

Blokboek Semester 2

Profiel ICT & Software

Versie: januari 2016

Auteurs:

Inge van Engeland - Danny Bloks

Voorwoord

Dit blokboek is in eerste instantie geschreven voor studenten die gekozen hebben voor het profiel Software Engineering van Fontys Hogeschool ICT. Maar natuurlijk is het ook geschikt voor geïnteresseerden, die willen weten hoe ons onderwijs in semester 2 is opgebouwd.

Je hebt als student FHICT het eerste semester succesvol doorlopen en een bewuste keuze gemaakt om software engineer te worden.

Om dit voor elkaar te krijgen, hebben wij een studeerbaar, interessant en uitdagend programma voor jullie ontwikkeld, met als centraal punt een realistische proftaak. Dat kun je in dit blokboek zien, maar de details zijn te vinden in de studentenhandleidingen van de vakken.

Ik weet dat jullie door hard te werken, initiatief te tonen, kritisch te zijn en met een onderzoekende houding dit semester succesvol zullen afronden.

Veel succes en plezier tijdens deze periode!

Inge van Engeland & Danny Bloks

Blokeigenaar Semester 2 Software Engineering

Inhoudsopgave:

1. Inleiding	4
1.1. Semester 2	4
1.2. Verbeteringen	4
2. Inhoud	6
2.1. Deelvakken van dit semester.....	6
2.1.1. Professionele Ontwikkeling (PPO).....	18
2.1.2. Assessment Professionele Ontwikkeling Propedeuse	18
2.2. Het versnelde programma	22
2.2.1. Professionele Ontwikkeling (PPOv)	23
3. Organisatie	24
3.1. Uitvoering.....	24
3.2. Werkplekomschrijving	24
3.3. Leermiddelen	24
3.4. Studiewijzer per onderwijsactiviteit S2.....	24

1. Inleiding

1.1. Semester 2

Semester 2 is het eerste semester dat in het teken staat van het basisprofiel “ICT & Software Engineering”. In dit semester krijg je daarom vooral onderwijs dat een basis legt voor de kennis en vaardigheden die je nodig hebt als software engineer.

Bij SE komen de eerste beginselen van software-ontwerp en een aantal belangrijke objectgeoriënteerde programmeerconcepten aan de orde. Bij het deelvak databases gaat datamodelleren, database-ontwerp en het gebruik van SQL naar een gevorderd niveau en leer je de vele mogelijkheden en toepassingen kennen van de databaseprogrammeertaal PL/SQL.

Het ontwerpen en realiseren van informatiesystemen, tegenwoordig meestal webapplicaties, betekent dat je te maken krijgt met integratie van programmacode met grote databases en een complexe ICT-infrastructuur. Deze 3 pijlers krijg je als deelvak in dit semester en ze komen geïntegreerd terug in een realistische opdracht, de proftaak.

Een belangrijke stap die je dit semester ook gaat maken is op het gebied van professionalisering. Wat is de juiste beroepshouding van een ICT'er? Hij/zij moet kunnen samenwerken, initiatief tonen, onderzoek kunnen doen, zelfstandig problemen oplossen, communiceren op een professioneel niveau, zowel mondeling als schriftelijk. Deze competenties kun je dit semester aanleren en/of aanscherpen in de verschillende vakken. Verder moet je jouw ontwikkeling van deze vaardigheden aantonen en bewijzen via het eindassessment PPS2.

In dit blokboek worden alle deelvakken waar je mee te maken hebt toegelicht. In de studentenhandleidingen kun je de details van deze deelvakken terugvinden. Zoals bijvoorbeeld een gedetailleerde weekplanning en meer informatie over het toetsen van een deelvak. Deze studentenhandleidingen zijn te vinden op het intranet bij het betreffende deelvak.

Niet voor alle studenten is S2 de 2e periode van de propedeuse. Studenten met een MBO-opleiding, die aanpalend is aan onze ICT-propedeuse of studenten die in de propedeuse van een informatica-opleiding op een universiteit hebben gestudeerd, zouden toestemming hebben kunnen krijgen om direct in S2 instromen. Voor deze studenten zijn de verschillen in het programma in deze handleiding aangegeven.

1.2. Verbeteringen

Uiteraard zijn we in ons onderwijs continu bezig met inhoudelijke en didactische verbeteringen. Leidraad hierin zijn naast nieuwe innovaties vooral ook jullie feedback. Daarom vinden we de blokevaluaties ook erg belangrijk. Niet alleen vragen we jullie om middels een enquête om jullie mening, maar we gaan aan de hand van die enquête ook het gesprek aan met de klassen.

Aan de hand van de feedback van afgelopen semester hebben we de organisatie van het semester weer verbeterd. Verder hebben we voor de volgende punten verbeteringen toegepast:

- Studenten gaven aan dat het regelmatig onduidelijk was wanneer een Workshop plaats zou vinden en voor wie de Workshop bedoeld is. We hebben de namen van de Workshops daarom aangepast. Wanneer een Workshop WSROL [rolnaam] heet, is deze alleen bedoeld voor studenten die in de proftaak die genoemde rol vervullen. Wanneer een Workshop WSALG [workshopnaam] heet, is deze voor alle studenten bedoeld. Daarnaast hebben we de Workshops in de Canvas Course voor PTS2 opgenomen, waardoor ook van daaruit duidelijk is wanneer welke workshop plaats zal vinden.

- Er werd door de meeste studenten aangegeven dat de keuzevrijheid bij de proftaak als erg prettig werd ervaren, echter was niet bij elke proftaak-opdracht de beschrijving even duidelijk. We hebben deze opdrachtbeschrijvingen en de bijbehorende studentenhandleiding daarom herschreven en deze is nu vele malen duidelijker.
- Een aantal studenten heeft aangegeven het vak IN2 iets te los vinden staan naast de rest van het semester en ook de volgorde van sommige practica waren onhandig. Om die reden hebben de vakeigenaren van SE2 en IN2 opdrachten gecreëerd waar kennis van beide vakken toegepast kan worden zodat duidelijk wordt waar deze vakgebieden elkaar raken. Dit gaan we in de toekomst ook nog met DBS2 doen. Ook hebben we de volgorde van de practica aangepast aan de opmerkingen en zijn er van overtuigd dat deze nu beter op elkaar aansluiten.
- Omdat de keuzemogelijkheid bij de proftaak zo positief is ontvangen, hebben we deze keuzevrijheid in verschillende mate ook op de andere vakken toegepast.

2. Inhoud

2.1. Deelvakken van dit semester

De onderwijseenheid bestaat uit een aantal deelvakken.

Dit hoofdstuk beschrijft de inhoud van de deelvakken van semester 2.

Code	Omschrijving	bijdrage eindcijfer
PTS2a	EyeCT (Proftaak)	6/22
SE2a	Software Engineering 2	6/22
IN2	Infrastructuren 2	5/22
DBS2	Databases 2	5/22

Elk deelvak wordt beoordeeld met een geheel cijfer (0 decimalen). De bepaling van dit eindcijfer kan per deelvak verschillen en is te vinden in dit blokboek en verder toegelicht in de studentenhandleiding. In geval van twijfel is de informatie in dit blokboek leidend. In de vakken moet je werken aan beroepsproducten, de beoordeling van zo'n beroepsproduct bepaalt mede je eindcijfer.

