

Notat om bruk av Use cases (brukstilfeller) for å beskrive et systems funksjonalitet

Dette notatet er en kort innføring i helt grunnleggende bruk av Use cases og Use case diagrammer for utarbeidelse av funksjonalitet i utviklingsprosjekter. Det låner fra tidligere notater, og henviser også til annen litteratur for grundigere gjennomgang. Spesielt takk til Birgit Krogstie for hennes notater om use cases.

Eivind Brevik (eivind.brevik@nith.no)/Oslo 20.1.2007

Beskrivelse av funksjonalitet

Kundens krav til et IT-system kartlegges ved at use case-beskrivelser og use case-diagrammer. Disse utarbeides i planleggingsfasen i MSF (Microsoft Solution Framework), i tillegg til visjon og tilleggsspesifikasjon med ikke-funksjonelle krav til systemet. En egen ordliste kan også utarbeides.

Hvilken funksjonalitet skal et system tilby? Hva skal en bruker kunne få utført ved hjelp av systemet? Dette kan beskrives mer eller mindre formelt/strukturert.

Definisjon:

Et Use case (brukstilfelle) beskriver en komplett funksjon (sett fra brukerens synspunkt) som produserer et nyttig resultat for den som bruker (initierer) funksjonen.

Sagt på en annen måte: Et use case beskriver en interaksjon mellom en aktør og systemet; en funksjon som har verdi for aktøren. Beskrivelsen bruker kundens vokabular.

Use cases starter med en *systemhendelse* der en aktør setter i gang en samhandling med systemet. Systemhendelsen kan beskrives ved hjelp av to nivåer.

Det ene er et use case-diagram som gir en visuell oversikt over alle use casene som gjelder et system, og over sammenhengen mellom disse. Diagrammet brukes hovedsakelig for å kommunisere løsningen mellom utviklings- og brukermiljøet. Videre er et use case hovedsakelig en tekstlig beskrivelse, og skal angi **hva** systemet/løsningen skal gjøre og **ikke hvordan**. De kan være mer eller mindre formelle og omfattende. Vi bruker et bestemt format som beskrives senere i notatet (se figur 1).

Noen viktige begreper

Hva er en aktør?

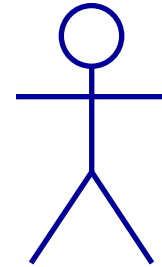
En aktør er en del av systemets omgivelser (for eksempel en bruker).

Den vanligste aktøren er en bruker, det vil si et menneske. En aktør kan også være et annet (eksternt) system, for eksempel: Visa Norge (for sjekking av kredittkort), et regnskapssystem (hvis "vårt" system genererer eller trenger regnskapsdata). Til og med **tiden** kan være en aktør, hvis f.eks. bestemte funksjoner i systemet skal initieres på visse tidspunkt, uten manuell inngripen, som produksjon av faste rapporter, backup-kjøringer og databasevedlikehold/reorganisering.

En aktør representerer en **rolle**, ikke en spesiell **person** eller **system**. For mennesker (brukere) gjelder at flere fysiske personer kan dele en rolle (eks.: "selger") og en person kan ha flere roller (eks.: "selger" og "kundemottaker"). Det er med andre ord **rollen** som har behov/krav overfor systemet!

En aktør **initierer** (vanligvis) et use case (brukstilfelle.)

Grafisk tegnes en aktør som en **fyrstikkmann**.



Hvordan identifisere aktørene som samhandler med et system?

Identifiser yttergrensene for systemet. Tenk på systemet som en slags "boks". Vi skal beskrive hvordan boksen skal brukes, sett fra utsiden. Vi skal ikke beskrive hvordan boksen "virker" inni.

Spørsmål man kan bruke for å identifisere aktørene:

- Hvem **bruker** systemet?
- Hvem **får** informasjon fra systemet?
- Hvem **gir** informasjon til systemet?
- Hvem **drifter/vedlikeholder** systemet?
- Hvem **starter opp / tar ned** systemet?
- Skjer det noe automatisk til visse tider?
- Hvem har nytte/glede av det systemet skal tilby av tjenester?

