



**LUNDS**  
UNIVERSITET

# ***Teoretiska rekommendationer kontra det praktiska arbetet vid IT-utvecklingsprojekt - med fokus på kritiska faktorer***

Kandidatuppsats, 15 högskolepoäng, SYSK02 i Informatik

Framlagd: 2013-08  
Författare: Johanna Svelenius  
Shpend Demiri  
Niklas Frithiof  
Handledare: Paul Pierce  
Examinator: Odd Steen  
Bo Andersson

Titel: Teoretiska rekommendationer kontra det praktiska arbetet vid IT-utvecklingsprojekt - med fokus på kritiska faktorer

Författare: Johanna Svelenius, Spend Demiri, Niklas Frithiof  
Utgivare: Institutionen för informatik  
Handledare: Paul Pierce  
Examinator: Odd Steen, Bo Andersson

Publiceringsår: 2013  
Uppsattstyp: Kandidatuppsats  
Språk: Svenska

Nyckelord: IT-utvecklingsprojekt, IT-projekt, Återkommande faktorer

## Abstract

Denna uppsats grundar sig i en kvalitativ studie som behandlar en analys av utvalda teoretiska rekommendationer ställda mot det praktiska arbetet vid IT-utvecklingsprojekt, med fokus på kritiska faktorer. Huruvida lyckat ett IT-utvecklingsprojekt är baseras ofta på hur väl kundens krav tillgodosätts av leverantören.

I vår studie valde vi att jämföra sex utvalda teoretiska modeller inom IT-utvecklingsprojekt, för att undersöka vilka faktorer som var mest återkommande, dessa ställdes sedan mot vårt empiriska resultat. Slutligen kom vi fram till att det viktigaste vid IT-utvecklingsprojekt är en välformulerad kravspecifikation, samt mjuka värden som syftar på kundkontakt, tillit och förståelse för hur båda parter organisationer fungerar.

### **Förklaring till ordvalet IT-utvecklingsprojekt**

Då vår undersökning innefattar både upphandlingen och utvecklingsarbetet vid IT-projekt, som saknar en korrekt gemensam benämning, valde vi att kalla hela processen för ett "IT-utvecklingsprojekt". När vi vidare i vår uppsats använder oss utav begreppet "IT-utvecklingsprojekt" syftar vi således till hela processen, från att det görs en föranalys till och med implementeringsfasen.

## Innehållsförteckning

1 Inledning och bakgrund .....	4
1.1 Problemområde .....	6
1.2 Forskningsfråga.....	6
1.3 Syfte .....	6
1.4 Avgränsningar.....	6
2 Teoretisk bakgrund och ramverk .....	7
2.1 PRV: 300 miljoner – för ingenting .....	7
2.2 Landstinget: En dyr säkerhetsrisk.....	7
2.3 Teoretiska systemutvecklingsmodeller .....	8
2.3.1 Sommervilles vattenfallsmodell.....	9
2.3.2 Sommervilles prototypintensiva utvecklingsmodell.....	10
2.3.3 Sommervilles återanvändningsmodell för mjukvaruutveckling .....	11
2.3.4 Heckmans modell för IT-utvecklingsprojekt.....	12
2.3.5 Bevan & Maguires modell för IT-utvecklingsprojekt .....	14
2.3.6 Verville & Halington modell för IT-utvecklingsprojekt.....	18
2.4 Matris över återkommande faktorer.....	20
2.4.1 De mest återkommande faktorerna .....	21
2.4.2 Icke avsevärt återkommande faktorer .....	23
2.5 Teoretisk referensram .....	25
3 Metod .....	26
3.1 Tillvägagångssätt .....	26
3.2 Val av forskningsstrategi .....	27
3.2.1 Kvantitativ metod.....	27
3.2.2 Kvalitativ metod.....	27
3.2.3 Val av teori.....	28
3.2.4 Val av respondenter .....	28
3.2.5 Respondent- och intervjuguide .....	28
3.2.6 Intervjugenomförande.....	30
3.2.7 Analys av data.....	32

3.3 Litteraturgranskning.....	32
4 Empiri .....	34
4.1 Föranalys.....	36
4.2 Kravspecifikation.....	37
4.3 Prototypbaserad systemutveckling.....	39
4.4 Validering/Verifiering.....	40
4.5 Iterationer .....	41
4.6 Tid/Kostnad/Kvalitet.....	41
4.7 Mjuka värden .....	42
5 Analys .....	44
5.1 Föranalys.....	44
5.2 Kravspecifikation.....	44
5.3 Systemutveckling/prototyper .....	45
5.4 Validering/verifiering .....	46
5.5 Iteration .....	47
5.6 Mjuka Värden .....	48
5.7 Tid, Kostnad och Kvalitet.....	49
5.8 Sammanfattning av analys .....	49
6 Diskussion och Slutsats.....	50
1 Intervjusammanställningar.....	54
1-A .....	54
1-B.....	58
1-C.....	63
1-D .....	67
1-E.....	70
2 Intervjufrågor .....	75
7 Referenser .....	76

# 1 Inledning och bakgrund

Dagligen genomförs mängder av IT-projekt och leveranser av system, mellan diverse kunder och leverantörer. Av dessa kommer de flesta inte längre än till förhandlingsstadiet innan de läggs ner, och projektet ses som misslyckat. Malmqvist (2012) skriver i sin artikel "*Det har gått troll i överklaganden*", om en undersökning utförd av Computer Sweden, där det avslöjas att överklaganden av generella IT-implementeringar i den offentliga sektorn ökat med 50 % under år 2012. Detta innebär ett stort problem i dagens samhälle, i form av hävda köp och miljonbelopp som förloras. (Malmqvist, 2012).

IT-utvecklingsprojekt är ett brett begrepp som vi tidigare nämnde kan omfatta allt från ren utveckling, outsourcing, en uppgradering eller implementering, till och med avveckling av IT-stöd. Några gemensamma nämnare för dessa typer av projekt är dock att de är komplexa, resurskrävande och tidskrävande i jämförelse med en ordinär, daglig rutinartad arbetsinsats. En studie genomförd av GartnerGroup som innehöll en analys av 70 000 projekt och 500 workshops med projektledare, projektmedlemmar och chefer visade att 30 % av IT-projekten aldrig når upp till de förväntade resultaten. I genomsnitt överskrider så mycket som 51 % sin utsatta budget.

Vad är då orsakerna till att så många IT-projekt misslyckas? Enligt GartnerGroup är de vanligaste orsakerna till att IT-projekt fallerar följande:

- Brist på engagerade projektägare
- Projektet stämmer inte överens med företagets strategiska målsättning
- Dålig resurshantering
- Otydligt projektmål
- En ej komplett projektplan
- Projektet innehåller inte ett formellt projektavslut där för- och nackdelar diskuteras, samt dokumenteras.

(Gartner, 2013-05-13).

Även Chaos Manifesto har genomfört en omfattande studie på lyckade och misslyckade IT-projekt, där resultaten visar på liknande statistik. Endast 30 % av alla IT-projekt lyckades, det vill säga levererades inom utsatt tidsplan, budget och med de nödvändiga funktionerna, 44 % av projekten blev försenade, gick över budget och saknade väsentliga funktioner och 24 % av projekten misslyckades helt, det vill säga kunde inte användas överhuvudtaget. (StandishGroup, hämtad 2013-05-12).

En rapport av Sveriges Kommuner och Landsting från 2008 visade att 84 % av alla IT-projekt som genomfördes i Sverige under detta år var misslyckade. Vad gäller systemutvecklingsprojekt var det färre än 10 % som lyckats. (Kontract, 2013-05-07).

Återkommande problem vid IT-projekt är att kostnader och tidsramar överskrids, vilket i sin tur medför oförutsedda extrakostnader för företagen. (Würtemberg, 2010). I en artikel av Brooks (1986) diskuteras svårigheterna med att utveckla och implementera IS/IT-system. Brooks menar att stora utvecklingsprojekt kan jämföras med en varulv, då projektet plötsligt kan utvecklas från något bekant, till något helt främmande. Det främmande grundar sig oftast i bristande planering och överskridna kostnader, vilket enligt Brooks resulterar i ineffektiva och felaktiga system. Ett direkt resultat av denna problematik, har lett till en efterfrågan av en fungerande metod, som kan hjälpa organisationer att skapa en kostnadseffektiv, säker, och pålitlig systemutvecklingsprocess. (Brooks, 1986). Trots att kunskap kring denna problematik fanns redan 1986, ökar fortfarande antalet misslyckade IT-projekt, hur kommer det sig? En förklaring kan vara att införande av IT-system är en komplicerad process där många olika faktorer påverkar hur väl utfallet motsvarar beställarens förväntningar. En annan vanlig orsak till misslyckande är att IT-projekt genomförs med en alltför kort tidsplan, och att centrala frågor inte ges den uppmärksamhet som krävs för att få önskad effekt. Utan att följa en strukturerad arbetsprocess, blir ofta slutresultatet ett projekt och avtal som kostar onödigt mycket på lång sikt. Ett exempel är dolda och stigande kostnader för underhåll och oförutsedda supportkostnader. (Kontract, 2013-05-07).

Det är vanligt att organisationer saknar en konsekvent och långsiktig IT-strategi. IT-projekt genomförs ofta under tidspress, och komplicerade och komplexa frågor behandlas inte tillräckligt ingående. De flesta företag tenderar att förhandla undermåligt och utsätter sig för större risker än de behöver i sina IT-affärer. Orsaken är ofta brist på resurser, kompetens och att företag underskattar betydelsen av att använda en genomtänkt systemutvecklingsprocess. För att minska riskerna, bör projektet genomföras enligt en på förhand definierad process och ett tydligt syfte, vilket ökar förutsättningarna för att projektet skall bli lyckat, och att organisationen uppnår de förväntade resultaten. (Kontract, 2013-05-07).

Utöver problematiken som tas upp ovan, har vi även under vår utbildning både fått höra och läsa, hur viktigt det är att leverantören stämmer av med kundens önskemål, och hela tiden arbetar mot att utveckla en produkt som motsvarar kundens vision. Det har även poängterats att kundens bild av det efterfrågade systemet, inte alltid stämmer överens med hur leverantören uppfattat det. Detta väckte således vår uppmärksamhet och skapade ett intresse för att undersöka fenomenet än närmre. Vi som studenter ville se hur de vanligt förespråkade, teoretiska kunskaperna skiljer sig från hur det ser ut i praktiken.

## 1.1 Problemområde

Att förstå vad användaren efterfrågar är en central del i design av informationssystem, och en kritisk faktor för ett lyckat resultat. Om leverantören vill ha framgång med att utveckla och implementera effektiva och bra system, bör denne starta med att skapa sig en förståelse för kundens behov och önskemål. (Bevan & Maguire, 2002).

I en rad dokumentationer (Antemar & Undall, 2011, Brorsson, 2009, Larsson, 2012) framgår det tydligt att medvetenheten kring problematiken av IT-projekt är hög. Trots denna medvetenhet, kvarstår fortfarande frågan som först skapade vårt intresse - *Vad är det som gör, att det är så svårt att genomföra ett IT-utvecklingsprojekt där kunden anses nöjd?*

## 1.2 Forskningsfråga

Efter att ha studerat utvald teori som behandlar problematiken kring IT-utvecklingsprojekt, tyckte vi det vore intressant att undersöka vilka faktorer som påverkar såväl positivt som negativt. Utefter detta gjordes valet att undersöka och jämföra de teoretiska rekommendationer som finns, med hur leverantörer arbetar i praktiken, vilket i sin tur ledde fram till följande forskningsfråga:

**Hur väl följer leverantörer de generella, teoretiska rekommendationer som finns, i sitt praktiska arbete?**

**– Om skillnader förekommer, vad beror detta i så fall på?**

## 1.3 Syfte

Vårt syfte med undersökningen är att utreda huruvida teorin, om hur leverantörer och kunder bör utföra ett IT-utvecklingsprojekt för att uppnå ett tillfredsställande resultat, stämmer överens med det praktiska arbetet.

## 1.4 Avgränsningar

Som en första avgränsning valde vi att inrikta oss på IT-utvecklingsprojekt utifrån leverantörens perspektiv. Detta då vi uppmärksammat att de undersökningar som finns gjorda idag, till största del enbart behandlar kundperspektivet. Då vi anser att det är svårt att mäta hur pass framgångsrikt ett IT-utvecklingsprojekt är, har vi valt att begränsa vår definition av huruvida ett projekt är lyckat eller ej, till uppfyllandet av kundens kvalitetskrav, grad av tillfredsställdhet, samt genomförbarheten av det önskade arbetet. Vi valde även att avgränsa vår teori till de sex utvalda teoretiska modeller som tillsammans utgör vårt teoretiska ramverk.

## 2 Teoretisk bakgrund och ramverk

I detta kapitel lyfter vi fram den teori vi valt ut för genomförandet av undersökningen, samt går in i detalj på de sex modeller som valdes ut för granskning. Till att börja med har vi valt att lyfta fram två svenska IT-projekt som har misslyckats. Dessa är exempel värda att begrundas och dra lärdom av.

### 2.1 PRV: 300 miljoner – för ingenting

I slutet av 90-talet beslutade Patent- och Registreringsverket (PRV) att införa ett nytt IT-system för kundservice, ekonomi och administration. Projektet startade under våren 1997 och förväntades vara klart i maj 1999. Projektbudgeten fastställdes till över 175 miljoner kronor, men resultatet blev verkligen inte det förväntade. Trots att det genomfördes en förstudie och en detaljerad upphandling, föll projektet på ett allt för ospecifikt och bristande avtal mellan PRV och den valda leverantören. Leverantören undvek jobbiga beslut om den övergripande designen, och fokuserade istället på de tekniska aspekterna. Detta resulterade i allt för många komplicerade detaljer, vilket medförde att införande av systemet misslyckades. Leverantörerna tvingades meddela att projektet förväntades överskrida budgeten med över 10 miljoner kronor, och bli flera månader försenat. Projektledningen valde trots detta besked att fortsätta arbetet. I maj 1999 visade det sig, att det överhuvudtaget inte fanns något system att lansera. Parterna tvingades då förhandla om avtalet, och ställa mer konkreta systemkrav. I juni 2000 levererades slutligen systemet till PRV. Inledningsvis verkade allt bra, men efter en tid upptäcktes det brister i funktionaliteten, och systemet var i stort sett omöjligt att använda. PRV tvingades höja avgifterna på bolagsavdelningen för att få ekonomin att gå ihop, eftersom projektets slutnota var 71 procent över budget. PRV har ännu idag inte lyckas införa ett fungerande IT-system som kan stödja de administrativa processerna. Hur kunde det gå så här fel?

Enligt Torsten Cegrell, professor vid Institutionen för Industriella Informations och Styrssystem på KTH, är PRV-fallet ett typexempel på otydliga affärsmål och dålig upphandling. Även bristande beställarkompetens tas upp som en orsak. Torsten Cegrell menar att många företags beställarkompetens är obefintlig. Han menar att misslyckandet inte hade behövt ske, om en koppling skett mellan kravspecifikationen och slutanvändarnas verkliga behov. Detta hade underlättat leverantörernas arbete väsentligt. (Würtemberg, 2010).

### 2.2 Landstinget: En dyr säkerhetsrisk

Ett annat projekt, som uppmärksammats mycket i media, var när ett flertal landsting skulle införa journalsystemet Cambio Cosmic. Tanken var god, ett gemensamt journalsystem för alla Sveriges sjukhus skulle leda till högre patientsäkerhet och mindre administrativt dubbelarbete. Tyvärr gick det inte som planerat. Systemet försenades, utsatte patienter för risk och blev alldeles för dyrt. En



analys visade att misslyckandet till stor del berodde på att det saknades en gemensam budget för implementeringen. Varje enskild klinik fick själva täcka kostnader för inköp av datorer och personalens utbildning. I Östergötlands landsting, var budgeten på 130 miljoner tänkt att täcka kostnaderna för licenser, servrar och utbildning. Resultatet visade att i enbart Östergötlands landsting, blev de totala utvecklingskostnaderna 250 miljoner - nästan dubbelt så mycket som beräknat.

Ett annat problem var att leverantören inte hade kompetens nog att leverera de funktioner som utlovats enligt fastställd tidsplan. Exempelvis saknade systemet en varningsfunktion när personalen matade in felaktiga uppgifter, vilket påverkade patientsäkerheten negativt. I efterhand har det även framkommit att misslyckandet till stor del berodde på bristande kommunikation med slutanvändaren, och att det saknades förståelse för de arbetsuppgifter som systemet skulle stödja. Detta ledde till att arbetsbelastningen för personalen ökade, istället för att minska. En annan svaghet i detta projekt var oförmågan att hantera de brister som identifierats i förstudien. Bristerna fanns därför kvar när systemet rullades ut. Torsten Cegrell menar att en vanlig orsak till att IT-projekt misslyckas är att organisationer helt enkelt väljer ett system som är onödigt avancerat. Ofta tror företag att de mer tekniska och komplexa systemen, är det bästa för deras behov. Konsekvenserna blir ofta att användarna tvingas lägga ner alltför mycket tid på administration, för att lära sig att hantera systemet. Även här sägs tydlig kommunikation med slutanvändarna av systemet vara avgörande för utfallet. (Würtemberg, 2010).

## 2.3 Teoretiska systemutvecklingsmodeller

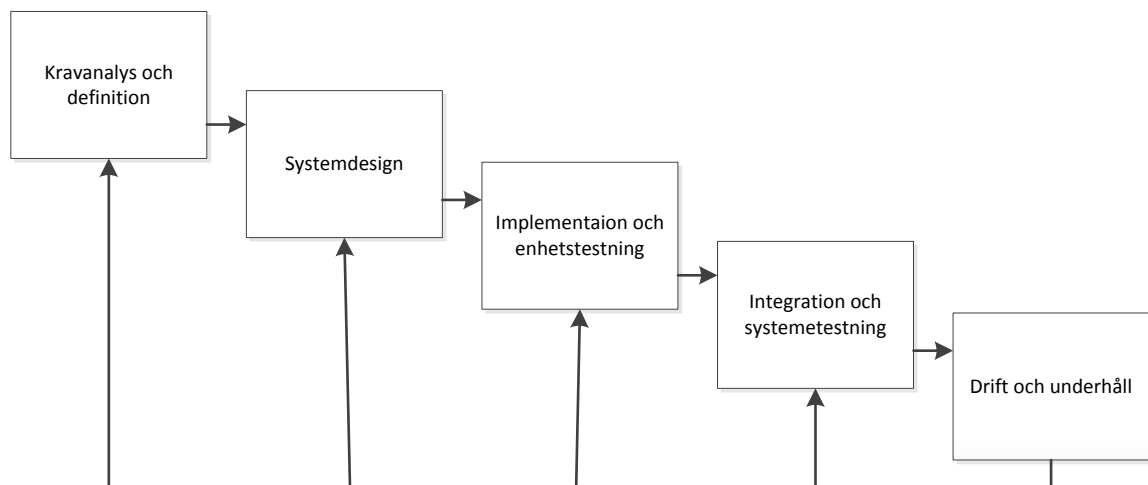
I detta avsnitt har vi sammanställt sex systemutvecklingsmodeller som vi ansåg var relevanta för vår undersökning. Modellerna valdes ut med grundtanken att hitta sex välkända modeller inom ämnesområdet med snarlikt innehåll och tankesätt. Med välkända menar vi att de är framtagna av etablerade författare inom området och därmed enligt oss användbara, och trovärdiga källor. Modellerna som vi valt att använda i vår studie kan självklart inte representera all teoretisk bakgrund inom området, men kan enligt oss ge en övergripande bild över vad de teoretiska rekommendationerna förespråkar.

Vi är medvetna om att det finns en större mängd modeller och därmed även de som skulle kunna vara bättre, dock var vi tvungna att begränsa oss. Vilket slutligen ledde till våra nuvarande utvalda modeller som senare ligger till grund för vår sammanställda teoretiska modell, som presenteras i slutet av kapitlet.

## 2.3.1 Sommervilles vattenfallsmodell

Denna modell har sitt ursprung från år 1970. Enligt Sommerville (2011) är modellen ”plandrivnen”, vilket syftar till att samtliga processer i utvecklingen måste planeras på förhand, innan själva utvecklingen kan börja. Stegen i vattenfallsmodellen är:

- **Kravanalys och definition:** Här fastställs systemets mål, restriktioner och funktioner. Dessa definieras sedan mer detaljerat, och bildar tillsammans en systemspecifikation.
- **Systemdesign:** I detta steg designas systemets arkitektur. Detta görs bl.a. genom att definiera systemets avskiljningar och deras relation till varandra. Denna information ligger sedan till grund för vilken hård- och mjukvara som krävs för att använda systemet.
- **Implementation och enhetstesting:** Designen tar form och olika komponenter testas. Därefter verifieras komponenternas användbarhet, och säkerställs så att de möter systemets specifikationskrav.
- **Integration och systemtestning:** Efter att komponenterna säkerställts så att de uppfyller de krav som fastställts, integreras slutligen systemet och testas i sin helhet. Därefter levereras systemet till kunden.
- **Drift och underhåll:** Denna fas innefattar installation och underhåll av systemet. Här behandlas de eventuella fel som tidigare inte identifierats, och nya krav utvecklas för systemet som sedermera implementeras allt eftersom.



*Sommervilles vattenfallsmodell 2.1*

Rent teoretiskt är varje enskild fas avskild från de andra, det vill säga att en fas tar vid där föregående fas slutar. Detta fungerar dock inte i praktiken. Kommunikation och samspel mellan

de olika faserna sker iterativt, och det växlas gång på gång fram och tillbaka mellan faserna. Exempelvis kan ett designfel vara dolt fram till dess att kodningen av systemet påbörjas, eller då ett fel i koden inte upptäcks förrän systemet implementeras. Ju fler iterationer som sker, ju mer kostsamt blir systemet, då valideringar i form av dokumentation måste göras om ett flertal gånger. (Sommerville, 2011, s.30).

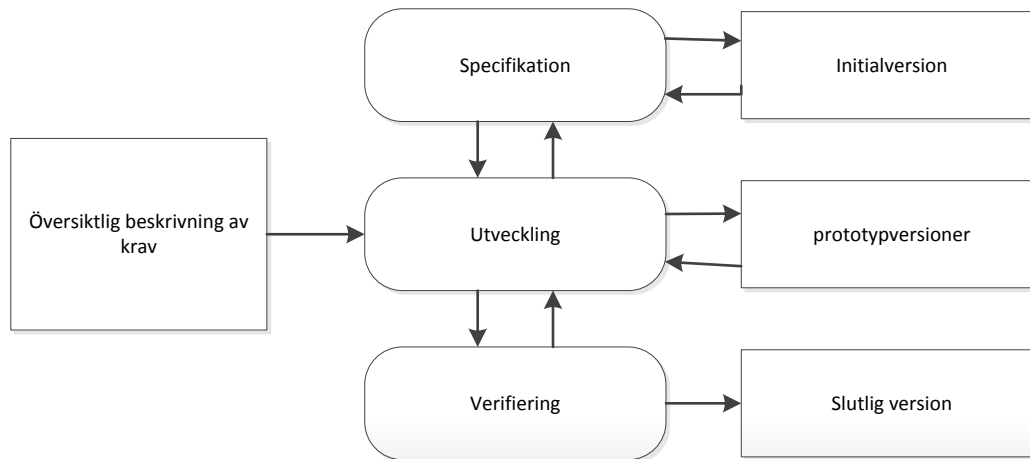
Sommerville (2011, s.30) menar på att vattenfallsmodellen oftast innefattar ett stort problem, då den inte är bra på att hantera förändringar i struktur, utan bygger på att varje fas som avslutas är rätt och inte behöver förändras i framtiden. Denna nackdel leder tyvärr till att vattenfallsmodellen vanligtvis medför högre kostnader, jämfört med de andra modellerna. (Sommerville, 2011, s.30).

