

INF3120 - Obligatorisk prosjektarbeid

INNHOOLD:

Krav til innleverte oppgaver ved Institutt for informatikk	2
<i>Gruppearbeid</i>	2
<i>Samarbeid</i>	2
1 Bakgrunn	3
2 Læringsmål.....	3
3 Vurderingskriterier	3
4 Organisering av prosjektarbeidet	4
4.1 Grupper.....	4
4.2 Viktige aktiviteter og leveranser	5
4.3 Støtte til prosjektarbeidet	6
4.4 Prosjektgruppene	6
5 Kravbeskrivelse.....	7
5.1 Fullstendig kravspesifikasjon.....	7
6 Retningslinjer for leveransene.....	7
6.1 Generelt	7
6.1.1 Tittelside.....	7
6.1.2 Dokumenter	7
6.1.3 Lagringsformat	8
6.1.4 Endringer.....	8
6.2 Leveranse I (innleveringsfrist: 22.09, vektning: 10%).....	8
6.2.1 Prosjekthjemmeside	8
6.2.2 Prosjektplan	8
6.3 Leveranse II (innleveringsfrist: 20.10, vektning: 30%)	9
6.3.1 Use case og domenemodell	9
6.3.2 UML design v.h.a. klassediagram.....	11
6.3.3 Kodegenerering og implementering av basisfunksjonalitet	11
6.4 Leveranse III (innleveringsfrist: 27.10, vektning: 30%)	12
6.4.1 Utdeling av felles basisløsning.....	12
6.4.2 Oppdatert prosjektplan (midtrapport).....	12
6.5 Leveranse IV (innleveringsfrist: 17.11, vektning: 30%).....	13
6.5.1 Design av ny funksjonalitet.....	13
6.5.2 Kodegenerering og implementering av ny funksjonalitet	13
6.6 Presentasjon av prosjektet (Dato: TBD).....	14
6.6.1 Presentasjon av prosjektet	14

Informasjon fra Institutt for informatikk

Det presiseres at den obligatoriske oppgaven

- må bestås for å kunne bestå emnet
- skal løses som del av en prosjektgruppe på 4-5 personer
- teller 25 % av totalkarakteren i kurset

Krav til innleverte oppgaver ved Institutt for informatikk

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 uetisk og kan medføre sterke reaksjoner fra Ifis side.

Derfor gjelder følgende:

1. Hvis du tar med tekst, programkode, illustrasjoner og annet som andre har laget, må du tydelig merke det og angi hvor det kommer fra.
2. 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.
3. Kursledelsen kan innkalle studenter til samtale om deres innlevering.

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

Reglene om kopiering 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 kan stå inne for det som leveres.

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

1 Bakgrunn

Et viktig prinsipp for INF 3120 er sekvensen: LÆRE - BRUKE - EVALUERE. Forelesningene og egenlæringen vil ofte ha fokus på tilegnelse av kunnskap (LÆRE). Øvelsene i grupper og, ikke minst, prosjektoppgaven har fokus på oppøvelse av ferdigheter (BRUKE). Prosjektoppgaven innebærer i tillegg øvelse i EVALUERING av eget og andres arbeid. Prosjektoppgavens rolle er altså i hovedsak å bidra til at kunnskapen fra forelesningen bidrar til ferdigheter og evalueringsevner innen systemutvikling. Dere vil, av ressursmessige grunner, ikke kunne gjennomføre et 100% realistisk prosjekt. Selv om læringseffekten har hatt størst fokus, har det likevel vært vårt mål å ha en stor grad av realisme i prosjektgjennomføringen. Graden av realisme avhenger imidlertid mye av dere!

For å kunne gå opp til eksamen i INF 3120 må prosjektgruppen du tilhører ha fått godkjent alle prosjektleveransene. Dette er kursets obligatorisk oppgave som teller 25% av total karakteren.

Viktig: Det er mye som skal klaffe for at gjennomføringen av prosjektoppgaven skal gå knirkefritt. Regn med at ikke alt er strømlinjeformet. Kom gjerne med forbedringsforslag og kritikk underveis. Det vil bli gjennomført en underveisevaluering midtveis i semesteret hvor dere kan komme med konstruktiv kritikk.

