Programmeervaardigheden

Aan te tonen vaardigheden: • Je kunt een applicatie programmeren die meerdere functionaliteiten ondersteunt. • Je kunt een database ontwerpen voor een eigen applicatie. • Je kunt een databasekoppeling realiseren voor een eigen applicatie.

Structurering van softwaresystemen

Aan te tonen vaardigheden: • Je kunt een klassendiagram opzetten als concept voor de implementatie van een probleem. • Je kunt een applicatie en klassendiagram realiseren die volledig overeenkomen. • Je kunt een applicatie programmeren waarin geen aantoonbare duplicatie in voorkomt. • Je kunt een applicatie programmeren met alleen maar klassen met een enkele verantwoordelijkheid. • Je kunt een applicatie programmeren die een correcte encapsulatie vertoont.

Generalisatie en abstractie

Aan te tonen vaardigheden: • Je kunt een klassendiagram opzetten waarin een oplossing voor variaties is verwerkt en onderbouwd. • Je kunt een databaseontwerp opzetten waarin een oplossing voor variaties is verwerkt en onderbouwd. • Je kunt een applicatie en klassendiagram met inheritance, abstracte klassen en interfaces volledig overeen laten komen. • Je kunt op een correcte manier gespecialiseerde functionaliteit verwerken in een afgeleide klasse.

Requirements en analyse

Aan te tonen vaardigheden: • Je kunt toepasselijke belanghebbenden en wensen voor een te realiseren systeem beschrijven. • Je kunt requirements afleiden uit een gesprek of document met daarin de wensen van een belanghebbende. • Je kunt de onderlinge prioriteit van requirements op basis van verkregen informatie vaststellen en onderbouwen. • Je kunt het verschil tussen functionele requirements, kwaliteitseisen en beperkingen herkennen en beschrijven. • Je kunt het gebruik van een systeem illustreren met behulp van scenario's of user stories. • Je kunt gedetailleerde systeeminteractie met behulp van use cases modelleren. • Je kunt de rol van user interface schetsen binnen de analysefase verklaren. • Je kunt testcases formuleren waarmee vastgestelde requirements eenduidig gevalideerd en geïnvalideerd kunnen worden.

Ontwerpen

Aan te tonen vaardigheden: • Je kunt op basis van zelf opgestelde use cases een klassendiagram modelleren waarin gemaakte keuzes en afgevallen alternatieve keuzes onderbouwd zijn. • Je kunt verklaren waarom abstractie in een softwaresysteem onderhoud vergemakkelijkt en hoe dit tot uiting komt in een geïmplementeerd systeem. • Je kunt de benodigde infrastructuur ten behoeve van het gebruik van het systeem schetsen in een netwerktekening.

Integratie van softwaresystemen

Aan te tonen vaardigheden: • Je kunt een extern softwarecomponent integreren in een zelf opgezette applicatie. • Je kunt in een zelf opgezet klassendiagram samenhangende klassen herkennen, deze bundelen in componenten, en deze structuur verwerken in een applicatie. • Je kunt databasecode ontkoppelen van de rest van het systeem. • Je kunt unit testen opstellen om de logica van een zelf opgezet systeem te testen.

Architectuur en onderhoud

Aan te tonen vaardigheden: • Je kunt met hedendaagse technieken webpagina’s semantisch opzetten, vormgeven en interactief maken. • Je kunt met een versiebeheersysteem zelf een project opzetten waar gestructureerd code mee beheerd en gedeeld wordt. • Je kunt een webapplicatie opzetten conform het MVC-patroon. • Je kunt een applicatie opzetten waarbij meerdere soorten grafische interfaces voor beschikbaar zijn gesteld. • Je kunt een lokaal opgezette applicatie met database deployen en beschikbaar stellen in een gehoste omgeving.

SQL

Je kunt stored procedures gebruiken voor het ophalen van data.

Je kunt triggers maken voor het afvangen van database constraints.

Je kunt toepassingen tonen voor zowel de INNER JOIN als de OUTER JOIN.

Je kunt groepsfuncties op een zinnige manier gebruiken in combinatie met een GROUP BY-clausule.

Je hebt voor uiteenlopende problemen een onderbouwde afweging gegeven voor de gebruikte oplossingsrichting.