



Stichting NIOC

Stichting NIOC en de NIOC kennisbank

Stichting NIOC (www.nioc.nl) stelt zich conform zijn statuten tot doel: het realiseren van congressen over informatica onderwijs en voorts al hetgeen met een en ander rechtstreeks of zijdelings verband houdt of daartoe bevorderlijk kan zijn, alles in de ruimste zin des woords.

De stichting NIOC neemt de archivering van de resultaten van de congressen voor zijn rekening. De website www.nioc.nl ontsluit onder "Eerdere congressen" de gearchiveerde websites van eerdere congressen. De vele afzonderlijke congresbijdragen zijn opgenomen in een kennisbank die via dezelfde website onder "NIOC kennisbank" ontsloten wordt.

Op dit moment bevat de NIOC kennisbank alle bijdragen, incl. die van het laatste congres (NIOC2025, gehouden op donderdag 27 maart 2025 jl. en georganiseerd door Hogeschool Windesheim). Bij elkaar zo'n 1500 bijdragen!

We roepen je op, na het lezen van het document dat door jou is gedownload, de auteur(s) feedback te geven. Dit kan door je te registreren als gebruiker van de NIOC kennisbank. Na registratie krijg je bericht hoe in te loggen op de NIOC kennisbank.

Het eerstvolgende NIOC vindt plaats op 18 maart 2027 in Arnhem en wordt georganiseerd door HAN University of Applied Sciences.

Reacties over de NIOC kennisbank en de inhoud daarvan kun je richten aan de beheerder:

R. Smedinga kennisbank@nioc.nl.

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.

Ontwerp van een 1-jarige Master Software Engineering aan de Open Universiteit

Auteurs

Prof.dr. Marko van Eekelen
Open Universiteit Nederland en Radboud Universiteit
Email: Marko.vanEekelen@ou.nl

Drs. Marleen Sint
Open Universiteit Nederland
Email: Marleen.sint@ou.nl

Samenvatting

De Open Universiteit heeft een 1-jarige Master Software Engineering in voorbereiding. Bij het ontwerp van deze Master zijn de nieuwste ACM en IEEE richtlijnen als input gebruikt. Een deel van de vakken van deze Master wordt ontwikkeld in samenwerking met de universiteiten van Leuven, Utrecht, Amsterdam en Nijmegen. De ontwerpbeslissingen worden beschreven met hun motivatie. Hierbij komt de koppeling met de ACM en IEEE richtlijnen expliciet aan bod. Het programma is zo opgesteld dat Certified Professional Programs van het curriculum kunnen worden afgeleid. Ten slotte wordt een idee gegeven van wat de schakelprogramma's voor HBO-ers waarschijnlijk in zullen gaan houden.

Trefwoorden

Software Engineering, Curriculum ontwerp, 1-jarige masters, open afstandsonderwijs.

Ontwerp van een 1-jarige Master Software Engineering aan de Open Universiteit

1 **Achtergrond**

Het plan om aan de Open Universiteit een 1-jarige Master Software Engineering op te starten is in eerste instantie voortgekomen uit de behoefte van OU-studenten die het volgen van een 2-jarige academische Master Computer Science met behulp van afstandsonderwijs, veelal naast een volledige baan, een behoorlijke belasting vonden. Waar een 2-jarige opleiding voor afstandstudenten al gauw een reële studieduur oplevert van 5-8 jaar, kan een 1-jarige Master in 2.5 tot 4 jaar afgerond worden. Tevens bleek dat er ook maatschappelijk een grote behoefte is aan academisch opgeleide Software Engineers.

Uiteraard zou een dergelijke opleiding ook een beperktere scope moeten hebben dan de 2-jarige Computer Science-opleiding. De OU had reeds positieve ervaringen met de 1-jarige opleiding Business Process Management and Information Technology (BPMIT), een opleiding die samen met de faculteit Managementwetenschappen is opgezet. Naast BPMIT was er de behoefte aan een meer technische ingerichte opleiding. De keus voor Software Engineering lag voor de hand gezien de sterk op software gerichte onderzoek- en onderwijs expertise binnen de faculteit. Bovendien zijn er voorbeelden van academische 1-jarige Software Engineering opleidingen Leuven, Oxford en Amsterdam.