2 Læringsmål

Gjennom prosjektarbeidet skal det bl.a. oppnås ferdigheter og evalueringskompetanse innen:

- 1) Prosjektplanlegging og gjennomføring
- 2) Beskrivelse av krav vha UML Use Cases
- 3) Utarbeidelse og gjennomgang av objektorientert design vha UML
- 4) Bruk av CASE verktøy i systemutviklingsprosessen

Ferdighetene skal i størst mulig grad være ihht prinsipper innen "software engineering".

NB: Det forutsettes at dere har noen forkunnskaper om UML (f eks fra INF1050/INF102) eller skaffer dere elementærkunnskap om UML på egenhånd.

3 Vurderingskriterier

Hver del leveranse vil inneholde konkrete kriterier for hva som skal evalueres og vurderes. Hver del leveranse må uansett oppfylle noen generelle krav. De generelle kravene for å få godkjent en leveranse må oppfylle følgende kriterier:

- Leveransen må være **levert innen angitt tid**
- Leveransen må oppfylle alle **SKAL-krav** i retningslinjene (beskrevet i dette dokumentet)
- Leveransen må ha **tilstrekkelig kvalitet**

Tilstrekkelig kvalitet vurderes av kursansvarlige og gruppelærere på bakgrunn av om leveransen:

- 1) er egnet grunnlag for videre arbeid i prosjektet, og
- 2) viser at studentene kan anvende kunnskapen fra INF 3120-forelesningene og andre relevante informasjonskilder.

Dersom en leveranse ikke blir godkjent i første omgang eller blir forsinket betyr dette at prosjektet må forbedre leveransen og får dårligere tid på de resterende leveransene. Prosessen underveis vil være et moment ved vurdering av godkjent/ikke godkjent av gruppens arbeid som helhet. Vi vil **IKKE** akseptere at noen "spekulerer" i å levere halvgode leveranser for at gruppelærer skal komme med innspill på hva som mangler for at leveransen blir godkjent. Dersom en leveranse ikke oppfyller alle SKAL-kravene vil prosjektgruppa få tilbake leveransen uten kommentarer med beskjed om at "leveransen mangler SKAL-krav". Utfyllende kommentarer til leveransene gis først når alle SKAL-krav er dekket.

Viktig: Det kan være gode grunner til at leveranser ikke blir fullført, f.eks problemer med verktøy eller sykdom. Dersom slike situasjoner inntreffer skal grunner til at leveransen ikke ble fullført beskrives og sendes til kursansvarlige, sammen med hva prosjektet har gjort for å løse dette. Skissér også en revidert realistisk plan for det videre arbeidet. På bakgrunn av en slik beskrivelse kan også ufullstendige leveranser godkjennes. Hvis en person er syk i noen dager, så ansees ikke det som en holdbar grunn for at prosjektet eventuelt er blitt forsinket. At noen i prosjektgruppa blir syke er noe man må ta hensyn til i selve prosjektplanleggingen (risiko-faktor).

