Kravhantering med hjälp av Use Case

(HS-IKI-EA-04-603)

Amir Esfahani (a00amika@ida.his.se)

Institutionen för kommunikation och information
Högskolan i Skövde, Box 408
S-54128 Skövde, SWEDEN

Examensarbete inom informationssystemsutveckling, vårterminen 2004

Handledare: Jesper Holgersson

[Kravhantering med hjälp av Use Case]

Examensrapport inlämnad av Amir K Esfahani till Högskolan i Skövde, för Kandidatexamen (B.Sc.) vid Institutionen för kommunikation och information.

[2004-06-05]

Härmed intygas att allt material i denna rapport, vilket inte är mitt eget, har blivit tydligt identifierat och att inget material är inkluderat som tidigare använts för erhållande av annan examen.

Signerat: _____

Förord

Ett stort tack till alla de respondenter som har ställt upp och gjorde denna undersökning möjligt.

Tack till examinatorn Eva Söderström och handledaren Jesper Holgersson som hjälpt och stöttat mig under examensarbetets gång.

Slutligen vill jag tacka min flickvän som hela tiden stöttat mig till 100% under examensarbetets gång.

Kravhantering med hjälp av Use Case

Amir Esfahani (a00amika@student.his.se)

Sammanfattning

Detta examensarbete ger en introduktion till området systemutveckling. Kravhantering har i alla tider varit en viktig del i systemutveckling. För att kunna lyckas med kravhanteringen under ett systemutvecklingsprojekt är det viktigt att använda sig av rätt teknik. En teknik som finns för att kunna hantera de krav som ställs på ett system är Use Case som härstammar från UML. Syftet med detta examensarbete var att ta reda på de för och nackdelar som har upplevts av användare som har arbetat med Use Case i samband med kravhantering. Presentation i detta examensarbete sker genom metoden intervju samt litteraturstudier. Det material som bearbetats fram via intervjuerna har analyserats med hjälp av olika litteraturer för att ge en klar bild av problemställningen. Resultatet visar att Use Case är en omtyckt teknik som innehar både fördelar och nackdelar där fördelarna överväger nackdelarna.

Nyckelord: Informationssystem, Systemutveckling, Use Case, Kravhantering, Requirements Engineering.

Innehållsförteckning

1 Introduktion 1
2 Bakgrund 2
2.1 Informationssystemutveckling 2
2.2 Krav 5
2.2.1 Kategorisering av krav 6
2.2.2 Requirements Engineering 7
2.3 Use Case 8
2.3.1 Use Case introduktion 8
2.3.2 Use Case-Diagram 10
3 Problemområde 12
3.1 Problemprecisering 12
3.2 Avgränsning 13
3.3 Förväntat resultat 13
4 Metod 14
4.1 Valda metoder 14
4.1.1 Intervju 14
4.1.2 Litteratur 15
4.1.3 Tänkt tillvägagångssätt 16
5 Genomförande 17
5.1 Tillvägagångssätt 17
5.2 Materialpresentation 18
5.2.1 Introduktionsfråga 18
5.2.2 Huvudfrågor 18
5.2.3 Kompletteringsfråga 22
5.3 Värdering av insamlat material 23
6 Analys 24
6.1 Use Case och kravhantering 24
6.2 Fördelar Use Case 25
6.3 Nackdelar Use Case 27
6.4 Diskussion kring tabellerna 29
6.4.1 Fördelar 30
6.4.2 Nackdelar 30

7 Resultat 31
7.1 Fördelar med Use Case 31
7.2 Nackdelar med Use Case 31
7.3 Diskussion kring resultatet 32
8 Diskussion 33
8.1 Erfarenheter kring arbetet 33
8.2 Reflektioner kring resultatet 34
8.3 Förslag på fortsatta arbeten 34
9 Referenser 36
Bilaga 1 Intervjufrågor
Bilaga 2 Introduktionsemail

1 Introduktion

Informationssystemsutveckling används för att studera hur olika individer arbetar för att utveckla ett databaserat system (Cronholm, 1998). Kravhantering är en viktig del i ett systemutvecklingsprojekt. Anledningen till detta är för att det är genom krav som utvecklaren kan tillfredställa konsumenten eller den som skall köpa systemet. Det hela handlar egentligen om att kunna komma överens med kunden som i ett senare skede skall kunna medföra till en bättre utveckling av det framtida systemet (Flensburg, 1987). Utvecklingen av datorbaserade system har på senare tid varit en plåga på grund av att system ofta överlämnats sent. Den största anledningen till detta är att hanteringen av krav inte är effektiv nog skriver Sommerville och Sawyer (1997). Det självklara är inte alltid så självklart. Att kunna se till att produkten som utvecklas håller exakt de krav som ställs av beställaren är inte alltid lätt. Systemutvecklaren måste alltid tänka på vad användaren har för mål. För att kunna uppnå de mål som kunden sätter måste kraven hanteras noga på den ofta korta tid som systemutvecklarna har på sig (Karlsson, 1996). Problem som leder till misslyckanden av utvecklingsprojekt är bland annat: försenad leverans, dålig kommunikation mellan utvecklare och kund som leder till misstolkningar, användaren vet själv inte vad denne vill ha för system samt att budgeten spricker och leder i sin tur till att kundens önskemål inte kan uppnås (Kotonya & Sommerville, 1998). Kravhantering går ut på att systemutvecklaren med hjälp av olika tekniker skall försöka samla in så mycket information som möjligt om kundens krav för att senare kunna analysera och sammanställa dessa (Celander, 1996).

En möjlig teknik till att kunna utvinna dessa krav är Use Case (Regnell, 1999). Det finns flera olika sorters Use Case-tekniker. Den teknik som har valts att studeras i detta examensarbete är Use Case som härstammar från Unified Modeling Language (UML) som enligt Schneider och Winters (2001) är en bred metod med en stor acceptans i många av dagens stora företag.

Rapporten kommer bland annat att undersöka de för och nackdelar som upplevs av UML:s Use Case i samband med kravhantering inom systemutveckling. Författaren av denna rapport anser att det är viktigt att ta reda på vilka för och nackdelar en teknik har i samband med kravhantering för att kunna vara säker på att tekniken inte genererar sämre resultat än om den inte använts. Detta är anledningen till varför denna undersökning är viktig att utföra för att kunna förhindra att Use Case- tekniken används vid fel situationer som leder till eventuella framtida problem.

Den metod som används för att slutföra denna undersökning är intervju kombinerad med litteratur**.** Anledningen till detta är för att kunna få en så bred syn som möjligt på problemet och samtidigt kunna komma i kontakt med personer som använder sig av Use Case. Det resultat som bearbetats fram visar tydligt att Use Case är en bra teknik för systemutveckling.

Materialet kommer att presenteras genom att först i kapitel två definiera och berätta om de centrala delarna som ligger till grund för den problemställning som skall förklaras i kapitel tre. Under kapitel tre kommer problemställningen att presenteras samt anledningen till varför problemet är intressant att undersöka. Vidare i kapitel fyra kommer den metod som valts till att samla in material för att kunna besvara på rapportens problemställning att presenteras. Kapitel fem visar rapportens tillvägagångssätt, det vill säga hur genomförandet av materialinsamlingen gick till. I kapitel sex redovisas den analys som bearbetats fram via kapitel fem och som ligger till grund för nästkommande kapitel nämligen resultat. Till sist presenteras diskussionskapitlet där både rapporten och resultatet diskuteras samt en ide för framtida arbeten.

2 Bakgrund

Detta kapitel kommer att presentera den grundläggande information som ligger till bakgrund för detta examensarbete. En kort presentation av informationssystem och systemutveckling kommer först att ges. Därefter kommer livscykelmodellen att förklaras. Vidare kommer krav och vikten av kravhantering att beskrivas. Hanteringen av krav har ett eget begrepp som kallas för Requirements Engineering[1] (RE) som kommer att presenteras som senare leder till rapportens huvud punkt nämligen tekniken Use Case[2] som härstammar från UML.

2.1 Informationssystemutveckling

Andersen (1994) anser att ett informationssystem är ett datoriserat system som är knutet till en speciell uppgift som både kan ta emot och förmedla information mellan olika parter. ”*Informationssystem är ett system för insamling, bearbetning, lagring, överföring och presentation av information*” (Andersen, 1994, sid 15). Idén med ett informationssystem är att kunna förbättra organisationens arbetsgång. Ett informationssystem som bidrar till sämre resultat i organisationen är inte korrekt utvecklat.

Andersen (1994) lägger mycket tyngd på att termen informationssystem inte får missförstås. Informationssystem utesluter inte den mänskliga närvaron utan behandlingen av uppgifterna kan både göras av maskiner och människor. Andersen (1994) menar att för att kunna utveckla ett informationssystem på bästa möjliga sätt behövs en övergripande bild som fastställer de viktigaste egenskaperna i arbetet, samt redogör för de metoder och tekniker som behövs för att utföra utvecklingen. Livscykelmodellen som presenteras i kap 2.1.1 är en modell som kan visa dessa. Ett väl utvecklat informationssystem skulle medföra en förstärkning i organisationens bas som i sin tur skulle medföra till ett bättre utvecklat företag (Eriksson & Penker, 2000).

Den process som sker vid utvecklingen av ett informationssystem kallas ofta för systemutveckling. Enligt Flensburg (1987) kan systemutveckling ses som en utformning av en modell av verksamhet samt förverkligandet av modellen till det blivande informationssystemet. Utifrån detta kan författaren av denna rapport konstatera att systemutveckling inte behöver enbart vara en nyutveckling från grunden. Termen kan användas både när ett system skall nyskapas eller när ett befintligt system vidareutvecklas. Avison och Fitzgerald (1998) skriver att utvecklingen av ett informationssystem kommer på ett eller annat sätt att gynna organisationen, dock är det väldigt viktigt att utvecklaren har rätt hjälpmedel för att kunna lösa problemet på bästa möjliga sätt.

Andersen (1994) har analyserat fram olika hjälpmedel för att kunna angripa en systemutvecklingsuppgift. Ett av dessa hjälpmedel är *modell* vilket definieras som en översikt över utvecklingsarbetet. En modell beskriver vilket arbete som bör göras samt vem som kan förväntas genomföra arbetet (Andersen, 1994). En modell brukar ofta täcka omfattande delar i ett arbete och är därför uppbyggd av olika delar. Dessa olika delar kan ibland kallas för faser, steg, arbetsområden eller metodområden. Ett typiskt exempel på en modell är livscykelmodellen som är en typ av utvecklingsmodell som beskriver hela

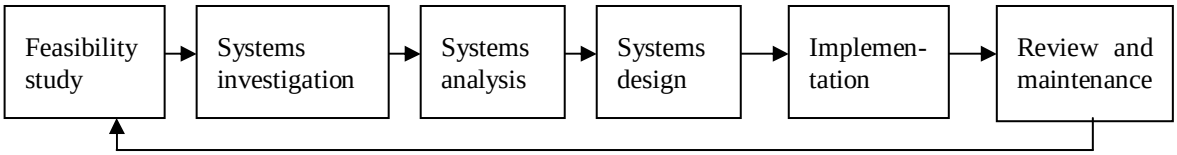
[1] Det finns ingen bra svensk översättning för Requirement Engineering, därför används det engelska ordet i denna rapport.

[2] Det svenska ordet för Use Case kan ses som användningsfall. Use Case är ett välkänt begrepp bland dagens företag, därför har författaren beslutat att använda det engelska ordet i denna rapport.

informationssystemets livslopp. De flesta utvecklingsmodeller brukar innefatta alla faser i en livscykel, allt från idé till färdigt implementation (Andersen, 1994).

Vidare kommer livscykelmodellen enligt Avison och Shah (1997), och Andersen (1994) att presenteras. Dessa författare ritar inte modellen på exakt samma sätt, dock går det utifrån deras beskrivningar konstatera att faserna är likvärdiga. Anledningen till att livscykelmodellen tas upp i detta examensarbete är för att rapporten har fokus i vissa av dess faser, nämligen de tidigare faserna som till exempel verksamhetsanalys och systemanalys där kravhantering spelar en viktig roll.

Utvecklingen av informationssystem kan ses och studeras på olika sätt. Ett sätt att se på informationssystemsutveckling är enligt Andersen (1994) genom livscykelmodellen. Denna process är en iterativ process där användaren av modellen måste göra analyser och testa sig fram genom utveckling. Genom att följa livscykelmodellen lär sig utvecklaren mer om själva situationen som denne befinner sig i, skriver Avison och Shah (1997). Nedan följer en presentation utav livscykelmodellen samt en kort beskrivning av varje fas.



Figur 1 Livscykelmodellen enligt Avison och Shah (1997, sidan 71)

Förändringsanalys (eng Feasibility study)

Förändringsanalys är enligt Andersen (1994) det första som görs i en informationssystemsutvecklingsstudie. Andersen (1994) och Avison och Shah (1997) säger att denna fas beskriver i helhet nuläget och den eventuella förändringsbehov som finns för den inblandade organisationen. Avison och Shah (1997) anser att under detta steg är ett projektexempel framtaget och det är nu dags att ta reda på om projektet går att förverkliga eller ej. Här tar utvecklaren hänsyn till bland annat hur dagsläget ser ut samt mål som redan finns i företaget.
Avison och Shah (1997) skriver att en ve de viktigare uppgifterna i denna fas är att svara på om informationssystemet bör förfärdigas eller ej. Ett godkänt resultat medför att projektet går vidare till nästa fas nämligen ”Systems investigation”

Verksamhetsanalys (eng Systems investigation)

Enligt Avison och Shah (1997) har ett projekt valts att genomföras om verksamheten kommit till denna fas. Under denna fas kommer djupare undersökningar om de personer som skall använda systemet och verksamheten att göras. Vidare är det här viktigt att ta reda på vilka behov de olika aktörerna har för att få ett så bra system som möjligt anser Andersen (1994).