In eerste instantie is overwogen om samen met de Universiteit van Amsterdam een afstandsonderwijsvariant op te zetten van het door de UvA gebruikte materiaal voor hun 1-jarige opleiding Software Engineering. Het bleek echter dat het onderwijsmateriaal en de door de UvA gebruikte didactische methode (waarbij intensief contact tussen docent en student centraal staat) zich moeilijk lieten vertalen naar afstandsonderwijs. Vervolgens is besloten om de samenwerking met de UvA op een ander niveau voort te zetten. Studenten zullen wederzijds naar elkaar worden doorverwezen wanneer het contactmodel dan wel het afstandsmodel voor de student het meest geschikt is. Bovendien zal het vak Software Evolutie hetgeen het meest geschikt bleek voor afstandsonderwijs, in goed overleg voor afstandsonderwijs worden ontwikkeld door de OU gebruik makend van de Rascal tooling die ook aan de behoeften van de OU onderwijsleeromgeving zal worden aangepast.

2 **Uitgangspunten**

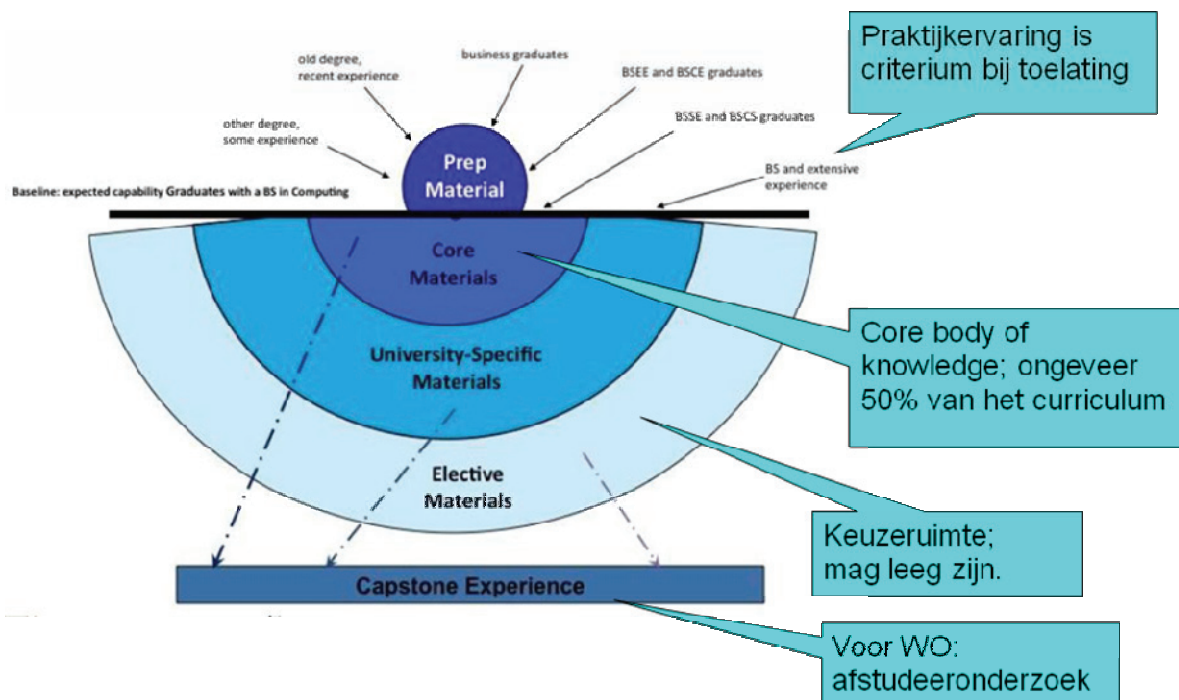
Uiteraard dient de opleiding geaccrediteerd te worden en dient de inhoud op academisch niveau te zijn en passend binnen de internationale kaders voor een opleiding Software Engineering. Uit het oogpunt van effectiviteit worden waar mogelijk cursussen uit de Master Computer Science hergebruikt. Verder is het voor de OU positief als uit de cursussen ook Certified Professional Programs (CPPs) samen te stellen zijn. Een CPP is een door de OU gebruikte onderwijsvorm voor bedrijven die gebaseerd is op een reeks regulier academische cursussen waarbij extra begeleiding gegeven wordt (soms 'in-house').