Naast deze reguliere onderdelen van het semester krijg je de gelegenheid om het vak TAV2: Toelating Academische Voorbereiding te volgen. Dit vak is facultatief en maakt geen deel uit van het reguliere lesprogramma.

Tevens wordt in dit semester het Assessment Professionale Ontwikkeling Propedeuse (APOP) afgerond. Dit is een aparte onderwijseenheid van 12 EC.

Informatie over PTS2a EyeCT

Inhoud:

De vakken (SE2a, DBS2, IN2) zijn ondersteunend voor de proftaak. Doel van het vak is om de student de vaardigheden die hij als ICT-er in dit semester leert integraal in de praktijk toe te passen. Het geheel is meer dan de som der delen.

Daarnaast oefent de student door het doen van de proftaak ook in rollen als samenwerken, projectmanagement en onderzoek. Het geleerde kan worden opgenomen in het portfolio voor professionele ontwikkeling.

Leerdoelen:

Doel van deze proftaak is om op het gebied van software ontwikkeling, infrastructuur en databases kennis en vaardigheden integraal te verwerven in de context van een realistische opdracht. De proftaak wordt parallel aan een aantal ondersteunende vakken aangeboden.

Toetsing en eindbeoordeling:

1 Hoe wordt het vak getoetst?

De beoordeling van de beroepsproducten gaat volgens de onderstaande tabellen:

Applicaties	Sterk (3)	Voldoende (2)	Matig (1)	Zwak (0)	Score
Volledigheid					
Functionaliteit					
Correctheid &					

Robuustheid					
Begrijpelijkheid & Efficiency					
Uitbreidbaarheid & Herbruikbaarheid					
Gebruikte technologieën (eindproduct)					
Documentatie bij code					
				Totaal	

Documentatie	Sterk (3)	Voldoende (2)	Matig (1)	Zwak (0)	Score
Kwaliteit en volledigheid info					
Opmaak en structuur					
Taalbeheersing en leesbaarheid					
Diagrammen en tabellen					
Agenda en notulen					
				Totaal	

Oplevering	Sterk (3)	Voldoende (2)	Matig (1)	Zwak (0)	Score
Inlevering					
Structuur bij oplevering (tijd, opbouw, inhoud)					
Ondervraging bij oplevering					
Omgang met derden					
Testplan uitvoering					
				Totaal	

Ieder beoordelingsaspect levert een score tussen 0 en 3 punten op. Het totaal geeft een normering voor het cijfer. Bij Applicaties geldt de normering:

Score	Cijfer
>12	8, 9, 10
6 t/m 12	6, 7
<6	<5

Voor oplevering geldt de normering:

>10	8, 9, 10
5 t/m 10	6, 7
<5	<5

2 Toegestane hulpmiddelen

Niet van toepassing.

3 Herkansing en/of reparatie

Bij de 1e tussentijdse oplevering (week 6) krijg je te horen of je meteen door mag of dat er nog een reparatie moet worden uitgevoerd. Bij de 2e tussentijdse beoordeling (week 10) krijg je een Go of NoGo.

Bij de eindoplevering (week 16/17) wordt het cijfer vastgesteld en ook hier kan er eventueel nog een reparatie worden gedaan. Als dit niet lukt, is er een herkansingsmogelijkheid na een halfjaar.

4 Hoe wordt de eindbeoordeling vastgesteld?

Tijdens een proftaak voer je een beroepstaak uit die leidt tot drie beroepsproducten.

- Documentatie
- Applicaties
- Oplevering

Deze deelcijfers moeten met ≥ 5.5 zijn beoordeeld om een eindcijfer te krijgen. De formule voor het eindcijfer is:

Eindcijfer = $0.5 \cdot \text{applicaties} + 0.3 \cdot \text{documentatie} + 0.2 \cdot \text{oplevering}$, maar nooit minder dan 1.

Onderwijsvorm:

In de weken 1-17 is 7 uur (1 dag) per week ingeroosterd voor het werken aan de proftaak. Studenten werken in groepen van maximaal 5 personen. Daarbij heeft iedere projectgroep wekelijks een spreekuur met de tutor. Hierin wordt de voortgang van de proftaak besproken, zowel procesmatig als inhoudelijk. De weken 6, 10 en 16/17 zijn gepland voor (tussentijdse) afronding van het project, waarbij week 6 en 10 een GO-NOGO moment behelst en in week 17 de eindbeoordeling plaats zal vinden.

Lesmateriaal:

Opdrachtoomschrijvingen, boeken, lesmateriaal uit alle flankerende vakken.

Informatie over SE2a Software engineering 2

Inhoud:

In de onderwijsactiviteit SE2a wordt het bouwen van software op een methodische wijze uitgewerkt. Daarbij stel je de eisen die je aan je systeem stelt (requirements) op met behulp van een use case rapportage. Tijdens de ontwerpfase geef je aan hoe je de software gaat bouwen. Daartoe maak je het beroepsproduct UML klassendiagram. Hierin beschrijf je welke classes een rol spelen in de software en welke rol dat is (methoden/attributen). Daarnaast beschrijf je de samenwerking tussen classes met behulp van associaties en multipliciteitsaanduiding.

Ten slotte realiseer je het systeem door het implementeren van het ontwerp met behulp van de OO-taal C#. Een OO-programma is derhalve het derde beroepsproduct.

Het 4e en laatste beroepsproduct is de ontwikkeling van een webapplicatie. Je maakt individueel een webapplicatie in ASP.NET volgens de methodologie geleerd in dit semester. Hierbij maak je gebruik van professionele tools en technieken om de kwaliteit te verkrijgen die vereist is.

Leerdoelen:

Onderwerpen:

- Programmeervaardigheden C#:
 - Event handling
 - Inheritance
 - Abstracte klassen en interfaces
 - Sorteren m.b.v. IComparable/IComparer
 - Unit testing
 - Streams en files
 - Exception handling
 - Databases en Data Access Layers
- UML (use case diagram, use cases en klassendiagram)
- Softwaredocumentatie (analysedocument, ontwerpdocument, acceptatie-testplan)
- Algoritmiëk
- Versiebeheer
- Codeerstijl
- Webprogrammeren met ASP.NET
- Deployment (ontwikkel- en productieomgevingen)

Toetsing en eindbeoordeling:

1 Hoe wordt het vak getoetst?

Toets 1: Toets van 1 lesuur met gebruik van de in de eerste twee weken ontwikkelde applicatie. Usecases en usecase diagrammen kunnen opstellen vanuit een gegeven beschrijving.

Toets 2: Toets van 1 lesuur op papier. Klassendiagram (domeinmodel) vanuit een gegeven beschrijving ontwerpen en een klassendiagram (implementatiemodel) maken vanuit gegeven code (reverse engineering).