Systemanalys (eng Systems analysis)

Föregående fas undersökte verksamheten som systemet skall verka i. Här skapas modeller av den information som beskriver data, processer och eventuella aktörer (Avison & Shah, 1997). Det är här som informationssystemets krav tas fram (Andersen, 1994). Avison och Shah

(1997) menar att de dokument som sparas här är det som kallas för kravspecifikation. Systemanalys och den tidigare fasen Verksamhetsanalys är bland de viktigare faserna för detta examensarbete. Det är bland annat under dessa faser som användarens och systemets krav hanteras.

Utformning (eng Systems design)

Utifrån föregående fasers dokumentering kan utvecklaren nu från kravspecifikationen från föregående fas börja designa systemet som skall stödja de identifierade aktörerna förklarar Avison och Shah (1997). Under denna fas skall utvecklaren utifrån kravspecifikationen kunna beskriva varför systemet skall byggas, till vem den är riktad till och vilka aktörer den skall stödja. Här tas även logiska modeller fram som vidare ger förslag till hur gränssnittet bör se ut (Avison & Shah, 1997).

Implementation (eng Implementation)

Avison och Shah (1997) menar att under denna fas är systemet redan färdigbyggd och det är här själva kodning av informationssystemet äger rum. Även testning av systemet samt utbildning av de nya användarna är något som sker under denna fas enligt Avison och Shah (1997).

Förvaltning och underhåll (eng review and maintenance)

Efter att ett system implementerats så måste det självklart underhållas. Skulle systemet ”krascha” så måste det gå att uppdatera. Allt detta kräver stöd och det är vad denna fas handlar om. Så småningom kommer underhållet att bli för dyrt och det är då det leder till ett nytt projekt (Avison & Shah, 1997). Se figur 1.

2.2 Krav

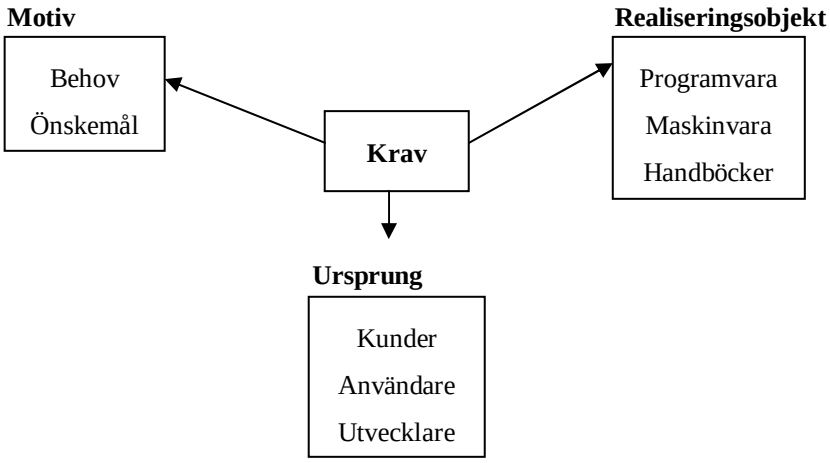
Under detta kapitel kommer begreppet krav att presenteras mer ingående. Att begreppet tas upp i detta examensarbete är för att kunna förmedla en tydligare bild av vikten med krav samt hanteringen av det. Krav är något som är viktigt i en systemutvecklingsprocess och fastställs under de tidigare faserna i informationssystemsutveckling som en detaljerad förteckning om vad som bör implementeras i det nya systemet. (Sommerville & Sawyer, 1997; Karlsson, 1996). Krav kan ses som en nödvändig funktion eller egenskap som en användare behöver för att kunna tillfredställa de behov som finns på ett system (Regnell, 1999)

Utan krav är det omöjligt att utveckla ett system. Det är även omöjligt att logiskt kunna koppla programmet eller systemet till kundens önskan. Dock är det viktigaste inte bara att ”ha” krav utan det är hanteringen av krav som är viktig. (Kulak och Guiney, 2000).

Karlsson (1996) anser att det är viktigt att inte se krav som någon egenskap som ett informationssystem nödvändigtvis måste uppfylla. Det är ytterst få krav som måste, oavsett kostnad och tid implementeras. Med andra ord skall krav i första hand betraktas som önskvärda egenskaper i ett system och inte som nödvändiga egenskaper.

Olika författare definierar krav på olika sätt, detta kan bero på deras personliga idéer och erfarenheter. Utifrån tidigare förklarade definitioner kommer krav i denna rapport att tolkas som något ett system måste uppfylla för att kunna tillfredställa sina användare. Anledningen till att denna tolkning valts är att författaren av denna rapport anser att krav har en viktig roll i systemutveckling. Detta bidrar till att hanteringen av krav blir ännu viktigare.

Ett krav är ofta uppdelad i olika delar. Här under presenteras en bild (figur 2) av ett kravs olika delar och därefter en kort beskrivning av varje del**.** Materialet är hämtat från Karlsson (1996).



Figur 2: Ett kravs olika delar, ur *Karlsson (1996 ,sidan 9)*

Ett krav har alltid ett visst *ursprung* från någon kund som har angett ett förslag på krav. Andra personer som kan ingå i denna del är slutanvändaren och eller utvecklaren.

Krav har även ett *motiv* som bidrar till kravets existens. Motiv samlar ofta in något speciellt för slutanvändaren. Behov och önskemål är några av de punkter som kan påträffas här

Till sist har ett krav alltid ett *realiseringsobjekt.* Detta kan ses som något som fullföljer eller implementerar det önskvärda systemet. Det vanligaste exemplet här är programvara med det är inte omöjligt att även maskinvara eller handböcker ingår.

Figur 2 visar att ett krav har en ganska komplicerad uppbyggnad. Kotonya och Sommerville (1998) menar att de vanligaste problemen som kan uppstå med krav är att kraven redan från början är inkonsistenta eller inte färdigställda. Kraven speglar ofta inte kundens verkliga behov vilket leder till missuppfattningar mellan kunden och utvecklaren, samt utvecklaren och programmeraren.

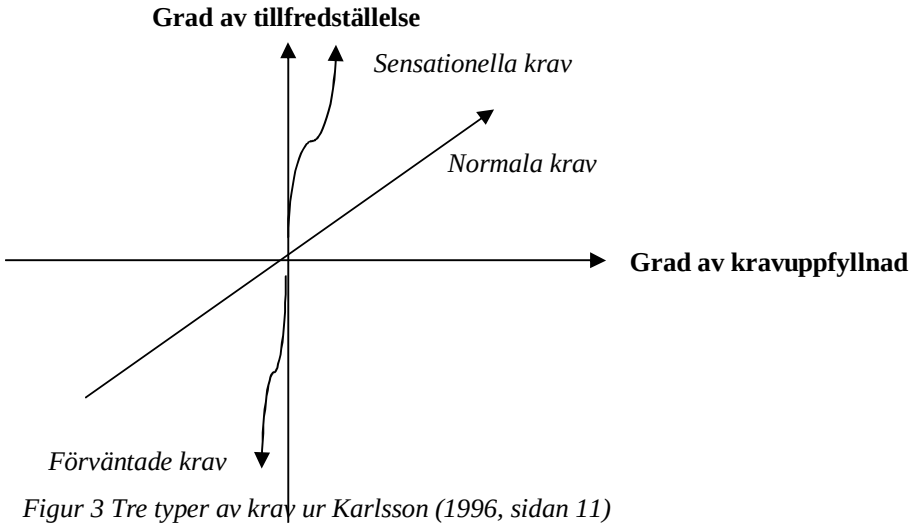
2.2.1 Kategorisering av krav

Enligt Figur 2 går det att konstatera att ett krav har en ganska komplicerad uppbyggnad**.** Enligt Karlsson (1996) kan krav delas upp i två olika kategorier nämligen funktionella och icke funktionella krav. De *funktionella kraven* är precis som de låter. De visar funktionaliteten i systemet, alltså de uppgifter som programvaran måste klara av (Karlsson 1996; Kulak & Guiney 2000). Funktionella krav brukar normalt enligt Karlsson (1996) och Sommerville och Sawyer (1997) kunna beskriva hur olika system praktiskt beter sig när de tar emot data från olika indataenheter. Funktionella krav är de krav som programvarusystem skall kunna erbjuda den framtida användaren. Vidare är dessa krav ofta händelsebaserade. Exempel på händelsebaserade krav kan vara då användaren trycker på x och feedback ges till användaren (Karlsson, 1996).

Enligt Karlsson (1996) kan de *icke funktionella kraven* beskrivas som egenskap hos det tänkta systemet och dessa svarar ofta på frågan vem och hur. De icke funktionella kraven är bland de viktigaste kraven på grund av att de berör systemet på ett mer betydelsefullt sätt, till exempel effektivitet och användbarhet (Karlsson, 1996). De icke funktionella kraven är ofta de krav som är dolda men samtidigt väldigt viktiga för användaren. Dessa krav beskriver ofta detaljer om kvalitet, snabbhet med mera (Kulak & Guiney, 2000). Till exempel kan ett icke-funktionellt krav vara att ett viss temperatur skall vara uppnådd inom två minuter efter ändrad inställning.

Karlsson (1996) anser att diskussionen om funktionella och icke funktionella krav är ofta mer till besvär än till nytta**.** Anledningen till detta är att det inte alltid är lika lätt att kunna skilja mellan funktionella och icke funktionella krav. Just därför har företag på senare år börjat att enbart använda de icke funktionella kraven mer som en checklista.

På grund av svårigheten att se på dessa olika krav har Karlsson (1996) beskrivit kompletterande sätt att se på krav. Dessa kan delas upp i sensationella, normala och förväntade krav och redovisas vidare i figur 3.



Figur 3 Tre typer av krav ur Karlsson (1996, sidan 11)

Enligt Karlsson (1996) är *sensationella* krav något som inte är definierad av användaren. Skulle dessa krav lyckas, skulle detta bidra till stor tillfredställande för användaren. Exempel på sådana krav kan vara tekniska aspekter som inte kommer att bidra till någon försämring om detta lyckas.

Normala krav som enligt Karlsson (1996) är definierade av kunden som ofta kommer att bidra till slutresultatet.

Förväntade krav är de som inte är definierade men som ofta förväntas av kunden. Det är ofta dessa krav som leder till kundens otillfredsställdhet (Karlsson, 1996).

2.2.2 Requirements Engineering

Requirements Engineering (RE) är ett begrepp som har skapats för att täcka alla de aktiviteter som hanterar dokumentering, utveckling och underhållning av krav för datorbaserade procedurer (Kotonya & Sommerville, 1998). RE kan ses som riktlinjer för att kunna implementera mjukvara, hårdvara och även till system där människan är inblandad (Sommerville & Sawyer, 1997). En god användning av RE-processen ökar chanserna till mer kontrollerad systemutveckling som i sin tur leder till kvalitetssäkring och sist men absolut inte minst kundens tillfredställelse (Kruchten, 2000).

Enligt Regnell (1999) täcker Requirements Engineering hela livscykelmodellen men fokuserar på de tidiga faserna inom livscykelmodellen, se Figur 1. En av dessa faser kan till exempel vara vad, alltså vad är det som skall implementeras. Anledningen till att RE har en så stor fokus på de tidigare faserna i livscykelmodellen är att det är under dessa faser som kraven för det slutgiltiga systemet iterativt bearbetas fram (Macaulay, 1996 & Kotonya and Sommerville, 1998).

RE handlar om många olika aspekter anser Kotonya and Sommerville (1998). De påpekar att RE-processen är ett utmärkt val för utvecklaren då denne med hjälp av den kan få fram vad organisationen och användaren egentligen vill ha. Dock är det här viktigt att tillägga att detta påstående inte alltid är lätt att uppfylla. Andersen (1994) påpekar bland annat att under en utveckling har ofta ägaren till det blivande systemet ingen aning om vad han egentligen vill ha. Detta leder i sin tur till att systemutvecklarens jobb blir mycket svårare att utföra.

2.3 Use Case

Under detta kapitel kommer begreppet Use Case samt dess uppbyggd att diskuteras. Eftersom examensarbetet innefattar Use Case utifrån modelleringsspråket Unified Modeling Language (UML) kommer här en kort beskrivning av UML. Fokus i kapitlet kommer dock att ligga på Use Case.

UML är ett standardspråk som används till att dokumentera, konstruera och specificera system. Idén bakom UML är att grafiskt modellera och dokumentera notationer för att kunna beskriva systemets beteende (Muller, 1997). Enligt Liang (2002) är UML baserad på tre olika objekt nämligen modellering, metod och tekniker.

För att kunna hantera den komplexa kravhanteringsprocessen måste utvecklarna använda sig av olika typer av tekniker skriver Kotonya och Sommerville (1998). En teknik är ett arbetssätt som även kan ses som ett ”recept” som utvecklaren bör utgå ifrån för att få ett lyckat projekt (Andersen, 1994). En teknik som är välutarbetat och passar bra till de mesta systemutvecklingarna är enligt Regnell (1999) UML:s Use Case-teknik

2.3.1 Use Case introduktion

Use Case är en beskrivningsteknik som effektivt kan hjälpa utvecklaren att hitta de krav som ställs på systemet skriver Fantechi m.fl. (2003). Författarna skriver även att Use Case ofta är beskrivet med ett naturligt språk för att kunna definiera bland annat ett systems beteende. Detta kan i sin tur medföra att utvecklarna lättare kan hantera dokumenten, även om de av någon anledning skulle lämnas över till någon annan utvecklare för vidare utveckling. Use Case är skapad för att kunna validera och ta fram krav och designförslag samt kunna försäkra utvecklarna att kraven är uppnådda menar Jacobson (2001).