3 Kaders

De gebruikte kaders zijn die van de grote international Informatica vakorganisaties: ACM en IEEE, met name de Computer Curricula 2005 van ACM/IEEE-CS/AIS (eisen aan (Bachelor)-programma's in o.a. Computer Science en Software Engineering), Computer Science curricula 2008 van ACM/IEEE-CS ('body of knowledge') en de Curriculum Guidelines for Graduate Degree Programs in Software Engineering (GSWE2009; in 2010 overgenomen door ACM en IEEE) .

4 GSWE 2009 Opbouw curriculum

Dit laatste document, de GSWE2009, was het belangrijkste kader bij het ontwerp van de 1-jarige Master Software Engineering. Dit document geeft enerzijds aan hoe een Software Engineering curriculum in het algemeen moeten worden opgebouwd: met praktijkervaring als criterium voor toelating, met ongeveer de helft van het curriculum de body of knowledge, met eventueel keuzeruimte voor meer gespecialiseerde onderwerpen en met afstudeeronderzoek voor het wetenschappelijke onderwijs (zie Figuur 1).

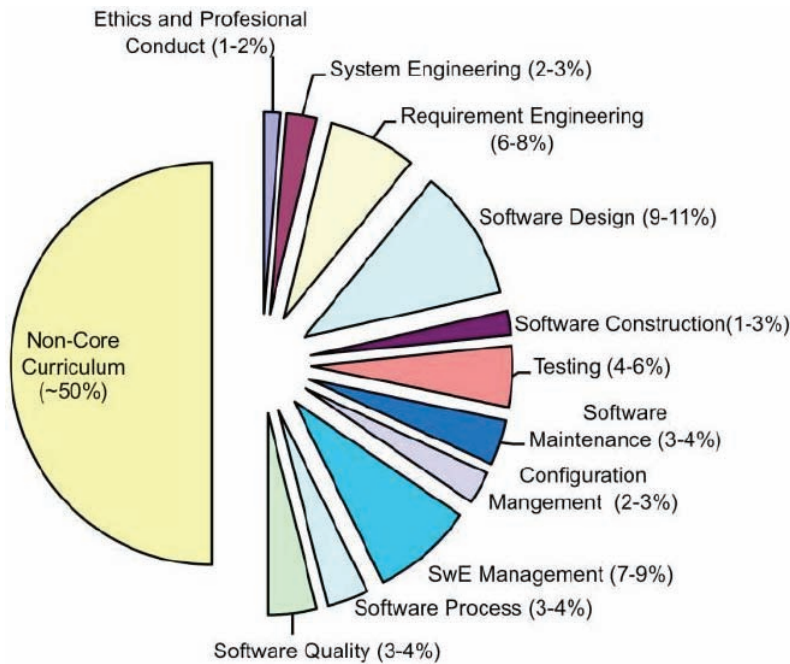


FIGUUR 1 Algemene Opbouw Curriculum Software Engineering volgens GSWE2009

5 GSWE Core body of knowledge

De Software Engineering body of knowledge (ook wel SWEBOK genoemd) is een erg uitvoerige lijst van onderwerpen waar aandacht aan besteed moet worden. Bij elk onderwerp is het percentage aangegeven wat de behandeling ervan bij voorkeur zou moeten zijn ten opzichte van het totaal van de behandeling van de hele body of knowledge.