### 2.3.2 Sommervilles prototypintensiva utvecklingsmodell

Modellen beskriver att IT-utvecklingsarbetet inleds med att skapa en ”Initial Version”, som enbart innehåller de grundläggande funktioner som krävs. Denna version används sedan för att erhålla feedback som kan användas för att skapa en kravspecifikation, vilket senare leder till att systemet utvecklas och slutligen verifieras. Sommerville (2011) menar att arbetet inleds med en övergripande beskrivning över hur systemet ska fungera. Därefter tillkommer tre huvudsteg som inbördes arbetar med varandra genom att skapa prototyper av systemet. Den första versionen har sitt ursprung från kravspecifikationen som hela tiden förändras, baserat på feedback från prototypsystemet. När specifikationen sedan anses tillfredställande, övergår arbetet till en mer utvecklingsfokuserad fas, där ett flertal prototypversioner iterativt agerar med varandra. När utvecklingsfasen är slutförd och systemet anses vara kapabelt att fungera så som det är avsett att göra, görs en validering innan en slutlig version av systemet skapas.

Jämfört med den tidigare presenterade vattenfallsmodellen, har den prototypintensiva utvecklingsmodellen i huvudsak tre stora fördelar:

1. Eftersom betydligt färre valideringsprocesser genomförs, samt att analysdokumentationer inte behöver presenteras ett flertal gånger, sparas en hel del resurser.
2. Då kunder kan interagera med systemet redan i prototypfasen, är det lättare att se fel som kanske inte är så lätta att upptäcka i dokumentationsform. Att lösa dessa fel innan systemet övergår till en färdig fas, kan med största sannolikhet resultera i en minskad kostnad.
3. Eftersom kunden kan använda sig av systemet trots att det inte är helt färdigt, sker implementationen enklare, då systemet börjar integreras redan vid prototypfasen.



*Sommervilles prototypintensiva utvecklingsmodell 2.2*

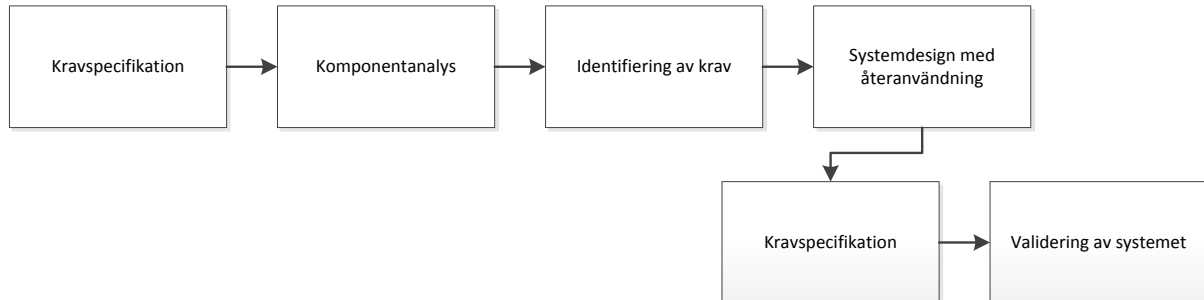
Sommerville (2011) nämner att det dock finns nackdelar vid den här typen av implementation. Då utvecklingsprocessen för systemet inte är tydlig i form av dokumentation, kan det bli svårt att mäta hur stor andel som är klar, samt hur mycket som återstår. Det blir helt enkelt för kostsamt att dokumentera varje version av prototypen. En annan nackdel är att systemets struktur blir oklar, ju fler funktioner som läggs till för varje ny prototyp. (Sommerville, 2011, s.32).

### 2.3.3 Sommervilles återanvändningsmodell för mjukvaruutveckling

Denna modell skiljer sig markant från de övriga som tidigare beskrivits. Hela idén bygger på att återanvända komponenter från tidigare system, förändra dessa och slutligen sammanställa allt till ett enda system. Det som gör att denna modell avviker från de övriga är de mellanliggande stegen, mellan kravspecifikationen och verifieringen av det färdiga systemet, som beskrivs mer ingående nedan:

- **Komponentanalys:** Arbetet börjar med att analysera kravspecifikationen, för att sedan börja leta efter komponenter som kan återanvändas efter modifiering. De flesta komponenter kommer dock inte ha exakt de funktioner som efterfrågas. Ett öppet och kreativt tänkande är därför av vikt.
- **Identifiering av krav:** I inledningsfasen identifieras de specifika komponenter som behövs, för att matcha systemets krav. Om leverantören stöter på komponenter som inte kan användas till systemets syfte, måste arbetet återgå till att analysera komponenter för att på så sätt leta efter avsaknad komponent.
- **Systemdesign med återanvändning:** Här designas ramverket, som baseras på de komponenter som är aktuella. Därefter görs en utvärdering, om det krävs ytterligare mjukvara för att fullända systemet, som då senare läggs till i designen.

- **Utveckling och integration:** Mjukvara som inte kan återanvändas måste utvecklas. De komponenter som tidigare samlats in och modifierats, integreras i detta steg, och bildar ett slutligt system som är redo att användas.



*Sommervilles återanvändningsmodell för mjukvaruutveckling 2.3*

Det självklara med den här modellen är att kostnaden för systemet drastiskt minskar på grund av återanvändandet av färdiga komponenter, vilket dessutom sparar enormt mycket tid. Nackdelen är att systemets alla krav inte alltid kan mötas, då det kan saknas komponenter till dessa funktioner. (Sommerville, 2011, s.35).

### 2.3.4 Heckmans modell för IT-utvecklingsprojekt

Heckman (1999) presenterar en modell för genomförande av IT-projekt, utvecklad av forskaren Robert Heckman. Modellen ger en övergripande beskrivning av hela processen för IT-implementeringar, och fungerar som ett verktyg för att kartlägga de viktigaste aktiviteterna, samt frågor som ett IT-projekt kan bestå av. Ramverket är även ett verktyg för att kunna analysera projektet på ett systematiskt sätt. (Heckman, 1999).

Modellen består av fem huvudprocesser:

- **Utvecklingsprocess:** Här genomförs de aktiviteter, som är nödvändiga varje gång det finns behov för utveckling av ett nytt IT-system eller produkt. Varje enskild utvecklingsprocess kan beskrivas i form av en livscykel, som börjar med fastställande av ett krav, de processer som är inkluderade i det faktiska inköpet, och avslutas med uppfyllande av de villkor som angivits i avtalet för processen.
- **Kravspecifikation:** Processen för att fastställa och kartlägga verksamhetens motiv, krav, specifikationer samt godkännande av projektet. Denna process består även subprocesser, som att sammanställa projektgrupper, genomföra riskbedömning och få ledningens godkännande, så att projektet kan genomföras.
- **Förvärv:** Är processen för utvärdering av olika leverantörer och genomförandet av projektet, samt av de produkter och tjänster som är nödvändiga. Denna del inkluderar även utvärdering av olika leverantörers förslag och avslutas med kontraktsförhandlingar

när företaget valt den leverantör som anses vara mest lämplig.

- **Fastställande av kontrakt:** I denna del av projektet genomförs de aktiviteter som är nödvändiga för att kontraktet ska kunna fastställas. Detta innefattar bland annat godkännande av nödvändiga produkter och tjänster, installation av systemet, kontraktsadministration samt garanti och underhållsarbete.
- **Styrningsprocessen:** Inkluderar de aktiviteter som är involverade i den övergripande styrningen av IT-processen. Dessa aktiviteter är inte specifika för någon särskild del i utvecklingsprocessen, utan kan snarare gälla generellt för alla dessa händelser.

Till styrningsprocessen finns det tre allmänna subprocesser för styrning av IT-utvecklingsprojekt, dessa är:

- **Leverantörshantering:** Inkluderar de aktiviteter som krävs för att optimera relationen mellan kund och leverantör, och därmed skapa värde för företaget. Denna fas innefattar även utvecklandet av relationer med nyckelleverantörer, att påverka deras prestationer, samt hantera kommunikationen med dem.
- **Kvalitetshantering:** Genomförs för att säkerställa kontinuerliga förbättringar i utvecklingsprocessen, och för samtliga produkter och tjänster som är nödvändiga. Detta görs genom produkttester och olika kvalitetskontroller med leverantören.
- **Kapitalförvaltning:** Genomförs för att optimera användandet av samtliga IT-resurser, för att kunna möta verksamhetens behov. Det inkluderar aktiviteter så som utvecklande av strategier och regler för kapitalförvaltning, utveckling, och underhåll av kapitalförvaltningssystem.



#### *Heckmans modell för IT-utvecklingsprojekt 2.4*

Heckman (1999) föreslår även fem åtgärds punkter för framtida insatser, som ska avhandlas mellan chefer för IT-projekt och akademiska forskare för att ytterligare effektivisera hanterandet av IT- utvecklingsprocesser. Dessa åtgärds punkter är:

1. Utveckla mätmetoder för utvecklingsprocessen och använda dessa för att kunna jämföra olika IT-projekt.
2. Fastställandet av olika roller i utvecklingsprocessen för att interna och externa relationer mellan involverade aktörer ska kunna utvecklas.

3. Använda ett slags ramverk som hjälpmedel för att utveckla IT-utvecklingsprocessen.
4. Använda ramverket som en guide för framtida forskning.
5. Använda ramverket för att strukturera träning och utbildning för IT-projekt.

(Heckman, 1999, s 61-67).

### 2.3.5 Bevan & Maguires modell för IT-utvecklingsprojekt

Bevan & Maguire (2002) menar att det är viktigt att förstå användares krav och att detta är en central del inom design och utveckling av informationssystem. Detta är helt avgörande för att IT-utvecklingsprojektet, systemet i sig, och dess användbarhet ska bli så bra som möjligt. Det är idag allmänt känt att det krävs en förståelse av användarnas behov och krav, för att kunna utveckla effektiva system. Utvecklingsarbetet bör därför alltid inledas, med att kartlägga och skapa en förståelse för de behov och krav, som finns på systemet. Detta kommer troligtvis att resultera i en ökad produktivitet, minskade kostnader för projektet, samt tillfredsställda slutanvändare. Dock är detta arbete inte helt enkelt.

Bevan & Maguire (2002) nämner några vanliga problem som kan uppstå under analysarbetet:

- Att lyckas hantera komplexa organisationer med många olika intressenter.
- Att användarna, på förhand ofta inte själva vet vad de egentligen efterfrågar och vill uppnå med systemet.
- Allt kortare utvecklingscykler, vilket minskar den tid som läggs på att analysera de behov och krav som är aktuella.

Studien diskuterar hur dessa problem kan lösas genom att använda ett ramverk för att effektivisera och stödja utvecklingsprocessen, samt analysera användarnas krav.

Modellen består av fyra faser med tillhörande stödfunktioner:

#### Steg 1: **Informationsinsamling**

Första steget i modellen går ut på att samla in bakgrundsinformation om användarna och analysera de arbetsprocesser som utförs i dagsläget. Följande metoder kan användas för att göra detta:

- **Intressentanalys:** Går ut på att identifiera samtliga användare och intressenter som kommer att påverkas av det nya systemet. Intressenter i detta fall är personal från olika delar av organisationen som till exempel: marknadsföring, inköp och support. Syftet med intressentanalysen är att säkerställa så att samtliga inblandade parter behov tas i åtanke.
- **Sekundär marknadsundersökning:** Detta steg innefattar studerandet av publicerade forskningsrapporter och demografisk information vilket kan ge en bild av den potentiella användarmarknaden.

- **Kartlägga användningsområdet:** Genomförs innan ett system ska utvecklas. Genom att skapa en förståelse för det sammanhang systemet kommer att användas i, är det lättare att uppnå högre kvalitet, användbarhet och tillgänglighet för systemet.
- **Analys av arbetsuppgifter:** Här studerar utvecklarna, användares nuvarande arbetsuppgifter och vilka åtgärder och funktioner som måste finnas som stöd.
- **Rika bilder:** Kan bidra till att användarna får förståelse för olika problem, och därmed har lättare att identifiera mer underliggande behov med systemet. Denna teknik går ut på att utforma olika skisser som illustrerar hur användarna och systemet förhåller sig till varandra.
- **Fältstudier:** Innebär att utvecklarna observerar användarna i deras arbetar och dokumenterar olika händelser.
- **Föra dagbok:** Denna går ut på att dokumentera användarnas beteende under en längre tidsperiod. Detta leder ofta till att det blir lättare att identifiera de funktioner som systemet ska innehålla.

(Bevan & Maguire, 2002).

## Steg 2: Identifiera användarnas behov

När informationsinsamlingen är genomförd kan utvecklingsteamet inleda arbetet med att kartlägga användarnas behov. Följande metoder kan användas för att göra detta:

- **Enkätundersökningar:** Innebär att användarna svarar på en uppsättning frågor. Svaren kan bidra till att deras behov kan identifieras, men även fånga deras uppfattning och idéer av det nya systemet.
- **Fokusgrupper:** Här bildas diskussionsgrupper med användare från olika delar av organisationen. Dessa tillfällen leder ofta till att både behov och eventuella problem kan uppmärksammas.
- **Intervjuer:** Är en vanligt förekommande teknik där användare, intressenter och experter blir utfrågade, för att på så sätt kartlägga deras behov och krav.
- **Användarfall:** Här genomförs realistiska exempel på hur användare kan tänkas utföra sina arbetsuppgifter i ett specifikt sammanhang. Användarfallen hjälper även till att identifiera mål kring systemets användbarhet



- **Utvärdering av nuvarande och konkurrenters system:** Denna metod kan ge värdefull information om i vilken utsträckning dagens system lever upp till användarnas behov, och kan även hjälpa till att identifiera potentiella användbarhetsproblem som bör undvikas. Även funktioner som upptäcks i konkurrenters system, kan bidra till att vissa problem undviks. (Bevan & Maguire, 2002).

### Steg 3: **Föreställning och utvärdering**

När en första uppsättning av användarkrav har fastställts, är det viktigt att utveckla en prototyp för att illustrera dem. Prototypen bidrar med att användarfeedback kan erhållas, för att definiera användarkraven ytterligare. Potentiella metoder för detta är:

- **Brainstorming:** Sammanföra ett antal design- och arbetsexperter, så att de kan inspirera varandra i den kreativa fasen av problemlösningsprocessen.
- **Blankettutvärdering:** Är en teknik för att avslöja den hierarkiska strukturen i ett projekt, genom att be användare gradera ämnen, nedskrivna på blanketter.
- **Prototyper:** Här skapar designers mjukvarubaserade simuleringar av element i användargränssnittet på ett statiskt eller dynamiskt sätt.
- **Allokering av funktioner:** En etablering av en rad alternativ för att kunna identifiera den optimala arbetsfördelningen. Detta bidrar i sin tur till ökad trivsel och effektivitet i hela arbetsprocessen.
- **Designriktlinjer:** Refereras till designers och HCI -specialister för vägledning inom ergonomiska frågor associerade med det system som ska utvecklas. (Bevan & Maguire, 2002).

### Steg 4: **Kravspecifikation**

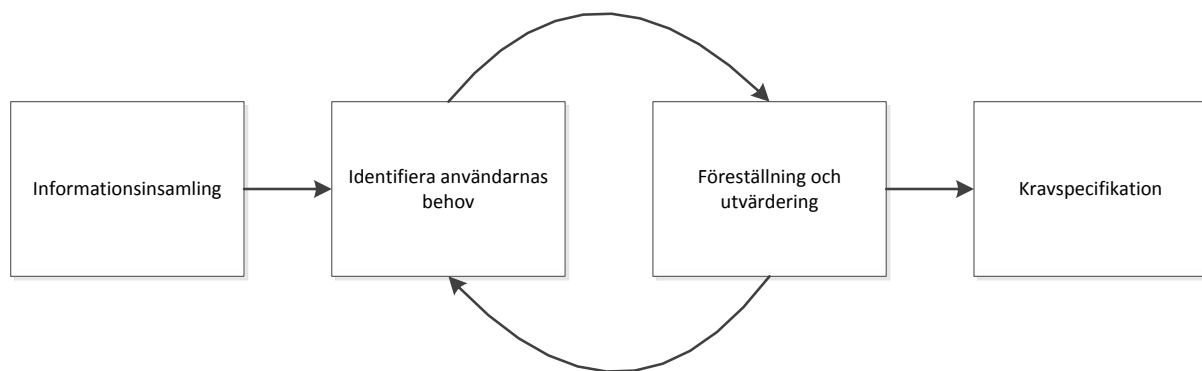
Bevan & Maguire (2002) rekommenderar att följande aspekter dokumenteras för kravspecifikationen: identifiering av berörda användare och andra intressenter, fastställa ett tydligt designmål, mätbara riktmärken så att designen kan testas under utvecklingsfasen, få berörda intressenters godkännande av lagstadgade eller rättsliga krav, till exempel inom hälsa och säkerhet. Det är även viktigt att kunna hantera förändringar av olika krav under utvecklingsprocessen.

Följande tekniker och metoder kan användas för att stödja sammanställningen av en organisation och anställdas kravspecifikation.

- **Identifiera berörda användare och intressenter:** Inkluderar en tydlig redogörelse av designmål, för att få intressenters godkännande, och att rättsliga krav följs, samt att hälsa

och säkerhet uppnås.

- **Kartläggning av arbetsuppgifter och funktioner:** Här specificeras systemets funktioner, som varje enskild användare behöver för att kunna utföra sina arbetsuppgifter på ett effektivt sätt.
- **Kategorisering av krav:** Innebär bland annat dokumentation av användarkrav, vilket är viktigt för att underlätta utformningen av själva systemet. Exempel på användarkrav kan vara kortfattade beskrivningar av de uppgifter som systemet måste stödja, samt de funktioner som ska finnas tillgängliga för att dessa ska kunna utföras.
- **Användbarhetskrav:** Detaljerade beskrivningar av de krav som finns för användbarheten är viktigt att ge, så att designteamet lättare kan göra prioriteringar.
- **Organisatoriska krav:** En förståelse för de organisatoriska kraven kan bidra till att skapa ett system som stödjer både organisationsstrukturen och den interna kommunikationen.
- **Prioritering:** Exempelvis att göra prioriteringar av användarkraven är viktigt, så att de resurser som tilldelas för utvecklingsarbetet kan allokeras så effektivt som möjligt.



#### *Bevan & Maguires modell för IT-utvecklingsprojekt 2.5*

För att säkerställa att resultatet av ovanstående analysarbete blir lyckat, måste designteamet tillfredsställa de behov och önskemål som användarna har när utvecklingsarbetet är genomfört. För att lyckas menar Bevan & Maguire att användarkraven inte enbart karläggas genom tekniker som enkäter, fokusgrupper, intervjuer med mera, utan bör även identifieras via simuleringar, för att sammanställa en prototyp av användarkraven. Kraven kommer då att formas i takt med att systemet utvecklas och en formell utvärdering genomförs. (Bevan & Maguire, 2002).

### 2.3.6 Verville & Halington modell för IT-utvecklingsprojekt

Modellen ger en övergripande bild över hur en IT- investering kan gå till. Enligt Verville & Halington (2003) kan utvecklingsprocessen av ett system delas upp i fem olika steg:

- **Planering:** När ett företag är i behov av att införskaffa ett nytt system startar en planeringsprocess. Planeringsarbetet är iterativt, och återkommer konsekvent genom hela IT-projektet. Målet med planeringsfasen är att samla in den information som är nödvändig, för att kunna fatta välgrundade beslut. Planeringsfasen syftar även till att utreda vilka behov som finns, för att sedan tillsammans med leverantören finna en passande lösning. Arbetet innebär att en analys av nuläget görs, som sedan utvecklas till en målbild att utgå ifrån. I planeringsfasen är det viktigt att företaget och leverantören tillsammans gör en bedömning om kommande förändringsbehov, och undviker att låsa sig vid en enda specifik lösning. Det gäller att ej ta för snabba eller förhastade beslut.
- **Informationsökning:** Här fastställs det faktiska behovet, och specifikationer kring detta förtydligas. Genom att bestämma kraven, får leverantören en bättre bild över vilken information som måste samlas in beträffande existerande lösningar, och det system som ska utvecklas.
- **Förhandling:** Här sammanställer företaget en offertförfrågan till leverantören, där saker som betalning, tidram och eventuellt efterarbete tas upp. Denna offertförfrågan brukar kallas för Request for Proposal (RFP). Målet med förhandlingsprocessen är att parterna ska godkänna offerten och gemensamt skriva under ett avtal.
- **Val:** Denna fas är ett direkt resultat av både informationsöknings- och förhandlingsfasen. Genom att ha samlat in tillräcklig information beträffande olika alternativ, och kommit överens om ett avtal, kan företaget och leverantören tillsammans gå vidare och ta beslut för vilken lösning som är mest optimal. Det är viktigt att båda parter har en gemensam bild över hur systemet ska se ut, och vilka funktioner det ska innehålla. Detta är av absolut vikt för att implementeringen ska lyckas.
- **Utvärdering:** Innebär att företaget gör en utvärdering av leverantörens arbete, men även av resultatet utifrån själva produkten/lösningen som tillhandahållits. Med andra ord en utvärdering för om systemet levde upp till den behovsbild som utarbetades i planeringsfasen. Även aspekter som tidsram, pris och implementeringen av systemet granskas.



*Verville & Haltingen modell för IT-utvecklingsprojekt 2.6*

Verville & Haltingen (2003) tar även upp betydelsen av leverantörens storlek och finansiella status. Detta är viktigt för företaget att ha i åtanke, eftersom en investering av ett nytt system medför en inlåsningsseffekt och en beroendeställning till den valda leverantören. Att välja en känd och väletablerad leverantör, medför en större trygghet för företaget. Det är naturligtvis mycket viktigt att leverantören kommer finnas kvar i framtiden, och kan leverera sina tjänster till företaget. (Verville & Hallingten, 2003).

## 2.4 Matris över återkommande faktorer

Efter att ha studerat ovanstående modeller mer djupgående, har vi här nedan listat upp de faktorer som nämns, i en mer lättöverskådlig matris. Syftet med matrisen är att vi lätt ska kunna lyfta fram och plocka ut de faktorer som är mest återkommande, och med det menar vi de faktorer som har minst tre eller fler kryss, för att sedan kunna använda dessa som en stabilare grund i vår undersökning.

	Sommervilles Vattenfallsmodell	Sommervilles prototyp-intensiva utvecklingsmodell	Sommervilles återanvändningsmodell för mjukvaruutveckling	Vervilles modell för IT-utvecklingsprojekt	Heckmans modell för IT-utvecklingsprojekt	Bevan & Maguires modell för IT-utvecklingsprojekt
Föranalys	X		X	X		X
Kravspecifikation	X	X	X	X	X	X
Tid				X		
Kostnad			X	X		
Kvalitet	X	X		X		
Återkoppling	X					
Prototypbaserad systemutveckling	X	X	X		X	X
Systemspecifikation						X
Design	X		X			

Validering/ verifiering		X		X		X
Iteration		X	X			X
Imple- mentering	X					

Matris över mest återkommande faktorer 2.7

### 2.4.1 De mest återkommande faktorerna

Nedan beskrivs de faktorer som vi i vår matris kunde se nämndes i minst tre eller fler modeller, vilket enligt oss ansågs vara tillräckligt för att en faktor skulle kunna definieras som mest återkommande.