4 Organisering av prosjektarbeidet

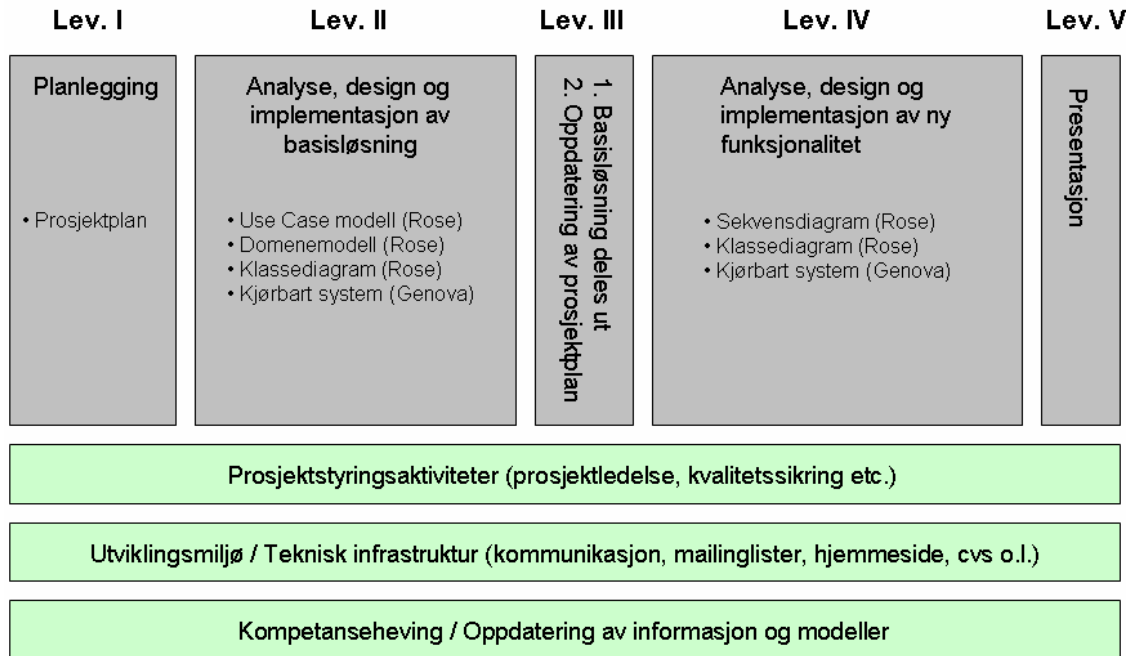
4.1 Grupper

Du vil tilhøre en gruppe på 4-5 personer. Gruppetilordning bestemmes av kursansvarlige på bakgrunn av en kartlegging av erfaring til den enkelte. Oppdragsgiver til prosjektet vil være en av gruppelærerne.

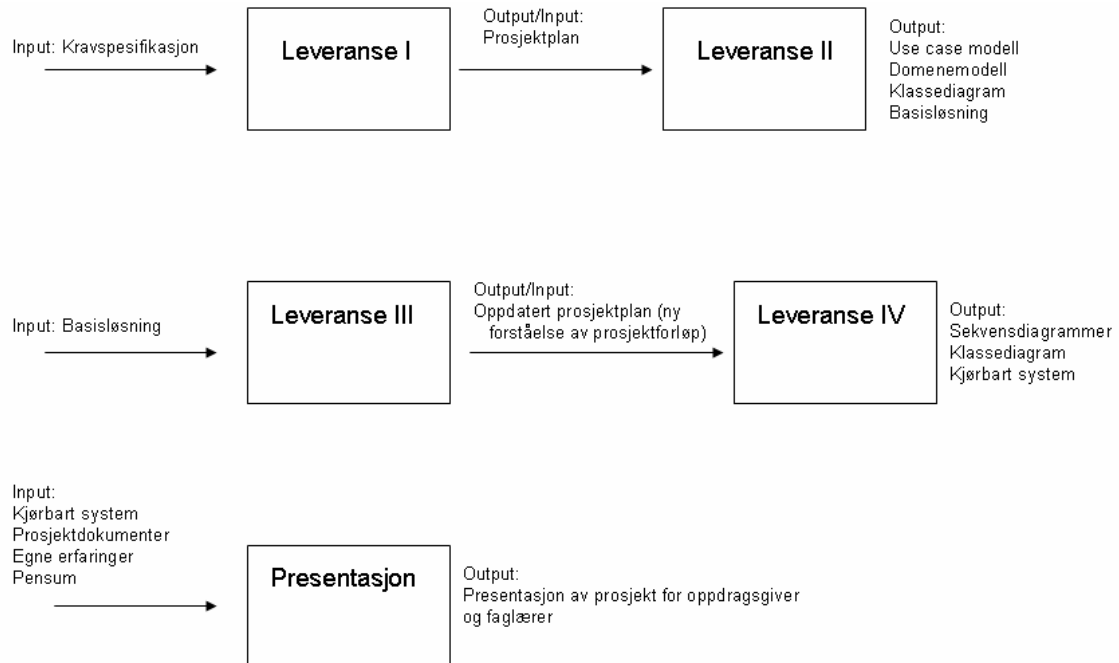
Alle i gruppen bør ha god oversikt over alt som foregår, men alle behøver ikke å utføre alt.

