

**NTNU**  
**Norges teknisk-naturvitenskapelige**  
**universitet**

**Fakultet for fysikk,**  
**informatikk og matematikk**

**BOKMÅL / NYNORSK / ENGLISH**

**Institutt for datateknikk**  
**og informasjonsvitenskap**



Sensurfrist: 17. Juni 2009

**Eksamen i fag**  
**TDT4140 Systemutvikling**

**27. mai, 2009**  
**kl 0900 - 1300**

**Hjelpemidler A1:**

Kalkulator tillatt

Alle trykte og håndskrevne hjelpemidler tillatt:

**Faglig kontakt under eksamen:**

Universitetslektor Sven Ziemer, tlf. 93488

Poengene viser hvor mange poeng det er mulig å få på hver oppgave. Innen en oppgave teller deloppgaver likt, med mindre annet er angitt.

**Lykke til!**

## Bokmål

### Innledning

Overalt i oppgaven der vi bruker ordet ”system” eller ”systemet” mener vi det systemet som er beskrevet i vedlegg A.

Dersom du trenger informasjon som ikke står i oppgaveteksten må du:

- Forklare kort hvorfor du trenger denne informasjonen
- Gjøre de nødvendige antagelsene. Disse antagelsene må beskrives i besvarelsen.

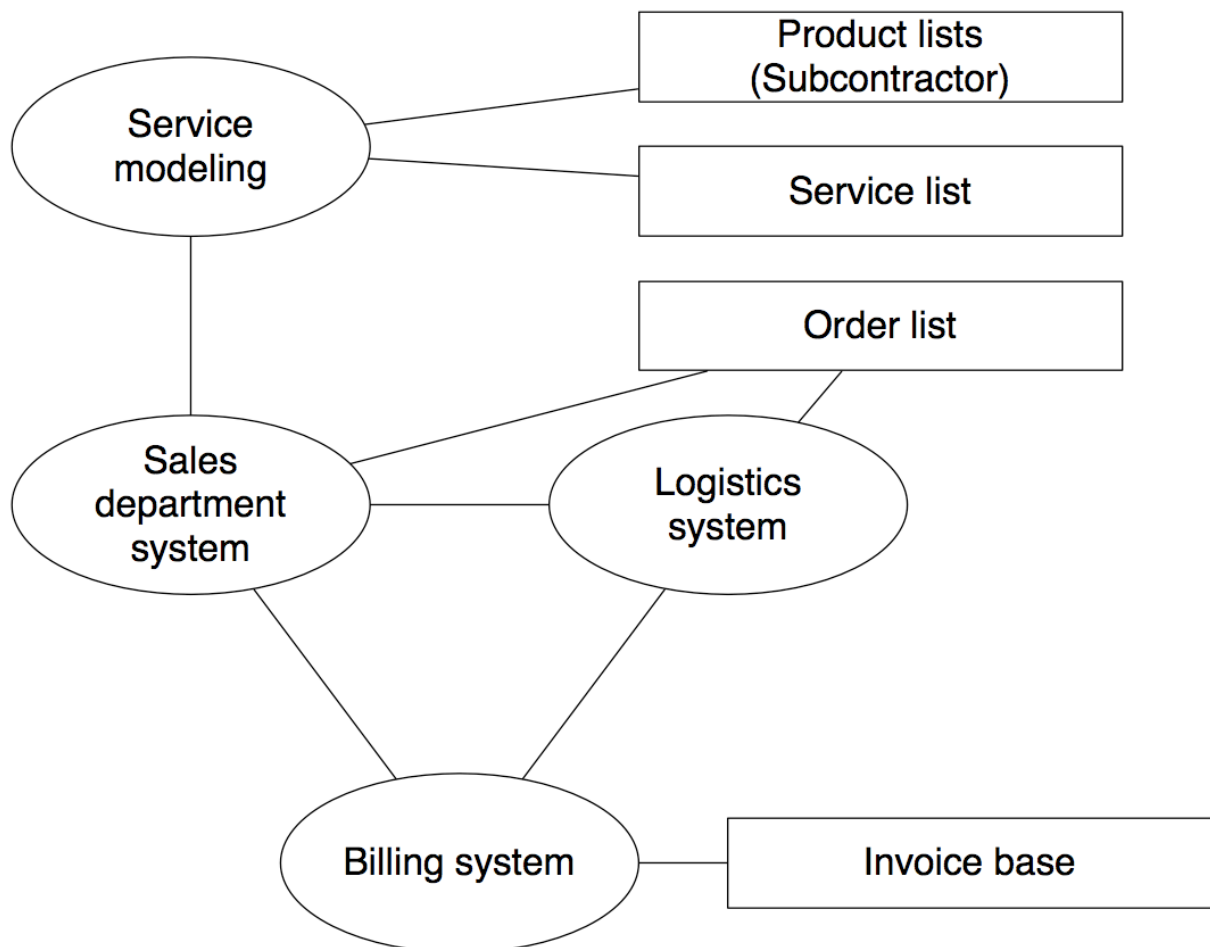
### Oppgave 1 – Dataflyt diagram, 10 poeng

