Systemutvikling Eksamen

# **Emnebeskrivelse**

## **KUNNSKAPER**

* Kan gjøre rede for prinsippene for iterativ og inkrementell systemutvikling
* Har satt seg inn i hvordan et prosjekt styres, dokumenteres og gjennomføres
* Har innsikt i hvordan Java og relasjonsdatabaser kan integreres for a utvikle et enhetlig informasjonssystem
* Kan gjøre rede for ulike typer tester og teststrategier

**FERDIGHETER**

* Bruke en iterativ og inkrementell systemutviklingsmodell til å utvikle et informasjonssystem
* Designe implementeringsuavhengige objektorienterte informasjonsmodeller basert på Unified Modeling Language (UML)
* Bruke informasjonsmodeller til å dokumentere informasjonssystemets visjon, krav, arkitektur og design
* Sette opp interessent- og risikoanalyse samt å handtere kostnads- og ressursstyringen knyttet til aktuelt prosjekt

**GENERELL KOMPETANSE**

* Kan jobbe effektivt og målrettet i et team mot et felles mål
* Kan dele og formidle informasjon og kunnskap med teamdeltakere og oppdragsgivere
* Har respekt for etiske regler knytt til utviklingen av informasjonssystem
* Har utviklet gode skrive- og framstillingsferdigheter både på norsk og engelsk
* Har praktisk erfaring med a anvende programmerings og databasekompetanse i samspill med en systemutviklingsmodell for a lose virkelighetsnære problemstillinger

# **Testing**

Eksamen 2018

1. **Hvorfor er testing viktig?**

Testing er viktig for avdekking av feil i koden. I et systemutviklingsprosjekt får oppdaging av feil i koden større økonomisk konsekvenser jo lengre ut i prosessen man er. Testing er også med på validering og verifikasjon der vi vet om programvaren tilfredsstiller brukerkravene eller ikke og om den oppfyller forretningskravene, og om den er utviklet i henhold til de riktige spesifikasjonene og metodene.

1. **JUnit er et rammeverk for automatisering av enhetstesting. Er dette å foretrekke foran manuelle tester på dette stadiet i utviklingsprosessen? Begrunn svaret ditt.**

Ja, fordi med automatisert testing vil testkoden i seg selv være veldig detaljert spesifikasjon av et testscript med test data og forventende resultater. Her vil testkoden si ifra om testen feiler og den vil minske kostnaden ved kodeendring inkl. refaktorisering. Junit gjør det også mer realistisk at testing gjennomføres ved hver endring. Noe vi ikke får med manuelle tester.

1. **Når kan vi si oss fornøyd/ ferdig med enhetstesting av en applikasjon?**

Vi kan avslutte enhetstesting når det ikke er noen kritiske feil i koden. Testscriptene må kjøre uten feil. Man vil også at feil oppdaget pr dag/ pr time er lav.Som regel stopper man enhetstesting når det ikke er mer tid, folk eller penger.

Eksamen 2017

**Oppgave 2 Testing (20%)**

1. **Testing beskrives på enkelt nivå som: For en gitt input forventer vi en gitt output. Vi skal her se på utviklingstesting – nærmere bestemt enhetstesting. Gjør rede for hva er enhetstesting og drøft fordeler og ulemper med denne formen for testing.**

Enhetstesting er testing hvor enkeltdeler av programmet blir isolert og tester at alle delene er feilfrie. For hver gang en modul blir endret på blir testen kjørt på nytt for å unngå regresjon. Altså sjekke konsekvensene av endringene vi gjorde ikke har introdusert nye feil. Fordelen med dette er at vi får oppdaget feil tidlig i kode prosessen, og vi forsikrer oss om at endringer ikke innfører nye feil. Ulempen med enhetstesting er at man ikke får testet integrasjon mellom modulene. Man får heller ikke bevist om koden er 100% feilfri siden den kun viser tilstedeværelse av feil og ikke fravær av feil. Det stiller også store krav til utviklingsprosess for å sikre effektivitet må testene følges opp til enhver tid. Det er også urealistisk å teste alle mulige kjørbare veier i programkoden.

1. **Er det ønskelig å automatisere enhetstesting for eksempel ved å bruke rammeverket JUnit? Begrunn svaret.**

Det er ønskelig. Ved automatisert enhetstesting vil testkoden i seg selv være veldig detaljert spesifikasjon av et testscript med test data og forventende resultater. Her vil testkoden si ifra om testen feiler og den vil minske kostnaden ved kodeendring inkl. refaktorisering. Junit gjør det også, er realistisk at testing gjennomføres ved hver endring.