Hvordan gå frem for å identifisere use cases

Utgangspunktet er visjonsdokumentet fra visjoneringsfasen i MSF. Dette innebærer at man har en rimelig klar ide om hva systemet skal gjøre, hensikten med å lage det, og hva kunden ønsker å oppnå. Deretter sørger man for å identifisere (de fleste?) aktørene og gjør man som følger:

Tenk på hver aktør/rolle:

- Hvilke funksjoner har denne aktøren behov for?
- Hva har systemet behov for fra denne aktøren?

Poenget er å tenke *funksjoner* og *tjenester*, sett fra brukernes synspunkt (problemrommet). Bruk *kundens vokabular*.

Finn et meningsfylt navn for hvert use case, og lag en kort (2-3 linjer) beskrivelse av hva brukstilfellet (funksjonen) skal gjøre. Dette gir de overordnede use cases. Samlet skal de definere den komplette funksjonaliteten i systemet.

Bruk gjerne følgende tabell (eksempel) for å få frem mulige use cases:

Mulige aktører	Mulige brukstilfeller (funksjoner)	Beskrivelser
Administrator	Registrere utstyrsdata	Adm. Legger inn data om type utstyr, bruksområde.
Kunde	Bestille utstyr	Kunden ringer inn bestilling og gir opplysninger om hva som skal lånes.
Kunde	Hente utstyr	Kunden henter utstyret
Kunde	Betale utstyr	Kunden betaler til en bankforbindelse som sender melding til bedriften om innbetalte beløp.
Tid	Purre utstyrlån	
Administrator	Registrerer kunde(data)	

Figure 1: tabell for å kartlegge use cases

Use case-diagrammer

Use case-diagrammer er et *supplement* til beskrivelsene og gir en visuell oversikt over use casene som gjelder et system, og over sammenhengen mellom disse. Under er et use case vist.

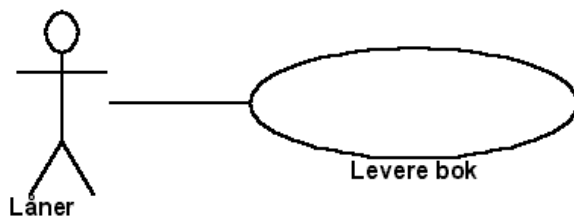


Figure 2: Enkelt use case-diagram

Under (se neste side) er det to eksempler på utarbeidelsen av et use case-diagram.

