



Het Nationale Informatica Onderwijs Congres (NIOC) 2009

ONTMOETINGSPLAATS VOOR ICT-ONDERWIJS

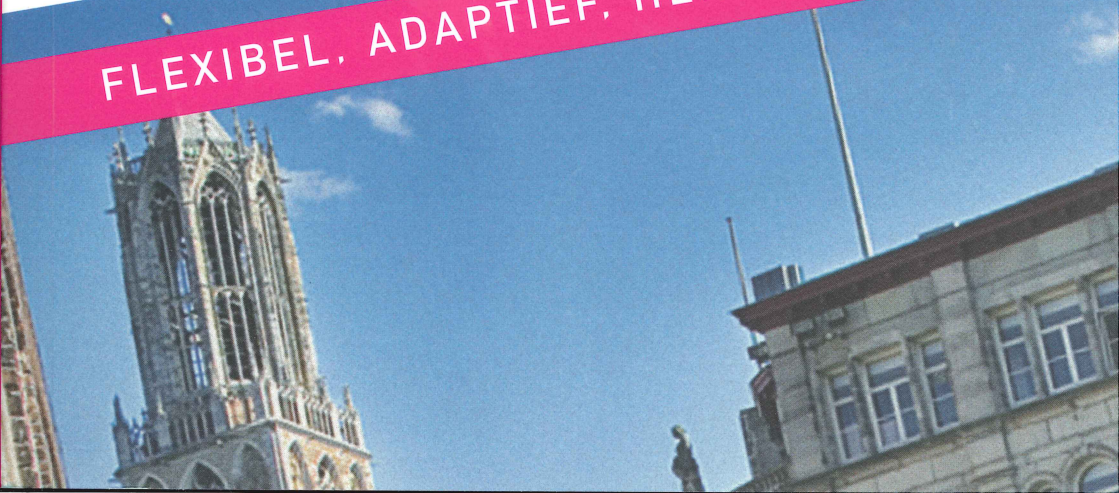
NIOC PROCEEDINGS

7 en 8 april 2009, Utrecht



Frans Vodegel en Marijke Loots, editors

FLEXIBEL, ADAPTIEF, HERBRUIKBAAR



Het Nationale Informatica Onderwijs Congres (NIOC) 2009

ONTMOETINGSPLAATS VOOR ICT-ONDERWIJS

NIOC PROCEEDINGS

7 en 8 april 2009, Utrecht

Frans Vodegel en Marijke Loots, editors



FLEXIBEL, ADAPTIEF, HERBRUIKBAAR



Inhoudsopgave

NIOC Proceedings 2009

5	Inleiding	109	16. Kids in control as creative designers, <i>Nienke Valkhoff, Nardie Fanchamps.</i>
7	1. Agile: een must voor het IT-onderwijs <i>Alexander Jongman, Nini Salet</i>	115	17. Kansen voor het e-book in het hoger onderwijs, <i>Frans Mofers</i>
12	2. Leerbedrijf WOSI biedt studenten professionele werkomgeving, <i>Jo Lahaye</i>	120	18. Onderwijssamenwerking in Business Process Management, <i>Johan Versendaal, Pascal Ravesteyn</i>
19	3. Testen geïntegreerd in het curriculum software engineering, <i>Ferd van Odenhoven, Pieter van den Homberh</i>	128	19. Het SoccerFun project: functioneel programmeren in het onderwijs met behulp van voetbal, <i>Peter Achten</i>
28	4. Aansluiting hbo en wo op instroomopleidingen Info Support, <i>Gert Jan Timmerman</i>	135	20. Een module over Model Checking voor het vwo, <i>Frits Vaandrager, David Jansen, Els Koopmans</i>
32	5. Media en ICT: gescheiden of toch liever een latrelatie? <i>Laura Benvenuti, Geeske Bakker, Gerrit C. van der Veer</i>	138	21. Universeel systeem voor bedrijfsvoering, <i>Deny Smeets, Arie Schinkel</i>
38	6. Bedrijven als 'Partners In Education' in het onderwijs van Fontys Hogeschool ICT <i>Frans van Alebeek, Ruud Ermers</i>	148	22. Hoe denken we er over? Informatica in het voortgezet onderwijs, <i>Nico van Diepen</i>
44	7. Een virtueel computersecuritylab <i>Harald Vranken, Herman Koppelman</i>	154	23. Taal- en spraaktechnologie, <i>Marijke Loots</i>
52	8. Software architectuur in het afstandsonderwijs, <i>Bastiaan Heeren, Sylvia Stuurman</i>	159	24. Instructional design for Java enterprise component technology, <i>Marco Marcellis, Ella Roubtsova, Bert Hoogveld</i>
63	9. How Industry Participates in Enterprise Resource Planning Education, <i>Pascal Ravesteyn, Adri Köhler</i>	162	25. Personal Branding, <i>Eric Slaats, Judith Gaffert</i>
73	10. Platform Beeldverwerking HBO, <i>Jaap van de Loosdrecht</i>	168	26. Europese samenwerking in het hoger ict-onderwijs, <i>Boudewijn Bouwense.</i>
76	11. Het Delta excellence programma. Bron: TINFON, 2008, nummer 4, <i>Eric Slaats</i>	172	27. Augmented reality voor gebruik bij toerisme, <i>Mohamed Abdelghany, Adrie van der Padt en Glenn Mosdall</i>
85	12. UvA Webklas Informatica: Wat is een Programma? Alban Ponse, <i>Stephan Schroevers</i>		
93	13. Werken 2.0: Van vak naar vaardigheid, <i>Peter Becker, Jan Krans</i>	176	Stichtingsbestuur NIOC - Stuurgroep NIOC 2009 -
97	14. Herbouw van legacy systemen: een onderbelicht onderwerp in het informatica onderwijs, <i>Herman Pootjes</i>	176	Programmacommissie NIOC 2009
		177	Sponsors
		178	Colofon
100	15. Grootschalig gebruik van versiebeheer in informatica onderwijsprojecten, <i>Jan Derriks, Ahmed Nait Aicha, Jacob Brunekreef</i>	179	Sorry....vuurtje

Inleiding



‘Flexibel - Adaptief - Herbruikbaar’ In juni 2008 vroeg de stuurgroep van het Nationale Informatica Onderwijs Congres (NIOC) aan de programmacommissie om een congresprogramma op te stellen waarin dit motto tot uitdrukking kwam. Op basis daarvan heeft de commissie een aantal actuele onderwerpen geformuleerd als suggesties voor congresbijdragen.

Deze onderwerpen werden gegroepeerd in vier thema’s. Om te beginnen het thema **samenwerking binnen de onderwijskolom**: hoe worden de kansen tussen vo/mbo/hbo/wo beter benut? Dit kan bijvoorbeeld door *dienstverlenende projecten* van het hbo voor het vo; door *good practices* beter te etaleren; door het organiseren van *voortrajecten binnen het basisonderwijs* voor ICT-opleidingen (zoals Java), door *onderwijsmodules in het vo door hogescholen* als kennismaking met ICT en bijvoorbeeld ook door middel van *wedstrijden binnen het bedrijfsleven als onderwijsproject*.

Een ander aspect had betrekking op **ontwikkelingen op het gebied van media & ICT**. Een voorbeeld dat in de commissie werd genoemd was *virtuele werelden voor het onderwijs*. Zo ook *gaming en ICT & Media*. Het derde thema behelsde de **onderwijskundige aspecten**. Bijvoorbeeld *werkend leren in het mbo als prestatieonderwijs en onderwijskundige profielkleuring*. En: *interculturele competenties in de ICT wereldwijd, wat kan er worden geleerd van dertig jaar ICT-onderwijs en internationalisering?*

Ten slotte werden aspecten genoemd die in verband staan met **nieuwe technologische ontwikkelingen** als *bio-informatica* (biologische systemen) en *security*. Of wat te denken van de *Segway*?

Uitgaande van deze thema’s stelden de verschillende deelcommissies het programma samen, en maakten ze hun achterbannen enthousiast om een bijdrage te leveren. Het resultaat was een congres met ruim negentig presentaties, en met deelnemers uit alle sectoren van het onderwijs.

De congresdagen op 7 en 8 april in Utrecht waren twee interessante dagen vol ideeën over het informaticaonderwijs. Een aantal punten viel op:

- De blik van de samenleving op het vak informatica moet verruimd worden. Het vak zou veel meer gezien moeten worden vanuit de functie die informatietechnologie (IT) heeft in de samenleving en niet als een vakgebied van technici die verstaan van wat er in die samenleving gebeurt.
- De zorg over de geringe studenteninstroom in het informaticaonderwijs blijft bestaan. Er moet nodig gewerkt worden aan een andere beeldvorming.
- Serious gaming (games om kennis en vaardigheden te verwerven) wordt een belangrijke trend in de komende jaren.
- Internationalisering van het informaticaonderwijs blijft ver achter bij andere onderwijsdomeinen.

Wij hebben alle sprekers uitgenodigd een artikel te schrijven naar aanleiding van de gepresenteerde bijdragen, ten behoeve van de proceedings. Na selectie en review kunnen wij u die artikelen nu aanbieden. Wij wensen u veel leesplezier.

Namens Programmacommissie NIOC 2009,
Frans Vodegel en Marijke Loots, editors

Agile: een must voor het IT-onderwijs



Alexander Jongman - curriculumcoördinator opleiding Software-ontwikkeling, Instituut voor ICT, Hogeschool Utrecht

Nini Salet - hoofddocent Information Engineering, Instituut voor ICT, Hogeschool Utrecht.

SAMENVATTING

Agile-softwareontwikkeling is een conceptueel raamwerk voor het uitvoeren van software-ontwikkelingsprojecten als alternatief voor traditionele starre methoden. Agile is niet alleen in het bedrijfsleven een hot item. Ook in het onderwijs wordt het steeds belangrijker dat studenten en docenten ermee werken. Aan het Instituut voor ICT van Hogeschool Utrecht is bij de opleiding Information Engineering (IE) geëxperimenteerd met Agile. Dit is in zeer goede aarde gevallen. Studenten krijgen regelmatig summatieve feedback op het resultaat van hun werk. Feedback die nodig is om bij een volgende iteratie tot betere samenwerking en oplossingen te komen.

De Agile-ontwikkelmethodiek moet ingebed zijn in het IE-curriculum. Studenten leren specialistische rollen bekleden, zijn afhankelijk van elkaars werk en brengen in overleg wederzijds verbeteringen aan. Binnen het project beoordelen studenten elkaar op uit te voeren taken. Daarnaast beoordeelt een college van vakdocenten de individuele en groepsprestaties in tweewekelijks terugkerende assessments. Studenten kunnen leren wat teamwork is en nemen een proactieve houding aan ten opzichte van de opdracht.

De beoordeling van projectgroepen betreft de Agile-werkmethode en het resultaat ervan. Docenten wisselen onderling kennis en ervaring uit, want ook zij streven naar verbetering en vernieuwing.

TREFWOORDEN

Agile, Teamwork, Curriculum, Projecten, Kwaliteit

INLEIDING

Projectonderwijs kent een aantal nadelen die met een Agile-aanpak deels opgelost kunnen worden. Structurele problemen van het traditionele projectonderwijs, zoals het meeliftgedrag van studenten en/of de motivatieproblematiek om in projectteams te werken, neemt een Agile-methode zelf ook niet weg. Maar door aan een Agile-werkmethode rollen met eigen deliverables toe te voegen en een authentieke en complexe opdracht te kiezen is dit wel mogelijk. Bij de opleiding Information Engineering (IE) is dit toegepast en daardoor is het projectonderwijs flink veranderd. Dit artikel geeft een overzicht van problemen die zich aandienen, hoe het curriculum hiervoor is aangepast, hoe de structuur van de projecten is ingericht en welke ervaringen en conclusies hieruit getrokken worden.

HET PROBLEEM: DE SITUATIE ZOALS DIE WAS

Het oude curriculum van de opleiding Information Engineering bevat meerdere voor de student onsamenhangende vakken. Projecten draaien om simulaties, opdrachten zijn gebaseerd op fictieve bedrijfssituaties, waarbij een docent de verschillende

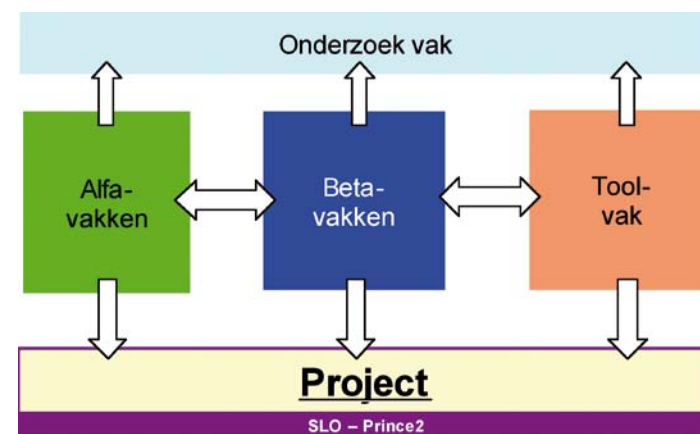
belanghebbenden vertegenwoordigd. De docent beschikt aldus over goede theoretische en didactische vaardigheden evenals over inlevingsvermogen en soms theatrale vaardigheden. Studenten worden voor het uitvoeren van deze practica vaak in teams ingedeeld. Hierbij valt op dat de teamvorming te laat, soms pas vlak voor de beoordeling echt op gang komt. Er is gelegenheid tot uitstelgedrag vanwege het gebrek aan leernood. Omdat alle teamleden alle taken kunnen verrichten voelt niemand zich aangesproken of verantwoordelijk om het initiatief te nemen. Het werk komt vooral neer op de plichtsgetrouwe en ijverige studenten. De rapportage of documentatie gebeurt door een reeks van schrijvers die elkaars stukken niet hoeven te lezen en het dan ook niet doen. Een lappendeken of stoplap, een rapport zonder rode draad met een overdaad aan redundantie. Een document dat de denkbeeldige gebruiker, ook een rol van de docent, niet aanzet tot service en onderhoud van de applicatie, het eindproduct.

HET NIEUWE CURRICULUM: ZOALS DE SITUATIE NU IS

Het vernieuwde curriculum van de opleiding IE bestaat uit semesterprogramma's. Het project staat in dit semester centraal. Studieloopbaanondersteuning en -begeleiding (SLO/SLB) is in het project geïntegreerd doordat onvoldoende resultaten onmiddellijk gecoacht worden. Niet de individuele studenten maar de situatie wordt geanalyseerd om tot verbeteracties te komen, die de eerstvolgende beoordeling zichtbaar moeten zijn. Blijft verbetering achterwege? Dan worden rollen explicieter beschreven en gecoacht waarbij projectleiders steeds gedetailleerder gaan plannen en bijsturen. Het bijsturen gebeurt met behulp van PRINCE2 deliverables zoals een PBS, PFD en (netwerk)planning. Gedetailleerd voor de korte termijn, grofmazig voor wat erna komt. Zodra de resultaten van de beoordeling voldoende zijn, verdwijnt de coaching door de SLO'er. Zie het als een tijdelijke dubbele besturing van een auto

tijdens het rijexamen. De coaching is een doelgerichte ingreep om onvoldoende resultaat te voorkomen.

Studenten krijgen de kennis voor het werken in het project zo veel mogelijk Just In Time (JIT) aangeleverd in alfa-, bèta- en toolvakken. Deze theorie wordt individueel getoetst en toegepast voor de oplossingen. Studenten die behoefte hebben om zich specifieke kennis eigen te maken doen dit tijdens het vak Onderzoek. De probleemstelling van het onderzoek kan per projectteam verschillen. Schematisch ziet een semester eruit als in figuur 1.

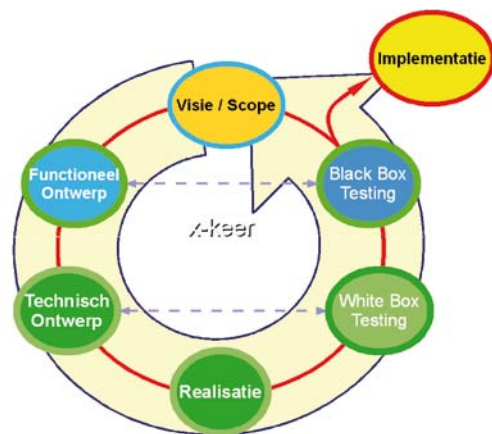


Figuur 1:
IE Semester structuur.

PROJECTAANPAK IN HET VERNIEUWD CURRICULUM

Het semester moet nu een 'real' project bevatten. Dit project draagt bij aan de oplossing van een bestaand probleem. Studenten staan in contact met echte gebruikers. Dit hoeven geen projecten voor externe bedrijven te zijn. Het kunnen opdrachten zijn die bijdragen aan oplossingen voor of verbeteringen van bedrijfsprocessen op de Hogeschool: voor de studenten herkenbaar omdat zij tot de gebruikersgroepen behoren. Denk aan interne problemen, zoals een aanwezigheidssysteem of een systeem ter ondersteuning van de stage- en afstudeer-administratie. Door een probleemanalyse ontdekken studenten dat de inhoud en focus door henzelf kan worden gedefinieerd. Dat zij als team een uniek product kunnen opleveren zonder dat een persoon het voortouw moet nemen. Zonder dat enkele mensen het leeuwendeel van het werk verzetten. Een team bestaat

uit vier tot zeven studenten en de volgende rollen: Analyst, Functioneel Ontwerper, Technisch Ontwerper, Development-Lead, Tester of Projectleider. Een student solliciteert aan het begin van het semester naar de rol die hij ambieert. In figuur 2 zijn de afzonderlijke rollen en het iteratief proces gevisualiseerd.



Figuur 2:
Methode en iteratie, ISD 2008.

Elke rol heeft een eigen set aan deliverables, waarop de betreffende student door zijn collega's aangesproken wordt. Voor de deliverables geldt de eis dat ze van een zodanige kwaliteit zijn dat het (deel)product overdraagbaar is aan de onderhoudsorganisatie en/of een andere eigenaar van de oplossing. Voor elk project wordt een coach uit het bedrijfsleven gezocht. Deze krijgt de positie van senior adviseur. De student kan bij hem terecht met vragen over zijn specifieke rol en de op te leveren deliverables. In figuur 2 is te zien dat het gaat om een Agile-benadering, er wordt iteratief gewerkt.

Vijf iteraties worden aan het eind afgesloten met een assessment waarbij docenten (die ook de kennisvakken verzorgen) deelproducten beoordelen. Elke beoordelingsronde bestaat uit het bespreken van het werk, de rollen en afhankelijkheden. Dit resulteert in een individueel en een teamcijfer. Het gemiddelde van deze twee cijfers is het resultaat van de iteratie. Er zijn 5 beoordelingen. In iedere iteratie loopt de weg

tussen de individuele beoordeling en die van het team op van 5%, 10%, 15%, 30% en 40%. Beoordelaars zien aan de cijfers of een individu of soms een heel team niet goed samenwerkt. De SLO-begeleider wordt dan ingeschakeld en nodigt het betreffende team uit voor een gesprek.

Studenten reflecteren op de rollen en geven elkaar feedback. Dit mondt uit in een herziening van doelen en een kortetermijnplanning, die binnen een week tot betere resultaten en samenwerking moet leiden. Zo niet dan herhaalt deze procedure zich.

ERVARINGEN EN CONCLUSIES

Deze aanpak werkt erg goed in het IT-onderwijs, zeggen studenten, docenten en externe bedrijfsbegeleiders. Het is voor allemaal erg wennen aan deze aanpak omdat fouten direct tot rollen en dus personen te herleiden zijn. Toch staat het product en niet de persoon centraal en wordt een gezamenlijk product opgeleverd. De beoordelingsmomenten fungeren als een dashboard met meters. Daaraan is te zien dat studenten tot de eerste twee iteraties nog niet werken als een team. Hun werk is individualistisch waardoor de samenhang tussen de deliverables ontbreekt. Er bestaat geen evenwicht tussen de ontwikkelde software en de documentatie. De software is ad hoc getest zonder van het functioneel ontwerp uit te gaan. Studenten werken checklists af zonder na te denken wat het doel en wie de gebruiker is van de oplossing. Zonder erbij stil te staan dat het document leesbaar moet zijn voor de gebruiker.

Regelmatig terugkerende beoordelingsmomenten en gedetailleerde feedback aan zowel groepsleden als het team bespoedigen verbetering van resultaten. Na de tweede iteratie is de verbetering zichtbaar en is bij de meesten het kwartje gevallen.

We concluderen dat studenten aan het einde van het semester als team werken, dat ze kritischer naar elkaars werk kijken en feedback durven te geven. Belangrijk is wel dat de docenten niet de druk op het aantal gerealiseerde features/de realisatie zelf leggen, maar op de balans van de verschillende deliverables. De student in de rol van projectleider is er over het algemeen nog niet aan toe om de spagaat te maken tussen de druk van de opdrachtgevers en die van de deskundigheid van het team.

Didactische voordelen zijn dat studenten leren wat teamwork is en ervaren dat meeliften niet kan. Studenten zijn erg tevreden dat zij zich in een rol kunnen verdiepen. Omdat de opdracht in het begin bewust vaag wordt gesteld, ontkomen ze niet aan een probleemanalyse en het gezamenlijk vaststellen van de scope.

Intern kent deze aanpak ook enkele voor- en nadelen. De voordelen (kwaliteitsverbetering) wegen zwaar. Formatietechnisch is het erg duur, omdat alle aan het project gerelateerde vakdocenten tevens projectbegeleiders zijn. Het voordeel hiervan is dat er kennisoverdracht tussen docenten onderling kan plaatsvinden, dat vakken beter op elkaar afgestemd worden en dat docenten inzicht krijgen of de onderwezen kennis door de studenten ook daadwerkelijk goed wordt toegepast. Een nadeel is dat er vlak voor de beoordeling een piekbelasting ontstaat bij beoordelaars. Tussen het inleveren en de feedback dient zo weinig mogelijk tijd verloren te gaan. Soms is het erg lastig om een authentieke opdracht te vinden, die binnen de gestelde kaders valt en goed aansluit bij de kennisvakken. Dit is tot nu toe altijd nog gelukt.



Leerbedrijf WOSI biedt studenten professionele werkomgeving

Jo Lahaye - projectmanager WOSI
J.Lahaye@hva.nl

SAMENVATTING

Het Woningcorporatie Open Standaarden Initiatief (WOSI) biedt hbo-studenten een professionele werkomgeving, waar leren net zo belangrijk is als presenteren. Het initiatief is gestart bij de Hogeschool van Amsterdam, maar inmiddels doen ook studenten van andere hogescholen mee, zoals Saxion uit Enschede. In de kern is WOSI een langlopend ICT-project, waar professionele (technische) projectleiders studenten voorbereiden op het in multidisciplinaire teams werken aan complexere, grote, projecten.

TREFWOORDEN

Open standaarden, open source software, leerbedrijf, woningcorporaties, HVA, Saxion, ICT.

INLEIDING

WOSI is een ICT-project voor studenten in het hoger onderwijs dat in 2006 geïnitieerd is door vier woningcorporaties (Servatius, Woonbron, Ymere en De Woonplaats) en de informaticaopleiding van de Hogeschool van Amsterdam. De doelstelling van WOSI is tweeledig:

- een professionele en realistische werkomgeving bieden, gericht op leren en de ontwikkeling van de student;
- een concrete bijdrage leveren aan de kwalitatieve verbetering van de informatiehouding van de corporaties en (maatschappelijke) instellingen in de informatieketen, zoals gemeenten en zorginstellingen.



WOSI wil uitgroeien tot een kennisinstituut dat straks een scala aan onderzoeksresultaten, een voorbeeldarchitectuur en een concrete toolset levert om technologie en gegevens daadwerkelijk te kunnen (her-)gebruiken en integreren. Om die ambitie waar te maken staat WOSI open voor participatie door meer corporaties, gemeenten, zorginstellingen en zeker ook voor onderwijsinstellingen. Daarvoor geldt bovendien dat het project ook interessante stage- of afstudeermogelijkheden biedt voor andere studies zoals rechten, communicatie, vormgeving, interactiondesign en dergelijke. Hieronder wordt eerst ingegaan op de doelstelling van het project, vervolgens op de (technische) inhoud van het project om af te sluiten met de ervaringen van studenten en de opleiding.

WAAROM WOSI

Op dit moment kenmerkt de informatiehouding van corporaties en hun maatschappelijke partners zich door veel branchespecifieke en gesloten systemen, waardoor samenwerking en het delen van informatie moeizaam verlopen. De corporaties constateren dat het gebruik van open standaarden goed past in het beleid van de overheid; open standaarden hebben in andere sectoren geleid tot meer efficiency, snellere ontwikkeling en innovatie. Daarbij wordt gewezen op diverse succesvolle XML-standaarden alsook op het internet dat gebaseerd is op open standaarden en daar zijn grote succes aan te danken heeft. WOSI moet uitgroeien tot een kennisinstituut dat straks een scala aan onderzoeksresultaten, een voorbeeldarchitectuur alsook een concrete toolset levert om technologie en gegevens daadwerkelijk te kunnen (her-)gebruiken en te integreren. Om die ambitie waar

te maken, staat WOSI open voor participatie door meer corporaties, gemeenten, zorginstellingen en zeker ook voor onderwijsinstellingen. Daarvoor geldt bovendien dat het project ook interessante stage- of afstudeermogelijkheden biedt voor andere studies zoals rechten, communicatie, vormgeving, interactiondesign en dergelijke. Maar ICT blijft de basis van het project, waarin gewerkt wordt met 'state of the art' open source software als Spring MVC, Hibernate, AJAX, en diverse J2EE componenten.

DOELSTELLING

Getracht wordt een moderne service-georiënteerde informatiearchitectuur (SOA) te ontwerpen, om de ICT-omgeving van corporaties en organisaties in de keten te verbeteren en flexibeler te maken. Hoofddoel is om te bezien of open technologie leidt tot hogere flexibiliteit en of het mogelijk is een referentiearchitectuur en een open standaard te ontwikkelen. De ontwikkeling van zo'n standaard vanuit een theoretisch model is zeer complex. Daartoe is afstemming nodig met alle corporaties en alle organisaties in de keten. Dat zijn langdurige trajecten die zonder dwang of hoge financiële voordelen moeilijk tot een goed einde te brengen zijn. Bovendien biedt de ontwikkeling van een dergelijke standaard sec geen garantie voor flexibiliteit in de applicatieontwikkeling. Vandaar dat het WOSI-project in de haalbaarheidsstudie naar meer flexibiliteit en betere gegevensuitwisseling twee insteken heeft:

1

Er wordt een aantal prototypen ontwikkeld op basis van het afgesproken framework (open en modulaar), waarbij gebruik wordt gemaakt van een nieuwe databasestructuur, met gegevens uit bestaande databases. De structuur van deze nieuwe database wordt dus in eerste instantie bepaald door de gegevens die voor de specifieke modules noodzakelijk zijn.