Fantechi m.fl. (2003) skriver att UML:s Use Case är väldigt lätt att förstå och detta kommer i sin tur att resultera bättre utvecklingsmöjligheter. Rosenberg och Scott (1999) anser att Use Case hjälper utvecklaren att genom modeller fånga användarens behov oavsett om det handlar om nyutveckling eller påbyggnad av befintligt system. En anledning till detta kan vara enkelheten att använda och förstå sig på Use Case. Rosenberg och Scott (1999) och Leffingwell (2000) skriver även att Use Case kan ses som sekventiella handlingar som bidrar till att de mål och eller krav som finns till ett system uppnås**.** Cockburn (2001) menar att Use Case-tekniken är ett bra val för utvecklaren, detta på grund av att Use Case kan hjälpa utvecklaren genom att tekniken enkelt visar systemets beteende i olika förhållanden. Vidare skriver Cockburn (2001); Kulak (2000) och Leffingwell (2000) att med hjälp av Use Case kan nu systemutvecklaren samla in de olika kraven och sammanställa dessa till ett användbart dokument inför utvecklingen.

Use Case underlättar för systemutvecklaren, dock behöver detta inte betyda att systemutvecklaren måste använda sig av Use Case under hela systemutvecklingsfasen. Use Case kan underlätta bland annat kommunikationen mellan utvecklarna som i sin tur medför att dokumentationen av kraven underlättas**.** Vissa systemutvecklare använder sig av Use Case i vissa speciella faser och andra använder sig av Use Case under hela utvecklingen. Use Case skall vara skrivet på ett så enkelt sätt som möjligt att det inte skall kunna misstolkas. Ett bra skrivet Use Case uppfyller alltid sitt syfte utan några större problem (Cockburn, 2001). Detta kan ses som en av de största anledningarna till varför Use Case blivit så framgångsrikt genom tiderna. Use Case-modellering är en iterativ process och passar bra för ett systemutvecklingsprojekt då användaren ändrar sig ofta under utvecklingsperioden. Det är då viktigt att låta aktören medverka så mycket som möjligt under processens gång för att slippa

gå tillbaka i projektet och ändra menar Dennis och Wixom (2000). Vidare skriver även författarna att det inte är helt ovanligt att under ett systemutvecklingsprojekt träffa på över åtta-nio skapade Use Case-modeller.

Leffingwell (2000) skriver att det är viktigt att en fullständig Use Case-modell innehar alla de aktörer och dess relationer som är kopplade till det specifika Use Caset som beskriver samspelet mellan användaren och systemet. När dessa olika Use Case-modeller sedan sätt ihop för att visa systemets flöde kallas de för Use Case-diagram (se kap 2.3.2). Det är viktigt för Use Case-modellen att ha med alla relationer till systemet så att systemutvecklaren inte skall missa viktiga detaljer vid uppbyggnaden. Missas viktiga detaljer eller relationer i Use Case-modellen kan detta leda till ett sämre system som i sin tur leder till missnöjd kund. Use Case-modellen och dess tydliga relationer bidrar till bättre förståelse för systemet. Det är viktigt att kunna vara säker på att Use Caset som utvecklats innehar de krav som beställaren kräver. Det enda möjlighet som finns till att se om de krav som beställaren har ställt stämmer överens med Use Caset är att tillsammans med beställaren gå genom Use Caset (Dennis och Wixom, 2000). En viktig fördel med tekniken är Use Case-diagrammet som kommer att presenteras i nästa kapitel. Diagrammet bidrar till att utvecklingen blir mer effektiv som genererar till att kravhanteringen bli smidigare.

Rosenberg och Scotts (1999, sidan 39) syn på ett korrekt genomfört Use Case-modellering är:

> *”The result of use case modelling should be that all required system functionality is described in the use cases”*

Utifrån detta kan vi konstatera att ett korrekt skapat Use Case bör inneha de krav som ställs på systemet från användarens synvinkel.

De viktigaste för och nackdelar som påträffats om Use Case i samband med kravhantering i detta kapitel kommer att sammanställas och presenteras i punktform nedan. Anledningen till att detta görs är på grund av att kunna förtydliga de som litteraturen har tagit upp för att vidare i kapitel 6 kunna visa vad som skiljer mellan litteraturens syn på för och nackdelar med Use Case i samband med kravhantering i förhållande till praktiken.

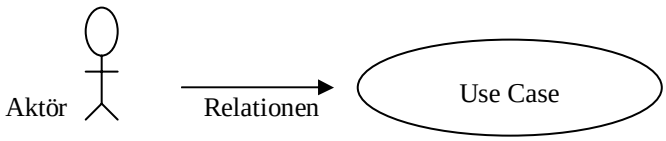
fördelar	***nackdelar***
• Use Case hjälper utvecklaren att effektivt hitta de krav som ställs på systemet samt att den visar systemets beteende. • Tekniken är skriven med naturligt språk vilket vidare genererar att kravhanteringen förbättras. • Enkelt för en utvecklare att fortsätta på en redan befintligt Use Case. • Lättförstålig teknik vilket bidrar till bättre utvecklingsmöjligheter. • Use Case visar beteendet på systemet på ett enkelt sätt vilket bidrar till att relationerna till systemet tydliggörs. • Underlättar kommunikationen så att utvecklingen går smidigare. • Är en iterativ teknik vilket underlättar förändringar under projektets gång. • Use Case diagrammet generar bättre hantering av de krav som finns till ett system.	• Kan få för många Use Case delar under en och samma utveckling. • Det enda sättet att kunna vara säker på att Use Caset är korrekt är att gå genom Use Caset med användaren.

2.3.2 Use Case-Diagram

Ett Use Case-diagram visar enligt Dennis och Wixom (2000) samspelet mellan de externa användarna och systemet. Use Case-diagrammet skall vidare fånga företagets krav till systemet samt innehålla alla aktörer och de Use Case-modeller som använts under projektets gång.

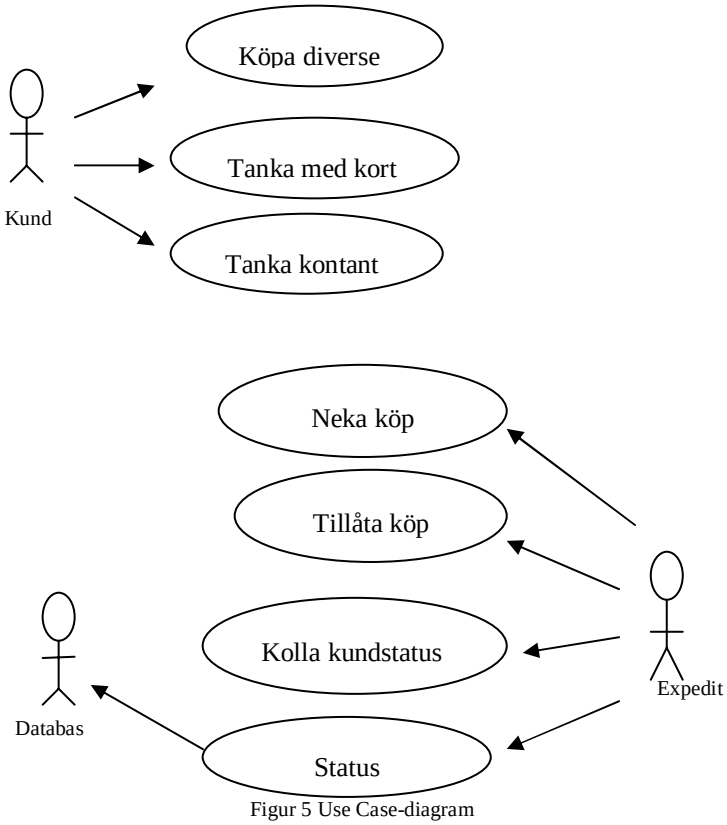
En anledning till att Use Case-digram eller Use Case i helhet är så effektivt att använda är enligt Leffingwell och Widrig (2000) att Use Case-diagrammets utseende alltid är likadant uppbyggt. Aktörerna i ett Use Case kan ses som någon eller något, exempelvis en person eller en databas. Aktörerna presenteras alltid som streckgubbar. Use Caset som innehåller det viktigaste av allt nämligen kraven, dokumenteras och presenteras ofta som ovalformade ringar, se figur 4. Pilen mellan Use Caset och aktörerna visar ofta att det finns en relation mellan Use Caset och aktören samt riktningen på relationen.

Use Case-diagrammet visar en helhet över vilka interaktioner som är viktiga att ha med, det vill säga, den fångar omgivningen vilket bidrar till en bra kravhanteringsprocess påstår Kulak (2000). Själva Use Caset kan ses som kraven i textform som definierar i detalj vad ett system måste uppfylla (Kulak 2000; Leffingwell 2000).



Figur 4 ett exempel av Use Case-modell

Vidare i figur fem presenteras ett exempel på hur ett Use Case-diagram över en bensinstation kan vara uppbyggd samt en kort förklaring till diagrammet.



Figur 5 Use Case-diagram

Utifrån bilden kan vi konstatera att Use Case-diagramet är uppbyggt av olika del Use Cases som tillsammans med aktörerna sammanställer den slutliga Use Caset, det vill säga den verkliga bilden över systemet som skall byggas. Varje Use Case betecknas här med dess namn som till exempel köpa diverse, tanka med kort och så vidare. Diagrammet visar tydligt att en kund skall både kunna tanka med bankkort eller med kontanter. Vill inte kunden tanka kan den även köpa andra saker från butiken. Relationen mellan kund och Use Caset visas med hjälp av pilarna riktning. En expedit kan både neka och tillåta ett köp. Vidare kan receptionisten alltid kolla upp kundens status ifall detta skulle behövas. expediten kan alltid se hur mycket bensin, eller annat som gått åt under dagen genom att koppla upp sig mot ”status” som är kopplad till en databas som ständigt uppdateras**.**

3 Problemområde

Det som kommer att presenteras i detta avsnitt är problemområdet, det vill säga en tydlig bild av problemet som finns när det gäller hantering av krav med hjälp av Use Case. Vidare kommer problempreciseringen och rapportens frågeställning samt ett argument till varför detta valts att presenteras. Kort därefter kommer författaren att berätta om rapportens avgränsning. På grund av resursbrist kommer författaren inte ha möjlighet att täcka in hela detta område i rapporten. Därför har enbart en del av detta område valts för att studeras. Slutligen kommer en kort beskrivning av den förväntade resultatet att presenteras.

Kravhantering är nästintill den viktigaste uppgiften i hela informationssystemutvecklings-processen. Dagens ekonomiska konsekvenser, hårda konkurrens och de enormt dyra applikationerna har medfört att det är viktigt att satsa på krav för att kunna behålla kunder så nöjda som möjligt.

Mjukvarusystem är väldigt komplicerade och detta medför i sin tur till att själva utvecklingen av mjukvarusytem kommer att bli väldigt problematiskt. För att lyckas med systemutvecklingsprojektet är det viktigt att hantera de krav som finns till systemet med noggrannhet (Regnell, 1999) Ett system som färdigutvecklas utan att tillfredsställa kundens behov kan få enorma konsekvenser. Systemet kan till exempel bli mycket dyrare än det var tänkt från början, detta för att utvecklaren blir tvungen att göra om delar av systemet (Rosenberg & Scott, 1999). Det finns olika typer av tekniker som utvecklaren kan använda för att kunna lyckas med kravhanteringen i ett systemutvecklingsprojekt. En möjlig teknik till att kunna hantera dessa krav med säkerhet kan vara Use Case (Macaulay, 1996).

Problem som ofta kan dyka upp i samband med Use Case och kravhantering är att det blir för många Use Case-modeller i ett systemutvecklingsprojekt (Regnell m.fl. 1995). Som det går att konstatera från tidigare kapitlen finns det även en hel del fördelar med Use Case. Dock framgår inte nackdelarna så tydligt som fördelarna i olika litteratur som studerats. Därför är det angeläget att i detta examensarbete undersöka för och nackdelar med UMLs Use Case i samband med kravhantering.

3.1 Problemprecisering

Det material som har presenterats tidigare i rapporten leder fram till frågor om valet av rätt teknik för kravhantering, detta har bidragit till examensarbetets frågeställning;

-Vilka för- och nackdelar innebär Use Case vid hantering av krav i informationssystemutveckling?

Tidigare i examensarbetet har det diskuterats mycket om krav och vikten av kravhantering. Tydliga tecken kan konstatera om hur viktig kravhanteringsprocessen är för ett systemutvecklingsprojekt. Vidare beskrivs en teknik som kan bidra till en bättre systemutveckling, UML: s Use Case. Det är då av intresse att undersöka vilka för och nackdelar tekniken Use Case bidrar med i samband med hantering av krav.

Syftet med detta examensarbete är att ta reda på de för och nackdelar som har upplevts av systemutvecklare som har arbetat med Use Case i samband med kravhantering. Det anses vara viktigt att ta reda på vilka för och nackdelar tekniken Use Case har i samband med kravhantering för att kunna vara säker på att tekniken inte genererar sämre resultat än om den inte använts.

3.2 Avgränsning

Som tidigare definierats i arbetet så finns det olika sorters tekniker och verktyg för att kunna utveckla ett informationssystem med.