Toets 3: Programmeertoets van 200 minuten. Voor deze toets geldt een bonusregeling: in de studentenhandleiding van het vak staat de precieze opzet beschreven. Door aantoonbaar actief met het vak bezig te zijn kan 1 punt bonus behaald worden. De beoordeling geschiedt volgens onderstaande matrix:

Leerdoel	Sterk	Voldoende	Matig	Onvoldoende
Klassendiagram omzetten	Het klassendiagram en de code komen exact overeen.	Het klassendiagram en de code komen in grote lijnen overeen. De klassen en relaties zijn juist, maar enkele methoden en attributen zijn niet aanwezig.	De klassen komen overeen, maar de relaties zijn niet overal juist, enkele methoden en attributen zijn niet aanwezig.	Klassen verschillen of zijn niet aanwezig.
Overerving	Overerving is compleet en juist geïmplementeerd.	Overerving is niet compleet geïmplementeerd,	Overerving is geïmplementeerd, maar methodes	Er is geen overerving toegepast.

		en/of methodes en constructoren roepen niet de baseklasse aan wanneer mogelijk.	en/of constructoren ontbreken.	
Sorteren	Sorteren is zowel met IComparable als IComparer juist toegepast.	Sorteren is met IComparable als IComparer toegepast, maar de invulling is niet geheel correct.	Sorteren is maar met 1 van de 2 interfaces toegepast.	Sorteren is niet met de gevraagde interfaces toegepast.
Streams	Er wordt correct gebruik gemaakt van de juiste stream om een klasse of bestand op te slaan.	De juiste stream wordt gebruikt, maar niet alles wordt correct opgeslagen.	Er wordt een stream gebruikt, maar niet de juiste.	Er worden geen streams gebruikt.
Exceptions	Risicovolle code wordt netjes afgevangen met exception handling; de gebruiker wordt wanneer relevant over de fout geïnformeerd.	Risicovolle code wordt netjes afgevangen met exception handling; de gebruiker krijgt geen duidelijke melding.	Een enkele keer wordt exception handling toegepast, maar op veel plaatsen ontbreekt deze; of exception handling wordt overdadig veel gebruikt.	Er wordt geen exception handling toegepast.
Interfaces	Interfaces worden op de juiste manier gemaakt, gebruikt en geïmplementeerd.	Interfaces worden op de juiste manier gemaakt en/of gebruikt, maar de implementatie is niet geheel correct.	Er is een interface aangemaakt maar niet gebruikt, of een gebruikte interface is niet geïmplementeerd.	Er is geen gebruik gemaakt van interfaces.
Unit testing	Unit tests zijn zinnig, specifiek en opgesplitst op basis van individuele functies; functies zijn volledig gedekt door de tests.	Unit tests zijn breed van opzet en testen slechts een deel van de mogelijkheden van de geteste functies.	Er zijn weinig unit tests aangemaakt, of deze zijn erg elementair en bieden weinig meerwaarde bij het ontwikkelproces.	Er zijn geen unit tests geïmplementeerd.
Functionaliteit	De gehele user interface werkt probleemloos; event handling is correct geïmplementeerd.	Nagenoeg de hele user interface is functioneel, maar een enkele knop heeft een onvolledige of onjuiste functie; niet overal is event handling correct toegepast.	Gegevens worden getoond bij het opstarten van de applicatie, en/of 1 of 2 knoppen werken naar verwachting.	Er kan geen functionaliteit gebruikt worden via de user interface.
Codeerstijl	Code is wanneer relevant te vinden in klassen; functies en blokken code zijn voorzien van commentaar; duidelijke verantwoordelijkheid voor iedere functie.	Functies zijn aangemaakt om dubbele code te voorkomen; functies zijn voorzien van beschrijvend commentaar; code heeft weinig nesting; functies hebben een	Variabelen hebben beschrijvende namen; af en toe is commentaar toegevoegd; veel dubbele code.	Er is geen aandacht besteed aan de codeerstijl: variabelen hebben onduidelijke namen; uitlijning is inconsistent;

		duidelijke verantwoordelijkheid.		veel dubbele code; overdadige nesting.
Architectuur	De applicatie heeft een duidelijke scheiding tussen de aanwezige lagen en er kan zonder problemen nieuwe functionaliteit toegevoegd worden.	Er is een scheiding tussen de lagen, maar de code in deze lagen staat niet op de goede plek. Het uitbreiden van de applicatie betekent dat bestaande code gewijzigd moet worden	Code is onderverdeeld in klassen, maar veel functionaliteit is nog altijd in de event handlers te vinden.	Er is geen rekening gehouden met de structuur van de applicatie: heel veel code staat direct in de event handlers.

Toets 4: Ontwikkeling webapplicatie met database. Hierbij wordt de onderstaande beoordelingsmatrix gebruikt.

Beoordeling	Sterk	Voldoende	Matig	Onvoldoende
Omschrijving opdracht	Opdracht voldoet aan de gestelde functionele eisen. Document heeft een goede opbouw en nette opmaak. Voldoet aan HBO-schrijfstijl. Is op tijd aangeboden ter beoordeling.	De opdracht is beschreven en duidelijk, maar het is geen geheel of heeft een te simpele schrijfstijl.	Er is een opdracht beschreven, maar niet alle vereiste onderdelen zijn inbegrepen.	Geen opdracht ingeleverd, of wat is ingeleverd is dermate beperkt dat er geen duidelijk beeld van de opdracht gevormd kan worden; of de opdracht is te laat ingeleverd.
Kennisdeling	Duidelijke presentatie met sprekende voorbeelden; meerwaarde voor anderen is duidelijk. Klas geeft gemiddeld hoger dan een 7,5.	Presentatie en demonstratie gegeven, maar de meerwaarde voor anderen is niet direct duidelijk. Klas geeft gemiddeld een voldoende, maar niet hoger dan een 7,5.	Er is een presentatie gegeven, maar het is onduidelijk wat de meerwaarde van het onderwerp is voor anderen. Klas geeft gemiddeld een onvoldoende.	Geen presentatie gehouden.
Analysedocument	Voor alle functionaliteit is een juiste use case aanwezig. Het use case diagram toont alle use cases; deze zijn gekoppeld aan de juiste actoren. Scenario's, requirements, ERD en UI schetsen aanwezig en van goede kwaliteit.	Er is een use case diagram, en voor de belangrijkste functionaliteiten zijn juiste use cases aanwezig. Scenario's, requirements, ERD en UI schetsen aanwezig en van voldoende kwaliteit.	Er is een use case diagram, maar een aantal relevante use cases ontbreken. Scenario's, requirements, ERD en UI schetsen aanwezig maar zeer matig.	Er is alleen een use case diagram en geen of slechts enkele use cases, of het Scenario's, requirements, ERD of UI schetsen ontbreken.
Ontwerpdocument	Bevat een klassendiagram en	Bevat de juiste diagrammen en	Diagrammen en onderdelen in	Er ontbreken diagrammen of

