



GÖTEBORGS UNIVERSITET

Användarvänlig systemdesign

En studie om användarcentrerade principer

User-friendly system design

A study of user centered principles

**CAMILLA JOHNSON
MATILDA WALLIN**

Kandidatuppsats i informatik

**Rapport nr. 2013:017
ISSN: 1651-4769**

Göteborgs universitet
Institutionen för tillämpad informationsteknologi
Göteborg, Sverige, Maj 2013

Abstrakt

Rapporten är skriven på svenska. För att utveckla användarvänliga system finns ett flertal riktlinjer angående hur man ska designa systemet. Rapporten tar huvudsakligen upp de designprinciper som Professor Donald Norman utvecklade, och som behandlar varje område i ett informationssystem. Det finns även andra riktlinjer som beskriver användarprinciper och universal design, och som med fördel kan följas vid design av ett informationssystem. En grupp på North Carolina State University i USA är en av många som har utvecklat sju principer för att behandla användarvänlighet[1].

Syftet i denna uppsats är att granska ett existerande system och att applicera dessa ovanstående nämnda principer på fallstudieobjektet för en utvärdering. En analys av fallstudieobjektet har skett genom ett utbildningstillfälle, observationer och intervjuer. I slutet av denna uppsats ges rekommendationer och riktlinjer på hur man kan skapa eller göra om ett informationssystem till att bli mer användarvänligt. Det ges även riktlinjer för det aktuella fallstudieobjektet.

Nyckelord: Designprinciper, Användarcentrerad systemdesign, Användbarhetsprinciper, Universal Design.

TACK

Vi vill tacka vår handledare och alla som hjälpt oss att genomföra vår fallstudie. Tack till informanterna som deltagit och tagit sig tid att svara på våra frågor. Vi vill också tacka vår handledare Urban Nuldén för all konstruktiv feedback.

Innehållsförteckning

1. Inledning	1
1.1 Syfte och frågeställning	2
1.2 Process och avgränsning	2
2. Relaterat arbete och teori	3
2.1 Användbarhet	3
2.2 Användbarhetsprinciper och regler	4
2.3 Universal Design	6
2.4 Action cycle - cykliskt handlande	8
2.5 Användarcentrerad systemdesign	9
2.6 Designprinciper	10
3. Fallstudieobjekt; ”Admin”	11
3.1 Vad är ”Admin”?	11
3.2 Funktionaliteten i ”Admin”	12
4. Metod	14
4.1 Utbildning	15
4.2 Observation	15
4.3 Intervjuer	15
5. Resultatredovisning/analys	16
5.1 Synlighet (Visibility)	16
5.2 Återkoppling (Feedback)	18
5.3 Navigering (Mapping)	20
6. Diskussion	21
6.1 Tidig fokus på användarna	21
6.2 Uppbyggnad för snabb inläring	22
6.3 Bekräftelse	22
6.4 Utbildning i mindre omfattning	23
7. Slutsats	23
8. Referenser	25
Bilaga 1-Intervjufrågor	26

1. Inledning

I systemutvecklingsprojekt strävar man efter att designa användarvänliga system. Det finns ett antal metoder och riktlinjer som vägleder en sådan design. Ett system som är användarvänligt bör också vara lätt för användaren att lära sig. Tiden för utbildning i ett användarvänligt system kan förhoppningsvis därmed hållas relativt kort. Användandet av informationssystem är idag utbrett. Ett företag förväntas använda ett effektivt informationssystem och medarbetare förväntas ha kunskap om hur detta system fungerar [3].

Det idag breda användandet av informationssystem borde vara en förutsättning för att utveckla användarvänliga system [13]. Tyvärr motsvarar detta oftast inte verkligheten. En stor problematik som varit betingad sedan länge, är att många användare idag har svårt för att förstå och lära sig olika informationssystem [3].

En av huvudorsakerna till detta problem beror på informationsteknologins ständiga utveckling. Nya sätt att interagera med datorer utvecklas hela tiden, och designen av dessa bör alltid vara anpassande för produktens ändamål. Därför kommer utformningen av design för informationssystem alltid vara både intressant att utveckla och även forska kring.

En annan orsak är att användaren kan tröttna på alldeles för svårbegripliga system och därmed överväger att hellre lämna systemet än att försöka få mer kunskap om det. Detta kan i förlängningen bidra till höga kostnader och därmed minskad vinst för företagen, då användarna av ett invecklat system kan vara ineffektiva och systemet kan dessutom kräva kostsam omdesign för att nå bruklig standard [8].

Idag använder anställda i företagen sig av aktiva arbetsätt genom exempelvis ökat projektsamarbete och detta bidrar oftast till en ökad kompetens generellt sett i arbetsgruppen. En bra och framåtskridande utveckling i gruppen hjälper användarna att lättare ta till sig och lära sig ny teknik. Fenomenet bör tas hänsyn till och utnyttjas då man vid implementeringen bör uppmärksamma den period som användaren har som lättast att lära sig ny teknik, d.v.s. när gruppmodnaden är redo [13].

När man skapar ett system så bör användarvänligheten stå i fokus med samma prioritet som funktionalitet och teknik. Genom att designa systemet användarvänligt finns mycket att vinna för både användaren och utvecklaren. Genom att granska ett existerande system där användarna har en varierande IT-kunskap kan generella riktlinjer tas fram utifrån flera användarperspektiv [3].

Denna uppsats innefattar en genomgång av litteratur inom användarvänlig design, dess krav och principer. Genomgången av litteraturen kommer att resultera i riktlinjer som verifieras genom en fallstudie där ett existerande system granskas. I denna fallstudie kommer observationer, ett utbildningstillfälle och intervjuer att genomföras. Detta för att dels få en förståelse för systemet och för att ta del av användarens synpunkter.

Det studerade fallstudieobjektet som vi kallar "Admin", är ett så kallat "Software as a Service" (SaaS) system som används inom bevakningsbranschen.

1.1 Syfte och frågeställning

Syftet med denna studie är att ta fram generella riktlinjer som gör informationssystem mer användarvänliga.

Med ovanstående syfte har följande frågeställning utformats:

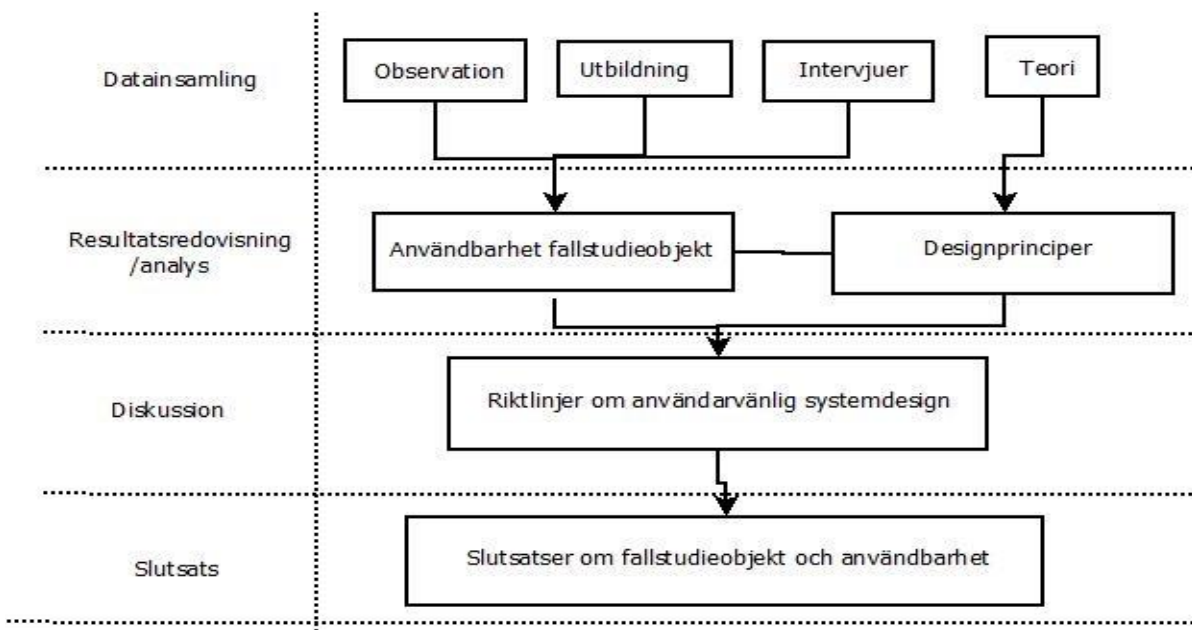
- *Vilka huvudsakliga principer och krav bör följas vid utformandet av användarvänliga informationssystem?*

1.2 Process och avgränsning

Det valda fallstudieobjektet är ett stort system. Därför ligger fokus på en avgränsad del av systemet. Då uppsatsen är en offentlig handling har företaget valt att vara anonyma. Därför kallas fallstudieobjektet för "Admin" och företaget som utvecklat detta system för "Bevakningsföretaget" i uppsatsen.

Teoriavsnittet beskriver olika möjliga designprinciper som sedan applicerats på fallstudieobjektet i resultatredovisningen. Denna studie är en empirisk undersökning där resultatet baserats på de erfarenheter som informanter presenterat och som uppmärksammats under studien. Vid studien av detta fallstudieobjekt tillämpades intervjuer, deltagande i ett utbildningstillfälle och observation som metod.

Normans designprinciper i studien presenteras även på engelska då de är svåra att översätta på ett korrekt sätt. En resultatredovisning med analys av materialet vävs ihop i slutet av uppsatsen. När begrepp som "Användarvänlighet" och "Användbarhet" behandlas i texten, refererar dessa begrepp endast till IT-området.



Figur 1. Bild över studiens upplägg.

2. Relaterat arbete och teori

Det finns en rad forskare som undersöker design för att få en klarare bild över vad som är viktigast i ämnet. Fokus i detta avsnitt kommer att ligga på användbarhet, vilka användbarhetskrav som finns, användarcentrerad systemdesign[3] och designprinciper[9], eftersom dessa punkter syftar till att underlätta för designen av användarvänliga system. Litteraturen är vald utifrån sökningar efter kända forskare och deras studier i ämnet. Fokus i resultat och analys kommer att ligga på Donald Normans utvecklade designprinciper [9].