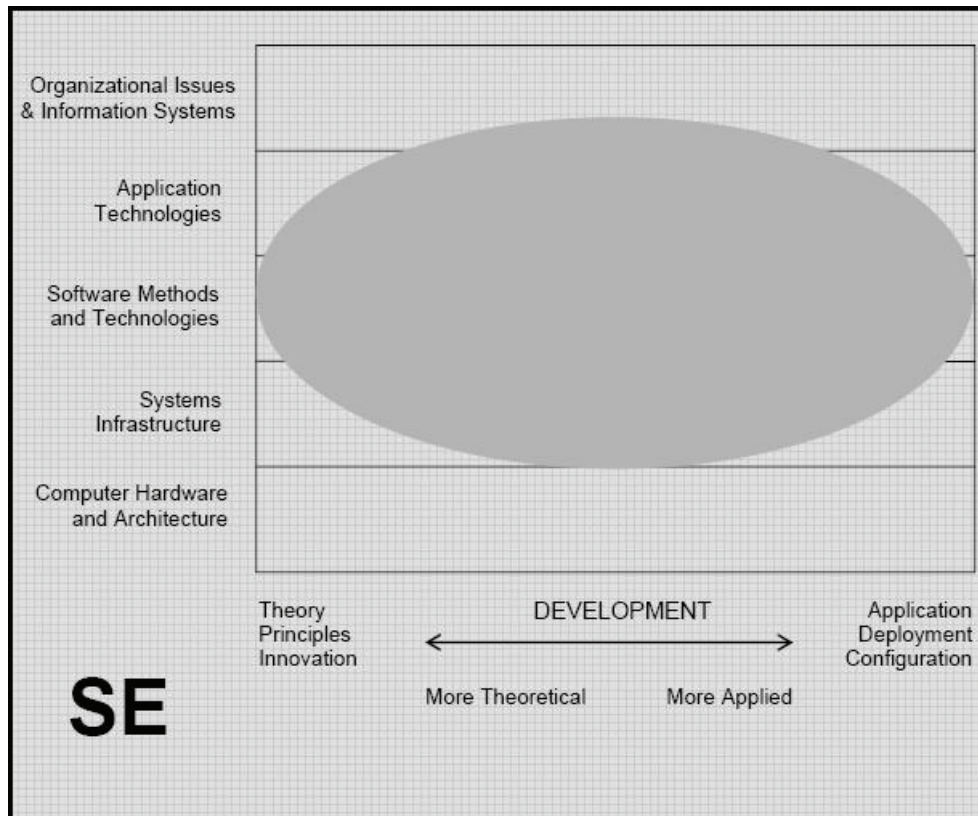
De uitvoerigheid van de lijst maakt het onvermijdelijk dat sommige onderwerpen slechts zijdelings aan bod komen of slechts even kort aangestipt worden. We hebben bij het ontwerp de lijst als een globale richtlijn beschouwd en waar de percentages hoog zijn ook meer aandacht gegeven in het curriculum. Er is echter geen exacte percentagegewijze correspondentie per deelonderwerp. Dat is ook niet het idee van de body of knowledge. Het gaat om globale richtlijnen waar in globale zin zo veel mogelijk aan voldaan dient te worden. In Figuur 2 is aangegeven hoe die percentages ongeveer verdeeld zijn.



FIGUUR 2 Verdeling percentages volgens Software Engineering Body of Knowledge

6 ACM2005 Karakterisering SwE

In de ACM 2005 Curriculum guidelines wordt het centrale karakter van Software Engineering in het Informatica vakgebied gevisualiseerd (Figuur 3). De figuur geeft aan dat het vakgebied Software Engineering op evenwichtige wijzen onderwerpen bevat variërend van theorie tot ontwikkeling enerzijds en van organisatie tot hardware anderzijds, waarbij de software methoden en technologieën centraal staan en de uitersten in sterk verminderde mate aan bod komen.



FIGUUR 3 Software Engineering in het Informatica vakgebied

7 Programma Globale Opbouw

Het programma van de OU Master Software Engineering is als volgt opgebouwd. Cursorisch onderwijs dekt de body of knowledge voor 34.2 ects. Dat zijn 8 vakken: een vak bij de OU is standaard 4.3 ects. De decimaal gegeven waarden zijn afrondingen.

Aan academische competenties wordt 4.3 ects besteed. Dit vak is specifiek bedoeld voor de ontwikkeling van academische competenties tot het niveau dat nodig is om te beginnen met het afstudeertraject. De nadruk ligt op de competenties noodzakelijk voor het zelfstandig bijhouden van het vak en het doen van onderzoek.

Aan inhoudelijke voorbereiding voor het afstuderen wordt ook 4.3 ects besteed. Dit is geen voorbereiding van algemene aard maar dit betreft het bestuderen van specialistische onderwerpen die voor het gekozen afstudeeronderwerp een noodzakelijke voorbereiding zijn.

Tot slot is voor het afstuderen zelf 17.3 ects ingeruimd. Dit sluit bij voorkeur aan bij het onderzoek van de Open Universiteit en het heeft een onderdeel waarin (eventueel in samenwerking met anderen) gebruiksoftware wordt ontwikkeld hetzij ten behoeve van de onderzoekers hetzij om resultaten van het onderzoek toe te passen in een bedrijfs-situatie. Hiermee is zowel het Software Engineering karakter als het academisch karakter gewaarborgd. In Figuur 4 is kort weergegeven hoe de opbouw van het programma is.