2

Vervolgens wordt geprobeerd tot een meer universeel datamodel te komen, dat naar de huidige inzichten zal ontstaan op basis van drie invloeden: studie naar de gegevensopbouw zoals beschreven door een aantal corporaties, op basis van de gegevens in de bestaande databases (want die moeten uiteindelijk migreerbaar zijn) en op basis van de gegevens die voor de applicaties nodig zijn.

MIDDELEN

De betrokken onderwijsinstellingen zorgen voor de beoordeling en de begeleiding van de studenten in het project, uiteraard in overleg met de projectleiding van WOSI. De projectleiding van WOSI kan in dit opzicht worden gezien als bedrijfsbegeleider. Met de docenten wordt doorlopend gesproken over verdere verbeteringen van het project en de manier waarop project en onderwijs in de toekomst beter op elkaar kunnen aansluiten. De participerende corporaties (Ymere/Amsterdam, Servatius/Maastricht, De Woonplaats/Enschede en Woonbron/Rotterdam) zorgen ervoor dat er voldoende middelen zijn voor projectmanagement en om de studenten ook technisch professioneel te kunnen begeleiden. Naast gastcolleges door vooraanstaande deskundigen van bijvoorbeeld IBM wordt ook gezorgd voor praktische zaken als boeken, ICT-middelen, hands-on workshops en dergelijke. WOSI is dus een echt bedrijf, met twee winnaars: het onderwijs en de corporaties.

WIN-WIN: CORPORATIES EN HUN OMGEVING

De ICT-leveranciers die actief zijn in de sector hebben allemaal hun eigen specifieke oplossingen, wat het moeilijk maakt om onderling gegevens uit te wisselen en bijvoorbeeld bij fusies snel te kunnen kiezen voor één systeem. Corporaties zijn non-profitinstellingen die zich richten op het beschikbaar stellen van betaalbare woningen. ICT is geen doel op zich, maar een hulpmiddel. De overwegingen van de corporaties om in

WOSI te participeren zijn:

- bevorderen van uitwisselbaarheid van software en gegevens (data);
- verbeteren van de kwaliteit en flexibiliteit van ICT-systemen en mogelijke kostenbesparing;
- vermindering afhankelijkheid van ICT-leveranciers (die straks kunnen aanhaken bij WOSI);
- meer ruimte voor nieuwe initiatieven en innovatie;
- meerwaarde creëren in de informatieketen (gemeenten, zorg) en voor collega-corporaties;
- maatschappelijke betrokkenheid (op samenwerking gericht open source model past goed bij karakter van corporaties, en ondersteuning van het beroepsonderwijs);
- studenten verwerven kennis en raken mogelijk geïnteresseerd in (de ICT-omgeving van) woningcorporaties.

WIN-WIN: ONDERWIJS

Ook voor het onderwijs zijn er goede motieven om in dit project te participeren:

- studenten leren in een 'echte' omgeving, met een klant die serieus op zoek is naar innovatieve oplossingen.
- Het is een langjarig project waar studenten moeten voortbouwen op hetgeen door voorgangers gemaakt is. Dit lijkt veel op de ICT-werkelijkheid (met veel aandacht voor overdrachtsdocumenten, changemanagement, coding standards, documentatie, etc.).
- Praktijkgerichte projectopdracht met directe klantcontacten leidt tot hogere motivatie.
- Initiatief levert een maatschappelijke bijdrage en iedereen kan en mag meedoen.
- Verbindend element tussen medewerkers en studenten van verschillende opleidingen.
- Studenten verwerven kennis over open standaarden en (state-of-the-art) open source softwareontwikkeling en de juridische, maatschappelijke, praktische en politieke drijfveren achter deze concepten.

ORGANISATIE

Het is een belangrijk onderdeel van WOSI dat studenten leren om op een gestructureerde wijze te werken. En bovenal dat ze binnen de ontwikkelde randvoorwaarden in staat zijn om verder te bouwen op het werk dat hun voorgangers hebben verzet.

Met elk halfjaar een nieuwe groep studenten, dus een verloop van 200 % per jaar, is dat geen eenvoudige opdracht. Om die reden is een groot aantal zaken vastgelegd in procedures. Denk daarbij aan de te gebruiken edittool (Eclipse), buildtool (Maven), coding standards, documentatiestandaarden en licentiewijzer. Die afspraken zijn niet in beton gegoten, maar wie een verandering wil, dient die goed onderbouwd aan de projectleiding voor te leggen. We gebruiken dus (lichte) ITIL-methodieken voor changemanagement. Een aantal uitgangspunten kan vanzelfsprekend niet worden veranderd, zoals het gebruik van open source software en dat alle ontwikkeling database-onafhankelijk en platformonafhankelijk moet zijn.

TECHNISCHE KEUZES EN PROJECTEN

Er is voor gekozen om gebruik te maken van open source software (OSS). De redenen daarvoor liggen voor de hand. Om te beginnen doet deze software inmiddels niet meer onder voor gesloten software. Grote ICT-onderzoeksbureaus als Gartner voorspellen dat OSS binnenkort op alle terreinen de competitie aan kan met gesloten (proprietary) software. Voor het onderwijsveld en de corporaties is het bovendien logisch om dat wat onderzocht en ontwikkeld wordt direct of in een communitymodel ter beschikking te stellen. Als corporaties besluiten om binnen WOSI ontwikkelde technologie daadwerkelijk in te zetten, zal deze technologie (uiteraard) worden overgedragen aan deskundigen bij de corporaties of ICT-bedrijven. Studenten gaan geen applicaties beheren of daar service op verlenen.

MODULAIRE OPBOUW

WOSI kent een modulaire opbouw om schaalbaarheid te borgen. Grote projecten worden vaak onderverdeeld in meerdere kleine projecten, om er op die manier voor te zorgen dat het voor studenten geen grote onbegrijpelijke moloch wordt. Door de permanente hands-onbegeleiding door ervaren architecten en developers, houden we het project op koers. De belangrijkste projecten worden gevormd door het werken aan een van de basismodules. Een module is een in functionele zin duidelijk afgebakende deelapplicatie, die gekoppeld wordt aan het WOSI-framework. Het aantal modules is niet beperkt, maar uitgangspunt is wel dat we proberen er per functie één te ontwikkelen. Belangrijk onderdeel is het formuleren van de principes (eisen) waaraan een module moet voldoen om gekoppeld te kunnen worden aan het framework. Op dit moment is een aantal modules in ontwikkeling, hieronder worden er enkele kort beschreven.

Verkoopproces

Woningcorporatie Servatius uit Maastricht had een probleem met het huidige verkoopproces van een deel van de woningvoorraad. De procedure bij de verkoop van (huur)woningen verloopt nu weinig transparant, is tijdrovend en dus kostbaar. Door studententeams die Maastricht en Enschede hebben bezocht zijn uitvoerige analyses gemaakt en inmiddels is de verkoopmodule klaar. Naar verwachting wordt de module in 2010 daadwerkelijk in productie genomen. Erkend wordt dat het een professionele tool is om het hele proces van een woning te begeleiden (van de start, de taxatie, het vinden van kopers tot en met de overdracht bij de notaris en het overzetten van het energiecontract). Bovendien wordt adequate managementinformatie geboden. Denk daarbij aan de tijd dat het duurt om een woning te verkopen, of de taxatiewaarden worden gehaald, enzovoorts. Mooie dashboards bieden snel het overzicht dat de managers tot voor kort ontbeerden.

Renovatie/Nieuwbouw

Vroeger bepaalde de woningcorporatie welke douche of keuken een nieuwe huurder kreeg. Tegenwoordig proberen corporaties hun klanten een keuze te geven, bijvoorbeeld op het moment dat de woning gerenoveerd moet worden of als ergens een nieuwbouwcomplex wordt neergezet. Op dat moment kunnen aanstaande huurders bijvoorbeeld kiezen voor een eigen kleur tegels, ander sanitair, andere kranen, schuifpuien in plaats van ramen en dergelijke. Dergelijke keuzes hebben meestal effect op de woningwaardering (hoe luxer, hoe meer punten) en soms ook op de huurprijs. Het betreft een uitvoerige module met veel verschillende rollen en rechten waar gegevens zorgvuldig geregistreerd moeten worden en gedurende een lange periode bewaard moeten blijven. Deze zogenaamde optielijstenmodule legt alle gemaakte keuzes vast in onveranderbare contracten, levert meterkastlijsten op voor de aannemers zodat die weten wat er in de woning moet worden aangelegd en cumuleert alle gegevens voor het doen van de juiste bestellingen (zoveel klanten type a, zoveel type b, enzovoorts).

Klachtenregistratie

De klachtenregistratie kan beter, vinden veel corporaties. Een bal door de ruit of een lekkende douche moet je als klant eenvoudig online kunnen melden. Natuurlijk moet er dan wel een check plaatsvinden of de woning wel in het bestand zit en of de melder ook daadwerkelijk huurder is. Dus is ook een goede en makkelijke identificatiemogelijkheid nodig. Vanuit Rotterdam zijn aan de studenten heel wat stimulerende ideeën meegegeven, tijdens het bedrijfsbezoek aan Woonbron. Bijvoorbeeld om te bezien in hoeverre een dergelijk systeem ook meer visueel gemaakt kan worden. Alleen al vanwege het feit dat in sommige wijken meer dan honderd verschillende talen worden gesproken, is dat wenselijk. Het woord kraan of sifon zal voor veel niet-Nederlanders

moeilijk zijn, maar een afbeelding herkent waar-schijnlijk iedereen. En zo wordt voorkomen dat een loodgieter uitrukt om een ruit te vervangen of een timmerman om de cv-installatie te repareren.

FRAMEWORK, DE BASIS

In technische zin is WOSI een behoorlijk complexe zoektocht. Dus met lekker veel uitdaging voor de studenten maar zeker ook voor de betrokken professionals. Een van de zaken die we permanent onderzoeken is welke functies thuishoren in een module en welke in het framework. Het framework bevat alle basisfunctionaliteit (authenticatie bijvoorbeeld) en alle koppelingsmogelijkheden (api's/hooks). Er wordt ook gekeken welke functionaliteit al beschikbaar is. Belangrijke vragen bij deze keuzes zijn: Is het product -eenvoudig- vervangbaar? Past het in de architectuur? Het gebruik van losstaande componenten zorgt namelijk niet automatisch voor een flexibele, schaalbare en duurzame infrastructuur. Technologie moet aansluiten en services moeten elkaar niet overlappen of op de verkeerde plek worden afgehandeld. Het betreft een complexe zoektocht, waarin studenten begeleid worden door meerdere professionals.

DATABASES

Het databaseproject is eigenlijk onderdeel van alle lopende projecten. De bestaande databases van woningcorporaties worden regelmatig op deelterreinen geanalyseerd. Door een aantal corporaties, onder andere De Woonplaats uit Enschede en Ymere uit Amsterdam, is lange tijd gewerkt aan een plan voor een (meer) systematische en strategische plannings-systeem. Uitgangspunt van dit model is dat ieder object (woning) uniek is; het moet dus ook met zijn unieke en gedetailleerde kenmerken op een uniforme wijze te beschrijven zijn. De verwachting is dat door bestudering van dit model er een belangrijke aanzet wordt gegeven voor een toekomstige open-data-standaard, die dan een belangrijke bijdrage zal leveren

aan een betere en efficiënte gegevensuitwisseling zowel tussen corporaties onderling als in de keten (zorginstelling, gemeente, belastingdienst, aannemer, et cetera).

TOEKOMST

Door op deze manier te werken wordt het straks mogelijk om een standaard voor gegevensuitwisseling en databasestructuur te ontwikkelen en bovendien om op een modulaire wijze deelapplicaties (componenten) te ontwikkelen. Dat maakt een geleidelijke overgang naar een volledig service-oriented architecture (SOA) mogelijk, en het maakt de gewenste functionaliteit eenvoudig uitbreidbaar. Om bestaande systemen te kunnen koppelen, wordt in 2010 gestart met een onderzoek naar de ontwikkeling van een enterprise-servicebus.

Om ervoor te zorgen dat de ontwikkelde technologie straks overal kan worden ingezet, is een groot aantal technische randvoorwaarden geformuleerd. Zo is bijvoorbeeld alles wat ontwikkeld wordt geschikt voor alle gangbare operatingssystemen en (vrijwel) iedere database. Dat maakt het voor de studenten niet een makkelijk maar wel een zeer professioneel en uitdagend project!

ERVARINGEN

Niet alle studenten die aan WOSI hebben meegedaan zijn even lovend over het project. Vooral in de eerste periode was het een zoektocht van de projectleiding en de Informaticaopleiding van de HvA naar het vinden van het juiste format. Externe deskundigen bleken vaak zoveel kennis te willen overdragen dat studenten in korte tijd het spoor geheel bijster waren. Dat heeft geleid tot een groot aantal veranderingen in het project. De (hands-on)begeleiding is geïntensiveerd, en door met teams van maximaal vijf studenten aan kleinere afgebakende cases te werken en de organisatie platter te maken, zijn de resultaten voor iedereen veel meer bevredigend. Studenten geven bij

de beoordelingen inmiddels vrijwel zonder uitzondering aan '(heel) veel geleerd' te hebben. Van grootschalige professionele software ontwikkeling, maar ook van het werken in teams, communicatie (onderling en met de studentenprojectleiders), het nut en noodzaak van het gebruik van coding standards, voortbouwen op het werk van voorgangers, enzovoorts.

Groot probleem was (en is) dat niet alle studenten dezelfde voorkennis hebben. Dat maakt het moeilijk om algemene colleges over bijvoorbeeld het gebruik van Spring te geven. Door mensen van ongeveer gelijk niveau in een team te plaatsen, kan kennis veel gericht worden opgebouwd en eindigt weliswaar niet iedereen op hetzelfde niveau, maar heeft wel iedereen de kans het beste uit zichzelf te halen. En dat is precies waarvoor we het allemaal doen.

VERANTWOORDING

De auteur is projectmanager van WOSI en vanaf het prille begin betrokken bij de ambities van de HvA om een levensecht leerbedrijf op te zetten, waar langlopende meerjarige ICT-projecten de basis zouden vormen. WOSI is sindsdien primair gekoppeld aan de minor Advanced software development van de informaticaopleiding. WOSI staat open voor deelname door meer corporaties en onderwijsinstellingen. Meer informatie over WOSI is te vinden op de site: www.wosi.org en verkrijgbaar bij projectmanager Jo Lahaye, J.Lahaye@hva.nl.



Bijpraten tijdens de pauze



Algemeen

- Joop Schoppers, Kennis moet je delen; ict optimaal georganiseerd? Netwit, Netwerk Woningcorporaties en Informatie Technologie. 97890811573-1-5
- Diverse auteurs woningcorporaties, Kennis moet je delen; Bedrijfs- en ICT-architectuur. 9789079472017
- Jo Lahaye (red.), Open Source Software en Standaarden Lesboek uitgegeven door Pearson Education. 9789043015868

Technisch/inhoudelijk

Modelleren

- Sander Hoogendoon, Pragmatisch modelleren met UML 2.0 9043006521

Software

- Phil Hanna, JSP 2.0: The Complete Reference, Second Edition 0072224371
- Craig Walls & Ryan Breidenback, Spring in Action 1932394354
- Anil Hemrajani, Agile Java Development with Spring, Hibernate and Eclipse 0672328968
- Will Iverson, Hibernate: A J2EE(TM) Developer's Guide 0321268199
- Eric Freeman & Bert Bates & Elisabeth A. Freeman, Head First Design Patterns 0596007124
- Martin Fowler, Patterns of Enterprise Application Architecture 0321127420
- Kent Beck, Test-Driven Development By Example

Database

- Ken Henderson, SQL Server 2005 Practical Troubleshooting The Database Engine 0321447743
- Ken Henderson, The Guru's Guide to SQL Server Stored Procedures, XML, and HTML 0201700468
- Leo Wiegerink & Jeanot Bijpost & Marco de Groot, Relationale Databases en SQL, Tweede herziene druk 9039522219
- David M. Kroenke, Beginselen, Ontwerp en Implementatie 9043008427
- Joe Celko's, SQL For Smarties 0123693799
- Ken Henderson, The Guru's Guide to SQL Server Architecture and Internals 9780201700473

Projectmanagement

- Roel Grit, Projectmanagement 9001347037
- Wilfred de Feiter, Basiscursus Project 2000 9039516618
- Mark van Onna & Brigit Hedricks & Gunther Schraven, De kleine Prince 2 907630498x
- Hans-D Litke & Ilonka Kunow, Zakboekje projectmanagement 9062283098
- Gerrit Koch, NCB Versie 3 - Nederlandse Competence Baseline 9087530266
- Bert Hedeman, Projectmanagement op basis van PRINCE2 9087530464

Groot aantal digitale uitgaven, zoals:

- Augustus 2007 John Casey, Vincent Massol, Brett Porter, Carlos Sanchez, Jason van Zyl Better Builds With Maven How to guide for Maven 2.0 Engels
- 2.5.x Spring Source Spring Reference Engels: <http://static.springsource.org/spring/docs/2.5.x/reference/index.html>
- SE Craig Walls Spring in action Chapter 1 Engels

3

Testen geïntegreerd in het curriculum Software Engineering.



Pieter van den Homberh - Fontys Hogeschool Techniek en Logistiek

Fred van Odenhoven - Fontys Hogeschool Techniek en Logistiek

SAMENVATTING

Testen van software is een speerpunt in onze opleiding Software Engineering. In de propedeusefase wordt de testgedreven softwareontwikkeling geoefend. De student wordt aangeleerd software met testen af te leveren. Als onderdeel van de toetsing werd een performance-assessment ontwikkeld, dat de mogelijkheid biedt modelleren, programmeren en testen integraal te toetsen. Studenten blijken deze nieuwe toetsvorm positief te waarderen. In het kader van competentiegericht onderwijs is dit performance-assessment een waardevolle toevoeging.

INLEIDING

Studenten die leren programmeren, leren een nieuwe taal, maar ook een nieuwe manier om problemen op te lossen met de computer. Als de programmeeropgaven wat ingewikkelder worden, neemt de programmeervreugde af als fouten niet snel gevonden worden. De volgende aanpak is illustratief: de code wordt net zolang opnieuw gecompileerd totdat de compiler tevreden is. Vervolgens wordt het programma net zo vaak uitgevoerd dat er uiteindelijk een uitvoerbaar programma ontstaat, dat doet wat de student op dat moment van het programma eist. Daarbij worden na elke run de fouten, die volgens de uitvoer nog aanwezig

zijn, gecorrigeerd. Het programma bepaalt in feite welke fouten de student moet corrigeren. Dit lijkt op testen maar is het niet!

Deze inefficiënte en onvolledige werkwijze kan op professionele wijze verbeterd worden door het inzetten van unittesten. Naast unittesten is het leren omgaan met een versiebeheersysteem van grote waarde voor de praktijk van de software engineer. Dat blijkt ook wel uit het feit dat de meeste IDE's zoals Eclipse en Netbeans diverse varianten van versiebeheer standaard ondersteunen.

CURRICULUM

De vakken in het eerste jaar, die relevant zijn voor dit verhaal, zijn in het eerste semester:

PRO1 programmeren in Java deel 1.

En in het tweede semester:

PRO2 programmeren in Java deel 2,

SEN1 software engineering deel 1,

MOD1 objectgeoriënteerd modelleren met UML deel 1,

PRJ2 software project: ontwerpen en realiseren van een webapplicatie met Java en een database.

De module SEN1 behoeft nadere toelichting: het hoofdthema is objectgeoriënteerd testen en testbaarheid, evenals het beheren van een softwareproject met een versiebeheersysteem. Verder worden algemene principes van de praktijk van softwareontwikkeling bijgebracht. Als leerboek werd een Duitstalig boek door ons in het Nederlands vertaald, Vigneschow [1].

Een hoofddoel van de module is studenten de gewoonte aan te leren software altijd met de bijbehorende (unit)testen af te leveren. Dit is een van de praktijkgerichte doelstellingen van het vak. In het project PRJ2 kunnen studenten in een projectgroep de diverse vakkennis in de praktijk toepassen. Zoals het werken met een versiebeheersysteem (momenteel Subversion) en het systematisch testen van de software.

PROGRAMMEREN

In de module PRO1 wordt Java onderwezen en maken de studenten voor het praktisch werk gebruik van BlueJ. In het eerste semester maken de studenten al kennis met unittesten, omdat BlueJ het werken met JUnit ondersteunt. Systematisch testen schrijven bij het ontwikkelen van een Javaprogramma wordt echter niet gedaan. De studenten zijn vooral bezig met het leren van de taal en maken nog geen ingewikkelde programma's.

In de module PRO2 wordt de Javakennis verder uitgebreid en wordt aandacht besteed aan web-programmering, in dit geval werken met JSP en JDBC. Deze onderdelen zijn noodzakelijk voor het project PRJ2. Hierin wordt een webapplicatie gerealiseerd met Java.

TESTEN

Aangezien BlueJ het werken met JUnit ondersteunt, maken studenten in het eerste semester al kennis met unittesten. In het tweede semester wordt Netbeans gebruikt voor het programmeerwerk. De aard van het project PRJ2 heeft tot deze keuze geleid.

In SEN1 wordt met kleinere programmeeropdrachten het testgedreven ontwikkelen en het gebruik van Subversion geoefend. Ook hiervoor is Netbeans een goede keuze.

Voor het vak SEN1 werd een performance-assessment ontwikkeld, dat het mogelijk maakt de kennis en kunde

van de student in een zo goed als mogelijke praktijk-getrouwe omgeving te toetsen. Daarnaast wordt met een schriftelijke toets de theoretische kennis getoetst. De resultaten van het begeleidend practicum worden ook meegenomen in de eindbeoordeling van de student. In het eindcijfer weegt het performance-assessment het zwaarst mee. Dit assessment toetst niet alleen het programmeren van de testen, maar ook de teststrategie en de kwaliteit van de testen, zoals onder andere de test-coverage en niet in de laatste plaats de keuze van de testdata op basis van equivalentieklassen.

VERSIEBEHEER

Het beheren van broncode wordt door professionele programmeurs gedaan met een versiebeheersysteem. De open source varianten Cvs, Subversion (svn) en het recentere Mercury of Git worden veel toegepast. Als wij professionele software-engineers willen opleiden, dan dienen onze studenten ten minste een van deze tools te leren gebruiken.

PERFORMANCE-ASSESSMENT

Ten behoeve van het performance-assessment m.b.t. testgedreven programmeren is een speciale op Linux gebaseerde toetsomgeving opgezet. Deze keuze geeft ons volledige controle over wat er wel en niet gedaan kan worden met de machines zonder afhankelijk te zijn van derde partijen. In feite implementeren we ons eigen policystelsel. De omgeving bestaat uit een speciaal daartoe ingerichte 'examserver' met clients, die een ingeperkte omgeving (rbash) aan de studenten aanbieden.

We hebben de volgende eisen aan de testomgeving gesteld:

- Het assessment moet een praktijktest zijn; de student moet getoetst worden op zijn/haar vaardigheden met betrekking tot ontwerpen en implementeren van de testen met gebruikmaking van de tools.



Twee van de ruim 300 bezoekers van het NIOO 2009

- Elke student moet de toets individueel ondergaan.
- De omgeving moet door ons gecontroleerd worden en voor elke deelnemer gelijk zijn.
- De omgeving moet beschikbaar zijn voor alle studenten voordat de toets wordt gehouden. Dus: de studenten moeten ervaring kunnen opdoen met betrekking tot de toetsomgeving.
- De omgeving moet zo realistisch mogelijk zijn en dus vergelijkbaar met de normale werkomgeving gedurende projectwerk en programmeerwerk.
- De studenten hebben de beschikking over een volledig functionele IDE (Netbeans of Eclipse).
- De studenten moeten een begrensde toegang hebben tot elektronische documentatie, zoals API-documentaties: van de applicatie, van het JUnit-framework en van Java zelf.
- De studenten moeten niet hun werk kunnen delen tijdens de test: alle netwerktoegang behalve naar de examenserver moet afgesloten zijn.
- Er vindt geen disksharing plaats.
- De omgeving moet zoveel als mogelijk fraude voorkomen.
- De studenten leveren hun werk in via een versiebeheersysteem. Subversion is op dit moment de keus.

Inrichting of implementatie van de omgeving:

- De student krijgt een speciaal account voor de duur van de toets met verse wachtwoorden. De homedirectory van deze gebruiker is op de lokale harde schijf en wordt tijdens inloggen aangeemaakt. Deze directories worden kort na de toets weer verwijderd. Deze home-dirs bevatten een setup voor de IDE en een browser (Firefox) met bookmarks naar de documentatie.
- Toegestane internetverkeer instelbaar tot op hostniveau en/of poortniveau.
- Zoals bekend kunnen Unixachtige machines alleen commando's uitvoeren die te vinden zijn op het pad, (dat wij bepalen) of die gespecificeerd zijn met behulp van een relatief of absoluut pad (bijvoorbeeld met / in het pad).
- We gebruiken rbash als shell, die het veranderen van directory en pad verhindert en commando's met een / er in onmogelijk maakt. Dit geeft ons volledige controle over wat de student wel en niet kan doen.

```
\begin{itemize}
\item \large
\EN{A few \LaTeX macros to help in making multilingual task descriptions.}
\NL{Een paar \LaTeX macros helpen bij het maken van meertalige
opgavenbeschrijvingen.}
\DE{Einige \LaTeX Macros zum erstellen von mehrsprachigen
Aufgaben-Beschreibungen.}
\end{itemize}
```

Snippet 1: Een paar Latex macro's helpen bij het maken van meertalige opgavenbeschrijvingen

We hebben de volgende tools ontwikkeld ten behoeve van de omgeving:

- Kleine aanpassingen aan de start-scripts van de display en windowmanager van de Linux X11 omgeving.
- De Windowmanager is IceWM, die is licht en gemakkelijk aan te passen.
- Kleine scripts om de SSH-verbinding op te zetten en de verbinding te testen.
- Voor de documentatietext: een paar macro's helpen bij het maken van meertalige opgavenbeschrijvingen, zie Codesnippet 1.
- Antbuildscripts genereren de Javadoc, zip-bestanden voor de studenten en een test-omgeving om onze eigen oplossing te testen.
- Scripts om de accounts en de repositories 'on the fly' te creëren.
- Om de studenten te voorzien van code die up-to-date is, waaruit onze oplossing verwijderd is hebben we een codestripperantask ontwikkeld. Zie Codesnippets 2, 3 en 4.

```
<taskdef name = 'codestripper'
  classname = 'org.fontysvenlo.codestripper.CodeStripper' />
<target name = 'strip'>
  <codestripper
    todir = 'out'
    dir = 'src'
    deletelines = 'true'
    includes = '**/*.java'
    dryRun = 'false'
    verbose = 'true'
    starttag = '^\\s*/[*]{2}.*'
    endtag = '.*[*]/.*'
  />
</target>
```

Snippet 2: Deze taak strip C-stijlcommentaar uit alle Java-bronbestanden, maar kan van alles strippen uit op tekst gebaseerde bestanden. (Zie: <http://javabits.fontysvenlo.org>)

```
int getCredits(Score score, PlayLine line) {
  //Start Solution::replacewith:://TODO
  if (line == PlayLine.D1 || line == PlayLine.D2) {
    return 2*this.getCredits(score);
  }
  return getCredits(score);
  //End Solution::replacewith::return 0;
}
```

Snippet 3: Code in de repository

```
int getCredits(Score score){
  //TODO
  return 0;
}
```

Snippet 4: Code na stripping

DE APPLICATIE

We wilden een applicatie, die niet goed testbaar is door hem gewoon te gebruiken. Daarmee verhinderen we dat code via de applicatie wordt geschreven in plaats van via testen. Wij denken dat je dat kunt bereiken door iets in te brengen, dat niet door de student te controleren is: randomness. Van daaruit is het een kleine stap naar een spel. De slotmachine was geboren. Ons initiële ontwerp was functioneel maar saai (voor de test zelf was het achteraf gezien prima). Nu hebben we een swingapplicatie die ook geluiden produceert afhankelijk van het spelverloop.