De riktlinjer som presenteras här är valda utifrån de mest förekommande och lättförståeliga principer i dagens design av informationssystem. Alla nämnda principer ska kunna gå att applicera på design av olika informationssystem. Denna studie presenterar teorin i upplevd sekventiell ordning. Detta för att det krävs förkunskaper för att kunna förstå de återkommande designprinciperna i uppsatsen [3].

2.1 Användbarhet

Användbarhet kan uppfattas som en självklarhet att ta hänsyn till för en systemutvecklare, vid designen och utvecklandet och av produkter och tjänster. Undersökningar har visat att så inte är fallet, då det idag finns många informationssystem som inte är användarvänliga. Detta beror främst på att systemutvecklarna saknar kompetens, intresse eller förmåga att utveckla en väl anpassad gränssnittsdesign. Eftersom systemutvecklarna huvudsakligen fokuserar på att utveckla system, så finns det sällan tid för att dessutom göra en användarvänlig design. Den dokumentation som skapas vid utvecklingsarbetet är oftast tekniskt svårt att förstå för andra som inte har deltagit i arbetet. Detta bidrar också till att det blir svårt för utomstående att komma in och endast hjälpa till med gränssnittsdesignen [3].

Begreppet användbarhet är ett kvalitetsmått. Kvalitetsmättet mäter hur effektiv en produkt är för användaren, hur väl användaren minns hur man arbetar i systemet, samt hur många fel som är möjliga att göra. Designprinciper har utvecklats med olika utgångspunkter för att underlätta för systemutvecklarna [4].

En internationell satsning har gjorts för att identifiera exakt vad användbarhet innebär. Därför har en ISO-standard utvecklats. ISO har försökt skapa ett internationellt samarbete för att stärka användbarheten inom produktutveckling. Standard för användbarhet är specificerat som ett krav enligt ISO 9241:11 [4].

ISO 9241:11 tar fokus på tre huvudområden. Den första är ”kraftfullhet” som förklarar i vilken utsträckning en uppgift blir avklarad. Den andra är ”effektivitet” och innebär, till skillnad från kraftfullhet, den grad av ansträngning användaren behöver lägga ner i informationssystemet. Den tredje är ”tillfredsställelse” och kopplas till graden av positiva känslor som produkten förmedlar till användarna [4].

Begreppen ”användarvänlighet” och ”användbarhet” är återkommande begrepp i informatiken, då man designar system. Båda orden används ofta tanklöst översatt av engelska ordet ”usability”, trots att dessa har olika betydelser. Användarvänlighet innebär

att ett system är enkelt och lätt att använda sig utav. Ett användarvänligt system ger inte användaren några onödiga problem. Ett sådant system är huvudsakligen logiskt, konsekvent, effektivt och intuitivt. Användaren ska inte uppleva systemet som jobbigt eller obehagligt att interagera med [2].

En hög grad av användarvänlighet innebär dock inte att ett system är användbart. En av de viktigaste delarna i användbarhet är ett systems nytta. Med detta menas att systemet skall leda användaren till tillfredställelse samt lösningar av problem och misstag. Ett system som inte ger eller erbjuder en möjlighet för användaren att nå upp till sitt mål ger inte heller någon nytta. Sammanfattningsvis kan man säga att för att kunna uppnå den nivå där man kallar sitt system *användbart*, behövs att både *nytta och användarvänlighet* finns i systemet. [2].

För att försöka specificera mer vad användarvänlighet innebär så har flera forskare försökt konkretisera begreppet användarvänlighet. De vill lägga in ett krav som ska uppnås för att produkten ska få kallas användarvänlig. Detta gäller både produkten och utvecklingsprocessen[4].

Användbarhetskrav uppfattas av många som en självklarhet i ett system, men tyvärr finns många exempel på att dessa inte uppfylls. Användbarhetskraven behandlar ändamålsenlighet, effektivitet och subjektiv uppfattning/tillfredsställelse för de specifika användarna, dess mål och även dess sammanhang [3].

Nielsen har tagits fram fem olika punkter som beskriver vad användbarhet innebär. Dessa är:

1. Lätt att lära sig

Riktlinjer: När användaren ska lära sig ett nytt system, ska det vara lätt att komma igång och lära sig.

2. Effektivt

Riktlinjer: När användaren lärt sig det nya systemet ska det vara enkelt och effektivt att arbeta med.

3. Lätt att komma ihåg

Riktlinjer: När användaren inte arbetat med systemet under en längre period ska det vara lätt att komma ihåg hur systemet fungerar.

4. Fåtal misstag

Riktlinjer: Användaren ska bara kunna göra ett fåtal misstag. Då användaren gjort något fel ska man kunna gå tillbaka i systemet tills innan felet uppstod.

5. Tilltalande

Riktlinjer: Användaren ska tycka om att arbeta i systemet [8].

2.2 Användbarhetsprinciper och regler

Ett system bör vara utformat med god användbarhet för att kunna hjälpa användaren att utföra sina uppgifter på allra bästa sätt. Niensens fem beskrivningar av användbarhet ovan kan liknas med nedanstående användbarhetsprinciper av Gulliksen & Göransson.

Skillnaden är att dessa principer ger lite mer detaljerad information till utvecklaren av systemet, i syfte att vidhålla ett användarfokus genom hela utvecklingsprocessen. Nielsens beskrivningar ovan ger dock en god översikt hur användbarhet i stort bör definieras i ett system när det väl tas i bruk.

Nedanstående användbarhetsprinciper kan ses som vägledning genom utvecklingsprojektet, och bör följas av de som ska designa och utveckla ett system: [3]

1. Användarfokus

Riktlinjer: Man bör anpassa utvecklingsprocessen till att ha ett användarfokus redan från början.

2. Aktiv användarmedverkan i utvecklingen

Riktlinjer: Systemutvecklaren bör låta användaren aktivt medverka under hela utvecklingsprocessen.

3. Utveckling

Riktlinjer: För att kunna uppnå bästa möjliga informationssystem, så är en utveckling baserad på iterativa möten med användaren mest effektiva och ger bäst slutprodukt.

4. En gemensam förståelse

Riktlinjer: Alla medlemmar som är involverade i utvecklingsprocessen ska förstå och ha samma vision med systemet, vilket kräver en omfattande dokumentation.

5. Prototyper

Riktlinjer: I utvecklingsprocessen bör fokus vara på att skapa prototyper redan i ett tidigt skede. Detta för att skapa förståelse för att systemet verkligen utvecklas till vad användaren förväntar sig.

6. Utvärdering

Riktlinjer: Utvärderingen genomförs av användare och är en viktig del, då det här verifieras att prototyper motsvarar förväntningarna på systemet.

7. Tydliga designuppgifter

Riktlinjer: Processen ska vara väl definierad från start. Designern ska inte vara tveksam till vilka uppgifter som är valda att genomföras. Alla uppgifter ska vara väl utvalda och ge resultat.

8. Tvärdisciplinära team

Riktlinjer: Ett bästa resultat ges när tvärdisciplinära team arbetar tillsammans, så att kompetenser blandas och bredden på kunskapen blir optimal.

9. Användbarhetsförespråkare

Riktlinjer: I dessa team bör förespråkare för användbarhet involveras redan i startskedet av processen. Användbarhetsförespråkare ska vara involverade på detaljnivå under hela processen för att få bästa möjliga resultat. Användbarhet i systemet ska genomsyra hela processen från början till slut.

10. Integrerad systemdesign

Riktlinjer: Systemet ska utvecklas parallellt under hela processen och hela processen ska ske kontinuerligt. Man måste se till att användningssammanhanget för den fysiska miljön beaktas.

11. Lokalanpassade processer

Riktlinjer: Varje process ska anpassas efter den aktuella organisationen. Organisationen måste själv ta ansvar för att de kompletterande aktiviteterna beaktas.

12. Användarcentrerad attityd

Riktlinjer: Det bör vara en hög kunskapsnivå hos projektets medlemmar för att få fram bästa möjliga resultat. Samtliga medlemmar i utvecklingsteamet behöver träffa de verkliga användarna för att kunna få en reell bild av användarna [3].

Ytterligare en beskrivning av användbarhet i syfte att utforma en design som stödjer användaren, ges nedan av Ben Schneiderman som har utvecklat åtta gyllene regler. Ett gränssnitt som enligt Schneiderman utformas efter endast ett par av dessa regler bör generera en mer användarvänlig design, med syfte att uppfylla både kvalitetsmått och effektivitetsmål [4].

Dessa gyllene regler fungerar på liknande vägledande sätt för designern, som de beskrivningar och principer ovan av Nielsen och Gulliksen & Göransson. En del av dessa regler tar även upp användbarhet ur samma vinkel som beskrivningarna och principerna ovan.

Schneidermans gyllene regler, som behandlar människans förmåga att ta till sig information och kognitivt beteende, är: [12]

- 1. Sträva efter kontinuitet**
- 2. Möjliggöra för frekventa användare att använda genvägar**
- 3. Erbjud informativ återkoppling**
- 4. Designa dialoger som bidrar till närhet**
- 5. Erbjud avvärjande av fel samt enkel felhantering**
- 6. Tillåt enkla sätt att gå bakåt i handlingar**
- 7. Stöd den inre känslan av kontroll**
- 8. Reducera belastningen på korttidsminnet**

2.3 Universal Design

Att använda sig av *universal design* innebär att skapa system som ska vara användbara för användare med olika förmågor och erfarenheter. På North Carolina State University i USA ligger centrum för Universal Design [1]. Här har författare, arkitekter, designers, ingenjörer och miljöforskare inom design samarbetat för att utveckla sju olika principer som beskrivs nedan.

Alla de ovan beskrivna principerna och reglerna behandlar design ur ett generellt perspektiv med användbarhet i fokus. Riktlinjer inom universal design går ett steg längre och specificerar innebörden i att man därutöver även ska kunna anpassa systemet efter användare med olika behov. Inte bara utifrån användarens teknikvana utan även utifrån

olika funktionshinder. Universal design hjälper till att se lösningar där användarcentrerad systemutveckling inte räcker till.