	databaseontwerp, architectuur- en infrastructuuromschrijving: deze zijn van een goede kwaliteit. Er is een inleiding en conclusie; tabellen en modellen zijn waar nodig toegelicht.	onderdelen, maar de toelichtingen, inleiding of conclusie ontbreekt; of de diagrammen zijn onvolledig.	het document zijn niet volledig of bevatten grote fouten, toelichtingen ontbreken.	onderdelen, of er is geen document opgeleverd.
Architectuur	Er is een heldere meerlaags structuur in de software te zien. Modellen, database en GUI zijn duidelijk gesplitst. De applicatie is goed testbaar via unit tests.	Meerdere lagen zijn te herkennen, maar er is code te vinden die beter in een andere laag had kunnen worden geplaatst.	Geen duidelijke splitsing tussen modellen, business logic en GUI; database code is verspreid over de gehele applicatie.	Business logic staat in de GUI, geen of weinig aparte klassen voor modellen.
Versiebeheer	Consequent en helder reden van wijzigingen aangegeven, commits bevatten afgebakende stukken functionaliteit.	Versies hebben duidelijke omschrijvingen, maar zijn gemaakt op tijd i.p.v. op functionaliteit.	Er is versiebeheer gebruikt, maar er zijn maar een handvol versies gemaakt.	Zo goed als geen gebruik gemaakt van versiebeheer.
Broncode	Robuust, begrijpelijk, uitbreidbaar, efficient, helder en duidelijk commentaar, goed gestructureerd.	Compleet en functioneel, maar beperkte foutafhandeling; communicatie met database kan crashen; code staat op de verkeerde plaats.	De applicatie werkt grotendeels, maar alleen als een specifiek pad bewandeld wordt; ontbrekende foutafhandeling, fouten in code.	De applicatie kan niet werkend gedemonstreerd worden; ernstige problemen met de code.
ASP.NET technieken	Prima MVC implementatie, doordachte Web API opgezet en een Web API van collega op functionele en robuuste wijze geïntegreerd.	Goed gebruik gemaakt van het MVC model, maar de implementatie kon efficiënter; geen goed gebruik van een Web API (eigen opzet, en/of gebruik van collega).	Het is een ASP.NET applicatie, maar er zijn bijna geen technieken gebruikt om de applicatie efficiënt op te bouwen.	De applicatie is slechts een HTML-pagina, met hier en daar wat dynamische componenten toegevoegd.
Documentatie van code	De verantwoordelijkheid van elke klasse is beschreven. Alle methodes zijn voorzien van een toelichting. Complexe stukken code zijn toegelicht.	Korte beschrijving van de klassen; methoden zijn voorzien van een toelichting.	Geen beschrijving bij klassen. Wel toelichting op methoden.	Geen, of alleen eenvoudig commentaar bij code en methoden.
Acceptatie testplan	Opgezet volgens eisen en compleet uitgevoerd. Conclusie bijgewerkt op basis van het resultaat van de tests.	Niet volledig volgens de eisen opgezet en uitgevoerd.	Er is een testplan opgezet en voorzien van een matrix, maar niet uitgevoerd; of	Geen compleet testplan opgeleverd, of niet uitgevoerd.

			het testplan dekt niet alle requirements af.	
Unit testen	Minstens 50% van de functies van de modellen wordt getest middels unittests; testen zijn zinnig en opgesplitst op basis van individuele functies.	Minstens 25% van de functies van de modellen wordt getest middels unittests; de testen zijn breed en testen slechts een deel van de mogelijkheden.	Er zijn slechts een handvol unittests aangemaakt, maar deze zijn erg elementair en bieden weinig meerwaarde bij het ontwikkelproces.	Er zijn geen unittests geïmplementeerd.
Deployment	Alles draait op de productieserver, deze is goed ingericht en wordt bijgewerkt na het doorlopen van een simpele softwareontwikkelstraat.	Alles draait op de productieserver, maar het bijwerken van de omgeving gebeurt handmatig.	De productieserver is onvolledig ingericht; alleen de database draait hier op.	Er is geen gebruik gemaakt van een productieomgeving: alles draait op de lokale pc.

Het cijfer voor de individuele opdracht is opgebouwd uit 3 deeltcijfers:

- Analyse: 20% van het eindcijfer
- Ontwerp: 30% van het cijfer
- Realisatie: 50% van het cijfer

Bij de oplevering van de analyse- en ontwerpfase kan een NoGo gegeven worden, waarna eenmalig een reparatie uitgevoerd mag worden.

2 Toegestane hulpmiddelen

Voor de UML-onderdelen is een cheatsheet beschikbaar: dit is het enige toegestane hulpmiddel bij de toetsen die op papier worden afgenomen. Overige toetsen worden op de eigen laptop afgenomen waar verder geen restricties gelden ten aanzien van de software en naslagwerken die lokaal beschikbaar zijn. Ook het internet is vrij toegankelijk.

Wel wordt er gecontroleerd op filesharing en chatapplicaties: het gebruik van deze tijdens de toets is verboden.

3 Herkansing en/of reparatie

Voor toets 1 en 2 is er een herkansing in de opvolgende lesweek. Toets 3 heeft een herkansing omstreeks week 13.

Na de beoordeling van het 4e beroepsproduct zal de docent bepalen of je in aanmerking komt voor een reparatie. Deze reparatie kan dan eventueel in week 17 worden uitgevoerd. Voor de andere beroepsproducten geldt standaard de FHICT-herkansingsregeling.

4 Hoe wordt de eindbeoordeling vastgesteld?

Toets 1 en toets 2 dienen beiden voldoende te zijn om in aanmerking te komen voor een eindcijfer. Het eindcijfer wordt bepaald door het resultaat van toets 3 twee keer en het resultaat van toets 4 een keer op te tellen en dit door drie te delen. In andere woorden: toets 3 bepaalt voor 2/3 je eindcijfer; toets 4 voor 1/3. Voorwaarde hierbij is dat alle toetsen met minstens een voldoende zijn afgesloten. Indien voor tenminste een van de twee toetsen een onvoldoende is behaald wordt het laagst behaalde cijfer geregistreerd in het cijfersysteem.

Onderwijsvorm:

In de weken 1 t/m 16 zijn er steeds drie blokken van 4 uren ingeroosterd. Dit is voor theorie en practicum samen. De docent is hierbij 5 uur per week aanwezig.

Lesmateriaal:

Er is geen lesmateriaal benodigd voor dit vak. Alle gebruikte materialen worden beschikbaar gesteld via de onderwijsomgeving.

Informatie over IN2 Infrastructuren 2

Inhoud:

In het vak Infrastructuren leer je de theorie en de praktijk omtrent het inrichten van computernetwerken. Daarbij komt zowel de theoretische grondslag in de vorm van onder meer het OSI-model aan bod, als de praktische kennis van het configureren van Windows-netwerken. Ook het ontwerpen van een netwerkstructuur voor webtoepassingen, het inrichten van webserver en netwerkrouting komt aan bod.

