Leeruitkomsten SE, clusters OOSDD, WEBDEV, QSD

**Inleiding**Deze leeruitkomsten zijn gebaseerd op HBO-i. Uitgangspunt is geweest om processen gestructureerd aan te kunnen bieden en te kunnen relateren aan HBO-i. Vooraf is een analyse uitgevoerd waarvan de gegevens zijn vastgelegd. Elk record in deze gegevens heeft een id. De onderdelen van de leeruitkomsten in dit document verwijzen naar deze brongegevens. Ook is gebruik gemaakt van een inventarisatie in het SE team waarin belangrijke standaarden zijn geïnventariseerd.

**Status**Dit document heeft een eerste status en zal in de komende tijd verder worden aangevuld en de verwachting is dat dit zelfs tijdens de ontwikkeling van de semesters zal doorgaan.

**Leeswijzer**De leeruitkomsten zijn gegroepeerd per HBO-i beroepsactviteit. Voor een sortering op proces en processtap kan gebruik worden gemaakt van de brongegevens. Per semester is een context geformuleerd overeenkomstig de contextbeschrijvingen per niveau in HBO-i.  
Elk onderdeel van een leeruitkomst heeft een Tuning formulering:  
  
voorbeeld: adviseert over selectie en aanschaf van software componenten op basis van functionaliteit en kosten. [14] [11]  
Tuning: <werkwoord> <onderwerp> <standaarden> <verwijzing naar brongegevens>

Zoals het voorbeeld laat zien zijn de standaarden niet concreet geformuleerd. Tijdens de ontwikkeling van het semester kan dit aangepast worden door óf de formulering te vervangen óf de expliciete standaarden toe te voegen, bijvoorbeeld tussen haken toe te voegen.