Dessa sju principer skapades för att vägleda design av system samt produkter för alla människor, oavsett behov: [1]

1. Likvärdig användning

Designen är användarcentrerad och anpassad till människor med olika förmågor.

Riktlinjer: Ge samma mening för alla användare vid användandet: identiska när det är möjligt, motsvarande när inte. Integritet och säkerhet ska vara lika tillgängliga för alla användare. Gör designen tilltalande för alla användare.

2. Flexibilitet i användning

Designen har ett brett utbud av individuella preferenser och förmågor.

Riktlinjer: Ge val i användningssättet. Ge höger- och vänsterhänta samma möjlighet för åtkomst och användning. Underlätta för användarens noggrannhet samt användarens individuella arbetstempo.

3. Enkel och intuitiv användning

Designens användning ska vara enkel att förstå, oberoende av användarens tidigare erfarenhet, kunskaper, språkkunskaper eller koncentrationsförmåga.

Riktlinjer: Var konsekvent med användarnas förväntningar och intuition. Ordna information i enlighet med dess betydelse. Ge effektiva uppmaningar samt feedback under och efter slutförda uppgifter.

4. Gör informationen tillgänglig

Designen gör behövlig information effektivt för användaren, oavsett miljöförhållanden eller användarens förmågor.

Riktlinjer: Använd olika tekniker (exempelvis bild, verbalt och taktil) för tydlig presentation av väsentlig information. Maximera "läsbarhet" av nödvändig information. Skilja funktionerna åt på ett sätt som kan göra det enklare att ge instruktioner eller andra direktiv. Skapa kompatibilitet med en mängd olika tekniker och hjälpmedel som används av personer med begränsningar.

5. Tolerans för fel

Uppbyggnaden ska minimera riskerna och de negativa konsekvenserna, av oavsiktliga handlingar.

Riktlinjer: Ordna funktionerna för att minimera risker och fel. Ge varningar för faror och fel och använd felsäkra funktioner.

6. Låg fysisk ansträngning

Designen ska kunna användas bekvämt och med ett minimum av trötthetskänsla.

Riktlinjer: Tillåt användaren att ha en neutral kroppsställning. Man bör minimera den fysiska ansträngningen genom att exempelvis undvika upprepade rörelsemönster.

7. Tillåtet utrymme för användandet

Lämplig storlek och utrymme ska erbjudas för olika tillvägagångssätt. Detta möjliggör för ett användande som är oberoende av användarens kroppsstorlek, kroppshållning eller

mobilitet.

Riktlinjer: Ge en klar siktlinje samt möjlighet att nå alla funktionerna bekvämt för sittande eller stående användare. Tillräckligt utrymme för användning av hjälpmedel[1].

2.4 Action cycle - cykliskt handlande

Cykeln för mänskligt handlande är en modell som beskriver de olika steg en människa tar när de interagerar med ett datorsystem. Modellen används huvudsakligen för att utvärdera effektiviteten av ett användargränssnitt utifrån designprinciperna och innefattar kognitiva samt fysiska aktiviteter. Cykeln beskriver hur en användare kan bilda mål och sedan skapa en serie av olika steg för att uppnå detta mål [9].

En användare ska inte behöva tänka på hur uppgifter i systemet ska utföras, utan detta ska vara uppenbart. En enkel ruta centrerad på sidan, som exempelvis Googles sökfält, visar på ett enkelt sätt hur detta gränssnitt förväntas användas. Målet med ett användarvänligt system är att det ska vara självförklarande för användaren. Ett exempel är hur en användare oftast inte läser en hel informationstext utan endast ögnar igenom denna text med förväntningen om att inte behöva läsa allt för att förstå innebörden. Därför måste också varje ord i ett användarvänligt system vara tillräckligt beskrivande för användaren [7]. Detta är ett starkt argument till varför man kontinuerligt bör utvärdera sitt användargränssnitt för att försäkra sig om att de ord och begrepp som används i systemet ger tillräcklig, adekvat och aktuell information.

Följande steg bör gås igenom vid utvärderingen av ett användargränssnitt:

1. Bilda ett mål.
2. Omsätta målet i en aktivitet eller en uppsättning aktiviteter.
3. Planera en åtgärdssekvens.
4. Utföra en åtgärdssekvens.
5. Perceptionera vad som händer.
6. Tolka resultatet enligt användarnas förväntningar.
7. Utvärdera vad som händer mot vad som var tänkt [Norman].



Figur 2. Sammanfattande bild av vad "Action Cycle" innebär[15].

Målet med universal Design är att ett system ska vara anpassat för användare med olika behov, men det behövs ordentligt med kunskap inom "action cycle" för att kunna avgöra hur användarna interagerar med systemet. Målet med action cycle är att påvisa vilken handlingscykel vi människor intuitivt använder oss av.

2.5 Användarcentrerad systemdesign

En användarcentrerad systemdesign är inte en vedertagen metod utan snarare ett svårdefinierat begrepp. Syftet med designen av system är att stödja användaren i sina arbetsuppgifter. I forskningen har man kunnat konstatera att användarfokus, mätbar användbarhet och iterationer är grunderna i användarcentrerad systemdesign[9].

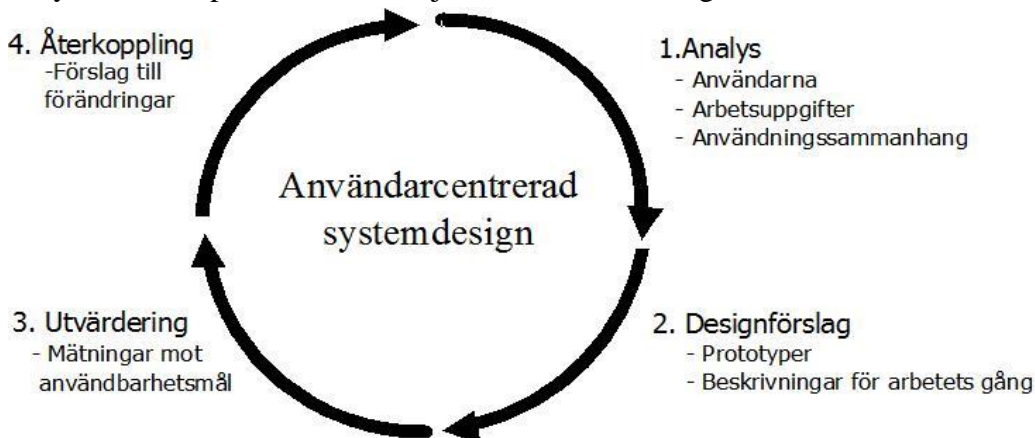
Att utveckla system så att de passar användaren är egentligen inte någon nyhet, men detta kriterium har däremot ofta glömts bort av många utvecklare. Att inte ta hänsyn till detta kan bli kostsamt för utvecklingsföretaget, både i utbildningskostnader och förlorade kunder.

Användarcentrerad systemdesign handlar om att arbeta i iterationer för att testa och utvärdera systemet kontinuerligt. Som designer och utvecklare ska man ta reda på vem användaren är och anpassa produkten utefter det. Att involvera användaren i tester av tidiga prototyper som genomförs är viktigt. Anledningen till detta är att användarna ska förstå systemet och känna att systemet blir som de har förväntat sig från början. Mycket av upplevelsen har med tidigare förväntningar att göra och bestämmer vilken inställning användaren har till systemet. I vissa fall kan det bli så att utvecklaren inte behöver lägga till extra funktioner, eftersom användaren känner att systemet redan är fullt effektivt[3].

Inom användarcentrerad systemdesign har flera forskare uttryckt åsikter om hur det egentligen bör genomföras. Vissa forskare antyder att användaren ska vara delaktig genom hela utvecklingsprocessen[11].

Gränssnittet ska styra hur användaren interagerar med systemet och sedan hur systemet praktiskt fungerar[8]. Man ska ha som mål att göra systemet användarvänligt, det är en iterativ och öppen process där potentiella användare är involverade. Detta på grund av att det kan vara olika förutsättningar, beroende på vilken typ av system som ska utvecklas[5].

För att slutresultatet av en utvecklingsprocess ska bli tillfredställande för användaren, bör användarens och designerns bild av det kommande systemet vara likartade. För att detta ska kunna ske krävs kommunikation genom hela processen. Användaren kan ibland uttrycka en sak på ett sätt men i själva verket mena något helt annat[3].



Figur 3. Visar processen hur man kan skapa användarcentrerad systemdesign.

2.6 Designprinciper

Flera forskare har tagit fram liknande typer av designprinciper men i denna studie läggs fokus på Normans designprinciper. Detta för att dessa principer är de mest kända och ofta används inom design. Dessa principer är aktuella och populära eftersom de går att applicera på nästan alla produkter och verktyg och inte bara är relaterade till IT. Dessa sex principer vill förmedla riktlinjer för ett produktskapande [9].

Designprinciperna kan användas som ett hjälpmedel då man ska ta fram en ny användarvänlig produkt. Principerna innebär inga bestämda regler om hur man kan skapa användarvänliga produkter. De innebär snarare riktlinjer och tankesätt för att kunna designa, eller designa om, en produkt så att den blir användarvänlig [9]. Man bör anpassa sin design utefter någon/några utvalda designprinciper. Att använda sig av alla designprinciper kan bli problematiskt eftersom de i praktiskt användande kan bli motsägelsefulla [11].

1. Affordance (Erbjuder)- *How do I use it? (Hur använder jag det?)*

Affordance innebär att ett objekt antyder med hjälp utav dess design, hur man ska använda den. Det finns två olika typer av begreppet affordance: verklig och uppfattad.

Den verkliga typen innebär fysiska objekt, exempelvis hur man håller i den fysiska produkten. Den uppfattade typen av affordance är kopplad till användargränssnitt och system. Denna typ innebär kunskapen om rutiner genom inlärd processer sedan tidigare. Detta kan exempelvis vara genom hjälplänkar som beskriver funktionaliteten i system[9].

