

INF 1050

OBLIGATORISK OPPGAVE 2

OBJEKT-ORIENTERT MODELLERING MED UML

4 sider

LEVERINGSFRIST: Fredag 3/4-2009, kl. 16:00

Evaluering: Bestått/Ikke bestått. Du må ha bestått denne obligatoriske oppgaven for å gå opp til eksamen. I bedømmingen av denne oppgaven, vil det legges vekt på forståelsen av pensum som du viser gjennom modelleringen.

Formål: I denne oppgaven skal du lære å gjøre en objekt-orientert analyse med UML, samt få trening i verktøyet Rational Rose.

Forutsetninger: Oppgaveteksten vil uten videre bruke begreper fra forelesninger, lærebøker og anbefalt lesestoff. Du kan laste ned en brukerveiledning/innføring i Rational Rose der du fant denne obligatoriske oppgaven.

Leveranser: Du skal levere en individuell besvarelse. Du skal levere en fil <fornavn_etternavn>OBLIG2.pdf som inneholder din besvarelse på oppgavene.

Levering: Fila skal sendes til **inf1050-<gruppenr>@ifi.uio.no**, der <gruppenr> er nummeret på gruppa du går på. I subject-feltet skal det stå **INF1050-oblig2**, slik at innleveringen kan registreres automatisk. Du bør dessuten sende eposten fra din Ifi/UiO-konto for å unngå at innleveringen din stanses av spamfiltre. Leveringsfristen er absolutt. Gruppelærer har intet mandat til å gi utsettelse (annet enn ved sykdom ved egenmelding). Eventuelle utsettelse må tas opp med studieadministrasjonen.

Krav om autentisitet: Se skriv fra instituttet på neste side.

Gjør deg kjent med instituttets reglement for obligatoriske oppgaver, som du finner her:
<http://www.ifi.uio.no/studier/skjemaer/oblig-retningslinjer.pdf>



Institutt for informatikk

Krav til innleverte oppgaver ved Institutt for informatikk (Ifi)

Ved alle pålagte innleveringer av oppgaver ved Ifi – enten det dreier seg om obligatoriske oppgaver, hjemmeeksamen eller annet – forventes det at arbeidet er et resultat av studentens egen innsats. Å utgi andres arbeid for sitt eget er både ulovlig og uetisk og kan medføre sterke reaksjoner fra Ifis og Universitetets side, for eksempel utvisning i ett eller flere semestre; se <http://www.uio.no/admhb/reglhb/studier/andre-regelverk/fuskesaker.xml>.

Vær derfor oppmerksom på følgende:

- Hvis du tar med tekst, programkode, illustrasjoner og annet som andre har laget, må du tydelig merke det og angi hvor det kommer fra.
- Det er greit å få hint om hvorledes en oppgave kan løses, men dette skal eventuelt brukes som grunnlag for egen løsning og ikke kopieres uendret inn.
- Du kan bli innkalt til samtale om dine innleveringer. Du må da kunne forklare innholdet i detalj og redegjøre for hvorledes det innleverte arbeidet er blitt til.

Gruppearbeid

I noen kurs skal det leveres gruppearbeid. Ifi krever da at alle medlemmer av gruppen kan gjøre rede for hovedtrekkene i det innleverte arbeidet. Dessuten må alle ha utført en rimelig del av det hele, og kunne identifisere og svare i detalj for sin del.

Samarbeid

Disse kravene betyr ikke at Ifi fraråder samarbeid – tvert imot, Ifi oppfordrer studentene til å utveksle faglige erfaringer om det meste. Men det kreves som nevnt at man kun leverer besvarelser man har produsert selv.

Hvis du er i tvil om hva som er lovlig samarbeid, må du kontakte gruppelærer eller faglærer.

PROBLEMBESKRIVELSE:

Du og IKT-ansvarlig hos NoChoice har kommet fram til en foreløpig liste over det dere anser som de viktigste funksjonelle kravene til systemet. De er gitt i tabellen under.

Krav nr.	Beskrivelse
1	Systemet skal kunne gi en oversikt over alle byene der NoChoice har hoteller.
2	Systemet skal kunne gi en oversikt over alle NoChoice-hotellene i en valgt by.
3	Systemet skal kunne vise detaljer for et gitt hotell. Disse detaljene skal minst inkludere beliggenhet, romtyper, ledige rom av forskjellige typer i et gitt tidsrom og priser.
4	Man skal kunne reservere et rom av en gitt romtype til en gitt pris for et gitt tidsrom for et gitt antall personer.
5	Systemet skal lagre informasjon om alle kunder som har reservert rom i NoChoice
6	Systemet skal håndtere kredittkortbetaling.
7	Reservasjoner skal kunne endres eller kanselleres.
8	En administrator skal kunne oppdatere listen av byer der NoChoice har hoteller.
9	En administrator skal kunne oppdatere listen av hoteller.
10	En administrator skal kunne oppdatere detaljer vedrørende et gitt hotell.