Dersom en leveranse ikke godkjennes må det arbeides mer på denne og det blir tilsvarende dårligere tid på de neste leveransene. Sørg for at det ikke skjer! Foreta gjerne noen kontroller mot gruppelærer underveis for å sikre at leveransen blir god nok.

4.2 Viktige aktiviteter og leveranser



Figur 1: Leveranser og prosesselementer i gjennomføringen av INF3120 prosjektet.



Figur 2: Input til og output fra de ulike leveransene.

Viktige aktiviteter:

- 1) Etablering av prosjektgruppene (inndeling i prosjektgrupper, opprettelse av prosjekt-hjemmesider, etablering av roller, etc.)
- 2) Oppdragsgivers kravspesifikasjon legges ut på INF3120-hjemmesiden (Prosjektnavn: Project Hospital 2006, Systemnavn: HSS)
- 3) Utarbeidelse av prosjektplan
- 4) Kurs i bruk av Rational Rose og Genova (holdes av Esito)
- 5) Utarbeidelse av Use Case Modell og domenemodell for basisløsningen.
- 6) Utarbeidelse av detaljert og fullstendig UML design for basisløsningen.
- 7) Kjørbart system: Generere kode og GUI fra Genova for basisløsningen
- 8) Midtrapport: Oppdatering/forbedring av prosjektplan basert på erfaringene så langt og ny informasjon om gjenstående arbeid.
- 9) Design og implementasjon av ny funksjonalitet (sekvensdiagrammer, klassediagrammer)
- 10) Generere kode vha Rose og Genova og eventuelt gjøre manuelle endringer i kildekoden
- 11) Presentere systemet for oppdragsgiver (gruppelærer)
- 12) Presentasjon av prosjektet for faglærere.

Etter hver leveranse vil gruppelærerne gi tilbakemeldinger og bestått/ikke bestått. Tidsfrister for leveransene er beskrevet i kapittel 6 i dette dokumentet.

4.3 Støtte til prosjektarbeidet

Prosjektene skal i størst mulig grad klare seg selv og bruke medstudenter. Dersom problemer ikke kan løses på denne måten er neste steg å ta kontakt med gruppelærer som er oppdragsgiver. Dersom det likevel er behov for oppklaringer eller støtte er ansvaret for dette hos:

Verktøy	RM (Rustam Mehmandarov)
Kravspesifikasjon/design	HG (Hans Gallis)
Prosjektplan/styring	MJ (Magne Jørgensen)
Annet	HG/MJ/DS (Dag Sjøberg)

4.4 Prosjektgruppene

Prosjektgruppene vil bli delt inn, så langt det er mulig, på bakgrunn av kartleggingen fra første forelesning. Informasjon om hvem som er på hvilken prosjektgruppe vil bli gitt på første gruppeøving og legges ut på hjemmesiden til kurset så snart inndelingen er klar.

5 Kravbeskrivelse

Ikke forvent at kravbeskrivelsen er 100% klar mht hvordan kravene skal forstås. Mindre viktige avklaringer tar dere innad i prosjektet. Viktige avklaringer tas med oppdragsgiver (gruppelærer). Avklaringene SKAL dokumenteres. Cirka en time av hver gruppeøvelse vil bli satt av til den obligatoriske oppgaven. Dersom prosjektgruppene forbereder seg godt vil man derfor kunne få avklart mye på gruppeøvelsene.

5.1 Fullstendig kravspesifikasjon

Finner dere på kurset hjemmeside (ProjectHospital2006.pdf).

6 Retningslinjer for leveransene

Tidsfrister for leveransene er beskrevet på kursets hjemmeside og i parentes ved forklaringen til hver leveranse i dette kapittelet. Dersom det slurves med retningslinjene og SKAL-kravene vil prosjektgruppene tape verdifull tid. Det er derfor viktig at prosjektgruppene sørger for at alle SKAL-krav er oppfylt når man leverer dokumentene til gruppelæreren.