Denna rapport fokuserar enbart på de tidigare faserna i livscykelmodellen, närmare bestämd verksamhetsanalys och systemanalys. Detta på grund av att det är under dessa faser som UML:s Use Case används som flitigast. Vidare kommer rapporten enbart att svara på de för- och nackdelar som Use Case medför vid kravhantering utifrån respondenter som arbetar på mer kända företag i Sverige.

3.3 Förväntat resultat

Det skrivs väldigt mycket om Use Case men dock inte mycket om nackdelar med denna teknik. Min teori om Use Case är att den innehåller brister som ofta inte nämns i olika litteraturer. Författaren till detta examensarbete skall genom att studera vetenskapliga artiklar och genomföra intervjuer ta reda på de för- och nackdelar som denna teknik bidrar med samt ta reda på om detta eventuellt behöver kompletteras.

4 Metod

I detta kapitel kommer författaren att presentera de metoder som valts till att lösa rapportens problemställning. Kapitlet inleds med beskrivning kring de valda metoderna samt en argumentation till varför dessa metoder valts. Vidare kommer en beskrivning över det tänkta tillvägagångssättet att presenteras.

4.1 Valda metoder

Det finns olika sätt att gå tillväga för att samla in information till ett arbete eller projekt. Patel och Davidsson, (1994) skriver bland annat om följande metoder:

- Intervju
- Litteraturstudie

Det är inte ovanligt att utvecklarna väljer att blanda dessa metoder för att kunna få ett så bra resultat som möjligt. En så kallad blandning av dessa två metoder kan medföra att författaren av denna rapport kan jämföra resultatet av intervjuerna med någon relevant litteratur för att kunna undersöka om det finns någon koppling till varför vissa respondenter svarat som de gjort. Det finns andra metoder som skulle kunna användas i detta examensarbete för att undersöka problemställningen. Exempel på en annan möjlig metod är enkätundersökning. Anledning till att en enkätundersökning inte användes i denna studie var för att det främst strävades efter en öppen samling av information. Enkätundersökning ansågs som en mer låst metod där intervjuaren inte skulle ha någon möjlighet att kunna påverka undersökningen genom att bland annat kunna förklara eventuella missuppfattningar.

4.1.1 Intervju

Metoden *Intervju* innebär formellt att en intervjuare, ställer ett antal frågor till en annan person. En intervju kan både ske hos respondenten och även via telefon. Det är viktigt att alltid ha en lista över begripliga frågor som skall ställas under intervjun för att processen skall gå så smidigt som möjligt (Patel & Davidson, 1994). Obegripliga frågor som inte kommer i rätt ordning bidrar till att respondentens intresse minskas som i sin tur leder till fel eller sämre resultat. Under en intervju kan intervjuaren ofta utveckla sina idéer på ett sätt som normalt inte hade gått (Wärneryd mfl, 1990).

För att kunna svara på rapportens problemställning med hjälp av metoden intervju kommer intervjuerna att ske både personligt, det vill säga hos respondenten samt genom telefon. Författaren till denna rapport är medveten om att dessa två intervjusätt inte är helt jämförbara. Under en telefonintervju kommer intervjuaren att gå miste om exempelvis respondentens kroppsspråk som ibland kan vara avgörande. För att lösa detta problem finns möjligheten att använda sig av den sista intervjufrågan nämligen kompletteringsfrågan (se bilaga 1). Metoden intervju är ett bra val i detta examensarbete då författaren får en kontakt med någon som arbetar eller har arbetat med tekniken i det verkliga livet. Författaren kan därefter utifrån problemställningen ställa relaterade frågor till intervjupersonerna för att få ut de fakta som är intressant för rapporten. Patel och Davidson (1994) menar att en intervju oavsett om det är en telefonintervju eller intervju på plats hos respondenten alltid kan styras. Författarna anser att information via intervjuer kan samlas in på två olika aspekter, nämligen genom standardisering eller strukturering.

Med *standardisering* menas hur mycket information som lämnas till intervjuaren med avseende på utformningen av frågorna och dess eventuella inbördes ordning (Patel & Davidson, 1994).

Med *strukturering* menar Patel och Davidson (1994) i vilken grad frågorna är fria för respondenterna så att de kan tolka dem. Respondenterna tolkar ofta frågor utifrån deras egna privata inställning eller erfarenheter.

En strukturerad intervju lämnar ofta enbart ett litet utrymme för respondenten att besvara på frågor. Det är inte heller omöjligt att kunna förutsäga de alternativa svaren i en strukturerad intervju. I en ostrukturerad intervju däremot har respondenten fritt utrymme för att besvara på frågorna hur denne själv vill (Patel & Davidson, 1994).

Eftersom examensarbetets problem är att ta reda på för och nackdelar med Use Case i samband med kravhantering har författaren för att kunna uppnå bästa resultat av intervjun valt att konstruera intervjufrågorna så öppna som möjligt, det vill säga valt en ostrukturerad intervju. Anledningen till detta för att kunna ge respondenten den frihet att svara i den utsträckning som denne själv önskar. Ifall utvecklaren utifrån behov skulle vilja vidare utveckla en fråga är det viktigt att frågorna för detta examensarbete inte är låsta av dess inbördes ordning. Anledningen till att en ostrukturerad intervju metod valds var för att det anses vara mer flexibel till att kunna svara på rapportens frågeställning. Utifrån ovanstående argument anser författaren att metoden intervju är rätt val till att kunna svara på rapportens frågeställning och syfte.

4.1.2 Litteratur

Den andra metoden som valts i denna undersökning är en litteraturstudie. När en litteraturstudie görs så hämtas information från böcker, rapporter och i vissa fall från tidskrifter (Patel & Davidson, 1994). Anledningen till att denna metod också valts var för att det finns en hel del skrivet inom ämnet som kan bidra till ett positivt resultat för rapporten. Det kan även ses som ett komplement till intervjuerna. För att kunna bedöma om de fakta som studerats är sannolikt måste de enligt Patel och Davidson (1994) värderas kritiskt. Detta innebär bland annat att kunna ta reda på när och hur dokumenten tillkommit samt varför och vilket syfte dokumenten har haft.

Litteraturstudie kommer att användas till att jämföra svaren som bearbetats fram via intervjuerna samt för att kunna hitta en orsak till varför vissa respondenter svarat som de gjort. Det är väldigt viktigt att dokumenten som informationen hämtas ifrån representerar en godtagbar beskrivning av den sökta verkligheten som undersökningen sker i. Denna rapport kommer att bestå av material som både stödjer och misstödjer vår egen uppfattning om ämnet och frågeställningen. Det anses viktigt att inte enbart presentera fakta som stödjer det egna tankesättet, dock enbart för att få en så verkligt resultat som möjligt.

Fördelen med litteraturstudie för detta examensarbete är att det finns mycket litteratur inom ämnet som medför att mycket information finns att inhämta till rapportens fördel. Genom litteraturstudier kommer författaren att komma i kontakt med all relevant information som är relaterad till rapportens ämne. Detta medför att författaren får mer kunskap om ämnet som i sin tur leder till att denne får en bättre utvecklat teoretiskt bild av problemet. Litteraturen i resultatet kommer enbart att ses som en komplement till intervjuerna och kommer att användas till att undersöka bland annat hur och varför författaren fått vissa svar av de tillfrågade respondenterna. Anledningen till att litteraturen används på detta sätt är för att få ett mer trovärdigt slutresultat.

4.1.3 Tänkt tillvägagångssätt

Att hitta anpassade respondenter för intervjufrågorna har författaren valt i första hand att använda sig utav befintliga kontakter och i senare skede kommer Internet att användas. Anledningen till att privata kontakter använts i första hand har varit på grund av resursbrist. Det anses vara enklare att börja söka upp respondenter via personligt kontakter än annat sätt. Anledningen till att Internet valts som sökmetod är att Internet idag är ett viktig kommunikationsmedium för företag. Det är därför lättare att leta efter företag med tanke på den breda sökalternativen som Internet erbjuder. Sökningen kommer enbart att koncentrera sig på mer välkända företag i Sverige. Detta dock enbart för att kunna komma i kontakt med systemutvecklare som arbetar med större projekt för att kunna få en så bred syn som möjligt på rapportens problemområde.

Efter att valda företag hittas på Internet kommer de att kontaktas via telefon för att ta reda på om de är intresserade att ställa upp i en eventuell intervju. De respondenter som inte befinner sig i samma län som författaren kommer att intervjuas via telefon. Anledningen till att detta är dels på grund av resursbrist och dels det långa avståndet till respondenterna. Vidare kommer ett introduktionsemail innehållande bland annat bakgrund av problemet, varför undersökningen görs samt tider då en eventuell intervju skulle passa att skickas till de intresserade företagen. För att inte gå miste om någon viktig information under intervjun kommer författaren att spela in alla intervjuer för att senare kunna studera dem.

Efter att intervjuerna är slutförda och sammanställda kommer en kopia av den sammansatta intervjun att skickas till respektive respondent. Anledningen till detta är för att kunna ta reda på om författaren har fått rätt bild och svar av dem som har intervjuats. Vidare kommer den bearbetade informationen även att studeras i samband med litteraturer. Författaren till detta examensarbete kommer att försöka hitta argument till varför vissa respondenter tycker eller upplever något som de gör bland annat genom olika litteraturer. Anledningen till att intervjuerna görs först är att författaren vill ha en så bred bild som möjligt av det verkliga livet för att senare kunna undersöka detta mer teoretiskt. För att hitta rätt litteratur som kan stödja intervjuerna samt frågeställningen används i första hand Internet. Författaren kommer att leta efter vetenskapliga artiklar i de olika databaser som erbjuds i högskolebiblioteket i Skövde. Därefter kommer även vid behov en del böcker om Use Case att studeras. Vikten av litteraturdelen läggs på artiklar som undersöker fall där Use Case har varit inblandad. Det insamlade materialet genom intervjuerna kommer att analyseras med hjälp av teori. Olika litteraturer kommer att studeras för att kunna ta reda på varför vissa respondenter svarat som de gjort och vad anledningen till detta kan vara.

5 Genomförande

I detta kapitel kommer författarens faktiska tillvägagångssätt utifrån de metoder som presenterades i kapitel 4, värdering av det insamlade informationen samt erfarenheter som ligger till grund för att lösa examensarbetes problemställning att presenteras.

5.1 Tillvägagångssätt

Att hitta en hjälpsam respondent via personlig kontakt var som författaren förmodade inga större problem. Dock visade sig att sökningen via Internet var mer komplicerad än det från början verkade. Författaren började söka företag efter nyckelord och upptäckte snabbt allt för många träffar och sökningen blev tvungen att specificeras. En specificering där sökningen skedde via bransch visade sig inte heller vara så framgångsrikt. Även i detta fall fick författaren för många träffar. Vad som gjordes här näst var att ringa de företag som verkade intressanta utifrån den lista som bearbetats fram via sökningarna på Internet. Efter att ha ringt till några företag visade det sig att det inte var lika lätt att hitta relevanta respondenter som det var att få tag på företag. Det svåra var att hitta en respondent som både hade kunskap om ämnet och samtidigt tid att ställa upp på en intervju. Efter att ha kontaktat 23 företag var sju av dem tillräckligt relevanta att gynna rapportens problemställning. Vad som gjordes härnäst var att ett introduktionsemail (bilaga 2) skickade till de intresserade respondenterna. Till rapportens besvikelse svarade enbart tre av de sju tillfrågade respondenterna på emailet och berättade när en eventuell intervju kunde äga rum. Vidare gjordes flera försök till att kontakta de respondenter som inte hörde av sig på det första emailet och tre av dem tackade nej till undersökningen samtidigt som den fjärde tackade ja till slut. Dessa fem respondenter och den tilltänkta litteraturen skall kunna besvara på rapportens problemställning. Eftersom en av respondenterna befann sig geografiskt sätt lång ifrån författaren var det enda alternativet att utföra en telefonintervju. Författaren är medveten om att det är bättre att göra en intervju på plats då intervjuaren annars kan gå miste om respondenters kroppsspråk som till exempel kan visa om de är osäkra på de svar de anger. Dock anses detta i det här fallet inte som något större problem då intervjufrågorna kommer att vara ställda på ett sätt att det går att ställa följfrågor till respondenten ifall tveksamheter upptäcks. Fyra av fem intervjuer har spelats in på band. I den sista intervjun som var en telefonintervju saknades bandspelare så möjligheten till att spela in intervjun fanns inte. Intervjun skrevs då ner på papper samtidigt som respondenten berättade om sina erfarenheter utifrån frågorna. Bearbetningen av intervjuerna gjordes så att författaren först lyssnade på banden och läste anteckningar efter telefonintervjun en gång efter alla intervjuer. Sedan började författaren att noga lyssna på banden igen samtidigt som relevant fakta skrevs ner på papper. Vad som gjordes näst var att en sammanfattning av de som författaren kommit fram till utifrån respondenternas svar mailades till respondenterna för att se att det inte uppstått några oklarheter. Efter besked från respondenterna sammanställdes sammanfattningarna och presenterades under kapitel 5.2. Samtliga intervjufrågor presenteras i bilaga 1.

5.2 Materialpresentation

Under denna del av rapporten kommer författaren att presentera de material som tagits fram via den metod som diskuterats under kapitel 4. Författaren kommer här att utifrån respondenternas synpunkter sammanfatta de olika intervjuerna. Presentationen kommer att ske genom intervjufrågorna (se bilaga 1) som är uppdelade i introduktionsfråga, huvudfråga samt en kompletterande fråga. Materialet kommer att presenteras fråga per fråga. Anledningen till detta är att kunna visa hur de olika respondenterna svarat på de olika frågorna så att likheter och olikheter visas på ett tydligt sätt. Eftersom respondenterna valt att vara anonyma kommer inga namn eller företag att presenteras. Materialet som presenterats här under har även viss grav av analys is. Vidare kommer materialet att ligga till grund för kommande kapitel nämligen analys och resultat.

