

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Eksamen i: INF1050

Eksamensdag: 3. Juni, 2008

Tid for eksamen: 09:00-12:00

Oppgavesettet er på 3 sider

Vedlegg: Ingen

Tillatte hjelpemidler: Alle trykte og skrevne

*Kontroller at oppgavesettet er komplett
før du begynner å besvare spørsmålene.*

Innledning

Forsikringsselskapet *Forsikring2* har en rekke spesialpakker med forsikringsløsninger som er tilpasset forskjellige befolkningsgrupper. Blant annet har de *NHO Worries* for NHO-medlemmer, *Distre Pluss* for akademikere og *Komplett Total* for de som har alt. Den nyeste satsningen til selskapet er pakken *Trang Hybel* for studenter. Ditt IT-selskap *Try-IT* har blitt forespurt om å delta i en anbudskonkurranse for mulig levering av en IT-løsning for det nye produktet. Produktet skal utelukkende tilbys og selges via internett.

Oppgave 1 Tidlig fase (20%)

- Skisser hvordan en moderne logisk og fysisk lagdelt arkitektur kan se ut.
- Man er svært opptatt av å lykkes med det nye produktet. For å fastsette kravene til IT-systemet til *Trang Hybel*, må synspunkter fra de rette interessentene ("stakeholders") tas hensyn til. Hvilke interessenter bør være representert, og hvilken kravinnhentingemetode vil du bruke for hver interessent? (Nevn minst 3 interessenter). Begrunn kort.
- Hva synes du er de tre viktigste ikke-funksjonelle kravene til IT-systemet for *Trang Hybel*. Begrunn. Skisser kort hvordan du kan evaluere at disse kravene oppfylles.
- Du (som leverandør av IT-systemet) og kunden diskuterer hvilken kontraktstandard dere skal bruke. Hvilke tre kontraktstandarder kjenner du til, og hva er den viktigste forskjellen mellom dem?

Oppgave 2 Prosjektstyring (20%)

- IT-ansvarlig i *Forsikring2*, som tidligere har vært med på en del mindre men svært vellykkete IT-prosjekter, sier at han tror det ville ta deg og ditt team omlag 9000 timesverk å lage første versjon av systemet for *Trang Hybel*. Han begrunner dette med at de prosjektene han var med på, var omtrent en tredjedel av størrelsen, og at de tok rundt 3000 timesverk. Du har bedt om å få se kravspesifikasjonene til de prosjektene han snakker om. Noen av spesifikasjonene er svært saklige og bruker ikke flere ord enn nødvendig, mens andre har lange utlegninger, bl.a. om de ønsket positive følgene av det ferdige systemet. Nevn noen faktorer ditt team må være oppmerksomme på, slik at dere estimerer tidsbruk så realistisk som mulig i dette prosjektet.

- b) Anta følgende: Du vet at kravinnhenting er det første som må gjøres. Siden må kravene modelleres i use cases. Men så snart et use case er ferdig beskrevet, kan en mer formell modellering gjøres av use case i form av sekvens- og klassediagrammer, uavhengig av om de andre use casene er ferdige. Ved hjelp av planning poker, har ditt team estimert at kravinnhenting K vil ta 20 enheter tid. Dere antar på dette stadiet at dere kommer til å få 4 overordnede funksjonelle krav A, B, C, D. Dere estimerer at det vil ta 5 enheter tid å lage et use case samt modeller for A, at det vil ta 4 enheter for B, 7 enheter for C og 5 enheter for D. Når use case og modellene for A og C er ferdige, kan kode, databaseskjemaer og brukergrensesnitt genereres for disse. Dette vil ta totalt 4 enheter. Når modellene for B og D er ferdige, kan kode, databaseskjemaer og brukergrensesnitt genereres for disse. Dette vil også ta totalt 4 enheter. Testing skal gjøres sammen for A og D (totalt 5 enheter), men skal gjøres enkeltvis for B (5 enheter) og C (5 enheter). Tegn et nettverks-diagram, og angi de aktivitetene som ligger på kritisk vei, samt den aktiviteten med størst slakk.