Het is een goedogende applicatie, die een model bevat met goede testmogelijkheden. Een voorbeeld is de methode met taak: 'bepaal de score op een bepaalde speellijn'.

Van de slotmachine hebben we meerdere varianten op voorraad:

- iconslotmachine
- numberslotmachine
- yachtreeslotmachine
- pokerslotmachine

Zie voorbeelden in figuren 1 en 2.

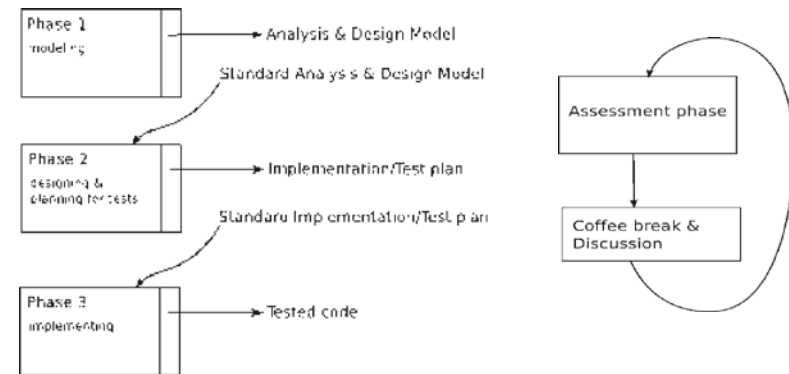
(DEMO <http://javabits.fontysvenlo.org/iconslotmachine/dist/launch.jnlp>)



Figuur 1:
Icon slotmachine



Figuur 2:
Poker slotmachine



Figuur 3:
Assessment-fasen

Figuur 4:
Assessment-cyclus

DE OPDRACHT

Projectcode is compleet geïmplementeerd inclusief testsoftware (EYODF!). De code is voorzien van Java-doc-commentaar. Onze zorg is de foutkans daarin zo klein mogelijk te maken. De codestripperanttask in de antbuildfile voor het codestrippen en aldus maken van het examenproject voor de studenten. De opgaven-code stemt zodoende altijd overeen met onze code, die getest is met onze code. Dit is ook het geval met de code in de overige documentatie die als PDF wordt aangeboden. De opgavenbeschrijving is een aparte PDF en de //TODO's in de code zijn natuurlijk ook een goed hulpmiddel. Alle code, configuratie en latex-bestanden bevinden zich in een docent-repository.

UITVOERING EN VERWERKING

Als we onze examenomgeving zelf getest hebben, kunnen we de omgeving openstellen voor de studenten. Het is mogelijk tijdens het assessment de voortgang te volgen door in de repositories van de studenten de commitopdrachten te monitoren. Weinig

commits van een student zeggen niet altijd dat hij of zij achterloopt, het kan ook zijn dat die student te weinig commits uitvoert, wat tegen de aanbevelingen is. Er moet namelijk regelmatig gecommited worden.

Onze eigen tests kunnen daarna ook gebruikt worden om de code van de studenten te testen. Het is niet verstandig om alleen op basis van deze testen een cijfer vast te stellen. Een tweede testmogelijkheid is om de test van de student te laten uitvoeren op onze eigen docentcode. Verder kunnen ook coveragetesten worden toegepast, maar het is raadzaam om de code van de studenten ook met het oog te bekijken.

Uitbreidingen

Het performance-assessment werd tot nu toe gebruikt voor de module SEN1. Het is natuurlijk ook een assessment voor de programmeervaardigheid zelf en zou daarom ook mogelijk ook een deel van de toets van de module PRO2 kunnen vormen.

Een wezenlijke uitbreiding is de combinatie van meerdere performance-assessments, zoals achtereenvolgens voor: modelleren, testontwerp en testuitvoering.

Na elke fase is een pauze wenselijk. Daarin kunnen studenten ook onderling over de afgelopen fase discussiëren. De volgende fase borduurt dan voort op hetzelfde thema, maar begint met eenzelfde beginsituatie voor alle deelnemers. Bijvoorbeeld: een modelleringopgave kan voorafgaan aan een implementatiefase. In de implementatiefase krijgt elke student een standaarduitwerking van de

modelleringfase. Zie ook de figuren 3 en 4. Drie fasen zijn waarschijnlijk teveel voor een assessmentdag.

CONCLUSIES

Onze ervaring beperkt zich tot nu toe tot het toetsen van het programmeren en het testen, en die ervaring is ronduit positief te noemen. Het opzetten van de omgeving om het performance-assessment uit te kunnen voeren is niet bijzonder complex, maar kost wel de nodige inspanning en controles. De studenten ervaren het als een eerlijke en nuttige aanvulling op de schriftelijke toetsen en het groepswork.



De centrale hal van de Faculteit Natuur en Techniek van de HU



[1]

Uwe Vigenschow Objectgeoriënteerd Testen, vertaald uit het Duits door Joop Wekking, Fontys Hogeschool voor Techniek en Bedrijfsmanagement 2007, ISBN 978-90-811523-1-02

Over de auteurs

Ir. Pieter van den Homberh en
Dr. Ir. Ferd van Odenhoven zijn
verbonden aan Fontys Hogeschool
Techniek en Logistiek te Venlo,
opleiding Software Engineering.

Aansluiting hbo en wo op instroomopleidingen Info Support

Gert Jan Timmerman - hoofd Kenniscentrum Info Support

SAMENVATTING

Al twintig jaar werft Info Support (een IT-dienstverlener) starters op de arbeidsmarkt onder mensen die op hbo- of wo-niveau zijn afgestudeerd in Informatica. Ter wederzijdse kennismaking bieden we studenten ook vaak een afstudeerplek aan. Aan het begin van het dienstverband worden deze starters, na een strenge selectie, verder opgeleid in ons eigen Kenniscentrum. In de afgelopen twintig jaar hebben we gemerkt dat de kloof tussen wat men geleerd heeft tijdens de opleiding en wat wij aan kennis en vaardigheden van onze instroom verwachten, steeds groter wordt. Zijn onze verwachtingen te hoog? Is de mismatch te verklaren door inhoudelijke verschillen van inzicht? Wat zou Info Support hieraan kunnen doen en wat kunnen hbo- en wo-opleidingen doen?

TREFWOORDEN

Starters, instroom, opleidingen, kennis, vaardigheden, vakmanschap

INLEIDING

Info Support is een IT-dienstverlener die maatwerk-software bouwt voor klanten, zowel met .NET- als met Java-technologie. Voor Info Support is vakmanschap een kernwaarde en dit blijkt onder andere uit het feit dat wij uitsluitend medewerkers aannemen (afgezien van stafmedewerkers) die een afgeronde hbo- of universitaire opleiding Informatica hebben. Ook het feit dat we deze medewerkers voortdurend opleiden draagt

hiertoe bij.

Starters (nieuwe medewerkers die rechtstreeks van een informaticaopleiding komen) krijgen een (standaard) opleidingstraject voordat ze ingezet worden op projecten. Bij deze initiële opleiding bouwen we, uiteraard, voort op de basis die in hun Informaticastudie gelegd is. Steeds vaker lopen we er echter tegenaan dat deze starters niet de kennis hebben waar wij wel op rekenen. Er is sprake van een steeds groter wordende mismatch tussen wat de starters aan kennis en vaardigheden hebben en wat wij daarvan verwachten. Deze mismatch is niet gekoppeld aan een bepaald opleidingsinstituut, maar we komen die tegen bij alle opleidingen Informatica in Nederland.

Er zijn verschillende verklaringen mogelijk voor een dergelijke situatie: zo kan het zijn dat wij te hoge eisen stellen aan de starters. Het kan ook zijn dat er inhoudelijke verschillen zijn in de manier waarop wij tegen zaken aankijken en hoe daarover gedacht wordt op hogescholen en universiteiten. De bedoeling van de sessie is om daar meer helderheid over te krijgen.

De hiervoor beschreven mismatch tussen de kennis van starters en de verwachtingen die wij daarvan hebben, spitst zich toe op een aantal onderwerpen.

SYSTEEMONTWIKKELING

Allereerst het systeemontwikkelingstraject. Alle systeemontwikkelingsmethodieken zijn uiteindelijk afgeleid van het klassieke Watervalmodel. Dat wil zeggen dat alle systeemontwikkelingsmethodieken op een of andere manier een volgorde aanhouden in het

verzamen en specificeren van requirements, het maken van een ontwerp (functioneel en technisch), realisatie en testen en ten slotte oplevering. In het Watervalmodel worden deze fasen sequentieel doorlopen en wordt in principe pas met een volgende fase gestart als de vorige afgerond is. In de meer Agile-methodieken wordt daarentegen vaak iteratief gewerkt, waarbij steeds stukjes van de functionaliteit ontworpen en gebouwd worden. Omdat het klassieke Watervalmodel de basis is voor al deze verschillende methodieken, is het van bijzonder groot belang dat studenten het Watervalmodel heel goed leren kennen. In onze ervaring hebben starters daarentegen een zeer verwrongen beeld van het Watervalmodel. Zo hebben zij vaak de indruk dat het Watervalmodel alleen geschikt is als 'alles van tevoren bekend is' en dat het onmogelijk is naar een vorige fase terug te keren. Ook denken ze dat het onmogelijk is in de loop van het traject wijzigingen te accepteren. Hierdoor hebben ze vaak de indruk dat het Watervalmodel nooit in de praktijk gewerkt kan hebben of uitsluitend geschikt is voor zeer kleine projecten. Wij zouden veel liever zien dat studenten eerst het Watervalmodel echt goed leren kennen en begrijpen, voordat ze in aanraking komen met meer Agile-methodieken, omdat het lastig is, zonder goede kennis van het klassieke Watervalmodel, andere methodieken goed te begrijpen. Een andere misvatting van starters gaat over Requirements: veel starters beginnen bij het opstellen van requirements bij de input, terwijl requirements meestal vooral over de output gaan. Uit de output kan vervolgens de benodigde input afgeleid worden. Organisaties laten systemen ontwikkelen vanwege de output: dat is (in het algemeen) het doel van een informatiesysteem. Dat er ook nog gegevens ingevoerd moeten worden, kost alleen maar tijd en geld. Dat is een vorm van investering die men moet doen om ooit gegevens uit het systeem te kunnen halen. Het is dus van belang om, gegeven de output, de input zo beperkt mogelijk te houden. Dat kan alleen als bij het opstellen

van requirements vooral naar de output gekeken wordt en de benodigde input hiervan afgeleid wordt.

FUNCTIONEEL EN TECHNISCH ONTWERP

Ook over de onderwerpen Functioneel en Technisch Ontwerp komen we veel misverstanden tegen bij starters: vaak denken zij dat het Technisch Ontwerp een detaillering is van het Functioneel Ontwerp en dat het Functioneel Ontwerp bijgevolg vooral globaal is. Ook zijn ze vaak van mening dat de Requirements het Functioneel Ontwerp vormen.

De werkelijkheid is echter dat het Functioneel en Technisch Ontwerp twee kanten van dezelfde medaille vormen. Beide zijn op hetzelfde detailniveau uitgewerkt. De belangrijkste reden om onderscheid te maken tussen Functioneel en Technisch Ontwerp is dat functionele en technische beslissingen door verschillende mensen (in verschillende rollen) genomen worden. Om te voorkomen dat functionele beslissingen impliciet genomen worden door een persoon die de rol heeft van Technisch Ontwerper en die dus geen goed beeld heeft van de gewenste functionaliteit, is het van belang om functionele beslissingen op te nemen in een Functioneel Ontwerp en Technische beslissingen in een Technisch Ontwerp.

In de praktijk wordt tegenwoordig veel gebruik gemaakt van UML. Ook Info Support gebruikt deze taal om ontwerpen weer te geven. Veel starters denken dat het Functioneel Ontwerp af is als ze een Use Case Model gemaakt hebben. Het Technisch Ontwerp bevat dan een Class Diagram en een aantal Sequence Diagrammen. In het Class Diagram zijn de verschillende classes weergegeven met de cardinaliteiten van de relaties (1-op-1, 1-op-veel, veel-op-veel). Dat zijn functionele beslissingen die op deze manier in een Technisch Ontwerp terecht komen en niet gedekt worden door een Functioneel Ontwerp dat deze relaties beschrijft. Ook de attributen (properties) van een class zijn functionele beslissingen (welke

attributen horen bij welke class, welke informatie wil ik modelleren?). Ook deze beslissingen komen op die manier terecht in het Technisch Ontwerp. Vaak worden er in het Class Diagram technische datatypen gebruikt bij de verschillende attributen (typen zoals int, double, datetime, string etc.), maar deze datatypen moeten natuurlijk gebaseerd zijn op de bijbehorende functionele domeinen. Als er in het Functioneel Ontwerp echter alleen een Use Case Model is opgenomen, zijn deze functionele domeinen nergens beschreven en is dus niet na te gaan of de technische datatypen de functionele domeinen voldoende afdekken. Naar onze mening zou een Functioneel Ontwerp niet alleen uit een Use Case Model moeten bestaan, maar ook uit een Analysis Model, waarin gedetailleerde Class Diagrams en Sequence Diagrams staan. Alleen op die manier is duidelijk wie (in welke rol) de functionele keuzes gemaakt heeft en wie (in welke rol) de technische keuzes gemaakt heeft. Ook is op die manier altijd vast te stellen of de technische beslissingen de functionaliteit voldoende afdekken en waar dat niet het geval is, kan het verschil inzichtelijk gemaakt worden, zodat hierover een akkoord verkregen kan worden.

SYSTEEMDOCUMENTATIE

Een ander misverstand bij starters is dat men denkt dat Functioneel en Technisch Ontwerp een onderdeel van de systeemdokumentatie is. Het grote verschil tussen Functioneel en Technisch Ontwerp enerzijds en systeemdokumentatie anderzijds, is dat ontwerpen van tevoren gemaakt worden (met als doel het expliciet vastleggen van genomen beslissingen) terwijl documentatie achteraf gemaakt wordt met als doel het systeem inzichtelijk te maken voor toekomstige beheerders of gebruikers. Documentatie wordt altijd gemaakt met een concreet doel voor ogen (bijv. installatiedocumentatie) en een bepaalde doelgroep (bijv. gebruikers of beheerders). De documentatie wordt op maat gemaakt voor het gestelde doel en de betreffende doelgroep. Onderdelen uit het Functioneel

Ontwerp en Technisch Ontwerp kunnen hierbij gebruikt worden, maar dat is een kwestie van hergebruik. Het doel van het maken van ontwerpen enerzijds en het maken van documentatie anderzijds is totaal verschillend. Het probleem van dit misverstand (dat veel starters denken dat ontwerpen documentatie zijn) is dat men vaak denkt dat ontwerpen ook heel goed achteraf gemaakt kunnen worden (dus eerst code schrijven en dan ontwerpen maken). Voor documentatie is dit prima, maar voor ontwerpen is dit principieel onmogelijk omdat na het schrijven van de code alle beslissingen (zowel functioneel als technisch) al genomen zijn en het ontwerp dus in feite al gemaakt is. Probleem hierbij is dat alle beslissingen impliciet genomen zijn zonder dat degene die de beslissing nam zich bewust was van de beslissing en zonder dat hij zich heeft afgevraagd of hij wel de rol had waarin hij die beslissing kon nemen. Ook worden impliciete beslissingen vaak genomen zonder dat men zich voldoende bewust is van de consequenties van de gemaakt keuze.

TRANSACTIES

Starters realiseren zich vaak niet hoe essentieel transacties zijn bij alle gegevensverwerkende systemen. Vaak hebben zij de indruk dat Transacties alleen van belang zijn bij het overboeken van geld door banken. In de praktijk is het zo dat alle acties waarbij meer dan één tabel betrokken is (en in een genormaliseerd model is dat al snel het geval) transacties nodig hebben. Het belang van correct werkende transacties voor de integriteit van informatiesystemen is bij veel starters niet voldoende bekend.

Transacties hebben vier kenmerken: Atomicity, Consistency, Isolation en Durability, ook vaak afgekort als ACID-properties. Van deze vier kenmerken, kennen veel starters alleen Atomicity. Vooral Isolation (concreet gemaakt in een aantal IsolationLevels) is bij veel starters onbekend. Het gevolg hiervan is dat

men omtrent IsolationLevels geen keuzes maakt in het Technisch Ontwerp en bij de realisatie dus kiest voor de (toevallige) defaultwaarde. Dit leidt tot een impliciete keuze met gevolgen die soms verstrekkend zijn. Wij zijn van mening dat Transacties, inclusief IsolationLevels, tot de essentiële leerstof van Informaticastudenten behoren.

CONCLUSIE

Ervaring leert dat veel starters op de arbeidsmarkt die een Informaticaopleiding afgerond hebben, essentiële kennis en vaardigheden missen. Daarnaast hebben ze vaak een verkeerd beeld van het toepassen van systeemontwikkelingsmethodieken en ontwerpmethoden. Hierdoor ontstaat er een mismatch tussen de kennis en vaardigheden van starters enerzijds en de verwachtingen die bedrijven, zoals Info Support, hiervan hebben.

Een van de vele keynote-sessies die druk werden bezocht

VERANTWOORDING

Gert Jan Timmerman is hoofd van het Kenniscentrum van Info Support. In deze functie is hij verantwoordelijk voor de vakinhoudelijke selectie van nieuwe medewerkers en voor hun (initiële) opleiding. In de afgelopen 13 jaar heeft hij in deze functie honderden starters aangenomen en opgeleid. Daarnaast is hij lid van Beroepenveldcommissies van verschillende hogescholen en ook als gecommiteerde is hij al meer dan 15 jaar betrokken bij het afstuderen. De inhoud van deze lezing is een weerslag van de ervaringen die hij in de afgelopen jaren heeft opgedaan. Uiteraard is hierbij sprake van een zekere generalisatie: niet alle starters zijn gelijk en ook de kennis van de ene starter is niet gelijk aan die van een ander. Wat hier beschreven is, geeft weer wat er in de volle breedte van het Informaticaonderwijs in Nederland zichtbaar is en hoe daar vanuit het perspectief van een bedrijf dat traditioneel veel starters aanneemt, tegenaan gekeken wordt.



Media en ICT: gescheiden of toch liever een latrelatie?



Laura Benvenuti - Open Universiteit Nederland

Geeske Bakker - Universiteit Utrecht

Gerrit C. van der Veer - Open Universiteit Nederland

SAMENVATTING

De informaticus en de mediaspecialist hebben elkaar nodig. Samenwerking tussen hun disciplines werpt boeiende vragen op, die volop in de belangstelling staan. Deze vragen kunnen echter ook fundamenteel zijn. Als zo'n moeilijke fundamentele vraag in zicht komt, roepen de betrokken faculteiten: 'Dat is niet onze core business!' en dan hapert de samenwerking. Dit dilemma wordt geïllustreerd met een voorbeeld, de module Virtuele Werelden. Onze vraag luidt: willen wij deze relatie voortzetten of moeten onze disciplines gescheiden verder?

TREFWOORDEN

Media, ICT, multidisciplinariteit

INLEIDING

'Goeiemoggel!' Eind 2007 was dit een ochtendgroet waarmee je bewees bij de tijd te zijn. Vóór de introductie van een nieuw reclamespotje van KPN eerder dat jaar zou aan het woord echter geen betekenis zijn toegekend en op het ogenblik word je voor wel erg oubollig versleten als je het nog gebruikt.

Mediacultuur is een bewegend verschijnsel. Ontwerpers en bouwers van crossmediale applicaties

kunnen die cultuur sturen. Maar wie zijn die ontwerpers en die bouwers eigenlijk en uit welke cultuur komen zij zelf?

MEDIA EN ICT, EEN WAT ONGEMAKKELIJKE VERBINTENIS

Mediaspecialisten zijn geïntrigeerd door betekenisverschuivingen. Spelen met betekenis is hun vak. Vaststaande betekenissen bestaan niet en als ze dreigen post te vatten, moeten ze beslist worden onderuit gehaald. Vanuit deze achtergrond experimenteren ze volop met de geavanceerde authoring tools die hen op het ogenblik ten dienste staan. Mbo-, hbo- en universitaire opleidingen hebben hier voortvarend op ingespeeld door talloze opleidingen aan te bieden in bijvoorbeeld (multi)mediadesign, communicatie en gaming die veel studenten trekken. Vanuit dezelfde invalshoek wordt daarnaast volop onderzoek gedaan naar het gebruik en het effect van deze tools, vooral op universitaire instellingen met media- en communicatieopleidingen, maar sinds de instelling van lectoraten ook binnen het hbo. Deze mediaspecialisten missen echter vaak de kennis om alle mogelijkheden van de geavanceerde software uit te buiten dan wel de tools voor hun gebruik aan te passen. Daarvoor hebben ze de informaticus nodig.

De informaticus die de tools maakt, probeert vanuit zijn achtergrond verandering in betekenis juist te voorkomen. De discipline stelt immers alles in het werk om de betekenis van software-uitingen te borgen. Vanuit deze eis hebben informatici theorieën en methoden

ontwikkeld die diametraal verschillen van die van mediaspecialisten en die daarom voor de laatstgenoemde professionals slechts mondjesmaat toepasbaar zijn (Benvenuti en van der Veer, 2009). Toch- of wellicht juist daarom - is het vakgebied van de mediaspecialist heel waardevol voor de informaticus: zij werpt vragen op die de aanzet geven tot ontwikkeling van het vakgebied informatica. In dat licht beziet de vakgroep Mens, Computer en Maatschappij van de Faculteit Informatica van de Open Universiteit Nederland (MCM) de verbintenis tussen media en ICT als intrigerend. De informaticus heeft hier de mediaspecialist nodig.

Waar moet de innovatie vandaan komen?

Innovatie vereist exploratie en onderzoek. In het geval van innovatie op het snijvlak van media en ICT wordt in Nederland volop geëxperimenteerd in gebieden die niet traditioneel tot de informatica horen. De media- en ICT-laboratoria in het mbo en het hbo ressorteren vaak onder faculteiten die informaticaonderwijs niet tot hun core business rekenen. De langetermijn-innovatie van software kan niet uit deze hoek komen. Informaticafaculteiten rekenen op hun beurt toepassingen in de media niet tot hun primaire gebied. Zij missen vaak de rechtstreekse ervaring en hebben daarmee minder zicht op de soms fundamentele vragen die deze toepassingen opwerpen. Uit het Visitatierapport Informatica 2007 blijkt dat de universitaire Informaticaopleidingen worstelen met de relatie met andere disciplines. 'De vraag rijst dan ook of dergelijke opleidingen niet beter als specialisatie binnen de bestaande informaticaopleiding ingericht kunnen worden.' (Visitatierapport Informatica 2007, pag. 31). Media en ICT leveren vanwege de verschillende uitgangspunten een wat ongemakkelijke verbintenis op. Toch is een kruisbestuiving nodig, om tot innovatie te komen. Het Nederlandse Informatica Onderwijs Congres 2009 heeft dan ook niet zonder reden de relatie tussen media en ICT als één van de thema's gekozen.

² De coördinator van de module is Bob Cruisberg.

Samen kan het beter

De vakgroep MCM vindt dat academisch onderwijs niet alleen kan worden aangeboden via tekstboeken. Studenten die uitsluitend andermans teksten bestuderen komen niet in aanraking met state-of-the-artdilemma's. Hun wordt de kans ontnomen om het vakgebied zelf te verkennen. De vakgroep vindt dan ook dat academisch onderwijs idealiter hoort plaats te vinden in laboratoria, waar studenten, docenten, onderzoekers en professoren samenwerken en van elkaar leren (van der Veer, 2008). De Open Universiteit Nederland is echter een instituut voor afstandsonderwijs; een dergelijk onderwijsmodel kan op dit moment hooguit virtueel worden geïmplementeerd. Daar is overigens wel een wereld mee te winnen.

De vakgroep heeft een werkbare modus proberen te vinden door aansluiting te zoeken bij de Faculteit Communicatie & Journalistiek van de Hogeschool Utrecht. Ze wil daarmee als in een latrelatie de eigen informatica-identiteit borgen, maar tegelijk voldoende voeling krijgen met authentieke toepassingen om tot nieuwe inzichten te komen in één van haar onderzoeksgebieden, te weten experience design. Het Lectoraat Crossmedia Content (Van Vliet, 2008) van deze faculteit houdt zich ook bezig met onderzoek naar experience design, vanuit de gebruikersinvalshoek. Zij richt zich op de wijze waarop vanuit de content met alle middelen die ons tegenwoordig ten dienste staan een ervaring kan worden opgeroepen. Als eerste aanzet tot samenwerking hebben wij gekozen voor een onderzoek naar de ervaringen van studenten in een experimentele leeromgeving: de module Virtuele Werelden.

EEN CASUS: DE MODULE VIRTUELE WERELDEN

De opleiding Digitale Communicatie van de Faculteit Communicatie & Journalistiek heeft, mede door het Lectoraat Crossmedia Content, een experimentele module Virtuele Werelden² ontwikkeld voor studenten in het derde studiejaar. De module was al gestart

in het studiejaar 2005/2006, maar in het studiejaar 2007/2008 werd voor het eerst als leeromgeving Active Worlds ingezet, naast de fysieke leeromgeving. Active Worlds is een virtuele omgeving die door Surfnets beschikbaar wordt gesteld aan het hoger onderwijs om mee te experimenteren³.

De fysieke leeromgeving bestond uit een klaslokaal met een draadloos netwerk en een groot projectiescherm. De 19 deelnemende studenten waren 4 uur per week aanwezig in dit lokaal. Ze beschikten allemaal over een eigen laptop en werkten na afloop van de lessen thuis door in de virtuele omgeving. Gezien de invalshoek van de opleiding lag de focus van de colleges bij het communicatieve gebruik van virtuele werelden. De leerdoelen waren als volgt geformuleerd: inzicht verwerven in het begrip virtuele wereld, in het ontwerpen van navigatie, in het vertellen van een interactief verhaal, in de rol van realisme, kennis ontwikkelen van best practices met betrekking tot zinvolle toepassingen van virtuele werelden, verwerven van basisvaardigheden in het ontwerpen in 3-D Studio Max en het ontwikkelen van een virtuele wereld.