<i>Component</i>	<i>Omvang in ects</i>
Cursorisch onderwijs	34.2
Academische competenties	4.3
Voorbereiding afstuderen	4.3
Afstuderen	17.2
TOTAAL	60.0

FIGUUR 4 Globale Opbouw OU Master Software Engineering

8 Programma SE Cursussen

Inhoudelijk is het programma van 8 vakken opgedeeld in drie lijnen die elk focussen op een specifiek aspect: Software quality management, Software architecture en Software quality assurance. Elk van deze lijnen zou wellicht ook voor een CPP gebruikt kunnen worden.

Software quality management richt zich op de kwaliteit in de fasen van de software lifecycle en het managen daarvan zowel aan het begin bij het bepalen van de requirements als later wanneer de software evolueert. Dit is ingevuld met de vakken Software Evolution, Requirements Engineering en Software Management.

Software architecture richt zich op het hoog-niveau ontwerp van goed gestructureerde en goed onderhoudbare software. Dit betreft de vakken Design Patterns, Software Architecture en Component based Development.

Software quality assurance richt zich op (formele) technieken voor kwaliteitsbewaking en kwaliteitsbepaling zowel wat functionele als niet-functionele eigenschappen betreft. Hierbij ligt de aandacht op Software Verification en Validation en op Software Security.

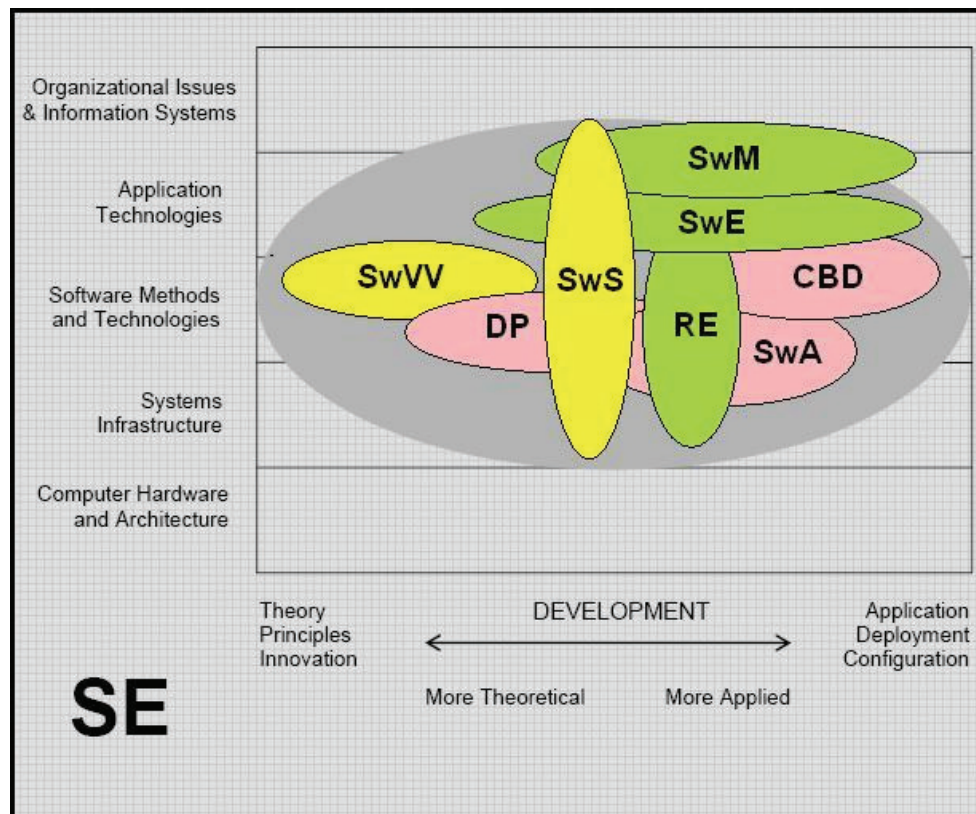
In figuur 5 zijn de aspecten en de bijbehorende vakken aangegeven.

