

Programmering i rapporten

Når du skriver om programmering er der tre forskellige faser du skal igennem.

I den første fase bør du beskrive og forklare de meste centrale og afgørende programmeringsteknikker og termer, som du anvender.

I den anden fase bør du beskrive produktet og den bagvedliggende kode. Det er vigtigt at læseren både forstår hvordan produktet ser ud og hvordan koden virker. Derfor er det nødvendigt både at vise forklarende skærbilleder af produktet og formidle koden gennem velvalgte diagrammer og direkte kodesekvenser.

I den sidste faser diskuterer du den anvendte metode. Hvis du har anvendt det objektorienterede paradigme, kan du for eksempel diskutere kodetekniske fordele og performance sammenlignet med alternativet.

Fase 1 : Terminologi og metode redegørelse

Indenfor programmering er der både direkte kode-tekniske termer og mere design og metodemæssige. Indenfor objektorienteret programmering er eksempler på tekniske termer "klasser", "objekter", "nedarvning" og "konstruktører". Eksempler på mere metodemæssige termer/metoder kunne være "single responsibility", "clean code" og "kode genbrug".

Det er din opgave at udvælge hvilke termer og metoder du vil forklare. Når du udvælger er det vigtigt du vælger noget du anvender i produktet, således der er en "rød tråd" i din rapport.

Samtidig skal de udvalgte termer og metoder være på et tilstrækkeligt fagligt højt niveau.

Eksempelvis er forklaring og beskrivelse af for-loops og variabler ikke på et højt nok fagligt niveau.

Fase 2: Design og produktbeskrivelse

Når du skriver produktbeskrivelsen er det vigtigt du skriver den på en måde så den kan læses og forstås af en programmør (på dit eget "forventelige" niveau), der ikke kender til dit produkt i forvejen.

Du bør derfor starte med det mindst tekniske beskrivelse af produktet, en slags "user journey" eller "runtime beskrivelse", der består af skærbilleder med dertilhørende forklaring. Denne

beskrivelse er muligvis allerede varetaget af kom-it-faget!

Derefter kan du gå i gang med den kode-tekniske forklaring. Du bør starte på et overordnet niveau og derefter bevæge dig imod det mere detaljeorienterede.

Man kan beskrive sin kode overordnet gennem diagrammer, f.eks. “use-case diagrammer”, “aktivitetsdiagrammer”, “flowcharts” og “klassediagrammer”

Det mest detaljeorienterede niveau i produktbeskrivelsen er at vise og beskrive direkte dele af koden. Det er klart, at du ikke skal forklare alt koden detaljeorienteret, men kun viser udvalgte “kode-sekvenser”. Du foretager selv udvælgelsen af “kode-sekvenser”, men det er vigtigt du i din udvælgelse lægger vægt på de vigtigste og mest avancerede dele af koden, der er på et tilstrækkeligt højt fagligt niveau.

Nedenfor følger et forslag til en opdeling af produktbeskrivelsen:

1. Start med en “runtime beskrivelse” også kaldet “user journey” (muligvis varetaget af kom-it faget)
2. Vis og forklar overordnede diagrammer, f.eks. “klassediagram” og/eller “flow chart”
3. Vis og forklar udvalgte en eller flere “kode-sekvenser”

Fase 3 : Metodediskussion

Det højeste taksonomiske niveau af enhver opgave er en diskussion eller vurdering. Det siger sig selv, at man kun kan lave en meningsfuld vurdering af en metode, hvis man både forstår og har afprøvet en metode. Det er derfor vigtigt at dit produkt er objektorienteret, hvis du skal kunne lave en meningsfuld vurdering af anvendelsen af det objektorienterede programmeringsparadigme til udvikling af produktet.