5.2.1 Introduktionsfråga

Allmän information

Alla respondenter arbetar på välkända svenska företag. Respondet ett, två, fyra och fem är i grunden systemvetare. Respondent ett har varit verksam i IT-branschen i över 15 år och har därmed den längsta erfarenheten av alla respondenter som deltagit i denna undersökning. Respondenten har under de senaste nio åren aktivt arbetat med Use Case och kravhantering. Respondent två har precis som respondent fem varit aktiv i IT-branschen i över fyra år. Respondent två har under de tre senaste åren arbetat med Use Case och kravhantering. Respondent tre är i grunden datavetare och har ca fem års erfarenhet av systemutveckling och Use Case. Respondent fyra har varit verksam i IT-branschen sedan 1992. De senaste sex åren har denne deltagit i olika projekt där Use Case använts som teknik. Respondent fem har deltagit i olika projekt där Use Case använts, dock ej de senaste två åren.

5.2.2 Huvudfrågor

Kravhanteringens betydelse

Under denna del var alla respondenterna positivt överens. Det var ingen av respondenterna som tvekade på svaret. Respondent 1 tyckte att det är viktigt att satsa på krav för att det är ett sätt att se att rätt system byggs och samtidigt kunna verifiera kundens behov. Respondent två och fyra svarade att krav är viktig eftersom det är kunden som betalar för den. De tycker att det är viktigt att satsa på kravhantering för att kunna få fram den produkt eller system som kunden eftersträvar, annars finns det ingen ide med att utveckla ett system för kunden. Respondent två påpekar även att denne ofta använder kraven som riktlinjer för att kunna slutföra ett system som verkligen eftersträvas. För respondent tre och fem var det självkart att satsa på krav, de såg inget annat alternativ till detta. Alla respondenter var överens om att kravhantering var ett måste men respondent fyra påpekade att kravhantering är något som eftersträvas men inte alltid är så enkelt att uppfylla.

Use Case som teknik för systemutveckling

Alla respondenter tycker att Use Case är något som gynnade både systemutvecklaren och utvecklingen. Respondent ett svarade att Use Case ofta gynnar utvecklaren för att tekniken tillåter utvecklaren att kunna avgränsa sig så att rätt krav behandlas. Respondent ett två, fyra och fem svarade att Use Case är en bra teknik för systemutveckling för att utvecklaren relativt enkelt kan se flödet på utvecklingen. Genom att se flödet kan utvecklaren hela tiden se vad som bör implementeras och vad som kan vara överflödigt. Respondent två tyckte att tekniken var bra även för icke kunniga personer och de skulle utan större problem kunna förstå processen och kraven. Det är även enkelt för utvecklaren att genom exempelvis bilder på Use Case-diagram kunna visa vad som skall byggas. Respondent tre och fyra menade att tekniken ofta hjälper att hitta de roller som finns i ett system. Genom att ha definierade roller kan utvecklaren enklare fortsätta med utvecklingen. Respondent fem tycker att tekniken är bra då den visar en översikt över själva utvecklingen så att det blir enklare att följa upp utvecklingen.

Use Case och livscykeln

Respondent två såg Use Case som en iterativ teknik som kan återkomma i de flesta faser. De andra fyra respondenterna var överens om att Use Case mest kommer till användning i de tidigare faserna i livscykel processen. Respondent två säger inte emot någon annan respondent men upplever dock att Use Case i en liten mäng kan komma till användning även i senare faser.

Förväntningar av Use Case

Respondent ett, tre och fem använde sig av Use Case främst för att enkelt visa olika roller eller aktörer till ett system. Även respondent fyra tyckte att den visade roller på ett bra sätt men denne använde Use Case för att kunna få lite hjälp eftersom respondenten anser att Use Case är en bra teknik till att dela resurser med bland annat andra utvecklare. Respondent två och fyra strävade efter att kunna få ut de olika rollerna och genom detta kunna lista ut flödet på systemet. Dessa respondenten anser att genom att ha koll på flödet blir det automatiskt mycket bättre interaktion mellan de olika parterna i utvecklingen. En annan sak som respondent två och fyra strävade efter var att genom detta kunna få ut bättre samarbete mellan de olika parterna vid en eventuell utveckling. Respondent tre använder Use Case för att med hjälp av tekniken kunna komplettera eventuella missuppfattningar. Respondenten tyckte som respondent två, att tekniken ofta hjälper utvecklare att lättare kunna kommunicera med varandra. Vad som eftersträvades av respondent fem var att Use Case skulle medföra till en bättre och enklare översikt över den totala utvecklingen. Efter att respondenten blivit tillfrågat vad denne menar kompletterades svaret med att denne tycker att diagrammen som används i tekniken ofta hjälper utvecklarna att lättare ha koll på det som händer i utvecklingen och det är det som strävas av Use Case när respondenten använder sig av tekniken.

Use Case, en teknik för kravhantering

Respondent ett, tre och fem anser att tekniken visar de olika rollerna till systemet som i sin tur bidrar till att det blir enklare att hantera de krav som ställs på systemet. Vidare säger även respondent ett att anledningen till att det blir enkelt är för att det är relativt lätt att involvera slutanvändaren så att denne kan se ifall något saknas i Use Case-modellen. Genom att kunna involvera användaren anser respondenten att det både blir lättare att hantera de krav som ställs på systemet samtidigt som utvecklaren då vet vem som är användaren. Respondent två menar att Use Case visar flödet och samtidigt bidrar till en enklare kommunikation mellan de olika inblandade parterna. Detta kan bidra till att kraven hanteras på ett enklare och smidigare sätt. Respondent tre och fyra påpekar båda att det är arbetssättet som gör det hela enklare. Ett enklare arbetssätt gör att det blir lättare att hantera de krav som kunden ställt och bidrar till ett bättre slutresultat genom att kraven hanteras på ett bättre sätt. Respondent fem anser att Use Case hjälper utvecklaren att utvinna krav på ett bra sätt och med hjälp av tekniken kan användaren hela tiden kunna se vad som händer. En anledning till detta enligt denna respondent är bland annat Use Case-diagram.

Fördelar med Use Case

Mycket av svaren som gavs här var en upprepning av de tidigare förklarade frågorna. Författaren kommer ändå att presentera en sammanfattning av det som diskuterades under denna fråga.

Alla respondenter var utan tvekan överens om att Use Case var en bra metod. Respondent ett påpekade att det var det bästa de gjort på deras företag när de började arbeta med UML och Use Case. Respondenten berättade vidare att Use Case har medfört att det nu är mycket enklare att arbeta med kravhantering då det blivit enklare att involvera kunden. Dock ville respondenten att kunden skulle ha mer frihet och även kunna ändra i Use Caset. Detta påstående anses inte relevant då ingen av de andra respondenterna nämnde något i närheten av detta. Respondent två tycker att Use Case är en mycket bra teknik för det är väldigt lätta att kunna följa utvecklingen genom bland annat Use Case-diagram. Det är inte lika lätt att tappa bort sig i projektet som det är när andra tekniker används. En annan stor fördel som tekniken har är enligt respondenten att det är lätt att ta över ett befintligt Use Case och fortsätta på den. Detta gör att tekniken även kan passa för vidareutveckling likaväl som nyutveckling.

Respondent två tycker att Use Cases sätt att planera bland annat med hjälp av Use Case-diagram är ett sätt som fått tekniken att blomstra bland företag. Även respondent tre håller med detta tankesätt och bidrar med att tankesättet underlättar arbetet på ett sätt som inte tidigare varit möjligt. Respondent två och tre anser att Use Case har en bra tankesätt som medför att kommunikationen mellan utvecklarna underlättas som senare i sin tur bidrar till att utvecklingen blir mycket enklare att bearbeta. Eftersom Use Case underlättar på så många sätt blir det genast mycket lättare att satsa på kravhanteringen säger respondent tre.

Respondent fyra anser också som respondent två och tre om Use Case-diagram och berättar mycket om enkelheten att dela resurser mellan sig. Denna respondent tycker att det är mycket lättare att hjälpa varandra och ta över ett befintligt Use Case och fortsätta på det. Respondenten anser även att Use Case sätt att dokumentera allt som händer är ett unikt sätt som gör att utvecklaren inte missar något viktigt. Dock är det viktigt att det inte blir för många Use Case delar som förvirra utvecklare påpekar respondenten. En fördel som är värd att nämna enligt respondent fyra är att Use Case underlättar förståelsen bland annat genom dokumentationssättet som den har. Detta medför då i sin tur att utvecklingen blir mindre komplex och lättare att fullfölja. Respondent fem tycker att den störta fördelen med Use Case

är att det är lätt att utvinna krav med den. Respondenten berättar att eftersom tekniken bidrar till att det enklare går att se strukturen och funktionaliteten på systemet så kommer utvecklaren att ha en bättre översikt över systemet som då medför att denne enklare kan hantera de krav som ställs på den.

Vidare kommer de viktigaste fördelar om Use Case och kravhantering som påträffats under detta kapitel att sammanställas och presenteras i punktform. Anledningen till att dessa presenteras i detta kapitel här är för att senare i nästa kapitel tillsammans med de nackdelar som framkommer kunna jämföra dessa med de för och nackdelar som framkommit i litteratur som studerats.

- Use Case visar flödet på systemet på ett enkelt sätt vilket bidrar till att kravhanteringen förenklas.
- Use Case- diagrammet förtydligar flödet som i sin tur bidrar till att kravhanteringen förbättras.
- Diagrammet bidrar till att rollerna definieras på ett enklare sätt.
- Tekniken bidrar till att det går enkelt att hitta de krav som ställs på systemet.
- Förenklar kommunikationen som underlättar för alla inblandade parter.
- Ganska enkelt att involvera slutanvändare.
- Enklare arbetssätt som bidrar till lättare och bättre kravhantering.
- Underlättar förståelsen vilket bidrar till att der är lätt att hjälpas åt och eller ta över en redan befintlig Use Case.
- Bidrar till att den slutliga funktionaliteten blir bättre då Use Case bidrar till bättre hantering av de krav som ställs på systemet.
- Use Case är en iterativ teknik där utvecklaren kan ändra i den under projektets gång.

Nackdelar med Use Case

Respondent ett tycker att tekniken kan upplevas som svår att lära in. Anledningen till detta är att den är annorlunda mot vad andra traditionella tekniker som de är vana med är. Respondenten anser även att tekniken inte alltid räcker som den enda teknik för utvecklingen. Det behövs mycket kunskap om systemutveckling för att kunna lyckas utnyttja tekniken på bästa möjliga sätt. En oerfaren utvecklare kommer att ha svårt att använda sig av tekniken. De flesta respondenter var överens om detta resonemang. Respondent två anser att Use Case ibland kan få dåligt rykte eftersom den är så annorlunda mot vad de andra traditionella teknikerna är. En anledning till detta kan enligt respondenten vara att Use Case saknar direkta riktlinjer som utvecklarna kan följa. Detta kan ibland ställa till problem för en utvecklare ifall denne fastnar på någon punkt i utvecklingen menar respondent två. Även respondent fyra håller med respondent två angående saknaden av riktlinjer. Vidare sa respondent två att tekniken är ganska lätt att missbruka. Eftersom den saknar riktlinjer kan utvecklare tolka som de vill under projektets gång och det är inte alltid så lyckat för slutresultatet. Dessa nackdelar bidrar till att kraven ibland kan vara svåra att placera. Kraven skall dock inte få för mycket uppmärksamhet men inte heller för lite menar respondenten.

Alla respondenter är överens om att det ibland kan bli för många Use Case-delar i ett och samma projekt. En stor nackdel med tekniken är att det inte finns något bra sätt att hantera dessa Use Case på. Detta kan vara anledningen till att tekniken kan vara svår för oerfaren utvecklare påstår respondent tre och fyra. De oerfarna skulle lätt missa viktiga detaljer då antalet Use Case-delar blev för många. Respondent fyra och fem anser att en nackdel som Use Case har är att utveckling med tekniken kan bli för tidskrävande. Dessa respondenter hade

ingen riktig förklaring till detta, dock trodde respondent fyra att det blev så när antalet Use Case-delar ökade i ett och samma projekt. Respondenten menade att det skulle bli för svårt att hantera de olika Use Case-delar.

Vidare kommer de viktigaste nackdelar om Use Case och kravhantering som påträffats under kapitlet att presenteras i punktform.

- Svår att lära in, dock enbart för att den skiljer sig mycket från de andra traditionella teknikerna de är vana med.
- Enbart kunskap om tekniken räcker inte för att klara av en utveckling på bästa möjliga sätt.
- Saknar direkta riktlinjer. Medför till att det blir svårt för nybörjare att använda sig av tekniken.
- Eftersom den saknar riktlinjer finns risken att tekniken missbrukas.
- Tenderar att få för många Use Case delar som inte har något bra sätt att hanteras på. Vidare leder detta till att tekniken bli tidskrävande.
- Kräver mycket dokumentering.