Ta utgangspunkt i beskrivelsen i vedlegg A.

1. Identifiser alle delsystemer som er beskrevet i vedlegg A.

*Tjeneste modellering, Salg system, Ordre system (Logistikk), Faktura system. Det viktigste er at disse fire system blir identifisert. De skal brukes som avgrensning i use case diagrammet i neste oppgave.*

2. Lag et dataflyt diagram som viser informasjonsflyten mellom systemene. La hvert delsystem være en prosess i diagrammet.

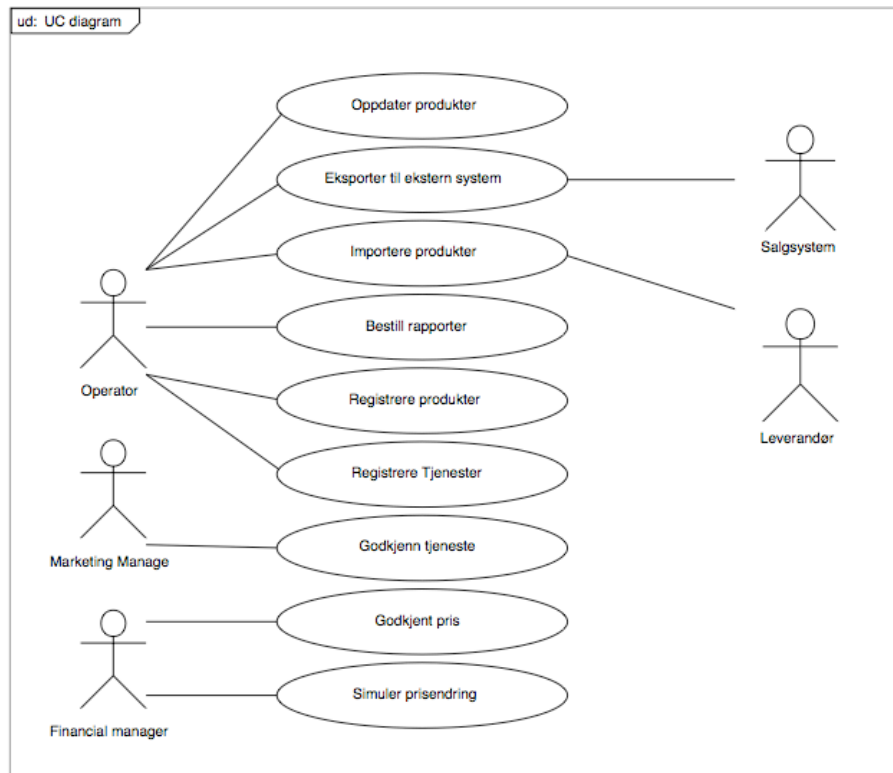


## Oppgave 2 – Use case, 25 poeng

1. Identifiser alle aktører.

*Aktørene er Operatør, Markeds ansvarlig, økonomi ansvarlig, Underleverandør, salgssystem. Underleverandør er mest sannsynlig også et eksternt system.*

2. Lag use case diagram for alle funksjonene til systemet



*Hva de enkelte UC heter er ikke så viktig, de kan ha ulikenavn. For å få full poeng sum må besvarelsene ha fått med seg de ulike aktører, spesielt salgssystem og leverandør.*