6.1 Generelt

6.1.1 Tittelside

Alle dokumentleveranser SKAL ha en tittelside som angir:

- Nummer på din prosjektgruppe (+ evt. firmanavn)
- Identifikasjon av leveranse (f eks: Design – Leveranse II)
- En liste over medlemmene i gruppen/firmaet
- Angivelse av hvem som har bidratt med hva (f eks: Prosjektledelse: Magne, Kvalitetssikring: Lise)
- Dato for ferdigstillelse (+ ev. oppdatering) av dokument

6.1.2 Dokumenter

Alle dokumenter SKAL være godt strukturerte og oversiktlige. Et tips er å benytte strukturen i de eksisterende malene. Språket skal være godt gjennomarbeidet slik at det er mulig å lese teksten med god flyt.

6.1.3 Lagringsformat

Alle dokumenter SKAL legges på prosjektets hjemmeside på formatet html, postscript, pdf eller word.

6.1.4 Endringer

Det SKAL beskrives i leveransene hva som eventuelt er endret siden forrige leveranse (forrige iterasjon). Dette skal beskrives i begynnelsen av dokumentet eller i begynnelsen av det aktuelle kapittelet slik at oppdragsgiver lett kan se hva som er endret.

6.2 Leveranse I (innleveringsfrist: 22.09, vekting: 10%)

6.2.1 Prosjekthjemmeside

Alle prosjekter SKAL ha en prosjekthjemmeside, der alle relevante prosjektdokumenter legges på en oversiktlig måte. Prosjektdokumentene skal være lette å finne for gruppelærere og forelesere. Ikke bruk mye krefter på "fancy" layout og funksjonalitet. Prosjekthjemmesiden vil kun evalueres ut fra hvor lett det er å gjenfinne prosjektdokumenter.

Brukernavn og passord til prosjektgruppens hjemmeområde vil kunne ta noe tid å få generert (f eks vil prosjektgruppe 12 ha sitt område på ~i3120g12). Start derfor gjerne med å utvikle på et prosjektmedlems eget hjemmeområde. Bruk gjerne andre enkle teknologier istedenfor å lage hele web-siden fra scratch. Se for eksempel WIKI og Blog (se under "Relevante web-sider/dokumenter" på kursets hjemmeside for mer informasjon om disse teknologiene).

6.2.2 Prosjektplan

Prosjektgruppen bør tilstrebe å være realistisk i rollen som leverandør. Dette innebærer at gruppen opptrer koordinert overfor oppdragsgiver (gruppelærer). Vurder om det er hensiktsmessig å tildele roller innad i gruppen (f eks at en utpekes som prosjektleder, en som er ansvarlig for hjemmesiden, en med hovedansvar for design, test, konfigurasjonsstyring, kvalitetssikring osv.).

Det er ingen krav om at gruppen velger en prosjektleder/gruppeleder, men erfaringsmessig (erfaringer fra tidligere prosjektoppgaver i INF 3120 (IN 219) og andre universiteter) er dette svært nyttig for prosjektet. Regelmessige prosjektmøter er viktig både for læringen og for prosjektsuksessen.

Det kan også være lurt å la en eller to personer være såkalte kvalitetssikrere. Det vil si at en eller to personer er ansvarlig for å sikre at alle SKAL-krav (som er beskrevet i dette dokumentet) er oppfylt og at rapporten oppfylder de generelle kriteriene før man leverer

til gruppelærer. På den måten kan man unngå at leveransen blir underkjent p.g.a. unødvendige slurvefeil.

Noen tips for å få størst mulig kvalitet i prosjektplan og i videre arbeid:

- Start på prosjektorganiseringen og planleggingen umiddelbart.
- Ha fokus på neste leveranse, men jobb hele tiden i parallell med å forberede andre aktiviteter. Det er f eks viktig at dere tenker igjennom hva som trengs av kunnskap for å gjennomføre de ulike aktivitetene.
- Når flere jobber på samme dokument/program bør konfigurasjonsstyring innføres eller i det minste regler for oppdatering og format på dokumenter og programmer avtales (filstruktur og navngiving av filer).
- Forbered dere på at estimering av ressursforbruk er vanskelig og må oppdateres jevnlig som følge av ny innsikt. Ikke forvent svært nøyaktige estimer.

