

Forelesning IMT2243

18. Februar 2009

Tema :

- ◆ Kravspesifisering : litt mer om prosessen
- ◆ Viewpoint – en "myk" tilnærming
- ◆ Use Case – en scenaribasert teknikk innen metoden Objektorientert Analyse – brukes til å avklare og dokumentere funksjonelle krav

Pensum : Kap. 7 i Sommerville,
Nr. 8 i artsaml (Kap 4 fra Stumpf ..)

Kravspesifisering

Kravspesifisering = arbeidet med å få frem en beskrivelse av oppdragsgiverens / kundens samlede krav til den nye programvaren på en strukturert, oversiktlig og forståelig måte. Beskrivelsen skal være på en form som blir forstått av brukere og beslutningstagere. På samme tid skal kravspesifikasjonen danne grunnlaget for alt videre arbeid med å utvikle programvaren.

Målformuleringen i emnebeskrivelsen viser at dette temaet er svært sentralt :

"Studentene skal ha forståelse for grunnleggende administrative og teknologiske aspekter ved spesifisering, utvikling, innføring og vedlikehold av datasystemer. De skal være i stand til å reflektere over IT-systemenes betydning for verdiskapningen i virksomheter og ulike tilnæringsmåter i systemutviklingsprosesser. **De skal kunne anvende metoder og teknikker for kravspesifisering og analyse.**"

Kravspesifikasjonen

Kravspesifikasjonsdokumentet skal dekke :

- ◆ Funksjonelle krav
- ◆ Ikke-funksjonelle krav (operasjonelle)
- ◆ Grensesnitt mot brukere, andre systemer og enheter
- ◆ Avgrensninger

Dokumentet er ofte sentralt som :

- ◆ Beslutningsgrunnlag for evt. videreføring av prosjektet.
- ◆ Vedlegg til en kontrakt mellom kunde og leverandør
- ◆ Grunnlag for detaljert tids-, kostnads- og ressursplan
- ◆ Utgangspunkt for alt designarbeid.
- ◆ Grunnlag for alt videre utviklingsarbeid
- ◆ Grunnlag for all testing

Strengt krav til kravene !!

- Validitet. Tilbyr systemet den funksjonalitet som best støtter kunden sine behov (Riktig system) ?
- Konsistente krav. Er det konflikter mellom ulike krav som er beskrevet ?
- Komplette. Er alle krav fra kunden inkludert ?
- Realisme. Kan kravene innfris innen de gitte budsjett- og teknologi-rammer ?
- Verifiserbarhet. Kan kravene etterprøves /sjekkes ?

Involverte i prosessen

Det er mange involverte parter i selve analysearbeidet

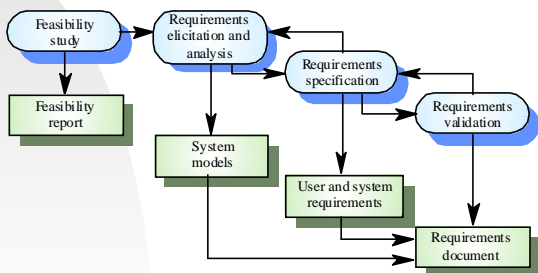
- sluttbrukere, utviklere, ledere, leverandører, tekniske eksperter, fagforeningsrepresentanter +++
- en analyseekspert bør "dra i trådene"

De involverte har et sterkt varierende forhold til og kunnskap om henholdsvis domenet/forretningsvirksomheten og om datateknologi

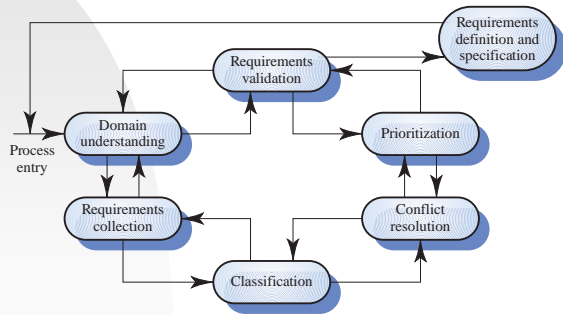
En av utfordringene ligger i å finne balansegangen mellom kreativitet og struktur.