Komplettering av Use Case

Respondent 1 tycker att Use Case borde vara uppbyggd på ett sätt att användaren är mer delaktig. Mer samarbete mellan respondent och utvecklaren önskas. Respondenten anser att även användaren bör kunna ändra i Use Caset. Detta skulle enligt respondenten förbättra kravhanteringen då kundens ändringar ses som förslag för vidare utveckling av utvecklarens idé. Respondent två önskade att det fanns domänmodellering i Use Case-tekniken. Detta dock enbart innan själva modelleringen har påbörjats. Respondent tre och fyra anser att de saknar klara riktlinjer att fullfölja. De skulle vilja att det fanns regler eller riktlinjer för utvecklarna att kunna följa för att få utvecklingen att gå så smidigare. Detta skulle även bidra till att de mindre oerfarna utvecklarna skulle få en möjlighet att kunna hantera de olika Use Case-delarna.

5.2.3 Kompletteringsfråga

Betyg till Use Case

Att betygsätta Use Case har enbart använts till respondent fem. Anledningen till detta var på grund av att respondentens svar i två fall upplevdes som osäker. Respondenten gav Use Case betyget 7. Efter att en sammanfattning gjorts av respondentens intervju kunde författaren av rapporten se en helhet om vad respondenter tycker om Use Case och kravhantering. Respondentens svar anses stämma överens med det som författaren av rapporten upplever och därför anses materialet som kommer från denna respondent som acceptabelt och kan användas vidare i analysen.

5.3 Värdering av insamlat material

Intervjuerna som gjordes anses enligt författaren ha en hög trovärdighetsgrad. Respondenterna svarade inte enbart utifrån vad de själva tyckte utan även mycket utifrån erfarenheter. Detta anses enligt författaren vara värdefullt. Det som gjorde intervjuerna mer trovärdigt var att nästan alla respondenter tyckte likadant om vissa saker angående Use Case. Efter att materialet i efterhand bearbetades med hjälp av litteratur ansåg värdet vara i en sådan grad att kunna besvara rapportens problemställning.

6 Analys

Under detta kapitel kommer det insamlade materialet från kap 5 att värderas för att ta reda på vilket sätt materialet kan svara på problemställningen. För att kunna utvärdera materialet så bra som möjligt kommer respondenternas svar jämföras med varandra och i vissa fall även kopplas tillbaka till informationen i bakgrunden.

Som en introduktion till de för och nackdelar som skall analyseras utifrån de tidigare gjorda intervjuerna kommer en sammanställning innehållande två tabeller om för- respektive nackdelar från kapitel 2.3 samt 5.2 att presenteras. Dessa tabeller kommer vidare i kapitel 6.4 att diskuteras. Intervjuerna som skall analyseras kommer att delas upp i tre olika delar som senare i kapitel 7 leder till ett resultat som kommer att presenteras. Eftersom intervjufrågorna var ställda på ett sådant sätt att respondenterna kunde svara fritt, har alla de ställda frågorna använts i de tre olika delarna men dock enbart i den utsträckning där de passat in i sammanhanget.

6.1 Use Case och kravhantering

Alla de tillfrågade respondenterna tyckte att det var viktigt med kravhantering i ett systemutvecklingsprojekt vilket även Flensburg (1987) hävdar. Det som framkom av respondenternas svar var att kravhantering är viktig inom systemutveckling då det är genom kravhantering som kundens önskemål uppfylls och kunden får ett system som den eftersträvar. Use Case kan ses som en utmärk teknik för att kunna hantera de krav som finns på ett system menar Muller (1997). Även de tillfrågade respondenterna svarade att de ansåg att Use Case var en bra teknik som hjälpte utvecklaren att hantera de krav som fanns på ett system. Anledningen till detta är dels enkelheten att förstå Use Case och dels Use Case-diagrammet som hjälper utvecklarna att bland annat enkelt följa flödet på systemet**.** Fyra av de fem tillfrågade respondenterna ansåg att Use Case användes i de tidigare faserna i livscykeln i ett systemutvecklingsprojekt. Anledningen till att respondenterna tycker så är på grund av att det är under de tidigare faserna som kravhanteringen bearbetas, samt att Use Case är en teknik som kan användas för kravhantering. Dock tyckte respondent två att tekniken var återkommande i de flesta faserna i livscykeln. Vid en närmare analys av respondentens svar visade det sig att respondenten egentligen inte motsäger de andra respondenterna. Respondent två upplever att tekniken även kan komma att användas i de senare faserna. Anledningen till att respondenten tycker så kan bero på att Use Case-tekniken är en iterativ teknik**.** Även Dennis och Wixom (2000) skriver att Use Case-tekniken är en iterativ process vilket innebär att utvecklaren alltid kan gå tillbaka i de olika faserna och ändra i Use Caset.

6.2 Fördelar Use Case

En sammanställning av de fördelar som påträffats tidigare i kapitel 2.3 samt 5.2 kommer nedan att redovisas i tabell 1[3].

Fördelar	**Litteratur**	**Respondenter**
Tekniken är skriven med naturligt språk vilket vidare genererar att kravhanteringen förbättras.	X	
Tekniken underlättar förståelsen vilket bidrar till att det blir lättare att samarbeta och eller fortsätta på en redan befintlig system.	X	X
Lättförstålig teknik som har enkel arbetssätt vilket kan generera bättre kravhanteringsmöjligheter.	X	X
Tekniken visar flödet på systemet vilket gynnar alla de inblandade parterna. Samtidigt generar detta till effektivare kravhantering.	X	X
Genom att flödet blir tydligare bidrar detta till att kommunikationen förenklas vilket resulterar till en effektivare utveckling.	X	X
En iterativ teknik vilket medför att utvecklaren har möjlighet att gå tillbaka i projektet och ändra i utvecklingen.	X	X
Use Case- diagrammet bidrar till att rollerna till systemet blir tydligare. Detta leder i sin tur till att utvecklingen blir enklare.	X	X
Diagrammet bidrar till att rollerna definieras på ett enklare sätt.		X
Bidrar till att den slutliga funktionaliteten blir bättre då Use Case bidrar till bättre hantering av de krav som ställs på systemet.		X
Ganska enkelt att involvera slutanvändare.		X

Tabell 1: En sammanställning av de fördelar som framkommit under litteratur samt intervjustudierna.

Tabell 1 visar tydliga likheter från respondenterna och litteraturen, samt även en del olikheter. En tydlig likhet som kan konstateras både från respondenter samt litteratur var att Use Case är en teknik som underlättar förståelsen för utvecklaren som i sin tur bidrar till en enklare utveckling.

Något som inte stämde överens mellan litteraturen och respondenterna var enkelheten med att involvera slutanvändare samt att tekniken är skriven med naturligt språk. Anledningen till dessa olikheter kan utvecklarens erfarenheter samt intresse för tekniken vilket vidare kommer att diskuteras i kapitel 6.4. Att tekniken har ett naturligt språk kan vara anledningen till att det blir enklare att involvera slutanvändarna eller vise versa.

Alla respondenter hade överensstämmande en positiv syn om att Use Case var en teknik som gynnade systemutvecklingen på ett eller annat sätt. Anledningen till att respondenterna upplever tekniken som positiv kan bero på att Use Case är en teknik som är lätt att förstå och som samtidigt hjälper utvecklaren att bland annat genom Use Case-diagram hitta de krav som ställs på ett system. Detta är även något som Muller (1997) poängterar med Use Case.

[3] Markering X visar om fördelen nämnts av litteraturen respektive respondenterna.

6 Analys

Tre av de fem tillfrågade respondenterna anser att Use Case-tekniken är en bra teknik för systemutveckling eftersom den ofta visar de olika roller som finns i ett system. Anledningen till att dessa respondenter tycker så är på grund av att tekniken kan beskrivas med naturligt språk för att kunna identifiera systemets beteende (Fantechi m.fl. 2003). De två resterande respondenterna nämligen respondent ett och två menar att Use Case-tekniken ofta bidrar till att det blir enklare att involvera användaren genom att rita upp Use Case-diagram som visar flödet på systemet. Respondenterna menar att Use Case-diagrammet ofta hjälper dem då diagrammets uppbyggnadssätt alltid är likadant. Detta bidrar till en bättre igenkänning och samtidigt bättre förståelse för Use Case-tekniken**.** Även Leffingwell och Widrig (2000) menar att en fördel med Use Case-tekniken är dess utseende som alltid är likadant uppbyggd.

Respondenterna två och tre anser att Use Case förenklar kommunikationen mellan utvecklarna. Anledningen att respondenterna upplever på detta sätt kan vara att under de projekt som Use Case använts har utvecklingen gått effektivare än under de projekt som slutfördes med andra tekniker. Respondenterna anser att de projekt där Use Case varit inblandad har stött på mindre problem. Detta för att Use Case-tekniken bidrar till att kommunikationen mellan de olika inblandade parterna med hjälp av bland annat Use Case-diagram blir enklare. Vidare leder detta till bättre och effektivare systemutveckling som medför till säkrare hantering av de krav som ställs på systemet. Även Dennis och Wixom (2000) menar att Use Case bidrar till att samspelet med de olika parterna förbättras. De andra tillfrågade respondenterna nämnde att enkelheten av Use Case medför till att en systemutvecklare utan några problem kan fortsätta på ett redan påbörjat Use Case. Detta kan dock tolkas som en förbättring av kommunikationen då systemutvecklaren förstår det befintliga Use Caset utan större problem. Anledningen till detta är att Use Case-tekniken är en teknik som förenklar kommunikationssättet som inte enbart behöver ske muntligt.

Respondenternas helhetsintryck av Use Case uppfattas som positivt då alla på ett eller annat sätt tycker att Use Case är en teknik som bidrar till att utvecklingen förenklas. Detta medför senare i sin tur till att kravhanteringen blir enklare och resulterar till ett bättre utvecklat system. Use Case har under de senaste åren blivit mer efterfrågat. Detta på grund av att Use Case fokuserar på de funktioner som påverkar användarens krav, det vill säga det funktionalitet som en kund eller användare vill ha på ett system. Användarens eller kundens mål kan ses som Use Casets mål, som bidrar till att Use Case resulterar till en bättre utvecklingsmöjlighet, där användarens mål med systemet kommer med hjälp av Use Case att uppfyllas på ett mer korrekt sätt. (Annie m.fl. 1999).

6.3 Nackdelar Use Case

En sammanställning av de nackdelar som påträffats tidigare i kapitel 2.3 samt 5.2 kommer nedan att redovisas i tabell 2[4].

Nackdelar	**Litteratur**	**Respondenter**
Svår att lära in, dock enbart för att den skiljer sig mycket från de andra traditionella teknikerna som utvecklarna är vana med.		X
Enbart kunskap om tekniken räcker inte för att klara av en utveckling på bästa möjliga sätt.		X
Saknar direkta riktlinjer. Medför till att det blir svårt för nybörjare att använda sig av tekniken.		X
Eftersom den saknar riktlinjer finns risken att tekniken missbrukas.		X
Tekniken tenderar att få för många Use Case- delar vilket bidrar till att den kan bli tidskrävande.	X	X
Kräver mycket dokumentering.		X
Det enda sättet att kunna vara säker på att Use Caset är korrekt är att gå genom Use Caset med användaren	X	

Tabell 2 : En sammanställning av de nackdelar som framkommit under litteratur samt intervjustudierna.

Till skillnad från tabell 1 så skiljer sig svaren på tabell 2 avsevärd. Anledningen till detta kan bero på olika saker och en del av dessa kommer att diskuteras i kapitel 6.4.

Under de insamlade materialen genom intervjuerna framkom det att Use Case i vissa fall ansågs som en udda teknik att lära in. Anledningen till att vissa respondenter uppfattare Use Case som en udda teknik kan vara för att teknikens arbetssätt skiljer sig mycket från de vanliga traditionella teknikerna som många utvecklare är vana med. Detta kan leda till att utvecklaren får svårigheter att sätta sig in i tekniken. Respondent två menar att en av de största nackdelarna som denne upplevt är just det att Use Case skiljer sig mycket från de traditionella teknikerna. Exempel på en skillnad kan vara striktare riktlinjer som Use Case saknar. Riktlinjer kan vara regler eller steg för steg beskrivningar som kan hjälpa utvecklaren genom utvecklingen ifall denne skulle få svårigheter eller stöta på problem under projektets gång**.** Närmare analys av respondenten visar att denne inte har så mycket erfarenhet av Use Case och därför kan det kännas svårt att använda sig av en teknik som inte har riktlinjer som kan hjälpa utvecklaren på vägen. Dessa problem kan enligt respondenterna förvirra utvecklaren vilket bidrar till att viktiga detaljer i kravhanteringen kan utelämnas.

Dennis och Wixom (2000) menar att i projekt där Use Case kommer till användning kan projektet i vissa fall bli för svårhanterligt på grund av att det kan bli för många Use Cases att ta hand om. Detta påpekade även de respondenter som intervjuades och tyckte att det var svårt att hantera tekniken när antalet Use Cases blev för många. En anledning till detta kan vara just att Use Case saknar riktlinjer som bidrar till att utvecklaren kan tappa kontrollen över de olika Use Cases. Vid en närmare analys av respondenterna visade det sig att det inte var någon som sa emot detta. Att komplettera tekniken med riktlinjer bidrar till att utvecklaren enklare kan följa tekniken från början till slut och därefter kunna åstadkomma en bättre utveckling. När det blir för många Use Cases att ta hand om kommer utvecklaren ofta på grund av resursbrist inte kunna koncentrera sig lika mycket på alla Use Cases. Detta leder till att vissa Use Case då

[4] Markering X visar om nackdelen nämnts av litteraturen respektive respondenterna.

förbises eller bearbetas mindre vilket kan resultera till sämre utveckling, bland annat i form av kravhantering.