2. Feedback (Återkoppling)- *What is it doing now? (Vad händer nu?)*

När man som användare navigerar runt i ett system och exempelvis ändrar några uppgifter så vill man ofta få feedback. Detta innebär att användaren får en bekräftelse på att ändringar genomförts eller inte genomförts. Man vill ofta få direkt respons och inte vänta eller tro att systemet hängt sig, eller att ändringarna inte utförts.

Det finns tre sorters feedback: audiolog, taktil och visuell. Dessa tre nämnda typer av feedback kan exemplifieras genom att en användare skickar ett mail via en iPhone. Då uppstår taktil feedback via knappen "skicka", som trycker på för att mailet ska skickas. När man trycker på "skicka"-knappen kommer en audiolog feedback fram, som presenteras via ett ljud. Samtidigt kommer en laddnings-ruta fram längst nere på skärmen så användaren kan förstå att mailet håller på att skickas, vilket kallas visuell feedback.

Feedback behöver inte utgöras av alla typerna av feedback samtidigt, dock uppfyller man en högre grad av användbarhet om man har flera olika typer av feedbacks samtidigt. Det ska även finnas gränser för vilken feedback som ska användas och var. För mycket feedback skapar irritation hos användaren och det är därför ofta viktigt att begränsa feedback-systemet[9].

3. Visibility (Synlighet) - *Can I see it? (Är det tydligt?)*

Visibility handlar om att alla funktioner ska vara synliga för användaren. En funktion ska vara lättillgänglig för att den också ska vara lätt att använda. Det ska dessutom tydligt framgå vad som händer om användaren väljer denna funktion. Som exempel kan man ta ett formulär där användaren fyllt i uppgifter. Ett formulär avslutas med en "spara"-knapp

där användaren får bekräftat att uppgifterna sparats när man klickat på den [9]. Visibility handlar även om att användaren ska känna sig bekväm med användandet av produkten. Att känna sig bekväm med produkten gör att användaren upplever den som intuitiv och lättanvänd [11].

4. Mapping (Navigering)- *Where am I and where can I go? (Vart finns funktionerna?)*

Mapping i ett system handlar om att användaren ska veta vart den befinner sig och vart den är på väg [9]. Människan har ett naturligt tillvägagångssätt att navigera sig fram på. Det finns många metoder som är knutna till den kultur vi lever i och dessa känns självklara för användaren. Ett exempel är att i västvärlden börjar ett flöde från vänster och fortsätter åt höger. En produkt blir lätt att använda när det finns tydliga och synliga vägledningar för att hur man ska gå vidare för att slutföra produktens uppgift. Med mapping i ett system kan användaren strukturera innehållet och anpassa det till speciella syften. Mapping hjälper även till att presentera informationen så att den blir lätt att förstå och komma ihåg [9]

7. Constraints (Begränsningar) - *Why can't I do that? (Varför kan jag inte göra det?)*

När man skapar begränsningar kan användaren lätt uppfatta det som att denna begränsas i genomförandet av sina planerade uppgifter. Detta är i sådana fall ett missförstånd som handlar om att designern vill att användaren ska utföra sina uppgifter på ett speciellt sätt. Anledningen till denna styrning är att minimera risken för att användaren ska kunna använda systemet på ett felaktigt sätt. Anledningen till att användaren kan känna sig begränsad handlar om invanda tidigare beteenden, som från exempelvis vanor från ett annat system[9].

6. Consistency (Igenkänningsbarhet) - *I think I have seen this before? (Känner jag igen detta sedan tidigare?)*

Att designa ett system som följer allmänna konventioner för design, gör att användaren känner igen sig och lättare kan ta till sig systemet. En återkommande funktion i ett informationssystem bör se likadan ut för användaren varje gång. Där igenom kan man minimera risken för misstag genom att tillåta användaren att komma ihåg åtgärder som utförts tidigare [9]. En funktion i ett system bör vara markerad på så sätt att användaren vet intuitivt vad som händer exempelvis när en knapp trycks på. Men att det är allmänt vedertaget behöver inte betyda att det är korrekt [11].

3. Fallstudieobjekt;

För att kunna applicera dessa ovanstående designprinciper och kunna koppla teori till praktik har det studerats ett verkligt fall av ett informationssystem vid namn ”Admin”. I detta avsnitt kommer vi beskriva vad ”Admin” är och redovisa för dess funktionalitet.

3.1 Vad är ”Admin”?

“Admin” är ett system för bevakningstjänster. Detta system har tre applikationer som är sammanlänkade. Det finns en mobilapplikation för den patrullerande väktaren, en officeapplikation för administrativa arbetsuppgifter och en webapplikation för kunden

som kan följa en väktares patrullering. Vidare innefattar systemet bland annat hantering av kontrakt, bevakningsobjekt, administrationsverktyg, uppdragsplanering samt kommunikation med kund.

“Admin” är ett webbaserat SaaS system, vilket i praktiken innebär att Bevakningsföretaget står för driften av systemet och dess kunder har alla sina uppgifter i deras servrar. Denna affärsidé är skapad eftersom de vill göra det lätt även för mindre företag inom samma bransch att använda systemet. Bevakningsföretaget är därför mån om att systemet ska vara användbart för alla typer av användare och genom erfarenhet vet de att de tekniska kunskaperna skiljer sig åt hos kunderna.

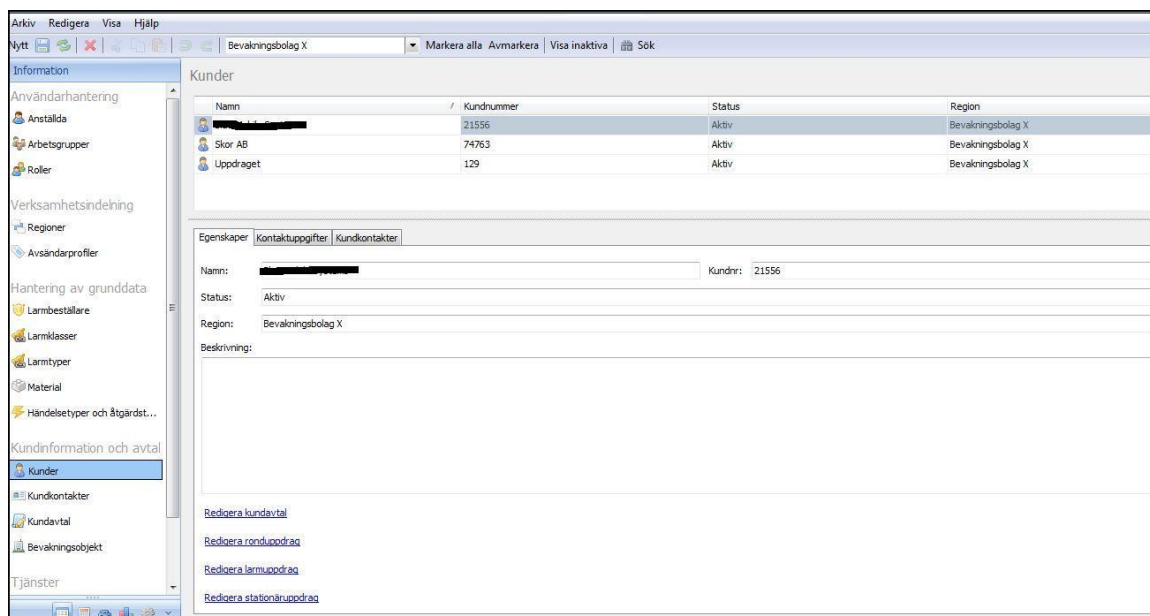
3.2 Funktionaliteten i ”Admin”

“Admin” har som huvudfunktionalitet att hantera kontrakt, planera ronder, hantera bevakningsobjekt. Systemet består även av analys- och rapporteringsverktyg.

Denna uppsats fokuserar på hantering av kunder, kundkontakt, kundavtal och bevakningsobjekt. Dessa funktioner är idag tidsslukande och kräver att användaren följer ett speciellt processflöde för att kunna fullfölja sina registreringar. Vid avvikelse från detta, exempelvis vid saknade uppgifter, så kan användaren inte slutföra sin registrering.

1. Kunder

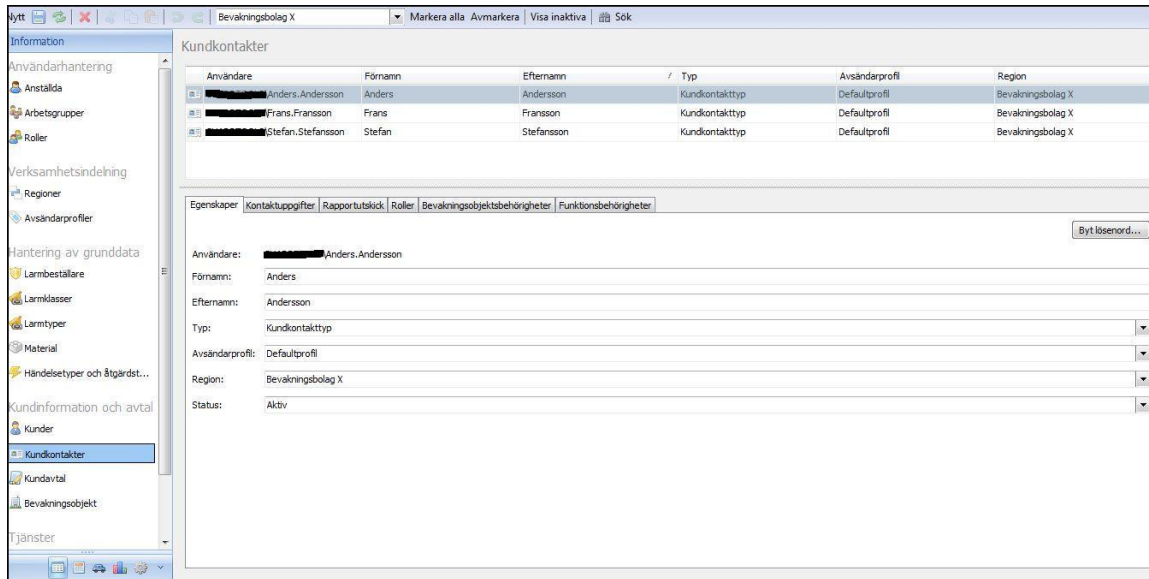
När användaren har tecknat ett kontrakt med en kund och vill lägga in avtalet i “Admin”, går användaren in på startsidan för kunder och registrerar den nya kunden. På denna vy registreras avtalet, kunduppgiften och kontakt uppgifter. Varje kund har ett avtal som gör att kunden är registrerad som aktiv i “Admin”.