#### **Föranalys/Informationsinsamling**

Informationsinsamling är enligt Bevan & Maguire (2002) den processen som innefattar urval av den bakgrundsinformation, som finns tillgänglig om användarna av det nya systemet. I artikeln *User requirements analysis: A review of supporting methods* (Bevan & Maguire, 2002) ingår bland annat analyser av intressenter, marknadsundersökningar, fältstudier med mera. Detta är första steget som genomförs innan en analys, vilket ger en bra grund för vidare arbete.

En föranalys kan innefatta flera olika definitioner. En del definierar system eller projektanalyser i form av kostnadsfördelsperspektiv (Layard, R & Glaister, S. 1994) medan andra definierar det som ett steg där mål, restriktioner och tjänster definieras (Sommerville, 2011). I andra artiklar kan analysdelen syfta på att identifiera användarnas behov (Bevan & Maguire, 2002).

Analysdelen i vårt sammanhang kommer, i den här uppsatsen, att innefatta en kombination av de två sistnämnda definitionerna, med andra ord förmågan att använda den information som finns tillgänglig om användningsområdet och för att strukturera upp problemet, för att sedan tillsätta de funktioner som systemet kräver. (Sommerville, 2011, s.30).

#### **Kravspecifikation**

Första steget vid fastställandet av systemkrav är att identifiera kundens behov. Detta kan verka självklart, men att komma fram till rätt behov och krav är ofta väldigt komplicerat. Vid fastställande av krav är det viktigt att systemets funktioner, attribut och begränsningar analyseras, så att både kund och leverantör uppnår en gemensam bild över vad systemet slutligen ska innehålla. (Lehmann & Lattemann, 1997). Detta kommer troligtvis att resultera i en ökad produktivitet, minskade kostnader för projektet och tillfredställda slutanvändare. Dock är detta arbete inte helt enkelt. (Olphert & Damodaran, 2002).

Lehmann Lattemann (1997) beskriver olika metoder för att lyckas identifiera kundens behov så korrekt som möjligt. Först steget är att kunden och leverantören skapar sig en gemensam förståelse över det system som ska utvecklas. Det är meningslöst att försöka fastställa exakta behov och specifikationer, om leverantören och kunden har olika uppfattningar om vad som krävs. Ett vanligt problem är att kunden och leverantören har olika uppfattningar om vad systemet ska innehålla. Det är därför viktigt med en god kommunikation mellan båda parter.

En annan viktig aspekt är att leverantören bör försöka kartlägga hur systemet kommer att användas, och i vilken miljö. Leverantören bör även försöka involvera rätt personer när kravspecifikationen fastställs. Det är bortkastad tid att diskutera krav med olämpliga personer och intressenter, som saknar insyn i de arbetsprocesser som finns i företaget.

Heckman (1999) nämner även att såväl utvecklare som kund bör genomföra en riskbedömning innan de bestämmer sig för det nya systemet. Det rekommenderas att riskbedömningen utförs innan utvecklingsprocessen startar för att den ska fungera som ett bra beslutsunderlag.

### **Kvalitet (Scope)**

Det finns enligt Chatfield & Johnson (2007) två typer av kvalitetsperspektiv. Den ena är *Product Scope*, som syftar på att projektets slutprodukt innefattar de kvalitets- och funktionskrav som fastställts. Den andra är *Project Scope* som hanterar de processer eller metoder som krävs för att uppnå målet, som tidigare fastställts i *Product Scope*. Ett projekt med stort fokus på kvalitet, skulle exempelvis kunna innefatta utveckling av en specifik del i en motor. Denna del måste då klara av en viss temperatur och ha exakta mått, så att den kan passa i motorn. Atkins (1999) förklarar att till skillnad från tids- och kostnadsperspektivet, så kan inte kvaliteten alltid mätas. Därmed kan det ibland vara svårt att utvärdera om ett kvalitetsfokuserat projekt blev lyckat eller misslyckat. Kvalitet förhåller sig till tid och kostnad (Chatfield & Johnson, 2007).

### **Prototypbaserad systemutveckling**

I de flesta modeller ser vi att den prototypbaserade systemutvecklingsfasen är ett steg som befinner sig precis efter att designen av systemet är gjord (Sommerville, 2011, s30, s.35, Bevan & Maguire, 2002). Sommerville (2011) beskriver i sin beskrivning av den prototypintensiva utvecklingsmodellen (Sommerville, 2011, s. 32) hur pass viktigt det är att hålla en nära och iterativ relation mellan design och utvecklingsfas. I sin version av *Vattenfallsmodellen* ser vi hur designsteget följs upp av utveckling, och indikerar samtidigt på att inte påbörja utveckling innan det skapats en tydlig design av systemets utformning.

När ett första utkast av kundens behov och krav har kartlagts, är det viktigt att utveckla olika prototyper av systemet för att kunna illustrera dessa. Prototyper medför en högre grad av realism än vad som är möjligt att uppnå med exempelvis skisser och en muntlig dialog. Genom att använda sig av prototyper i ett tidigt skede, kan användarnas/kundernas åsikter om systemet erhållas, vilket är viktigt och en förutsättning för att det slutgiltiga systemet ska matcha kundens behov. Prototyper underlättar även arbetet, då de gör det lättare att identifiera eventuella fel och

buggar i ett tidigt skede. På så vis undviks det att fel utvecklas och leder till större problem, som kan vara svåra att hantera när det slutgiltiga systemet satts i drift. (Bevan & Maguire, 2002).

### **Validering och Verifiering**

Systemvalidering är processen för utvecklandet av ett dokumenterat bevis, för att det färdiga systemet kommer att fungera i den tänka driftmiljön. Det är viktigt att det används tydliga standarder för verifiering och validering, så att testspecifikationerna kan kopplas till systemets kravspecifikation. Syftet med systemvalidering är att kontrollera att det slutliga systemet fungerar och kan utföra de förväntade funktionerna, samt att det uppfyller kraven för säkerhet, revision och tillförlitlighet. Ett vanligt problem i valideringsprocessen är att säkerställa objektivitet hos de personer som utför valideringen. En lösning på detta problem kan vara att låta valideringen utföras av ett utomstående team, som är inte är involverade i själva utvecklingen av systemet. (Pstestware, hämtad 2013-04-15).

### **Iteration**

Skär mestadels i planeringsfasen då den ofta upprepas och återkommer konsekvent genom hela utvecklingsprocessen (Magnusson & Olsson, 2010). Sommerville (2011) förklarar vikten av ett iterativt förhållande, mellan olika steg i utvecklingsprocesser i sin prototypintensiva utvecklingsmodell, där systembeskrivning och prototypstestning sker ett flertal gånger, tills dess att parterna känner sig helt nöjda och slutligen godkänner systemet.

## **2.4.2 Icke avsevärt återkommande faktorer**

Nedan beskriver vi de faktorer som enbart hade ett eller två kryss i vår matris och som därmed enligt oss, inte anses vara tillräckligt frekventa för att kunna definieras som särskilt återkommande.

### **Tid**

Chatfield & Johnson (2007) nämner hur vi ända sen barnsben lärt oss att förstå tidsperspektiv, då vi använder oss utav klockor, scheman och andra verktyg för att hantera tid. De nämner dessutom att det är vanligt förekommande i en projektgrupp, att endast en del av deltagarna kan budget eller mål i detalj, medan tiden är något som alla kan i exakta mått. Ett exempel på ett tidsbegränsat projekt kan vara en takläggning på ett hus innan regnsäsongen startar, där är sannolikheten stor att tid prioriteras före de andra perspektiven. Tid förhåller sig till kvalitet och kostnad (Chatfield & Johnson, 2007).

### **Kostnad**

Chatfield & Johnson (2007) beskriver kostnad som en aspekt som sträcker sig längre än endast budget. Kostnad står enligt dem för alla resurser som krävs i projektet, för att kunna klara av att leverera projektresultatet. Detta kan inkludera bl.a. kompetens, verktyg, material med mera. Ett väldigt tydligt exempel på projekt styrda av kostnad, är de som från början har en stram budget, som under inga omständigheter får överskridas. I ett sådant scenario ställer sig kostnad över alla



andra aspekter, vilket påverkar såväl kvalitet som tidsmarginal negativt. Kostnad förhåller sig till kvalitet och tid.(Chatfield & Johnson, 2007).

### **Återkoppling**

På senare tid har det gjorts allt fler studier av arbetet kring utvärdering, efter det att ett affärssystem eller intranät har implementerats. Det har exempelvis gjorts undersökningar på hur företagen bör agera, efter att systemet har börjat användas, för att få maximal nytta av systemet på lång sikt. Det är viktigt att investeringar och förväntade nyttoeffekter följs upp, för att säkerställa att systemet levde upp till de behov och krav som först fastställdes i kravspecifikationen. Levison (2003) menar dock att det är vanligt att företag gärna undviker att utvärdera och följa upp effekterna av systemet. Vanliga orsaker till detta är att arbetet kräver mycket tid och personal, samt att och de som genomför utvärderingen kan vara oroliga för att presentera negativa resultat för ledningen. (Levison, 2003).

Ett vanligt syfte med utvärderingsarbetet är att identifiera eventuella fel och brister i systemet, för sedan åtgärda dessa på ett tillfredställande sätt. Utvärdering syftar även till att stärka kvalitén på systemet, och dra lärdomar av eventuella misstag inför framtiden (Nicolaus, 2004).

Utvärderingsarbetet skiljer sig dock beroende på dess omfattning och syfte. Vid utvärdering av informationssystem ligger ofta fokus på funktionalitet och teknisk kvalitet. De personer som är involverade i utvärderingsarbetet, brukar även skilja sig från projekt till projekt. Kunden kan exempelvis använda en extern konsult som arbetar specifikt med utvärderingar, men arbetet kan även utföras av de personer som ingick i projektgruppen. Med andra ord personer både från företaget och från leverantören. (Gulliver, 1987).

Utvärderingen kan vara av formell och informell karaktär. Vid en informell utvärdering ligger fokus på användarnas åsikter av systemet. Det är vanligt att den bild som användarna har, inte stämmer överens med den mer formella utvärdering som genomförs av projektgruppen, eller externa konsulter. (Hirschheim & Smithson, 1999).

Utvärderingsarbetet kan även bestå av användarenkäter, som kan ge svar på hur de anställda upplever systemet, och på vilket sätt det används. Detta ger på så vis användbar feedback och en bättre insikt kring systemets såväl fördelar som brister, åt företaget. Den kan även ge indikationer på om det satsades tillräckligt mycket på exempelvis utbildningar med mera, vilket kan ge goda lärdomar för framtiden. Leverantören kan även hjälpa företaget att identifiera fördelarna med systemet och andra nyttoeffekter, som de förmodligen inte hade förväntat sig. Det är därför viktigt att behålla kontakten med leverantören efter implementeringen, för att få support, om problem skulle uppstå eller ifall det förekommer oklarheter kring hur systemet skall användas. (Adam & O'Doherty, 2000).

### **Design/Systemspecifikation**

Designfasen vid utveckling av nya system inleds efter analysfasen, då funktioner och problem definierats (Sommerville, 2011). I artikeln *Sustainable Project Life Cycle Management: the need*

*to integrate life cycles in the manufacturing sector* (Labuschagne & Brentb, 2007) tas det upp ett flertal utvecklingsplaner för systemutveckling. I de flesta modellerna som förs fram befinner sig designfasen precis innan utvecklingsfasen och innefattar framförallt hur pass möjligt det är att uppnå de funktioner som krävs av kunden. Dessutom nämner Sommerville (2011) att det krävs ett iterativt förhållande mellan design och utveckling.

### **Implementering**

Implementering är processen för införandet av systemet i en verksamhet. Denna fas är komplex och kräver ofta mycket arbete från både företaget och leverantören. Målet med implementeringen är att kunden ska få de nyttoeffekter, som de på förhand förväntade sig av systemet. Design och implementeringsfasen är nära sammankopplade och det är därför viktigt att tar hänsyn till frågor angående införandet redan under designfasen (Davenport, 2000).

Under implementeringen är det även viktigt att leverantören analyserar hur företagets processer och strategier ser ut, detta för att se om det krävs några strukturella förändringar. (Davenport, 2000). Företaget och leverantören måste därför föra en dialog angående var i organisationen det är mest lämpligt att inleda implementeringen. Exempelvis kan det vara mer optimalt att införa systemet initialt i mindre delar av företaget, eller att lägga större fokus på huvudkontoret.

Aladwani (2001) menar att en kritisk faktor för att implementeringen ska bli lyckad är personalens attityd gentemot de förändringar som det nya systemet ofta innebär. Det rekommenderas att personalen får både stöd och utbildning, så att de känner sig trygga med det nya systemet. Denna trygghet leder ofta till en mer positiv inställning till förändringen och systemet i stort. (Aladwani, 2001).

## **2.5 Teoretisk referensram**

Efter att ha gått igenom de olika faktorer som ingår i de teoretiska modeller vi lyft fram, har vi sållat bort dem som vi inte ansåg var tillräckligt återkommande enligt tidigare fastställda kriterier. Det är således utefter dessa val som vi valt basera vår empiriska undersökning på dessa faktorer:

- Föranalys
- Kravspecifikation
- Kvalitet
- Prototypbaserad systemutveckling
- Validering/verifiering
- Iteration

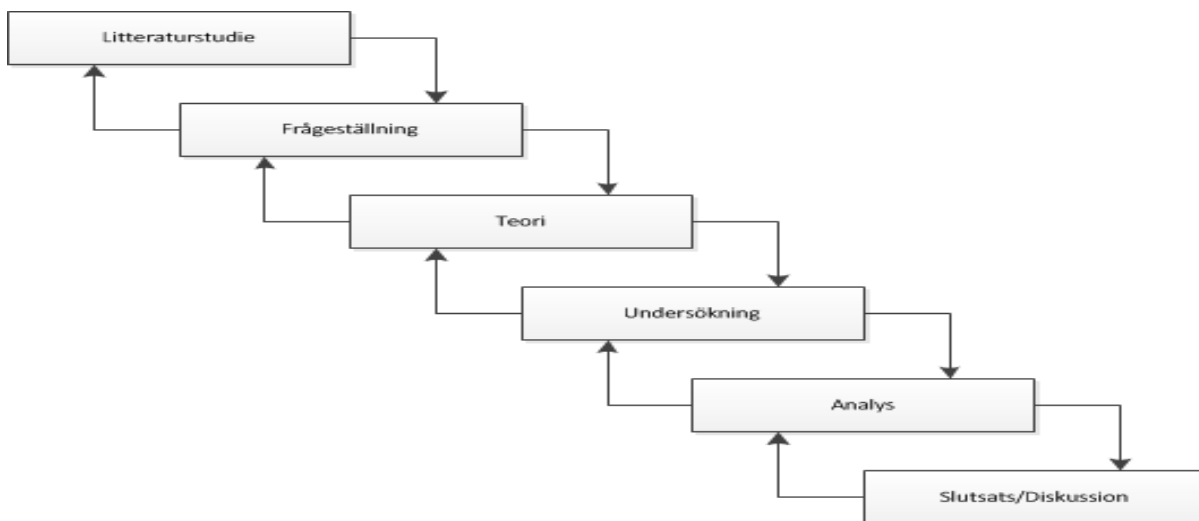
# 3 Metod

I följande avsnitt redogörs vårt tillvägagångssätt för studien. Här ingår metodiken bakom intervjuerna, strukturen vi använt oss utav under skrivandet av själva uppsatsen, framställningen av vårt teoretiska ramverk, val av insamling och bearbetning av information, samt undersökningen av dess validitet. Vi tar även upp vilka intervjufrågor som vi valt att ställa och varför, samt hur vi valt att använda informationen.

## 3.1 Tillvägagångssätt

Vi började med att presentera bakgrund taget från diverse litteratur för att ge läsaren en bild av det problem som finns i dagsläget, samt att förklara översiktligt hur IT-utvecklingsprojekt går till. Detta smalnades sedan av till en frågeställning, som baserats på de problem som tidigare beskrivits. Frågeställningen stod därefter till grund för den teoretiska referensram, bestående av de mest återkommande faktorer uttagna från våra sex utvalda modeller, som vi använt oss utav för att framställa undersökningen. Undersökningen innefattades i sin tur sedan av en empirisk uppsamling som ledde fram till det resultat som förklaras i analysdelen. Den teoretiska bakgrunden och det empiriska resultatet står tillsammans som grund för analysen, som sedan använts för att skapa en slutsats, samt diskussionsdel.

Nedan visas en figur som översiktligt beskriver hur vi arbetat:



*Modell för tillvägagångssätt 3.1*

## 3.2 Val av forskningsstrategi

I vårt val av forskningsstrategi, för att möta ett svar på forskningsfrågan, stod valet främst mellan två huvudsakliga intervjumetoder, kvalitativ kontra kvantitativ.

### 3.2.1 Kvantitativ metod

En kvantitativ metod går på bredden, det vill säga undersöker många enheter, men undersöker relativt få nyanser. Vid en kvantitativ intervju är syfte att finna omfånget, frekvensen eller utsträckningen av ett fenomen. Ett tydligt exempel på en kvantitativ undersökning är att utföra enkätundersökningar med givna svarsalternativ. (Jacobsen, D.I. 2002).

### 3.2.2 Kvalitativ metod

En kvalitativ metod ämnar koncentrera sig på ett fåtal enheter och lyfta fram många nyanser. Vid en kvalitativ intervju är målet oftast att ta fram en detaljerade åsikter om specifika fall, och gå in med antagelsen att det finns flera olika synsätt om samma sak. Detta görs lättast genom användning av öppna frågor vid individuella, alternativt gruppintervjuer. (Hedin, 2011).

Det finns en rad olika metoder som kan användas vid ett följande av en kvalitativ intervjumetod, en av dessa är semikonstruerad metod. Semikonstruerade intervjuer syftar på användning utav specifika frågor, för att sedan utveckla ledfrågor baserat på respondentens svar (Hedin, 2011).

I vårt fall ansåg vi att en kvalitativ undersökning var att föredra då vi ville jämföra svaren mellan våra olika respondenter samt se vilken bakgrund och erfarenhet de besitter.

Då målet var att få inblick i hur utvecklingsprojektet fungerar, för att sedan kunna jämföra detta med det teoretiska ramverket kände vi att det var viktigt att få djupgående svar kring de olika faktorerna och faserna.

Anledningen till att vi valt intervjuer som metod för uppsamling av empiriskt material beror på att vår teoretiska ram består av processinriktade metoder som utförs på administrativ nivå. Att skicka enkäter till slutanvändare hade inte genererat tillräckligt med värde för rapporten då den inte går att jämföra med specifika processer som används vid IT-projekt, därmed prioriteras enkätundersökningar bort från denna rapport. Utöver detta anser vi att den information vi samlat in via intervjuer kan ses som ett bra komplement till den litteratur som finns idag inom ämnet.

Då vi valde att använda oss utav semikonstruerade intervjuer hade vi endast grundläggande typfrågor som vi följde under intervjugenombången. Målet under intervjuerna var att hålla oss till en viss mall, men samtidigt skapa utrymme för respondenten.

Till en av intervjuerna använde vi oss utav en *strukturerad* intervjumetod som innebär att det ställs specifika frågor som sedan kategoriserar svaren utifrån det formulär som skapats. (Kvalitativ metod, 2013-04-13).

### 3.2.3 Val av teori

Vi började med att undersöka tidigare studier kring ämnet genom att studera akademiska uppsatser, läroböcker, tidningsartiklar, företags hemsidor och andra trovärdiga artiklar på internet som föll inom det tänkta området.

Till vår teoridel valde vi ut sex välkända systemutvecklingsmodeller, med god relevans för vår undersökning. Som vi tidigare nämnt gjordes urvalet utefter kriterierna att det skulle vara modeller med snarlikt innehåll och tankesätt, samt framtagna av etablerade författare inom området.

Vår utvalda teori fungerar ungefär som ett kvalitativt stickprov, där en undersökning av en liten mängd teori kan användas för att ge en övergripande bild för vad en större mängd teoretiska modeller står för. Då det finns en större mängd modeller och ingen uttalad best practice, för hur ett IT-utvecklingsprojekt bör utföras, är det problematiskt att finna en heltäckande sammanställning av vad teorin förespråkar. Trots detta anser vi att vår metod kan ge ett bra perspektiv och god insikt i hur teorin förhåller sig till praktiken.

### 3.2.4 Val av respondenter

Vi samlade in vår empiriska data genom intervjuer med utvalda organisationer som sysslar med IT-utvecklingsprojekt på leverantörens sida.

Den målgrupp vi helst ville ha svar från var beslutsfattare från IT-avdelningar i leverantörsorganisationer. Vi tror att en kombination av aspekter från olika organisationer behövs för att komplettera de olika synsätten, vilket slutligen bidrar med en överskådlig bild av hur IT-projekt genomförs i praktiken.

Då en av respondenterna ville förbli anonym, tog vi beslutet att låta alla vara detta. Istället för att nämna någon vid namn har vi således valt att döpa de olika respondenterna till A, B, C, D och E.

### 3.2.5 Respondent- och intervjuguide

Till intervju A, B, C och E använde vi oss utav en semikonstruerad intervjumetodik, till skillnad från intervju D som utfördes genom en strukturerad intervjumetodik. Till respondent D skickades frågorna via e-mejl, där svaren sedan kom tillbaka genom samma medel. Det skriftliga svaret har sedan kopierats rakt av och använts som empiriskt underlag.

Utöver detta skiljer sig respondenter från varandra i form av erfarenhet och arbetssätt men har en sak gemensamt, de arbetar alla på något sätt med att förbättra kund och leverantörsrelationer vid utvecklingsprojekt inom IT. Tanken var att vi ville få svar från olika bakgrunder för att sedan se ifall det kunde ha en bakomliggande faktor till den åsikt de presenterade, detta visade sig dock inte ha någon större betydelse i slutändan.

Första intervjun genomfördes med person A som i dagsläget arbetar som konsult och projektledare vid projekt rörandes Intranät. Företaget som A arbetar på har ca 70 000 anställda fördelat över 40 länder och arbetar med att förbättra företagsprocesser genom utveckling av IT-lösningar.

Intervjuperson B ansvarar för Intranätet på ett svenskt företag med ca 12 000 anställda. Till skillnad från person A så befinner sig B på kundsidan av upphandlingsprocessen. Dock har respondent B även en lång erfarenhet som leverantör sedan tidigare och kunde på så vis genom sin kunskap ge oss en bra inblick i de skillnader som förekommer mellan de olika parterna.

Respondent C arbetar som konsult på ett mindre företag med ca 20 anställda. C har betydligt mindre arbetslivserfarenhet som konsult, jämfört med respondent A och B. Vi anser dock att de synpunkter som C kom med är minst lika viktiga som de övriga, då C besitter färsk kunskaper av det teoretiska på grund av nyligen avslutade högskolestudier. Vi tror därmed att C har lättare för att relatera till de teoriska modeller vi lyft fram och på så vis kan jämföra dessa lite mer ingående med det praktiska tillvägagångssättet.

Till intervju D blev vi tvungna att använda oss av en strukturerad intervjumethodik. Intervjun genomfördes skriftligt med kommunikation via e-mejl, där vi skickade iväg våra intervjufrågor och fick tillbaka svar i skriftlig form. Respondenten i detta fall har ca 20 års erfarenhet av projekt som innefattar systemutveckling. Företaget D arbetar på fungerar som ett stöd och en slags mellanhand mellan kund och leverantör i upphandlingsprocessen. Företagets främsta uppgift är att från kundens sida gå igenom utvecklingsarbetet och välja rätt leverantör för rätt tjänst.

Intervjuperson E arbetar som ”Business Development Executive” på ett företag med ca 400 000 anställda världen över. Företaget arbetar bland annat med att utveckla Intranät och andra liknande system.

