

Tentamen Software Engineering, 26 augustus 2003

Duur: 3 uur (08.30 – 11.30)

- Geef bij je uitwerking op elk vel duidelijk je naam, studentnummer, het vraagnummer en het tentamen aan.
-

Vraag 1. Software ontwikkel processen (1.5 pt.)

Drie belangrijke deliverables¹ in een software ontwikkelproces zijn: Een specificatie, b.v. in de vorm van use-cases; een ontwerp; een implementatie (omvattend zowel code als het run-time systeem).

In een modern ontwikkelproces (zoals het Unified Process) is de implementatie niet alleen een deliverable van de laatste fase, maar wordt de implementatie incrementeel uitgebreid. De implementatie kan dus bijvoorbeeld ook in de vorm van een prototype een deliverable zijn van eerdere fases.

Geef minimaal zes manieren waarop elk van die drie soorten deliverables kan dienen als communicatie tussen verschillende belanghebbenden en tussen verschillende activiteiten van het ontwikkelproces. Benoem daarbij de belanghebbenden en/of de activiteiten. Geef per manier telkens een korte(!) uitleg.

Vraag 2. Requirements (1 pt.)

Wat is het verschil tussen user requirements en system requirements. Welke kan je wel, welke kan je niet makkelijk valideren en waarom?

Vraag 3. Requirements engineering (1 pt.)

Wat zijn de doelen van requirement management? Welke middelen kan je gebruiken om die doelen te halen?

Vraag 4. Systeem modellen (1 pt.)

Noem 4 soorten systeem modellen die door Sommerville worden beschreven. Geef van elk van deze modellen kort weer wat het beschrijft.

Vraag 5. Architectural Design en Software Hergebruik (1 pt.)

Leg uit waarom een goed architectuur ontwerp belangrijk is voor grootschalig software hergebruik. Geef voor 2 van de belangrijke systeem structuren(!) zoals die door Sommerville beschreven een voorbeeld van een product familie waarin die structuur gebruikt kan worden en op welke wijze de architectuur variatie binnen de familie toestaat.

Vraag 6. Object-geïntendeerd Ontwerp en Software Hergebruik (1 pt.)

Beschrijf de structuur van een design pattern, en leg uit hoe je design patterns kunt gebruiken in het ontwerp proces, met name: wanneer kan je een pattern gebruiken en wat moet je dan doen?

Vraag 7. Verificatie en validatie (1 pt.)

Leg duidelijk uit wat het verschil is tussen verificatie en validatie. Leg ook uit waarom validatie eigenlijk belangrijker is dan verificatie. Welke relaties zijn er tussen de test activiteit en zowel Verificatie als Validatie?

¹ Deliverable is: "iets wat opgeleverd moet worden"

Vraag 8. Kwaliteit management (1.5 pt.)

Het klassieke beeld van kwaliteit is dat een product voldoet aan de verwachtingen van de klant. Kwaliteitsmanagement wordt in hoofdstuk 24 van Sommerville beschreven. In andere hoofdstukken beschrijft Sommerville hoe met name validatie en verificatie (V&V) gebruikt worden om producten aan de wensen van de klant te laten voldoen. In deze vraag zullen we nagaan wat de relatie is tussen quality management en V&V.

- a) In quality assurance worden standaarden aangelegd voor documenten (document structuren) en processen.
 1. Leg uit hoe goede document standaarden van nut kunnen zijn voor de kwaliteit van de inhoud van die documenten, en hoe die standaarden van nut kunnen zijn voor de V&V. Tip: Gebruik concrete voorbeelden, zoals standaarden voor een requirements of een design document.
 2. Leg uit wat proces standaarden te maken hebben met (goede!) V&V en daarom de goede eindkwaliteit van het product
- b) In quality control worden diverse procedures en metingen gebruikt. Welke kwaliteits controle procedure kan goede V&V bevorderen? Tip: Bedenk welke procedures Sommerville noemt, en welke daarvan direkt te maken heeft met de bovengenoemde standaarden.