De theoretische achtergrond

De theorie werd via het intranet van de opleiding aangeboden. In de eerdere versie van de colleges Virtuele Werelden was de discussie op grond van de artikelen echter niet van de grond gekomen en haalden te weinig studenten het theoretische tentamen. Bij de herziene versie - die hier aan de orde is - werd daarom besloten in de fysieke klas expliciet op de aangeboden artikelen in te gaan alvorens over te gaan tot discussie.

³ De opleiding heeft de ervaringen met Active Worlds toegelicht tijdens een evaluatiebijeenkomst van dit Surfproject in juni 2008

De opdracht

Als opdracht bouwden de studenten in tweetallen één huis in de stijl van Asterix en Obelix in 3-D Studio Max (zie figuur 1). De huizen werden vervolgens in een gemeenschappelijke Active Worlds-'wereld' geplaatst. De studenten bouwden dus samen aan een dorp. Ze bedachten daarnaast een zinvolle communicatieve toepassing voor hun huis (het huis van de visboer, de zanger, de druïde etc.). Dit concept moest worden uitgevoerd. Daarnaast ontwikkelden ze een concept voor een zinvolle toepassing voor het dorp als geheel. Het laatste bleef bij een concept.



Figuur 1:

Virtuele Werelden, 2008 - Huis van de smid, huis van de bard.

Constructivistische onderwijsopvatting

De module was opgezet vanuit een constructivistische onderwijsopvatting zoals die aan de Faculteit Communicatie & Journalistiek van Hogeschool Utrecht wordt gehanteerd. Deze opvatting houdt in: het proces van ontwikkelen, delen en evalueren van kennis, vaardigheden en begrip in een samenwerkende omgeving door interactie met die omgeving.

De kennis wordt geconstrueerd op grond van een ervaring die een mentale activiteit inhoudt. (Orey, 2001). Belangrijk zijn daarbij begrippen als 'flow' en een uitdagende ervaring, training, experimenteren, deel zijn van een samenwerkende omgeving, leren door te doen, observeren van anderen, motivatie en reflectie. De virtuele leeromgeving was naast de fysieke leeromgeving gekozen om te onderzoeken of een dergelijke crossmediale aanpak het constructivistische leerproces zou kunnen ondersteunen.

De resultaten

Na de cursus hebben de studenten een vragenlijst ingevuld, samengesteld door de vakgroep MCM in het kader van het onderzoek naar experience design. De vraag stond centraal in hoeverre er sprake was geweest van een leeromgeving die had bijgedragen aan een betekenisvolle ervaring. Het bleek dat de studenten de colleges uitdagender vonden dan ze van tevoren hadden gedacht. Vrijwel alle respondenten gaven aan actief, betrokken en verrast te zijn geweest. Dit ondanks de technische problemen. Deze werden gezamenlijk overwonnen. Er was veel ruimte voor experiment en het observeren van het werk van anderen. Daarbij was sprake van een grote motivatie, vooral door de wisselwerking tussen de fysieke en de virtuele leeromgeving (Benvenuti et. al, 2008). Het experiment was duidelijk zeer succesvol. De wisselwerking tussen de virtuele en de fysieke leeromgeving bleek een omgeving op te leveren die tegemoet kwam aan de eisen die het constructivistisch onderwijs aan een inspirerende leeromgeving stelt. De leerdoelen werden meer dan gemiddeld gehaald. Alle studenten slaagden voor het theoretische tentamen en alle studenten rondde de opdracht met een voldoende af.



Figuur 2:

Virtuele Werelden 2008, dorpsgezicht

IS ER EEN TOEKOMST?

Duidelijk zal zijn dat dit slechts een eerste aanzet is tot samenwerking. Wij hebben alleen gefocust op de vraag in hoeverre de aangeleverde virtuele omgeving kan bijdragen aan het bewerkstelligen van een betekenisvolle ervaring. Veel vragen zijn blijven liggen, bijvoorbeeld: wat maakte deze cursus tot zo'n succes? Was dit vooral de combinatie klassikaal onderwijs/virtuele wereld of speelden ook de didactische setting, de gehanteerde dorpsmetafoor, het meedoen aan een experiment en de samenstelling van de groep een belangrijke rol? De antwoorden op deze vragen zouden aanwijzingen geven voor het ontwerp en gebruik van e-learning-software. Een ander aspect, dat in dit onderzoek nog helemaal buiten beeld is gebleven is de ervaring van de docenten. In het onderzoek naar het gebruik van innovatieve onderwijsvormen gerelateerd aan ontwikkelingen in de ICT is dit een internationale trend die bevestigd werd door de inhoud van het CELDA Congres in oktober 2008. De student staat centraal. Dit is niet terecht. De docent is een gebruiker van de tool die een bepalende rol speelt in de toepassing ervan. Om tot voldoende resultaat te komen zou onze samenwerking langdurig en structureel moeten zijn.

Maar zou dat wenselijk zijn? Kan op deze manier zinvolle innovatie ontstaan op het snijvlak van deze twee disciplines?

Cultuurverschillen

Soms merken wij dat de behoeften van de twee vakgebieden media en ICT te ver uit elkaar liggen. Lang niet alle technische vragen waar de mediaspecialist mee worstelt zijn van belang voor de informaticus. De meeste zijn te specifiek voor een gekozen platform, configuratie of toepassing. Optimaal gebruik van software is een reëel probleem voor de mediaspecialist. Zo konden de studenten van de module Virtuele Werelden veel van hun ideeën niet realiseren door problemen met de toepassing van de techniek. Dat is jammer voor alle partijen. De informaticus zal de wenkbrauwen fronsen omdat er kansen blijven liggen. Bacheloropleidingen in de creatieve sector hebben echter andere belangen dan de puntjes op de technische i's te zetten. Zij willen de nieuwste technologieën toepassen, desnoods niet volgens alle regelen der kunst. Het eerste wil de informaticus ook, het laatste is hem een doorn in het oog.

In de samenwerking tussen informatici en mediaspecialisten treedt veel ruis op. De disciplines kennen diametraal verschillende onderwijs- en onderzoeksculturen. Docenten aan creatieve opleidingen die verdieping zoeken van de technische aspecten van hun werk zullen moeite hebben om hun expertise te verzilveren. Techniek valt daar immers onder de vileine aspecten van het vak. Informatici die de 'zachte' kanten van hun vakgebied verkennen moeten binnen de eigen faculteit verdedigen waarom zij energie steken in onderzoek dat in het gunstigste geval als 'vaag' wordt aangeduid. Het overbruggen van deze verschillen kost veel energie, dat is ook onze ervaring. Structurele samenwerking veroorzaakt structurele ruis, die in de moederorganisaties niet wordt verzacht door een gemeenschappelijke fascinatie voor het

vakgebied. Is al die ruis ons de kruisbestuiving waard? Moeten we hier een latrelatie van maken of kunnen we toch beter elk weer onze eigen weg gaan?

Het antwoord van de toehoorders

Deze vraag hebben wij gesteld aan de aanwezigen tijdens het NIOC Congres 2009. Tot onze verbazing kregen wij opstand in de zaal. Het publiek eiste dat er een derde optie werd toegevoegd aan onze metafoor: de langdurige samenlevingsvorm. Na afloop van de discussie is er gestemd. Van de 26 aanwezigen was niemand voor de scheiding; 8 personen stemden voor de Latrelatie en de rest, dus de overgrote meerderheid, koos de langdurige samenlevingsvorm.

Wij zijn aangenaam verrast met de massale steun van Informaticadocenten voor een vorm van structurele samenwerking die verder gaat dan onze stoutste dromen. Echter, om een langdurige samenlevingsvorm te kunnen realiseren in het Nederlandse hoger onderwijs zullen veel faculteitsdirecties veel water bij de wijn moeten doen.



- [1] Bakker, G., F. Meulenberg and J. de Rode (2003). 'Truth and Credibility as a Double Ambition: Reconstructions of the Built Past, Experiences and Dilemmas.' *The Journal of Visualisation and Computer Animation* 2003(14): 159-167.
- [2] Benvenuti, L., E.J. Hennipman, E.J. Oppelaar, G.C. van der Veer, B. Cruisberg, G. Bakker, (2008), *Experiencing Education with 3D Virtual Worlds*, paper presented at the IADIS Int. conference CELDA2008, Freiburg, Germany, October 13 to 15, 2008.
- [3] Benvenuti, L., E.J. Hennipman, E.J. Oppelaar, G.C. van der Veer, B. Cruisberg, G. Bakker, 'Experiencing and Learning with 3D Virtual Worlds', in: J. Michael Spector, Dirk Ifenthaler, Pedro Isaias, Kinshuk, & Demetrios G. Sampson (Editors) *Learning and instruction in the digital age* (to be published).
- [4] Benvenuti, L., G.C. van der Veer (2009): 'Multimedia design kun je leren, kunnen wij het ook doceren?'. 13th Computer-Human Interaction Netherlands Conference, 2009, Leiden, pp. 39-41,
- [5] Dede, C. (1995). 'The Evolution of Constructivist Learning Environments: Immersion in Distributed, Virtual Worlds.' *Educational Technology* 35(5): 46-52.
- [6] Informatica (2007), *Visitatierapport Quality Assurance Netherlands Universities (QAUNU)*, Utrecht
- [7] Orey, M.(Ed.). (2001). 'Emerging perspectives on learning, teaching, and technology.' Retrieved December 13, 2008, from <http://projects.coe.uga.edu/epltt/>
- [8] Veer, G.C. van der (2008), 'Human-Computer-Society at the QUN, Vision of the Domain', Heerlen
- [9] Van Vliet, H. e.a. (2008) *Idola van de crossmedia*, Utrecht.
- [10] Vyas, D. and G.C. van der Veer (2006). 'Rich evaluations of entertainment experience: bridging the interpretational gap'. 13th European Conference on Cognitive Ergonomics, 2006, Switzerland. pp. 137-144.

Bedrijven als 'Partners In Education' in het onderwijs van Fontys Hogeschool ICT



Frans van Alebeek - docent Bedrijfskundige Informatica, Fontys Hogeschool ICT

Ruud Ermers - curriculumeigenaar ICT & Technology, Fontys Hogeschool ICT

SAMENVATTING

Fontys Hogeschool ICT heeft een belangrijke spilfunctie voor de regio Eindhoven, het technologisch middelpunt van Nederland. Het bedrijfsleven schreeuwt om pasafgestudeerden en Fontys Hogeschool ICT kan maar mondjesmaat leveren. Een structurele samenwerking met een aantal bedrijven, betrokken bij alle aspecten van de opleiding, is een mogelijke oplossing die door Fontys Hogeschool middels het concept 'Partners In Education' wordt vormgegeven.

Dit artikel gaat in op het gevolgde proces, het achterliggende onderwijsconcept, de eerste resultaten en ten slotte op enkele geconstateerde knelpunten.

TREFWOORDEN

Curriculumontwikkeling, bedrijfsparticipatie, kennisdeling

INLEIDING

Bij Fontys Hogeschool ICT (FHICT) studeren jaarlijks ruim 350 ICT'ers af. Voor de regio Eindhoven, het technologisch middelpunt van Nederland, is dit veel te weinig, zo geven ook bedrijven aan. De hogeschool wil de uitstroom met vijftig procent verhogen door de instroom te vergroten en de uitval te beperken. Daarom zal de aantrekkelijkheid van een ICT-studie al vroeg

onder de aandacht gebracht moeten worden van potentiële studenten, moet de opleiding aantrekkelijk zijn, zowel inhoudelijk als uiterlijk en dient de opleiding goed aan te sluiten op de gevraagde kwalificaties van het beroepenveld. FHICT heeft al tientallen jaren op min of meer gestructureerde wijze veel contacten met dat beroepenveld, maar gezien de ambities is gezocht naar een nieuwe vorm van samenwerking. Directeur Ad Vissers schreef in september 2007 de notitie '*Van gedeeld belang naar gedeelde verantwoordelijkheid: Partnership met Fontys Hogeschool ICT*' met het concept Partners in Education. Tijdens twee feestelijke tekenbijeenkomsten in februari en mei 2008 heeft Fontys Hogeschool ICT (FHICT) met een dertigtal (grotendeels) regionale (ICT)bedrijven een samenwerking voor vier jaar vastgelegd. Met elk bedrijf zijn concrete afspraken gemaakt, die ook vastgelegd zijn in een convenant. Voorbeelden van afspraken:

- het beschikbaar stellen van expertise;
- actieve participatie in en medeverantwoordelijk zijn voor een onderwijsblok van tien weken;
- excursies en uitwisseling van docenten en medewerkers van de bedrijven;
- activiteiten organiseren ter oriëntering voor leerlingen van het voortgezet onderwijs en het (mede)beoordelen van producten van studenten.

In deze bijdrage willen we een beeld schetsen van de aanpak, wat voorbeelden geven van de concrete invulling, reacties van partners en nog openstaande knelpunten en dilemma's.

VISIE EN ONDERWIJSMODEL:

BASISPROFIELEN & INNOVATIEROUTES

Om een belangrijke speler in de ICT-markt te blijven zijn bedrijven op zoek naar veel hoogopgeleide medewerkers. FHICT vormt een belangrijke schakel in de keten die leidt van middelbare scholieren die nog niet goed weten wat zij willen, tot afgestudeerde ICT'ers, die goed zijn voorbereid op de marktvraag vanuit het bedrijfsleven. Die wereld van ICT is onderhevig aan grote veranderingen, nieuwe methoden en technieken volgen elkaar in snel tempo op. Om op bovengenoemde aspecten goed in te kunnen spelen is de volgende onderwijsvisie ontwikkeld:

Zorg dat je studenten een uitdagende leeromgeving aanbiedt die hen op de juiste manier aanspreekt, maar borg uiteraard de kwaliteit van de opleiding Bachelor of ICT. Zorg dat je op ieder moment studenten opleidt tot datgene waar de markt op dat moment naar vraagt.

Om aan deze visie invulling te geven is het volgende onderwijsmodel opgesteld:

- Bied binnen de brede Bachelor of ICT vier basisprofielen aan, die door de tijd heen hetzelfde blijven. Deze profielen zijn:
 - ICT & Software Engineering
 - ICT & Technology
 - ICT & Business
 - ICT & Media
- Bied een aantal innovatieroutes aan die door de tijd kunnen wijzigen. Momenteel worden de volgende innovatiegebieden aangeboden:
 - ICT & Mobile Computing
 - ICT & Lifestyle
 - ICT & Management & Security
 - ICT & Game Design & Technology

Door het slim kiezen van onderwijsactiviteiten binnen deze profielen en routes kan een student een eigen,

voor hem/haar uitdagende studieroute doorlopen.

Door het wisselen van innovatiegebieden kan de Hogeschool inspelen op de marktvraag. Daar ligt altijd een profiel aan ten grondslag, waardoor de Body Of Knowledge wordt gewaarborgd.

Om aan deze visie invulling te geven is een goede en structurele samenwerking met toonaangevende partnerbedrijven bijna onontbeerlijk. Enerzijds kunnen die Partners meehelpen de studenten een aantrekkelijke leeromgeving aan te bieden, door middel van innovatieve labs en real life cases, anderzijds kunnen (en zullen) diezelfde partners duidelijk aangeven waar de markt naar toe beweegt en kan de Hogeschool door de nauwe samenwerking veel sneller een kwalitatief hoogwaardig stuk nieuw onderwijs ontwikkelen.

EEN VOORBEELD: INVULLING PARTNERSHIP BIJ PROFIEL ICT & TECHNOLOGY

Een van de aspecten waar de samenwerking met bedrijven in gezocht is, is het opzetten van het nieuwe curriculum. Deze curriculumvernieuwing vindt plaats sinds 2007 en bevindt zich qua ontwikkeling nu in het tweede jaar. In dit tweede jaar is bij het profiel ICT & Technology een viertal Partners betrokken, per kwartaal één. Het vinden van deze Partners ging zeer eenvoudig, gezien de goede contacten die Technische Informatica had met het bedrijfsleven, via een zogeheten klankbordgroep. Het waren de bedrijven Logica, Sioux, Actemium en TASS die bereid waren te participeren in het tweede jaar, als de studenten nog ver van de eindstreep verwijderd zijn. Bij het invullen van het partnership werd door de betreffende curriculumverantwoordelijke (Ruud Ermers) een aantal doelen geformuleerd zoals:

Curriculum Ontwikkeling: Partners zijn gevraagd een stuk curriculum mede te ontwikkelen waar zij zelf expert in zijn. Hierdoor kunnen zij zowel bijdragen in de Body Of Knowledge omtrent het onderwerp als aangeven hoe zij hier bij huidige projecten zelf mee om gaan.

Qua onderwijsuitvoering is hierbij gekozen dat de Body Of Knowledge verzorgd wordt door docenten van Fontys, en dat vooral de innovaties aangeboden worden vanuit de bedrijven. (Dit laatste vindt overigens plaats in de vorm van een gastcollegecyclus waarbij

ook van de studenten een duidelijke bijdrage verwacht wordt ten behoeve van portfoliovulling).

Projectuitvoering: In ieder kwartaal maakt een project onderdeel uit van het kwartaalonderwerp. Het onderwerp van het project is in overleg met de Partner vastgesteld, zodat het een uitdagende, en ook realistisch case is. Bij projectuitvoering is de Partner betrokken als klant of expert. De klantrol in deze is die van een klant die het technische systeem goed kent en dus ook bijdrage kan leveren in gewenste architectuur, ontwerp en cetera. In beide gevallen zal de partnerrol een zodanige bijdrage opleveren dat studenten zien hoe in de praktijk soortgelijke problemen worden opgelost, zowel technisch inhoudelijk als procesmatig.

Architectuursessie: Een van de punten waar studenten later zeker mee te maken gaan krijgen zijn vragen rondom architectuur, ontwerpmethoden en cetera. In een zogeheten architectuursessie (die meestal bij het bedrijf plaatsvindt) wordt studenten een door Partner ontwikkeld product voorgelegd. Vervolgens wordt in een interactieve sessie met studenten de architectuurkeuzes doorgesproken, de opgeleverde documentatie besproken, modellen getoond, wellicht enkele ontwerp patronen behandeld om zo inzicht aan de student te geven over hoe ontwikkeling in de praktijk plaatsvindt.

Tot nu zijn vier kwartalen volgens deze richtlijnen ingevuld, waarbij bedrijven vaak nog met meer ideeën kwamen, zoals het organiseren van een microproces-

sorcursus of het aanbieden van andere cursussen. Het ligt in de lijn dit concept verder uit te bouwen richting derde en vierde jaar. Tevens zullen (in een afgeslankte vorm) partnerbedrijven betrokken gaan worden bij de propedeuse. Per studiejaar zal een lokaal worden ingericht waar de partnerbedrijven en hun projecten permanent zichtbaar zijn.

RESULTATEN

Zoals eerder aangegeven is het concept 'Partners In Education' breed ingezet binnen Fontys Hogeschool ICT. Dit blijkt uit het volgende:

Onderwijs

Net zoals dat bij het profiel ICT & Technology is ingevoerd zal bij iedere kwartaal van iedere studieroute ('basisprofiel' c.q. 'innovatieroute') een 'Partner' actief betrokken worden. Zo hebben Isah, Ericsson en TomTom zich bereid verklaard de studieroute 'Mobile Computing' mee op te zetten, waarin allerlei aspecten van mobiele systemen in combinatie met ERP aanbod komen. Zo wordt niet alleen gekeken naar het programmeren voor mobiele devices, maar bijvoorbeeld ook naar de veranderingen in het werkproces van medewerkers na invoering van een mobiel systeem. Simac ICT Nederland verzorgt mede de studieroute IMS (ICT, Management & Security). Daarnaast vind ook samenwerking met Partners plaats in het 'oude onderwijs', zoals bij het vak BPI (business process improvement) voor laatstejaarsstudenten Bedrijfskundige Informatica. Partner SNS-bank (IT-afdeling) geeft een workshop in 's-Hertogenbosch over 'ambities, masterplan & architectuur' en de daarvoor ingerichte bedrijfsprocessen en projecten. Deze kijk in de keuken leverde boeiende discussies met de studenten op over zaken zoals het SNS-imago, de SWOT's, de bankencrisis en het inrichten en managen van (de change van!) bedrijfsprocessen.



Ook tijdens een congres gaat het werk gewoon door

Laboratoria

In samenwerking met Simac ICT Nederland heeft FHICT een spiksplinternieuw ICT-securitylab. Docenten en studenten doen hier onderzoek naar ICT-security-onderwerpen zoals informatiebeveiliging, hacking&cracking, netwerkbeveiliging, VoIP-security en toegangsbeveiliging. Ook worden er, om bijvoorbeeld de veiligheid van een webserver en websites te testen, securityscans voor bedrijven uitgevoerd. Door een team van docenten van Fontys en Simac-technici

is gezamenlijk gewerkt aan een ontwerp, aanschaf van apparatuur en inrichting van het lab. Studenten werken aan onderwerpen die variëren van 'wireless hacking' en 'modern day firewalling' tot 'fysieke toegangscontrole met vingerafdrukken'. Een tweede lab wordt momenteel opgezet in samenwerking met Accenture. Dit zal een kopie van het Technology Lab zijn, zoals Accenture die kent in onder andere India en Frankrijk.

Professionalisering

TASS verzorgt een training op het gebied van FPGA ontwikkeling. In de DK ontwikkelomgeving van Agility (www.agilityds.com) zijn engineers in staat door middel van de taal Handel-C in zeer korte tijd FPGA implementaties te realiseren zoals een equalizer of een Mandelbrotalgoritme.

Overig

Na een half jaar blijkt dat met bijna alle bedrijven concrete 'Partneractiviteiten' lopen. Registratie van partnergegevens vindt centraal plaats bij FHICT (Centrum IT voor Zakelijke Dienstverlening en Stages/Afstuderen), er is een intranetsite (<http://fhict.fontys.nl/PiE/default.aspx>) waar alle Partners en overeenkomsten zichtbaar zijn voor medewerkers en studenten, driemaal per jaar wordt een 'Partnermail' gemaakt waarin deze partneractiviteiten worden geschetst voor studenten en medewerkers. Partners mogen tweemaal per jaar een (gerichte) mailing naar studentgroepen doen met aanbod van stage/afstudeeropdrachten en banen. Een tiental Partners is betrokken bij scriptieprijs, 13 bedrijven waren aanwezig bij de alumnidag in het PSV-stadion (en gaven een sponsorbedrag). Bij de laatste open dag leverde de aanwezigheid van een tiental Partners een wezenlijke bijdrage aan het informeren en enthousiasmeren van de bezoekende scholieren (en hun ouders!).

EERSTE REACTIES VAN PARTNERS

De Partners blijken blij zijn met het samenwerkingsinitiatief dat Fontys Hogeschool ICT heeft genomen. Een paar reacties:

Annemieke Verberne, HR-consultant bij Simac: 'Wij erkennen meer dan ooit dat het van groot belang is de instroom van jonge professionals te stimuleren en de dalende trend concreet aan te pakken. Natuurlijk vinden wij ook de actualiteit van het onderwijs belangrijk, dat maakt dat afgestudeerden of afgestudeerders

goed in staat zijn snel te schakelen en inzetbaar zijn. Studenten kunnen bij Simac een kijkje in de keuken komen nemen of afstuderen. Daarnaast neemt het bedrijf deel aan expertisegroepen op het gebied van beheer en security en levert het gastdocenten. Hierbij verzorgen medewerkers voor een bepaalde periode een collegereeks rond een vooraf afgesproken thema. Tot slot richt de dienstverlener een state-of-the-artnetwerklaboratorium in en biedt hij ondersteuning bij het beheer en exploitatie van het lab. 'In ruil hiervoor worden wij op de hogeschool gepresenteerd als betrouwbare partner, met een hoog kennis- en kwaliteitsniveau en zijn wij aanwezig op opendagen en bedrijvendagen om in contact te komen met potentiële nieuwe arbeidskrachten. Bovendien bieden eerdergenoemde activiteiten ook mogelijkheden voor onze eigen medewerkers. Zij krijgen op deze manier de kans om op een andere manier dan gebruikelijk hun kennis te delen met studenten en docenten en leren daar op hun beurt ook weer van.'

Heino Peters, Competence Manager bij Logica: 'Wij doen hieraan mee omdat het ons op een aantal terreinen verder helpt: er is een groot tekort aan nieuwe instroom in de technische functies. Via deze samenwerking willen we Fontys ook helpen meer mensen te interesseren voor dit beroepsveld. Wij laten graag zien welke werk je later kunt doen als je een technische opleiding gaat volgen. Met deze samenwerking zetten we ons bedrijf Logica als mogelijke werkgever voor technische automatisering op de kaart bij studenten van de Fontys Hogeschool. Op dit moment werken we al intensief samen aan de invulling van de leerstof voor het eerste semester van het tweede studiejaar. In september 2008 zullen deze studenten met ons bedrijf gaan kennismaken. Vooralsnog richten we ons even op de ontwikkeling van embedded en realsoftware, maar we beperken ons daar op termijn zeker niet toe. Dit convenant is voor ons belangrijk omdat het zowel bij Fontys als bij

ons de vrijblijvendheid die tot dusver bij dit soort kruisbestuiving gold weghaalt. Zo wordt er meer mogelijk en kun we beiden meer bereiken.'

Nicole Hobbelen, graduate recruiter bij Accenture Technology Solutions: 'Accenture Technology Solutions (ATS) is zeer enthousiast over het onder tekende convenant. Op dit moment zijn we in overleg om concreet een aantal gebieden samen op te pakken en te ontwikkelen voor het komend jaar. Eerste gedachten zijn: Innovatie Lab, SOA/BPEL, CRM, Business Intelligence en ERP. In december 2008 werd een bezoek gebracht aan het Technology Lab in Nice (Frankrijk) met duidelijke voorbeelden van een 'Next Generation Bank' en nieuwe retailtoepassingen. Er wordt nu bekeken of en hoe een replicatie van onderzoeks- en demonstratieopstellingen bij FHICT kan worden gerealiseerd'.

KEUZES EN KNELPUNTEN

Keuze van Partners

Het bedrijvenbestand van FHICT bestaat uit ruim 2500 bedrijven. Bedrijven waar we de afgelopen 35 jaar concreet mee is samen gewerkt, voornamelijk door uitvoering van stage- en afstudeeropdrachten, maar ook zakelijke dienstverlening en onderwijsprojecten. Het maken van een expliciet onderscheid tussen dertig Partners en 'overige bedrijven' is een hachelijke zaak en werd zorgvuldig aangepakt. Al heel snel bleek dat heel veel bedrijven Partner wilden worden, ook toen we vroegen om een 'inspanningsverplichting in principe' van minimaal 100 uren per jaar. Criteria waarmee de eerste (top-down)selectie gedaan werd waren:

- eerder bewezen samenwerkingsresultaten;
- spreiding over de verschillende 'basisprofielen en innovatieroutes';
- regionale aanwezigheid (straal van 40 kilometer).