3. Lag tekstlige use case for følgende scenarier:

- Registrer produkt
- Registrer tjeneste (service)
- Oppdater pris på tjeneste

•

Navn	Registrer produkt
Prebetingelser	Pålogget systemet
Trinn	
1	Aktiver funksjonen Registrer produkt
2	Oppgi navn, beskrivelse for produktet
3	Velg produktets leverandør
4	Velg produktets kategori
5	Legg til priser for både innkjøp og salg, med de typer som er relevant. Sett startdato.
6	Arkiver produkt
3.1	Ønsket leverandør finnes ikke. Start funksjon for å registrere ny leverandør.
4.1	Ønsket kategori finnes ikke. Start funksjon for å registrer ny kategori.

Navn	Registrer tjeneste
Prebetingelser	Pålogget systemet
Trinn	
1	Velg funksjon Registrer tjeneste.
2	Oppgi navn og beskrivelse
3	Velg produkter som skal inngå i tjenesten
4	Velg tjenester som skal inngå i tjenesten
5	Velg kategori for tjenesten
6	Legg til priser for både innkjøp og salg, med de typer som er relevant. Sett startdato.
7	Arkiver tjeneste
3.1	Ønsket produkt finnes ikke. Start funksjon Registrer produkt.
4.1	Ønsket tjeneste finnes ikke. Avbryt funksjonen for å registrer ønsket tjeneste først.
5.1	Ønsket kategori finnes ikke. Start funksjon for å registrere ny kategori.

Navn	Oppdater pris på tjeneste
Prebetingelser	Pålogget systemet
Trinn	
1	Velg tjenesten som skal oppdateres
2	For hver pris (salgspris) som skal endres, settes en stop dato på den allerede eksisterende pris
3	Legg inn ny pris med ønsket beløp, start dato, og type.
4	Lagre tjenesten
1.1	Tjenesten er ikke prissatt fra før av, hopp over steg 2.
4.1	Det er overlapp mellom en pris' stopp dato og en annen pris' startdato. Gi Feilmelding. Gå tilbake til steg 2, for å rette opp datoer.

Oppgaveteksten er ikke entydig. Besvarelsen må derfor ta opp antakelser som er gjort, slik det er konsistens mellom use case diagrammer, tekstlige use case og sekvensdiagrammer.

### Oppgave 3 – Planlegging, 30 poeng

Du har fått i oppgave å planlegge et prosjekt som skal implementere dette systemet. De som skal bruke systemet trenger dette i løpet av kort tid. Din oppgave blir å gi en anbefaling om når systemet kan settes i drift og hvordan prosjektet skal bemannes.

1. Lag en WBS – Work Breakdown Structure – og et kostnadsestimat for systemet baser på WBS. Dokumenter de forutsetningene du gjør.

Nr	Aktivitet	Min	Normal	Max
1	Registrer produkt Gui		15	
2	Registrer produkt Mellomlag		15	
3	Importer produkt Gui		10	
4	Importer produkt Mellomlag		20	
5	Registrer Tjeneste Gui		20	
6	Registrer Tjeneste Mellomlag		20	
7	Endre Tjeneste Gui		15	
8	Endre Tjeneste Mellomlag		10	
9	Oppdater produkt Gui		10	
10	Oppdater produkt Mellomlag		10	
11	Godkjenn tjeneste Gui		10	
12	Godkjenn tjeneste Mellomlag		15	
13	Godkjenn Pris på tjeneste Gui		10	
14	Godkjenn Pris på tjeneste Mellomlag		10	
15	Bestill rapport Gui		10	
16	Bestill rapport Mellomlag		30	
17	Eksporter til eksternt system Gui		10	
18	Eksporter til eksternt system Mellomlag		20	
19	Databasemodell		60	
20	Test Registrer produkt		10	
21	Test Importer produkt		15	
22	Test Registrer Tjeneste		20	
23	Test Endre Tjeneste		10	
24	Test Oppdater produkt		10	
25	Test Godkjenn tjeneste Gui		10	
26	Test Godkjenn Pris på tjeneste Gui		10	
27	Test Bestill rapport Gui		10	
28	Test Eksporter til eksternt system Gui		10	
29	Prosjektledelse		85	
	Total		510	

Har bare angitt normal verdi, men både min og max kan bli estimert.