Kravene til hva prosjektplanen SKAL inneholde, samt forslag til struktur og estimeringsstøtte, er beskrevet i maler (se mal_prosjektplan_2005.rtf) og estimeringsark (se mal_aktivitetsplan_2005.xls) som finnes på INF3120-sidene. Disse malene SKAL brukes. Tips og råd fra Sommerville-boka bør også tas hensyn til.

Kriteriene vi bruker for å evaluere leveranser der plan og estimerer inngår er i hovedsak:

- 1) *Hvor godt planen egner seg som styringsinstrument, dvs i hvilken grad de tingene som er viktig for styringen (SKAL-kravene) bidrar til at prosjektet blir effektivt gjennomført. Dette vil vi evaluere både ved å se på planen, men også på selve gjennomføringen – f eks om godkjent leveranse leveres på tid.*
- 2) *Om alle SKAL-kravene er oppfylt.*
- 3) *I hvilken grad planen viser at pensum om prosjektplanlegging og styring er forstått og brukt.*

Viktig: Husk at dere SKAL registrere timeverk dere (hver enkelt) har brukt på aktivitetene/leveransene. Bruk samme aktivitetsstruktur som i estimeringsarket for denne registreringen. Presentasjonen (leveranse V) vil IKKE bli godkjent dersom sammenligning av estimert og virkelig forbruk av arbeidsmengde per aktivitet mangler! Det er altså viktig at hver deltaker i prosjektet registrerer timer underveis.

6.3 Leveranse II (innleveringsfrist: 20.10, vekting: 30%)

6.3.1 Use case og domenemodell

Use case modellen SKAL dekke hele systemet, det vil si all funksjonalitet som er beskrevet i kravspesifikasjonen. Vurdér oppdragsgivers krav og tekniske konsekvenser som følger av disse. Se også tabellen i figur 6.17 i Sommerville for en oversikt over hva

en kravspesifikasjon bør inneholde. Dette vil gi dere gode tips om hva som er relevant informasjon.

Use case modellen skal bestå av

1. et use case diagram som viser aktører og use case,
2. en kort beskrivelse av hver aktør og
3. en tekstlig beskrivelse av hvert use case i henhold til Larman seksjon 6.8 (fully dressed). Fem av use casene skal beskrives detaljert, mens de resterende skal beskrives kort.

Lag også en domenemodell som beskriver entitetene i use casene. Domenemodellen skal dekke hele systemet.

Beskriv de ikke-funksjonelle krav knyttet til de enkelte use casene eller til systemet som helhet.

Fokuser på at use case modellen inneholder nok informasjon til å designe systemet og senere teste at systemet tilfredsstiller de funksjonelle kravene.

Use case modellen SKAL kvalitetssikres. Følgende sjekklister støtter denne kvalitetssikringen:

1. Aktører:
Se på aktørene i use case modellen.
 - Er alle aktørene som deltar i use casene klart definert?
 - Er det klart hvilke aktører som er involvert i hvilke use case?
 - Er beskrivelsen av hvordan systemet interagerer med en aktør konsistent med beskrivelsen av aktøren?
 - Er forventet input og output definert for hver aktør?
2. Use case modellen som helhet:
Se på use case diagrammet, de tekstlige beskrivelsene av use casene og domenemodellen.
 - Er det samsvar mellom use cases og domenemodell, dvs. gjenfinner dere konseptene som beskrives i use case modellen i domenemodellen og vice versa?
3. Hvert enkelt use case:
Gå gjennom hver enkelt use case beskrivelse.
 - Er beskrivelsene av hvert use case slik at det er forståelig hvilke aktiviteter som er nødvendig for at aktøren skal nå sitt mål?
 - For hvert use case, er systemets output på alle aktørens input for både gyldig og ugyldig hendelsesflyt definert?
 - Er det ikke-verifiserbare ord eller setninger i use casene, som f.eks. "fungerer bra", "rask", "god ytelse", "skjer vanligvis"?
 - Er hendelsesflyten i use casene beskrevet i konkrete termer og med målbare kvantiteter?