Figur 4. Bild över vyn för kunder.

2. Kundkontakter

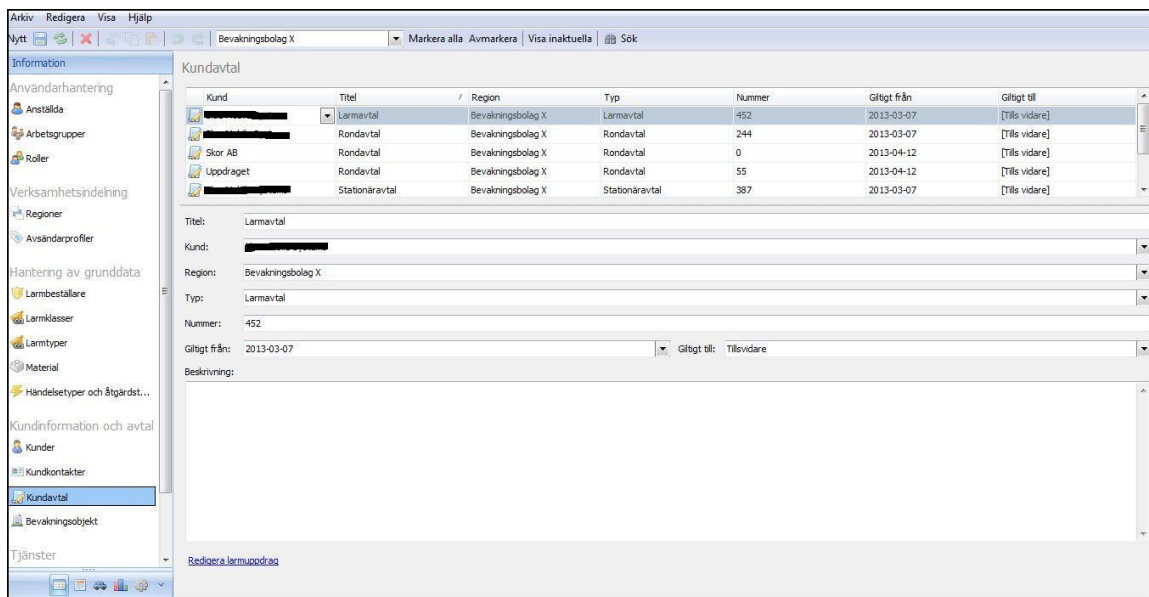
En kundkontakt är den som bevakningsföretaget har kontakt med på det företag som köpt en bevakningstjänst. För varje kundkontakt bestäms vilken behörighet denna ska ha i webbportalen, hur ofta och vilken typ av rapporter som ska skickas till kundkontakten samt hur dessa rapporter ska skickas.



Figur 5. Bild över vyn för kundkontakter.

3. Kundavtal

Kundavtalet är den information som rapporteras in för varje bevakningsobjekt, och specificerar vad varje sådant avtal innehåller. Varje delavtal har egna id-nummer, giltighetsdatum samt information om de bevakningsronder som ingår i avtalet. Varje ramavtal innehåller mycket detaljerad information.

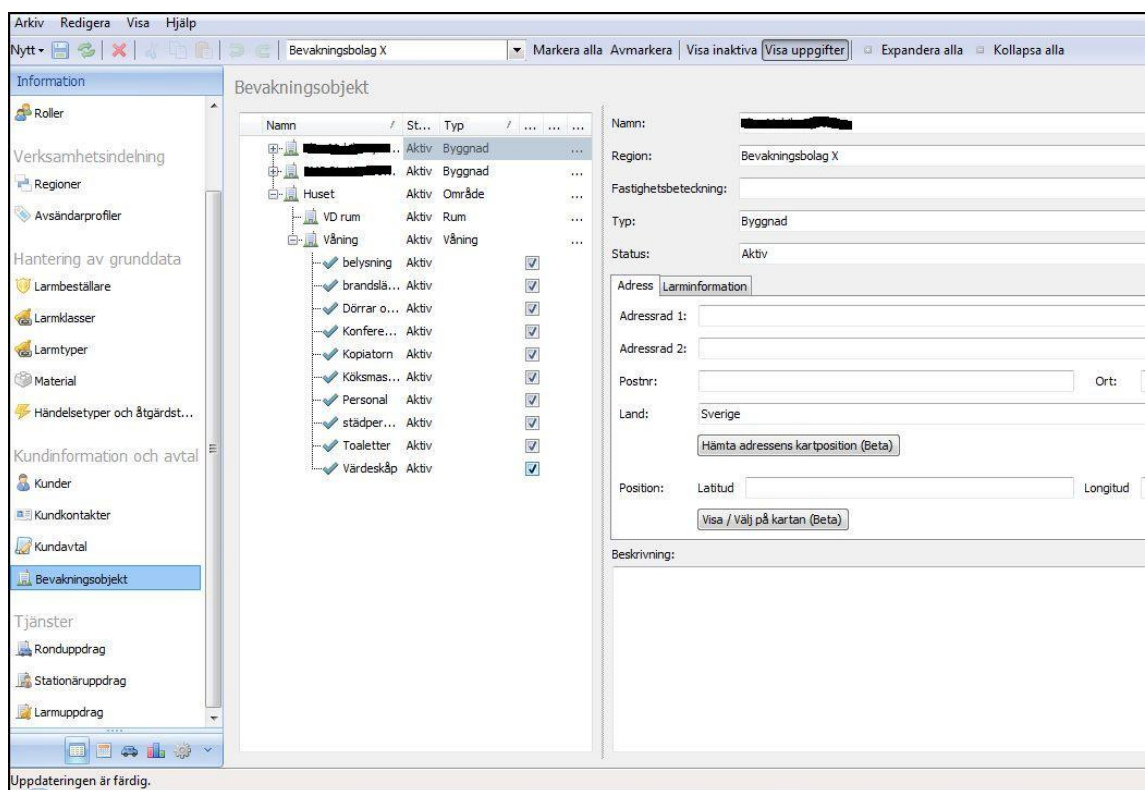


Figur 6. Bild över vyn för kundavtal.

4. Bevakningsobjekt

Ett bevakningsobjekt är den fysiska plats som väktaren ska patrullera och kontrollera. När den fysiska kontrollen görs, ska det finnas registrerade ett antal kontrollpunkter. Vid varje kontrollpunkt så finns en instruktion om hur kontrollerandet ska gå tillväga och väktarna ska verifiera att de har utfört kontrollen.

Dessa kontrollpunkter läggs in i väktarens handenhet och rapporteras till bevakningsföretagets Officeapplikation och webbportal, så att kunden kan ta del av innehållet och läsa de skapade rapporterna. Användaren har även i detta läge möjlighet att placera ut checkpunkter via en karta istället för att manuellt skriva in de. Detta för att det kan underlätta att planera en effektivare rond när en visualisering presenteras.



Figur 6. Bild över vyn för bevakningsobjekt.

4. Metod

En fallstudie av ett befintligt system har genomförts. Denna fallstudie har genomförts med stöd utav företaget som utvecklar systemet. En kombination av intervjuer, observationer och utbildning har gjorts för att få en egen utvärdering av systemet. Detta för att kunna relatera till litteratur och kunna ta fram förbättringar till systemet. Detta arbetssätt ger en vid infallsvinkel på problemområdet [10].

Det har i studien valts litteratur efter omfattande litteratursökningar. Författarna av uppsatsen har tagit hänsyn till och varit medveten om att undersökningen och dess resultat kan ha påverkats av bland annat forskarens inom teorins egna uppfattningar och förhållningssätt. Därför har jämförelse av litteratur skett för att studera olika författares åsikter.

De finns en del olika principer som är bra att följa och ha med sig då man hanterar en studies insamlingsdata. Dessa principer behandlar bland annat att varje studie är unik och innefattar olika resultat, feltolkningar och motsägelsefulla förhållanden i de uppgifter som samlats in i studien. Vid analysering av insamlingsdata har principer utvecklade av Klein och Myers används som stöd [6].

4.1 Utbildning

För att skapa en uppfattning av hur användarnas introduktion till systemet sker, så genomgicks det en deltagande observation vid ett utbildningstillfälle som utvecklingsföretaget tillhandahåller. Utbildningstiden varade under en halvdag. Denna observation genomgicks eftersom det behövdes en grundförståelse för systemet inför kommande moment. Utbildningstillfället genomfördes med föreläsningar och navigering i systemet. Ingen access till systemet gavs förrän efter utbildningen.

4.2 Observation

En fördjupad inblick i systemet har gjorts och en utvärdering med hänsyn till olika designprinciper[9]. Efter att ha granskat systemet delades systemet upp i olika vyer och några vyer valdes till ett mer ingående studerande. De utvalda delarna utgjorde grunden till intervjufrågor med olika användare. De delar av systemet som utvaldes var de delar där användaren registrerar kunden, delen om kundens avtal och den del om bevakningsobjekt som hanteras av ett avtal. För att få en bättre förståelse av systemet så genomfördes övningsuppgifter i systemet för att försöka få insikt i hur det dagligen används. Förhoppningen med detta var bland annat för att få insikt i vilka delar av systemet som anses vara svårförståeliga och varför de är det. Denna simulerade användningen av systemet är nödvändig för att kunna avgöra vilket fokus intervjuerna ska följa.

4.3 Intervjuer

Ett val gjordes att utföra intervjuer eftersom användarnas synpunkter utgör en behövlig grund till analysen av ”Admin” användarvänlighet. De utvalda informanterna består av dels användare som arbetat i systemet i flera år, medan andra är nybörjare. Detta för att få en spridd kunskap om systemet, som bas till analysen. Informanterna har vid intervjutillfället befunnit sig framför systemet för att ha en aktuell bild av det, eftersom valet av vyer inte ska utgöra ett hinder.