<i>Aspect</i>	<i>Vakken</i>
Software quality management	Software evolution (SwE) Requirements engineering (RE) Software management (SwM)
Software architecture	Design patterns (DP) Software architecture (SwA) Component based development (CBD)
Software quality assurance	Software verification and validation (SwVV) Software security (SwS)

FIGUUR 5 Verplichte vakken in het OU Master Software Engineering Curriculum

9 OU Master SE in Kader ACM

Figuur 6 toont een ruwe positionering van de vakken uit onze opleiding binnen de ACM karakterisering van Software Engineering zoals die was aangegeven in Figuur 3. Elk aspect heeft daarbij een eigen kleur gekregen: groen voor quality management, roze voor software architecture en geel voor quality assurance. De figuur laat zien dat dit vakkenpakket het SE-gebied goed afdekt. Iedere lijn heeft bovendien zijn eigen deelgebied: software quality management zit vooral bovenin (application technologies), software architecture bestrijkt vooral het middengebied terwijl quality assurance duidelijk het meest theoretisch karakter heeft.



FIGUUR 6 Positionering van OU Master SE vakken in ACM karakterisering

10 Samenwerking andere universiteiten

Het Software Engineering programma is tot stand gekomen in samenwerking met andere universiteiten. Bij de Open Universiteit is het gebruikelijk om voor elk vak een externe referent aan te stellen die de inhoud en de uitwerking van het vak beoordeelt en daarmee garant staat voor de academische kwaliteit ervan. Dit is uiteraard ook bij de Master Software Engineering toegepast. Daarnaast zijn echter ook meer intensievere vormen van samenwerking benut bij de ontwikkeling van de Master Software Engineering.

Allereerst betreft dit natuurlijk de al eerder genoemde samenwerking met de Universiteit van Amsterdam op het gebied van doorverwijzen en Software Evolution en de Rascal Tooling van Professor Paul Klint. Bovendien betreft dit samenwerking met de Katholieke Universiteit Leuven (Requirements Engineering, gebaseerd op het boek van Professor van

Lamsweerde die ook als extern referent van de cursus optreedt) en de Universiteit Utrecht (Software Architecture gebaseerd op een vak wat aan de Universiteit van Utrecht werd gegeven door Professors Jeuring en Bijlsma) en de Radboud Universiteit Nijmegen (Software Verification and Validation waarbij materiaal van Professor Vaandrager gebruikt wordt die ook als extern referent optreedt en Software Security wat gebaseerd is op materiaal van Dr. Poll waarbij Professor Jacobs als extern referent optreedt).

Een overzicht van de samenwerking met andere universiteiten bij de verschillende vakken is gegeven in figuur 7.

<i>Aspect</i>	<i>Vakken</i>
UVA Amsterdam (Klint) KU Leuven (van Lamsweerde)	Software evolution (SwE) Requirements engineering (RE) Software management (SwM)
U Utrecht (Jeuring, Bijlsma)	Design patterns (DP) Software architecture (SwA) Component based development (CBD)
RU Nijmegen (Vaandrager) RU Nijmegen (Poll, Jacobs)	Software verification and validation (SwVV) Software security (SwS)

FIGUUR 7 Overzicht van OU Master SE vakken met samenwerking met andere universiteiten

11 Programma Academische Competenties

De academische competenties nodig om te beginnen met het afstudeertraject worden aangebracht in 1 vak wat verdeeld is over drie verdiepingsopdrachten bij drie verschillende vakken. Deze verdiepingsopdrachten maken geen onderdeel uit van die vakken maar ze sluiten er wel nauw op aan. Op deze manier staat de academische competentie niet los van de inhoud maar wordt die nadrukkelijk aangeleerd in de context van de academische inhoud. Dit is als volgt verdeeld:

Design patterns	zoeken van literatuur schrijven argumenteren
Software evolution	destilleren onderzoeksvraag uit artikel destilleren onderzoeksmethode uit artikel reflectie op verzamelen van gegevens
Verification and validation	destilleren onderzoeksmethode uit artikel zelf formuleren onderzoeksvraag (vervolgvraag bij artikel)

12 Weg naar goedgekeurde opleiding

De voorbereiding heeft bijna 2 jaar gekost. De doelmatigheidsaanvraag is eind december 2010 ingediend. De zusterfaculteiten zijn om advies gevraagd en op dit moment (eind mei 2011) zijn we in afwachting van een besluit hierover. Alle voorbereidingen voor de accreditatiedocumenten zijn al getroffen zodat de accreditatieaanvraag in de vorm van een beperkte Toets Nieuwe Opleiding direct ingediend kan worden. Onder voorbehoud van verlening van de accreditatie, kan de opleiding naar verwachting eind 2011 van start gaan.