Det framkom av två respondenter att en utveckling med Use Case kan bli tidskrävande. Efter närmare analys av dessa respondenter visade sig att anledningen till att respondenterna tycker så beror på att antalet Use Cases kan bli för många och att tekniken kräver mycket dokumentering som leder till att projektet blir långvarig.

6.4 Diskussion kring tabellerna

Syftet med de tabeller som presenterats tidigare i detta kapitel var att visa en sammanfattning av de tydligaste för och nackdelar med Use Case i samband med kravhantering som framkommit under litteraturstudien respektive intervjuerna under examensarbetes gång. Innehållet i tabellerna är en sammanfattning av de sammanställda för och nackdelarna som finns förklarat i punktform på respektive del i rapporten. Det är viktigt att påpeka att den information som är framtagen inte behöver i alla lägen och situationer vara på detta sätt. Det kan i speciella situationer framkomma att vissa av de för och nackdelar som framkommit i tabellen inte stämmer överens med utvecklares tankar och idéer. Ifall detta skulle inträffa kan det bero på att de för och nackdelar som framkommit i denna rapport utspelar sig på en ganska liten målgrupp och erfarenheter.

Tabell 1 och 2 kan användas till att snabbt kunna jämföra teori och praktik med avseende på fördelar och nackdelar med Use Case i samband med kravhantering. Tabellerna visar en tydlig sammanställning av de för och nackdelar som framkommit med Use Case i samband med kravhantering i detta examensarbete. Dessa tabeller kan både användas av kunniga samt icke kunniga personer. Informationen i tabellerna kan hjälpa olika individer i olika situationer. Exempelvis kan en utvecklare genom att studera tabellerna bedöma lämpligheten med att använda sig av Use Case i en viss situation. Även så kallade icke kunniga personer kan förbättra sin kunskap om Use Case och kravhantering genom att studera dessa tabeller. Tabellerna kan även användas till för en snabb överblick om Use Case och kravhantering för personer som av någon anledning inte kan läsa hela rapporten.

De fördelar som finns med tekniken i samband med kravhantering ser i stort sett likadant ut både i teori och praktik, medan nackdelarna skiljer sig markant. Anledningen till att nackdelarna ser ut som de gör kan bero på att teorin inte tar upp alla nackdelar som egentligen finns. Detta för att till exempel kunna höja teknikens status. Detta leder då till att utvecklarna själva under tiden de arbetar med tekniken kommer att komma på nackdelar som inte står i olika litteraturer som finns om Use Case och kravhantering. Det kan anses som oacceptabelt då vanligt framkommande nackdelar som ibland kan påverka det slutliga resultatet i ett projekt inte nämns i litteraturen. Att fördelarna ser ut som de gör anses enligt författaren av denna rapport som ganska självklart. Författarna av de olika litteraturerna har självklart grävt djupt i tekniken med mestadels fördelar i fokus. Detta bidrar till att utvecklarna ofta kommer att stöta på dessa fördelar som har genererats i tabell 1. Det kan även framkomma fördelar eller nackdelar som inte tas upp av teorin eller praktiken vilket kan bero på utvecklarens egna bakgrund och erfarenheter samt intresset för tekniken.

Motsägelser angående teknikens lätthanterlighet har framkommit. I ett fall nämns det att tekniken är lättförståelig och i ett annat fall har det framkommit att den är svår att lära in. Anledningen till att dessa har valts med i tabellerna är på grund av att kunna visa att tekniken är lätt att förstå och samtidig är smidig för kravhantering. Att den upplevs som svår för inlärning kan i de flesta fall bero på att tekniken skiljer sig mycket från andra tekniker vilket kan leda till att utvecklaren upplever tekniken som svår.

Vidare kommer de för och nackdelar som tidigare definierats att diskuteras.

6.4.1 Fördelar

Genom att studera tabell 1 går det att se att de största fördelarna som framkommit under litteraturstudien har i stort sätt också framkommit under intervjuerna som gjorts. Fördelar som enbart yttrar *”tekniken är bra”* eller liknande fördelar har inte tagits hänsyn till. Dock har det även framkommit en del andra fördelar som författaren av denna rapport inte lyckats hitta i litteratur som studerats. Efter att närmare analyserat dessa punkter anser författaren av rapporten att anledningen till detta kan vara teknikens frihet. Det har framgått att tekniken är ganska fri och saknar striktare riktlinjer. Dess frihet kan då utnyttjas från utvecklarna så att den passar deras behov bättre. Detta kan vara anledningen till att det dels finns fördelar med tekniken som inte hittats i litteratur samt att tekniken är omtyckt. Att de fördelar som framkommit både kan konstateras i litteratur och intervjusvaren kan enligt författaren av rapporten ses som att litteraturen har lyckats med att förmedla de fördelar Use Case har i samband med kravhantering.

6.4.2 Nackdelar

Vidare genom att studera tabell 2 påvisas nästan inga likheter med tabell 1. De nackdelar som framkommit under intervjuerna har inte kunnat definieras i litteraturen som studerats till detta examensarbete. Det är enbart en av de nackdelarna som både teori och praktik tar upp nämligen att tekniken tenderar att få för många Use Case delar i en och samma utveckling. Dock betonades inte denna nackdel på exakt samma sätt i litteraturen som under intervjuerna. I litteraturen betonades den som en nackdel, dock inte i någon större sammanhang medan under intervjuerna kunde denna nackdel orsaka problem som kunde bidra till att utvecklingen försämrades. En nackdel som framkom i litteraturen påpekades inte under intervjuerna. En närmare analys av intervjuerna har gjorts för att ta reda på varför denna nackdel inte kom upp under intervjuerna. Anledningen som kan ligga till grund för detta är att rapporten i helhet handlar om för och nackdelar med Use Case i samband med kravhantering. De frågor som ställts under intervjuerna hade inte till syfte att kunna resultera något svara som hade med hur man tog reda på att Use Caset är korrekt utfört eller inte. Detta är dock något som borde ha gjorts. Anledningen till att många av de nackdelar som framkom under intervjuerna inte hittade i litteraturen som studerats kan bero på flera olika aspekter. En av dessa aspekter kan vara att fel sökteknik har använts när artiklar söktes. Under arbetets gång har antal olika försök till att hitta relevanta artiklar, det vill säga artiklar som hanterar Use Case och kravhantering, gjorts. Dock har detta inte varit så lyckat. En annan anledning till att rätt artiklar inte hittats kan bero på att det inte gjort någon större undersökning på denna front. Anledningen till detta kan vara att tekniken i helhet är bra och att det inte behöv någon djupare undersökning till detta. En annan aspekt som kan ha orsakat att resultatet av nackdelarna ser ut som de gör är att de litteraturer som studerats har haft en direkt koppling till UML och Use Case. En författare som skriver en bok om Use Case har med stor sannolikhet positiv syn på tekniken och tar då inte upp nackdelarna i så stor utsträckning. Dessa aspekter har bidragit till att tabell 2 ser ut som den gör.

7 Resultat

I detta kapitel kommer den tidigare skrivna problemställningen att besvaras. Resultatet kommer att delas upp på två olika kategorier, nämligen fördelar och nackdelar med Use Case i samband med kravhantering. Materialet som presenteras härnäst har bearbetat utifrån den analys som gjorts i kap 6 med avseende på de utförda intervjuerna. Kapitlet avslutas med en diskussion kring resultatet.

7.1 Fördelar med Use Case

- Use Case är en bra teknik för systemutvecklingsprojekt. Tekniken bidrar till att utvecklaren ganska enkelt kan avgränsa sig så att enbart relevant information tas med när systemet skall byggas. Detta bidrar i sin tur till att kravhanteringen kan underlättas.
- Tekniken är iterativ och detta betyder att den kan komma att användas i alla livscykelns olika faser utan någon inbördes ordning. Dock används den som flitigast i de tidigare faserna. Eftersom tekniken är iterativt är det inga större problem för utvecklaren att lägga till nya krav eller ändra eventuella missuppfattningar som kan uppkomma under projektets gång.
- Use Case-diagrammet bidrar till att utvecklaren har lättare att förstå och följa upp flödet. Detta leder till att utvecklaren enklare kommer att kunna hantera de krav som finns på systemet. Skulle en utvecklare fortsätta på en redan påbörjad Use Case-diagram kan den utan större problem se vilka krav som finns på systemet.
- Use Case bidrar till att kommunikationen mellan utvecklare och användare underlättas. Eftersom utvecklaren får en enklare kommunikation med användaren kommer denne att ha en bättre förståelse för utvecklingen och kan därför lättare förmedla sina idéer vidare till programmeraren. Detta medför även i sin tur att kommunikationen mellan utvecklare och programmerare förbättras som leder till att kravhanteringen hanteras på ett effektivare sätt.
- Use Case gynnar alla parter i ett systemutvecklingsprojekt som bidrar till bättre kravhantering och en nöjd kund. I första hand blir utvecklingen enklare för utvecklaren. Kommunikationen mellan de olika parterna förenklas. Detta genererar till en effektivare utveckling som i sin tur bidrar till att de krav och eller mål som kunden hade på systemet uppfylls.

7.2 Nackdelar med Use Case

- Tekniken har en tendens att få för många Use Case-delar som kan förvirra ofta oerfarna utvecklare. Detta kan bland annat leda till att utvecklingen blir tidskrävande som leder till att utvecklaren missar viktiga detaljer vilket kan påverka det slutliga resultatet.
- Riktlinjer som ger en vägledning i tekniken saknas. Detta kan generera otydligheter som exempelvis missförstånd och kan leda till sämre resultat.
- Tekniken kräver mycket dokumentering som leder till att projektet både kan bli långvarig och i vissa fall svårhanterlig på grund av att det kan bli för många Use Cases att hantera.

- Tekniken kan uppfattas som om den har svår inlärningsprocess då den skiljer sig mycket från andra mer traditionella tekniker och kan leda till att utvecklarna tappar intresset för den.

7.3 Diskussion kring resultatet

Det slutgiltiga resultatet som har besvarats visade sig bli som jag antog. Innan jag påbörjat mina intervjuer hade jag utifrån den litteratur som jag studerat för att kunna skriva bakgrunden fått en bild av Use Case och kravhantering och resultatet visade sig bli som jag trodde. Alla respondenter hade en positiv syn på Use Case och fördelen med tekniken vid kravhantering. Samma problem som jag hade med att hitta nackdelar i litteraturen hade jag när jag gjorde mina intervjuer. Respondenterna hade svårt att förklara sina nackdelar, dock efter ett tag under intervjuernas gång kunde jag bearbeta fram vissa nackdelar med tekniken. Respondenterna var mer angelägna att prata om fördelar med Use Case-tekniken än nackdelar. Detta kan vara anledningen till att tekniken har en positiv syn ute i arbetslivet.

Under projektets gång har många fördelar och nackdelar bearbetats fram. Dock har en fördel respektive nackdel väckt extra mycket känsla hos författaren av denna rapport.

En fördel med Use Case och kravhantering som ständigt återkommer och anses som viktigt av författaren av denna rapport är att tekniken underlättar förståelsen vilket bidrar till bättre samarbete och utvecklingsmöjligheter. Med dagens konkurrenskraftiga marknad är det viktigt att välja en teknik som genererar så många fördelar som möjligt. Dock anses det enligt författaren av denna rapport att det är extra viktigt att tekniken är lätt att arbetat med och samtidigt bidra till att olika utvecklare utan större problem kan samarbeta med varandra, även vid utveckling av redan befintliga system.

Största nackdel som bearbetats fram genom intervjuerna samt litteraturstudierna som även i vissa fall kan vara avgörande för projektet är enlig författaren av rapporten att tekniken tenderar att få för många Use Case delar som bidrar till att utvecklingen kan bli tidskrävande. Detta är något som både framkommit av respondenterna samt den litteratur som studerats. Anledningen till att detta ses som en viktig nackfördel enlig författaren av rapporten är att det är viktigt att utvecklarna är medvetna om hur omfattande tekniken kan vara och vad konsekvenserna till detta kan bli.

8 Diskussion

Under detta kapitel kommer en diskussion kring de erfarenheter som erhållits under examensarbetets gång att introduceras. Slutligen presenteras även idéer för vidare studier kring Use Case-tekniken.

8.1 Erfarenheter kring arbetet

Problempreciseringen för detta examensarbete har till uppgift att bearbeta fram och ge en bättre förståelse för de för och nackdelar med Use Case i samband med kravhantering. Efter att ha genomfört denna undersökning anses att en bättre förståelse om problemet erhållits. Detta dels genom att ha studerat litteratur och dels genom intervjuer med personer som är insatta i Use Case-tekniken.

Det som upplevdes svårast i denna rapport var att hitta bra litteratur som på något sätt kunde gynna rapportens syfte. Detta har medfört att den personliga tidsramen som varit uppsatt för att avsluta varje del av rapporten ofta varit väldigt nära till att spricka. En annan anledning till detta har varit att det i vissa delar i rapporten använts information som egentligen inte medförde nytta för rapporten. Dessa har dock i efterhand fått justeras. Det har inte heller varit lätt att hitta nackdelar med Use Case. De litteratur som bearbetats nämnde mycket om fördelar med Use Case men dock inte så mycket om nackdelar.

Att kombinera metoden litteratur och intervju ansågs vara relevant för att kunna besvara på rapportens problemställning. Genom att intervjua respondenter som har erfarenhet av tekniken och sedan kunna kombinera detta med befintligt litteratur bidrar till att information som kan gynna rapporten bearbetas fram. Denna metod visade sig vara svårare än vad jag från början trodde. De svåraste var att hitta respondenter som hade tid att ställa upp på en intervju. Sedan var det viktigt att de också hade kunskap och erfarenhet från Use Case-modellering och kravhantering. Dock låg svårigheten inte enbart under de tidigare nämnda problemen. Förutom tidsplanen som fanns för detta examensarbete blev författaren tvungen att anpassa sig efter de respondenter som tackat ja till en intervju.