Efter valet av vyerna skapades det huvudteman som stod i fokus under intervjuerna, vilka är grundade på Normans designprinciper. Varje intervju spelades in för att kunna underlätta jämförelserna av de olika svaren. En intervju tog ca 20 minuter att genomföra och denna tid var tillräcklig för att förstå användarens synpunkt. Samtliga informanter gav medtycke till inspelning av intervjun.

De intervjuade använder dagligen fallstudieobjektet. Informanterna är i olika åldrar och med olika erfarenheter av IT, vilket bidragit till att den insamlade datan givit en bred bild av hur systemet fungerar användarmässigt.

Informant 1 - Har använt systemet i två år. Gått från tidigare pappersarbete till arbete med systemet. Är en ovan IT-användare

Informant 2 & Informant 3 - Har använt systemet sen det skapats. Har gått från tidigare pappersbaserat system. Är en van IT-användare.

Informant 4 - Har använt systemet i fyra månader. Är en ovan IT-användare.

Informant 5 & Informant 6 - Har använt systemet i flera år. Anser sig vara vana IT-användare.

5. Resultatredovisning/analys

Målet i detta stycke är att diskutera generella riktlinjer för att göra informationssystem mer användarvänliga samt att ge konkreta förbättringsförslag till "Admin".

Efter att ha analyserat systemet kommer designprinciperna visibility, feedback och mapping vara huvudfokus för den resultatredovisning som detta stycke behandlar. Detta eftersom det efter utbildningstillfället och observationen är de designprinciper som fallstudieobjektet har flest brister i. Nedan finns resultatet redovisat med citat från informanterna, en förklaring på vad citatet innebär samt ett konkret förbättringsförslag till utvecklaren av produkten "Admin".

5.1. Synlighet (Visibility)- Can i see it?

Systemet visar inte tydligt hur användaren ska gå tillväga för att registrera en kund i systemet. Användaren får gissa sig till var man ska börja och hur man ska gå vidare. Det finns genvägar i systemet, men dessa är otydliga.

Utifrån intervjuerna har det inom designprincipen visibility framkommit brister. Här identifieras dessa brister och förbättringsförslag ges. Nedan har informanterna citerats och det redovisas även förklaringar till informanternas åsikter.

Brister med förbättringsförslag:

1. "Har man en någorlunda teknisk bakgrund så borde man kunna klara sig igenom det"- Informant 2

Förklaring: Användarna måste ha en viss teknisk bakgrund för att kunna förstå systemet. **Förbättringsförslag:** Att lägga in tillräckligt med hjälp- och informationstexter som ska kunna hjälpa användare med en mindre "teknisk bakgrund". Eftersom gränssnittet inte är tydligt behövs bättre information för hur man exempelvis ska fylla i vissa fält. Därutöver bör man förbättra och förtydliga onlinehjälpens ytterligare.

2. "Vi har valt att fylla i nyckelnummer i anteckningarna" - Informant 1

Förklaring: Vissa specifika fält saknas, som i detta fall ett för nyckelnummer. Detta medför att användaren tvingas att fylla i vissa uppgifter under fältet "Anteckningar" istället. Detta är dåligt av flera orsaker, varav en är att "Nyckelnummer" inte skulle kunna bli sökbart i systemet.

Förbättringsförslag: Alla fält som är behövliga bör finnas specificerade och anteckningsfältet borde kunna hållas fritt från specifika sökvärden. Om användarna

behöver ett nytt fält bör detta kommuniceras till systemdesigners som ett användarkrav, vilka senare kan implementeras i en ny version av systemet.

3. "Det tar lång tid att lära sig systemet. Det finns säkert mängder av funktioner som jag inte nyttjat" – Informant 3

Förklaring: Det upplevs som ett avancerat system som är svårt att lära sig samt att det saknas en överskådlig bild över systemet.

Förbättringsförslag: Informera om online-hjälpen så att alla användare vet hur man hittar denna och hur den fungerar. Förbättra online-hjälpen vid behov till att bli mer lättillgänglig och lätt att förstå. Utöka online-hjälpen med fler överskådliga bilder.

4. "För många menyer som man ska jobba emellan"- Informant 1

Förklaring: Det finns för många huvudmenyer att arbeta med.

Förbättringsförslag: Göra det möjligt att kunna se, öppna och i bästa fall även kunna arbeta i, flera menyer på samma gång. Det bör också vara enkelt att växla mellan olika menyer. Man bör jobba igenom menysystemet och göra det mer sammanhängande med undermenyerna. Man bör göra en användarundersökning för att ta identifiera de menyer som upplevs som mest röriga.

5. "Jag vill kunna söka ut särskilda värden i listor och inte bläddra igenom t.ex. 60 punkter"- Informant 5

Förklaring: Många listor i systemet är endast scrollbara istället för sökbara, vilket gör att man måste bläddra mycket mellan olika bevakningsobjekt.

Förbättringsförslag: Gör det möjligt att kunna söka med sökfunktion efter valda objekt istället för att scrolla, alternativt göra det möjligt att med hjälp av urvalskriterier klicka sig fram till önskat objekt. Försöka få plats med så mycket viktig information på en sida med som möjligt, och i möjligaste mån undvika för långa scroll-listor.

6. "Det kan bli mycket tydligare angående vart man ska börja. Jag måste använda hjälpen som finns" Informant 4

Förklaring: Det är svårt att förstå i vilken vy och i vilket fält som man ska starta i, om man ska utföra en viss uppgift.

Förbättringsförslag: Ge en tydlig indikation på var man befinner sig i systemet. T.ex. genom "breadcrumbs" [14]. Använd flera olika färger och blinkande markörer för att visa var i systemet användaren befinner sig. Ge pop-up-hjälp [14] vid rutor som ska fyllas i och detta i en rätt ordning. Vid behov kan bevakningsföretaget även anordna seminarium där man utbildar användaren i de vanligaste scenariorna de kommer befinna sig i systemet. I online-hjälpen skulle man också kunna förevisa olika scenarion genom ett visuellt flöde visa användaren hur en funktion är uppbyggd.

7. "Man måste jobba kontinuerligt i systemet för att inte glömma bort funktioner"- Informant 6

Förklaring: Systemets funktioner är lätta att glömma om man inte jobbar ofta i systemet, eftersom många funktioner är avancerade.

Förbättringsförslag: Gör lättillgängliga step-by-step beskrivningar på de vanligaste funktionerna och placera dessa i online-hjälpen. Försök att förenkla för användaren med så mycket olika sorters feedback och hjälp som möjligt [9].

Riktlinjer: Användaren ska förstå vad som är nästa steg i systemet. En användare ska inte behöva skriva egna mallar för hanterandet av ett system.

Under intervjuerna framkom även att användare behöver fylla i samma uppgifter flera gånger. Detta är inte användarvänligt och tar mer tid i anspråk än nödvändigt. Ett ytterligare förbättringsförslag under är att möjliggöra för en automatisk kopiering av de värden som ska fyllas in på ett flertal positioner. Detta för att användarna inte ska behöva fylla i samma värde flera gånger vid registreringen av exempelvis en ny kund. Istället ska systemet själv sköta detta.

Systemet bör därutöver vara anpassat för vana IT-användare som dagligen använder sig av kort-kommandon (ex. Ctrl+S).

5. 2. Återkoppling (Feedback) - *What is it doing now?*

Användaren får sparsamt med feedback från systemet[9]. Den feedback som ges är framförallt vid två olika tillfällen: en ruta kommer upp och frågar om användaren vill spara gjorda ändringar samt en ruta som kommer upp då användaren matat in ett felaktigt värde. Feedback i systemet ges främst via popup-rutor[14] utan ljud.

Utifrån intervjuerna har det inom designprincipen feedback framkommit brister. Här identifieras dessa brister och förbättringsförslag ges. Nedan har informanterna citerats och här redovisas även förklaringar till informanternas åsikter.

Brister med förbättringsförslag:

1. "Du måste veta att allt du gör är rätt", "Det vore bra att få upp en ruta om man skriver in felaktiga uppgifter"- Informant 1

Förklaring: Eftersom systemet knappast ger någon feedback, måste användarna själva veta att det de gör är rätt. Ingen feedback ges vid inmatning av felaktiga uppgifter, t.ex. om ett inmatat postnummer inte finns.

Förbättringsförslag: Ge feedback till användaren om denna lägger in ett felaktigt värde, ogiltigt värde eller om ett obligatoriskt värde saknas[9]. Designa systemet så att det kontrollerar att användaren lägger in rimliga värden och dessutom ger feedback till användaren på de ställen i systemet där detta saknas. Exempelvis om ett postnummer inte finns ska det ges feedback till användaren om att värdet är felaktigt inmatat.

2. "Har man glömt spara ner det, eller det blir någon bugg så kan allt försvinna" – Informant 3

Förklaring: Användarna är rädda för att information ska försvinna. De trycker på "spara"-knappen hela tiden för att försäkra sig om att inget går förlorat.

Förbättringsförslag: Fråga "Vill du spara de ändringar du gjort? Ja/Nej?" Ställ denna fråga med jämna mellanrum till användaren och därigenom slipper användaren själv komma ihåg att spara efter varje ändring.

3. ”Har inte någon fått någon rapport så hör de av sig”- Informant 2

Förklaring: Istället för att få feedback från systemet som sig bör, så får användarna istället veta från kunderna om systemet inte har genererat någon rapport.

Förbättringsförslag: Användaren bör få feedback genom en pop-up[14] varje gång en rapport har sänts till en kund.

4. ”Svårt att veta att allt kommit med, vad man än trycker på”- Informant 4

Förklaring: Systemets informationsrutor innehåller otillräcklig information och otydliga val.

Förbättringsförslag: Förtydliga de val och ändringar som användaren gjort innan dessa sparas. Ett exempel är att markera viss text eller fält som har ändrats genom att ge annan färg på texten eller liknande, så det blir tydligt för användaren vad denna har ändrat.

5. ”Jättetacksam om jag skulle få flera feedback-rutor” – Informant 4

Förklaring: För lite feedback till användaren vid vardaglig användning.