2. Lag et Gantt-diagram for utviklingen av dette systemet, og finn ut hvor mange personer som trenges for å kunne levere dette prosjektet i løpet av to måneder.

Aktivitet	Uke						
	1	2	3	4	5	6	7
1	D1 (15t)						
2	D1 (15t)						
3		D1 (10t)					
4		D1 (20t)					
5	D2 (20t)						
6	D2 (10t)	D2 (10t)					
7		D2 (15t)					

8		D2 (5t)	D2 (5t)				
9			D1 (10t)				
10			D1 (10 t)				
11			D2 (10t)				
12			D2 (15t)				
13			D1 (10t)				
14				D1 (10t)			
15				D1 (10t)			
16				D2 (30t)			
17				D1 (10t)			
18					D1 (20t)		
19	DB (15t)	DB (15t)	DB (10t)	DB (10t)		DB (4t)	DB (4t)
20						D1 (10t)	
21						D1 (15 t)	
22						D1 (5t)	D1 (15t)
23						D2 (10t)	
24						D2 (10t)	
25						D2 (10t)	
26							D2 (10t)
27							D2 (10t)
28							D2 (10t)
29	PM (12t)	PM (12t)	PM (12t)	PM (12t)	PM (12t)	PM (12t)	PM (12t)

Ressurser:

D1 – Utvikler 1

D2 – Utvikler 2

DB – Database ansvarlig

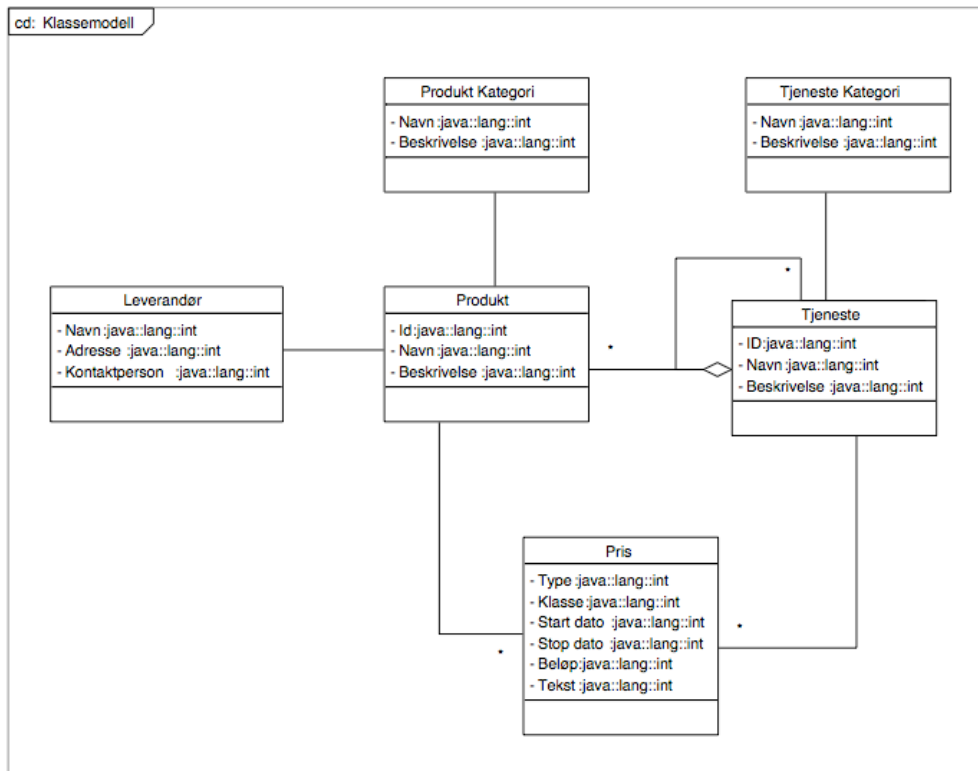
PM – Prosjektleder

*Prosjektet kan utvikles i løpet av to måneder med 3 eller 4 personer. Svaret er 3 personer når prosjektleder og database ansvarlig er samme person. Dersom dette er ulike personer trenger prosjektet 2 utvikler og en prosjektleder og en database ansvarlig på deltid.*