Masseren en wildgroei

Echte betrokkenheid van Partners ontstaat door structurele inzet bij het onderwijs. Dit vergt een structurele visie door de 'curriculumeigenaar' per basisprofiel en innovatiegebied (bottom-up) en inzicht wat Partners kunnen bieden (top-down). Hier is een verschil tussen de verschillende studieroutes waardoor potentieel onderwijsaanbod van Partners niet wordt gebruikt. Top-down 'informatie en massage' blijft nodig en desondanks is bij enkele Partners nog geen structurele inzet. Gewekte verwachtingen 'drukken' en worden gemanaged door extra evaluatiegesprekken van deze Partners en de directie van FHICT. Anderzijds zijn er (soms te) spontane onderwijsinitiatieven van docenten waarbij aan nieuwe bedrijven partnershipbeloften worden gedaan die moeilijk zijn terug te draaien en dus groeit de lijst met Partners. En dan zijn er nog de bedrijven die op de 'partnership-wachlijst' staan en ondertussen betrokken zijn bij onderwijsactiviteiten. Een dilemma: concrete onderwijsvernieuwing ontstaat bij de docenten en 'curriculumeigenaren' van de (8!) basisprofielen en innovatieroutes, bottom-up dus met Partners. Anderzijds dient 'top-down' gelet te worden op eerder en elders gewekte verwachtingen bij bedrijven en de communicatie met alle Partners.

TOT SLOT

De inbreng van Partners is het afgelopen jaar voor FHICT zeer waardevol geweest. Ook voor andere beroepsopleidingen lijkt het concept Partner in Education bruikbaar, mits er sprake is van een duidelijk onderwijsmodel (met curriculumeigenaren of soortgelijk) en een beroepenveld dat uitkijkt naar en geld/tijd over heeft voor, de opleiding van nieuw jong hbo-talent.



Een virtueel computersecuritylab

Harald Vranken - Open Universiteit Nederland, Faculteit Informatica. harald.vranken@ou.nl, tel. 045 - 576 2373

Herman Koppelman - Open Universiteit Nederland, Faculteit Informatica. Postbus 2960, 6401 DL Heerlen

INTRODUCTIE

Aan de Open Universiteit Nederland is in 2007-2008 een cursus ontwikkeld op gebied van security en IT. Een prominent onderwerp in deze cursus is de veiligheid en de beveiliging van computernetwerken. Hierbij is een virtuele omgeving ontwikkeld (het virtuele computersecuritylab) waarin een student op zijn eigen pc een virtueel computernetwerk kan configureren en simuleren. In dit netwerk kan de student vervolgens experimenten en opdrachten uitvoeren op gebied van security. De student kan in het virtuele lab verschillende rollen aannemen. In de rol van hacker kan hij aanvallen uitvoeren op het netwerk; in de rol van systeembeheerder of beveiligder kan hij het netwerk tegen aanvallen beveiligen; in de rol van gebruiker van computersystemen in het netwerk kan hij de effecten van aanvallen en beveiligingsmaatregelen ervaren. Het virtuele lab beoogt enerzijds om de cursus aantrekkelijk te maken, wat een positief effect heeft op de motivatie van studenten, en anderzijds om de opgedane theoretische kennis met praktische opdrachten te verdiepen en te borgen.

De indeling van dit artikel is als volgt. In sectie 1 beschrijven we de inhoud en de didactische opzet

van de cursus Security en IT. In sectie 3 geven we een beknopt overzicht van de verschillende soorten computersecuritylabs zoals die in de afgelopen jaren in de wetenschappelijke literatuur zijn beschreven. Aan de hand daarvan positioneren we ons virtuele computersecuritylab. In sectie 4 gaan we nader in op de architectuur en de implementatie van het virtuele computersecuritylab. In sectie 5 lichten we kort de opdrachten toe die studenten in het virtuele lab uitvoeren. Sectie 6 sluit het artikel af.

DE CURSUS SECURITY EN IT

De cursus Security en IT geeft een breed overzicht van het vakgebied security, waarbij de nadruk ligt op de technische aspecten van beveiliging. De cursus bestrijkt de basisprincipes van cryptografie en de beveiliging van software, besturingssystemen, databases en computernetwerken. Bij elk onderwerp wordt achtereenvolgens bekeken: welke kwetsbaarheden zijn er, welke aanvallen zijn daardoor mogelijk, hoe kunnen we deze aanvallen voorkomen en hoe kunnen we geslaagde aanvallen ontdekken. Daarnaast is er ook aandacht voor beheer, economische aspecten, privacy en actuele ontwikkelingen. De cursus is gebaseerd op het Engelstalige tekstboek *Security in Computing* van C.P. Pflieger en S. Lawrence Pflieger (4e editie, 2006), en enkele hoofdstukken uit *Computer Security* van D. Gollmann (2e editie, 2006). Daarnaast heeft de Open Universiteit twee cursusboeken ontwikkeld die de leidraad vormen voor de student. In deze cursusboeken is de leerstof gestructureerd in een aantal leereenheden. Elke



leereenheid bevat een afgerond deel van de leerstof dat in 4 tot 8 uur bestudeerd kan worden en bestaat uit een korte inleiding, een overzicht van de leerdoelen, studeeraanwijzingen waarin wordt aangegeven welke delen uit de tekstboeken bestudeerd moeten worden, toelichtingen en aanvullingen bij de tekstboeken, opgaven, praktische opdrachten, een afsluitende zelftoets en een uitgebreide terugkoppeling met antwoorden bij de opgaven, opdrachten en zelftoets. Met behulp van deze cursusboeken kan de student geheel zelfstandig de leerstof bestuderen. De cursus is daarom vooral geschikt voor afstandsonderwijs zoals dat door de Open Universiteit wordt verzorgd. Bij de cursus hoort verder een elektronische leeromgeving (ELO) waar de student aanvullende studiematerialen kan vinden, zoals verwijzingen naar achtergrondinfor-

matie, actuele mededelingen, software downloads, oefententamens en een discussieruimte. Het contact tussen studenten en docenten vindt plaats via de ELO en e-mail.

In de cursus voeren de studenten diverse praktische opdrachten uit. Bij het onderwerp cryptografie maken ze gebruik van interactieve webpagina's, bij het onderwerp bufferoverflows voeren ze simulaties uit, en bij het onderwerp netwerken voeren ze diverse opdrachten uit in het virtuele computersecuritylab. De praktische opdrachten in het virtuele lab beslaan circa 10% van de studietijd.

De cursus is sinds 2008 een verplicht onderdeel van de bacheloropleiding Informatica aan de Open Universiteit. Masterstudenten Computer Science

Opname van de Openings sessie



Samen een borrel drinken hoort er ook bij



kunnen deze cursus doen in een schakelprogramma of als keuzevak. De cursus is begin 2008 voor het eerst door 15 studenten gevolgd. De ervaringen van de studenten zijn uitgebreid geëvalueerd en waren doorgaans zeer positief. Aan de hand van deze ervaringen is het cursusmateriaal verder verbeterd. De cursus is ook toegankelijk voor studenten die niet de gehele bachelor- of masteropleiding volgen maar slechts een enkele cursus doen uit interesse of als bijscholing. Om de cursus toegankelijk te maken voor zulke losse cursisten, zijn de ingangseisen voor de cursus beperkt gehouden. Als benodigde voorkennis is globale kennis vereist van de architectuur van het internet en computernetwerken, relationele databases, besturingssystemen en een programmeertaal. Bij het onderwerp cryptografie ligt de nadruk op de werking van algoritmes en toepassingen, waarbij de wiskundige achtergrond bewust is vermeden. (Het tekstboek bevat een hoofdstuk dat dieper op de wiskunde ingaat, maar dat is geen verplichte leerstof. In de bacheloropleiding

wordt overigens in een aparte wiskundecursus wel aandacht besteed aan getaltheorie waarop cryptografie is gebaseerd.)

COMPUTERSECURITYLABS

In veel onderwijsinstellingen zijn computersecurity-laboratoria te vinden. Dit zijn veelal ruimtes waarin computernetwerken zijn ondergebracht die volledig zijn geïsoleerd van de rest van de wereld (bijvoorbeeld [1]). Deze volledige isolatie voorkomt dat activiteiten in het lab via het netwerk de buitenwereld kunnen bereiken. Zulke 'fysieke labs' zijn echter ongeschikt voor afstandsonderwijs, omdat studenten vanwege tijd- en plaatsbeperkingen het lab niet kunnen bereiken. Een alternatief is een fysiek lab dat door studenten op afstand bereikt kan worden via het internet. Het lab bestaat dan uit een geïsoleerd netwerk met een enkele verbinding naar de buitenwereld. Deze externe verbinding dient zeer goed beveiligd te zijn, bijvoorbeeld door middel van een firewall en toepassing van

SSH of VPN. Voorbeelden van zulke labs zijn [2][3][4][5]. Hoewel zulke labs op afstand bereikbaar zijn, zijn er toch beperkingen voor toepassing in afstandsonderwijs. Bij grote aantallen studenten moet er een reserveringssysteem opgezet worden, waarin studenten van tevoren een tijdslot kunnen reserveren om het lab te gebruiken. Studenten aan de Open Universiteit kunnen op elk moment met de cursus beginnen, waardoor het aantal studenten op een gegeven tijdstip zeer moeilijk in te schatten is. Het bepalen van de benodigde capaciteit in een lab is dan een lastige klus, en zal leiden tot overdimensionering. Naast de administratieve complexiteit rondom een reserveringssysteem, bestaat de kans dat dit studenten ook belemmert in hun vrijheid van studietijd en studietempo. Het beheer van fysieke labs was in het verleden vaak problematisch. Studenten werken in het lab met root rechten en kunnen naar believen systeemconfiguraties aanpassen. Na afloop van een sessie is het daarom noodzakelijk om de configuraties van systemen weer op orde te brengen, wat soms zelfs complete herinstallaties vereist. In veel labs wordt daarom tegenwoordig gebruik gemaakt van gevirtualiseerde omgevingen, waarmee herconfiguratie en herinstallatie van systemen aanzienlijk vergemakkelijkt en zelfs geautomatiseerd kan worden (bijvoorbeeld [6][7]).

Als alternatief voor een fysiek lab op afstand is voor de cursus Security en IT een stand alone virtueel lab ontwikkeld, dat bestaat uit een virtuele omgeving die elke student op zijn eigen pc kan installeren. De benodigde (open source en gratis) software voor het virtuele lab wordt op een dvd aan de student verstrekt. De student hoeft dus geen verbinding te maken met een lab op afstand. Deze vorm vermijdt exploitatie en beheer van een lab op afstand, en kan voor een ongelimiteerd aantal studenten toegepast worden. Onderhoud en beheer van het lab wordt nu bij de individuele student gelegd, maar door toepassing

van virtualisering is dit een minimale taak die beperkt is tot enkele muisklikken en nauwelijks tijd vergt. Het virtuele lab draait op nagenoeg alle Linux- en Windowsplatforms, en is daarom geschikt voor vrijwel alle studenten-pc's. Deze opzet is uitermate geschikt voor afstandsonderwijs, en kan zonder problemen ook bij contactonderwijs ingezet worden. Deze opzet sluit nauw aan bij [8], waar de voordelen van (open source) virtuele omgevingen bij toepassing in (afstand)-onderwijs verder worden toegelicht.

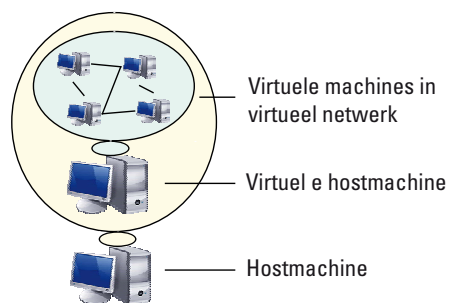
ARCHITECTUUR VAN HET VIRTUELE LAB

Het virtuele computersecuritylab maakt gebruik van virtualisatiesoftware. Deze virtualisatiesoftware introduceert een extra softwarelaag met bijbehorende interface die logische abstractie van de onderliggende hardware en software realiseert [9]. Het principe van virtualisatie dateert al uit de jaren zestig (IBM 370 mainframes), maar heeft sinds eind jaren negentig aanzienlijke, hernieuwde aandacht. Tegenwoordig is er tal van softwareomgevingen beschikbaar waarmee meerdere besturingssystemen in aparte, geïsoleerde omgevingen op een enkel systeem kunnen worden uitgevoerd. Slimme softwarearchitecturen en voorzieningen in de hardware van microprocessors minimaliseren de afname in performance ten gevolge van virtualisatie.

Het virtuele computersecuritylab bestaat uit een virtuele omgeving op de pc van de student. In deze virtuele omgeving kan de student diverse virtuele machines instantiëren en aan virtuele netwerken koppelen. De student kan op deze virtuele machines met root rechten werken en kan de virtuele netwerken configureren.

De architectuur van het virtuele lab is tot stand gekomen door overweging van de volgende uitgangspunten. Allereerst dient de virtuele omgeving geschikt te zijn voor diverse Windows- en Linuxplatforms, zodat nagenoeg elke student deze thuis op zijn pc

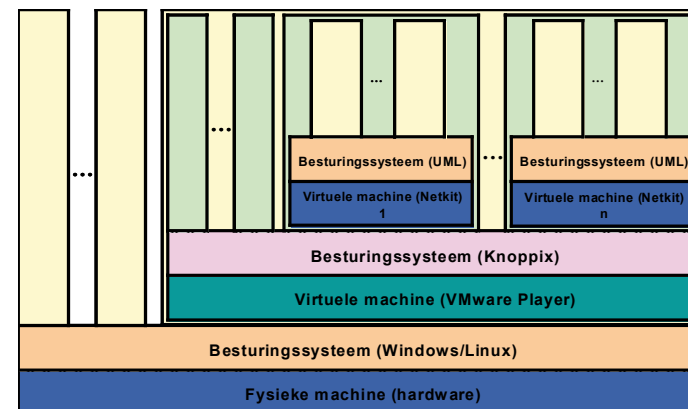
kan installeren. De virtualisatiesoftware dient verder open source of gratis zijn, en tevens betrouwbaar, gebruikersvriendelijk en toekomstbestendig (vooral door ondersteuning van nieuwe versies van Windows en Linux). Verder dient in de virtuele omgeving een natuurgetrouwe simulatie van een computernetwerk mogelijk te zijn. Ten slotte moet de virtuele omgeving volledig van de buitenwereld afgeschermd kunnen worden zodat een student daarin alle mogelijke activiteiten kan uitvoeren op gebied van security, zoals het uitvoeren van aanvallen die strafbaar of op zijn minst ongewenst zijn in een concreet computernetwerk. Het virtuele computersecuritylab bestaat uit twee virtualisatielagen, zoals conceptueel weergegeven in figuur 1. De pc van de student, met een willekeurig Windows- of Linuxbesturingssysteem, duiden we aan als de hostmachine. Een eerste virtualisatielaag creëert een virtuele omgeving op de host machine. Deze virtualisatielaag wordt gerealiseerd door een softwareapplicatie, die net als elke andere applicatie draait op de hostmachine. Deze eerste virtualisatielaag creëert een virtuele machine waarbinnen het Linux besturingssysteem draait. We duiden dit aan als de virtuele hostmachine. In principe zou in de virtuele host machine ook een ander besturingssysteem gekozen kunnen worden. Het voordeel van Linux is echter dat dit open source software is en onder studenten verspreid mag worden zonder licentiekosten. Op de virtuele hostmachine kunnen Linuxapplicaties worden uitgevoerd. We passen een tweede virtualisatielaag toe, die wordt gerealiseerd door een Linuxapplicatie die draait op de virtuele hostmachine. We kiezen hierbij voor virtualisatie op basis van User Mode Linux (UML), wat het mogelijk maakt om een Linuxkernel als een applicatieproces uit te voeren binnen Linux. Op deze wijze kunnen relatief eenvoudig meerdere virtuele machines worden gecreëerd binnen de virtuele host machine. De virtuele machines beschikken over virtuele netwerkinterfaces en kunnen in virtuele netwerken worden aangesloten.



Figuur 1:
Twee virtualisatielagen

Figuur 2 toont de architectuur van het virtuele lab in meer detail. De eerste virtualisatielaag wordt gerealiseerd door VMware-Player, een softwareapplicatie die draait op de hostmachine. VMware-Player creëert een abstractie van de hardware en het besturingssysteem van de hostmachine. In VMware-Player kunnen we een enkele virtuele machine instantiëren, de virtuele hostmachine die we voorzien van een Linuxbesturingssysteem (Knoppix). In de virtuele host machine passen we de Netkit software toe om meerdere virtuele machines te instantiëren via UML [10]. Op elk van deze virtuele machines kunnen we ten slotte een of meer applicaties uitvoeren.

De software voor het virtuele lab wordt op een dvd aan studenten geleverd. De dvd bevat de installatiesoftware voor VMware-Player op een Windows- en Linux-pc. Verder bevat de dvd een virtualmachine-configuratiefile en een ISO-bestand. Bij het opstarten van VMware-Player, selecteert de student de virtualmachineconfiguratiefile, waarmee de virtuele hostmachine wordt geconfigureerd en het ISO-bestand wordt geladen. Het ISO-bestand bevat het Linuxbesturingssysteem met daarin alle applicaties (waaronder Netkit) die de student nodig heeft voor het uitvoeren van de opdrachten.



Figuur 2:
Architectuur van het virtuele lab

Voor de installatie van het virtuele lab volstaat het dus om VMware-Player te installeren, wat met behulp van enkele muisklikken in een wizard kan gebeuren. De virtuele hostmachine maakt gebruik van een virtuele disk. Het file system van de virtuele hostmachine wordt opgeslagen op deze virtuele disk, die wordt geïmplementeerd met enkele gecomprimeerde bestanden op de fysieke disk van de hostmachine. Ook de virtuele machines hebben elk een virtuele disk, waarbij elke virtuele disk wordt geïmplementeerd met enkele gecomprimeerde bestanden op de virtuele disk van de virtuele hostmachine. Na installatie kan het virtuele lab met een enkele muisklik worden opgestart. Het virtuele lab kan naar keuze worden afgesloten op twee manieren. Bij de eerste manier wordt de toestand van de virtuele host machine niet bewaard en wordt de virtuele host machine gereset. Bij de tweede manier wordt de toestand van het virtuele lab opgeslagen. Als het virtuele lab vervolgens weer wordt opgestart, bevindt

het zich in dezelfde toestand als bij het afsluiten. Dit is handig bij het uitvoeren van opdrachten die veel tijd vergen. Als een student onvoldoende tijd heeft om een opdracht volledig te voltooien, kan hij het lab afsluiten en later de draad weer oppakken. De ervaringen van studenten met het virtuele lab zijn uitgebreid geëvalueerd. Studenten zijn doorgaans zeer positief en hebben nauwelijks problemen ondervonden met de installatie en het gebruik van de software. Een beperking is dat in het virtuele lab alleen met Linux gewerkt wordt. Verder heeft de virtuele hostmachine weliswaar een Windowsomgeving (KDE), maar de virtuele machines kennen slechts tekstgebaseerde in- en uitvoer.

OPDRACHTEN IN HET VIRTUELE LAB

Een student voert een aantal opdrachten uit in het virtuele lab. In elke opdracht instantieert en configureert de student enkele virtuele machines en verbindt deze in virtuele netwerken. Bij de complexere opdrachten

kan de student een script downloaden van de elektronische leeromgeving waarmee de instantiatie en configuratie van virtuele machines en netwerkconnecties wordt geautomatiseerd. Een eerste reeks opdrachten is gericht op het voorbereiden van een aanval, waarbij met behulp van portscanning en packetsniffing informatie verzameld wordt over een doelwit. In een tweede reeks opdrachten gaat de student een man-in-the-middle-aanval en een denial-of-service-aanval uitvoeren. In een derde reeks opdrachten gaat de student een gedemilitariseerde zone (DMZ) opzetten tussen een virtueel intranet en een virtueel internet met behulp van diverse soorten firewalls. Hierbij wordt ook een intrusiondetectionsysteem toegepast. In een vierde reeks opdrachten oefent de student met toepassing van asymmetrische cryptografie voor het uitwisselen van berichten met behulp van PGP. In de cursusboeken wordt uitleg gegeven over de opbouw en werking van het virtuele lab. De opdrachten

staan eveneens beschreven in de cursusboeken. Hierin wordt stapsgewijs aangegeven welke commando's uitgevoerd moeten worden en wat het resultaat daarvan is. Er worden vragen gesteld, die veelal betrekking hebben op het verklaren van bepaalde waarnemingen. De antwoorden op de vragen zijn in het cursusboek opgenomen. De opdrachten die de student uitvoert, worden niet expliciet beoordeeld. Voor deze opzet is met name gekozen om de hoeveelheid werk voor de docent te beperken. Omdat studenten de opdrachten zelfstandig uitvoeren zonder bijzijn van de docent, zouden tevens maatregelen nodig zijn om fraude te achterhalen. De kennis die de student geacht wordt op te doen bij de opdrachten in het virtuele lab, wordt op het tentamen getoetst.

CONCLUSIE

Ten behoeve van de cursus Security en IT aan de Open Universiteit Nederland is een virtueel computer-securitylab ontwikkeld. In dit lab kunnen studenten in diverse rollen (hacker, gebruiker, beveiligder/systeembeheerder) aan de slag. Het virtuele lab bestaat uit software waarmee twee virtualisatielagen worden gecreëerd. Met deze aanpak kan elke student eenvoudig een stand alone virtueel lab op zijn eigen pc creëren en zelfstandig opdrachten uitvoeren.

DANKWOORD

We danken Marcel Spruit en Pieter Burghouwt van de Haagse Hogeschool, die ons attendeerden op de Netkit software en een aantal praktische opdrachten ontwikkelden. We danken Jörg Keller en Ralf Naues van de FernUniversität in Hagen voor discussies rondom het virtuele lab.



Assistentes van de congresorganisatie blij om het verloop van het congres



- [1] J. Hill et al., Using an isolated network laboratory to teach advanced networks and security. SIGCSE Bull., 33(1), pp. 36-40, 2001.
- [2] K. Krishna et al., V-NetLab: a cost-effective platform to support course projects in computer security. Proc. Coll. for Information Systems Security Education, 2005.
- [3] J. Hu, D. Cordel en Ch. Meinel, Virtual machine management for Tele-Lab. IEEE Symp. on Computers and Communications, pp. 448-453, 2005.
- [4] J. Keller en R. Naues, Design of a virtual computer security lab. Proc. IASTED Int. Conf. on Communication, Network, and Information Security, pp. 211-215, 2006.
- [5] H. Lahoud en X. Tang, Information security labs in IDS/IPS for distance education. Proc. Conf. on Information Technology Education, pp. 47-52, 2006.
- [6] B. Hay en K. Nance, Evolution of the ASSERT computer security lab. Proc. Coll. for Information Systems Security Education, 2006.
- [7] M. O'Leary, A laboratory based capstone course in computer security for undergraduates. Proc. Techn. Symp. on Computer Science Education, pp. 2-6, 2006.
- [8] A. Gaspar, S. Langevin en W. Armitage, Virtualization technologies in the undergraduate IT curriculum. IEEE IT Pro, pp. 10-17, July-August 2007.
- [9] J. Smith en R. Nair, Virtual machines: versatile platforms for systems and processes (Morgan Kaufmann, 2005).
- [10] M. Pizzonia en M. Rimondini, Easy emulation of complex networks on inexpensive hardware. Proc. Int. Conf. on Testbeds and Research Infrastructures for the Development of Networks & Communities, pp. 1-10, 2008.

Software-architectuur in het afstandsonderwijs



Bastiaan Heeren - Faculteit Informatica,
Open Universiteit Nederland

Sylvia Stuurman - Faculteit Informatica,
Open Universiteit Nederland, Postbus 2960, 6401 DL
Heerlen {bhr,stm}@ou.nl

SAMENVATTING

In dit artikel bespreken we een experimenteel ontwikkeltraject waarin de mastercursus Software Architectuur van de Universiteit Utrecht geschikt is gemaakt voor het afstandsonderwijs van de Open Universiteit Nederland. De cursusopzet moest dermate flexibel zijn dat deze ook nog inzetbaar was als onderdeel van korte studieprogramma's, waarvoor een beperkt aantal groepsbijeenkomsten wordt georganiseerd. We beschrijven de overwegingen en gemaakte keuzes bij het herinrichten van de cursus, samen met onze eigen ervaringen en reacties van de studenten.

TREFWOORDEN:

Software architectuur, afstandsonderwijs

INTRODUCTIE

Software architectuur is een vakgebied binnen de Informatica dat steeds meer aandacht krijgt vanuit de industrie, de onderzoeksinstellingen en het Informatica onderwijs. Wat er precies onder software architectuur wordt verstaan en waar de grenzen van het vakgebied liggen is niet altijd duidelijk: de onderwerpen kunnen variëren van de samenhang binnen een organisatie

en de bedrijfsprocessen tot software technologische concepten zoals componentmodellen en formele architectuur beschrijvingstalen (ADL). Hoewel de Open Universiteit Nederland (OU) al vrij vroeg het belang van dit vakgebied heeft onderkend, is de cursus getiteld *Software Architectuur* pas in 2007 van start gegaan. Redenen van dit uitstel waren ondermeer de onduidelijke grenzen van het vakgebied, het ontbreken van een consensus over de inhoud van het vak, en de afwezigheid van de noodzakelijke expertise bij de OU.

De cursus is uiteindelijk tot stand gekomen in een samenwerkingsverband met de Universiteit Utrecht. In Utrecht werd het vak al geruime tijd gegeven, en was dus al de nodige ervaring opgedaan. De cursus *Software Architectuur* was één van de drie cursussen afkomstig uit deze samenwerking, maar daar waar de andere twee cursussen (over grammatica's en over het implementeren van programmeertalen) redelijk eenvoudig in te passen waren in het afstands-onderwijs van de OU was dit niet het geval bij de cursus *Software Architectuur*. De cursus in Utrecht bestond uit een reeks hoorcolleges en een omvangrijke opdracht waarvoor in vijftallen een software architectuur document opgesteld diende te worden. Iedere groep kreeg een echte (externe) opdrachtgever toegewezen om de rol van klant te vervullen. Deze onderwijsvormen zijn slecht toepasbaar binnen het afstandsonderwijs dat kenmerkend is voor de OU. Deze mismatch maakt dit specifieke geval van het hergebruiken van onderwijsmaterialen en het omvormen van de cursusopzet tot een interessante casus.

In werkelijkheid lag de situatie bij de OU nog iets ingewikkelder. Naast het geschikt maken van de cursus voor studenten in de masteropleiding van de OU was het aantrekkelijk om de cursus ook onderdeel te maken van het Certified Professional Program (CPP) getiteld Gecertificeerd Software Architect. Zo'n programma is een kort studieprogramma voor professionals bestaande uit een aantal opeenvolgende cursussen met een duidelijke lijn, welke wordt afgesloten met een diploma. Het programma in kwestie bestond uit de cursussen Analyseren en Ontwerpen met UML, Design Patterns en Component-Based Development, en werd al enige tijd aangeboden. In dit programma worden per cursus vijf begeleidingsbijeenkomsten georganiseerd: de rest blijft over voor zelfstudie. De omvang per cursus is aanzienlijk kleiner dan die in Utrecht (4,3 ECTS in plaats van 7,5 ECTS) wat nog een ander praktisch probleem opleverde.