Förbättringsförslag: Generellt sett ge mer feedback i olika vyer som beskriver vad som händer i systemet exempelvis vid ändringar eller under processtid/väntetid. Gärna en tidsmarkör som visar hur långt processen har hunnit eller alternativt en rörlig symbol som beskriver att systemet ”tänker”[9].

6. ”Jag måste själv veta att man måste trycka på uppdatera efter spara”- Informant 6

Förklaring: Det räcker inte med att spara utan användaren måste därutöver trycka på att ”uppdatera”. Det finns ingen hjälptext som beskriver detta.

Förbättringsförslag: Antingen lägga in en hjälptext ”Du måste trycka på UPPDATERA för att se dina ändringar!” alternativt fundera på om man vill designa om systemet så att vyn uppdateras automatiskt varje gång efter en användare har sparat och därigenom ta bort uppdatera-funktionen.

7. ”Är du en van IT-användare har du säkert ett bättre hum om hur system är uppbyggda och kan acceptera en lägre grad av feedback.”- Informant 3

Förklaring: Vissa behöver få mer hjälp och feedback än andra när det gäller hanteringen av systemet.

Förbättringsförslag: Möjlighet att välja en ”hjälp-nivå” eller motsvarande. Detta innebär att man gör två nivåer av feedback och hjälptexter. Användaren har möjlighet att välja nivå när man loggar in i systemet genom att välja ”nybörjare” respektive ”avancerad”. Detta kan dock bli en ganska svår lösning. En enklare lösning är att ha en och samma nivå för alla, men designa den för nybörjare, där man är generös med feedback, hellre en pop-up [14] för mycket än en för lite.

Riktlinjer: Var generös med feedback till användaren. I ett system bör det finnas feedback som harmonierar med användarens behov. Att kunna anpassa feedback efter användaren är bra om det går, men viktigast är att ge ordentligt med feedback så att användaren känner sig trygg. Oftast vill användare få bekräftelse på att alla uppgifter är ifyllda, sparade och att de skickats vidare till rätt person. För en ovan användare är kontinuerlig feedback viktigare än för en van användare som ibland kan uppleva mycket

feedback som ineffektivt, men en regel bör vara att hellre vara övertydlig än att systemet ger för lite feedback. Tänk igenom var det är viktigt med feedback för att skapa ett användarvänligt system.

5.3 Navigering (Mapping) - where am I and where can I go?

Systemet är utformat utifrån ett pappersbaserat system. Systemet har senare vuxit och blivit mer omfattande. I det ursprungliga pappersbaserade systemet registrerades uppgifterna utifrån ett processflöde, vilket inte syns i det nya systemets design. Det är därför svårt att förstå vid en första anblick vart användaren ska börja arbeta om man är ny i bevakningsbranschen i. Har användaren arbetat tidigare i bevakningsbranschen så är processflödet naturligt och användaren vet vad den ska leta efter.

Utifrån intervjuerna har det inom designprincipen Mapping framkommit brister. Här identifieras dessa brister och förbättringsförslag ges. Nedan har informanterna citerats och här redovisas även förklaringar till informanternas åsikter.

1. "Man tror att man är klar, men så kommer de upp något man glömt några dagar senare"- Informant 4

Förklaring: Man vet inte riktigt när man är "klar" med en arbetsuppgift.

Förbättringsförslag: Förmedla när användaren har gått igenom alla stegen och är klar med den nuvarande uppgiften.

2."Det finns ingen sammanhängande tråd i systemet"- Informant 5

Förklaring: Svårt att veta vart man är på väg i systemet och att man utfört allt som behövs.

Förbättringsförslag: Ge tydligare hjälp för användaren och tydliggör var alla uppgifter i systemet bör befinna sig. Ge användaren en tydligare förklaring genom hjälptexter angående vad varje fält ska innehålla.

3. "Man får ibland fuska sig igenom för att det ska fungera"- Informant 5

Förklaring: Ibland får användarna gå omvägar och navigera efter eget huvud för att systemet ska fungera. Systemet bör vara mer begränsat angående vad användarna kan utföra och när. System ska endast leda användaren genom rätt arbetsprocess.

Förbättringsförslag: Använd olika kontrollfunktioner som gör det omöjligt att kunna felnavigera i systemet. Kontrollera att användare fyllt i rätt uppgifter och ge användaren tydlig vägledning via online-hjälpen.

4. "Om jag skulle vilja göra likadana ronder så går det inte att klippa och klistra"- Informant 2

Förklaring: Det saknas en funktion för att kunna klippa och klistra med exempelvis kortkommandon.

Förbättringsförslag: Att implementera standardkommandon, exempelvis Ctrl+C och Ctrl+V för att användaren exempelvis ska kunna kopiera och klistra mellan olika vyer på ett enklare sätt.

5. "Ibland måste man fylla i samma uppgifter minst fyra gånger"- Informant 6

Förklaring: Identiska uppgifter fylls i flera gånger vid exempel registrering av en ny kund. **Förbättringsförslag:** Använd autofyllnad på samma uppgifter i olika vyer för att underlätta för användaren.

Riktlinjer: En process i ett system ska kunna följas och uppgifterna som fyllts i på en vy ska automatiskt kopplas till nästa vy för samma kund och objekt. Om användaren har glömt att fylla i något fält så ska det tydligt förmedlas till användaren.

6. Diskussion

Här presenteras de riktlinjer som är de viktigaste för konstruktionen av användarvänliga system. Dessa fyra riktlinjer är "Tidigt fokus på användarna", "Uppbyggnad för snabb inläring", "Bekräftelse" och "Utbildning i mindre omfattning".

Ett system ska vara enkelt för användaren att manövrera och att utföra sina arbetsuppgifter i. Användaren ska även kunna utföra svåra uppgifter på ett enkelt sätt, exempelvis kan snabbkommandon främja effektiviteten. Olika möjligheter behöver finnas så att en van användare inte upplever att systemet är ineffektivt och att systemets kapacitet utgör ett hinder. Ett flexibelt utformande av systemet bör eftersträvas för att kunna tillgodose olika användares behov[3].

Denna studie har fokus lagts på utvalda designprinciper från Norman [9] och inslag har tagits från andra forskare såsom Nielsen et.al. I fallstudien har brister uppmärksammats som kan identifieras med hjälp av Normans designprinciper och influenser från andra forskare [8, 12]. Med detta som grund har studien försökt att skapa ett vidare begrepp av riktlinjer för designern. Det som har kommit fram från studien och det som är viktigt för litteraturen är sammanvävt i detta avsnitt.

6.1. Tidig fokus på användarna

En av de fyra riktlinjerna som framkommit under studien är tidig fokus på användare. Denna riktlinje innebär att de som designar systemet ska börja med att göra en analys om vilka deras användare är, deras behov och vilken IT-vana användarna har [3]. I utvecklingsprocessen bör verksamhetens effektivitetsmål uppfyllas för att användarna ska få bästa möjliga användning av systemet. Dessa effektivitetsmål uppfylls genom designen [4]. Exempel på möjliga effektivitetsmål är att vyerna är placerade i en logisk ordning [14], att man ska kunna registrera en ny kund på en rimlig tid.

För att utvecklaren ska kunna ge feedback [9] till deras användare bör utvecklaren samla in och dokumentera alla användares kommentarer. Därefter ska man i utvecklingsprojektet ta upp kommentarerna och avgöra om man ska vidta några åtgärder. Sedan bör man meddela användarna om systemet har ändrats till följd av deras kommentarer och i så fall varför eller varför inte de har gjorts [3].

När man fokuserar tidigt på användaren innebär det att man fokuserar på hur man genom gränssnitt och utformning kan stötta användaren. Även en ovan användare ska klara av att utföra mer komplexa arbetsuppgifter. Det ska inte krävas att en användare är expert på systemet för att kunna arbeta i det [3].

6.2 Uppbyggnad för snabb inläring

I liknelse med riktlinjen för tidig fokus på användare, visade studien att även uppbyggnad för snabb inläring är viktigt. Genom att anpassa gränssnittet till ett logiskt flöde [14] skapar detta en snabbare inlärningsprocess. Detta flöde ska vara utformat så att det är tydligt hur man som användare ska gå tillväga vid registreringen av sina uppgifter som tex en visuell [14].

Studien har visat att ett systems gränssnitt ska vara utformat så att det är lätt att lära sig det. För att det ska vara lätt att lära sig bör gränssnittet innehålla allmänt vedertagna utformningar och sträva efter kontinuitet [12]. Det ska finnas en tydlighet i vad som är den kronologiska ordningen och det ska tydligt visas för användaren. Tydlighet kan beskrivas som att användaren ska förstå vid en första anblick hur flödet ser ut tex en wizzard. Användaren ska kunna se att det finns fler steg än det aktuella i processen och hur långt användaren har kvar tills att uppgiften är slutförd. Att visa antalet steg för användaren kan minimera stressfaktorn hos användaren på grund av att användaren känner att tidsåtgången inte blir så stor som användaren förväntat sig [14]. Som exempel kan man designa det till flikar som användaren måste följa och användaren ska tydligt se att det är sparad och att alla obligatoriska uppgifter i de olika fälten är ifyllda.

Ett lättlärt system med mycket feedback [9] ökar toleransen hos användaren för att göra fel. Genom att användaren får reda på vad som är fel ifyllt så kan användaren lätt åtgärda felet själv. Det minskar irritationen hos användaren och ger användaren en känsla av kontroll av systemet, användaren känner att de har kontroll över systemet [1]. Ett lättlärt system ger en bättre minnesbild av systemet och användaren kommer lättare ihåg hur de gjorde senast. Minnesbilden är bestående och den är positiv istället för att den sista känslan av systemet är att systemet känns omständigt [11].

Det ska vara möjligt för användaren att kunna individualisera hur systemet används eftersom användare har olika intresse i att hantera systemet. Det bör i möjligaste mån vara styrbart av användaren för att motsvara användarens förkunskap [1].

När man implementerar ett informationssystem kan verksamheter gå från pappersbaserade system till ett informationssystem. Detta har medfört att de flesta informationssystemen är uppbyggda utefter de pappersbaserade systemen. Användarcentrerad systemdesign är därför inte i fokus då utvecklarna endast försöker "översätta" de pappersbaserade systemet till ett informationssystem.