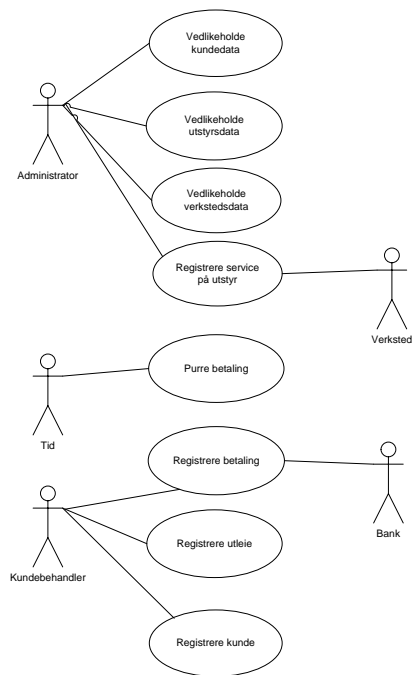


Figure 3: Use case-diagram for et system med utlån av redskap

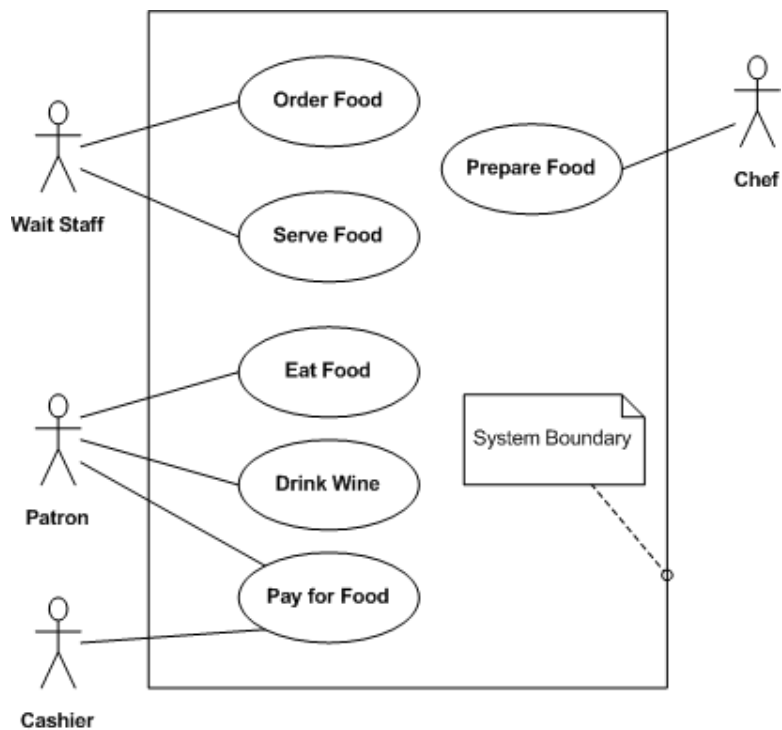


Figure 4Use case-diagram for et system i en restaurant

Eksempel på utarbeidelse av use case

Som et eksempel kan vi tenke oss et use case 'Låne bok' som beskriver en del av de funksjonelle kravene til et bibliotekssystem¹:

Navn	Låne bok
Beskrivelse	Lån av en bok foretas av låner
Aktør	Låner
Forbetingelse	En låner har et fysisk bokeksemlar med en bok-ID
Normalutføring	<ol style="list-style-type: none">1. Låner identifiseres i systemet ved hjelp av sin låner-ID2. Eksemlar registreres som utlånt ved at bok-ID registreres i systemet.3. Låner mottar en kvittering
Alternativ utførelse	1a) Låner-ID finnes ikke i systemet. Går til use case Registrer låner. Returner til 2.
Etterbetingelse	Boken er utlånt

Figure 5: Beskrivelse for use case Låne bok

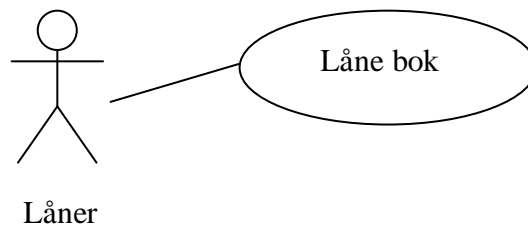


Figure 6: Use case-diagram for Låne bok

Et annet eksempel på use case kan være registrering av kunde i et kundesystem.

Brukstilfelle	Registrere Kunde
Aktør	Administrator
Beskrivelse	Kunden skal registreres med opplysninger, og det skal tildeles et kundennummer av systemet.
Forbetingelse	Kunden er ikke registrert fra før
Normal utføring	<ol style="list-style-type: none">1. Registrere navn, adresse og tlf2. Registrere kundetype3. Opprette kundennummer
Alternativ utførelse	Ingen
Etterbetingelse	Kunden er registrert i systemet og tildelt et Kundennummer

Figure 7: Use case for Registrere kunde

¹ Eksempel lånt og lett tilpasset fra Lasse Bernzens bok *Moderne systemutvikling med MSF*, NITHs skriftserie 2002

Vi utarbeider flere use case slik at all funksjonalitet er dekket. Husk at use cases hovedsakelig er tekstlig, men at use case-diagrammene benyttes for å visualisere systemet spesielt med tanke på kommunikasjon overfor brukere og kunder.

Bruk av Visio til å lage use case diagrammer

Til å lage use case-diagrammer kan man benytte seg av Visio tegneprogram. Velg **Fil** > **Ny** > **Programvare** > **UML** og deretter **UML Brukstilfeller** (norsk utgave). Under er skjermbilde av engelsk utgave, men ellers er det likt.

Man kan avmerke systemgrensene (drag-and-drop), samt opprette aktører og use cases (gi dem navn ved å dobbeltklikke på objektene etter å ha dratt dem ut på arket). Videre knytter dere assosiasjoner mellom aktør og use case ved å benytte “kommuniserer-med” koblingen.