*Antakelser: 30 t per uke per person, 6 uker utvikling og 2 uker testing,*

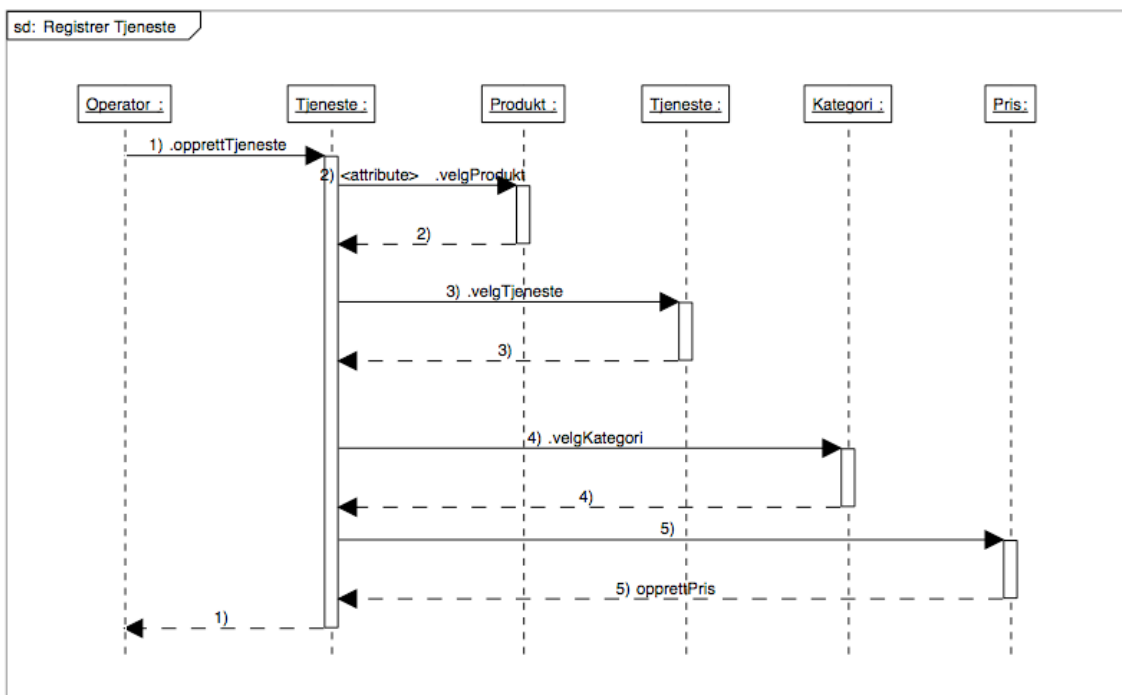
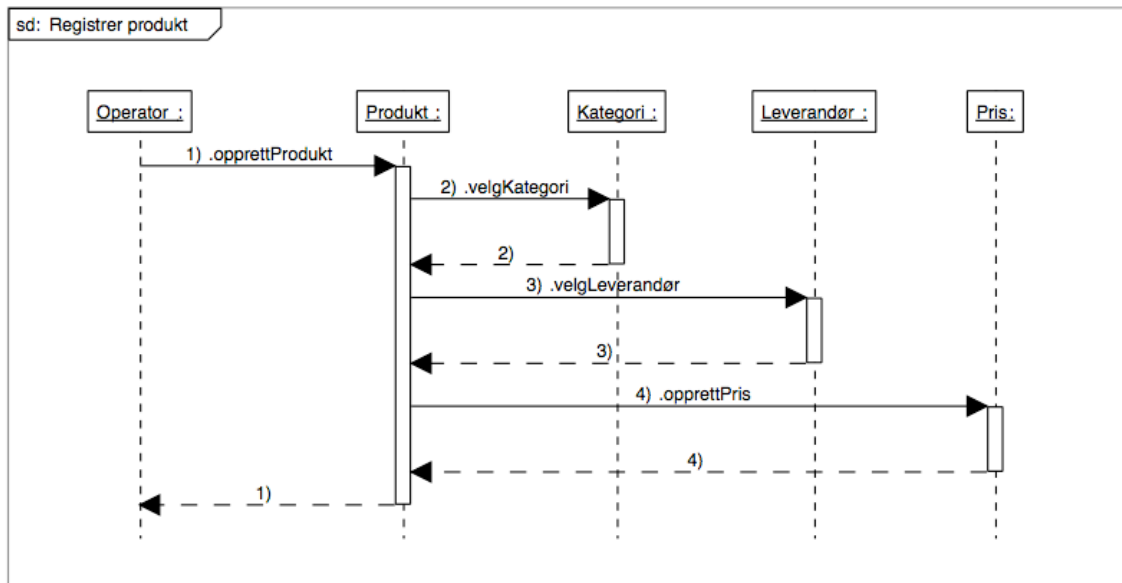
## Oppgave 4 – Klassediagram, 25 poeng