Oppgave 1: Ved hjelp av use case-modellering skal du nå strukturere kravene som er uformelt beskrevet i tabellen. Merk at hvert krav potensielt kan gi opphav til flere use cases. Det kan også være felles funksjonalitet mellom de forskjellige kravene (som kan gi opphav til felles <<include>> use cases). Lag et use case-diagram med aktører og use cases i Rational Rose. NB: Som det fremkommer i neste oppgave skal et av use casene hete *Reserver rom*.

Du skal nå fokusere på en liten del av systemet, nærmere bestemt et bestemt bruksmønster (use case) som heter *Reserver rom*, hvor aktøren *Bruker* (som kan være *Hotellansatt* eller *Kunde*) skal kunne reservere et rom med en gitt romtype (enkeltrum eller dobbeltrum) for et gitt tidsrom og for et gitt antall personer. Bruksmønsteret starter ved at Bruker oppgir ønsket hotell (anta at hotellnavn er unike), romtype (enkeltrum eller dobbeltrum), antall personer samt periode (ankomstdato og antall døgn). Systemet vil deretter opplyse om hvorvidt ønsket romtype er tilgjengelig samt evt. pris (hvis ønsket rom er tilgjengelig). Bruker må deretter bekrefte reservasjonen og hvis kunden ønsker rommet må bruker oppgi kundeinfo. Du skal anta at systemet ønsker å ha oversikt over alle kundene som på et eller annet tidspunkt har gjort en reservasjon, og at kundeinfo for nye kunder kan spesifiseres som en del av bruksmønsteret. Du skal for enkelhets skyld anta at alle kunder er unikt identifiserbare basert på sitt navn og sin fødselsdato, og at kundeobjektene må inneholde navn, fødselsdato, adresse, tlfnr og kredittkortnr.

Oppgave 2: Lag en tekstlig spesifisering av bruksmønsteret *Reserver rom* med prebetingelser, postbetingelser, hovedflyt og alternative flyt. Beskriv eventuelle antakelser.

Oppgave 3: Du skal nå lage sekvensdiagrammer i Rational Rose for bruksmønsteret du spesifiserte i oppgave 2. I den forbindelse kan følgende objekter med tilhørende ansvarsområder og metoder være nyttige (beskrevet i tilnærmet CRC-format), men merk at dette forslaget ikke nødvendigvis er komplett, og du står fritt til å endre eller utvide forslaget (spesifiser endringene/utvidelsene dine):

Kant: Videreformidler meldinger (informasjon) fra aktør til kontrollobjekt og meldinger fra kontrollobjekt til aktør.

ReserverRom: Ansvarlig for handlingsforløpet. Kommuniserer med kantobjektet samt forretningsobjektene (du må avgjøre hvilke).

Hotellkjede: Vet sine hoteller og kunder. Hint: Du kan anta at Hotellkjede har (minst) følgende metoder som du ikke trenger å spesifisere innholdet av i sekvensdiagrammene:

finnHotell(hotellNavn): Hotell

finnKunde(navn, fødselsdato): Kunde

opprettKunde(navn, fødselsdato, adresse, tlfnr, kredittkortnr): Kunde

Hotell: Vet sitt navn (anta at det er unikt), vet hvilke rom hotellet har, vet hvordan man finner et enkeltrom eller dobbeltrom som ikke er reservert i en gitt periode. Hint: Du kan anta at Hotell har (minst) følgende metode som du ikke trenger å spesifisere innholdet av i sekvensdiagrammene:

finnLedigRom(ankomstdato, antallDøgn, romtype): Rom

Rom: Vet om det er enkeltrom eller dobbeltrom, døgnpris, og sitt romnummer. Hint: Du kan anta at Rom har (minst) følgende metode som du ikke trenger å spesifisere innholdet av i sekvensdiagrammene:

beregnPris(ankomstdato, antallDøgn): Integer

Kunde: Vet sitt navn, fødselsdato, adresse, tlfnr og kredittkortnr. Vet sine reserverasjoner (slik at man ved en annen anledning kan finne en gitt reserverasjon for kunden).

Reserverasjon: Vet sin kunde (du kan anta at en reserverasjon er knyttet til nøyaktig en kunde selv om reserverasjonen omfatter flere enn en person), hvilket rom som er reservert for en gitt periode, antall personer (typisk 1 eller 2) rommet er reservert for.

a) Lag et sekvensdiagram for bruksmønsterets hovedflyt

b) Lag et sekvensdiagram for alternativ flyt hvor kunden ikke eksisterer

Oppgave 4: Lag et klassediagram i Rational Rose som tilsvarer de to sekvensdiagrammene du laget i oppgave 3. Klassediagrammet skal inkludere metoder, assosiasjoner og eventuelle attributter som er nødvendig for utførelsen av bruksmønsteret med hovedflyt og variasjonen at kunden ikke eksisterer.

Til slutt: Det du har gjort i denne obligatoriske oppgaven hører hjemme under C 4.2 og C 4.3.1 i bilag C i PS2000, men du behøver *ikke* å kopiere dette over i PS2000-kontrakten.

SLUTT