### 3.2.6 Intervjugenomförande

Nedan följer vår frågeställning som bygger på de faktorer som togs fram i vår teoretiska referensram (se s.26) som sedan använts vid intervjuerna, samt en förklarande text till varje utvald fråga. Som vi tidigare nämnt i inledningen var det svårt att hitta ett gemensamt begrepp som täckte hela den process vi valt att undersöka. Vi valde därmed vid intervjutillfällena att kalla detta för IT-upphandlingar, men förklarade den bakomliggande innebörden (se förklaringen av IT-utvecklingsprojekt, s.1) för respondenten.

- Kan du börja med att berätta lite om dig själv. Vilken befattning har du i dagsläget?
  - Kan du ge en kort beskrivning av ert företag?
  - Vilken erfarenhet har du kring IT-upphandlingar?

*Med denna fråga ville vi få en inblick i vem respondenten var, för att sedan kunna kategorisera vår respondent i form av erfarenheter och företagets bakgrund.*

- Kan du beskriva hur en IT-upphandling vanligtvis går till på ert företag?
  - Använder ni något speciellt ramverk/modell som ni utgår ifrån vid era upphandlingar eller varierar detta från kund till kund?

*Målet var att kunna fastställa om företagen förhåller sig till någon specifik modell. Vi gick in med tanken att de större företagen möjligtvis följde någon "best practice" som i så fall hade kunnat granskas.*

- Vilka faktorer i en IT-upphandling anser du vara viktigast? Varför?

*Denna fråga var viktig för att kunna besvara vår huvudfråga till uppsatsen. Vi ville även se och jämföra hur de olika respondenterna ställde sig till vilken faktor som var mest kritisk enligt dem.*

- Vilka faktorer i en IT-upphandling anser du vara mindre viktiga? Varför?

*Så här i efterhand insåg vi att den här frågan inte var så bra, då väldigt få eller rättare sagt ingen svarade på att något i en process är "onödigt". Hade behövt formuleras om.*

- Om en IT-upphandling skulle anses vara misslyckad, det vill säga kunden får inte det den ville ha - var i upphandlingsprocessen tror du att misstaget generellt sätt sker? Varför?

*Detta var en mer indirekt fråga, som egentligen var tänkt att ge svar på vilken faktor som är viktigast, men som vi medvetet valde att kamouflera med andra ord. Förhoppningen var att få ett annorlunda svar än till den tidigare frågan, vilken vi i sin tur gick att spinna vidare på.*

- Hur går ni till väga när ni sammanställer en kravspecifikation?

*Kravspecifikation var en av de faktorer som var mest återkommande bland teorierna, därför kändes det lägligt att fråga detta för att se om det fanns någon koppling.*

- Hur går ni tillväga när ni analyserar de eventuella problem som kan finnas i kundens verksamhet?

*Föranalys var en av de faktorer som var mest återkommande bland teorierna, därför kändes det lägligt att fråga detta för att se om det fanns någon koppling.*

- Hur arbetar ni för att försäkra er att kunden är nöjd med er produkt?
  - Förhåller ni er till tid-, kostnad- och kvalitet- perspektivet (projektriangeln)?

*Då vi såg att kvalitet är en återkommande faktor, blev vi nyfikna på att ta reda på hur företagen förhåller sig till de övriga aspekterna i projektriangeln, då de påverkar varandra.*

- Hur går utvecklingen av systemet till?

*Systemutveckling med inriktning på prototyper var en av de faktorer som var mest återkommande bland teorierna, därför kändes det lägligt att fråga detta för att se om det fanns någon koppling.*

- Använder ni någon form av validering/verifiering med kunden? För att försäkra att båda parter är överens.

*Validering/verifiering var en av de faktorer som var mest återkommande bland teorierna, därför kändes det lägligt att fråga detta för att se om det fanns någon koppling.*

- Kan du förklara hur de olika stegen i er upphandlingsmetod agerar iterativt med varandra?

*Iteration var en av de faktorer som var mest återkommande bland teorierna, därför kändes det lägligt att fråga detta för att se om det fanns någon koppling.*

**Dessa frågor var inte av lika hög prioritet och ställdes endast då de övriga frågorna gick fortare än planerat:**

- Vilken/vilka av ovanstående faktorer anser du är viktigast för att en upphandling ska bli lyckosam? Varför?
  - Anser du att någon ytterligare aspekt än de vi nämnt har stor påverkan för ett lyckat resultat vid en upphandling?

*Målet med denna fråga var att ta reda på om det fanns ytterligare en faktor som inte nämnts i teorin men som trots detta anses vara viktig.*

- Fungerar dagens modeller/metoder vid upphandlingar?
  - Krångliga att använda? Bra stöd? mm.



*Vi ville ta reda på om det var så att de teoretiska modellerna användes och om de i så fall var bristande på något sätt.*

**Nedanstående frågor ställdes enbart av personligt intresse och är därmed inget som vi valt att behandla i uppsatsen:**

- Användarens delaktighet beryktas vara viktig, vad anser ni?
  - Stämmer/stämmer inte?
  - Varför?
  
- Vad har ni för krav på kundens IT-kunskaper?
  - Bra/Dåligt att kunden har tidigare IT-kunskaper och vet vad de vill ha/behöver till sin verksamhet?

### 3.2.7 Analys av data

Resultatet som togs fram av att ställa teorin och empirin mot varandra analyserades sedan för att tydligare lyfta fram likheter och skillnader. Dessutom tog vi reda på om de mest kritiska faktorerna, enligt respondenterna, var lika återkommande som de faktorer som endast nämndes i minst tre modeller. Detta gjorde vi för att kunna dra paralleller mellan hur belyst en faktor var i teorin och hur viktig den ansågs vara i praktiken.

Sammanfattningsvis tog vi dessa paralleller och försökte på något sätt förklara varför vi tror att resultatet blev som det blev. Vi försökte på så vis, på bästa sätt, besvara vår forskningsfråga samt ge råd till framtida, mer djupgående studier som hanterar liknande ämnesområden.

## 3.3 Litteraturgranskning

Vår litteraturgranskning har grundat sig i sökande efter bra och omfattande modeller som berör IT-projekt ända från upphandlingsprocessen till och med själva implementeringen.

**Sökord som användes frekvent vid sökningen var:** IT-upphandling, IT/ICT/IS -procurement, requirement specifications, best practice in IT-development/IT-projects, IT/ICT/IS-project failures, misslyckade IT-projekt, IT-projekt, IT-project models, projektstrategi, modeller för IT-projekt och IT-utvecklingsmetoder.

Sökandet resulterade i att vi slutligen hittade sex olika modeller som vi valde att undersöka närmre. Dessa användes även till vårt teoretiska ramverk, för vår undersökning och för att lägga en strukturell grund.

Då vi varit medvetna om risken att färgas av författarnas tankar och åsikter har vi under litteraturgranskningens gång försökt ha ett kritiskt förhållningssätt till den litteratur vi tagit del av genom att inte blint förlita oss på en författare, utan i rimlig mån även väga in andra källor.

Genom att ta flera källor i beaktande har vi således minimerat risken av att vi färgas enbart av vad en författare anser i en fråga.

## 4 Empiri

I det här kapitlet presenterar vi det resultat som framkom, efter utförda intervjuer med fem respondenter från olika företag.

Nedan har vi gjort en sammanställning av vårt empiriska resultat, som finns bifogade som bilaga 1. Intervjusammanställningar 1-A till och med 1-E. Vi har valt att först illustrera dessa i tabellform för att ge en översiktlig bild av respondenternas svar. Därefter har vi i textform sammanfattat deras åsikter, samt ställt dessa mot varandra.

Vi kan genom en snabb överblick av tabellen, se att respondent D skiljer sig markant från de övriga respondenterna. Detta beror främst på att de enbart sköter förhandlingar mellan kund och leverantör, och därmed inte har samma arbetssätt som de övriga IT-leverantörerna. Vi har även valt att i nedanstående matris inkludera eventuellt nya faktorer som framförts av respondenterna.

	Resp A	Resp B	Resp C	Resp D	Resp E
Föranalys	Viktigt med riskanalys, så att båda parterna är medvetna om hur kostnaden påverkas av detta.	Viktigt att analysera alla relevanta avdelningar inom organisationen och inte bara inkludera beslutsfattare.	Finns stora brister i offentlig sektor där leverantören inte får vara med och utföra analysen.	Arbeta tillsammans med kunden för att fastställa vilka behov som finns och kan uppfyllas med hjälp av teknik.	Viktig att denna fas utförs korrekt så att kravspecifikationen kan utföras korrekt.
Kravspecifikation	Viktigt att företagen tidigt fastställer kostnader och risker. Ju större projektet är desto mer detaljerad bör kravspecifikationen vara.	Är den viktigaste faktorn. Om leverantören vet exakt hur systemet ska se ut vid denna fas ökar chansen att projektet blir lyckat.	En för detaljerad kravspecifikation brukar oftast ge dåliga resultat.	Viktigast. Utan den finns absolut inga förutsättningar att kunden skall få det denne behöver.	Viktigast. En bra kravinsamling leder till att kravspecifikationen utförs korrekt.
Prototypbaserad	Att använda sig utav	Förädeltester av	Prototyper används.	Ingen åsikt.	Prototyper tillsammans

system-utveckling	prototyper och konstant testa funktioner tillsammans med kund.	systemet var 6:e vecka för att försäkra sig om att de är på rätt väg.			med en frågeställning på en konceptuell nivå.
Validering/verifiering	Att gång på gång låta kunden testa delar av systemet är det bästa sättet att försäkra sig om att kunden i slutändan blir nöjd.	Använder sig utav demonstrationer som upprepas var sjätte vecka. Vid större projekt genomförs det upp till fyra releaser på ett år.	Företag C arbetar med leveransgodkännanden, vilket innebär att kunden skriver under att de fått vad de efterfrågat efter projektets avslutande – implementeringen	Genom interna tester efter 30%, 60%, 90%, vid avslut, samt 6 månader efter implementeringen.	Vid systemets färdiga tillstånd förs dialoger med kunden, de går även igenom hur väl de olika kraven uppfyllts.
Iteration	Nära kundkontakt och personkemi leder till att arbetet blir mer iterativt, vilket är att föredra.	Iteration är viktigt för att säkerställa att leverera det som kunden begär. Resulterar i en mer precis kravspecifikation.	Med en nära kontakt till kund går det snabbare upptäcka fel och korrigera dessa.	Ingen åsikt.	Är nödvändigt. Företag E förklarar dock att det kan bli svårt att arbeta iterativt i offentlig sektor.
Tid/Kostnad/Kvalitet	Mogna kunder kan prioritera väl och underlättar utvecklingsarbetet.	Bra om kunder kan ta ställning till de olika perspektiven.	Tid, kostnad och kvalitet är viktiga bitar att ha i åtanke under hela upphandlingen	Åsikt saknas.	I offentlig sektor kan t.ex. kostnad vara rörligt även om det tidigare fastställts i kravspecifikationen.
Mjuka värden	Viktigast.	En nära	Viktigt. Med en	Det är	Viktigt, men

	Mjuka värden, d.v.s. kundkontakt är essentiellt för att lyckas.	fysisk kontakt med kunden är essentiellt för att kunna bolla idéer och ta snabba beslut.	nära kontakt till kund går det snabbare att upptäcka fel, samt att korrigera dessa.	viktigt att kommunicera förväntningar både internt och externt.	blir svårt i en offentlig sektor där leverantörernas råd kan anses vara partiska. Det blir då svårare att skapa tillit.
--	---	--	---	---	---

Tabell för empirisköversikt 4.1

## 4.1 Föranalys

I företag A är det projektets ordervärde som påverkar hur pass omfattade föranalysen blir. Respondent A förklarar att samtliga förfrågningar de får in gällande utvecklingsprojekt, går igenom en intern kvalificeringsprocess. Denna process kan se annorlunda ut beroende på projekt, men går kort ut på att analysera om företaget ska svara på förfrågningen eller inte. Formalitet och omfattning beror helt på projektets ordervärde och risker. I vissa fall genomförs även en speciell riskbedömning i föranalysen av projektet, där företaget kartlägger om de kan leverera det kunden efterfrågar med lönsamhet och inom utsatt tidsplan. (Bilaga 1A).

Företag B lägger stor vikt vid föranalysen, då de anser det mycket viktigt att identifiera de faktiska affärsbehoven tidigt i projektet. Detta blev väldigt tydligt då vi ställde frågan om vilken del i ett utvecklingsprojekt som företaget anser är viktigast. Respondent B svarade då analysarbetet, och poängterade vikten av att verkligen få det rätt från början, samt att vara tydlig med vad det är som ska uppnås. Flexibilitet är också något som utmärker företagets inställning till arbetet. B menar att bara för att de börjat arbeta med något dag 1, så behöver det inte nödvändigtvis leva med genom hela projektet. Företag B har förståelse för att kunden ändrar sina preferenser och behov under projektets gång och genomför själva föranalysen inom kundens samtliga affärsområden. Detta görs för att få en helhetsbild och samla in behov, samt åsikter från olika typer av slutanvändare. Respondent B tar upp ett generellt problem vid analysarbetet, rörandes de svårigheter som finns med att få tillgång till de personer som faktiskt ska använda systemet, vilket gör det svårare vid fastställandet av de faktiska behoven. (Bilaga 1B).

Hos Företag C är det vanligt att förstudien kommer in direkt från kund. Kunden har då på egen hand analyserat sin verksamhet samt de anställda, där de tittat på hur deras befintliga system används, och vilka brister som finns. Denna information förmedlas sedan till företag C som sedan utgör underlaget för den kommande kravspecifikationen. (Bilaga 1C).

Företag D har en roll i utvecklingsprojektet som skiljer sig från övriga företag, då de är konsulter och fungerar mer som ett externt stöd mellan kund och leverantör, samt fokuserar på att göra

processen smidigare för båda parter. I föranalysen arbetar de nära kunden för att hjälpa dem förstå vilket system de behöver köpa in. Respondent D anser att ett detaljerat underlag som speglar behoven är det absolut viktigaste. Saknas denna information har leverantören väldigt svårt att förstå kundens önskemål och kan då inte föreslå en passande lösningar. (Bilaga 1D).

Företag E arbetar med föranalys på ett liknande sätt som företag A. Föranalysen inleds med att kartlägga kundens förfrågan, samt kvalificera den. Kvalificeringsprocessen går sedan ut på att värdera kundens förfrågan och besluta om den är värd att svara på. Det beslut som fattas baseras på vad det är för typ av kund, men även den totala storleken på affären. Företaget tittar även på vilken chans de har att ”vinna” affären. Detta går kort ut på att jämföra vilka konkurrensfördelar företaget självt besitter, för att sedan ställa dessa i relation till andra aktörer på marknaden. Respondent E anser att kunden ofta gör för mycket vid föranalysen. E menar att kunden helt enkelt ställer felaktiga krav då de inte riktigt vet hur de ska formulera sig, i vad exakt de vill få ut av systemet. Respondent E tar även upp problematiken som kan uppstå om kunden tillhör den offentliga sektorn. Detta beror enligt E främst på strikta regler om hur arbetet med kravinsamling och analys ska gå till. I intervjun använder respondent E exemplet: att om ett företag som specialiserar sig på att utveckla en ”blå” produkt, blir rådgiven av leverantören att ”blå” är den bästa lösningen för produkten. Kan detta anses som partiskt då leverantören anspelar på beställarens preferenser för att vinna affären. (Bilaga 1E).

## 4.2 Kravspecifikation

Respondent A förklarar att deras kravspecifikationer ofta är väldigt omfattande och kan vara upp till flera hundra sidor långa. Detta beror främst på att företaget till stor del arbetar med stora projekt/kunder. Företaget ser dock ett problem med att arbeta med större kunder då de ofta kan vara för tydliga i vad de vill ha. Detta medför en risk då det kan ställas orimliga krav och förväntningar på det system som ska utvecklas menar A. (Bilaga 1A).

Respondent B anser fortfarande, efter 15 år i branschen, att det är svårt att fastslå vad en kravspecifikation faktiskt ska bestå av. B menar att de självklart vill ha målbilden klar, men samtidigt är det inte den färdiga målbilden av produkten. Erfarenheterna som B har är att kravbildens detaljnivå och omfattning skiljer sig väldigt mycket från kund till kund. Enligt B är kunden inte speciellt skicklig på att fastställa den totala kravbild. Kunden har ofta en klar bild av IT delen av systemet, men övriga aspekter, och hur systemet verkligen ska användas glöms oftast bort. Detta är ett stort problem då B anser att en korrekt kravspecifikation är essentiell för att projektet ska bli lyckat. Företag B erhåller vanligtvis kravspecifikationen från kundens IT-avdelning som tidigt i processen kartlagt affärsbehoven. Dock kan denna kravbild landa snett hos leverantören. Respondent B tar även upp att en kravspecifikation ibland kan vara ofullständig. Kunden har kanske tagit med faktor ”A till K” men resterande delar saknas. Då gäller det att leverantören förklarar för kunden att de inte kommer få ut full effekt av systemet om dessa delar saknas, och på så vis uppmana kunden att även inkludera ”K till Ö”. Företag B arbetar mycket

med att hitta de kringliggande behoven, och involvera samtliga affärsområden. Respondent B förklarar att den kravspecifikation som de vanligen erhåller, oftast inte är den som skickas tillbaka till kunden. Specifikationen diskuteras ett flertal gånger och analyseras tillsammans med kunden för få fram vad de faktiskt efterfrågar. (Bilaga 1B).

Företag C arbetar ofta med mindre kunder och ser en problematik kring de kravspecifikationer de får in. De är ofta för detaljerade och kan bestå av orimliga krav. Respondent C menar att detta beror på att mindre kunder saknar den interna kompetens som krävs för att utforma systemets specifikationer och användningsområden. C tar även upp att en för detaljerad kravspecifikation aldrig brukar resultera i ett bra resultat. Detta blir snarare ett hinder, eftersom kunden ofta ändrar sig efterhand. Därför kan egenframställda kravspecifikationer, från kunden, vara ett stort problem, menar respondent C. (Bilaga 1C).

Företag D anser att IT-projekt ofta misslyckas på grund av en bristande kravspecifikation. De jobbar därför nära kunden för att försäkra sig om att arbetet blir utfört så korrekt som möjligt. Arbetet med att sammanställa kravspecifikationen påbörjas efter att föranalysen är avklarad. Detta arbete genomförs i ett slags workshopsformat med kunden. Utifrån kravspecifikationen skapas sedan ett förfrågningsunderlag som skickas till tänkbara leverantörer. (Bilaga 1D)

Respondent E förklarar att kravspecifikationen, oftast är något som styrs av kunden. Företag E får med andra ord in kundens uppfattning av vad som efterfrågas. Kravspecifikationen blir på så vis en självklar faktor som företaget måste leva upp till. Respondent E anser dock att det ofta är här som felet kan uppstå. Kunden ställer helt enkelt felaktiga krav, på grund av att de inte vet hur de ska formulera sig, i fråga om vad exakt de vill få ut av systemet. E menar att detta ofta resulterar i att systemet slutligen inte kan användas. Då företag E arbetar med utvecklingsprojekt både i offentlig och privat sektor, kan arbetet kring kravspecifikationen variera. E förklarar att vid ett offentligt projekt är arbetet mer komplext. Leverantören kan då uppfattas som partisk om de påverkar projektet från kundens sida förklarar E. Respondent E förklarar att offentliga IT-upphandlingar är extremt formella. Kraven ska formuleras på ett visst sätt och innehålla svar på specifika frågor. Vid privata upphandlingar är det lite lättare vad gäller denna bit, då kraven inte är lika höga säger E och menar på att företaget kan föra en parallell dialog med kunden utan att någon höjer på ögonbrynen. (Bilaga 1E).

Samtliga respondenter (A, B, C, D och E) understryker vikten av att så tidigt som möjligt fastställa kraven för det system som ska utvecklas. B, D och E ansåg även att denna faktor var den mest kritiska utav de sju faktorer vi lyft fram i vår teori. Samtliga respondenter ansåg att genom att lägga stor vikt vid kravspecifikationen redan vid början av projektet, lär det ge positiv utdelning i slutändan då de övriga stegen blir lättare att genomföra. Vårt empiriska resultat tyder alltså på att den mest kritiska faktorn är kravspecifikationen. Dock påpekade alla respondenter att de resterande faktorerna inte bör underskattas.

## 4.3 Prototypbaserad systemutveckling

Företags A använder sig av prototyper i olika omfattning beroende på vilket projekt de arbetar med. Vid ett projekt som exempelvis rör intranät, är prototypdemonstrationer en central del och en effektiv metodik. I dessa fall förespråkar företaget att så tidigt som möjligt börja med demonstrationer, för att kunna göra justeringar allt eftersom. Respondent A förklarar att detta arbetssätt möjliggör att tidigt få kundens åsikter på vad som är rätt/fel eller vad som fattas. A förklarar även att kunden känner en större delaktighet, om denna arbetsmetodik används. Kunden kan då fortlöpande testa av systemet, och känna att utvecklingen är på väg i rätt riktning. Företaget betonar vikten av att hela tiden låta kunden testa något konkret. Detta ger helt enkelt kunden ökad möjlighet att påverka utvecklingen av arbetet. (Bilaga 1A).

Företag B arbetar främst med prototyper i form av rika bilder. Kort går detta ut på att demonstrera systemet i bildmiljö för kunden. Dessa demonstrationer upprepas sedan ca var sjätte vecka. Vid större projekt kan det genomföras upp till fyra del-releaser av ett system under ett år. Respondent B benämner detta arbetssätt som ”testdriven utveckling”, vilket är ett agilt arbetssätt. Allt byggs upp i en utvecklingsmiljö och först när de uppnått 60 % av alla testcase godkänns den första delen. Efter det går arbetet vidare till något som kallas ”pre-production”, som är en form av valideringsmiljö. För att sedan gå vidare måste 80 % av alla testcase vara utförda och slutligen 100 % innan det går ut i produktion. Företagets krav för detta är extremt strikta, och ingen produkt får gå direkt till livemiljön utan att ha gått igenom samtliga nivåer. (Bilaga 1B).

Respondent C förklarar att de på sitt företag utför prototyparbete på så vis att kunden får verifiera delprototyper under projektets gång. Detta möjliggör att det blir lättare att identifiera fel och oklarheter, samt om så krävs, gå tillbaka och revidera kravspecifikationen. (Bilaga 1C).

Företag E förklarar att de i vissa fall använder prototyper men att det varierar. I vissa projekt frågar de enbart kunden vilka grundläggande funktioner systemet ska hantera. Dessa frågor ställs då på en konceptuell nivå. (Bilaga 1E).

Vid utvecklandet av system som exempelvis intranät, ansåg respondent A, B, C och E att det var viktigt att använda sig av prototyper. Detta för att försäkra sig om att kund och leverantör har samma uppfattning kring det system som ska utvecklas. Det agila arbetssättet framhålls starkt utav A, B, C och E. Det handlar inte enbart om att få instruktioner av kunden för att sedan på egen hand lösa problemet, och slutligen presentera det färdiga systemet (Respondent A, Bilaga 1A). Respondenterna menar att det är viktigt att kontinuerligt stämma av med kunden och säkerställa att inga missförstånd skett i kommunikationen mellan parterna. Respondent A och B förklarar mer detaljerat tillvägagångssätten för prototyputveckling. Respondent B förklarar att de använder sig av 6 veckors iterationer där de visar upp systemet i bildmiljö för kunden, en form av prototyparbete helt enkelt. Detta gör de för att säkerställa att det arbete som utförts är korrekt och att det har anpassats efter kundens behov. (Bilaga 1A, 1B, 1C och 1E).