Kravspesifiserings-prosessen

(Sommerville fig. 7.1)



Detaljert kravanalyse (Sommerville, 7.ed)



Viewpoints – en “myk” spesifiseringsmetode

- En arbeidsmetode der man forsøker å finne flest mulige aktuelle perspektiver på den nye løsningen
- For hvert viewpoint (perspektiv) forsøker man å identifisere dette viewpointet's krav til programvaren
- God måte å få i gang arbeidet på, da fokusen er så intuitiv at alle kan delta konstruktivt uten "forkunnskaper"

Fremgangsmåte for Viewpoints

1. Identifisere ulike viewpoints
2. Strukturere viewpoints
 - Gruppere i viewpoint
 - Avdekke overlapp og konflikter i viewpoints
 - Lage hierarki
3. Dokumentere med viewpointbeskrivelser

Metoden egner seg som en oppstart før man går dypere inn arbeidet med å avdekke kravene med metoden Objektorientert Analyse

Steg 1 : Identifisering

Brainstorming for å komme opp med ulike forhold rundt løsningen

Se etter interessenter, brukergrupper, tjenester, komponenter, personer, organisasjonsenheter, hendelser, tilstander osv. rundt systemet

Forsøk deretter å finne ut hvilke av disse som er viewpoints. Grupper / kategoriser også de andre "idéene" som er dukket opp (ønskede tjenester, operative krav, komponenter, tilstander osv.)

Viewpoints identification

(Sommerville, 7.ed)



Steg 2 : Strukturering

Gruppér de ulike viewpoints, se etter overlappinger

Knytt de aktuelle tjenester til det enkelte viewpoint

Gi en kortfattet beskrivelse av viewpointet

Sett opp et viewpoint-hierarki

Viewpoint service information

ACCOUNT HOLDER	FOREIGN CUSTOMER	BANK TELLER
Service list	Service list	Service list
Withdraw cash Query balance Order cheques Send message Transaction list Order statement Transfer funds	Withdraw cash Query balance	Run diagnostics Add cash Add paper Send message

Scenarios

(Sommerville s. 153)

- Scenarios are real-life examples of how a system can be used.
- They should include
 - ◆ A description of the starting situation;
 - ◆ A description of the normal flow of events;
 - ◆ A description of what can go wrong;
 - ◆ Information about other concurrent activities;
 - ◆ A description of the state when the scenario finishes.

USE CASE

Use Case er en utbredt scenaribasert teknikk innen systemutvikling som har sin styrke i at den egner seg godt i spesifisering av funksjonelle krav.

Et use case (brukstilfelle) er en tjeneste i form av en handlingssekvens som systemet skal utføre for omgivelsene. Denne initieres normalt av en hendelse ute i systemets omgivelser, og skal alltid gi et observerbart resultat for en bestemt aktør. Use Case benytt

Karakteristika :

- Et Use Case viser en tjeneste systemet skal yte
- En av aktøren initierer vanligvis et Use Case (pil – notasjon)
- Et Use Case gir alltid en verdi til brukeren
- Et Use Case skal være komplett

USE CASE (2)

En teknikk innen Objektorientert Analyse som brukes for å illustrere det nye systemets

- forventede "tjenestetilbud"
- omgivelser (i forma av aktører)
- samspillet mellom disse

Det er vanlig å samle alle Use Cases i et "Use Case diagram" for hele systemet. Det er derimot Use Case beskrivelsene som i særklasse er viktigst ved denne modelleringen.

Use Case beskrivelsene utgjør de funksjonelle kravene til programvaren og danner grunnlaget for planlegging av det videre arbeidet i systemutviklingen (er et kjerneartefakt i RUP).

Use Case modellering

Arbeidsgangen når man setter opp Use Case :

A. Skisser et Use-Case diagram som viser helheten

- systemavgrensningen
- aktørene
- use casene
- relasjonene.

B. Lage en "High-level" Use Case Beskrivelse for hvert Use Case i diagrammet

C. Lag detaljerte beskrivelser av de Use Case som vurderes til å ha størst risiko (samlet teknologisk, forretningsmessig og prosjektmessig).

"High-level" Use-Case Beskrivelse

Dette er en overordnet tekstlig beskrivelse av det enkelte Use Case. Benytt en mal for beskrivelsen for å få en god struktur.

Use Case (Navn) :

Aktør :

Mål :

Beskrivelse :

"Expanded" Use-Case Beskrivelse

Use Case (Navn) :
Aktør :
Mål :
Beskrivelse :
Type :
Pre betingelser :
Post betingelser :
Spesielle krav (eks. operasjonelle krav) :
Detaljert Hendelsesforløp ("Main Success Scenario") :

Alternative scenarier :
 Ulike Varianter
 Ulike Feilsituasjoner

LIBSYS use cases Fig. 7.7. Sommerville