- Er alle pre-betingelser og post-betingelser beskrevet på en slik måte at de kan testes?
4. Sammenhengen mellom use casene:
 Se på hvordan use casene tilsammen beskriver ønsket funksjonalitet.
- Brukes include-relasjonen til å faktore ut felles oppførsel?
 - Matcher pre- og post-betingelsene for de enkelte use casene?

6.3.2 UML design v.h.a. klassediagram

Designmodellen skal dekke basisløsningen (funksjonaliteten er merket med "Basisløsning" i kravspesifikasjonen "Project Hospital 2006"). Selve designmodellen består av ett eller flere klassediagram. Det er ikke påkrevd at dere leverer sekvensdiagrammer i leveranse II men det kan lønne seg å utarbeide sekvensdiagrammer i designprosessen for å få en bedre forståelse av hva det er dere skal lage. Dersom dere utarbeider sekvensdiagrammer så må dere gjerne legge ved disse. Designmodellen skal dekke hele systemet. Følg designprinsippene fra Larman seksjon 16 (bruk av patterns som Information expert, Creator og Controller samt prinsippene om lav kobling og høy kohesjon) så langt som mulig.

For klassediagrammene gjelder følgende:

- Alle attributter og metoder skal ha en type (public, private, protected).
- Alle assosiasjoner skal ha angivelse av roller, multiplisitet og navigerbarhet (retning).

Det vil være behov for å beskrive hvilke antakelser og avgjørelser som er gjort. Ved behov vil det være hensiktsmessig å beskrive designet tekstlig (i selve UML-modellen eller som et vedlegg til selve modellen). Beskrivelsen bør være et svar på "hvorfor er det hensiktsmessig å gjøre det slik?".

Prosjektene vil få tilbakemelding fra gruppelærere (og evt. foreleser) på styrker og svakheter ved deres design.

6.3.3 Kodegenerering og implementering av basisfunksjonalitet

Dere vil få en steg-for-steg beskrivelse av hvordan kodegenereringen skal foregå vha Rational Rose og Genova. Det kan være hensiktsmessig å tidlig finne ut hva Genova "krever" av klassediagrammet for å generere kode før dere designer basisfunksjonaliteten (hva som MÅ være med av relasjoner mellom klasser, attributter, parametere etc. for at dere skal få generert kode).

På bakgrunn av den designmodellen (klassediagram) skal dere utvikle den nye funksjonaliteten ved hjelp av Rational Rose og Genova.

Dere trenger ikke å levere koden skriftlig. I dokumentet til denne leveransen skal dere beskrive hvor filene til det "nye" kjørbare systemet ligger slik at oppdragsgiver

(gruppelærer) kan teste det. Koden skal kunne lastes ned fra hjemmesiden til prosjektgruppa.

Det anbefales å benytte et konfigurasjonsstyringsverktøy til implementeringen (for eksempel cvs eller svn). Mer informasjon om dette finner dere på kursets hjemmesider.

Dere bør teste systemet skikkelig før dere setter det i produksjon (leveres kunden). Dere bestemmer selv hvordan dere skal teste systemet (basisløsningen), men dere SKAL beskrive i leveransen hvordan dette ble gjennomført.

6.4 Leveranse III (innleveringsfrist: 27.10, vektning: 30%)

6.4.1 Utdeling av felles basisløsning

Prosjektgruppene vil få utdelt en ferdig basisløsning som skal benyttes i det videre arbeidet. Denne basisløsningen kan være identisk med det dere selv har designet eller et eksempel på en alternativ måte å designe basisløsningen på. Alle prosjektgruppene må benytte dette i det videre arbeidet da prosjektgruppene skal ha samme utgangspunkt for å designe og implementere utvidelsen til basisløsningen (ny funksjonalitet).

Den viktigste oppgaven her er å sette seg inn i løsningen og forstå hvordan den fungerer. Denne informasjonen skal benyttes til å oppdatere prosjektplanen.

6.4.2 Oppdatert prosjektplan (midtrappert)

I denne leveransen skal dere stoppe opp i en kort periode for å se om prosjektet styres i riktig retning til riktig tid.

Funksjonaliteten som nå skal utvikles i den neste fasen er knyttet til Turnusplanlegging. Denne funksjonaliteten er markert i kravspesifikasjonen med "Leveranse IV".