1. **Du er ferdig med koding på et produkt og du har gjennomført enhetstesting med 100% kodedekning og ingen av testene feiler. Kan du da anse deg som ferdig med all testing i prosjektet og klar til å overlevere produktet til kunden? Begrunn svaret.**

Som nevnt i oppgave a) så får man ikke bevist at koden er 100% feilfri siden enhetstesting kun viser tilstedeværelse av feil og ikke fravær av feil.

Eksamen 2016

**Oppgave 2 Testing (20%)**

Murphy’s lov er et godt utgangspunkt for de som tester programvare: «Det er alltid feil». Testing foregår i ulike nivåer/ deler av utviklingsprosessen og i denne oppgave skal vi se på enhetstesting.

1. **Hva er enhetstesting og hva er motivasjonen for å gjennomføre enhetstesting?**

Enhetstesting isolerer enkeltdeler av programmet og viser at alle delene er feilfrie. Testen kjøres på nytt for hver gang en modul endres for å unngå regresjon. Dette er for å sjekke konsekvensene av endringene og at vi ikke har introdusert nye feil. Motivasjonen bak en enhetstesting er å finne feil så tidlig som mulig. Jo lenger ut i utviklingen av programvaren vi finner feilen jo større økonomisk konsekvens har det. Det er med på å vise at systemet funker, noe som er viktig siden brukere har blitt mer kresene etter årene.

1. **Hvordan gjennomføre enhetstesting? Strategi?**

Strategien for enhetstesting er testing av isolerte enkeltdeler av programmet. Disse testes for å oppdage evt. bugs i koden. Hver gang en modul endres må testen kjøres på nytt for å unngå regresjon. Dette er for å sjekke konsekvensen av endringene og at vi ikke får introdusert nye feil i koden.

1. **Som et mål på hvor godt vi har enhetstestet måles ofte kodedekningsgrad (codecoverage). Dersom vi har dekningsgrad 100%, kan vi da være trygge på at koden er feilfri? Begrunn svaret.**

Nei for med enhetstesting kan man kun vise tilstedeværelse av feil, ikke fravær av feil.

Kont eksamen 2016

**Oppgave 2 Testing (15%)**

1. **Hva er motivasjonen for å gjennomføre testing av programvare?**

Motivasjonene bak testing er å avdekke feil i programkoden så tidlig som mulig. Den er med på å sjekke at kravspesifikasjonen til koden er oppfylt. Med testing får man en testkode som fungerer som dokumentasjon og demonstrasjon. Jo tidligere feil blir avdekt jo mindre økonomiske konsekvenser får det.

1. **Vi deler testene inn i to ulike områder – funksjonelle- og ikke funksjonelle tester. Hva er forskjellen på disse to? Gi eksempler på tester innenfor de to områdene.**
2. **Hvor lenge skal man teste?**

Man skal teste så lenge ressursene rekker, men i mange tilfeller har man ikke mer folk, tid eller penger. Da avsluttes testingen når ingen kritiske feil lenger er tilstede i koden. Når alle testscript kjører uten feil og antall feil funnet pr dag/ pr time er lav.

# **Prinsipper**

**Eksamen 2018**

**1. Kan du forklare flere gode grunner til å bruke wireframing i et systemutviklingsprosjekt?**

**2. I hvilke faser av et prosjekt mener du det er nyttigst å bruke wireframing og hvorfor?**

**3. Hva legger du i begrepene iterativ systemutvikling og evolusjonær systemutvikling?**

**4. Hvordan passer disse prinsippene sammen med vannfallsmodellen?**

**5. Hvordan passer de eventuelt sammen med de fire sentrale verdiene i "The Agile Manifesto"?**

**I systemutviklingsprosjekter:**

**1. Hvorfor dokumenterer vi?**

**2. Hvordan bør det dokumenteres?**

**3. Hva bør dokumenteres?**

**Eksamen 2017**

**Oppgave 3 Prosjektstyring / dokumentasjon (20%)**

**Misforståelser mellom forskjellige aktører/interessenter i prosjekter er vanlig, for eksempel mellom oppdragsgiver, domeneeksperter og utviklere. Dette kan føre til at man utvikler feil produkt, eller at man ikke når tidsfrister. Hvordan kan man forsøke å unngå slike problemer? Vurder dette både i forhold til muntlig kommunikasjon og dokumentasjon**

**Eksamen 2016**

**Oppgave 3 Prinsipper (20%)**

1. **Gjøre rede for prinsippene for iterativ og inkrementell systemutvikling**

**Eksamen 2016 kont**

**Oppgave 3 Prinsipper (25%)**

**a) Hva er Unified Process (UP)?**

**b) Hvilke faser og disipliner (hovedaktiviteter) består UP av? Forklar hva som legges i de ulike fasene og sammenheng mellom faser og disipliner.**

**c) UP er Use Case-drevet, hva innebærer det?**