Leerdoelen:

Onderwerpen:

- Windows X cliënt en Server 20XX installatie
- Active directory
- Remote beheer
- IP-configuratie en beheer
- DHCP-configuratie
- Netwerkmonitoring TCP/UDP
- DNS
- Mail
- Web/FTP
- Security

Toetsing en eindbeoordeling:

1 Hoe wordt het vak getoetst?

Door het uitvoeren van practicumopdrachten toon je aan dat je begrippen zoals benoemd in het deelvak in de praktijk kunt toepassen. De volgende producten moeten worden opgeleverd:

- LAN-netwerkinstallatie
- Windows Active Directory installatie
- DHCP-configuratie
- DNS-serverinstallatie en -configuratie
- Client-server omgeving t.b.v. email
- Webserver met beveiligingsvoorzieningen

Er zijn twee opleveringen (lesweek 8 en lesweek 16) van de voorafgaand gemaakte practica. Tijdens deze opleveringen moet je (in tweetallen) een demonstratie geven van je gemaakte systeem. Je licht het toe en beantwoordt vragen van de docent. De docent bepaalt het cijfer naar aanleiding van de kwaliteit van de demo, de beantwoording van de vragen en de kwaliteit van de vooraf ingeleverde documentatie. Zie voor details de rubrics in studentenhandleiding.

2 Toegestane hulpmiddelen

- o **De theorie en opdrachten op Canvas.**
- o De InfraWiki-pagina.
- o Het InfraForum.
- o Online boek Computer Networking; Principles, Protocols and Practice (CNP3).

3 Herkansing en/of reparatie

Indien je producten niet hebt gehaald kun je deze in samenspraak met de docent repareren door middel van een alternatieve opdracht. Dit kan alleen als je een gedeelte van het practicum niet hebt kunnen afronden.

4 Hoe wordt de eindbeoordeling vastgesteld?

Tijdens de toetsing krijgt de student twee deelresultaten (de twee opleveringen).

Het eindcijfer is het gemiddelde van deze twee cijfers. Voldoende is eindcijfer ≥ 5.5 .

Onderwijsvorm:

In de weken 1 t/m 16 worden er wekelijks 4 contacturen met de docent ingeroosterd. Tijdens deze uren worden de genoemde onderwerpen toegelicht en toegepast in practicumopdrachten. Verder wordt er nog per week 2 uur zonder docent ingeroosterd.

Lesmateriaal:

Boek: Computer Networking : Principles, Protocols and Practice, Olivier Bonaventura, [Hier online beschikbaar](#)

Informatie over DBS2 Databases 2

Inhoud:

Hoe vang je zo'n oneindig uitbreidbare menustructuur van een website in een database? Hoe koppel je aan een nieuwsbericht andere relevante nieuwsberichten zonder alles dubbel in je database op te hoeven slaan?

Wat is de slimste oplossing als je informatie over je muziekcollectie in een database kwijt wil kunnen? Alles in één tabel opslaan? Of in twee? Of kan het nog beter?

En hoe dwing je af en controleer je dat gegevens die in een database worden opgeslagen wel correct zijn?

En hoe maak, vul en wijzig je de database direct vanuit je software op een snelle en efficiënte manier?

Als je bovenstaande praktijkpuzzels op een professionele manier op wil kunnen lossen komen de onderwerpen en technieken uit dit vak goed van pas.

Leerdoelen:

Eerste deel semester:

- Gegevensmodellering, ERD-model (voor gevorderden)
- Datamodel omzetten in databaseontwerp (voor gevorderden)
- Realiseren databaseontwerp in DDL (SQL)
- Manipuleren van gegevens m.b.v. DML (SQL) (voor gevorderden)

- Integriteit bewaken d.m.v. SQL-constraints

Tweede deel semester:

- Programmeren op databaseniveau middels PL/SQL
- Meer complexere integriteitsregels bewaken d.m.v. PL/SQL

Toetsing en eindbeoordeling:

1 Hoe wordt het vak getoetst?

Een belangrijk beroepsproduct voor het deelvak DBS2 is het volledig, individueel doorlopen van het ontwikkeltraject van een relationele database.

Dat betekent:

- je maakt een databasemodel op basis van verkregen requirements,
- je maakt het databaseontwerp,
- je creëert en vult de database met relevante testgegevens,
- je schrijft de query's om de informatie op te halen.

Een ander beroepsproduct is een SQL-toets. Als laatste beroepsproduct is er de PLSQL-opdracht.

Alle beroepsproducten dienen minimaal met een 5.5 afgerond te worden. Deze producten worden getoetst d.m.v. beoordelingsmatrix.

Ontwikkelopdracht

Beoordeling	Goed	Voldoende	Matig	Onvoldoende
Beschrijving	Tekstuele beschrijving van de entiteiten en hun onderlinge relaties. Voor ieder attribuut een beschrijving van de betekenis en het type.	Functionele eisen aan de database worden duidelijk beschreven.	Er wordt beschreven welke informatie uit de database kan worden gehaald.	Er wordt beschreven welke gegevens worden opgeslagen in de database.
Entity-relationship diagram	Voor subtypering is discriminator, inclusief/exclusief en verplicht/niet-verplicht correct aangegeven. Recursieve relatie is geannoteerd.	ERD bevat subtypering en tenminste een recursieve relatie. Kardinaliteiten zijn correct weergegeven.	ERD bevat subtypering of een recursieve relatie. Een-op-een, een-op-veel en veel-op-veel relaties zijn weergegeven.	ERD bevat tenminste 8 entiteiten en relaties.
Relationeel database ontwerp	Relationeel database ontwerp is weergegeven als strokendiagram. Relatie tussen foreign key en bijbehorende primary key is weergegeven.	Primary keys en foreign keys zijn correct aangegeven op de gebruikelijke manier.	Een-op-veel relaties en veel-op-veel relaties worden correct weergegeven in tabellen en attributen.	Tabellen en attributen worden tekstueel weergegeven.
SQL create script	Database is gevuld met relevante voorbeeld data en wordt uitvoerig getest.	Andere constraints aanwezig en SQL statements om constraints te testen.	Create statements voor alle tabellen en constraints voor foreign keys.	Create statements voor alle tabellen.
SQL view script	View en tenminste select statements met outer-join, auto-join en gecorreleerde subquery.	View en twee voorbeelden van select statements met outer-join, auto-join of gecorreleerde subquery.	View en een voorbeeld van select statement met outer-join, auto-join of gecorreleerde	View en eenvoudige select statements.

			subquery.	
--	--	--	-----------	--

SQLtoets

Bij deze toets moeten, op een gegeven database, ingewikkelde informatievraagstukken met behulp van SQL worden opgelost.

Criteria

- Gegeven constraints (tekst) kunnen indelen in de verschillende typen constraints
- SQL Data Definition Language (DDL) scripts maken vanuit een relationeel database ontwerp
 - CREATE TABLE statement
 - DROP TABLE statement
 - CREATE VIEW statement
 - DROP VIEW statement
 - Hierbij implementeren van constraints
- SQL Data Manipulation Language (DML) scripts maken vanuit een relationeel databaseontwerp
 - UPDATE statement
 - DELETE statement
 - INSERT statement
 - SELECT (INNER JOIN, OUTER JOINS, AUTO JOIN, gecorreleerde subqueries)

De normering wordt gegeven bij de toets.