## 4.4 Validering/Verifiering

Företag A använder acceptanstestning när deras kunder ska validera/verifiera systemet. Detta innebär att kunden får utvärdera och testa det slutgiltiga systemet och ge sitt godkännande. Om kunden inte är helt nöjd träffas parterna och går igenom vad som finns kvar att göra. Dessutom genomför företaget fortlöpande interna tester, för att kunna identifiera möjliga brister i systemet. (Bilaga 1A).

Företag B tar ofta själv beslut angående upplägg och omfattning av valideringsprocessen. Det gäller att kunden är beredd att acceptera detta arbetssätt, om de vill inleda ett samarbete. Respondent B poängterar att de lägger stor vikt vid test och utvärdering av systemet. B förklarar att om de sammanställer timmarna i ett projekt, så är utvecklingstiden mycket liten i jämförelse med den tid som läggs på test. Företaget erbjuder en garantiperiod på 6 månader efter att systemet gått "live". Garantiperioden kan dock variera och specifikationer kring detta brukar finnas i upphandlingens kontrakt. (Bilaga 1B).

Företag C fastställer tidigt i projektet, om eventuellt efterarbete, form av verifiering, ska förekomma. Företaget arbetar med leveransgodkännanden efter projektavslutet alternativt implementeringen. Vilket innebär att kunden bekräftar att det system som levererats, verkligen var det som efterfrågades. Det sker dock verifieringar under arbetets gång, för att säkerställa att det inte blir fel på vägen. Efter lansering, erbjuder företaget även utbildning för de kunder som känner ett behov av det. Detta brukar i så fall lyftas fram i kravspecifikationen. (Bilaga 1C).

I företag D erbjuds validering/verifiering i form av interna uppföljningar, efter 30%, 60% och 90% av projektet. Företaget gör även uppföljningar med kunden under, och direkt efter att projektet avslutats. (Bilaga 1D).

Respondent E förklarar att deras företag utför olika typer av valideringsprocesser. Exempelvis om kunden ställt upp 20 krav på vad systemet ska innefatta, så används dessa som valideringskrav. Företaget betonar vikten av att föra dialoger med kunden, så att de kan hantera ytterligare krav som kan dyka upp under IT-utvecklingsprojektets gång. Företaget gör även kundundersökningar för att försäkra sig om att de har gjort ett bra jobb. E förklarar att de kontinuerligt återkopplar med kunden och håller en tät kontakt. Att arbeta på detta sätt leder ofta till att företaget möter kundens krav i slutet på projektet. (Bilaga E).

Validering/verifiering syftar i teorin på det avslutande steget, som innefattar att kunden godkänner leverantörens arbete. Respondenterna hade dock ibland svårt att avskilja denna fas från de övriga faktorerna, då de till viss utsträckning enligt dem är sammansatta (A, B, C, D och E). Respondent A förklarade hur det underlättar att arbeta agilt, alternativt iterativt med kunden för att på så sätt förstärka den avslutande fasen, som innebär någon form av validering från kunden. Att använda sig utav prototyper för att med ökad noggrannhet utveckla den färdiga produkten är, med andra ord, något som respondenterna A, B, C och E understryker. Det blir av

denna anledning svårt att avskilja denna fas från övriga faktorer. (Bilaga 1A, 1B, 1C, 1D och 1E).

## 4.5 Iterationer

Respondent A förklarar att ett iterativt och agilt arbetssätt är nyckeln för att lyckas med ett IT-utvecklingsprojekt. Enligt A kan leverantören och kunden på så vis tillsammans lösa problem mer effektivt, om båda parterna har större förståelse för de aktiviteter som utförs i nuläget. (Bilaga 1A).

Nära kontakt med kunden är något som även företag B värdesätter. Att ta reda på vem som är ägaren av systemet, är en viktig del vid säkerställandet av god kvalitet. Som sagt tidigare arbetar företag B med 6 veckors iterationer, för att sedan demonstrera systemet i bildmiljö för kunden. Demonstrationerna upprepas sedan var sjätte vecka. Vid större projekt genomförs det upp till fyra releaser på ett år, och även om inte iterationer är den mest kritiska faktorn enligt Respondent B, så är den trots allt relevant. (Bilaga 1B).

Respondent C anser att iterationer med kunden är väldigt viktigt för att lyckas med IT-utvecklingsprojekt. C förklarar att återkoppling gör det möjligt att hantera och korrigera fel i exempelvis kravbilden, eller vid möjliga designfel. Genom att lägga stor vikt vid återkoppling kan risken för att fel uppkommer minimeras. (Bilaga 1C).

Respondent E ansåg också att det iterativa arbetssättet är nödvändigt. E tog upp ett exempel kring problematiken som kan uppstå vid exempelvis en upphandling med företag inom den offentliga sektorn, där en alltför nära relation mellan kund och leverantör inte ens är tillåten. Detta kan enligt honom ställa till det för båda parterna då bra beslut från kundens sida blir svåra att ta. (Bilaga 1E).

Enligt respondenterna A, B, C och E är ett iterativt arbetssätt vid IT-utvecklingsprojekt väsentligt. Alla respondenter är överrens om att en hög nivå av iteration mellan kund och leverantör, resulterar i att IT-utvecklingsprojekt blir lättare att modifiera, då kontakten sker mer effektivt. (Bilaga 1A, 1B, 1C och 1E).

## 4.6 Tid/Kostnad/Kvalitet

Respondent A lyfte fram att vissa kunder har väldigt tydliga prioriteringar, men att det handlar om en mognadsfråga. A menar att omogna kunder ofta har svårt för att prioritera, medans de mogna kunderna oftast är ärliga och kan erkänna att exempelvis ett visst datum är livsviktigt. Om tidsaspekten är det centrala kan företaget lägga mindre vikt vid systemets funktionalitet. Det förekommer dock även projekt där kostnaden är det viktigaste. Exempelvis kanske kunden enbart

har ett visst belopp att röra sig med, då brukar de vara mer flexibla angående tid och funktionalitet. (Bilaga 1A).

När vi nämner tid-/kostnads-/kvalitets- perspektiven för respondent B, förklarar B att det kan kopplas till det agila arbetssätt de använder. Genom att arbeta agilt, måste företaget kontinuerligt tänka på vad som ska göras och hur det ska prioriteras. (Bilaga 1B).

C Anser också att tid, kostnad och kvalitet är viktiga bitar att ha i åtanken under hela projektet. (Bilaga 1C).

Respondent E instämmer och förklarar att i offentlig sektor kan exempelvis kostnad vara rörligt, även om den tidigare fastställts i kravspecifikationen. E förklarar att om kunden exempelvis inte besitter kunskaper nog att förstå vad produkten, som leverantören presenterar, innehåller så kan dessa kunder inte förhålla sig till projektriangeln på rätt sätt. Detta då de inte kan skilja mellan kostnad och kvalitet. (Bilaga 1E).

Respondenterna A, B, C och E håller alla med om att IT-utvecklingsprojektet bör förhålla sig till aspekterna tid, kostnad och kvalitet, men förklarar det på olika sätt. A går mest in på fördelarna som innefattar kundens mognadsfråga och vilka positiva egenskaper det innefattar. E förklarar istället problematiken som kan uppstå när kunden inte förstår funktionaliteten (scopet) bakom produkten, och därmed inte kan förstå hur kostnaden är beroende av det. B och C håller med om att det är viktigt, men förklarar inte närmre orsaken till detta. (Bilaga 1A, 1B, 1C, och 1E).

## 4.7 Mjuka värden

Mjuka värden var en faktor som inte behandlades nämnvärt i teorin men som samtliga respondenter tog upp som en avgörande kritisk faktor för att lyckas med ett IT-utvecklingsprojekt.

Respondent A förklarar att det finns en risk med att enbart fokusera på de hårda bitarna, det vill säga faktorer som teknik, avtal och pengar. A menar att de mjuka delarna bör prioriteras mer. A anser även att det är en fördel att arbeta med återkommande kunder, på grund av att kunden känner till företaget och företaget känner till kunden. A gick även in på att kundens mognadsnivå kunde vara en avgörande faktor som stod i vägen för att projektet ska kunna lyckas. Han förklarade att en omogen kund varken kan prioritera eller resonera med leverantören, beträffande vilka uppoffringar som måste göras för att systemet ska bli så bra som möjligt. Respondent A förklarar att affärer trots allt handlar om människor, och vikten av att komma överens. Ett vanligt misstag är att enbart fokusera på de hårda bitarna, det vill säga teknik, avtal och pengar. De mjuka delarna är det primära enligt respondent A och en förutsättning för ett lyckat resultat. De mjuka värdena är därmed enligt A den mest kritiska och därför så även den mest avgörande faktorn. (Bilaga 1A).

B håller också med om att en nära fysisk kontakt med kunden är essentiellt, för att kunna bolla idéer och ta snabba beslut. Detta steg leder enligt B till att de andra faktorerna blir mer korrekta. (Bilaga 1B).

Respondent C förklarar att god kommunikation krävs för att exempelvis kunna övertala kunden om att en viss funktion är betydelsefull. Detta krävs enligt C för att kunna bygga ett välfungerande system. C förklarar att om de mjuka värdena saknas blir det ofta svårt att genomföra ett projekt med lyckat resultat. Respondenten menar att ett gott samarbete mellan kund och leverantör är enormt viktigt och helt avgörande för IT-utvecklingsprojektet. C anser, liksom A att den viktigaste faktorn i ett IT-utvecklingsprojekt är de mjuka värdena. (Bilaga 1C).

Enligt respondent D är det viktigt att kommunicera förväntningar både internt och externt. (Bilaga 1D).

Respondent E har en liknande uppfattning som de övriga och nämner hur de mjuka värdena är viktiga, men att det kan vara svårt att applicera vid IT-utvecklingsprojekt inom den offentliga sektorn, då en nära kontakt med kunden inte är tillåten. En allt för nära kontakt med kunden, kan i dessa fall resultera i att leverantören försöker påverka marknaden, genom att insinuera att deras produkt är den som är nödvändig, för att skapa ett bra lösningsförslag. Tillit mellan kund och leverantör blir då med andra ord svårare att skapa. (Bilaga 1E).

De mjuka värdena är något som samtliga respondenter anser är viktigt (A, B, C, D och E) och som enligt A och C är den absolut viktigaste. Detta motiverar de genom att förklara att med en bra relation till kunden, går det att lösa de flesta problem då tillit och förståelse är en given faktor.

# 5 Analys

I vår analys ämnar vi att jämföra det resultat som framkommit vid vår genomförda undersökning, och ställa detta i relation till den teori som presenterats i litteraturgenomgången.

## 5.1 Föranalys

Om vi studerar empirin och teorin mer ingående ser vi att föranalysen är en faktor som kan vara svår att specificera. Respondenterna A, C och E förklarar hur de som leverantörer ibland inte behöver göra en föranalys, då detta sker internt hos kunden. I vissa fall kan det dock vara så att kunden inte är kompetent nog att analysera sina behov med hjälp av tekniska termer. Då är det bättre att leverantören går in och gör analysen åt dem. Respondent E nämnde dessutom att vid ett IT-utvecklingsprojekt inom den offentliga sektorn kan detta bli problematiskt då en föranalys framförd av en leverantör kan anses vara partiskt. (Bilaga 1A, 1C och 1E). I de modeller där ”föranalys” ingår (Sommerville, 2011, s.30, s.35, Bevan & Maguire, 2002) nämns det inte i detalj varför det görs eller vilken problematik som kan uppstå vid detta steg, utan modellerna beskriver endast bakomliggande metoder och tillvägagångssätt för hur en föranalys borde genomföras enligt dem. Det vi kan se här är att vårt empiriska resultat (Bilaga 1A, 1C och 1E) kompletterar vår valda teori genom att beskriva problematiken mer ingående.

Sommervilles prototypintensiva utvecklingsmodell (Sommerville, 2011, s.32) lägger ingen energi alls på föranalysen utan baserar sitt arbete på ett dokument där kunden listat upp sina behov. Detta blir, med andra ord det enda dokument som används som grund till utvecklingsfasen, vilket är ett tydligt exempel på hur lite föranalyser prioriteras i vissa modeller. Det är dock enligt respondenterna ovanligt att inte föra någon sorts analys vid ett IT-utvecklingsprojekt (Bilaga 1A, 1C och 1E).

## 5.2 Kravspecifikation

Respondent B, D och E anser att kravspecifikation är den absolut viktigaste faktorn i ett IT-utvecklingsprojekt. Respondent B motiverade detta genom att relatera till sin egen erfarenhet och hänvisade till att, om grunden är dåligt genomförd så kommer projektet troligtvis inte att bli lyckat. (Bilaga 1B, 1D och 1E). Kravspecifikationens betydelse är även något som bekräftas av de utvecklingsmodeller som ingår i vår studie, där flertalet lägger stor vikt vid denna faktor (Sommerville, 2011, s.30, s.32, s.35, Bevan & Maguire, 2002, Heckman, 1999, Verville & Haltingen, 2003). Även respondenterna A och C anser att kravspecifikationen har betydelse, men de anser inte att det är den mest kritiska faktorn i utvecklingsarbetet (Bilaga 1A och 1C). Rent statistiskt är således kravspecifikationen den mest återkommande faktorn, både enligt teorin och enligt den empiriska informationen.

En intressant aspekt att ta upp, är att kravspecifikationens omfattning och detaljnivå verkar påverkas av projektets omfattning och risknivå. Detta kan kopplas till företag A som ofta är involverade i större projekt. Företaget förklarar att det inte är ovanligt att de kravspecifikationer de hanterar ofta kan vara på flera hundra sidor. Dock ser de ett problem i detta, och menar att en för detaljerad kravspecifikation kan resultera i orimliga krav och förväntningar. Kunderna blir helt enkelt alltför tydliga i vad de vill ha. (Bilaga 1A). Företag C har samma uppfattning och menar att en för detaljerad kravspecifikation kan vara ett hinder, då kunden ofta ändrar sig under projektets gång (Bilaga 1C). Detta kan relateras till det generella problem som Lehmann & Lattemann (1997) tar upp, där de menar att leverantören och kunden ofta har olika uppfattningar, om vad systemet faktiskt ska innehålla.

Torsten Cegrell, professor vid KTH, tar upp ett liknande problem angående kundens beställarkompetens. Kundens uppfattning är oftast att ju mer avancerat ett system är, desto bättre är det, vilket sällan är fallet. (Würtemberg, 2010). I empirin framgår det även att en generell uppfattning är att kunden inte är speciellt skicklig på att fastställa den totala kravbild. Studien har visat på att kunden ofta har en klar bild av en viss del av systemet, men att övriga aspekter och hur systemet verkligen ska användas glöms bort. Exempelvis menar företag E att kunden ofta ställer felaktiga krav, då de har svårt att formulera sig i vad deras användare faktiskt vill få ut av systemet. En tänkbar lösning på problemet är, att leverantören arbetar mer aktivt och detaljerat med att fastställa kundens behov. Det gäller även att detaljerat kunna diskutera detta med kunden, som därmed kräver en bra dialog mellan parterna. Detta kan då förhoppningsvis resultera i att kunden får en inblick i vad systemet faktiskt ska innehålla. (Bilaga 1A, 1B, 1C, 1D och 1E). Med andra ord så kan vi se att kravspecifikationen är något som både teorin och empirin lyfter fram som en viktig faktor.

## 5.3 Systemutveckling/prototyper

Respondenterna A, B, C och E nämner i intervjuerna, hur systemutveckling bör ske med hjälp av prototyper. Återigen kommer de in på hur det iterativa/ agila arbetssättet är essentiellt, och hur viktigt det är att hela tiden stämma av med kunden, för att försäkra sig om att det som utvecklas faktiskt är det som kunden efterfrågar. (Bilaga 1A, 1B, 1C och 1E). Detta påminner oss om de ord vi fått höra under hela utbildningen på Lunds Universitet och som vi lyfte fram i bakgrunden.

I de teoretiska modeller som ingår i vår studie är prototypbaserad utveckling något som ofta lyfts fram som ett effektivt sätt att arbeta på. Även vårt empiriska material styrker detta, där exempelvis respondent A ser en stor fördel med att använda sig av en prototypbaserad metodik för systemutveckling. Erfarenheterna som respondent A har är att prototyper leder till att deras kunder känner en större delaktighet i projektet, och att de har chans att påverka utvecklingen. (Bilaga 1A). Något vi uppmärksammat är att prototypbaserad utveckling även kan användas som ett verktyg för verifiering.

I vårt empiriska material framgår det att kunden ofta får utföra regelbundna tester och godkänna olika delversioner av system. Generellt anser företagen i vår studie att detta underlättar möjligheten att identifiera brister, och därmed kunna åtgärda dessa i ett tidigt skede. (Bilaga 1A, 1B, 1C och 1E). Detta kan bero på att prototyper medför en högre grad av realism än vad som är möjligt att uppnå med exempelvis skisser och muntlig dialog (Bevan & Maguire, 2002). Därigenom kan företag undvika att dessa fel utvecklas, vilket annars kan leda till större problem som kan vara svåra att hantera när det slutgiltiga systemet är i drift. Även Sommerville (2001) framhåller detta i sin prototypbaserade utvecklingsmodell. Sommerville (2001) beskriver att prototyper av olika slag gör det lättare att upptäcka fel i inledningsfasen av projektet. Om leverantören åtgärdar dessa fel på ett effektivt sätt kommer det med sannolikhet resultera i att projektets totala kostnader blir lägre. Däremot kan det bli kostsamt att skjuta upp problemlösningar till ett senare skede till exempel på grund av tidspress. (Sommerville, 2001).

Vårt att nämna är att företag D inte gav några kommentarer huruvida prototypbaserad utveckling förekommer i deras organisation. Vi tror att detta beror på att företaget mer fungerar som ett stöd och en slags mellanhand mellan kund och leverantör i utvecklingsprocessen. Bortser vi från denna avvikelse kan vi tydligt se, att resultatet pekar på att prototypbaserad systemutveckling är att föredra, båda enligt teorin och empirin.

## 5.4 Validering/verifiering

I vår empiri kunde vi se att validering/verifiering var något som alla respondenter arbetade med mer eller mindre, i olika former. Interna tester, prototyptester, hög kundkontakt, iteration, återkoppling, kundundersökning och god kommunikation är de faktorer som nämns vid intervjuerna. Det vi kan se är att faktorerna överlappar varandra och att vissa faktorer står till grund för valideringen. (Bilaga 1A, 1B, 1C, 1D och 1E).

Respondent A förtydligar att verifiering/validering och iteration är något som ligger nära varandra. A menar på att hålla en nära kontakt och konstant stämma av delar av systemet, under utvecklingsfasen, är det bästa sättet att få kundens godkännande. Även i teorin ser vi att den utvecklingsmodell som tagits fram av Verville & Haltingen (2003) innefattar ett avslutande steg som syftar på att kunden, efter utrullningen, bör ha någon sorts utvärdering av systemet. I denna utvärdering kan det exempelvis visas vilka behov som mötts och tvärt om. Dessutom finns det i detta steg en form av koppling till de tidigare nämnda aspekterna tid, kostnad och kvalitet som tas upp i denna modell. Kund och leverantör bör alltså inte bara se om behoven har mötts utan också se hur väl de har mötts gentemot den tidsram och budget som fastställts i planeringsfasen. Bevan & Maguire (2002) går också in på samma spår, men menar på att enkät/intervju - undersökningar inte räcker för att bedöma hur väl behoven tillfredsställs, utan måste kompletteras med någon form av simulering, exempelvis via prototypstestning som respondent A, B, C och E starkt rekommenderar. Respondent B förklarade exempelvis att de i deras

organisation gör dessa tester ungefär var 6:e månad för att kunna försäkra sig om att de är på samma spår som leverantören. (Bilaga 1A, 1B, 1C och 1E).

I den övriga teorin såg vi att vattenfallsmodellen (Sommerville 2011) hanterade kontrollering av kvalitet i form av implementering och enhetstestning, då först efter att systemet utvecklats. Detta kan innebära stora problem för projektet då en hel del arbete kan vara förgäves om det skulle visa sig att det färdiga systemet inte uppfyller kvalitetskraven. Sommervilles prototypintensiva utvecklingsmodell behandlar validering genom att använda sig utav prototyper och enhetstestning under projektets gång. Sommervilles återanvändningsmodell för mjukvaruutveckling (Sommerville, 2011, s.35) nämner inte verifiering i sin modell, utan är mer fokuserad på återanvändandet av föreliggande komponenter. Enligt Heckman (1999) sker verifiering genom utförandet av produkttester och olika kvalitetskontroller med leverantören. Bevan & Maguire (2002) implementerar också validering i delstegen ”kartläggandet av användningsområdet”, ”utvärdering av nuvarande och konkurrenters system”, samt i hela steg 3 ”frågeställning och utvärdering”. Validering/verifiering är alltså något som både teorin och empirin tar upp och anser är en viktig faktor, dock inte i samma utsträckning som exempelvis kravspecifikationen.

## 5.5 Iteration

Det iterativa förhållandet var enligt A, B, C och E väldigt viktigt vid IT-utvecklingsprojekt. (Bilaga 1A, 1B, 1C och 1E). Detta understryks även av några av de teoretiska modeller vi lyft fram i teoriavsnittet: Sommervilles prototypintensiva utvecklingsmodell, Sommervilles återanvändningsmodell för mjukvaruutveckling och Bevan & Maguires modell för IT-utvecklingsprojekt som förklarar hur och varför det är viktigt att arbeta iterativt. I Sommervilles återanvändningsmodell syftar författaren på det iterativa arbetet mellan de olika faserna i processen, medan i den prototypintensiva utvecklingsmodellen menar Sommerville på att iterationen mellan kund och leverantör är det essentiella. Bevan & Maguires modell liknar mer den sistnämnda modellen, genom att iteration är något som krävs för att verkligen fastställa kundens behov. (Sommerville, 2011, s.32, s.35. Bevan & Maguire, 2002).

Respondenterna A och C förklarar hur ett iterativt förhållande till kunden, där ett konstant avstämmande av funktioner i systemet, är nödvändigt för att kunna kvalitetssäkra systemet som ska utvecklas. Precis som med en bra kundkontakt, så tillåter det leverantören att gå tillbaka till en föregående fas och ändra i IT-utvecklingsprojektet vid eventuella fel (Bilaga 1A, 1B, 1C och 1E). Respondent C anser att ett gott samarbete mellan kund och leverantör, samt ett stort engagemang från kundens sida, måste finnas. Respondent C förespråkar det iterativa arbetet, med en tät kundkontakt, då det medför hög flexibilitet kring arbetssätt och en möjlighet att anpassa sig efter kundens behov. Iteration anses även vara ett sätt att kvalitetssäkra. (Bilaga 1C). Respondenterna B och E nämner också att en tät relation med kunden är centralt, men anser inte att det är här som det bör läggas mest fokus (Bilaga 1B och 1E). De olika modellerna vi lyft



fram, förklarar iteration på två olika sätt: Kund- Leverantörsrelation och iteration mellan faser (Sommerville, 2011, s.32, s.35, Bevan & Maguire, 2002). Respondenterna A, B, C och E ser det däremot som en och samma funktionalitet, och förklarar att genom hög iteration mellan kund och leverantör öppnas möjligheten att arbeta iterativt mellan de olika faserna. Förmågan att kunna gå tillbaka och ändra utan att påverka utvecklingen för mycket, är det respondenterna strävar efter. (Bilaga 1A, 1B, 1C och 1E). Respondent A och C definierar detta som “att arbeta agilt”, vilket enligt dem är det mest optimala arbetssättet för ett IT-projekt (Bilaga 1A och 1C).