Team SE

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **HBO-i niveau 2**  Context: Software systeem bestaand uit subsystemen | | | | **HBO-i niveau 3**  Context: Schaalbaar software systeem en aansluitend bij bestaande systemen | |
|  | **OOSDD** | | **WEBDEV** | | **QSD** | |
|  | Context: Desktop applicatie met database connectie. Eén stakeholder betrokken. | | Context: Webapplicatie met database en API. Meerdere stakeholders betrokken. | | Context: Samenwerkende deelsystemen aansluitend op bestaande systemen en meerdere betrokken stakeholders met verschillende perspectieven. | |
| **Adviseren** | De student  - adviseert over de kwaliteit van de architectuur.[17]  - adviseert over selectie en aanschaf van software componenten op basis van functionaliteit en kosten. [14] [11]  X GEGEVEN | **PUBLICATIETEKST**  Je adviseert over de kwaliteit van de architectuur.  Je kan adviseren over de selectie en aanschaf van softwarecomponenten op basis van functionaliteit en kosten. | De student  - adviseert over de kwaliteit van de architectuur.[17]  - adviseert over selectie en aanschaf van software componenten op basis van functionaliteit en kosten. [14][11]  X GEGEVEN | **PUBLICATIETEKST**  Je adviseert over de kwaliteit van de architectuur.  Je kan adviseren over de selectie en aanschaf van softwarecomponenten op basis van functionaliteit en kosten | De student  - adviseert over de kwaliteit van de architectuur[[1]](#footnote-1) rekening houdend met een aantal kwaliteitsaspecten zoals kosten, beschikbaarheid, performance, security en schaalbaarheid (ISO25010). [17]  - adviseert over de toepassing van technologie, die nieuw is en rekening houdend met de duurzaamheid, security en privacy.[14][11]  - adviseert over de inrichting van het ontwikkelproces (test- en releaseproces).[6] | **PUBLICATIETEKST**  Je adviseert over de kwaliteit van de architectuur, rekening houdend met kwaliteitsaspecten zoals kosten, beschikbaarheid, performance, security en schaalbaarheid.  Daarnaast adviseer je over de toepassing van nieuwe technologieën, met oog voor duurzaamheid, security, privacy, en de inrichting van het ontwikkel-, test- en releaseproces |
| **Analyseren** | De student  - analyseert voor selectie en aanschaf voor een softwaresysteem rekening houdend met functionaliteit, veiligheid, ontwerp en interfaces. [13]  - definieert het probleemdomein gebruikmakend van domeinmodellen en volgens de richtlijnen ontwerpen SE HBO-ICT.[7]  - verzamelt en analyseert de requirements voor het domein rekening houdend met kwaliteitsaspecten waaronder duurzaamheid en security . [8][9]  - specificeert requirements voor een softwaresysteem. [10] | **PUBLICATIETEKST**  Je analyseert, specificeert en selecteert een softwaresysteem door het probleemdomein te definiëren met behulp van domeinmodellen, hierbij rekening houdend met functionaliteit, veiligheid, ontwerp, interfaces, duurzaamheid en security, conform de SE-richtlijnen voor HBO-ICT | De student  - analyseert voor selectie en aanschaf voor een softwaresysteem rekening houdend met functionaliteit, veiligheid, ontwerp en interfaces. [13]  - definieert het probleemdomein gebruikmakend van domeinmodellen.[7]  - verzamelt en analyseert de requirements voor het domein rekening houdend met kwaliteitsaspecten waaronder duurzaamheid en security (ASVS). [8][9]  - specificeert requirements voor een softwaresysteem. [10] | **PUBLICATIETEKST**  Je analyseert en selecteert een softwaresysteem rekening houdend met functionaliteit, veiligheid, ontwerp en interfaces, en definieert het probleemdomein met behulp van domeinmodellen.  Je verzamelt en analyseert de requirements, inclusief duurzaamheid en security (ASVS), specificeert de requirements, en ontwikkelt en voert functionele testen en unittesten uit. | De student  - analyseert voor selectie en aanschaf voor een softwaresysteem rekening houdend met functionaliteit, veiligheid, ontwerp en interfaces. [13]  - definieert het probleemdomein gebruikmakend van domeinmodellen.[7]  - verzamelt en analyseert de requirements voor het domein rekening houdend met kwaliteitsaspecten waaronder duurzaamheid, security (ASVS), privacy en toegankelijkheid. [8][9]  - specificeert requirements voor een softwaresysteem. [10] | **PUBLICATIETEKST**  Je analyseert en selecteert een softwaresysteem rekening houdend met functionaliteit, veiligheid, ontwerp, interfaces en kwaliteitsaspecten zoals duurzaamheid, security, privacy en toegankelijkheid.  Je definieert het probleemdomein met behulp van domeinmodellen en specificeert de requirements voor het softwaresysteem. |
|  | **HBO-i niveau 2**  Context: Software systeem bestaand uit subsystemen | | | | **HBO-i niveau 3**  Context: Schaalbaar software systeem en aansluitend bij bestaande systemen | |
|  | **OOSDD** | | **WEBDEV** | | **QSD** | |
|  | Context: Desktop applicatie met database connectie. Eén stakeholder betrokken. | | Context: Webapplicatie met database en API. Meerdere stakeholders betrokken. | | Context: Bestaande systemen en meerdere stakeholders. | |