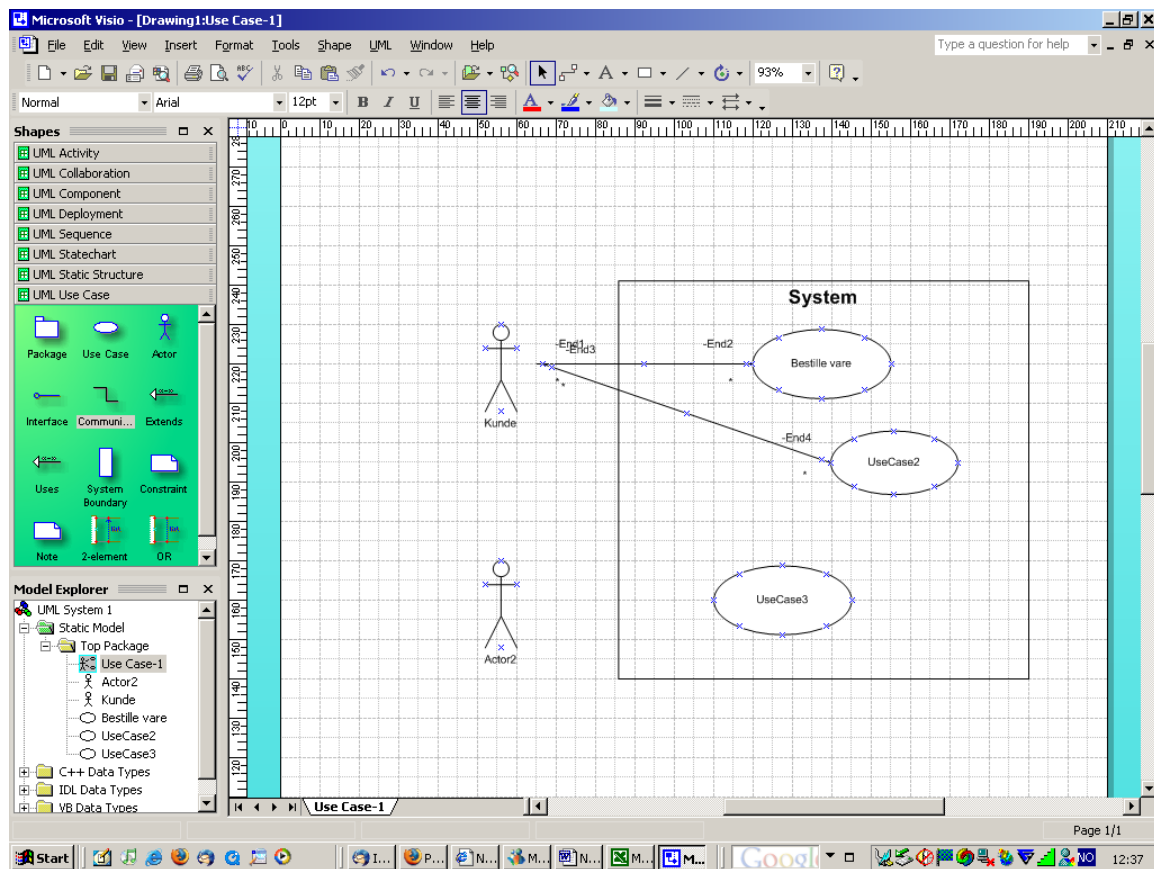


Figure 8: Skjermbilde fra Visio for tegning av use case-diagram

Referanser og anbefalt litterature

Berntzen, Lasse (2002) *Moderne systemutvikling med MSF*, 2. utgave, NKI-forlaget, ISBN 82-562-5755-5

Brevik, Eivind (2002) *Forelesningsnotater SY102 Systemutvikling 1* tilgjengelig via intranettet, NITH

Fowler, Martin (2003) *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, Third Edition, ISBN 0-321-19368-7

Krogstie, Birgit (2002/2003) *Forelesningsnotater Systemutvikling 1 og Notat om RUP*, tilgjengelig via intranettet NITH

Krutchén, Phillipe (2003) *The Rational Unified Process: An Introduction*, Third Edition, ISBN 0-201-70710-1

Developer.com om UML, <http://www.developer.com/design/article.php/2109801>

Wikipedia om use case, http://en.wikipedia.org/wiki/Use_case