## 5.6 Mjuka Värden

De mjuka värdena är en i stort sett ny faktor som lyfts fram av våra respondenter, som syftar på personkemi, gemensam förståelse, tillit och god kommunikation mellan parterna i ett IT-utvecklingsprojekt, och som vi valde att ta med i beräkningen efter att ha analyserat vårt empiriska resultat. Denna faktor är enligt A och C den absolut viktigaste. Detta motiveras genom att en god relation med kunden resulterar i att de övriga faktorerna alltid kan modifieras. Respondenterna (A, B, C och E) förklarar hur en god relation till kunden kan underlätta de övriga faser och faktorer som ingår i ett IT-utvecklingsprojekt. (Bilaga 1A, 1B, 1C och 1E).

Anmärkningsvärt är att mjuka värden enbart tas upp i en av de modeller som tagits med i vår studie, dessutom nämns det bara ytligt. Modellen som tar upp detta är Heckmans modell för IT-utvecklingsprojekt där det ges en kort beskrivning kring mjuka värden. Han förklarar att leverantörshantering bör ingå i ett IT-utvecklingsprojekt, då detta steg syftar på att kunden anordnar aktiviteter med leverantör för att på så sätt optimera relationen. (Heckman, 1999). A nämner att genom en god relation till kunden, skapas förtroende för att leverantören prioriterar kundens intressen och gör sitt bästa för att leverera sin lösning. C understryker detta och förklarar att med en nära kontakt till kunden, kan det gå snabbare att upptäcka fel och korrigera dessa. Att mjuka värden är viktigt, håller B, D och E också med om, men inte i samma utsträckning som med kravspecifikationen. E förklarar dock hur de mjuka värdena är svårare att skapa i IT-utvecklingsprojekt som rör sig inom den offentliga sektorn. Anledningen till detta är enligt E, att en för nära relation till kunden kan anses vara partisk och är därför otillåtet. Att till exempel gå in och föra en offentlig kundanalys och samtidigt vara den som utvecklar lösningen, är något som kan verka fördelaktigt och är därmed även otillåtet. A går dessutom in på vilka fördelar som ingår om en och samma leverantör ansvarar för både analysen av problemet och står för själva lösningen, och hur de mjuka värdena i ett sådant fall är essentiella. (Bilaga 1A, 1B, 1C, 1D och 1E). Det är intressant att se hur en faktor som i praktiken håller ett stort värde, inte poängteras lika mycket i teorin.

## 5.7 Tid, Kostnad och Kvalitet

De olika aspekterna tid, kostnad och kvalitet är något som både i teorin och praktiken verkar vara självklarheter, och som lyfts fram som en betydande faktor. Exempel på detta är Sommerville som i sin vattenfallsmodell (Sommerville, 2011, s.30) nämner hur den traditionella vattenfallsmodellen inte tillåter iteration mellan faser, vilket sparar på tid och kostnad, men leder till sämre kvalitet. Även Verville & Haltingen (2003) förklarar hur kunden vid utvärderingsfasen bör kartlägga IT-utvecklingsprojektet i ett tids-, kostnads- och kvalitetsperspektiv, för att se hur väl behoven mötts. Om det visar sig att de inte mötts, bör påverkan av kostnads- och tidsram överses. Respondent A tycker också att dessa aspekter är viktiga men förklarar att det främst handlar om kundens mognadsnivå. En mogen kund kan enligt honom prioritera rätt gällandes kvalitet, tid och kostnad. (Bilaga 1A). Respondenterna B, C och E håller med om att dessa aspekter är viktiga att ha i åtanke. (Bilaga 1B, 1C och 1E).

Tid och kostnad är ännu två faktorer som försumrats i vår utvalda teori, i den mån att de inte var tillräckligt återkommande, men som starkt lyfts fram i vår empiriska undersökning. Det som kan sägas om faktorernas förhållningssätt till praktiken, är att det verkar vara en självklarhet från respondenternas sida. I många av fallen nämnde respondenterna dock endast kort hur viktig kostnaden är, men då vi bad dem beskriva detta närmre gick de även in på de övriga två aspekterna. (Bilaga 1A, 1B, 1C och 1E).

## 5.8 Sammanfattning av analys

Efter att ha analyserat och ställt teorin mot vårt empiriska material kan vi konstatera att det framkommit faktorer som i vår teori inte var tillräckligt återkommande, men som i praktiken är av stor relevans för ett IT-utvecklingsprojekt. Dessa är: ”mjuka värden” och tid, kostnad och kvalitet. Det intressanta här är att de teoretiska modeller vi valt inte behandlar dessa aspekter i någon större utsträckning, med undantag för kvalitetsperspektivet. Vårt att notera är även att mjuka värden ansågs, enligt våra respondenter, påverka ett projekt i större omfattning än tid, kostnad och kvalitet. Däremot ansågs tids-, kostnads- och kvalitetsperspektiven vara relevanta, men betonades inte i samma grad som mjuka värden. Analysen har även bekräftat kravspecifikationens betydelse, något som understryks av både våra respondenter och utvalda teoretiska modeller. (Bevan & Maguire, 2002, Heckman, 1999, Sommerville, 2011, s.30, s.32, s.35, Verville & Hallingten, 2003).

En annan tanke som uppkom efter att ha analyserat respondenternas åsikter, är att de mjuka värdena är svåra att avskilja från de övriga faktorerna i ett projekt. Ett av många exempel är arbetet med kravspecifikationen. Respondent A, B, C och E poängterar att en bra kravspecifikation skapas genom en tät kommunikation med kunden (Bilaga 1A, 1B, 1C och 1E). Alltså borde ”mjuka värden” vara centralt, och nära sammankopplat med detta arbete.

## 6 Diskussion och Slutsats

I detta kapitel ämnar vi lyfta fram vårt resultat av forskningsfrågan “Hur väl följer leverantörer de generella, teoretiska rekommendationer som finns, i sitt praktiska arbete?”, samt egna åsikter och slutsatser kring det resultat som framkommit av studien. Vi ämnar även belysa de upptäckter vi gjort och ge förslag på vidare forskning.

Vi påbörjade studien med den gemensamma uppfattningen att vår utvalda teori inte kommer täcka alla faktorer som förekommer i praktiken. Bakgrunden till detta var att vi trodde att organisationer ofta har sina egna metoder för att förbättra genomförandet av IT-utvecklingsprojekt, och därför kommer det alltid finnas någon fas eller faktor som specifikt skiljer sig från organisation till organisation. Dessutom förväntade vi oss att teorin skulle utesluta viktiga faktorer, som oftast inte skrivs ner i till exempel en kravspecifikation eller analys. Dessa faktorer skulle då kunna vara sådana som innefattar personkemi eller tillit. En annan förutfattad mening vi hade var att större företag har svårare för att anpassa sig till det som senare kom att kallas för ”mjuka värden” och att de troligtvis utifrån detta föredrog en mer strikt metodform där varje steg planerats, innan IT-utvecklingsprojektet ens påbörjats. Vi kan med denna studie konstatera att det teoretiska ramverk vi presenterat, baserat på de mest återkommande faktorerna i vår utvalda teori, inte fullt ut täcker samtliga faktorer som existerar i praktiken. De faktorer vi identifierade i praktiken som inte var tillräckligt återkommande i vår valda teori var:

- Mjuka värden, och
- Tids- kostnads- och kvalitetsperspektivet.

Våra förutfattade meningar stämde då relativt bra överens med det som framkommit i vårt resultat. Det vill säga, de ”mjuka värdena” var något som i våra respondenters ögon var viktigt, men som inte nämndes i någon större utsträckning i teorin. Tids-, kostnads- och kvalitetsperspektivet förutspåddes dock inte, då vi mer eller mindre antog att dessa faktorer skulle ingå i såväl teorin som praktiken.

Mjuka värden och kravspecifikation var de två faktorer som enligt våra respondenter var absolut viktigast, och som borde betonas mer i IT-utvecklingsprojekt. Kravspecifikationen, till skillnad från de ”mjuka värdena”, var dock tillräckligt återkommande i teorin enligt vår urvalsmetod och ingick därmed i vår teoretiska referensram. En annan förutsägelse som vi hade var att större företag följer specifika metoder, och att de mindre företagen har större utrymme till att fokusera på de mjuka bitarna, men denna tanke visade sig enligt vår undersökning vara felaktig. De respondenter vi träffat från större företag, nämner ett flertal gånger hur viktiga de ”mjuka värdena” är, vilket även bekräftats av de mindre organisationer som ingått i vår studie. De större organisationerna verkar inte heller ha ett strikt arbetssätt utan anpassar sig efter kunden, vilket som vi tycker är överraskande positivt. Vi antog att ett större företag inte kunde anpassa sig till kunder i lika stor utsträckning som de mindre, då de borde vara tvungna att ha ett gemensamt arbetssätt, för att kunna skapa en enhetlighet inom organisationen.

Enligt den här undersökningen är med andra ord de mest betydelsefulla faktorerna kravspecifikation och mjuka värden. Kravspecifikationen är något som i teorin endast nämns och beskrivs som metod, det förklaras dock inte varför den är viktig, eller hur stor påverkan den kan ha på ett IT-utvecklingsprojekt. Respondenterna lyfter däremot upp problematiken som uppstår, i de fall då parterna inte fastställer kravspecifikationen på ett korrekt sätt. Logiken är ju ganska slående, det agila arbetssättet används främst för att försäkra sig om att det arbete leverantören lägger in är rätt, samt stämmer överens med kundens efterfrågan och på så vis skapar ett större förtroende för leverantören. Därmed kan slutsatsen dras att kravspecifikationen är viktig av samma anledning, då leverantören måste försäkra sig om att den har fått tydliga instruktioner på det som kunden efterfrågat. Även nära kundkontakt är något som verkar vara viktigt av samma anledning som kravspecifikationen, det vill säga att leverantören vill försäkra sig om att denne tolkat kunden rätt.

Baserat på det resultat vi fått i vår undersökning kan vi dra slutsatsen att de organisationer som levererar informationssystem värdesätter förståelse för kunden, även om de valt olika faktorer som mest kritiska. Det intressanta här kan tyckas vara varför just "kravspecifikation" valts ut. Ett agilt arbetssätt som genererar förståelse mellan kund och leverantör kan ju till exempel användas under utvecklingsstadiet, för att försäkra sig om att allt är korrekt, genom att till exempel använda sig av prototyper för att kunden ska känna sig säkrare. Ett svar på detta kan vara att kravspecifikationen symboliserar förståelsen för kundens begäran, och är en skriftlig, säker tolkning mellan kund och leverantör. När ett system levereras och det inte möter de krav som ställts av kunden, är det då möjligt att gå tillbaka, och i och med detta steg se hur en specifik funktion definierats. En prototyp som fyller en funktion kan godkännas av kunden även om den inte testats till 100 %. Låt oss, som ett enkelt exempel, säga att en knapp bör vara blå ur designsynpunkt, men kunden har bestämt sig och vill ha den i gult. Kunden i fråga kan missa denna knapp helt vid testandet och godkänna den, för att sedan först vid leverans inse felet och då lämna ett klagomål. I detta läge kan leverantören hänvisa till kravspecifikationen, och säga att knappens färg fastställdes till gult på kundens begäran.

Utöver detta kan vi till övervägande del se att faktorerna som vi behandlat är överlappande. Detta gör det svårt att skapa en tydlig avskiljning i vissa fall. Exempel på detta är mellan iteration och kvalitetssäkring. När vi frågade våra respondenter hur de gjorde för att kvalitetssäkra så svarade många: "genom att hålla en nära kontakt till kunden och arbeta agilt". De säkerställde alltså att de analyserade kraven var korrekta, genom att ständigt stämma av med kunden. Vi kan därför egentligen inte säga att mjuka värden var viktigare än kvalitetssäkring, eller att kravspecifikation är viktigare än prototyputveckling, då dessa till stor del kompletterar varandra. Det verkar som att ett arbetssätt som innefattar "mjuka värden" står som gemensam nämnare till att övriga faktorer ska lyckas.

Aspekterna tid, kostnad och kvalitet, togs inte självmant upp av någon respondent. Däremot fanns där en underton till att dessa fanns i åtanke när de förklarade deras arbetssätt. Vi frågade således om de menade att de implementerade projekttriangeln (de tre aspekterna tid, kostnad och

kvalitet) i sina projekt, vilket visade sig enligt dem vara självklart och dessutom ganska viktigt. Exempelvis nämnde respondent A att projekttriangelns aspekter är en mognadsfråga från kundens sida. En mogen kund kan prioritera rätt och förstå att exempelvis tiden och kostnaden inte är lika viktig som att alla de funktioner som fastställts har tillgodosetts, eller tvärt om. Återigen kommer vi tillbaka till de mjuka värdena. Det handlar alltså om att personen som tar de viktiga besluten, måste våga kompromissa med leverantören om vad som kan prioriteras bort. Frågan är hur leverantören bör gå till väga för att kringgå ett sådant hinder, och få kunden att förstå hur den bör prioritera. Här tror vi att skapandet av tillit är det bästa sättet. Om kundens förtroende är vunnet kan exempelvis oförutsedda kostnadsökningar och justeringar i systemet, troligtvis accepteras utan större klagomål.

Vi tror att dessa faktorer (mjuka värden och tids-, kostnads- och kvalitetsperspektivet) som lyfts fram av vår studie är två viktiga byggstenar som bör ingå mer detaljerat i de teorier som hanterar IT-utvecklingsprojekt. Denna slutsats baserar vi på vårt resultat som pekar på att det är otroligt viktigt med en bra kundkontakt, och att vetskapen om vad som kan hända om den skulle försummas är avgörande för ett lyckat resultat vid IT-utvecklingsprojekt.

En annan viktig aspekt att ha i åtanke och som vi tog upp kort i bakgrunden, är att det system som kunden "vill ha" inte alltid är vad denne behöver. Kunden vet oftast vilken information som behövs, men inte vilka funktioner som bidrar till att rätt information kommer tillhanda. Därför tror vi att det är viktigt att som leverantör lägga stor vikt på att skapa en nära kontakt med och god förståelse för kunden, vilket är avgörande för att lyckas möta kundens önskemål på bästa möjliga sätt. Vi anser självfallet utöver detta att båda parter har ett likvärdigt ansvar för att ett IT-utvecklingsprojekt fullföljs, och når ett positivt resultat. Dock slutar det oftast med att leverantören har det största ansvaret, för att den produkt som ska levereras stämmer överens med kundens krav och förväntningar.

Vårt syfte med den här undersökningen var att utreda huruvida teorin, om hur leverantörer och kunder bör utföra ett IT-utvecklingsprojekt för att uppnå ett tillfredsställande resultat, stämmer överens med det praktiska arbetet. Detta svarar vi på genom att förklara att teorin är väl utvecklad men brister på dessa två punkter (mjuka värden och tids-, kostnads- och kvalitetsperspektivet) som framkommit genom studien.

Sammanfattningsvis kan vi säga att vår studie påvisat att leverantörer i praktiken, följer de teoretiska rekommendationerna i stor utsträckning. Vårt att notera är dock att de teorier och modeller som blev utvalda att ingå i vår studie, långt ifrån täcker alla viktiga faktorer som kan inkluderas i ett IT-utvecklingsprojekt. Det är därmed svårt att ge ett konkret svar på huruvida den allmänna teorin följs eller inte. Vi kan bara ge ett svar utifrån de modeller som vi valt att studera.

En fråga som uppstår såhär i efterhand är varför de nya faktorer som framkommit i vår studie inte var inkluderade i vår utvalda teori? Då syftar vi speciellt på mjuka värden. Kan det vara så att faktorer som exempelvis mjuka värden är svåra att mäta, och därmed även svåra att

dokumentera? En annan tanke som dök upp var en fundering kring i vilken utsträckning det är leverantörens bristande engagemang och sätt att följa teorin som leder till att ett så stort antal IT-utvecklingsprojekt misslyckas? Dessa funderingar är något som vi tycker vore intressant att undersöka närmare, och därmed skulle vilja se behandlas i en framtida forskning.

# 1 Intervjusammanställningar

## 1-A

A inleder och förklarar att A jobb i dagsläget är att arbeta som konsult. Han sysslar främst med intranät och delning av information i programmet SharePoint. Dock kan A själv inte utveckla eller skriva kod, utan A jobb är mer inriktat på de mjuka bitarna, vilket är det han själv anser vara viktigast.

### **Kan du ge en kort beskrivning av ditt företag?**

Företag A är ett Kanadensiskt företag, med ca 70 000 anställda med verksamhet i över 40 länder. Främst i Nordamerika och västra Europa, men finns även i Indien.

### **Vad har du för erfarenheter av IT-upphandlingar?**

A förklarar att det vara många år sedan har själv var involverad i en upphandling. Nu går A arbete främst ut på att sitta på andra sidan, och svara på upphandlingar.

### **Kan du beskriva hur en IT-upphandling brukar se ut?**

Upphandlingsprocessen hos företag A brukar vanligtvis börja med att företaget erhåller en förfrågan. Denna förfrågan kan vara både muntlig och av mer formell karaktär. A förklarar att ju högre ordervärde upphandlingen har, desto mer formell är den. Dock går alla förfrågningar igenom en intern process hos företag A. Den interna processen varierar mycket men återigen, desto större ordervärde, ju mer formell är processen. A förklarar att den interna processen är en slags kvalificeringsprocess där företaget beslutar om de ska svara på förfrågningen eller inte. företag A svarar per automatik på inkommande förfrågningar. Det har mycket att göra med vem kunden är, om det går leverera med kvalitet och med lönsamhet. företag A gör även en omfattande riskbedömning där man kartlägger om det som efterfrågas går att leverera med lönsamhet. Ju högre risken bedöms vara, ju mer omfattande blir riskanalysen.

Företag A tittar på flertal faktorer, exempelvis hur kunnig kunden är, och om företag A själva har den kunskap som krävs. A menar att företag A är otroligt duktiga på att leverera att att företaget har en lönsamhet på 95 % för sina projekt, vilket är betydligt högre än branschstandard. Han förklarar att det inte enbart handlar om själva upphandlingsarbetet, utan att kontinuerligt mäta om projektet kommer bli lönsamt eller inte.

Vid större förfrågningar kan kravspecifikationen vara flera hundra sidor lång. Vid dessa större projekt arbetar företag A med ett stort team på 40 personer från olika delar av världen och med

varierad kompetens. Vid dessa större affärer berättar A att det verkligen är viktigt att analysera riskerna, för att se om det är lönsamt.

Det förekommer även förfrågningar från kunder som företag A tidigare inte arbetat med, och då är det väldigt sällan man väljer att svara på dessa. A förklarar att man vanligtvis svarar på anbud från kunder man har en god relation till, eller kunder man vill utveckla en relation med.

### **Har företag A något speciellt ramverk/modell de arbetar utifrån**

Företag A arbetar utifrån sitt eget ramverk och processer. Detta ramverk går kortfattat ut på att en förfrågan går igenom ett antal ”gater”. A är dock för dåligt insatt i vilka dessa ”gater” är.

Ramverket består av mycket kalkylerande, både vad det gäller risker och lönsamhet. Vidare bygger företag A sin verksamhet utifrån tre grundpelare: sina kunder, anställda och aktieägare.

### **Vilka faktorer tycker du är viktigast vid en IT-upphandling?**

A uppfattning är att en av de viktigaste faktorerna är delaktiga och matchar upp väl från sin sida. Kunden måste vara beredd att vara involverad i projektet och att de är öppna med vad de vill ha och uppnå. Även relationen med kunden är en faktor som är viktig för att projektet ska bli lyckat. A förklarar att affärer trots allt handlar om människor, och man måste känna att man kommer överens. Ett vanligt misstag enligt A är att man enbart tittar på de hårda bitarna, som t.ex. avtal och pengar, men de mjuka delarna mycket viktigare menar A. Det finns då en fördel att arbeta med återkommande kunder, på grund av att kunden känner till företag A och företag A känner till kunden.

### **Vilka faktorer anser du är mindre viktiga vid en IT-upphandling?**

A kommer inte på något, utan poängterar återigen öppenheten och de mjuka delarna.

### **Anpassar företag A sina projekt efter tid, kostnad och kvalitet?**

A menar att vissa kunder har väldigt tydliga prioriteringar. Det handlar om en mognadsfråga. Omogna kunder brukar ha svårt för att prioritera. Däremot de mogna kunderna oftast ärliga och kan erkänna att t.ex. ett visst datum är livsviktigt. När tidsaspekten är det centrala brukar det vara OK att lägga mindre vikt vid systemets funktionalitet. Det förkommer även projekt där kostnaden är det viktigaste. Exempelvis kanske kunden enbart har ett visst belopp att röra sig med, men brukar då vara mer flexibla angående tid och funktionalitet.

### **Om en IT-upphandling skulle vara misslyckad, vad tror du anledningen är?**

A klargör att kommunikationen är livsviktig. Även att scopet är otydligt och att kunden inte har en klar bild över vad de själva vill ha. A tar även upp att olika förväntningar mellan kund och leverantören lätt få ett projekt att haverera.



### **Finns det fall där ni själva går in och utvecklar en kravspecifikation?**

Då företag A mest sysslar med större projekt är detta mycket ovanligt. A tar upp ett generellt problem med att arbeta med större kunder: De är ofta väldigt tydliga i vad de vill ha, men är inte beredda att betala för det. Det är en omogen bransch på det viset. Kunden måste själva lägga ner ett gediget arbete och analysera anbudet.

### **Hur arbetar ni för att försäkra er om att kunden kommer att bli nöjd med er produkt?**

Här förklarar A hur allt går tillbaka till det iterativa förhållandet till kunden, att man inte jobbar avskilt med ett projekt och sedan presenterar resultatet för kunden i ett slutgiltigt tillstånd, utan hela tiden arbetar tillsammans för att säkerställa att man utvecklar något som kunden är nöjd med.

A förklarar även att det finns olika sätt att göra detta på, ett sätt är att företaget A:s projektledarroll arbetar nära med kundens beställare och då kan känna av hur relationen mellan båda parterna är. Ett annat sätt som är lite mer formellt, är att ha en styrgrupp av beslutsfattare från båda organisationerna som känner av hur kvaliteten av produkten är.

### **Använder ni er utav prototyper när ni utvecklar system?**

A menar att det beror på vilket typ av projekt man jobbar med. Är det specifikt Intranät man utvecklar så är prototypdemonstrationer en väldigt effektiv metodik, att man jobbar agilt. Den agila metoden är enligt A det mest effektiva sättet att hantera denna typ projekt. A menar att detta gör det lättare att revidera något som skett tidigt i ett senare stadie av utvecklingen.

A förespråkar även att man börjar med demonstrationer så tidigt som möjligt, och sedan ”bygger en story” och gör justeringar allt eftersom. På detta sätt kan man tidigt få synpunkter på vad som är rätt/fel eller fattas. A förklarar att kunden känner en större delaktighet om man jobbar på det här sättet.

### **Utöver att använda sig utav prototyper, kan du förklara vad ni tänker på vid utvecklingsfasen?**

Öppenhet, är enligt A essentiellt även här. Att man är tydlig med att kunden verkligen förstår hur projektet utvecklas och vilka eventuella motgångar man stött på. Ett bra exempel är testning, att man är ärlig med vilka buggar som systemet innefattar just nu, och hur man tänkt gå till väga för att åtgärda dessa. Återigen nämner A hur viktigt det är med att ha en bra relation med kunden.