PL/SQL-practicum

In het PL/SQL-practicum werk je aan een aantal grote practicumopdrachten. Tegen het einde vindt er een oplevergesprek plaats van ongeveer 10 minuten, waarin je je PL/SQL-code live kan demonstreren, verantwoorden en onderbouwen. Tevens zullen er vragen over je werk worden gesteld. De docent bepaalt het cijfer naar aanleiding van de kwaliteit van de demo, de beantwoording van de vragen en de kwaliteit van het document. Zie voor details de rubrics in studentenhandleiding.

2 Toegestane hulpmiddelen

Alle hulpmiddelen zijn toegestaan. Bronnen dienen vermeld te worden. Tijdens het SQL-tentamen mag er niet samengewerkt worden.

3 Herkansing en/of reparatie

Voor de ontwikkelopdracht is er een herkansingsmogelijkheid tijdens het tweede deel van het semester. Tevens is er een reparatiemogelijkheid als blijkt dat de opdracht op kleine puntjes nog kan en moet worden verbeterd voor een voldoende. De reparatie dient binnen 1 week te worden ingeleverd en beoordeeld. Reparatie en/of uitstel is dus niet mogelijk als je (te) weinig tot niets en/of te laat inlevert en/of niet net onvoldoende is!

Voor SQL toets is er een herkansing in week 13 of 14 van het semester. Dit kan mogelijk plaatsvinden in de avonduren. Zie hiervoor het rooster. Deze toets mag je ook opnieuw doen om een hoger gemiddelde te halen. Inschrijven voor de herkansing gaat via Progress en is verplicht.

Voor het PL/SQL-practicum is de herkansingsmogelijkheid een schriftelijke toets in week 18. Hiervoor moet je inschrijven via Progress.

4 Hoe wordt de eindbeoordeling vastgesteld?

Het eindcijfer DBS2 is een geheel cijfer tussen 1 en 10 en wordt berekend volgens de volgende formule:

0.4 * Ontwikkelopdracht + 0.2 * SQL-Toets + 0.4 * PL/SQL practicum.

Indien één van de drie beroepsproducten onvoldoende is geldt ook bovenstaande formule, gemaximaliseerd tot een 5.

Onderwijsvorm:

In de weken 1 t/m 16 worden er wekelijks twee keer twee contacturen met de docent ingeroosterd. Tijdens deze uren worden de genoemde onderwerpen toegelicht en geoefend met allerlei practicumopdrachten. Verder wordt er nog per week 2 uur zonder docent ingeroosterd.

Lesmateriaal:

Boek: Leerboek Oracle PL/SQL, 2e of 3e druk; Gilbert Rattink; Academic Service; ISBN: 13 978-90-395-2661-3 / EAN: 9789039526613

Boek: Databases, Beginselen, ontwerp en implementatie; 11e of 10e druk; D. M. Kroenke; Pearson Education; ISBN: 978 90 43019 87 3

2.1.1. Professionele Ontwikkeling (PPO)

Bij PO werk je verder aan je ontwikkeling tot ICT-professional. Het PO-dossier wordt verder opgebouwd met feedback van de verschillende onderwijseenheden. De SLB'er zal je hierin weer ondersteunen en een coachende rol spelen bij je verdere ontwikkeling.

Verder zijn er workshops waarin je kunt oefenen in de beroepsvaardigheden projectmanagement, onderzoeksvaardigheden, informatie verzamelen en persoonlijk leiderschap.

2.1.2. Assessment Professionele Ontwikkeling Propedeuse

Het assessment professionele ontwikkeling propedeuse (APOP) is een aparte onderwijseenheid van 12 ECTS. Deze onderwijseenheid is de afronding van de professionele ontwikkeling (PO) die je gedurende de propedeuse hebt doorgemaakt en je moet ook nogmaals aantonen dat je alle technische competenties in huis hebt van het propedeuseniveau van Software Engineering. In deze onderwijseenheid word je beoordeeld op het niveau van "de lerende professional". Het is daarom een propedeuse-assessment.

Het APOP bestaat uit de onderdelen PPO, het LPA en een CGI.

Informatie over deelvak TAV2 Toelating Academische Voorbereiding (facultatief)

TAV2 is een toelatingseis tot de differentiatieroute Academische Voorbereiding, deze route is bedoeld voor studenten die na hun HBO-ICT-opleiding een masteropleiding aan een Universiteit willen volgen. Vooralsnog bestaat deze studieroute alleen voor de masters Computer Science and Engineering (CSE), Data Science in Engineering (DSE), Embedded Systems (ES) en Business Information Systems (BIS) aan de Technische Universiteit Eindhoven.

<http://www.tue.nl/en/education/tue-graduate-school/masters-programs/> (Links to an external site.)

TAV2 is ook een facultatief vak voor alle studenten van FHICT. Tegenwoordig hebben de HBO-ICT-opleidingen weinig tot geen wiskunde meer in hun curriculum. Dat is ook geen probleem omdat je breed wordt opgeleid en je eigen weg kan kiezen. Maar op het gebied van 'goed' programmeren is een wiskundebasis wel belangrijk. Denk hierbij aan het analyseren van algoritmen, het berekenen van efficiëntie of betrouwbaarheid van code en cryptografie. Ook voor bepaalde innovatiegebieden of specialisaties is TAV2 een goede basis, zoals bijvoorbeeld op het gebied van graphics of data science.

Let op: Er worden maximaal 24 studenten toegelaten (1 klas). Studenten die de academische route gaan doen hebben voorrang.

Inschrijven gaat via sharepoint:

<https://portal.fhict.nl/IS/S2/Lists/Keuze%20TAV2/NewForm.aspx?Source=https%3A%2F%2Fportal%2Efhict%2Enl%2FIS%2FS2%2FLists%2FKeuze%2520TAV2%2Foverview%2Easpx> (Links to an external site.)

Doel van TAV2 is bepalen of je geschikt bent voor het pre-masterprogramma van de TU/e. Voor studenten die niet de academische voorbereiding gaan doen, is het doel een stevige wiskundige basis te leggen om op 'scientific' niveau als programmeur te gaan functioneren.

De onderwerpen verzamelingen, logica en lineaire algebra worden in 4 uur per week in klassikale lessen behandeld. Hierbij moet nog ongeveer 4 per week aan zelfstudie worden gedaan.

De lessen lopen van week 2 t/m 7 en van week 10 t/m 16. In week 8 en 17 zijn de toetsen. Herkansing is mogelijk na een ½ jaar.

Informatie over Assessment Professionele Ontwikkeling Propedeuse S

Inhoud:

Een belangrijke stap die je dit semester ook gaat maken is op het gebied van professionalisering. Wat is de juiste beroepshouding van een ICT'er? Hij/zij moet kunnen samenwerken, initiatief tonen, onderzoek kunnen doen, zelfstandig problemen oplossen, communiceren op een professioneel niveau, zowel mondeling als schriftelijk. Deze competenties kun je dit semester aanleren en/of aanscherpen in de verschillende vakken. Verder moet je jouw ontwikkeling van deze vaardigheden aantonen en bewijzen via het Assessment Professionele Ontwikkeling Propedeuse S.