Prosjektplanen (fra leveranse I) SKAL oppdateres ihht til ny kunnskap og økt forståelse (erfaringer fra prosjektarbeidet som allerede er gjennomført). Hele prosjektplanen SKAL evalueres i fellesskap innad i gruppa. De delene av prosjektplanen som krever tiltak skal endres og endringene SKAL merkes slik at gruppelærer lett kan se hva som er endret fra leveranse I. Prosjektgruppa bestemmer selv hva som krever oppdatering (hva har fungert dårlig og hvorfor, og hva har fungert bra og hvorfor), men følgende SKAL evalueres og oppdateres:

- Aktivitetsplan med tilhørende oppgaver og estimater.
- Risikoplan (topp 10 risikoliste)

Hvordan man gjennomfører denne evalueringsprosessen er opp til prosjektgruppa å bestemme. Et tips er å benytte Post Mortem Analyse (PMA). Dokumentasjon angående

PMA finnes på kursets hjemmeside. Hvordan man setter seg inn i det eksisterende (ferdig implementerte) systemet (basisløsningen) er også opp til prosjektgruppa å bestemme. Et par forslag kan likevel nevnes: For eksempel kan dere lese UML-modeller og kode enkeltvis og deretter ta en felles diskusjon for å oppklare uklarheter, eller dere kan diskutere koden parvis.

6.5 Leveranse IV (innleveringsfrist: 17.11, vekting: 30%)

6.5.1 Design av ny funksjonalitet

Dere skal benytte klassediagrammet fra basisløsningen, som ble utdelt ifm leveranse III, som utgangspunkt (og ikke klassediagrammet dere selv utarbeidet i leveranse II). Dette klassediagrammet skal utvides slik at det dekker turnusfunksjonaliteten (se kravspesifikasjonen for detaljert beskrivelse av hvilken funksjonalitet som skal implementeres). Designmodellen skal tilfredsstillende de samme kravene som i Leveranse II. I tillegg skal dere lage sekvensdiagrammer for den nye funksjonaliteten.

For sekvensdiagrammene gjelder følgende SKAL-krav:

- Et sekvensdiagram skal modellere hendelsesflyten i et og bare et use case.
- Alle use case som er relatert til den nye funksjonaliteten skal være modellert v.h.a. sekvensdiagrammer.

Dokumentet fra leveranse IV SKAL inneholde de opprinnelige UML modellene dere laget i begynnelsen av leveranse IV og oppdaterte UML modeller som beskriver det faktiske implementerte systemet (det er ikke sikkert at dere ender opp med det samme designet dere begynte leveranse IV med). Gruppelærerne vil kreve å se den opprinnelige designmodellen dere benyttet til implementeringen. Hvis det er betydelige endringer fra det opprinnelige klassediagrammet og det faktiske implementerte systemet SKAL årsakene til dette beskrives kort (hvorfor ble det slik?).

NB: Det SKAL (se del-leveranse nedenfor) genereres Java-kode vha Rational Rose og Genova på bakgrunn av designet. Ta hensyn til dette!

6.5.2 Kodegenerering og implementering av ny funksjonalitet

På bakgrunn av den nye designmodellen (klassediagram) skal dere utvikle den nye funksjonaliteten ved hjelp av Rational Rose og Genova som i leveranse II.

Dere trenger ikke å levere koden skriftlig. I dokumentet til denne leveransen skal dere beskrive hvor filene til det ”nye” kjørbare systemet ligger slik at oppdragsgiver (gruppelærer) kan teste det. Koden skal kunne lastes ned fra hjemmesiden til prosjektgruppa.

Dere bør teste systemet skikkelig før dere setter det i produksjon (leveres kunden). Dere bestemmer selv hvordan dere skal teste systemet, men dere SKAL beskrive i leveransen hvordan dette ble gjennomført.

6.6 Presentasjon av prosjektet (Dato: TBD)

6.6.1 Presentasjon av prosjektet

Prosjektgruppen SKAL avholde en presentasjon for faglærere hvor man presenterer systemet og prosjektet. Dere vil motta konkrete retningslinjer for gjennomføringen av presentasjonen utover i semesteret.