1. Lag samtlige klassediagram for systemet på et nivå som passer til bruk tidlig i utviklingen av systemet – overordnet design.

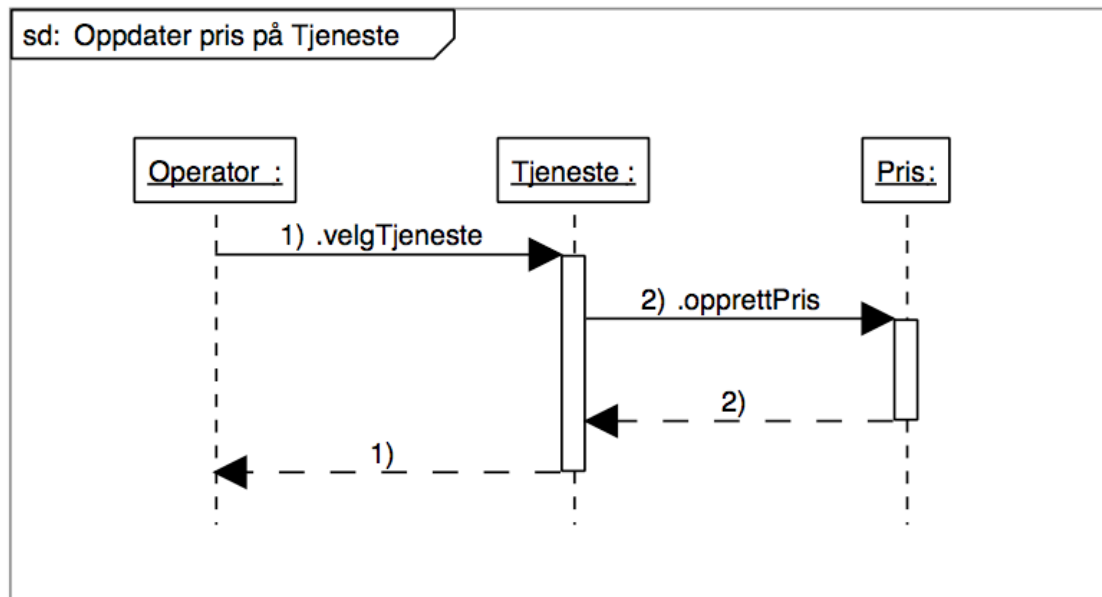


*De viktigste klasser er produkt, tjeneste, kategori, pris, leverandør (sub-contractor) og pris. De kan modelleres på ulikt vis, men en tjeneste må kunne settes sammen av både produkter og tjenester, priser må kunne være både kjøpspris og salgspris, samt ha en av de angitte typer. Det er ikke trekk om multiplisiteten for en assosiasjon ikke er angitt på alle assosiasjoner.*

2. Lag sekvensdiagrammer for de scenariene som er beskrevet i oppgave 1, deloppgave 2.







*Avhengig av antakelser gjort tidligere kan sekvens diagrammene variere; dette gjelder blant annet om det er gjort antakelser om arkitektur som er tatt med i sekvens diagrammer. Sekvens diagrammene her er minimum for å få poeng.*

3. I fremtiden vil det kunne være aktuelt å ta i bruk flere pristyper enn de tre som er beskrevet i Vedlegg A. Lag et design som gjør det mulig å legge til nye pristyper dynamisk (bruk pattern).