Leerdoelen:

Leerdoel van de PO-lijn is dat de student zich aantoonbaar ontwikkelt tot een ICT-professional. Hiervoor zijn ontwikkelingsgerichte rubrics opgenomen in de OER. In deze onderwijseenheid wordt getoetst of de ontwikkeling enerzijds op het vlak van professionele ontwikkeling (de rubrics) voldoende is en anderzijds op het inhoudelijk vlak van de studierichting die de student heeft gekozen.

Toetsing en eindbeoordeling:

1 Hoe wordt het vak getoetst?

De student maakt in twee dagen tijd een opdracht met de methoden en technieken die hij in zijn propedeusejaar heeft geleerd (de Live Performance). Dit is te vergelijken met een spoedklus in het bedrijfsleven, waar in zeer korte tijd een prototype moet worden opgeleverd.

Tegen het einde van elke dag krijgt hij bezoek van zijn assessoren die de voortgang monitoren en vastleggen.

De uitvoering van de opdracht levert drie beroepsproducten op:

- Het product. Een stuk software zijnde het gebouwde prototype.
- Het verslag, zijnde de documentatie over het product (de opgestelde requirements, de use-cases, het ontwerp van de database en de software).
- Een demonstratie van het prototype (tijdens het Criteriumgericht Interview, CGI; zie hieronder)

Daarnaast heeft de student een portfolio opgebouwd gedurende het hele eerste jaar, dat vooraf door zijn leerprocesbegeleider is goedgekeurd. In dit portfolio reflecteert de student op zijn ontwikkeling aan de hand van de rubrics uit de OER.

Na het live performance vind er een CGI-gesprek plaats tussen de student en de assessoren. Tijdens dit gesprek geeft de student een demonstratie/presentatie van zijn prototype en beantwoordt inhoudelijke vragen van de assessoren.

Elk van de vier genoemde onderdelen wordt met een cijfer beoordeeld. Bovendien wordt er feedback gegeven over de professionele ontwikkeling van de student. De beoordeling en de feedback worden vastgelegd op de formulieren. Deze zijn voor de vier profielen van FHICT alsook de drie fasen (propedeuse, kernfase en eindfase) hetzelfde. Vandaar dat niet alle in de formulieren genoemde criteria van toepassing zijn op dit assessment.

Voor de live performance worden de volgende rubrics gebruikt bij de beoordeling:

	Sterk (3)	Voldoende (2)	Matig (1)	Onvoldoende (0)
Planning	Gedetailleerde planning die ook daadwerkelijk is gevolgd of met reflectie aangegeven waarom van de planning afgeweken is.	Er is een planning opgesteld en de student heeft zich daar in grote lijnen aan kunnen houden.	Er is een grove planning opgesteld die niet of nauwelijks is gevolgd.	Er is geen planning opgesteld.
Professionaliteit documentatie	Alle documentatie past bij elkaar en ziet er keurig verzorgd uit. Alle hoofdstukken en diagrammen zijn voorzien van duidelijke en verklarende uitleg.	Documenten zien er verzorgd uit. Diagrammen zijn voorzien van korte en bondige uitleg.	Er is geen aandacht besteed aan de opmaak van de documentatie. De tekst bevat spelfouten.	Er is geen documentatie. De ontwerpen die zijn gemaakt zijn als losse plaatjes aangeleverd.
Analysedocument	Voor alle functionaliteit is een juiste use case aanwezig. Het use case diagram toont alle use cases; deze zijn gekoppeld aan de juiste actoren. Scenario's, requirements, ERD en UI schetsen aanwezig en zijn van goede kwaliteit.	Er is een use case diagram en voor de belangrijkste functionaliteiten zijn juiste use cases aanwezig. Scenario's, requirements, ERD en UI schetsen aanwezig en zijn van voldoende kwaliteit.	Er is een use case diagram, maar een aantal relevante use cases ontbreken. Scenario's, requirements, ERD en UI schetsen zijn aanwezig maar matig van kwaliteit.	Er is alleen een use case diagram en geen of slechts enkele use cases, of de scenario's, requirements, ERD of UI schetsen ontbreken.

Acceptatietestplan	Opgezet volgens eisen en compleet uitgevoerd. Conclusie bijgewerkt op basis van het resultaat van de tests.	Niet volledig volgens de eisen opgezet en uitgevoerd.	Er is een testplan opgezet en voorzien van een matrix, maar niet uitgevoerd; of het testplan dekt niet alle requirements af.	Geen compleet testplan opgeleverd, of niet uitgevoerd.
Datamodel (ERD)	Complex model met bijv. recursie, veel entiteiten en relaties, subtypering, etc.	Entiteiten, attributen en relaties zijn allemaal correct. Identifiers waar mogelijk aangegeven.	Niet alle entiteiten en attributen goed aangegeven. Kleine fouten in relaties.	Geen of zeer slecht ERD aanwezig.
Databseontwerp (DBO)	Er is een zeer sterk en uitgebreid DBO gemaakt en voorzien van heldere uitleg.	Alle aspecten van het DBO zijn aanwezig. Sleutels zijn correct weergegeven. Het DBO is voorzien van tekst en uitleg.	Het DBO mist wat entiteiten en/of attributen en/of heeft een verdeerde relatie. Er is geen of slechte uitleg.	Er is geen of een zeer slecht database ontwerp gemaakt.
Klassendiagram	Uitstekend klassendiagram (abstract, static, interfaces, etc.) met uitgebreide verantwoording.	Goed klassendiagram met verantwoording.	Klassendiagram bevat wat foutjes en/of geen verantwoording aanwezig.	Geen of een zeer slecht klassendiagram aanwezig.
SQL Create Scripts	Zeer goede constraints aanwezig.	Alle tabellen met sleutels correct. Een aantal constraints aangemaakt.	Alle tabellen zijn correct aangemaakt. Een aantal sleutels ontbreken. Geen constraints.	Er is geen of een zeer slecht create script aanwezig.
Klassendiagram omzetten	Het klassendiagram en de code komen exact overeen.	Het klassendiagram en de code komen in grote lijnen overeen. De klassen en relaties zijn juist, maar enkele methoden en attributen zijn niet aanwezig.	De klassen komen overeen met ontwerp, maar de relaties zijn niet overal juist, enkele methoden en attributen zijn niet aanwezig.	Klassen verschillen van ontwerp of zijn niet aanwezig.
Functionaliteit	Alle must-haves plus een aantal should en could have's aanwezig.	Bijna alle must-haves zijn aanwezig. Eventueel nog een aantal should-haves.	Ongeveer de helft van alle must-haves aanwezig.	(bijna) Geen functionaliteit aanwezig.
Architectuur	Er is een heldere meerlaags structuur in de software te zien. Modellen, database code en GUI zijn duidelijk gesplitst. De applicatie is goed testbaar via unit tests.	Meerdere lagen zijn te herkennen, maar er is code te vinden die beter in een andere laag had kunnen worden geplaatst.	Geen duidelijke splitsing tussen modellen, business logic en GUI; database code is verspreid over de gehele applicatie.	Business logic staat in de GUI, geen of weinig aparte klassen voor modellen.
Codekwaliteit	Functies en blokken code zijn voorzien van commentaar. Duidelijke verantwoordelijkheid	Functies zijn aangemaakt om dubbele code te voorkomen. Functies zijn voorzien van	Variabelen hebben beschrijvende namen. Af en toe is commentaar toegevoegd. Er is	Er is geen aandacht besteed aan de codeerstijl. Variabelen hebben onduidelijke