### **Vi märker ganska tydligt att det Iterativa är det viktigaste i era projekt**

Det beror helt på projektet menar A. Pratar vi Intranät så instämmer A i det vi tar upp. Att kunden är mogen för att använda en agil metodik med hög grad av öppenhet och ärlighet. A poängterar att det tyvärr inte är så många organisationer som jobbar på detta sätt. Det kräver helt enkelt mer av kunden.

### **Vilken form av validerings/verifieringsprocess använder ni?**

Acceptanstestning är då väldigt viktigt, enligt A. Att kunden hela tiden får testa av systemet så att de känner att det är på väg i rätt riktning. Att hela tiden arbeta med demonstrationer så att de får se och testa något konkret är nyckeln. Dessutom ger det kunden ökad möjlighet att påverka utvecklingen. Och så kan man avsluta med att köra acceptanstester där kunden får testa hela systemet och ge en bekräftelse på att detta verkligen är något som efterfrågades. A förklarar att de projekt som blir bra är de som körs iterativt där man har nära kontakt med kunden.

### **Återkopplar ni med kunden efter det att valideringen av ert projekt är gjort?**

Många gånger sitter man alltså ner med kunden och går igenom vad som finns kvar att göra, och vad man alltså valt att prioritera bort innan dess att systemet går ”live”. Dessutom testar man internt hela tiden för att kunna ta fram allt som möjligen kan gå fel med systemet.

### **Tycker du att dagens generella modeller för utformning av intranät fungerar bra?**

Det viktigaste man bör tänka på, oavsett modell är att man involverar kunden, och tillsammans kan gå igenom de problem som uppstår under projektets gång.

### **Underlättar det om kunden är IT-kunnig?**

Både ja och nej. Det är en mognadsfråga. A förklarar att man får ibland ha kunden som säger ”vi behöver ett intranät, hjälp oss” och ibland kunder som specifikt vet vad de vill ha. Men i slutändan handlar det hur väl kunden är beredd att samarbeta och kommunicera.

### **Finns det ett generellt system från er sida som ni använder er utav vid dessa projekt?**

Nej, oftast inte. Våra projekt struktureras upp individuellt, svarar A. Däremot berättar han att från kundens sida finns det mer specifika steg man följer vid en sådan här upphandlingsprocess. Desto större affär desto striktare regler och mål sätts upp från företag A:s sida, man matchar alltså kundens behov. A lägger till att relationen är mycket viktigare än själva metodiken.

### **Vill du tillägga något?**

Ja, det är viktigt att inte förringa de mjuka värdena i en upphandling, alltså relationerna. Respekt, tillit och förståelse är enligt A nyckeln för ett framgångsrikt utvecklingsprojekt. Dessa faktorer är svåra att kvantifiera i skrift, men är ändå viktigare och man bör aldrig overse dem.

## 1-B

B inleder med att presentera sig själv och sin befattning. Hon jobbar inom Corporate-IT och hennes roll är som platsformsansvarig för SharePoint. Denna plattform innefattar företagets Intranät, e-business och samtliga www-sajter världen över.

### **Om vi säger IT-upphandlingar av Intranät, vad tänker du på då?**

Enligt B handlar en IT-upphandling mycket om skapa en affärsförståelse och kartlägga de bakomliggande behoven, snarare än renodlad teknik. Hon förklarar att projekt lätt kan misslyckas p.g.a. att man fokusera för mycket på en specifik plattform eller applikation som ska användas. B menar att om man inte förstår affärsbehoven, spelar det ingen roll vilken plattform och teknik som används. B betonar att det centrala är att så tidigt som möjligt identifiera de faktiska affärsbehoven. Detta arbete tar mycket tid, men det är en förutsättning för ett lyckat resultat. Upphandling av Intranät handlar inte enbart om att leverera ett system menar hon. Hon betonar återigen att skapa en helhetsbild, och involvera de personer som i slutändan ska använda systemet. Detta kan vara svårt menar B, då man ibland endast levererar delar av ett system.

### **Kan du beskriva hur de olika stegen som en IT-upphandling innehåller, ur leverantörens synvinkel?**

Som IT-leverantör finns det ett generellt problem redan i inledningen av en upphandling menar B. Hennes uppfattning är att det är extremt viktigt att inte enbart arbeta med IT-avdelningen i kundorganisationen, utan även involvera samtliga affärsområden. B tar exempel med HR-avdelningar. Enligt henne att dessa avdelningar otroligt viktiga att involvera, men att det oftast glöms bort, då man fokuserar för mycket på renodlade IT-aspekter.

B byter spår och kommer in på kravspecifikationer. Hon förklarar att kravbilderna vanligtvis kommer från IT-avdelningen i kundens organisation som tidigt har kartlagt affärsbehoven. Dock kan denna kravbild landa snett hos leverantören.

Bs erfarenhet från tidigare projekt är att det oftast går att göra IT-avdelningen hos kunden nöjda, men när systemet rullas ut, är det inte det kunden förväntade sig. Felet uppstår sällan direkt menar hon, utan när systemet börjar användas, och då är det oftast för sent att göra något åt det. Hon tar även upp att en kravspecifikation ibland kan vara ofullständig. Kunden har kanske tagit med "A till K" men resterande delar finns inte med. Då gäller det att förklara för kunden att de inte kommer få ut full effekt av systemet. Då gäller det att leverantören kommunicerar och uppmana kunden även inkluderar "K till Ö". B menar att det handlar mycket om att hitta kringliggande behov och involvera samtliga affärsområden. Hon understryker att den kravbild som de vanligen erhåller oftast inte är den som skickas tillbaka till kunden. Kravbilderna diskuteras ett flertal gånger och man träffar kunden för att analysera vad de verkligen vill ha. Dessa möten leder till att båda parter får en tydlig bild av det som ska levereras. B förklarar att

den offert som skickas tillbaka oftast är betydligt mer omfattande än den första versionen. Enligt henne är kunden inte speciellt skicklig på att fastställa den totala kravbilden. Kunden har ofta en klar bild av IT delen av systemet, men övriga aspekter, och hur systemet verkligen ska användas glöms bort.

IT upphandlingar handlar enligt B minst lika mycket om affärsbehov, som det politiska spelet. Det vill säga, vem man ska prata med vid rätt tidpunkt. Det gäller att få rätt mandat med sig.

### **När det gäller utvecklingsdelen av upphandlingen, finns det något specifikt ni tänker på då?**

B förklarar att man främst jobbar utifrån agila utvecklingsmetoder. Denna arbetsmetodik går ut på att bryta ner samtliga funktioner i mindre paket som man utvecklar var för sig. Detta möjliggör konstant feedback menar B. Mer specifikt innefattar arbetet 4-8 veckors "sprintar". B tror att detta arbetssätt skapar ett känsla av lugn i kundens organisation. De känner att det hela tiden händer någonting och att utvecklingen går framåt.

Gällande kravspecifikationen diskuteras det ofta på vilken nivå den ska vara. Exempelvis om kravbilden ska innefatta affärsprocesser, rika bilder osv. B anser fortfarande efter 15 år i branschen att det är svårt att fastslå vad en kravspecifikation faktiskt ska bestå av. Självklart vill man ha en målbild klar, men samtidigt är det inte den färdiga målbilden av produkten. Hennes erfarenheter är att kravbildens detaljnivå och omfattning skiljer sig väldigt mycket från kund till kund.

### **Hur sker den slutliga valideringen?**

B berättar att man jobbar med 6 veckors iterationer, för att sedan demonstrera systemet i bildmiljö för kunden. Demonstrationerna upprepas sedan var sjätte vecka. Vid större projekt genomförs det upp till fyra releaser på ett år.

### **Vem är det som beslutar om detta upplägg? Är det kunden eller ni som leverantör?**

Hon förklarar att det oftast är leverantören som tar dessa beslut. Det gäller att kunden är beredd att acceptera detta arbetssätt om de vill inleda ett samarbete.

### **Om man jämför de olika traditionella stegen: Analys, kravspecifikation, design, prototyper, utveckling och verifiering. Vilket av dessa anser du vara mest kritiskt?**

Första steget, dvs. analysarbetet är enligt B viktigt. Hon förklarar att det är viktigt att verkligen att få det rätt från början och vara tydlig med vad man vill uppnå, rätt kravnivå helt enkelt. Just användartesting är extremt viktigt i detta sammanhang och involvera rätt användargrupp i arbetet. En korrekt kravspecifikation är essentiellt!

### **Om vi tar valideringsprocessen. Är det något som är mer strikt på någon av sidorna (Kund-Leverantör)?**

B klargör att man sammanställer timmarna i ett projekt så är utvecklingstiden mycket liten om man jämför med den tid som läggs på test. Hos oss läggs stor vikt vid validering förklarar hon, att få systemet testat. Företaget använder något som kallas "testdriven-development" vilket är ett agilt arbetssätt. Allting byggs i en utvecklingsmiljö och inte förens man har uppnått 60 % av alla testcase godkänns den första delen. Efter det går man vidare till något som kallas "pre-production" som är en form av valideringsmiljö. För att sedan gå vidare måste 80 % av alla testcase vara utförda och slutligen 100 % innan det går ut i produktion. Detta är extremt strikta krav och ingen produkt får gå direkt till livemiljön, utan att ha gått igenom samtliga nivåer.

### **Har leverantören något ansvar efter "live utrullningen"?**

B berättar att man oftast har en garantiperiod på 6 månader. Här försäkras att systemet rullar, vilket oftast är det mest kritiska momentet. Detta brukar finnas med i den kontraktuella delen av upphandlingen.

### **Anser du att det finns något annat steg än de vi nämnde tidigare som du anser är viktigt?**

Att involvera användargrupperna genom hela processen är verkligen viktigt menar B. Hon förklarar att bara för att man börjar arbetet med något dag 1, så behöver det nödvändigtvis inte leva med genom hela projektet. Denna input får man från kunden och då gäller det att ha en tydlig dialog.

B tar ett exempel från leverantörens sida. Hon menar att man ofta får en omfattande kravspecifikation som kunden lagt enormt mycket tid vid. Kunden är då oftast trötta och lägger över hela ansvaret på leverantören. Detta funkar inte speciellt bra. B anser därför att det centrala är att leverantören involveras i ett mycket tidigare skede och utvecklar kravbilderna tillsammans med kunden. Inte bara att den slängs över bordet. Kunden måste verkligen ta sitt ansvar och inse att de måste lägga mycket tid för att projektet ska bli lyckat.

Hon tar även upp kombinationen av det agila arbetssättet med att ha en fysiskt nära kontakt med kunden. Detta möjliggör enligt B att bolla idéer och ta snabba beslut. Brister i kommunikationen är det dömt att misslyckas säger hon. Även att vara tydlig med prisbilderna från början är en kritisk faktor. Man budgeterar enbart för ett projekt, men glömmer ofta bort efterarbetet som drift, förvaltning och support.

### **Om vi ser det från ett leverantörsperspektiv. Är det lättare om kunden är IT-kunnig eller om de endast är medvetna om att de behöver ett intranät.**

Utan att tveka väljer B det sistnämnda. Hon menar att om kunden har mycket IT-kunskap kan det ibland bli för detaljerat och orimliga krav. Hennes erfarenheter är att det mest optimala är att låta leverantören agera expert kring teknikfrågor, och låta kunden koncentrera sig på sin verksamhet

och förstå vilka behov som finns. En viktig fråga att ställa sig som leverantör är ”Vem äger Intranätet i kundorganisationen?” Är det HR, kommunikation eller produktion? Detta är väldigt olika men det är de som är den primära kunden förklarar B.

**Om vi tar den generella modellen som finns ute just nu. Tycker du att den fungerar bra? Eller finns det något som ofta går fel?**

B tror att man generellt är för stelbent när man diskuterar analys och kravspecifikationer. Dessa delar ses ofta som två helt isolerade saker som ska resultera i ett perfekt underlag som man inte behöver hantera senare i upphandlingen. Problemet enligt B är att man ofta inte inser att dessa två faser behöver växa med under tiden. Man ska hellre prata om affärsbehov än krav menar hon.

**Om du jämför rollen som leverantör och kund. Var i processen brukar man bli oense?**

B börjar på kundens sida. Hon tar upp att kunden ibland kan ha svårt att inse hur ett system kan kosta så mycket som det gör. Kunden inser inte hur mycket som faktiskt ska finnas på plats. Du behöver ha en bra struktur, hantera profiler och allt runtomkring. Även validering och testning är inbakat i prisbilden, men kunden ser ofta inte denna helhet. B menar att kundsidan ofta har en naiv bild av det hela. Att rulla ut ett Intranät är inte en traditionell IT-utrullning menar B. Det ska finnas jätte mycket med på banan. Ofta slängs systemet bara över och det är ingen som kan hantera det. Detta har egentligen inte så mycket med upphandlingen att göra förklarar hon. Det måste finnas ett ansvar från kunden, att man själva kan hantera systemet när det levererats.

B byter spår och förklarar att det måste finnas en skicklig business analys med i processen, men det är ofta en del som prioriteras bort. Man väljer istället att satsa på traditionella utvecklare och någon som driver projektet. Men det som pågår där emellan tas ofta bort. Det måste finnas en brygga mellan utvecklare och projektgruppen menar hon. Kunden tror traditionellt att man kan sköta detta internt för att spara pengar, men det funkar sällan i praktiken. Ett annat problem som hon ser är att leverantören inte får tillgång till de personer som faktiskt ska använda intranätet. Kommunikationen sker endast med IT-avdelningen.

**Hur tror du man kan säkerställa att kunden blir nöjd?**

Nyckeln är att arbeta agilt berättar B. Agila metoder möjliggör kontinuerliga avstämningar och kunden kan då se utvecklingen på kontinuerlig basis. Även att beställaren är tydlig och är väl förberedd är viktigt. Exempelvis att man har en klar bild över vad man vill ha och vilka aspekter som exkluderas.

**Du känner säkert till den traditionella projektriangeln. Är det något som man bör ha i åtanke tycker du?**

Ja absolut, och detta är något som kan kopplas till det agila menar hon. Du får då hela tiden tänka på vad man ska göra först, och vad som är viktigast.

### **Är det något som skiljer upphandlingar av Intranät från traditionella system?**

B understryker att Intranät är inget man enkelt rullar ut. Dessa system kräver konstant arbete för att hela tiden ha koll på vad som är aktuellt och när man ska implementera nya applikationer. Detta är ett dagligt arbete som ofta glöms bort menar hon. Det kunden får levererat vid en viss tidpunkt är kanske inte det de behöver om 6 månader. B nämner även att det är viktigt att veta var man ska dra gränsen. Var börjar och slutar ett Intranät? Är det enbart nyheterna eller alla lokala applikationer? Detta är en vanlig fälla som är jätte svår att hantera säger B. Redan vid analysen måste man sätta ramarna för var systemet börjar och slutar. B vill undvika ordet kravspecifikation i detta sammanhang och väljer istället att tala om behovsanalys. Vad är det du faktiskt vill kunna göra, men även vara tydlig med det man vill exkludera.

B avslutar med att återigen poängtera att kundens engagemang i det hela är nyckeln till framgång.

## 1-C

C inleder intervjun med att berätta att hon jobbar på företag i ett års tid. C började som utvecklare men har nu gått vidare till att jobba mer med projekt och kundansvar. Idag fungerar C som ansiktet utåt. C tjänstebeskrivning är SharePoint specialist med projekt och kundansvar.

### **Kan du ge en kort beskrivning av företaget?**

C berättar att företaget arbetar med SharePoint i dagsläget. Men företaget bygger även portaler och intranät till deras kunder. Kunderna varierar väldigt mycket, exempelvis kan en kund vara en organisation med endast 20 anställda, till kommuner och större företag med 1000-tals anställda. Dock är utgångspunkten alltid SharePoint. C sammanfattar företagets arbete i att det går ut på att analysera kunden och kartlägga var behoven finns, för sedan kunna anpassa produkten efter deras behov.

### **Vad har du för erfarenheter av IT-upphandlingar?**

Företagets upphandlingar brukar se väldigt olika ut pga att man jobbar med olika typer av kunder. Exempelvis fungerar offentliga upphandlingar mycket annorlunda. Då måste anbudet gå ut till ett visst antal för att det ska kunna ske en upphandling. Hos de större kunderna har det även gjorts förstudier på exempelvis deras befintliga intranät. Då har kunden själva analyserat deras verksamhet och anställda. De tittar på hur systemet används idag och vilka brister som eventuellt finns. Denna information är sedan underlaget för den kommande kravspecifikationen. C förklarar att det ofta finns en förstudie som grund och att man tillsammans med kunden utvecklar en kravspecifikation och arbetar utifrån den.

### **Kan du ge exempel på hur en upphandling brukar gå till?**

C framhåller återigen att just förstudien och kravlistan nästan alltid kommer från kunden själv. C ger ett exempel där en kund har tittat på SharePoint och insett att det är precis vad de behöver. Ofta används redan en tidigare version av SharePoint och de vill uppdatera till den senaste versionen. Då sammanställer kunden ett antal krav som de lämnar över. C tar upp att dokumenthantering är ett vanligt återkommande problem i denna fas av upphandlingen. Även ägare av informationen brukar vara ett problem som måste lösas, exempelvis att det finns inaktuell och irrelevant information i deras gamla intranät. Om kunden är en mindre organisation tar de ofta första kontakten och ber om ett möte där det diskuteras vilka möjligheter SharePoint har i att lösa de problem som kunden har i dagsläget. I dessa fallen görs ingen förstudie, utan kunden kommer direkt till oss och vi diskuterar tillsammans dessa möjligheter. C menar att de mindre företag kan ha svårt att själva sammanställa en kravspecifikation eftersom det kan saknas kompetensen finns internt i organisationen.



### **Ställer ni några krav på kundens IT-kunskaper?**

C förklarar att de inte ställer några krav på kundens IT kunskaper. C menar på att ett samarbete underlättas genom att kunden på förhand har en klar bild över vad de vill ha. Men det finns även fall där kunden saknar kompetens och själva erkänner att de inte förstår och ber då om hjälp och utbildning. C menar på att inget av dessa två fallen är att föredra, utan att personkemin är det centrala. Allt bygger på en bra kommunikation och förståelse.

### **Förekommer det ett speciellt ramverk/modell som ni utgår ifrån när ni får ett jobb?**

C beskriver att de arbetar utifrån en egen metod som utgår ifrån agila utvecklingsmetoder, exempelvis liknande lösningar. Det agila arbetet blir aktuellt när kravbilden är sammanställd och utvecklingsarbetet ska börja. C poängterar att återkoppling med kunden görs kontinuerligt genom hela processen.

### **Vilka faktorer tycker du är viktigast i upphandlingsprocessen?**

C berättar att återkoppling med kunden är det absolut viktigaste faktorn i hela upphandlingsprocessen. C menar att återkoppling gör det möjligt att hantera och korrigera fel i exempelvis kravbilden eller om det upptäcks designfel. Återkoppling minskar risken för fel helt enkelt.

### **Vilken faktor tycker du är mindre viktig i upphandlingsprocessen?**

En för detaljerad kravspecifikation brukar aldrig resultera i ett bra resultat förklarar X. En för specificerad kravlista kan oftast bli ett hinder eftersom kunden alltid ändrar sig efterhand.

Gällande om eventuellt efterarbete och underhåll ska erbjudas så fastställs det mycket tidigt i upphandlingsprocessen. C menar att efterarbete brukar förekomma, exempelvis vid lansering och utrullning av systemet.

### **Har ni någon form av återkoppling med kund efter genomfört arbete?**

Efterarbete, utbildning och underhåll av systemet genomförs om det ingår i upphandlingsavtalet, med andra ord bestäms detta redan från början om det ska ingå och hur omfattande den ska vara.

Efter lansering ges en utbildning, om det inte redan finns personal som har den kompetensen, då behövs inte detta, mm. Allt detta brukar stå specificerat i kravspecifikationen, respondenten C känner sig dock lite tveksam på just denna bit.

### **Om en IT-upphandling skulle anses vara misslyckad, dvs kunden får inte det den ville ha - var i upphandlingsprocessen tror du att misstaget generellt sätt sker? Varför?**

En IT-upphandling misslyckas oftast när det kommer till förankringen av systemet i organisationen. Ex. någon har beställt ett intranät, då bildas en projektgrupp av olika delar i organisationen, dessa ställer sedan upp krav i form av kravspecifikation och, eller

systemspecifikation, därefter utvecklas systemet som senare stannar av och förblir oanvänt på grund av funktionalitetsbrist.

Detta beror oftast på att någon annan än användaren "HR", exempelvis IT-avdelningen, ställer upp kravspecifikationen efter antaganden om förmodliga arbetsuppgifter. Användaren (en läkare) gillar inte systemet, på grund utav att det saknar funktioner alternativt har fått överflödigt med funktioner som inte behövs, detta skapar i sin tur en missnöjdhet då användaren inte känner att han har fått säga sitt och nu står med ett system som inte går att använda på grund utav funktionalitetsbrister.

Utöver misslyckad förankring av systemet kan egenframställda kravspecifikationer, från kunden, vara ett stort problem, säger respondent C.

### **Hur går ni tillväga när ni analyserar de eventuella problem som kan finnas i kundens verksamhet?**

Frågar kunden och ställer frågor till kunden för att få fram vad det är som de behöver, samt vad de tror sig behöva, svara respondent C.

### **Hur arbetar ni för att försäkra er att kunden är nöjd med er produkt?**

Kundåterkoppling ger företag C en försäkran om kundnöjdhet.

### **Använder ni någon form av prototyper vid ert utvecklingsarbete av systemet?**

Ja, Prototyper används! säger respondent C.

### **Hur går utvecklingen av systemet till?**

Först bryts kravspecifikationen ner i mindre sjok, som sedan läggs in i en planering, för att sedan brytas ner i ännu mindre delar så att vi hamnar på aktivitetsnivå, berättar respondent C.

Exempelvis: Vi har en funktion, som analyseras, för att se hur lång tid den lär ta att bygga upp. Tiden här utgörs efter vilka delar som behövs.

### **Använder ni någon form av validering/verifiering med kunden? För att försäkra att båda parter är överens.**

Företag C arbetar med leveransgodkännanden, vilket innebär att kunden skriver under att de fått vad de efterfrågat efter projektets avslutande - implementeringen. Detta sker även via återkoppling under arbetets gång för att säkerställa att det inte blir fel på vägen.

### **Kan du förklara hur de olika stegen i er upphandlingsmetod agerar iterativt med varandra?**

Iteration sker genom att kunden får verifiera prototyper efter varje iterering, och godkänna varje

delmoment utefter tester. Det sker även genom om något skulle vara oklart så kontakter man kund och skriver, om det så krävs, om kravspecifikationen.

**Anser du att någon ytterligare aspekt, än de vi nämnt här ovan, har stor påverkan för ett lyckat resultat vid en upphandling?**

Kravspecifikation, analysera, implementera, lansera, underhålla, iterera (testning kontinuerligt), leveransfas (testning och migrering av innehåll).

Gott samarbete mellan kund och leverantör, ett högt engagemang från kundens eller användarens sida, är något som måste finnas. Då det är just ett samarbete som man bygger upp för att kunna leverera intranätet tillsammans med kunden.

- De mjuka värdena saknas ett gott samarbetet mellan kund och leverantör är enormt viktigt och gör oftast hela arbetet.

- En design (arkitektturnivå) som är god är viktig för att undvika fel.

- (vi nämner)Tid, kostnad och kvalitet (scope) är viktiga bitar att ha i åtanke under hela upphandlingen.

**Fungerar dagens modeller/metoder vid upphandlingar?**

Företag Cs metod fungerar bra, då den liksom respondent C förespråkar det iterativa arbetet. Med en tät kundkontakt, går det fort att skifta arbetssätt och anpassa sig efter kunden.