Att resonera med respondenter upplevdes väldigt positivt då jag både fick samtala med människor som varit verksamma i branschen samtidigt som jag själv fick vara ute i verkligheten och se hur min framtida arbetsplats kan se ut. I helhet upplevdes respondenterna som väldigt insatta i ämnet och samtidigt mycket trevliga och hjälpsamma. Detta medförde till att intervjuerna gick väldigt smidigt till. Att information om intervjun skickades ut via email till respondenterna innan själva intervjudagen gjorde att respondenterna var pålästa på problemet som skulle undersökas. Detta medförde i sin tur att de då var mer aktiva under intervjun. Den efterkontroll som gjordes med intervjuerna, genom att en sammanfattning av intervjun mailades tillbaka till respektive respondent och sedan fått den godkänd, visar att det insamlade materialet har en hög grad av acceptans. Detta innebär att respondenterna inte blivit missförstådda på något sätt då de själv godkänt sammanställningen av intervjuerna som skickades till dem via email.

En fråga på hur utvecklarna kan vara säkra på att den Use Case som utvecklats är korrekt borde ha ställts under intervjuerna. Genom denna fråga skulle författaren av rapporten få ut eventuella för eller nackdelar som kunde ha påverkat resultatets tyngd.

8.2 Reflektioner kring resultatet

Resultatet som bearbetats fram i kap 7 anses ha en viss grad av pålitlighet. Detta på grund av att rapportens metod innehåller både information från personer som har erfarenhet av problemet som undersöktes samtidigt som den innehåller information från litteratur. Ingen av de tillfrågade respondenterna arbetade på samma företag vilket har bidragit till att resultatet är från en bred grupp av systemvetare. Fördelen med att de inte arbetar på samma företag är att de inte delar samma tankar och idéer vilket bidrar till mer acceptabel nivå på resultatet. Anledningen till att resultatet inte kan uppskattas som 100% pålitligt är bland annat på grund av resursbrist. Detta har bidragit till att jag inte haft möjlighet att göra en djupare undersökning. Det hade varit intressant att se hur resultatet hade blivit om jag exempelvis hade 10 företag till som deltog i undersökningen kombinerat med fler litteratur. Dock anses resultatet rikta åt ett positivt håll vilket är en bra början för ett större projekt kring Use Case-tekniken.

Det gjordes fem olika intervjuer av personer som haft erfarenhet av både Use Case och kravhantering. Dessa fem personer visade sig kunna bidra med tillräcklig information för att kunna besvara på rapportens problemställning. En av intervjuerna gjordes via telefon. Detta orsakade dels att jag gick miste om respondentens kroppsspråk samt att samtalet inte gick att spela in på band. Dock anses detta inte vara något problem för det bearbetade resultatet då kompletteringsfrågan använts för att kunna jämföra om respondentens bild av Use Case var det samma som jag fått fram efter att ha sammanställt intervjun. Det har annars inte upplevts som någon skillnad mellan de insamlade materialen.

Anledningen till att mycket information om nackdelar med Use Case i samband med kravhantering inte kom fram i studien är på grund av att jag inte har hittat tillräckligt information i de litteratur jag har studerat under arbetets gång. Försök till att hitta artiklar som undersöker fall där Use Case och kravhantering är inblandad har även gjorts men dock misslyckats. Orsaken till misslyckandet kan bero på att fel sökteknik gjordes under sökningarna. Om mer relevanta artiklar om Use Case och kravhantering hade påträffats i förhållande till rapportens problem kunde detta påverka resultatet, i alla fall på de negativa aspekterna.

Anledningen till att det inte hittades många nackdelar om Use Case och kravhantering i de olika böcker som studerats kan egentligen bero på flera olika orsaker. En orsak som författaren av denna rapport misstänker är att de böcker som är skrivna om Use Case är ofta från författare som personligen är positivt insatta för tekniken och därför framkommer inte de nackdelar som egentligen finns med tekniken.

8.3 Förslag på fortsatta arbeten

Efter att ha forskat en hel del i Use Case och hanteringen av krav har intresset väckts för att undersöka om Use Case upplevs likadant i båda stora och små projekt. Samtidig är det tänkvärt att kunna undersöka om tekniken använts i samma faser i livscykeln i både dessa projekt. Det vore exempelvis intressant att ta reda på om det resultatet på något sätt skiljer sig med det resultat som jag har kommit fram till i kap 7 och vad anledningen till detta kan vara.

En nackdel som framkom med Use Case var att den saknade riktlinjer. Ett förslag på framtida arbeten kan vara att ta fram riktlinjer eller undersöka möjliga riktlinjer som kan användas för tekniken.

9 Referenser

Böcker

Andersen, E. (1994) *Systemutveckling principer, metoder och tekniker* studentlitteratur Lund.

Avison, D. och Fitzgerald G (1998) 2nd Edition *Information systems development: Methodologies, Techniques and tools* The McGraw-Hill Companies

Avison, D. och Shah, H. (1997) *The information systems development life cycle: A first course in information systems* The McGraw-hill companies

Cockburn, A. (2001) *Writing effective use cases, Addison - Wesley*

Cronholm, S(1998) *Metodverktyg och användbarhet, en studie av datorstödd metodbaserad systemutveckling* Linköping

Dennis, A. Wixom, B. (2000) *System analysis and design, an applied approach*. John wiley & sons, Inc.

Eriksson, H och Penker, M. (2000) *Business modeling with UML- Business Patterns at Work.* New Yourk: John Wiley and Sons, Inc.

Flensburg, P. (1987) *Systemutveckling med människan i centrum,* Student litteratur lund.

Jacobson, I. *applying use cases, A practical guide,* second edition, Addision – Wesley.

Karlsson, J (1996) *Framgångsrik kravhantering,* Sveriges verkstads industrier

Kotonya, G. och Sommerville, I. (1998) *Requirements Engineering, Process and Techniques* John Wiley & Sons

Kruchten, P (2000) *Rational Unified Process, en introduktion* Addision-Wesley

Kulak, D. och Guiney, E. (2000) *Use Cases, requirements in context* Addision-Wesley

Leffingwell, D. och Widrig, D. *Managing software requirements, A unified approach* Addison – Wesley

Macaulay, L. (1996) *Requirements engineering,* Springer – Verlag London

Muller, P (1997) *Instant UML* Wrox Press

Patel, R. och Davidson, B. (1994), *Forskningsmetodikens grunder, att planer och genomföra och rapportera en undersökning.* Studentlitteratur, Lund

Regnell B (1999) *Requirements Enginering with Use Cases- a Basis for software development*, Tekniskt rapport nr 132 Lund

Rosenberg, D. och Scott, K. (1999) *Use case driven object modelling with UML, A practical approach:* Addison-Weley

Schneider, G och Winters, J. (2001) *Applying Use Cases, second edition, A practical guide.* Addison Wesley Longman

Sommerville, I och Sawyer, P. (1997) *Requirements Engineering, A good practice guide* John Wiley & Sons

Wärneryd, B. mfl (1990) *Att fråga, om frågekonstruktion vid intervjuundersökningar och postenkäter* Statistiska Centralbyrån.

Artiklar

Annie, I, Anton. John, H, Dempster. Devon, F, Siege. (1999) *Deriving Goals from a Use-Case Based Requirements Specification for an Electronic Commerce System.* North Carolina State University.

Celander, L (1996) *Ta kravet som ett object* Mediearkivet Stockholm.

Fantechi, A. Gnesi, S. Lami, G. Maccari, A. (2003*) Application of linguistic techniques for Cse Case analysis* Springer- Verlag London

Liang, Y. (2002) *From Use Cases to classes : a way of building object model with UML* Department of computer and information systems, University of Paisley.

Bilaga 1 Intervjufrågor

Intervjumetoden kommer som tidigare definierats rapporten att vara en ostrukturerad intervju. Då intervjun är öppen kan nya frågor tillkomma samtidigt som vissa kan uteslutas under intervjuns gång. Fråga ett ses enbart som en introduktionsfråga. De kommande frågorna, det vill säga fråga två till åtta, ses som huvudfrågor och har till syfte att bearbeta fram de för och nackdelarna som finns med Use Case. Det är genom dessa frågor som examensarbetets frågeställning är tänkt att besvaras. Fråga nummer nio är tänkt besvara på de eventuella förbättringsförslag som kan finnas. Fråga 10 är en kompletteringsfråga som enbart ställas ifall respondenten visar osäkerhet i sina svar bland annat genom kroppsspråk. En mer detaljerat förklaring uppdelat per fråga följer nedan.

Introduktionsfråga

1-*Information om respondenten, titel, utbildning, antal år i branschen, antal år med Use Case modellering!*

Denna fråga är tänkt att generera en bild av respondenten. Det anses viktigt att veta vilken utbildning, som respondenten har. Detta för att få möjlighet att jämföra de olika respondenterna syn på Use Case. Samtidigt är det viktigt att veta hur många år respektive respondent arbetat i systemutvecklings branschen varav hur länge med Use Case. En respondent som arbetat med Use Case i 15 år behöver inte ha samma bild av Use Case som en som arbetat i 5 år. Därför anses denna fråga som viktigt.

Huvudfrågor

2- Är det viktigt att satsa på kravhantering?

2.1 Varför?

Eftersom rapportens frågeställning är för och nackdelar med Use Case vid hantering av krav anses det viktigt att veta vad respondenten tycker om kravhantering. Oavsätt vad respondenten svara på nästa fråga så kommer en följdfråga som tvingar respondenten att förklara varför de tycker som de gör.

3- Anser du att Use Case är en bra teknik för systemutveckling?

3.1 varför

Fråga tre skall kunna ta reda på vad respondenten tycker om Use Case för systemutveckling. Författaren vill här ta reda på om Use Case är en bra teknik för systemutveckling eller inte. För att få en djupare svar på frågan kommer även här en följdfråga där respondenten blir tvungen att specificera sig.

4- Under vilka faser i livscykeln kommer Use Case mest till användning?

Under kapitel 2.1.1 definierades livscykel modellen och de faser där Use Case används som flitigast. Syftet med fråga 4 är att ta reda på om Use Case används i de faser som det definierats i litteraturen eller ej.

5- Vad är det du som utvecklare vill få ut utav Use Case modellering?

Fråga 5 har till syfte att redovisa vad som förväntas utav Use Case-modellering. Författaren anser att den är viktigt att ta reda på vad det egentligen är som efterstävas av Use Case. Genom att ta reda på detta blir det lättare att ta reda på om tekniken används vid fel ställen.

6- Hjälper Use Case till att bättre hantera kundens krav på systemet?

6.1 Varför, på vilket sätt?

Fråga sex undersöker vad respondenten tycker om Use Case-modellering i samband med hantering utav krav. Författaren vill här ta reda på om respondenter upplever att Use Case hjälper de med krav hantering. För att kunna forska ännu djupare i frågan så har författaren en följdfråga som kommer att användas i behov. Skulle det visa sig att svaret på frågan inte varit tillräckligt kommer fråga 6.1 att ställas.

7- vilka fördelar ser du med användningen utav Use Case?

8– vilka nackfördelar ser du med användningen utav Use Case?

Fråga 7 och 8 utgår ifrån samma syfte. Dessa skall ta reda på vilka för och nackdelar som respondenterna upplever med användningen av Use Case. Det anses vara värdefullt att veta hur respondenterna upplever detta för att vid senare skede kunna komma fram till ett resultat om användningen av Use Case.

9-Tycker du Use Case bör kompletteras med något för att medföra till bättre resultat än den gör idag?

Fråga nio är egentligen den sista frågan som har med undersökningen att göra. Den visar om respondenten saknar något i metoden. Det anses lättare att komma fram till en slutsats till rapportens problemställning om författaren kan ta reda på vad som saknas i Use Caset. Ett annat syfte med denna fråga är att ta reda på om alla respondenter känner samma sak eller om svaret här har med erfarenheten att göra.

Kompletteringsfråga

10 – På skala 1 till 10 vad ger du för betyg till Use Case modellering där 1 står för dåligt och 10 för bra?

Denna fråga är enbart en kompletteringsfråga som kommer att ställas ifall respondenten verkar osäker på något eller några av sina svar. Med osäkerhet menas då respondenten tvekar eller ändrar sig mycket i svaret. Anledningen till att denna fråga ställs är för att se om författaren av denna rapport får den bild av Use Case som respondent givit under intervjuns gång.

Bilaga 2 introduktionsemail

Vi pratades vid idag den X april 2004 angående en intervju för ett examensarbete som jag skriver på Högskolan i Skövde.

Jag skickar härmed lite information om vad själva intervjun går ut på så att du blir insatt i det.

Examensarbetet som jag håller på att skriva handlar om Kravhantering. Jag har valt att undersöka vilka för och nackdelar Use Case bidrar med i samband med kravhantering .

Jag skulle härmed vilja ha en intervju med er någon gång mellan X till X april.

Du som respondent har självklart rätt att vara anonym och när som helst avböja intervjun om det inte skulle passa. All data kommer garanterat att behandlas konfidentiellt. Vidare kan även ett exemplar av den slutgiltiga rapporten skickas till dig om så önskas.

Intervjun tar ca 20 min.

Vore tacksam för ett snabbt besked angående när vi kan ha intervjun.
Tack på förhand

Med vänlig hälsning

Amir Esfahani

xxxx-xx-xx Skövde