	voor iedere functie.	beschrijvend commentaar. Code heeft weinig nesting. Functies hebben een duidelijke verantwoordelijkheid.	veel dubbele code aanwezig.	namen. Uitlegging is inconsistent. Veel dubbele code. Overdadige nesting.
Correct & robuust	Alle complexe code is voorzien van foutafhandeling en is voorzien van goede unit tests.	De code is voorzien van goede foutafhandeling en bevat een aantal degelijke unit tests.	De code bevat nauwelijks foutafhandeling en/of er is maar een enkele unit test gemaakt.	Er is niets gedaan in de code om de applicatie robuust te maken.

De rubrics planning, professionele documentatie, analysedocument, acceptatietestplan, datamodel (ERD), databaseontwerp (DBO) en klassendiagram geven samen de beoordeling voor het verslag (30%).

De rubrics SQL-create scripts, klassendiagram omzetten, functionaliteit, architectuur, codekwaliteit en correct & robuust geven samen de beoordeling voor het product (30%).

2 Toegestane hulpmiddelen

Alle hulpmiddelen, behalve expliciete hulp van derden.

3 Herkansing en/of reparatie

Dit is een praktijkgerelateerde toets. Herkansing en/of reparatie is slechts mogelijk als er sprake is van bijzondere omstandigheden en op expliciet verzoek van de student aan de examenkamer.

4 Hoe wordt de eindbeoordeling vastgesteld?

De eindebeoordeling is een gewogen gemiddelde van de vier deeltijfers die worden gegeven tijdens het criterium gericht interview, mits alle onderdelen met een voldoende zijn beoordeeld.

De wegingsfactoren zijn als volgt:

Onderdeel	Weging
Product	30 %
Verslag	30 %
Presentatie	15 %
Dossier professionele ontwikkeling	25 %

Onderwijsvorm:

Integrale toets.

Lesmateriaal:

Dit onderdeel betreft een integrale toets. Lesmateriaal is niet van toepassing, omdat er geen lessen voor dit onderdeel worden gegeven. Feitelijk zijn alle lessen en alle gebruikte lesmateriaal uit het eerste jaar relevant voor dit onderdeel.

2.2. Het versnelde programma

Studenten met een relevante vooropleiding MBO niveau 4 komen mogelijk in aanmerking voor deelname aan het versnelde programma. Indien het een opleiding netwerkbeheerder betrof krijgen ze ook vaak vrijstelling voor het van IN2 (Infrastructuren 2). Deze studenten krijgen na een speciale intake-procedure vrijstelling voor het startsemester en stromen direct in in dit semester 2 van de opleiding.

Qua toetsing gelden voor hun exact dezelfde eisen als in het reguliere programma dat hierboven is beschreven. Echter, omdat gebleken is dat hun voorkennis verschilt met de studenten die regulier eerst het startsemester hebben doorlopen, krijgen zij een apart onderwijsprogramma, met name op het gebied van software engineering en databases. In hun rooster staan deze vakken vermeld als SE2v en DB2v. Deze varianten zijn qua inhoud iets uitgebreider dan hun reguliere varianten en vergen wat meer inzet. In zekere zin bevatten deze varianten van de vakken nog extra informatie die andere studenten al in het startsemester hebben gehad.

2.2.1. Professionele Ontwikkeling (PPOv)

Bij PO werk je aan je ontwikkeling tot ICT-professional. Dit wordt uiteindelijk afgesloten met een reflectieverslag (PO-dossier) dat refereert aan het PI-profiel. Met je SLB'er ga je bepalen op welk niveau je bent als je start in dit semester. Hierna ga je werken om op het eindniveau van de propedeuse te komen in een semester. De SLB'er zal je coachen tijdens deze tocht. De informatie over je ontwikkeling, zoals reflecties, feedback en cijfers, die je doormaakt wordt bijgehouden/vastgelegd in een document (PO-dossier).

Verder zijn er workshops in deze periode waarin je kunt oefenen in de beroepsvaardigheden, informatie verzamelen, vergaderen, projectmanagement en persoonlijk leiderschap.

3. Organisatie

3.1. Uitvoering

In semester 2 zijn er 20 contacturen per klas met een docent ingepland, dit betekent dat studenten minstens 50% van de tijd zelfstandig moeten werken/studeren.

Op het Intranet vind je de namen, foto's en de emailadressen van de docenten, die lesgeven in jouw periode. Zij zijn voor jou het eerste aanspreekpunt over de deelvakken, waar zij verantwoordelijk voor zijn. Met welke docent je specifiek te maken hebt blijkt uit het lesrooster.

De blokeigenaar is verantwoordelijk voor de organisatie en uitvoering (inhoudelijk en didactisch) van de onderwijseenheden voor de semester. Ook hij/zij is altijd aanspreekbaar op dit gebied.

Verder is natuurlijk je studieloopbaanbegeleider beschikbaar voor allerlei hulpvragen.

3.2. Werkplekomschrijving

In het schooljaar 2015-2016 hebben klassen een eigen lokaal, waar ze de hele week gebruik van kunnen maken. Hierbij wordt ook verwacht dat studenten tijdens deze dagen in het lokaal aan het werk zijn ook al is er geen docent aanwezig.

Voor bijna alle onderwijsactiviteiten gebruikt de student zijn of haar eigen laptop. Voor PTS2a en voor IN2 is het mogelijk dat gebruik wordt gemaakt van het SecLab waarin PC's staan opgesteld met een verwisselbare harddisk. Eventueel benodigde harddisks worden dan via de ISSD ter beschikking gesteld. Ook is het mogelijk dat hiervoor gebruik wordt gemaakt van virtualisatie-oplossingen. Verder kunnen diverse projectruimtes worden gereserveerd.

3.3. Leermiddelen

Op Canvas vind je de studiewijzers, diverse handleidingen, lesmaterialen, roosters en verdere aanvullende informatie (inclusief leermiddelen).

3.4. Studiewijzer per onderwijsactiviteit S2

Het lesprogramma van elke deelvak is in een studentenhandleiding weergegeven en op het intranet te vinden. Als er verschillen zijn tussen de informatie in een studiewijzer en dit blokboek, is de informatie in dit blokboek steeds leidend. Aan informatie uit de studentenhandleidingen kunnen geen rechten worden ontleend.