

# INF 1050

## UKEOPPGAVER 10: ARKITEKTUR OG BRUKERGRENSESNI TT

---

**Innledning:** Arkitektur er noe vi faktisk har snakket om implisitt siden starten av kurset. De overordnede skissene dere skulle lage i ukeoppgave 2 og i PS2000 i obligatorisk oppgave 1 var skisser over (fysiske) arkitekturer. Den ansvarsdrevne metodikken med kant-, kontroll- og entitetsobjekter fra modelleringsforelesningene er logisk arkitektur. Disse oppgavene trekker derfor noen tråder til tidligere oppgaver. Videre er formålet med disse oppgavene å forstå lagdeling og å jobbe med ikke-funksjonelle krav for å komme frem til en arkitekturskisse.

---

### Oppgave 1. Om lagdeling.

- a) Hvor mange fysiske lag er det i et program som Microsoft Word? Kan vi uten å ha mer informasjon om arkitekturen si noe om antall logiske lag?
- b) Kan vi ha flere fysiske lag enn logiske lag i en applikasjon?

### Oppgave 2. Om ikke-funksjonelle krav

I ukeoppgaver 2 om systemutviklingsprosesser og prosjektarbeid, oppgave 1 er det et eksempel med flyselskapet NorFly.

“NorFly er et lite selskap med regionale rutefly I Norden. Selskapet har en flåte med et par dusin mindre passasjerfly og kontorer i alle nordiske land. NorFly er et nytt lavpris flyselskap som har som målsetting å holde samme prisnivå som Norwegian, men samtidig yte bedre service. For å kunne yte riktig service er NorFly opptatt av å samle inn alle flypassasjerenes tilbakemeldinger. De vil derfor lage et system for passasjertilbakemeldinger slik at de kan tilpasse servicen etter passasjerenes behov. Primærfokus skal være på tilbakemeldinger fra en spesifikk flytur som en passasjer har vært med på. Tilbakemeldinger skal kunne sendes inn ved en spesiell terminal I hallen for bagasjehenting ved de forskjellige flyplassene, over internett fra kundens PC og som tekstmelding via mobiltelefon.”

Diskuter hvilke av de følgende ikke-funksjonelle krav som er mest aktuelle for denne typen systemer:

- (Security) Høyt sikkerhetsnivå så aldri noen kan finne ut av hva andre passasjerer har svart.
- (Installability) Enkel eller ingen installasjon ute hos sluttbrukere.
- (Performance) At responsen hos sluttbrukerene er under et sekund.
- (Scalability) At systemet kan håndtere et stort og varierende antall brukere
- (Maintainability) At systemet lar seg vedlikeholde over mange år
- (Usability) At systemet lar seg bruke uten opplæring

Finn flere mulige ikke-funksjonelle krav. Hvordan kan de måles?

### Oppgave 3. Om arkitekturkomponenter

I ukeoppgaver 3 om kravhåndtering er det et eksempel med et bookingsystem som dere har hentet ut kravene for. Dere har både funksjonelle og ikke-funksjonelle krav (hvis ikke må dere definere kravene først).

Prøv først å tegne en fysisk arkitekturskisse med de maskinene som er med i en løsning med et hovedkontor og to reisebyrå. Det vi er ute etter, er tilsvarende figur som det dere skulle lage i oppgave 1 a for ukeoppgave 2.

Det er mange mulige svar på oppgaven spesielt siden vi ikke vet noe om hvordan den sentrale booking løsningen er laget. Her er noen tips for å komme i gang. Gjør en antagelse om at det er en trelags klient-tjener arkitektur i den sentrale løsningen. Men hva om det er en to-lags arkitektur sentralt? Hvordan ser det ut hvis dere velger en browser-basert klient i det nye systemet? Hvordan ser det ut om vi i tillegg skal ha en håndholdt løsning hvor en mobiltelefon er klienten?

Hvis de lokale reisebyråene skal kunne fortsette å jobbe selv om de mister kommunikasjonen med det sentrale bookingsystemet trenger de en eller flere moduler I tillegg. Hvilke? Prøv å tegne de logiske komponentene dere trenger for å støtte bookingsystemet hos de lokale reisebyråene med følgende funksjonalitet:

“Hvis forbindelsen til hovedkontoret faller ut vil vi fortsette å selge billetter under den forutsettelsen at det er vanligvis ok å overbooke billetter og at vi kan rydde opp I overbooking når kommunikasjonen kommer tilbake. (Vi er så grådige at vi aldri ønsker å gå glipp av et salg). Hvor komplekst dette er komme litt an på om det er nummererte billetter eller bare antall plasser totalt“

### Oppgave 4. Om brukervennlighet og design

Anta at du begynner å føle den hete pusten av Statens Lånekasse for Utdanning i nakken og er nødt til å se deg om etter en jobb. Det står så dårlig til at du vurderer å flytte ut. Du må

bruke websidene til NAV i Norge og Arbetsförmedlingen i Sverige. Prøv å finne frem jobber under programmerer/systemutvikler i Norge og programmerare/systemutvecklare i Sverige. Prøv både søkefunksjonene og navigering frem til riktig underkategori. Tenk deg at du ikke bare er interessert i programmering men også en jobb som selger innen IT.

Hvordan har de to løst funksjonalitet for å lagre et søk? Vurder brukergrensenettet opp mot de syv tommelfingerreglene for godt brukergrensesnitt. Sjekk spesielt ut navigering, hvordan man kommer seg fem og tilbake i programmet. Til slutt gjør samme oppgaven på finn.no.

Linker til tjenestene er <https://www.nav.no/sbl/stillingssok/enkelt.do> og <http://platsbanken.arbetsformedlingen.se>

\*\*\*