at manøvrere. Det er på dette punkt, at brugerne har det største hjælpebehov, og hvor systemerne svigter brugerne.

En grundigere gennemgang af Navigationssedlerne gives i kapitel 7, men princippet er altså, at alt huskestof fjernes fra underviserens gennemgang. En vigtig pointe er, at alle de informationer, som brugeren selv kan se og forstå på skærmen, *ikke* placeres på Navigationssedlen. Dvs. at den primært fokuserer på at få brugerne det rigtige sted hen; kun de informationer, som er svære at forstå på skærmen, skal med på Navigationssedlen.

## Brugeruddannelse i praksis

Dette betyder, at den enkelte Navigationsseddel selv ved meget vanskelige systemer i 99 % af tilfældene kan holdes på en A4 side. Dette er afgørende, for derved opnås, at brugerne faktisk anvender materialet, hvorved målsætningen om at gøre brugerne autonome understøttes.

### **Forståelsesstrategien**

Når oplæggene tømmes for detaljer og huskestof, bliver der mulighed for at fokusere på at skabe forståelse og overblik. Det IT-pædagogiske koncept indeholder en vifte af metoder og værktøjer, som hjælper underviserne med at producere forståelse. F.eks. den såkaldte pointeanalyse, som er en metodisk tilgang til, hvordan man finder ud af, hvad det konkret er, brugerne skal sige *aba* til.

Resten af bogen går i dybden med forståelsespædagogikken, her skal blot nævnes, at omdrejningspunktet for forståelsen på et brugerkursus er de såkaldte ”3 P’er”:

- **Problemet** (også kaldet ”Pehovet”)
- **Principperne**
- **Procedureoverblikket**

For at brugeren i det hele taget kan se relevansen og behovet for IT-løsningen, indledes altid med at belyse **Problemet**. Underviseren giver et eksempel på en situation i dagligdagen, hvor systemet kan anvendes til at skabe en (bedre) løsning. Hvis kurset f.eks. handler om elektronisk dokumenthåndtering (EDH/ESDH) i en kommune, kunne et situationseksempel være, at en borger ringer ind for at følge op på en sag, men borgeren kan ikke få et svar, fordi den pågældende sagsbehandler er syg. Pointen er, at det nye system fremover giver en anden medarbejder mulighed for at give borgeren en status, selv i denne situation.

I næste omgang skal brugerne forstå **Principperne**: Hvis det stadig drejer sig om EDH, skal de f.eks. forstå, hvordan man definerer en sag, og hvordan systemet kategoriserer sager.

Sidst men ikke mindst skal **Procedureoverblikket** sørge for, at brugerne har et overblik over de logiske trin, som brugeren skal igennem for at følge forretningsgangen og bruge systemet. Hvilke faser skal man f.eks. igennem for at oprette en ny sag.

### **Udviklingen**

Hvis en enkelt underviser en sjælden gang imellem skal holde et kursus, er det begrænset, hvor meget planlægning og udvikling, der kan ligge forud for selve undervisningen, hvilket betyder, at den pædagogiske kvalitet i høj grad skabes i nuet.

Situationen er en helt anden, når en gruppe af undervisere skal rulle et system ud til mange medarbejdere. Typisk skal man inden for nogle få uger afholde det samme kursus rigtigt mange gange, og det er vigtigt, at kurserne har en høj kvalitet fra dag et. I denne situation er den fælles kursusudvikling helt afgørende, og grundidéen i det IT-pædagogiske koncept er, at op imod 90 % af den pædagogiske kvalitet kan og skal skabes i udviklingsfasen.

Det IT-pædagogiske koncept indeholder et helt arsenal af metoder og principper, som kan sikre, at der udvikles et kursus, som har en høj pædagogisk kvalitet, og som er så veldokumenteret, at alle underviserne kan lave et godt kursus.

### **Opsamling**

Gennem 20 år er der oparbejdet en fasttømret tradition for, hvordan man afvikler brugerkurser. Kapitlet har påpeget de mange problemer på kort og lang sigt, som denne traditionelle kursusmodel giver anledning til. Det IT-pædagogiske koncept beskriver en radikal anden måde, som bygger på indlæringspsykologisk

Brugeruddannelse i praksis

teori, og som har vist sig særdeles effektiv i praksis. Kapitlet har introduceret konceptets bærende principper, resten af bogen går i dybden med såvel principper som metoder.