6.3 Bekräftelse

Studien har visat att en användare behöver en bekräftelse av sitt handlande i likhet med designprincipen feedback. I detta sammanhang handlar det om att få en visuell feedback på det användare nyss utfört [9].

För en van användare får man som systemutvecklare göra en noggrann avvägning var feedback ska finnas för att inte skapa känslan av ett ineffektivt system [9]. Det kan till exempel handla om att systemet innehar funktionen "autospar" och varje gång "autospar" sker så aktiveras en räknare, så att användaren ser hur lång tid som är kvar. Denna "loading indicator" [14] måste kunna aktiveras samtidigt som användaren använder

systemet. Avbryter räknaren användandet upplevs det som en belastning istället för en hjälp.

Studien har visat att en ovan användare har ett större behov av feedback på sin användning. En ovan användare behöver få en tydlig beskrivning över vad som gick fel i användandet för att kunna åtgärda felet. Feedback ska komma antingen omedelbart eller direkt efter ett steg i utförandet. Användaren ska inte hunnit glömma av det steg som precis blivit fel. Felet ska fångas upp inom en rimlig tidsgräns så att det inte ger användaren en känsla av ineffektivitet [9]. Anpassad feedback/bekräftelse utefter användaren gör att användaren upplever att systemet motsvarar deras förväntningar. Användare med olika förutsättningar har olika typer av användarupplevelser och förkunskaper.

6.4 Utbildning i mindre omfattning

Studien har visat att ett system är som mest användbart om gränssnittet är utformat efter nämnda designprinciper och användbarhetskrav enligt Nielsen, Gulliksen & Göransson, Schneidermans och Norman. Detta ska bidra till att användaren ska kunna komma igång med systemet så snabbt som möjligt och förstå dess funktionalitet utan specifika instruktioner [8]. Därför ska en avancerad utbildning på ett system inte behövas eftersom systemet redan från första starten ska vara tillräckligt intuitivt och visa en tydlig design [1]. Studien har visat att det borde endast vara förkunskaper som behövs för att kunna använda ett system ska handla om yrkeskunskap. Yrkeskunskap gör att systemet blir ett komplement till tidigare kunskap, detta sammanvävt skapar en helhet för användaren.

När man använder sig av ett informationssystem i sitt arbete är systemets roll att vara ett komplement. En användare ska kunna följa sin "Action cycle" och då kunna använda systemet på bästa sätt. Genom att intuitivt tänka i "Action cycle", ska systemet vara utformat så att användaren vet var den är, vart den är på väg och när den är klar. Ett system utformat på detta sätt medför att avancerad utbildning inte behövs[9].

Ett gränssnitt ska inte behöva påfresta användarens minne på ett orimligt sätt. Det ska vara tydligt begränsat i systemet vad som är tillåtet och hur användaren ska gå tillväga [12]. Studien har visat att en användare inte ska behöva återgå till sitt utbildningsmaterial för att komma ihåg hur man arbetar i systemet.

7. Slutsats

Frågeställning: Vilka principer och krav bör följas vid utformandet av användarvänliga informationssystem?

De huvudsakliga punkterna som har framkommit i denna studie har handlat om designprinciper. Dessa är några av de viktigaste principerna som ska följas vid utvecklandet av ett användarvänligt informationssystem. En blandning mellan designprinciper utformade med olika syften har genererat generella riktlinjer anpassade för individens systemanvändning.

Eftersom många användare har olika erfarenheter av IT-arbete bör alla informationssystem vara utformat mer enligt en universal design som ska passa alla typer

av användare. Genom att följa generella riktlinjer vid designen utav ett system så kommer systemet att bli mycket mer användarvänligt, vilket gör att utbildningsbehovet för användarna av systemet minskar.

Ett utbildningstillfälle i systemet ska hjälpa användaren att starta upp systemet och ge en bas till användandet. Genom denna bas ska användarens själv kunna gå vidare i inläringen av systemet på egen hand. Det ska inte behövas mycket extern support för att kunna använda ett informationssystem.

Något av det viktigaste är att man redan vid påbörjandet av utvecklandet av ett system ska fokusera på användarna. Framförallt eftersom att det är komplicerat att bygga om ett helt system i efterhand för att göra det mer användarvänligt.

De riktlinjer och principer som har behandlats i studien kan appliceras på en mängd informationssystem idag. Ett system som är användbart och effektivt för ett företag är idag en ekonomisk fråga. Genom att spara tid på utförandet av en uppgift blir det en kostnadsbesparing för företagen.

Fallstudieobjektet i denna uppsats är ett typexempel på en produkt som inte följer användbarhetskrav och designprinciper. Fallstudieobjektet har visat tydliga brister inom funktionaliteten. Det har bland annat visat sig att det knappt ges någon feedback alls till användarna, vilken är en viktig bit att arbeta vidare med.

Granskandet av fallstudieobjektet har visat att systemet inte har något speciell fokus på användaren, utan är uppbyggt enligt föregående papperbaserade system. Detta har lett till att många användare som tidigare arbetat med pappersystem, har lättare att lära sig systemet än de som aldrig jobbat med papperbaserade system.

När ett system blir användarvänligt så skapar det en positiv attityd hos användare. Idag finns studier som visar på att en hög andel av den enskilda individens irritation oftast orsakas av system som är svåra att använda. Den enskilda individen har helt enkelt vant sig vid att kunna använda system vid alla möjliga tänkbara situationer. Om det vid några tillfällen inte skulle fungera så skapas en irritation eftersom systemet då upplevs som begränsande.

8. Referenser

1. Connell. B, Jones.M, Mace.R, Mueller.J, Mullick.A, Ostroff.E, Sanford.J, Steinfeld.E, Story.M, och Vanderheiden.G, NC State University, The Center for Universal Design, Version 2.0 - 4/1/97; 1997 [Hämtad 2013-04-25]
Tillgänglig: http://www.ncsu.edu/www/ncsu/design/sod5/cud/about_ud/udprinciplestext.htm
2. Emilsson Hammarberg.K, *Användbarhet vs användarvänlighet*; Lumano; 2002
[Hämtad 2013-06-04]
Tillgänglig: <http://www.lumano.se/Artiklar/AnvandbarhetVersusAnvandarvanlighet>
3. Gulliksen.J, Göransson.B, *Användarcentrerad systemdesign – en process med fokus på användare och användbarhet*. Lund: Studentlitteratur; 2002.
4. ISO 9241-11 *Ergonomic requirements for office work with visual display terminals (VDTs) – part II: guidance on usability*. Switzerland: International Organization for Standardization; 1998
5. Karat.J, *User Centered Design: Quality or Quackery?*. The ACM/SIGCHI Magazine; 1996
6. Klein, H.K., Myers, M.D, *A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems*, MIS Quarterly 23 (1), 67-93; 1999
7. Krug. S, *Don't make me think - A Common Sense Approach to Web Usability* (2:a uppl.), New Riders Publishing, Berkeley, Kalifornien, USA; 2006
8. Nielsen.J, *Usability Engineering*. Morgan Kaufmann Publishers; 1993.
9. Norman.D, *The Design of Everyday Things*. The MIT Press. London; 1998
10. Patel.R. & Davidsson.B, *Forskningsmetodikens grunder. Att planera, genomföra och rapportera en undersökning*, (Tredje upplagan). Studentlitteratur AB; 2003
11. Preece.J., Rogers.Y., Sharp.H., Benyon.D., Holland.S., and Carey.T., *Human-Computer Interaction*. Wokingham, UK;1994
12. Shneiderman.B, *Designing the User Interface*, United States of America: Addison-Wesley Longman, Inc; 1998
13. Tengblad.P, Walldius.Å, *Användningsdriven utveckling av IT i arbetslivet*: ATK Arbetsliv; Vinnova; 2007
14. Tidwell.J, *Designing Interfaces*, O'Reilly Media Inc, Sebastopol;2005
15. Bild: Figur 2, [Hämtad 2013-05-17]
Tillgänglig:http://searchuserinterfaces.com/book/sui_ch3_models_of_information_seeking.html

Bilaga 1 – Intervjufrågor

Inledande

- *Får vi spela in?*
- *Är du en van användare av IT-stöd?*
- *Hur ofta använder du systemet?*
- *När gick du utbildningen?*
- *Vilken typ av system hade ni före "Admin"?*

Visibility

- *Tycker du "Admin" utger en tydlig bild på hur man ska använda systemet genom designen? (åstadkomma de man vill i de olika processerna).*
- *Är det tydligt visat vad man kan klicka på och vart man befinner sig i systemet?*
- *Förstår du vilka uppgifter som ska registreras och vad du behöver de till? Varför?*
- *Tog det lång tid innan du kände att du verkligen förstod hur man skulle göra och varför?*
- *Vet du vart du ska börja vid exempel registrering av ny kund?*
- *Vilka uppgifter från en vy skulle du vilja få med dig till nästa vy? så du slipper fylla i samma uppgifter två gånger?*

Feedback

- *När du registrerat in uppgifter får du nån feedback på att de uppgifter du behöver för nästa steg i flödet är korrekt?*
- *Vet du att en rapport har skickats iväg eller att alla uppgifter är ifyllda i den?*
- *Hade du önskat någon audiolog feedback? Eller taktil feedback? Varför?*
- *När hade du velat ha feedback? när du sparar? eller om nåt gått fel? på vilka vyer?*
- *Feedback i systemet vid exempelvis inmatning av ny kund och postnummer. Ska feedback komma bredvid rutan om inmatat postnummer inte finns eller senare som en ruta vid försök att spara?*

Mapping

- *(Ex. du ska lägga till en ny kund) Hur går du tillväga för att utföra dina arbetsuppgifter? Varför?*
- *Vad gör du härnäst? Anser du att nästa steg i processen du ska utföra är självklar eller finns det hinder du stöter på? (Visas det tydligt vad som ska vara nästa steg i registreringen? Varför?)*
- *Vet du när du är helt klar med nåt? varför?*

Avslutning

- *I helhet, vad tycker du om systemet?*
- *Är det något du saknar? Funktioner, fält osv?*