Zelfs in het reguliere onderwijs is de didactiek van een vak over software architectuur problematisch. Het vakgebied bestaat niet uit een welomschreven theorie met harde regels, maar eerder uit een verzameling wijze raadgevingen, enkele feitjes, en gedocumenteerde best practices. Een illustratief voorbeeld van zo'n raadgeving is de Architecture Tradeoff Analysis Method (ATAM) [2], een bekende methode om een architectuur te evalueren. Deze methode schrijft negen stappen voor om een architectuur te evalueren, en suggereert zelfs een tweedaagse agenda inclusief koffiepauzes en lunches. Waarom nu juist deze stappen tot het beste resultaat leiden wordt slechts matig onderbouwd, behalve dan dat het in de praktijk goed blijkt te werken. Het inherente gevaar van onderwijs over software architectuur is dat studenten na afloop van de cursus de raadgevingen hebben bestudeerd, maar niet in staat zijn om zelf een architectuurontwerp op te stellen of te evalueren. Een deel van de kennis kan enkel worden verkregen door het zelf te hebben gedaan. In dit artikel beschrijven we de cursus Software

Architectuur van de Open Universiteit, welke na samenwerking met een reguliere universiteit geschikt is gemaakt voor het afstandsonderwijs. We bespreken de opzet van de cursus en de keuzes die zijn gemaakt. Verder blikken we terug aan de hand van onze eigen ervaringen en de evaluaties en reacties van studenten. Dit artikel is met name bedoeld voor andere universiteiten, hogescholen en commerciële opleidingen die een invulling proberen te geven aan het vakgebied van de software architectuur, of die de verschillende mogelijke werkvormen afwegen om een praktijkgericht vak zoals software architectuur te doceren.

De rest van het artikel is als volgt opgebouwd: als eerste bespreken we de nieuwe cursusopzet en hoe een reeks collegeslides is omgevormd tot een werkboek dat zelfstandig kan worden bestudeerd (sectie 2). Daarna schetsen we de aangepaste eindopdracht in sectie 3, die ondanks de beperkingen van onderwijs op afstand wel zo realistisch mogelijk moest blijven. We eindigen met een reflectie op de gekozen opzet (sectie 4) en met enkele afsluitende woorden.

NIEUWE CURSUSOPZET

De leerdoelen voor de OU cursus zijn onveranderd overgenomen. In grote lijnen moet een student in staat zijn om:

- voor een gegeven probleem verschillende oplossingen te kunnen evalueren en vergelijken, op het architectuur niveau;
- een architectuur te beschrijven en te modelleren, door middel van diagrammen en tekst;
- conflicterende belangen van stakeholders te ontdekken, en een manier vinden om die conflicten op te lossen;
- de betekenis van begrippen uit te kunnen leggen, zoals standaard architecturen, kwaliteitsmodellen, patronen voor bedrijfsapplicaties, architectuur beschrijvingstalen, modellen en de IEEE 1471 standaard.

Het materiaal voor de OU-cursus bestaat uit een tekstboek, een werkboek en een reader met artikelen. Een student wordt beoordeeld door middel van een openboek tentamen met open vragen en aan de hand van een eindopdracht. Als eerste bespreken we de totstandkoming van de reader en het tekstboek.

Reader

Bij het samenstellen van de reader stuiten we op twee problemen: studenten in Utrecht hebben de beschikking over de universiteitsbibliotheek, en een groot aantal relevante boeken waren ook binnen de faculteit in te zien. Bij de colleges werd dan ook verwezen naar een groot aantal boeken en tijdschriftartikelen als achtergrondinformatie of als illustratie bij wat er op het college werd verteld.

Voor de OU-cursus moesten we de omvang van deze leesopdrachten terugbrengen, en alle bronnen meeleveren omdat het voor studenten op afstand veel tijdrovender is om boeken uit een universiteitsbibliotheek te lenen. Als oplossing voor dat probleem hebben we gekozen voor het tekstboek van Albin [1] en een samengestelde reader met een beperkt aantal artikelen, met leesopdrachten in het werkboek. Op de cursuswebsite worden bovendien actuele links naar extra informatie bijgehouden.

Werkboek

Het werkboek is tot stand gekomen door voor elk college een leereenheid op te nemen met een inleiding, een opsomming van de leerdoelen, een aantal leesopdrachten (voor hoofdstukken uit het tekstboek of artikelen uit de reader), de kern van de leereenheid met de op het college behandelde stof, en een aantal discussievragen.

De leerdoelen vertellen een student wat van hem verwacht wordt: aan de hand van de leerdoelen kan een student bepalen of hij onderdelen van de stof moet kunnen reproduceren, of alleen begrijpen, of kunnen hanteren in een concrete situatie. De volgende leerdoelen zijn bijvoorbeeld geformuleerd voor de eerste, inleidende leereenheid.

Na het bestuderen van deze leereenheid wordt u geacht in staat te zijn om:

- uit te leggen wat de bestaansreden van een architectuur is;
- de relatie tussen stakeholders en de architectuur van een systeem te beschrijven;
- uit te leggen welke rol het sluiten van compromissen speelt bij het opstellen van een architectuur;
- het verband te beschrijven tussen architectuur en hergebruik.

De discussievragen worden bij de bijeenkomsten gebruikt om studenten te activeren en discussies uit te lokken: tijdens die discussies werd de onderbouwing van een standpunt extra benadrukt. Voor studenten op afstand helpen de discussievragen om te reflecteren op de behandelde stof. De vragen die studenten op het openboek tentamen krijgen zijn van eenzelfde aard: studenten moeten tijdens het tentamen vooral laten zien dat ze kunnen redeneren over voor- en nadelen van de verschillende keuzes waarvoor een software architect in de praktijk kan komen te staan. Naast deze discussievragen krijgen studenten ook een uitgewerkt voorbeeldtentamen om zich op het tentamen voor te bereiden.

Welkom door theatergroep Art of Events voor het avondprogramma



De keuze van de leerstof is geheel bepaald door de samenstelling van het vak zoals dat in Utrecht gegeven wordt. De leerstof bevat:

- een inleiding in het vakgebied;
- twee leereenheden over de primaire werkzaamheden van een software architect: requirements engineering, en het beschrijven en evalueren van software architecturen;
- vier hoofdstukken over patronen (gedocumenteerde bewezen oplossingen voor terugkomende problemen): architectuurpatronen, patronen voor bedrijfsapplicaties, integratiepatronen en service oriëntatie;
- twee leereenheden over onderliggende technieken: componentmodellen en design by contract.

Bij het schrijven van de leerstof is de grootste inspanning gaan zitten in het bedenken van voorbeelden ter illustratie van begrippen. Bij een hoorcollege komen voorbeelden als vanzelf ter sprake, hetzij naar aanleiding van vragen van studenten, hetzij in discussies, hetzij 'on the fly' tijdens het presenteren. Een geschreven werkboek dat alleen uitschrijft wat er op de sheets voor een college staat kan de stof niet tot leven wekken, en blijft op zo'n hoog abstractieniveau steken dat het voor studenten moeilijk wordt om te leren begrijpen wat de begrippen in de praktijk betekenen. Dat geldt in extra mate voor het vak softwarearchitectuur, omdat er voor de meeste begrippen geen scherpe definities te geven zijn: de belangrijkste functie van veel begrippen is laten zien wat de problemen zijn waar een softwarearchitect voor gesteld wordt, en wat de aanpak is om tot een oplossing te komen. Veel begrippen komen dus pas tot leven bij het werkzaam zijn als softwarearchitect.

Een voordeel van de OU hierbij is dat studenten veelal werkzaam zijn binnen de IT, en zich daardoor goed voor kunnen stellen wat de rol van een software

architect bij een complex project is. Aan de andere kant geldt dat niet voor elke student: cursussen moeten zo zijn ingericht dat ook studenten die niet zo'n werkomgeving hebben de studie kunnen volgen.

Extra lastig bij het schrijven van een werkboek aan de hand van colleges is dat studenten op afstand veel minder snel een vraag zullen stellen wanneer iets ze niet duidelijk is. Bij het uitschrijven van een werkboek moet de auteur er dus alert op zijn dat lastige stof liefst op verschillende manieren wordt uitgelegd, en dat de voorbeelden die ter illustratie worden gebruikt geen onduidelijkheid scheppen.

De punten van aandacht bij het geschikt maken van een bestaand vak voor afstandsonderwijs bestaan er dus uit dat expliciet gemaakt moet worden wat er van studenten verwacht wordt, dat de leerstof concreet gemaakt moet worden met voorbeelden die aansluiten bij de voorkennis die we veronderstellen, en dat er een evenwicht gezocht moet worden tussen extra heldere uitleg (omdat de student over het algemeen geen vragen zal stellen) en het zelf leren redeneren van de student (te stimuleren door middel van discussievragen waar geen eenduidig antwoord op valt te geven).

EINDOPDRACHT

Aan de eindopdracht is noodzakelijkerwijs een hele nieuwe invulling gegeven. De eindopdracht bestaat, net als in Utrecht, uit het opstellen van een software architectuur document van een realistisch systeem. Dit maakt de cursus praktisch, en het helpt studenten om de behandelde theorie beter te begrijpen door deze zelf toe te passen. De opzet uit Utrecht was om verschillende redenen niet direct bruikbaar:

- De opdracht werd in groepen van 4 à 5 personen uitgewerkt. Zo'n groepsopdracht vereist onderlinge coördinatie en overleg. Dit is problematisch aangezien OU studenten vaak een verschillend

studietempo hebben, en vanwege banen en overige werkzaamheden voor andere studiemomenten kiezen. Er zijn bovendien niet voldoende masterstudenten actief om meerdere startmomenten per jaar voor een dergelijke groepsopdracht te faciliteren.

- De omvang van de opdracht was te groot, deels omdat het op vijftallen was afgestemd, deels omdat er meer studiepunten aan de cursus verbonden zijn (7,5 ECTS ten opzicht van 4,3 ECTS bij de OU). In de nieuwe opzet was er plaats voor een opdracht van ongeveer 40 uur per student.
- Alle groepen in Utrecht kregen een externe opdrachtgever toegewezen om de rol van klant te vervullen. Onderdelen zoals het achterhalen van de requirements door middel van een interview en het presenteren van een ontwerp aan de stakeholders kregen door deze aanpak een bijzonder realistisch karakter. In het afstandsonderwijs is dit niet te realiseren door het vele afstemmen dat deze werkvorm vereist.

Vanwege de bovenstaande argumenten is er besloten om de opdracht individueel te laten maken, en deze volledig te herschrijven. De moeilijkheid bij het herformuleren van de opdracht was om de opdracht toch zo realistisch mogelijk te houden, met echte problemen uit de praktijk, maar met minder middelen en in minder tijd. Bij voorkeur zouden bestaande documenten gebruikt moeten worden om de 'echtheid' van de opdracht te garanderen.

De keuze is uiteindelijk gevallen op het opstellen van een software architectuur document voor de Bachelor Master Schakelsite (BaMaS) [5] van de Digitale Universiteit. Deze website geeft informatie over de schakelmogelijkheden tussen de verschillende bachelor- en masteropleidingen in Nederland, aan bachelorstudenten, maar ook aan middelbare scholieren en voorlichters. De website is vrij toegankelijk en te vinden op www.bamas.nl, wat het

systeem zeer concreet maakt (maar zie ook de kritische kanttekening in de volgende sectie). Omdat de OU betrokken is geweest bij het ontwerpen en realiseren van het BaMaS systeem waren er meerdere ontwerpdocumenten voorhanden. De documenten over BaMaS zijn ook opgenomen in het TISO project [10] dat dossiers van authentieke automatiseringsprojecten verzamelt en beschikbaar stelt.

De documenten over BaMaS beschrijven diverse kwaliteitseisen en wensen van de verschillende betrokken partijen, zonder daarbij oplossingen in het ontwerp aan te dragen. Zo zijn de documenten bijvoorbeeld erg expliciet over de gewenste schaalbaarheid (het aantal gebruikers dat de website gelijktijdig moet kunnen bezoeken), de uitbreidbaarheid (tot Europees niveau), de portabiliteit (met betrekking tot de web-browsers) en de beveiliging voor het auteursgedeelte om de content van de schakelmogelijkheden in te voeren. De meeste studenten zijn bij een ontwerp-cursus over UML en patronen al in aanraking gekomen met het BaMaS systeem, wat een bijkomend voordeel is.

Aan het te schrijven document worden wel eisen opgelegd. Zo moet het worden opgesteld volgens de geldende normen, wat in de praktijk neerkomt op het volgen van de IEEE 1471-2000 standaard [9] waarin enkele algemene richtlijnen te vinden zijn, en het 4+1 model van Kruchten [6]. Dit model geeft vier gezichtspunten van waaruit een architectuurontwerp gegeven moet worden, waarbij de +1 staat voor een verzameling usecases om de vier views aan elkaar te relateren. Het is toegestaan om in plaats van Kruchten een alternatieve verzameling van gezichtspunten te gebruiken voor de opdracht, zoals die van het Software Engineering Institute (SEI) [3] of van Rozanski en Woods [7]. Hoewel sommige cursisten moeite hebben met deze extra vrijheid, maakt het hen wel meer bewust van de voor- en nadelen.



Studenten van het ROC Midden Nederland verzorgden de beveiliging

Er heerst geen wereldwijde consensus over de precieze indeling van een software architectuur document. De formulering van de opdracht laat de indeling dan ook vrij. Wel zijn er verschillende templates in omloop die mogen worden gebruikt, maar zelfs met zo'n template is de opdracht verre van een invuloefening. De geboden vrijheid stelt onze cursisten in staat om een template te kiezen dat aansluit bij de werksituatie, bijvoorbeeld RUP op Maat [4] of een template dat door het bedrijf zelf is ontwikkeld. Afgezien van de keuze van het template wordt er wel verlangd dat de stakeholders en de requirements zo helder mogelijk worden verwoord, dat er een kwaliteitsmodel wordt gebruikt om de requirements te classificeren (zoals Quint2 [8]), en dat alle ontwerpkeuzes en beslissingen worden beargumenteerd.

Tijdens de eerste run van de cursus in een kort studieprogramma is besloten om alle studenten de mogelijkheid te geven een conceptversie op te sturen en om deze te voorzien van commentaar. Het bleek erg nuttig om tussentijds al suggesties aan te kunnen dragen, ze te wijzen op onvolkomenheden en advies te geven bij onduidelijkheden. Dit verving tot op zekere hoogte de onderlinge discussies die er bij een teamopdracht wel zijn. De conceptversie had als neveneffect dat het stimuleerde om vroegtijdig aan de opdracht te beginnen: bij deeltijdopleidingen is het constant houden van een studietempo nog al eens een probleem.

ERVARINGEN EN EVALUATIES

De cursus Software Architectuur is in het najaar van 2007 gestart. Inmiddels is de cursus aan vijf groepen gegeven (met de extra begeleidingsbijeenkomsten), met in totaal zo'n 65 deelnemers. Daarnaast hebben verschillende studenten van de masteropleiding de cursus afgerond zonder deze bijeenkomsten. De cursus is bijzonder goed ontvangen, wellicht door de ervaring die al was opgedaan in Utrecht. Aan de ontwikkelde eindopdracht is nog wel geschaafd, om

onduidelijkheden weg te halen, en omdat slechts weinigen deze binnen de gestelde 40 uur af kregen. Toch werd juist dit onderdeel als leerzaam ervaren door de studenten.

Toegevoegde waarde bijeenkomsten

Bij het aanpassen van de cursus is rekening gehouden dat deze op afstand zou worden aangeboden. Aangezien er ook een variant is inclusief begeleidingsbijeenkomsten is de vraag wat de toegevoegde waarde is van deze bijeenkomsten ten opzichte van de zelfstudie variant. Vooral de discussies en de onderlinge interactie hadden een positieve bijdrage, en deze kregen na verloop van tijd een steeds prominentere rol tijdens de bijeenkomsten.

Per onderwerp zijn er meerdere discussievragen opgesteld die opzettelijk open zijn geformuleerd. Voorbeelden van vragen die tot een discussie leidden zijn:

- Do all stakeholders concerns carry equal weight? If not, what criteria exist for arbitrating among them?
- What is the role of software architecture with respect to agile methods such as Extreme Programming?
- What tactics would you advocate to promote the security sub-characteristic?

Cursisten met een andere achtergrond hebben dikwijls hele andere opvattingen over software architectuur, en kijken anders tegen dit soort vraagstukken aan. Dit maakt dat er inhoudelijk echt gediscussieerd kan worden over deze vragen, wat de bijeenkomsten een grote meerwaarde geeft. Studenten in de zelfstudie variant lopen dit helaas mis.

Ervaringen met eindopdracht

Uit de evaluatie (en persoonlijke communicatie) is gebleken dat de deelnemers de eindopdracht als leerzaam ervaren, maar toch ook als omvangrijk. Vooral het beginnen met het schrijven van het document blijkt bij de meeste deelnemers een obstakel te zijn. Een verklaring hiervoor is wellicht dat de gangbare theorie en de definities niet voldoende concreet zijn om ze direct in de praktijk toe te kunnen passen. De verschillende templates voor een SAD helpen wel het schrijfproces te structureren, maar verhelpen de moeizame start niet.

Een aardige contradictie in de opdracht over BaMaS is dat er een ontwerp moet worden gegeven voor een systeem dat al bestaat. In de praktijk blijkt dit niet zo'n probleem te zijn. De functionaliteit van BaMaS is goed te overzien, wat natuurlijk wenselijk is om de opdracht niet te uitgebreid te laten zijn. De opmerking dat het systeem een rechttoe rechtaan informatiesysteem is en dus eigenlijk te simpel is voor het opstellen van een volledig software architectuur document is zo nu en dan door deelnemers geplaagd. Toch zijn er uit de opgestuurde uitwerkingen hele verschillende ontwerpen naar voren gekomen, en zijn er meerdere composities van het systeem denkbaar.

Academisch niveau

De cursus is afgestemd op het masteronderwijs. Enige ervaring met het ontwikkelen van software en alles wat daar bij komt kijken is absoluut noodzakelijk om de rol van een software architect te kunnen begrijpen. De gemiddelde student bij de OU heeft al de nodige praktijkervaring opgedaan om de noodzaak van software architectuur in te zien.

De open formulering van de opdracht en de geboden vrijheid om het probleem aan te pakken is kenmerkend voor de masterfase. De opdracht maakt een duidelijke splitsing mogelijk tussen het inventariseren van de

stakeholders, de requirements en de conflicten tussen de requirements enerzijds (de probleemanalyse), en een beargumenteerde technische uitwerking anderzijds (de oplossing). Vooral het beargumenteren van de gemaakte ontwerpkeuzes blijkt een vaardigheid te zijn die doorgaans niet aanwezig is bij onze studenten.

Technische onderwerpen

In de gekozen cursusopzet is een duidelijke focus aangebracht op het software architectuur document. Een alternatief is om ook enkele technische onderwerpen te behandelen in de cursus. Een goede softwarearchitect hoort tenslotte voldoende geïnformeerd te zijn om te kunnen oordelen over de haalbaarheid van een bepaalde technische oplossing, en de consequenties van de invoering.

In de cursus komen wel enkele technische onderwerpen aan bod, maar deze spelen een minder belangrijke rol. Zo worden er enkele architectuur beschrijvingstalen genoemd, waarvan sommige een stevige theoretische onderbouwing hebben (zoals de theorie van de Communicating Sequential Processes). Dit wordt vaak als 'te academisch' ervaren. Andere voorbeelden zijn de invarianten, pre- en postcondities in het kader van de design by contract benadering, en de co-variantie en contravariantie die wordt besproken bij type systemen.

De gemiddelde student bij de OU heeft een minder formele achtergrond dan de student bij enkele technisch georiënteerde masteropleidingen waar software architectuur ook wordt gegeven. Hoewel hier rekening mee is gehouden bij het aanpassen van de cursus vinden we het wel belangrijk om ze hier iets van mee te geven.

CONCLUSIE

Software Architectuur is een lastig vak om te doceren omdat het vooral praktische vaardigheden zijn die moeten worden geoefend. Het trainen van deze vaardigheden met de beperkingen van onderwijs op afstand is een extra uitdaging. In dit artikel hebben we de cursus besproken zoals die aan de Open Universiteit wordt gegeven. Deze cursus is geschikt voor zelfstudie en is door de studenten zeer positief ontvangen.

In de groepen van de korte studieprogramma's zaten meerdere studenten die in het dagelijks leven werkzaam zijn als software architect, of een vergelijkbare functie vervullen. Ook voor deze groep bleek de cursus een waardevolle aanvulling te zijn. Door de zeer levendige discussies die tijdens de bijeenkomsten plaatsvonden is de cursus steeds verder verfijnd.

De cursus is tot stand gekomen in samenwerking met de Universiteit Utrecht, na een experimenteel ontwikkeltraject om cursusmaterialen aan te passen aan de onderwijssituatie van de OU. De aanpassingen hebben toch nog relatief veel tijd gekost: de materialen konden niet zondermeer worden overgenomen. Wel heeft de samenwerking en het hergebruik een betere cursus opgeleverd, wat het traject zeer zeker de moeite waard heeft gemaakt. Nog steeds vindt er regelmatig overleg plaats tussen de twee universiteiten over software architectuur.



Onze dank gaat uit naar iedereen die betrokken is geweest bij het realiseren van de cursus. In het bijzonder noemen we Johan Jeuring, Ella Roubtsova, Marleen Sint, en Lex Bijlsma die in de tijd van de samenwerking het vak Software Architectuur doceerde in Utrecht. Gert Florijn is de oorspronkelijke docent van het vak uit Utrecht en heeft dit mede vorm gegeven.

Over de auteurs

Bastiaan Heeren is als uitvoerend docent betrokken bij de cursus *Software Architectuur* zoals die wordt onderwezen binnen de korte studieprogramma's. **Sylvia Stuurman** is het aanspreekpunt voor studenten die de cursus op afstand volgen, en is tevens betrokken geweest bij het schrijven van het werkboek en het opstellen van de eindopdracht.

[1]

Stephen T. Albin. The Art of Software Architecture: Design Methods and Techniques. Indianapolis: John Wiley & Sons, 2003.

[2]

P. Clements, R. Kazman, and M. Klein. Evaluating Software Architectures: Methods and Case Studies. Addison-Wesley, 2002.

[3]

Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Robert Nord, and Judith Stafford. Documenting Software Architectures: Views and Beyond. Addison-Wesley, 2003.

[4]

Remi-Armand Collaris and Eef Dekker. RUP op Maat. Sdu uitgevers, 2008.

[5]

Humphrey Ferdinandus, Joop Kaldewey, Wolter Kaper, Aad Sloodmaker, and Evert van de Vrie. Schakelen tussen bachelor- en masteropleidingen. Tinfon, (4):119-121, Oktober 2004.

[6]

Philippe Kruchten. Architectural blueprints: the 4+1 view model of architecture. IEEE Software, 12(6):42-50, 1995.

[7]

Nick Rozanski and Eóin Woods. Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives. Addison-Wesley Professional, April 2005.

[8]

Software Engineering Research Center (SERC). Quint2, the Extended ISO Model of Software Quality. <http://www.serc.nl/quint-book/>.

[9]

Software Engineering Standards Committee of the IEEE Computer Society. IEEE Recommended Practice for Architectural Descriptions of Software-Intensive Systems, Std 1471-2000, 2000.

[10]

TISO: onderwijs en onderzoek met authentieke dossiers van automatiseringsprojecten, <http://www.tisoweb.nl/>.

9

How Industry Participates in Enterprise Resource Planning Education

Pascal Ravesteyn - HU University of Applied Sciences Utrecht

Adri Köhler - HU University of Applied Sciences Utrecht

Nijenoord 1, 3552 AS, Utrecht, NL
pascal.ravesteijn@hu.nl, adri.kohler@hu.nl

ABSTRACT

During the last two decades many businesses around the world have adopted Enterprise Resource Planning (ERP) systems. Consequently the growth in the number of ERP implementations has had an enormous impact on the demand for ERP skills. This is one of the main reasons that many universities have adopted ERP in their curriculum. This paper describes a joint effort by industry and the HU at designing an ERP curriculum that bridges the gap between skills taught by academia and those valued by industry.

INTRODUCTION

During the last two decades, many businesses around the world have adopted Enterprise Resource Planning (ERP) systems. ERP systems are enterprise-wide information systems that replace separate systems for such tasks as sales, purchasing, finance, logistics, manufacturing and warehousing. 'They are designed to address the problem of information fragmentation or 'islands of information' in business organizations (Muscatello & Chen, 2008).' In essence, ERP developers claimed to be able to automate all of a company's primary process and also improve them by incorpora-

ting best business practices. Consequently the growth in the number of ERP implementations has had an enormous impact on the demand for ERP skills. This is one of the main reasons that many universities have adopted ERP in their curriculum. Many universities, however, have struggled with the complexity of ERP software and the way in which to incorporate them into the curricula.

ERP EDUCATION IS COMPLEX

Although the use of ERP in education might be coming more common, the problems related to it are far from being solved. Several authors have noted the considerable cost and effort involved with the use of ERP (Becerra-Fernandez, Murphy, & Simon, 2000; Nelson & Millet, 2001; Davis, 2004; Hawking et al., 2005). The software can be acquired at reasonable cost or sometimes even no cost (Noguera & Watson, 1999) but the time and resources needed to maintain the software, acquire the necessary skills to work with it, and to develop a new curriculum is tremendous. It is not only teachers who have to invest a lot of time, but this is also the case for the staff (e.g. IT shared service centers) that is involved. Therefore, these efforts are often dependent on relatively few staff; and once they leave a university or change direction, the curriculum usually flounders (Hawking et al., 2005). Rosemann, Scott and Watson (2000) propose collaboration between universities in teaching ERP courses to overcome this problem; this, however, causes extra problems such as curricula and rosters that need to be synchronized.

Whereas introducing ERP in an educational environment means overcoming barriers, such as the cost of implementing and maintaining an ERP application and a lack of ERP related skills, the most difficult task is often to develop proper course material. Sometimes the ERP software developer provides its own training material, but transforming this to suitable material for educational purposes is not an easy process. Commonly, the provided materials require a pre-configured data set, which might not be made available; and also, commercial training exercises are often just snapshots to reinforce particular features of the system instead of comprehensive exercises illustrating the end-to-end processes that can be supported and that are relevant in an educational environment (Hawking et al., 2005).