13 Toelatingsprocedure

De toelating tot de opleiding zal vereisen dat iemand minimaal HBO-Bachelor niveau heeft op Informaticagebied of HBO-Bachelor op ander gebied plus drie jaar recente werkervaring op het niveau van een HBO-Bachelor Informatica.

Toelating zal plaatsvinden via een gesprek met een toelatingscommissie en op grond van een door de student samengestelde portfolio. Dit kan leiden tot het vaststellen van een schakelprogramma om ontbrekende kennis aan te vullen. Die ontbrekende kennis kan nodig zijn om het programma te doorlopen (als voorkennis van een van de vakken) of om aan de algemene academische eindtermen op Master niveau te voldoen.

Voorbeelden van schakelonderwerpen kunnen bijvoorbeeld zijn: kennis van predikaten-logica en contextvrije grammatica's die nodig is om het programma te doorlopen en kennis van databases is nodig om de eindtermen te halen aangezien van een Software Engineer basiskennis van databases verwacht mag worden.

Door de Facultaire toetsingscommissie van de Faculteit Informatica van de Open Universiteit zijn standaardschakelprogramma's vastgesteld die periodiek afgestemd worden met de actuele curricula van de betreffende opleidingen:

- Voor een student met een afgeronde wo-bachelor-opleiding Technische informatica, technische afstudeerrichting of een wo bacheloropleiding Informatica van de Open Universiteit en voor studenten met de graad scientific bachelor Informatica afkomstig van een van de universiteiten die in 2002 deelnamen aan de visitatie Informatica, is het standaardschakelprogramma leeg. Dit geldt ook voor afgestudeerden met een wo-opleiding Informatica (doctorandusopleiding) of het Vlaamse (universitair) licentiaat Informatica. Deze studenten komen wellicht in aanmerking, afhankelijk van vakkenpakket en specialisatie, voor additionele vrijstellingen in de Master.
- Voor afgestudeerden met een opleiding hbo Informatica, of het Vlaamse equivalent van industrieel ingenieur Informatica, geldt een standaardschakelprogramma van 8,6 studiepunten: Logica en informatica en Formele talen en automaten.
- Voor studenten met een vooropleiding wo bachelor (Technische) informatica, bedrijfskundige afstudeerrichting geldt een standaardschakelprogramma van 13,2 studiepunten: Concepten van programmeertalen, Logica en informatica en Formele talen en automaten.
- Voor studenten met een vooropleiding hbo Technische informatica, hts-e/tck, of de Vlaamse equivalenten van industrieel ingenieur Elektronica en Elektrotechniek, geldt een standaardschakelprogramma van 30 studiepunten: Concepten van programmeertalen, Databases, Objectgeoriënteerd analyseren en ontwerpen, Objectgeoriënteerd programmeren in Java 2, Software engineering, Logica en informatica en Formele talen en automaten.
- Voor studenten met een vooropleiding hbo Bedrijfskundige informatica die na 31 mei 2004 zijn afgestudeerd, of de Vlaamse equivalenten daarvan, geldt een standaardschakelprogramma van 25,8 studiepunten: Concepten van programmeertalen, Datastructuren en algoritmen, Objectgeoriënteerd programmeren in Java 2, Software engineering, Logica en informatica en Formele talen en automaten.

De standaardschakelprogramma's die hierboven vermeld zijn, vormen het uitgangspunt voor het gesprek met de toelatingscommissie. Op grond van het cv, het gesprek en het bewijsmateriaal wordt het persoonlijke schakelprogramma vastgesteld. *Dit kan zowel een uitbreiding als een inkorting van het standaardschakelprogramma betekenen, afhankelijk van de persoonlijke achtergrond van de student.*

14 **Samenvattend**

De opleiding Software Engineering aan de Open Universiteit is een 1-jarige universitaire Master met een op ACM/IEEE kaders gebaseerd programma, met studiemateriaal dat voor een deel in overleg met andere universiteiten (Amsterdam, Utrecht, Nijmegen, Leuven) is ontwikkeld, waarbij een groot deel van de opleiding naar verwachting ook via bedrijfs-cursussen (CPPs) te volgen zal zijn en die onder het voorbehoud van het verlenen van accreditatie naar verwachting eind 2011 zal kunnen starten.