| **Ontwerpen** | De student  - ontwerpt een softwaresysteem volgens de richtlijnen ontwerpen SE HBO-ICT.[15]  - stelt de kwaliteit van het ontwerp vast volgens de kwaliteitseisen opgenomen in richtlijnen ontwerpen SE HBO-ICT.[16]  - gebruikt een gegeven teststrategie.[1]  - maakt een testontwerp.[2] | **PUBLICATIETEKST**  Je ontwerpt een softwaresysteem volgens de SE HBO-ICT ontwerprichtlijnen, stelt de kwaliteit van het ontwerp vast op basis van de gestelde kwaliteitseisen en maakt een testontwerp gebruikmakend van een gegeven teststrategie. | De student  - ontwerpt een softwaresysteem.[15]  - stelt de kwaliteit van het ontwerp vast.[16]  - gebruikt een gegeven teststrategie.[1]  - maakt een testontwerp.[2] | **PUBLICATIETEKST**  Je ontwerpt een softwaresysteem en stelt de kwaliteit van het ontwerp vast.  Je maakt een testontwerp gebruikmakend van een gegeven teststrategie | De student  - ontwerpt een softwaresysteem.[15]  - stelt de kwaliteit van het ontwerp vast.[16]  - maakt een teststrategie met systeem- en compliancytesten.[1]  - maakt een testontwerp.[2] | **PUBLICATIETEKST**  Je ontwerpt een softwaresysteem en stelt de kwaliteit van het ontwerp vast.  Je ontwikkelt een teststrategie met systeem- en compliancytesten en maakt een testontwerp |
|  | **HBO-i niveau 2**  Context: Software systeem bestaand uit subsystemen | | | | **HBO-i niveau 3**  Context: Schaalbaar software systeem en aansluitend bij bestaande systemen | |
|  | **OOSDD** | | **WEBDEV** | | **QSD** | |
|  | Context: Desktop applicatie met database connectie. Eén stakeholder betrokken. | | Context: Webapplicatie met database en API. Meerdere stakeholders betrokken. | | Context: Bestaande systemen en meerdere stakeholders. | |
| **Realiseren** | De student  - richt de ontwikkelomgeving in.[18]  - gebruikt een beheersysteem met SCRUM.[24]  - stelt het softwaresysteem beschikbaar met GIT.[25]  - bouwt het softwaresysteem rekening houdend met integriteit, veiligheid en performance (FURPS, SOLID, DRY). [19,20]  X  - ontwikkelt, voert uit testen waaronder functionele testen en unittesten.[3,4,5]  - oplossen van fouten in software op basis van logs en het doorlopen van de IDE. [28] | **PUBLICATIETEKST**  Je richt zelfstandig een ontwikkelomgeving in en maakt hierbij gebruik van een beheersysteem gebaseerd op SCRUM.  Je stelt het softwaresysteem beschikbaar via GIT en bouwt het systeem waarbij je rekening houdt met belangrijke ontwerpprincipes zoals integriteit, veiligheid en performance (volgens de FURPS, SOLID en DRY-principes).  Daarnaast ontwikkel je testen voor het systeem, waaronder functionele testen en unittesten, en voer je deze nauwkeurig uit om de correcte werking van het systeem te waarborgen | De student  - richt de ontwikkelomgeving in.[18]  - gebruikt een beheersysteem. [24]  - stelt het softwaresysteem beschikbaar.[25]  - bouwt het softwaresysteem rekening houdend met integriteit, veiligheid en performance. [19,20]  - beredeneert bouwkeuzes [21]  - ontwikkelt, voert uit en evalueert testen waaronder unittesten en UI-testen.[3,4,5]  - oplossen van fouten in software op basis van logs en het doorlopen van de IDE. [28] | **PUBLICATIETEKST**  Je richt de ontwikkelomgeving in, gebruikt een beheersysteem en stelt het softwaresysteem beschikbaar, waarbij je rekening houdt met integriteit, veiligheid en performance.  Je bouwt het systeem, beredeneert de bouwkeuzes en ontwikkelt, voert uit en evalueert testen, waaronder unittesten en UI-testen | De student  - richt de ontwikkelomgeving in.[18]  - gebruikt een beheersysteem. [24]  - stelt het softwaresysteem beschikbaar.[25]  - bouwt en refactort het softwaresysteem rekeninghoudend met performance (ordecomplexiteit en prestatie efficiëntie) mogelijk met frameworks. [19,20]  - beredeneert bouwkeuzes [21]  - ontwikkelt, voert uit en evalueert testen waaronder unittesten, UI-testen, regressie-, integratie en systeemtesten.[3,4,5]  - oplossen van fouten in software op basis van logs en het doorlopen van de IDE. [28] | **PUBLICATIETEKST**  Je richt de ontwikkelomgeving in, gebruikt een beheersysteem, stelt het softwaresysteem beschikbaar en bouwt of refactort het systeem met behulp van frameworks, waarbij je bouwkeuzes beredeneert.  Je ontwikkelt, voert uit en evalueert verschillende testen, waaronder unittesten, UI-testen, regressie-, integratie- en systeemtesten |
| **Manage en**  **Control** | De student  - managet en beheert het ontwikkelproces. [22, 23]  X  X  - beheert requirements.[12] | **PUBLICATIETEKST**  Je managet en beheert het ontwikkelproces en je beheert de bijbehorende requirements. | De student  - managet en beheert het ontwikkelproces. [22, 23]  X  X  beheert requirements.[12] | **PUBLICATIETEKST**  Je managet en beheert het ontwikkelproces en je beheert de bijbehorende requirements | De student  - managet en beheert het ontwikkelproces. [22, 23]  - evalueert de ontwikkelstraat. [26]  - voert uit configuratie-, change- en releasemanagement[27]  - beheert requirements.[12] | **PUBLICATIETEKST**  Je managet en beheert het ontwikkelproces en evalueert de ontwikkelstraat.  Je voert configuratie-, change- en releasemanagement uit en beheert de requirements. |

1. Architectuur betreft zowel de architecturele patterns als de software patterns [↑](#footnote-ref-1)