Besides the mentioned hurdles that fast moving developments in IT have made, that industry no longer requires employees with just ERP related skills. Today a broader range of skills that support the development, implementation and maintenance of e-business solutions is required due to the continuing evolution of ERP. One of the major trends in ERP is the focus on front-office applications and inter-organizational business processes (McGaughey & Gunasekaran, 2008). ERP skills are just a subset of this. These developments are part of the so called second wave of ERP education (Hawking et al. 2005).

In this paper, a joined effort of industry and the HU University of Applied Science at the development and implementation of an Enterprise Resource Planning minor is described that is aimed at overcoming the mentioned obstacles (such as costs, lack of skills etc.), while at the same time preparing students for the new industry needs. In the following section, the choices made in the design of the curriculum are discussed. Section 3 then describes the process of carrying out the minor for the first time; and in Section 4, the out-

comes and lessons learned are highlighted. The final section gives a discussion on the future of the minor.

CURRICULUM DESIGN

As a start to the project, a study was conducted to find best-practices of other universities about the development of an ERP curriculum to see if the mentioned problems might have already been solved. This still didn't seem to be the case. The study did give us a useful overview of different approaches that can be taken when developing an ERP curriculum. The following four approaches are described by (Hawking & McCarthy, 2000; Hawking et al., 2005; Jensen, Fink, Moller, Rikhardsson, & Kraemmergaard, 2005):

- 1 ERP training
- 2 ERP via business processes
- 3 information systems approach
- 4 ERP-concepts

The first approach is basically instruction or training in a particular ERP system. This is very similar to the training courses that the ERP developers and suppliers provide to their customers and could be done by reusing commercial training material. The second approach focuses on business processes and related concepts (e.g. financial administration or production scheduling and planning) and uses ERP to assist in the presentation and clarification of these methods and concepts. For this approach, commercial training material is not sufficient; and new material will have to be developed. The third approach uses ERP to illustrate information systems concepts. It is very similar to the second approach; only the target group or goal differs. Instead of teaching business students and concepts, the target group will most likely be



St. Janskerk met gedekte tafels voor het congresdiner

computer science / information systems students and the concepts that are taught are different. Finally, the last approach is to teach about ERP related skills, such as selection and implementation of ERP software. It is not really necessary to use a real live ERP system for this although it could give a clearer picture of the complexity of such systems. Of course, it is also possible to combine aspects of all approaches to create a more hybrid approach.

Before choosing a development approach, it was decided that the new ERP curriculum would be given in the form of a minor, which is a coherent program of 30 ECTS. Because we wanted a broad minor that attracts students with many different backgrounds, this had to be taken into account when developing the curriculum. For this reason, it was decided to use all of the four approaches mentioned above. Furthermore, to really enhance the learning experience of students, we used a research initiative that was sponsored by the National Science Foundation in the United States to identify the skills required in relation to large information systems such as ERP. In their recommendations, written down in the 'Information Systems-Centric Curriculum Document' (1999), one of the important outcomes was the advice to have ongoing collaboration with industry that also involves students. Additionally, experts suggest three different groups of skills that an industry ready IT graduate should have. The first group is personal skills, such as systematic-thinking, problem-solving, critical-thinking, risk-taking, personal discipline, persistence, and curiosity. The second group consists of interpersonal skills, like communication (oral, written, listening, and teamwork), collaborative skills and conflict resolution skills. The final group consists of technical knowledge skills, which consist of knowledge about enterprise computing architectures and delivery systems, information abstraction, representation and organization, concepts of information and systems distribution,

human behavior and computer interaction, process management and systems development, among other topics. For the minor we developed, it was decided that the personal and interpersonal skills should receive a lot of attention. This decision was based on interviews with representatives from industry that time and again noticed the lack of these skills with our students. Besides due to the different backgrounds of the students that can enroll in the minor, a heavy emphasis on technical knowledge skills is not possible. Because an important part of the minor should involve real life projects for the students and because we hoped collaboration with industry could help overcome most of the identified problems in earlier initiatives, we set out to find industry partners that were willing to participate in this project. The first and foremost important partner would be a supplier of an ERP package. We decided to partner with Microsoft and use their MS Dynamics AX solution. The most important factor that influenced us in making this choice was not that Microsoft provided a better offer regarding their ERP solution (this was basically the same, so free of charge) or that the software is better. Microsoft said they would positively recommend us within their partner network, and we wanted that network to actively contribute to the minor. Because minors are always followed in the final stages of a student's education, the added value to partner companies is high.

PARTNERSHIP

Even with Microsoft's recommendation, it still took several information sessions and visits to interested partners before we had enough participants. In the end, actual partnership contracts were signed with Microsoft for the free delivery of the MS Dynamics Axapta solution. This partnership also gave us free participation to official certification program, Mprise to deliver an ERP training, five companies (SBA, HGH, Accenture, Centric and Avanade), which could provide

ENTERPRISE RESOURCE PLANNING MINOR - INTRODUCTION WEEK				
Period 1 (9 weeks)	Introduction Enterprise Resource Planning	Vision, Strategy and Change	Project Management (by Expecto)	Functional ERP Training (by Mprise)
Period 2 (10 weeks)	Industry Project - introductionperiod	Business Process Management	Customer Relationship Management	
	Industry Project (at either Accenture, Centric, SBA, HGH or Avanade)			

Table 1:

Overview of the ERP Minor Curriculum

guest lectures, sponsorship, and onsite projects for students. Expecto provided training and certification in Prince 2 project management foundations. As stated above, we used a combination of the different approaches available in developing the ERP minor curriculum. In Table 1, an overview of the curriculum is shown. The course started with a week of introduction. During that first week, students got to know each other and the lecturers; they also get to know about the companies via guest lectures and company visits. Because students were expected to come from a large variation of different bachelor studies and a large variance in knowledge and skills was expected, a basic introduction in writing papers and reports (including the topic of plagiarism) was given. On the first day, students have to make teams in which they would be working on an initial assignment during the remainder of the week. Presentations had to be given on Friday, and papers had to be handed in on Monday morning.

First period

After the introduction week, the first period mainly consisted of modules that taught business concepts that can be automated by ERP, as well as ERP concepts. The module, Introduction ERP, covered a wide variety of topics, that ranged from 'What is ERP?' and 'How to select and implement ERP?' to explaining

the business activities that can be implemented, such as material requirements planning, forecasting, distribution, and sales. The module Vision, Strategy and Change covered the broader perspective of how organizations function and what should be taken into account when management decides that the organization should start an ERP project. Because one of the major critical success factors when implementing ERP is a lack of project management skills (Holland & Light, 1999; Muscatello & Chen, 2008), a module on project management was included in the minor. This module consisted of the Prince2 foundation course and was given by consultants from Expecto. Students had the choice of taking the official certification exam as an assessment and receiving the certificate when they pass or taking an exam supplied by the university if they don't want to receive certification. Besides emphasizing the importance of project management by the way in which this module was taught, it also heightened the market value for students if they obtained certification.

Halfway during the first period, a module on functional ERP training started. Again, this module was entirely organized by one of the program partners (in this case Mprise, a Microsoft Gold certified partner in learning solutions). By working together with Mprise, the



The Blue Grass Boogiemer zorgden voor de muzikale omlijsting van het diner

university doesn't have to train teachers who wish to maintain their ERP skills. This module is also commercially available; and again, it was possible for students to receive an official Microsoft certificate. In industry, the most wanted skills are those in trade and logistics; therefore, the module consisted of a basic MS Dynamics AX training. In total, the module consisted of 12 consecutive training days that started at 9 a.m. and ended at 16 p.m. Students received vouchers from Microsoft to take an official certification examination at one of the locations provided by Microsoft. Students were told to take the exam within four weeks after the module was finished.

Second period

The second period consisted of two new theoretical modules and the introduction of the industry period. The customer relationship management module addressed the change in focus of ERP towards the front-office of organizations (as stated by McGaughey & Gunasekaran, 2008). The second theoretical module, business process management, taught students why it is essential to analyze and, if needed redesign, business processes as part of an ERP implementation. Students also learned the basics of process modeling with the Business Process Modeling Notation (BPMN). Because the skills of this course are a prerequisite for the industry project, it was given within 4 weeks by scheduling multiple sessions per week. Finally, the second period started with an introduction to the companies where students would work on real projects. During this introduction period of four weeks, students were present at the company for two days a week; they received an orientation to the company,

were assigned to a company supervisor and given time to discuss the project they would be undertaking. Also a supervisor from within the university was assigned to oversee the project. At the end of the period, a project initiation document (PID), according to the Prince2 standard, was to be handed in to the university supervisor attesting to agreement between students and company supervisor as to their tasks. When this was done, the project started; and students were expected to work at the companies for four days a week. On Fridays, students come back for the CRM course at the university; and if necessary, they could meet with their supervisor to discuss their projects.

The minor ended with a seminar, during which students, teachers and supervisors from both the university and partner companies were present. Students presented their projects and results and discussed these with the audience. Also, official certificates were given to students who had passed the exams. Students who had not chosen to participate in earning certificates took separate university examinations.

OUTCOMES AND LESSONS LEARNED

The Enterprise Resource Planning minor started in September 2008 and finished on January 31, 2009. A total of eleven students participated in the entire minor, and we also had several students that followed a few modules as a replacement for subjects from their major. Students had their backgrounds in five different bachelor courses (commercial economy, logistics and economy, business informatics, industrial management, and software engineering) from three

different faculties. Because students were following different majors, their entry level knowledge on ERP also differed from a low level of knowledge, the student from commercial economy, to an intermediate level of knowledge, students from business informatics.

The industry-based projects provided an active learning environment that enabled the students to bring into practice what they had learned during the first period. Also, the projects gave students a tremendous opportunity to learn a lot from practitioners with several years of experience. As it turned out, the

industry partnership bridged the gap between the skills of university graduates and those required by industry. Also, the projects enabled the university to build closer links with ERP related industry.

Students evaluated the minor very positively. Besides acquiring a lot of knowledge on ERP in theoretical courses, students were highly appreciative of the projects and the possibility to get certificates that are highly valued in industry. Students really found that this minor was able to bridge the gap between the skills taught at a university and those required by industry.

Henk Plessius, voorzitter van de NIOC-stuurgroep, bedankt Sacha Koch voor haar assistentie bij de organisatie van het congres



One of the prime reasons to involve industry in this minor was to prevent the problems existed during earlier initiatives at our and other universities as described in the introduction. So did it work? For the most part, it did indeed. By cooperating with industry, we no longer need to train our staff in ERP; and we were able to teach students competencies that are really valued by industry by using ERP to clarify business and information science concepts. As described at the beginning of the paper, one of the major problems is costs; and while we were able to lower them, they are not entirely gone. Instead of maintaining an ERP knowledge base within the university, we now hire it from industry. For this minor, the partners contributed exactly the same amount of money needed to hire outside trainers. This means that each year these costs and sponsorships have to be budgeted; and if there is a gap, the minor runs the risk of being aborted.

CONCLUSIONS AND FUTURE OUTLOOK

In conclusion, we can state that the way in which the minor curriculum has been developed helped prevent the major problems are involved with using ERP in an educational context and at the same time helps students attain skills that are highly valued by industry.

This does come with a major organizational effort, however. Each year, partners from the Microsoft network will have to be contacted and asked to participate in the minor. However, we expect to solve this problem by negotiating long term partnership contracts. Also, all partners have to offer projects where students can participate; and to maintain the educational quality, these projects have to be checked by the minor manager.



[1]

Becerra-Fernandez, I.M., Murphy, K., & Simon, S. (2000). Integrating ERP in the Business School Curriculum. *Communications of the ACM*, 43(4).

[2]

Davis, C., & Comeau, J. (2004). Enterprise Integration in Business Education: Design and Outcomes of a Capstone ERP-based Undergraduate e-business Management Course. *Journal of Information Systems Education*.

[3]

Hawking, P., & McCarthy, B. (2000). Industry Collaboration: A Practical Approach for ERP Education. *Australasian Conference on Computer science education 2000*. 8, 129-133.

[4]

Hawking, P., McCarthy, B., & Stein, A. (2005). Integrating ERP's Second Wave into Higher Education Curriculum. *PACIS 2005 Proceedings*. Paper 83. <http://aisel.aisnet.org/pacis2005/83>.

[5]

Holland, C.P., & Light, B. (1999). A Critical Success Factors Model for ERP Implementation. *IEEE Software* May/June 1999.

[6]

Jensen, T., Fink, J., Moller, C., Rikhardsson, P., & Kraemmergaard, P. (2005). Issues in ERP Education Development - Evaluation of the Options Using Three Different Models. *ICESAcc 2005 Proceedings*.

[7]

McGaughey, R.E., & Gunasekaran, A. (2008). Enterprise Resource Planning (ERP): Past, Present and Future. *International Journal of Enterprise Information Systems*. 4(1), 23-35.

[8]

Muscatello, J.R., & Chen, I.J. (2008). Enterprise Resource Planning (ERP) Implementations: Theory and Practice. *International Journal of Enterprise Information Systems*. 4(1), 63-78.

[9]

Nelson, R., & Millet, I. (2001). A Foundation Course in ERP and Business Processes: Rationale, Design, and Educational Outcomes. *AMCIS 2001 Proceedings*. Paper 193. <http://aisel.aisnet.org/amcis2001/193>.

[10]

Noguera, J., & Watson, E. (1999). Effectiveness of Using Enterprise Systems to Teach Process-Centered Concepts in Business Education. *AMCIS 1999 Proceedings*. Paper 279. <http://aisel.aisnet.org/amcis1999/279>.

[10]

Rosemann, M., Scott, J., & Watson, E. (2000). Collaborative ERP Education: Experiences from a First Pilot. *AMCIS 2000 Proceedings*. Paper 127. <http://aisel.aisnet.org/amcis2000/127>.

10

Platform Beeldverwerking HBO



Jaap van de Loosdrecht - Noordelijke Hogeschool Leeuwarden/Hogeschool Kenniscentrum Computer Vision j.van.de.loosdrecht@tech.nhl.nl

SAMENVATTING

Beeldverwerking is het automatiseren van visuele inspecties met behulp van camera's en computers. Het Platform Beeldverwerking HBO heeft als doel een brugfunctie te vervullen bij het vergaren, uitwisselen en verankeren van de kennis en expertise op het gebied van beeldverwerking. Het platform stimuleert onderwijs en contractactiviteiten door hbo-instellingen. Eén van de resultaten is dat er cursusmateriaal op hbo-niveau beschikbaar is en dat 25 docenten van 9 hogescholen door de Noordelijke Hogeschool Nederland (NHL) zijn opgeleid om het vak te kunnen geven.

TREFWOORDEN

Computer Vision, Beeldverwerking, Contractactiviteiten, Cursusmateriaal, hbo.

INLEIDING

Beeldverwerking of Computer Vision is het automatiseren van visuele inspecties. Met behulp van een computer worden beelden geïnterpreteerd die met een camera zijn vastgelegd. De, op deze wijze verkregen, informatie kan vervolgens worden gebruikt om andere processen aan te sturen.

Voorbeelden daarvan zijn:

- Kwaliteitscontrole: het detecteren van MRSA-bacteriën;
- Positie- en oriëntatiebepaling: autonoom voortbewegende robots;
- Sorteren van producten op lopende banden: doosjes met medicijnen sorteren.

Voor veel bedrijven die producten maken of verwerken is visuele inspectie of meting belangrijk. Met behulp van Computer Vision is het in een groot aantal gevallen mogelijk om deze inspecties of metingen geautomatiseerd te laten uitvoeren. Dit draagt in veel gevallen bij aan een goedkoper, flexibeler en/of arbeidsvriendelijker productieproces.

NHL KENNISCENTRUM COMPUTER VISION

Het Kenniscentrum Computer Vision (KCV) van de NHL is in 1994 gestart en is inmiddels uitgegroeid tot een goed uitgerust laboratorium met een landelijke bekendheid.

De missie van het KCV is om:

- hét kenniscentrum te worden op het gebied van Computer Vision voor het bedrijfsleven in Noord-Nederland;
- hét kenniscentrum te blijven op het gebied van Computer Vision voor het hbo in Nederland.

De staf van het KCV bestaat uit een coördinator, een docent, een junior onderzoeker en twee projectingenieurs. Ruim 200 studenten hebben hun stage- of afstudeeropdracht bij het KCV Lab uitgevoerd. Sinds 1996 zijn er meer dan 100 projecten voor het bedrijfsleven gestart en succesvol afgerond.

Een typische projectomvang ligt tussen de € 2.500 en € 40.000. Er is in totaal voor meer dan € 2.000.000 aan projecten uitgevoerd. Opdrachtgevers variëren van Friese eenmanszaken tot multinationals afkomstig uit geheel Nederland. Ongeveer de helft van de opdrachten zijn vervolgoopdrachten.

Vanaf 1997 wordt het vak Computer Vision gegeven aan studenten van de NHL. Dit vak zit in het curriculum van de opleidingen Informatica, Technische Informatica, Elektrotechniek en Werktuigbouwkunde en Forensic Science. De lesstof is in de loop der jaren geïnspireerd op ervaringen die zijn opgedaan bij de in opdracht van het bedrijfsleven uitgevoerde projecten. De lesstof wordt jaarlijks geactualiseerd en uitgebreid. De NHL heeft speciaal voor dit vak een practicumzaal ingericht met daarin tien werkplekken met pc's, camera's en belichtingsapparatuur. Het vak Computer Vision is in 2009 voor de twaalfde keer gegeven als cursus van een week voor mensen uit het bedrijfsleven. Daarnaast heeft het KCV een workshop Computer Vision voor middelbare scholen en roc's ontwikkeld.

PLATFORM BEELDVERWERKING HBO

Op verzoek van Senter Novem IOP Beeldverwerking (Economische Zaken) heeft het NHL KCV een plan opgesteld om de kennis op het gebied van beeldverwerking te verankeren bij het hbo en mkb in Nederland. Dit heeft in 2005 geleid tot de oprichting van het Platform Beeldverwerking HBO. In dit platform vervult de NHL een voortrekkersrol. Negen andere hogescholen hebben zich inmiddels aangesloten bij dit platform. Eén van de activiteiten van dit platform is het opleiden van hbo-docenten tot docent Computer Vision. Meer dan 25 hbo-docenten volgden daartoe de vijfdaagse cursus Computer Vision van de NHL.

Het Platform Beeldverwerking HBO heeft als doel:

- een brugfunctie te vervullen voor het hbo bij het vergaren, uitwisselen en verankeren van de kennis en expertise op het gebied van beeldverwerking;
- het als zodanig fungeren als kennis(sen)netwerk rond beeldverwerking in het hbo in Nederland;
- een kennistransfer tussen hbo-instellingen en het bedrijfsleven te bewerkstelligen door o.a. het ontwikkelen van curricula/practica/cases t.b.v. cursussen voor het bedrijfsleven;
- het stimuleren van onderwijs en contractactiviteiten op het gebied van beeldverwerking door hbo-instellingen.

De hoofdactiviteiten van Platform Beeldverwerking HBO zijn:

- Het opleiden van hbo-docenten tot docent Computer Vision: de docenten volgen de praktijkgerichte vijfdaagse cursus Computer Vision van de NHL. Deze cursus is de afgelopen 10 jaar gegeven aan studenten van de opleidingen Informatica, Elektrotechniek en Technische Informatica en Werktuigbouwkunde. De cursus omvat voor de reguliere studenten een workshop van 40 uur en 7 dagdelen project. De NHL heeft voor deze workshop en dit practicum de beschikking over een speciaal ingerichte zaal met 10 pc's, camera's, lenzen en belichtingsapparatuur. Het idee is dat deze cursus de basis gaat vormen voor het onderwijs op het gebied van beeldverwerking bij de andere hbo-instellingen.
- Het opzetten en bemannen van een helpdesk voor hbo-instellingen die onderwijs geven op het gebied van beeldverwerking.
- Het verder ontwikkelen van cursusmateriaal. De hogescholen maken onderling afspraken over het gebruik van elkaars cursusmateriaal.

- Het geven van cursussen beeldverwerking voor het mkb.
- Het organiseren van themabijeenkomsten en workshops. De themabijeenkomsten en workshops hebben als doel om zowel bedrijven ervan te overtuigen dat beeldverwerking een meerwaarde kan hebben voor hun bedrijf, als om bedrijven en andere deelnemers van het platform te informeren over ervaringen, trends, producten en diensten. Tevens is een doelstelling om het relatienetwerk uit te breiden. Doelgroep is zowel het mkb als het hbo.

De deelnemende hogescholen zijn:

- Penvoeder: NHL Hogeschool
- Avans Hogeschool
- Fontys Hogeschool
- Hogeschool van Amsterdam
- Hogeschool van Arnhem en Nijmegen
- Hogeschool Utrecht
- Hogeschool Rotterdam
- Hogeschool Rijswijk
- Hogeschool InHolland

De resultaten na vier jaar zijn:

- samenwerking tussen negen hogescholen;
- drie of vier bijeenkomsten per jaar bij een van de deelnemende hogescholen;
- kennistransfer tussen de negen hogescholen;
- 25 hbo-docenten opgeleid om het vak Computer Vision te kunnen geven;
- Cursus materiaal Computer Vision ter inzage op: <http://www.nhl.nl/werkenstudie/3333/computer-vision/studieopbouw/inhoud-studie>;
- meerdere hogescholen hebben een Computer Vision Lab ingericht en zijn met contractactiviteiten begonnen.

TOEKOMST

De vraag van het bedrijfsleven naar automatisering van visuele inspecties neemt de laatste jaren sterk toe. Omdat kwaliteitscontrole van producten steeds belangrijker wordt is de verwachting dat deze groei zal doorzetten. Kwaliteitscontrole m.b.v. Computer Vision is vaak sneller en betrouwbaarder dan het menselijk oog. Bij veel bedrijven zijn de mogelijkheden van Computer Vision nog onbekend en bedrijven die ermee zijn gestart zien steeds meer toepassingsmogelijkheden. Het Platform Beeldverwerking HBO ziet hierin een belangrijke rol om de kennis m.b.t. Computer Vision uit te dragen bij zowel het bedrijfsleven als het hbo. Het Platform Beeldverwerking HBO kan door het ontwikkelen van toegepast onderzoek een brugfunctie vervullen tussen het fundamenteel onderzoek van universiteiten en de toepassingen van Computer Vision in het bedrijfsleven.



Over de auteur

Jaap van de Loosdrecht
Coördinator NHL Hogeschool
Kenniscentrum Computer Vision
Tesselschadestraat 12,
8913 HB Leeuwarden
Telefoonnummer: (058) 251 1193,
mobiel: (06) 139 49 207,
fax: (058) 251 1115
j.van.de.loosdrecht@tech.nhl.nl
www.nhl.nl/computervision

Het Delta excellence programma

Bron: TINFON, 2008, nummer 4

IMDelta, een alternatieve wijze voor het inrichten van een excellenceroom.

Eric Slaats - Associate lector bij Fontys Hogeschool ICT
e.slaats@fontys.nl

SAMENVATTING

IMDelta bestaat 4 jaar en is gericht op het intensiever ontwikkelen van kennis en vaardigheden van excellente, talentvolle studenten. Hiervoor krijgen deze naast een deel van het reguliere programma een aparte, realistische en meer uitdagende leeromgeving aangeboden. Ze bouwen hiermee aan een sterk en onderscheidend portfolio. Dit geeft toegang tot betere (meer uitdagende) stages en afstudeerplaatsen en daarmee een betere start als beroepsbeoefenaar. Het programma is zodanig opgezet dat zowel Delta-studenten als de reguliere studenten, de opleiding en de bedrijfscontacten er baat bij hebben. Kenmerkend is een budgettaire neutrale inrichting.

TREFWOORDEN

Excellence, talent, portfolio

INLEIDING

Excellence heeft in het recente verleden weinig aandacht gekregen. De aandacht voor betere studenten uit zich in het hbo meestal in de vorm van verkorte leerroutes en schakelprogramma's richting universitaire opleidingen. Afgelopen jaar groeit het besef dat aandacht voor excellence mag en moet. Het meest in het oog springend is op dit moment het Siriusprogramma (1). Dit programma stelt 50 miljoen euro ter beschikking voor excellenciprogramma's in het hoger onderwijs voor de periode van 2008-2013. Doelstelling is de 10 tot

15% beste studenten actief op een zo hoog mogelijk niveau brengen. Daarnaast ziet een aantal instellingen

de mogelijkheid om het major-minormodel te gebruiken voor het inrichten van honourprogramma's. In die gevallen doen studenten naast hun reguliere programma een extra minor.

Veel opleidingen besteden vooral aandacht aan de gemiddeld presterende en zwakkere studenten. Dit wordt mede geprikkeld door bekostiging op doorstromen. Op zich is hier niets mis mee; uiteindelijk leven opleidingen van al hun studenten. Als dit echter tot gevolg heeft dat er te weinig aandacht is voor de betere studenten - met als argument: 'die halen de opleiding toch wel' - dan blijft een aantal kansen liggen, zowel voor deze studenten als voor de opleidingen. Voor excellente studenten geldt dat ze zich moeilijk extra kunnen profileren. Mogelijk gaan ze zich vervelend omdat ze hun talenten binnen de opleiding niet optimaal kunnen ontwikkelen. Opleidingen laten hierdoor kansen liggen om zich te profileren met excellente afstudeerders en - mede daardoor - extra aantrekkelijk te worden voor betere instromers. Immers, wie zich richt op middelmaat, krijgt een middelmatige uitstroom.

IMDELTA

In 2005 is ICT Media Design (IMD), een afstudeer-variant van Bedrijfskundige Informatica binnen Fontys Hogeschool ICT, gestart met een programma voor excellente studenten: IMDelta. Dit programma was



een initiatief van Miranda Valkenburg (destijds propedeusecoördinator en medeontwikkelaar van IMD) en de auteur van dit artikel (destijds projectleider IMD).

Aanleiding voor IMDelta was een samenloop van omstandigheden. IMD was een jaar eerder gestart en begon op dat moment aan het tweede studiejaar. Het eerste cohort studenten voelde zich duidelijk pionier en was erg betrokken. Tijdens klankbordsessies met studenten vroeg een aantal studenten expliciet om meer uitdaging. In het voorjaar van 2005 bezocht een aantal medewerkers tijdens de studiereis van het HBO-I een aantal Amerikaanse universiteiten waar excelleren heel normaal gevonden wordt (2). De wijze waarop daar met excellente studenten wordt omgegaan, was een eyeopener en een bron van inspiratie. Tegelijkertijd werd op een school voor voortgezet onderwijs in Eindhoven een buitenschools programma voor betere leerlingen ingetrokken met als gevolg duidelijk zichtbare verveling en leegloop. Zo moet het dus niet. Tot slot hadden we de ambitie om IMD binnen en buiten het eigen instituut op een krachtige wijze onder de aandacht te brengen en te laten zien waartoe onze studenten in staat zijn.