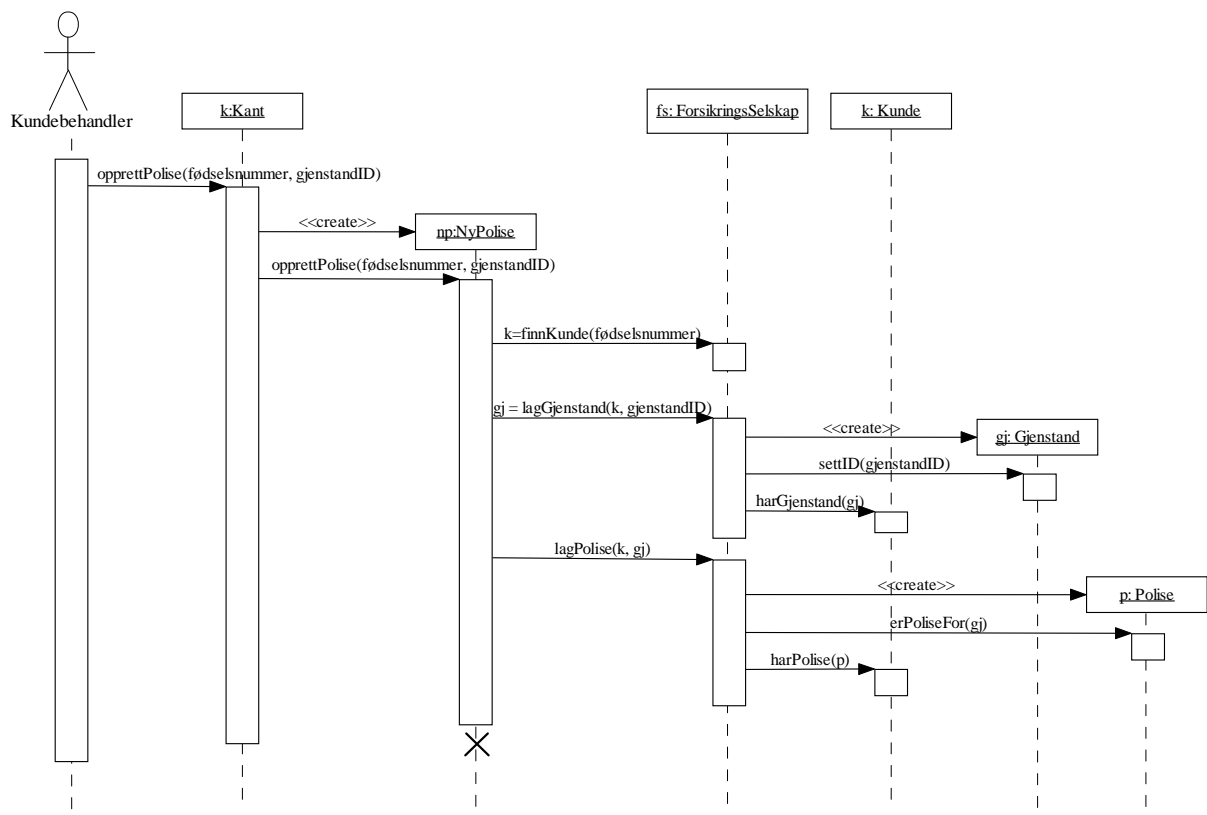
Oppgave 3 Modellering (40%)

Du skal nå modellere den delen av systemet som registrerer et salg av en polise under *Trang Hybel*. Systemet skal holde oversikt over kunder, poliser og tilhørende gjenstander (bolig, mobiltelefon, sykkel, og lignende) som er forsikret i *Forsikring2*.

For enkelhets skyld skal du anta følgende:

- (1) Kunder er identifisert ved *fødselsnummer*.
 - (2) Alle mulige gjenstander som kan forsikres kan identifiseres ved hjelp av en identifikator som heter *gjenstandID*. Du trenger ikke spesifisere hvordan denne identifikatoren lages.
 - (3) Alle gjenstander som forsikres kan ha kun en polise som er assosiert med kun en kunde.
 - (4) En polise kan forsikre kun en gjenstand.
- a) Vi tenker oss at denne delen av funksjonaliteten til *Trang Hybel* kan modelleres som tre Use Cases (bruksmønstre): "Ny Polise", "Ny Kunde" (som aktiveres dersom kunden som ønsker en polise ikke eksisterer i systemet fra før) og "Ny Gjenstand". Primær aktør er *Kundebehandler*. Lag et UML Use Case diagram med disse tre bruksmønstrene. Spesifiser evt. avhengigheter mellom dem ved hjelp av *include* og/eller *extend*. Gjør rede for antakelser.
- b) Lag en tekstlig spesifikasjon av bruksmønsteret "Ny Polise" med hovedflyt og alternative flyt, prebetingelser og postbetingelser. "Ny polise" skal registrere en ny polise på en spesifisert gjenstand for en spesifisert kunde. Du skal anta at i normaltilfellet så er kunden allerede registrert men at gjenstanden *ikke* er registrert i systemet. Dersom kunden ikke eksisterer må ny kunde opprettes. Dersom gjenstanden allerede er registrert må man sjekke at det ikke allerede eksisterer en polise på gjenstanden. Gjør rede for evt. andre antakelser du gjør.

Under ser du et sekvensdiagram for hovedflyt for "Ny Polise".



- Hvilke forretningsobjekter finnes i sekvensdiagrammet?
- Gi en vurdering av i hvilken grad sekvensdiagrammet har en sentralisert eller delegert kontrollstil.
- Lag et klassediagram som tilsvarende sekvensdiagrammet, hvor alle klasser og metoder samt assosiasjoner mellom forretningsobjektene med multiplisitet er spesifisert. Du trenger *ikke* spesifisere attributter eller avhengighetspiler fra kant- og kontrollobjekt.
- Med utgangspunkt i sekvensdiagrammet, lag et nytt sekvensdiagram hvor kontrollobjektet ber om *fødselsnummer* og *gjenstandID* fra kantobjektet (i stedet for at kontrollobjektet får disse som parameter i metoden *opprettPolise*). Du trenger ikke tegne hele sekvensdiagrammet på nytt, men kun de delene hvor det blir endringer.

Oppgave 4 Testing (20%)

- Hvorfor er det typisk flere testaktiviteter (f.eks. enhets-, integrasjons-, systemtesting) i systemutvikling?
- Når to komponenter som skal kommunisere med hverandre har passert enhetstestene sine, hvorfor er det viktig å gjøre integrasjonstesting?
- Hva er fordelene og ulempene ved henholdsvis whitebox- og blackbox-testing? Illustrer med eksempler på teknikker innen de to kategoriene.