Återigen nämner C hur det iterativa arbetssättet och nära kundkontakt är essentiellt för att kunna anpassa sig efter kundens behov.

## 1-D

### **Kan du börja med att berätta lite om dig själv. Vilken befattning har du i dagsläget?**

Se bifogat CV

### **Kan du ge en kort beskrivning av ert företag?**

Företaget är en oberoende tjänsteleverantör inom verksamhetsnära IT-tjänster och infrastruktur.

Vi bistår våra kunder med att säkerställa maximal effekt och nytta av de investeringar som görs inom IT och IT-infrastruktur. Detta uppnår vi genom att förena vår breda erfarenhet inom IT-styrning och ledningsarbete med vår djupa tekniska specialistkunskap.

Företaget grundades under 2006 och består av ett tjugotal erfarna konsulter. Med Örebro som utgångspunkt stöttar vi medelstora och stora organisationer i mellansverige.

Oberoende utgångspunkt i nyttan

I vårt oberoende, utan affärsmässiga bindningar till leverantörer eller produkter, utgår vi odelat från kundernas behovsbild. Vi bistår våra kunder med oberoende rådgivning samt att identifiera, utforma och införa ett kostnadseffektivt IT-stöd som bidrar till ökad verksamhetsnytta.

Det värde vi tillför avspeglar sig tydligast genom det höga förtroende som vi regelbundet åtnjuter från våra kunder. För oss är detta det bästa bevis vi kan erhålla på det värde våra tjänster ger våra kunder.

Bland våra kunder återfinns framgångsrika organisationer såsom Atlas Copco, Servera, Korsnäs, Ahlsell och Mediabolaget Promedia

### **Vilken erfarenhet har du av IT-upphandlingar?**

Se bifogat CV. Jag har drivit ett stort antal IT-driftsupphandlingar, några IT-produktupphandlingar samt någon Intranäsupphandling.

### **Kan du beskriva hur en IT-upphandling vanligtvis går till på ert företag?**

Först gör vi en analys så att vi vet vad kunden behöver köpa. Därefter jobbar vi fram kravställningar tillsammans med kunden som vi sedan skapar ett förfrågningsunderlag på. Detta skickar vi till tänkbara leverantörer. Under anbudstiden svarar vi på frågor om upphandlingen. Därefter tar vi emot offerterna. Vi läser igenom och utvärderar. Oftast erbjuder leverantörerna att presentera sina förslag så att vi kan ställa frågor så att vi förstår offerterna. Därefter utvärderar vi

färdigt och ger kunden vår rekommendation. Efter detta hjälper vi kunden med förhandling och avtalstecknande.

**Använder ni er av något speciellt ramverk/modell som ni utgår ifrån vid era upphandlingar eller varierar det från kund till kund?**

Vi har ett tydligt ramverk som vi alltid följer. Dock kan kunden välja att göra delar av det själv.

**Vilka faktorer i en IT-upphandling anser du är viktigast? Varför?**

Ett bra underlag som speglar behoven är absolut viktigast. Annars kan inte leverantörerna föreslå passande lösningar.

**Vilka faktorer i en IT-upphandling anser du vara mindre viktiga? Varför?**

Oj vilken svår fråga. Alla delar i vår process är viktiga.

**Om en IT-upphandling skulle anses vara misslyckad, dvs kunden får inte det den ville ha - var i upphandlingsprocessen tror du att misstaget generellt sätt sker? Varför?**

Dålig kravspec.

**Hur går ni till väga när ni sammanställer en kravspecifikation?**

Normalt jobbar vi i workshopformat med kunden

**Hur går ni tillväga när ni analyserar de eventuella problem som kan finnas i kundens verksamhet?**

Vi ställer mycket frågor till olika personer, gör okulärbesiktningar och analyserar dokumentation.

**Hur arbetar ni för att försäkra er att kunden är nöjd med er produkt?**

Vi gör uppföljningar internt efter 30%, 60% och 90% av uppdraget samt med kunden under, direkt efter avslutat uppdrag samt 6 månader efter avslutat uppdrag.

**Använder ni någon form av prototyper vid ert utvecklingsarbete av systemet?**

N/a

**Hur går utvecklingen av systemet till?**

N/a

**Använder ni någon form av validering/verifiering med kunden? För att försäkra att båda parter är överens.**

Vi gör uppföljningar internt efter 30%, 60% och 90% av uppdraget samt med kunden under, direkt efter avslutat uppdrag samt 6 månader efter avslutat uppdrag.

**Kan du förklara hur de olika stegen i er upphandlingsmetod agerar iterativt med varandra?**

**Vilken/vilka av ovanstående faktorer (kravspecifikation, analys av problem, kundnöjdhet, prototyper, systemutveckling, validering/verifiering, iteration) anser du är viktigast för att en upphandling ska bli lyckosam? Varför?**

Kravspec. Utan den finns absolut inga förutsättningar att kunden skall få det de behöver.

**Anser du att någon ytterligare aspekt, än de vi nämnt här ovan, har stor påverkan för ett lyckat resultat vid en upphandling?**

Det är viktigt att kommunicera förväntningar både internt och externt.

**Fungerar dagens modeller/metoder vid upphandlingar?**

Ja

**Krångliga att använda? Bra stöd? mm.**

**Användaren/kundens delaktighet beryktas vara viktig, vad anser ni?**

Stämmer absolut. Varför? Vi kan ju inte gissa vad som är viktigt för dem.

**Vad har ni för krav på kundens IT-kunskaper?**

Inga, vi kan hjälpa dem ändå

**Bra/ Dåligt att kunden har tidigare IT-kunskaper och vet vad de vill ha/ behöver till sin verksamhet?**

Det är viktigare att de törs ta ställning än att de vet vad de vill ha. Vi kan hjälpa dem att identifiera vad de vill ha bara de törs ta ställning till våra frågeställningar.

## 1-E

E inleder intervjun med att förklara att han jobbar på GBS(Global Business Services). Han ansvarar för ett antal större myndigheter och jobbar i stort sätt med alla sorts verksamhetstjänster, bland annat Intranät.

### **Vilka erfarenheter har du av upphandlingar?**

Gustav säger att han har arbetat med olika former av IT-upphandlingar sedan 2006.

### **Hur går en typisk IT-upphandling till enligt dig?**

E berättar att upphandlingen inleds med att man går igenom kundens förfrågan och kvalificerar den. Kvalificeringsprocessen går ut på att värdera kundens förfrågan och besluta om man ska svara på den. E förklarar att beslutet baseras på vad det för typ av kund, men även den totala storleken på affären. Han poängterar även att man analyserar vilken chans man har att ”vinna” affären. Detta går kort ut på att man jämför vilka konkurrensfördelar man själv besitter och ställer dessa i relation till andra aktörer på marknaden.

### **Kan du förklara stegen från kravuppsamling till utveckling?**

E anser att IT-upphandlingen från leverantörens sida avslutas när man fått uppdraget, det vill säga när utvecklingsdelen påbörjas. Utöver detta beror det på vilket sorts projekt man driver menar han. En konsulttjänst planeras oftast i timmar, medans mer tekniska projekt kan innefatta specifikt hur utveckling ska gå till. E tar ett exempel med just Intranät och menar att den tekniska sidan av lösningsförslaget är av största vikt och detta måste därför inkluderas i offerten/avtalet.

### **Vilken kritisk faktor tycker du är viktigast i upphandlingen?**

Här förklarar E att kravinsamlingen är något som oftast styrs av kunden, man får alltså deras uppfattning av vad som efterfrågas. Kravsamlingen är alltså en självklar faktor som leverantören måste leva upp till menar E. Själva principbyggandet är ganska viktigt då det är en faktor som skiljer sig från organisation till organisation. E berättar att konkurrensfördelar kan spela avgörande roll och blir därför en kritisk faktor. Detta resulterar i en korrekt kravspecifikation.

### **Var i en IT-upphandling uppfattar du att det sker mest fel?**

E anser att från kundens sida sker ofta felen i början. Kunden ställer helt enkelt fel krav då man inte riktigt vet hur man ska formulera sig i vad exakt man vill få ut av systemet. E använder färger som liknelse för att förklara problematiken: Om kunden nu vill ha en blå lösning, och vi som leverantör endast erbjuder gult som faktiskt fungerar utmärkt. Denna problematik resulterar ofta i att leverantören med gul lösning inte får uppdraget. Istället hamnar uppgiften hos en leverantör som säger att de kan leverera blått, men att det är något som man vill gå in på senare i

projektet. När utvecklingen sedan kommit en bit på vägen så förklarar de att blått kostar betydligt mer än vad som nämnts tidigare. E menar att detta ofta resulterar i att med att systemet inte kan användas. Möjligtvis kommer man även fram till att den gula lösningen faktiskt var den bästa lösningen. Kunden trodde redan från början att den ”gula leverantören” inte kunde leverera blått, men i själva verket gjorde de det med vetskapen om att det inte passade den här lösningen. Tolkning från kundens sida är alltså något som kan bli problematiskt menar E.

### **Hur sammanställer ni ett lösningsförslag/kravspecifikation utifrån från kundens information?**

E förklarar att man inleder med att fastställa vilka krav som finns, och hur dessa krav kan uppfyllas. Sedan går man igenom vilka komponenter som är nödvändiga för att skapa lösningen. Man går alltså igenom saker som innefattar hur utvecklingen ska ske. Exempelvis funktioner som man måste utveckla på sidan, om man ska konfigurera, eller använda sig utav ren extrakod.

### **Händer det att ni går in i en organisation och utvecklar kravanalysen själva?**

E förklarar att vid en offentlig upphandling är det oftast svårt göra det. Det kan anses som att man är partisk och påverkar upphandlingsprocessen från kundens sida menar han. Ett exempel på detta är att föreslå en blå lösning, vetandes att vår organisation specialiserar sig på blått. Han poängterar dock att det är fullt möjligt i privat sektor.

### **Utför ni någon form av valideringsprocess?**

E svarar att man utför olika typer av valideringsprocesser. Exempelvis om kunden ställt upp 20 krav på vad systemet ska innefatta så används de som valideringskrav. E betonar här vikten av att föra dialoger med kunden så att man kan hantera ytterligare krav som kan dyka upp längs vägen. Man gör även kundundersökningar för att försäkra sig om att man har gjort ett bra jobb. Utöver detta bör man kontinuerligt återkoppla med kund och hålla nära kontakt säger E. Han menar att man gör detta på ett effektivt sätt möter man oftast kundens krav i slutet på projektet.

### **Vad tycker du om de generella modeller som finns ute idag?**

Det finns enligt E tre stora fel med upphandlingar i offentlig sektor. Det första är att man vill ha ett fast pris till ett rörligt mål. Att man t.ex. säger att X kostar 10 miljoner, men sedan säger kunden att de vill ha Y och Z också, då blir ju kostnaden högre. Det andra problemet är avtalskrav – vilket innebär att man måste acceptera de avtal som finns, vilket exempelvis kan innefatta att man måste acceptera alla storlekar, skadestånd etc. E ger exemplet om man gjort hemsidan med fel blåa nyans, då kan kunden kräva leverantören på ofantligt stora summor i skadestånd.



### **Hur kan man åtgärda detta?**

E rekommenderar att man inte ska anta upphandlingen om den ser ut på det sättet. Han förklarar att det trots allt är en fri marknad och man kan alltid väja att inte svara. Dessutom kan man ställa frågor till kund och förklara att det är olämpligt med förhoppningen att de ska ändra sig.

### **Är det lättare om kunden har stora IT-kunskaper i jämförelse med någon som är helt okunnig?**

Ja oftast säger E och förklarar att mycket IT-kunskap leder till att det blir tydligare med vad de faktiskt vill ha. Han menar att det stora problemet inte ligger i att kunden ställer för höga krav, utan snarare i att de har otydliga krav. E illustrerar problemantiken med ett exempel på en bilförsäljning:

Kunden säger: Jag vill ha en bil som ska vara blå. I detta fall kan försäljaren sälja vilken bil som helst bara den är blå. Kunden kan specificera sig lite mer, och säga att bilen maC får kosta 100 000kr. Försäljaren får då en snävare begränsning, men har fortfarande ett antal bilar att välja på bara de ligger inom kraven ”blå färg och max 100 000kr”.

Kunden köper en bil och opponerar sig med: Men denna har ju inga nämnvärda hästkrafter och ni sa att ni hade bra bilar? Det saknas ett GPS-system och det är ju ingen cabriolet.

Bilförsäljaren: Du ville ha en bil för 100 000 kr, vad trodde du att du skulle få för det priset?

### **Kundsidan verkar ha striktare steg i sin upphandlingsprocess till skillnad från leverantören som verkar avsluta processen redan vid kravspecifikationen? Stämmer detta?**

Ja det stämmer för det mesta, säger E. Problemet ligger i att hela processen för offentliga upphandlingar är extremt formell. Först ska man svara på ett visst sätt, ange specifika svar på specifika frågor, det ska stå i en speciell ordning mm, förklarar han.

Vid privata upphandlingar är lite lättare vad gäller denna biten säger E och förklarar att man då kan föra en parallell dialog med kunden.

Vid offentliga upphandlingar (myndigheter, kommuner, landsting mm.) får man knappt ha någon kontakt med kunden efter att man skickat sitt anbudssvar. Detta är ett omständigt sätt att arbeta på menar E. Däremot ger det större möjligheter, och en mer rättvis konkurrens då man undviker ”kompis köp”.

**Enligt den litteratur och teori som vi har behandlat i vår uppsats görs en upphandling genom att man börjar med att analysera kraven, ställer upp en kravspecifikation, designar systemet, utför prototypstestning och systemtestning som leder till en verifiering. Utöver dessa, finns det något som du tycker att man bör lägga till/ta bort?**

E säger att dessa stegen enbart beskriver en systemleverans men att det självklart är relevant för den typen av leveranser. För alla andra som behandlar annat måste den breddas. E fortsätter och berättar att många upphandlingar går ut på att visa att kraven uppfylls och att det går att bygga, inte den faktiska utvecklingen. Han menar att detta ibland kan innebära att upphandlingen stannar upp redan innan man ska bygga en prototyp.

**Hur ser detta ut specifikt för intranät?**

Ibland byggs det prototyper, andra gånger frågar man sig bara vad de grundläggande produkterna klarar av, säger E. Går det att skapa funktionen? Hur skulle det i så fall fungera? Dessa frågor ställs på konceptuell nivå.

**Hur skiljer sig intranätsupphandlingar från andra?**

E menar att det är väldigt systemintensivt och mer tekniskt. Vid andra upphandlingar ska man redovisa vad konsulten kostar per timme, medans vid Intranät blir det mer en fråga om man kan leva upp till de ”skall”-krav som finns för systemet.

**Hur agerar de olika stegen i upphandlingsprocessen, iterativt med varandra?**

E berättar att det först kommer in ett antal krav som de analyserar. Dessa mappas sedan mot eventuella produkter. Därefter designas en lösning som man stämmer av mot kraven så att lösningen matchar kraven. Lösningen jämförs sedan med prisbilden. Behövs något justeras för att matcha priskravet? Denna fråga måste besvaras och lösas innan ett totalsvar skickas till kunden, så att man inte ger ett toppen förslag som överstiger prisbilden. E poängterar att leverantören hela tiden måste ha koll på så att ”skall”-kraven i kravspecifikationen uppfylls. På så vis arbetar de iterativt. De 3 huvudstegen, lite hårddraget, är att definiera och avgöra om man ska svara ja eller nej på förfrågan, utveckla ett lösningsförslag och slutligen säkerställa rätt pris och projektgrupp.

**Återkopplar ni något med kunden efter utrullning?**

Ja, vi brukar ha uppföljningar efter en viss tid för att säkerställa att kunden är nöjd, säger E. Vid en större utrullning försöker vi ha kontakt med kunden löpande därefter för att se om vi kan hjälpa till med mer. Oftast när man redan har levererat en gång och kunden har fått en positiv inställning så har man en chans att få till fler eller nya beställningar/affärer.

### **Hur viktigt är kundens delaktighet i hela projektet?**

E säger att det är extremt viktigt med kundens delaktighet i projektet. Kunden måste vara involverad till hög grad. Vad gäller själva upphandlingen så är det däremot svårare.

Vid en privat upphandling kan man ha en viss dialog, men vid offentliga upphandlingar får man i princip inte ha någon kontakt/dialog med kunden. Därmed blir det således totalt oviktigt med kundens delaktighet.

Det är däremot jätteviktigt att under projektets gång med kundens delaktighet pga att det inte är värt något om det inte lever upp till kundens förväntningar, där kvittar det hur pass coolt/bra det är i övrigt. Exempelvis om leverantören bygger en Porsche trots att kunden enbart ville ha en elbil, då blir det bara fel menar E.

### **IT-upphandlingen verkar sluta när kravbilden fastställts från leverantörens sida? Stämmer det?**

E håller inte riktigt med och menar att det beror på om det är en offentlig eller privat upphandling. Enligt honom pågår upphandlingen tills att man svarat på förfrågningsunderlaget och gett svar på upphandlingen. När man sedan får uppdraget påbörjas själva projektet som fortlöper fram till utrullningen. Upphandling ses annars som den skriftliga delen säger E.

## 2 Intervjufrågor

- Kan du börja med att berätta om dig själv. Vilken befattning har du i dagsläget?
- Kan du ge en kort beskrivning av ert företag?
- Vilken erfarenhet har du av IT-upphandlingar?
- Kan du beskriva hur en IT-upphandling vanligvis går till på ert företag?
- Använder ni något speciellt ramverk/modell som ni utgår ifrån vid era upphandlingar eller varierar det från kund till kund?
- Vilka faktorer i en IT-upphandling anser du vara viktigast? Varför?
- Vilka faktorer i en IT-upphandling anser du vara mindre viktiga? Varför?
- Om en IT-upphandling skulle anses vara misslyckad, dvs kunden får inte det den ville ha - var i upphandlingsprocessen tror du att misstaget generellt sätt sker? Varför?
- (Antagandes att respondenten inte redan nämnt de)
- Hur går ni till väga när ni sammanställer en kravspecifikation?
- Hur går ni tillväga när ni analyserar de eventuella problem som kan finnas i kundens verksamhet?
- Hur arbetar ni för att försäkra er att kunden är nöjd med er produkt?
- Hur går utvecklingen av systemet till?
- Använder ni någon form av validering/verifiering med kunden? För att försäkra att båda parter är överrens.
- Kan du förklara hur de olika stegen i er upphandlingsmetod agerar iterativt med varandra?
  
- Vilken/vilka av ovanstående faktorer anser du är viktigast för att en upphandling ska bli lyckosam? Varför?
  - Anser du att någon ytterligare aspekt än de vi nämnt har stor påverkan för ett lyckat resultat vid en upphandling?
- Fungerar dagens modeller/metoder vid upphandlingar?
- Krångliga att använda? Bra stöd? mm.
- Användarens delaktighet beryktas vara viktig, vad anser ni?
- Stämmer/stämmer inte?
- Varför?
- Vad har ni för krav på kundens IT-kunskaper?
- Bra/Dåligt att kunden har tidigare IT-kunskaper och vet vad de vill ha/behöver till sin verksamhet?

# 7 Referenser

Adam, F. & O'Doherty, P. (2000): Lessons from enterprise resource planning implementations in Ireland – towards smaller and shorter ERP projects

Aladwani, A. M. (2001): Change management strategies for successful ERP implementation. *Business Process. Management Journal*, 7(3): 266-275.

Antemar, G & Undall, B. (2011): Statliga IT-projekt som överskrider budget

Atkinson, R. (1999): *Project management: cost, time and quality, two best guesses and a phenomenon, it's time to accept other success criteria*

Bevan, N & Maguire, M. (2002): User requirements analysis: A review of supporting methods

Brooks, F. (1986), No Silver Bullet: Essence and Accidents of Software Engineering

Brorsson, M. (2009): Företagande.se: Stora problem med IT-projekt

Chatfield, C & Johnson, T. (2007): Microsoft Office Project 2007 Step by Step: A short course in project management.

Davenport, T.H. (2000): Mission Critical, Realizing the Promise of Enterprise System. USA: Library of Congress Cataloging in Publication Data.

DeLone, W.H & McLean, E.R. (1992): Information Systems Success: The Quest for the Dependent Variable

Gulliver, F. (1987): Post-project appraisals pay, Harvard Business Review

Heckman, R.L. (1999): Managing the IT Procurement Process

Hirschheim, R & Smithson, S. (1999): *Evaluation of Information Systems: a Critical Assessment, Beyond the IT productivity paradox: assessment issues*

Jacobsen, D.I. (2011): Vad, hur och varför: Om metodval i företagsekonomi och andra samhällsvetenskapliga ämnen. uppl: 1:13. ISBN: 978-91-44-04096-7. 503 s.

Johansson, A. Sandgren, B. (2004): Kritiska faktorer vid införande av IS/IT-system

Labuschagne, C & Brentb, A. (2007): Sustainable Project Life Cycle Management: *the need to integrate life cycles in the manufacturing sector*

Lattemann, F and Lehmann, E. (1997): "A methodological Approach to the Requirement Specification of Embedded Systems".

- Larsson, H. (2012): Stora IT-projekt innebär ofta stora problem
- Layard, R & Glaister, S. (1994): *Cost-Benefit Analysis* s.1-56. ISBN 9780521466745
- Levinson, M. (2003): It Ain't Over Until You Do The Post-Implementation Audit
- Magnusson J, Olsson B. (2008): Affärssystem. 2:a uppl. ISBN 978-91-44-05366-0
- Nicoulau, A. (2004): Quality of postimplementation review for enterprise resource planing systems
- Olofsson, M, Uddbom, J. (2011): Finns det någon list med digitala tavlor?: En kvalitativ studie av översiktstavlor inom agila projekt och förutsättningar för digitala sådana
- Olphert, C.W. & Damodaran, L. (2002), Getting what you want, or wanting what you get? - beyond user centred design
- Saarinen, T. Vespäläinen, P.J.(1994): Procurement Strategies for Information Systems
- Sommerville, I. (2011): *Software Engineering*. 9:e uppl. ISBN: 978-0-13-703515-1. 790 s.
- Talluri, S. Narasimhan, R. Viswanathan, S. (2007): *Information technologies for procurement decisions: a decision support system for multi-attribute e-reverse auctions*
- Verville, J & Halington, A. (2003): A six-stage model of the buying process for ERP software

## Webbsidor

Gartner (Hämtad 2013-05-13):

<http://www.gartner.com/technology/about.jsp>

Idg (Hämtad 2013-05-01)

Malmqvist, M. (2012): *Det har gått troll i överklaganden.*

<http://www.idg.se/2.1085/1.444968/det-har-gatt-troll-i-overklaganden>

Kontract (Hämtad 2013-05-07)

<http://www.kontract.se/blogg/fem-vanliga-fallgorpar-i-it-projekt>

<http://www.kontract.se/white-papers/undvik-6-vanliga-misstag-i-din-it-upphandling>

KvalitativMetod (Hämtad 2013-04-13)

<http://kvalitativmetod.webs.com/intervjuer.htm>

Marcks von Würtemberg, Liv. 2010 (Hämtad 2013-05-13): Därför floppade projekten: Tre svenska it-fiaskon under lupp

<http://cio.idg.se/2.1782/1.326833/darfor-floppade-projektentre-svenska-it-fiaskon-under-lupp>

Standishgroup (Hämtad 2013-05-12)

[www.Standishgroup.com](http://www.Standishgroup.com)

Pstestware, (Hämtad 2013-04-15)

<http://www.pstestware.com/index.php?cid=12&scid=125>