*Bruk av Factory pattern, og modeller pristype som egen klasse.*

## Oppgave 5 – Testing, 10 poeng

1. Lag en plan for testing av brukervennlighet. Planen skal inneholde eventuelle forbedringer av brukervennligheten viss testene viser at brukervennligheten ikke er god nok.

*Hvordan denne testen gjennomføres vil avhenge av hvordan man definerer brukervennlighet for dette systemet. Jeg har valgt å bruke definisjonen under som bakgrunn for testen. Andre definisjoner vil lede til andre tester. Det er imidlertid viktig at studenten forteller hva han legger i begrepet bruker vennlighet.*

*Definisjon: Et system er brukervennlig viss brukerne kan utføre sine daglige gjøremål etter et en dags kurs. Foreslår derfor følgende testopplegg:*

- Hold et en dags introduksjonskurs for 20 tilfeldig valgte brukere.
- La hver bruker bruke systemet i to timer der de utfører de viktigste av sine daglige oppgaver.

- *Etter de to timene fyller hver bruker ut et spørreskjema der de viser hvor fornøyde de er med systemet. Ett av spørsmålene må være "På en skala fra 1 til 10, hvor fornøyd er du med brukervennligheten til systemet?" eller liknende.*
- *Minst 80 % må vurdere brukervennligheten til å være 7 eller høyere for at vi skal godta systemet som brukervennlig.*

## Vedlegg / Appendix A – ICT Service Register

A Norwegian company is offering ICT services, such as data communication, server operation and maintenance, application support and total management of IT systems. To be successful in this market, this company is planning to improve the IT systems they are using to design, offer, order, maintain and bill their service. Some of these operations are done without the support of any IT system today, resulting sometimes in late responses or inaccurate data. Therefore, the company plans to build up a family of small IT systems that will replace the ordering and the billing system, which are the two only systems that are in use today.

The first system to build is a system to design, define and configure the services that are to be offered, the companies ICT Service Register. This system is a tool to build services out of products and other services, to set prices and to register other information relevant to that service. The service register is the first system in the value chain of the company, and provides data to all other systems directly or indirectly. The sales department need the information from the service register to offer services that are supported by the company, and to have enough information on how to price the offered services to the potential customers. After orders have been made, the order system receives data from both systems. From the sales system, the services that are contained in an order are provided. Information on where the services can be order, how long the delivery will take, etc. will be provided from the service modelling tool. The invoicing tool will need data from both the sales system and the order system in order to produce monthly invoices. The invoice system will also be used to simulate – i.e. performing a what-if analysis – changes in the price structure, and will need information from the service register.

The service register uses products and other services as building blocks to design and build new services. Products are items that the company orders from sub-contracting companies, such as software and hardware. Several products can be combined into one service; the products in a single service do not necessarily come from the same sub-contractor. Services are to be sorted, using a list of categories.

Both products and services share the same price structure. Each entity can have more than one price. First, a price is either a purchasing price or a retail price; the purchase price is imported from the sub-contractor. The financial manager, depending on demand, market situation and other considerations, decides on the retail price. When no retail price is specified the system should suggest a retail price that is a 20 percent increase from the purchasing price. A price does also have a type:

- Product, a price that is paid once for the acquisition of a product.
- Subscription, a price that is recurring monthly and that specifies a fixed amount
- Consumption, a price that is recurring, and that is specifying the unit price of a resource (i.e. a price for disk usage, NOK 5 for 10 MB).

Finally, prices have a start and a stop date. This makes it possible to specify future price changes that do not take immediately effect.

Services cannot be deleted. When a service is not supported anymore it may still be in use by some customers. Services need therefore to be marked as inactive.

The typical use of the system includes

1. Registration and import of products from sub-contractors

2. Registration and import of price changes for the retail prices of products
3. Registration of services
4. Updating services
5. Approval of purchasing prices
6. Approval of services
7. Simulation of price changes
8. Producing statistics
9. Exporting services to other systems

This use indicates also the workflow involved in registering services. New services are suggested, including the purchasing prices. Next, both the pricing and the service need to be approved by a financial manager and a product manager. Only services that are approved by both the financial manager and the product manager can be exported to other systems.