CONTEXT

Om de werking van het Deltaprogramma te kunnen waarderen is het prettig om de context te kennen waarbinnen dit programma gestalte heeft gekregen. Deze paragraaf beschrijft globaal het profiel IMD en enkele onderdelen van de gehanteerde didactiek. IMD opereert op het snijvlak van ICT, communicatie en design. Samen met algemene beroepsvaardigheden vormen deze drie onderdelen de leerlijnen die gedurende de eerste twee jaar elke lesperiode in het rooster terugkomen. Samen dekken deze vier componenten het beroepenveld af waarvoor IMD opleidt. Elke lesperiode werken studenten aan een integraal project waarin het verband tussen de onderdelen tot leven komt.

Na het tweede studiejaar volgt een half jaar stage, waarna de studenten nog een jaar in de schoolbanken moeten voordat ze gaan afstuderen. In het laatste theoriejaar volgen ze voor de helft van de tijd een externe minor. De andere helft is meer thematisch ingericht, bijvoorbeeld gamedesign of e-learning. Tot slot zijn de studenten verplicht om elk studiejaar twee vrije studiepunten (EC) in te vullen met opdrachten uit de beroepspraktijk.

Bij IMD worden studenten beoordeeld op basis van de beroepsproducten die ze binnen de studieonderdelen en projecten maken. Deze beroepsproducten zijn in feite portfolio-elementen die de studenten als bewijslast hanteren voor competent gedrag. Een bijzonder kenmerk is dat veel van deze studentenproducten direct bruikbaar zijn voor eindgebruikers. Voorbeelden van producten zijn een webshop, een game, een viral of een uitgewerkte crossmediale campagne.

DOEL

Het Deltaprogramma is erop gericht om getalenteerde studenten (de Delta's) zichtbaar op een hoger niveau te brengen. Dit moet zijn beslag krijgen in extra kennis en vaardigheden, maar ook in een aantal andere tastbare extra's, zoals een excellent portfolio, een excellente stage- en afstudeerplaats, extra onderwijs en een certificaat.

Excellent portfolio

Voor alle IMD-studenten is een portfolio (*showcase*) van essentieel belang. Dit is een (digitaal) platform waarmee ze aan bijvoorbeeld bedrijven hun vaardigheden en capaciteiten kunnen tonen. De Delta's krijgen de mogelijkheid een excellent portfolio op te bouwen met opdrachten die reguliere studenten minder snel krijgen. Dit excellent portfolio, vaak ondersteund door testimonials van derden, geeft de Delta's een vliegende start na hun afstuderen.



Henk Plessius opent het buffet

Excellente stage- en afstudeerplaatsen

IMD spant zich in om voor de Delta's excellente stage- en afstudeerplaatsen te werven. Inmiddels heeft dit zich ontwikkeld van aanbodgericht naar vraag-gestuurd. Stagebiedende organisaties vragen expliciet naar Delta's.

Extra onderwijs

De opdrachten die Delta's krijgen, zijn zeer van uiteenlopende en vaak complexe aard. Het reguliere onderwijs kan onmogelijk vooraf inspelen op de benodigde kennis en vaardigheden voor al deze specifieke opdrachten. Delta's kunnen daarom aangeven op welk gebied ze extra ondersteuning en onderwijs wensen. Hierdoor verwerven ze een beter kennis- en uitvoeringsniveau dan reguliere studenten.

Certificaat

Wanneer ze afstuderen, ontvangen Delta's bij hun getuigschrift een certificaat waarin staat dat zij hebben deelgenomen aan het excellenceprogramma.

Het Deltaprogramma moet niet alleen meerwaarde hebben voor studenten, maar ook voor de organisatie. Ten eerste heeft het aanbieden van het programma een wervend effect. Tijdens open dagen merken we dat een aantal scholieren vooral hierop afkomt. Verder heeft het programma invloed op het imago: de uitstroom van studenten die duidelijk excelleren, versterkt het imago van de hogeschool zowel naar potentials als naar beroepenveld. Ten derde is het voor docenten een uitdaging om de Delta's te begeleiden. Welke docent smult immers niet van een groep gemotiveerde en getalenteerde studenten? Tot slot dragen Delta's actief bij aan het reguliere curriculum. Een van hun verplichtingen is hun eigen kritiekpunten onder de loop te nemen en verbeteringen aan te dagen.

Van dit laatste profiteren ook de reguliere studenten. IMDelta levert rijker en beter onderwijsmateriaal op. Van de aantrekkingskracht die de Delta's hebben op het werkveld, hebben alle IMD-studenten voordeel.

INRICHTING

In het onderwijs zijn de middelen schaars, zeker voor een extra programma. De ambitie was dan ook een budgettair neutrale inrichting qua middelen. Een tweede uitdaging vormde de werving van excellente studenten. Een extra programma moet bij studenten concurreren met vrienden en vriendinnen, met bijbaantjes, met hobby's enzovoort. Een boodschap in de trant van 'het is goed voor je' is niet voldoende. Bij de inrichting van IMDelta zijn beide uitdagingen - kostendekkend en concurrerend - getackeld.

Middelen

De tijd die Delta's moeten besteden aan het Delta-programma wordt gehaald uit vrijstellingen voor de reguliere studentenprojecten en de vrije studiepunten. Hiermee is tevens een deel van de bekostiging geregeld. De inzet die vrijkomt voor de project-begeleiding, kan gestoken worden in het begeleiden van Delta's. Om te kunnen concurreren met bijbaantjes mogen de Delta's 80% van het geld houden dat ze met de projecten verdienen. Hiervan gaat 10% in een eigen onkostenpot. De andere 20% gaat naar de opleiding en wordt gebruikt om Deltagroepen te faciliteren.

Opdrachten

In het tweede leerjaar voeren de Delta's geen voorgeschreven projecten uit met vaste deliverables, maar acquireren ze zelf opdrachten. Hierbij moet het leereffect voorop staan (en niet de verwachte inkomsten). Bij de acquisitie zijn de Delta's zelf verantwoordelijk voor een evenwichtig portfolio: alle studieonderdelen moeten aan bod komen. In de praktijk betekent dit dat

een groep van zo'n 10 Deltastudenten in een studiejaar geen 4 projecten uitvoert, zoals reguliere studenten, maar ongeveer 15. Omdat Delta's niet in vaste onderwijsperiodes (blokken of semesters) hoeven te werken, zijn ze veel vrijer en flexibeler in het aannemen en uitvoeren van opdrachten.

Vanuit het onderwijs geldt een aantal voorwaarden. De eerste opdracht die de Delta's krijgen, is zichzelf te organiseren. Dat betekent een voorzitter aanwijzen, rollen verdelen, een website aanmaken voor hun portfolio, een eigen huisstijl ontwikkelen en logo ontwerpen, visitekaartjes maken, enzovoort. Verder moet de groep minimaal 1 benefietproject voor een goed doel uitvoeren, zoals KiKa, CliniClowns en War Child. Ook voeren ze minimaal 1 opdracht uit voor de hogeschool. Opdrachten worden pas aangenomen als de Deltabegeleiding een fiat geeft. De beoordeling vindt plaatst via assessments, waarin de Delta's hun portfolio meenemen.

Examenfase

In de examenfase (derde/vierde studiejaar) ziet het programma er iets anders uit. Doordat Delta's een minor volgen hebben ze minder tijd beschikbaar voor Deltaopdrachten. Voor de examenfasegroep geldt dat de projecten dan ook vooral intern gericht moeten zijn. Er mogen projecten voor externe opdrachtgevers uitgevoerd worden, maar de opdracht is: 'laat een permanente indruk achter binnen de opleiding en verbeter het onderwijs'.

Dit betekent dat deze studenten innovatieve projecten binnen de hogeschool kunnen oppakken en ook delen van onderwijs ontwikkelen. Hierbij doen ze kennis en vaardigheden op met technologieën en werkwijzen uit het voorfront van de ontwikkelingen in het veld. Enkele voorbeelden: werken met AJAX toen dit uitkwam, het ontwikkelen van een Multitouch-tafel, een touch screen bouwen met een Wii-remote.

Voortgang

Delta's blijven niet zomaar Delta's. In het tweede jaar vindt halverwege het jaar een voortgangsgesprek plaats met elke Delta. Het komt voor dat er studenten uit de Deltagroep gehaald worden. Andersom komt ook voor. Een student waar 'Deltatwijfel' over bestond, kan zich dusdanig bewezen hebben dat toetreden tot de groep mogelijk wordt.

Als studenten terugkomen van hun stage, dan volgt voor de Delta's een nieuwe intake voor het examen-fasedeel. Sommige kiezen er expliciet voor om het reguliere programma te volgen. Met anderen worden afspraken gemaakt over hun persoonlijke ontwikkeling in de Deltagroep.

SELECTIE

Wanneer is iemand een excellente student? Uit het bovenstaande blijkt dat Deltastudenten niet alleen excellent zijn in het verwerven van kennis en vaardigheden, maar ook de nodige conceptuele creativiteit bezitten. Daarnaast moeten ze organisatorisch vermogen hebben, niet bang zijn voor het onbekende, met trots de vlag van hun opleiding dragen en proactief willen en kunnen reageren.

Dat is nogal wat. Hoe selecteer je aan je aan het einde van de propedeuse dergelijke studenten uit de populatie?

De gehanteerde definitie luidt: *een Deltastudent is in staat om succesvol in een hoger tempo door de stof heen te gaan met een minimale bemoeienis van de docent. Hierdoor heeft een Delta meer tijd om rondom een bepaalde kenniscomponent producten te maken en dingen te leren. Deze extra tijd wordt in het programma optimaal benut om meer en diepere kennis en vaardigheden te ontwikkelen.* In feite wordt dus extra werk in vrijgekomen tijd verwacht. Beter, sneller, meer, excellent!

IMD-studenten krijgen vrij snel na aanvang van de studie, in het eerste semester van de propedeuse, voorlichting over het Deltaprogramma. Ze hebben dan

gedurende de propedeuse de tijd om zich te profileren en na te denken of ze in een dergelijk programma kunnen en willen meedraaien.

Het Deltaprogramma bestaat vier jaar en heeft vier groepen Delta's opgeleverd. Daarnaast is er een kleine selectie geweest voor de februari-instroom. Deze laatste groep levert echter beduidend minder deelnemers op.

De eerste lichting

Het eerste cohort Deltastudenten is geselecteerd door Miranda Valkenburg en de auteur van dit artikel. Als initiatiefnemers, curriculumontwikkelaars, onderwijs-uitvoerders en dagelijkse leiding van IMD kenden wij alle circa 130 studenten persoonlijk. We observeerden hun functioneren dagelijks en maakten hen mee in les- en projectsituaties. De selectie kon daardoor redelijk 'intuïtief' plaatsvinden. Uiteraard hebben we ook gekeken naar studieresultaten, maar de groeps-samenstelling, de onderlinge verhoudingen en de (verwachte) groepsdynamica waren doorslaggevend. De 11 geselecteerde studenten waren er trots op om de pioniersgroep te zijn en hebben zich daarnaar gedragen. Deze groep, die zichzelf de naam Mindburst gaf, heeft uitstekend gefunctioneerd. In de examen-fase heeft slechts één deelnemer afgehaakt.

De tweede lichting

Bij de selectie van de tweede lichting is aan docenten die onderwijs verzorgden in de propedeuse gevraagd om *potentials* voor te dragen. Daarnaast mochten studenten solliciteren naar een Deltapositie. Alle voorgedragen studenten en sollicitanten zijn uitgenodigd voor een gesprek, mits hun studieresultaten een goed beeld gaven. Zij hebben een gesprek gehad met de eerste lichting Delta's en met de begeleiders. Op basis daarvan is een selectie gemaakt. Deze groep functioneerde beduidend minder dan de eerste lichting. Een aantal studenten is tussentijds uit

de Deltagroep gehaald, omdat hun prestaties onder de maat bleven. Een aantal van hen haakte zelf na een jaar af.

De derde lichting

Op basis van deze ervaring is besloten een intake-assesment in te richten. Kandidaten moesten door minimaal twee docenten voorgedragen zijn en er werd gekeken naar hun prestaties in de propedeuse. Hierna volgden gesprekken met de zittende Delta's en met docentenduo's, die gestandaardiseerde vragenlijsten hanteerden. Groepsdynamiek en motivatie waren de belangrijkste gespreksonderwerpen. Alle studenten hadden een uitstekende staat van dienst in de propedeuse. Deze 'koninklijke weg' heeft de slechtst functionerende groep opgeleverd! Een aantal zeer goed presterende studenten wilde er na een aantal weken uit, omdat Delta te veel tijd kostte. Verder deugde de groepsdynamica niet. Ze kwamen hun afspraken niet goed na, konden zaken niet voor de deadline afronden en hadden geen *sense of urgency*. Na een half jaar zijn twee nieuwe studenten aan de groep toegevoegd, waarna het tijt keerde en de kwaliteit steeg.

Wat hebben we hiervan geleerd? Uit een evaluatie bleek dat docenten bij selectie verschillende criteria hanteerden, al dan niet bewust. Deze werkwijze levert uitstekend functionerende individuen op, maar niet per definitie een goed functionerende groep. Met twee gelijkgestemden docenten die de selectie uitvoerden lukte dat wel.

De vierde lichting

Op basis van voorgaande ervaringen heeft de selectie van het vierde cohort als volgt plaatsgevonden. Docenten die propedeusestudenten begeleiden, dragen potentials voor. Wanneer zij positief scoren op de verschillende studieonderdelen, worden ze als potential benaderd. De uiteindelijke selectie wordt

gedaan door twee docenten die tevens de Delta's begeleiden. Veel aandacht is er voor motivatie, samenwerking en groepsvorming. Bij de selectie tellen meetbare prestaties even zwaar als het 'gevoel' dat de twee begeleiders hebben.

Deze vierde lichting is nu drie maanden aan de slag. De eerste signalen zijn hoopgevend. Het is een actieve, enigszins vrijevochten groep die alle gevraagde kwaliteiten in zich heeft. De ambities en prestaties tot nu toe liggen in ieder geval al hoger dan de vorige groep.

Om als Delta goed te kunnen functioneren, is het niet alleen noodzakelijk dat de studenten individueel goed presteren, maar dat tevens de groep als geheel goed functioneert. Functioneert de groep niet optimaal, dan kunnen de groepsleden ook niet excelleren. Het is dus aan de opleiding om voor optimale formaties en omstandigheden te zorgen.

BEGELEIDING

Begeleiding van de Delta's wordt voor een deel bekostigd uit de middelen die vrijkomen door vrijstellingen. Tevens genereert de groep zelf middelen. De Delta's moeten een hoge mate van zelfstandigheid hebben en krijgen hier ook de ruimte voor. Dat laatste overigens ook letterlijk: ze hebben een eigen lokaal tot hun beschikking. De groep wijst een projectleider of voorzitter aan, die de dagelijkse leiding heeft. Voor elk project dat de Delta's opnemen regelt de voorzitter dat er een projectgroep wordt samengesteld met projectleider.

Daarnaast is er inmiddels een aantal documenten gemaakt die de groep helpen om projecten op een juiste manier aan te pakken. Denk aan workflows, breakdowns, opstellen van offertes en facturen, enzovoort. Eerdere groepen hebben hier expliciet om gevraagd.

De Delta's voeren projecten inhoudelijk zelfstandig uit. Wanneer ze vragen hebben of tegen problemen aanlopen, kunnen ze een beroep doen op vakdocenten. De coaching heeft zich in de loop der jaren sterk ontwikkeld. Voor de huidige lichting gelden de volgende afspraken. Ten eerste worden alle Delta's door twee docenten begeleid; de ene docent begeleidt de tweedejaars, de andere de examenfasestudenten. Dit geeft eenduidigheid in de begeleiding. Ten tweede krijgt elk project dat de groep aanneemt, een heldere briefing en debriefing. De begeleidende docent, die daarbij aanwezig is, bepaalt uiteindelijk of het project wordt aangenomen. Deze werkwijze voorkomt te lichte of vage projecten en zorgt voor een haalbare hoeveelheid projecten. Ten derde kunnen Delta's met inhoudelijke vragen terecht bij vakdocenten. Ten vierde maken alle Delta's een POP (persoonlijk ontwikkelplan) waarin met name hun leervraag naar voren komt. De begeleider houdt in de gaten of hier voldoende aandacht naar uitgaat. Tot slot houdt elke groep een presentatie voor de nieuwe propedeusestudenten om het Deltaprogramma onder de aandacht te brengen. Dit zorgt ervoor dat de projecten presenteerbaar naar een breder publiek moeten worden opgeleverd.

Begeleiders hebben daarnaast met name de taak de groep te stimuleren en zijn ding te laten doen.

Bestuurders van Utrechtse onderwijsinstellingen in gesprek in de openingssessie: Jan Bogers (HU), Bernard Fransen (ROC Midden NL) en Hans Stoof (rector UU). Karin Verkerk (ROC ASA) nam ook deel aan de openingssessie maar staat niet op de foto.



CONCLUSIES

Het Deltaprogramma is in brede zin zeer succesvol geweest voor zowel IMD als de participerende studenten. De eerste lichting is net afgestudeerd en de afstudeerresultaten van deze groep zijn zeer bevredigend. De spin-off naar de opleiding is positief. Zo heeft de eerste groep een aantal 'blijvende' producten en memorabele projecten gedraaid die hun sporen hebben achtergelaten. Daarnaast is Delta een unieke selling point voor IMD aan het worden. Bedrijven waar contacten mee bestaan, kennen het Deltaprogramma en vragen er soms expliciet naar. Het verwerven van opdrachten is ook geen enkel probleem meer. Delta lijkt een selffulfilling prophecy te worden. Studenten die deel uitmaken van deze groep, willen graag excelleren.

Echter, we zijn er nog niet! Het feit dat groepsdynamica zo'n stempel drukt op het slagen van Delta, is een zorgpunt. Gezien de totaal verschillende aard van de groepen is hier nog winst te behalen. Daarnaast is het feit dat ook 'gevoel' bij selectie een rol speelt wat onbevredigend. Ervaringen in de examenfase zijn nog pril. Daarnaast wordt geëxperimenteerd met een multidisciplinair project met andere instituten om hiermee het ervaringspalet voor de studenten verder uit te breiden. Hierdoor komen tevens projecten in beeld die anders simpelweg niet uit te voeren zouden zijn.

Delta heeft zich voor IMD bewezen en zal een blijvend onderdeel van het curriculum vormen. De missie is nu om deze werkwijze proberen elders te 'verkopen' zodat samenwerking op dat niveau mogelijk wordt over de grenzen van vakgebieden heen.



[1]

www.siriusprogramma.nl

[2]

Smeets, D.A.J. en M.W. Valkenburg (2005), HBO-I on tour in Silicon Valley. In: TINFON 2, 24e jaargang, p.49-56.

IMD : <http://fhict.fontys.nl/M-Profiel/Pages/default.aspx>

Redsquare: www.redsquare.nl

12

UvA Webklas Informatica: Wat is een Programma?

Alban Ponse - Universiteit van Amsterdam

Stephan Schroevens - Universiteit van Amsterdam
{alban, sschroev}@science.uva.nl

SAMENVATTING

De Webklas Informatica 'Wat is een programma?' is een korte cursus voor scholieren uit 5 en 6-vwo met een Wiskunde B-profiel. Deze webklas wordt sinds 2004 online aangeboden. De deelnemer leert een eenvoudige, imperatieve programmeertaal (programmeerervaring is niet nodig). Het inleveren van opgaven en de bespreking van resultaten en problemen gaat per e-mail. Bij sommige opgaven wordt een toolkit gebruikt, andere zijn theoretisch. De studielast is 10 uur en na succesvolle afronding volgt een certificaat.

TREFWOORDEN

Webklas, Internetcursus, Imperatief Programmeren

INLEIDING

We bespreken een sterk verkorte, maar serieuze versie van een collegeserie van de Universiteit van Amsterdam (UvA) over een actief onderzoeksgebied op de UvA. Deze cursus is bedoeld voor scholieren uit 5 en 6-vwo met een profiel met Wiskunde B 1/2, maar de ervaring leert dat ook een enkele vierdeklasser of hbo-student deze cursus met plezier en succes kan doorlopen.

De UvA biedt verschillende webklassen [7] aan. Een UvA webklas is een vierwekelijkse interactieve cursus die individueel via het internet kan worden gevolgd; de studielast van een webklas is tien uur. Succesvolle



deelname aan een webklas wordt beloond met een certificaat. Sinds 2004 wordt de Webklas Informatica met de titel 'Wat is een programma?' aangeboden [3]. De opzet van deze webklas is de deelnemer kennis te laten maken met de programma-algebra. Programma-algebra is een wat formele benadering van de in de titel opgeworpen vraag en bedacht door Jan Bergstra (zie [2] voor een uitgebreide introductie in het Nederlands).

DE WEBKLAS INFORMATICA

De op [7] aangeboden Webklas Informatica wordt als volgt ingeleid:

Je hebt ermee te maken wanneer je iemand opbelt, geld opneemt, aan het chatten bent, een boek opvraagt bij de bibliotheek en wanneer je op reis gaat: de computer. De computer doet echter niets vanzelf. Als je de computer iets wilt laten doen, dan heb je een programma nodig. In deze webklas gaan we in op de vraag 'wat is een programma?'. Dit is geen eenvoudige vraag, zelfs niet voor de deskundigen. We behandelen enkele fundamentele aspecten rond deze vraag en definiëren een paar eenvoudige programmeertalen. We kijken op een systematische wijze naar het grote verschil tussen syntax (de programmatekst) en semantiek (het gedrag van een programma bij uitvoer). Ten slotte bestuderen we een paar toepassingen met behulp van interactieve tools, waaronder het 'rekenen zonder getallen'.

De opzet van de Webklas Informatica is zodanig dat van de deelnemers vooraf geen enkele ervaring met programmeren wordt verwacht. Gedurende de eerste twee weken werken de deelnemers voornamelijk met programma's die abstracte instructies (a, b, c, . . .)

uitvoeren en wordt het verschil tussen syntax en semantiek duidelijk gemaakt. Een programma is een rijtje instructies, bijvoorbeeld `+a;b;c` waar `+a` een positieve testinstructie en `b` en `c` basisinstructies worden genoemd.

Het idee is dat uitvoering van een positieve testinstructie een antwoord `true` of `false` oplevert; na antwoord `true` wordt de volgende instructie uitgevoerd en na `false` wordt de eerstvolgende instructie overgeslagen. Na uitvoering van een basisinstructie wordt de volgende instructie uitgevoerd. Als er geen instructies meer zijn uit te voeren volgt terminatie.

De semantiek (betekenis) van programma's wordt beschreven door threads. Deze threads hebben een andere notatie dan programma's en zijn opgebouwd uit constanten voor terminatie en deadlock, en uit acties die gelijknamig zijn aan de basisinstructies. De semantiek van het programma hierboven is de thread

`b.c.S <a> c.S`

Deze thread beschrijft het gedrag

if `a` then `b.c.S` else `c.S`

en kunnen we als volgt stap voor stap uitleggen: de thread `b.c.S <a> c.S` start met de actie `a`. Als deze actie `true` oplevert, volgt de linkertak van `..<a>..`, dus `b.c.S`, en na antwoord `false` de rechtertak van `..<a>..`, dus `c.S`. De thread `b.c.S` start met de actie `b`, gevolgd door actie `c`, gevolgd door terminatie, aangeduid door de letter `S`. Het zich voltrekken van een actie leidt altijd tot een antwoord `true` of `false` en mogelijk tot een verschillend vervolgedrag zoals bij `a` hierboven. In andere gevallen doet dit antwoord er niet toe zoals in dit voorbeeld na de acties `b` en `c`. Een eindige thread eindigt altijd in terminatie (`S`) of deadlock (dit laatste fenomeen bespreken we hier niet).

De uitvoering van het programma `+a;b;c` begint dus met uitvoering van de eerste instructie `+a` wat tot de actie `a` leidt; na antwoord `true` volgt uitvoering van de volgende instructie (dus `b`) en actie `b` vindt plaats; na antwoord `false` wordt de eerstvolgende instructie overgeslagen en dus de `c`-instructie uitgevoerd (de actie `c` vindt plaats). Na uitvoering van de basisinstructie `b` volgt ook uitvoering van `c`. Na uitvoering van de basisinstructie `c` termineert dit programma.

Programma's kunnen ook controle-instructies zoals labels en go-to's bevatten, bijvoorbeeld

`+a;##L0;b;c;L0`

Hier is `##L0` de instructie 'go-to `L0`'. De instructie `L0`, spreek uit 'label 0', heeft alleen als doel een positie in het programma te markeren en levert bij uitvoer geen gedrag op. Het gedrag van het programma `+a;##L0;b;c;L0` is de thread

`S <a> b.c.S`

Informatiemarkt



(dus de go-to-instructie `##L0` en de labelinstructie `L0` worden in dit voorbeeld gebruikt om over de basisinstructies `b` en `c` heen te springen). Een korter programma met hetzelfde gedrag is `+a;##L0;b;c` want de afspraak is dat een go-to instructie naar een niet in het programma voorkomend label tot terminatie leidt.

Tot slot wordt ook nog kennis gemaakt met instructies die `if ... then ... else ...`

representeren: het programma `+a{b;c};c;b;` voert `b;c` uit indien de actie `a` antwoord `true` oplevert, en anders `c;b`. Het vertrouwd raken met de syntax van programma's en threads, en het afleiden van threads uit programma's volgens de gegeven regels voor threadextractie is het belangrijkste leerdoel in de eerste twee weken. Deze stof is niet makkelijk en vergt oefening: in deze periode moeten 26 korte opgaven worden gemaakt en is de stof opgedeeld in zes hoofdstukken. Daarnaast zijn er drie hoofdstukken

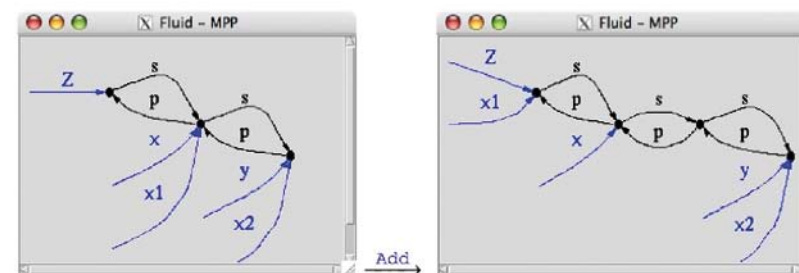
met een asterisk gemarkeerd: deze hoofdstukken gaan over het beschrijven van oneindig gedrag, projectie-semantiek en over input-outputrelaties, en hebben elk hun eigen opgaven (genummerd met Romeinse cijfers) die deelnemers facultatief mogen inleveren.

De stof van Week 3 gaat over een eenvoudige theorie genaamd Moleculaire Dynamica (naar analogie met de chemie) die toestanden in een vloeistof beschrijft. Een vloeistof bestaat uit moleculen, welke op hun beurt zijn opgebouwd uit atomen. Atomen zijn verbonden door velden: gerichte bindingen (pijlen) met een label. Een focus is een unieke atoomreferentie. Verder wordt een verzameling basisinstructies (MPP) voor de manipulatie van een vloeistof geïntroduceerd, zoals `x=new`, de instructie die de creatie van een nieuw atoom met focus `x` voorschrijft. Er wordt vanaf dit punt niet meer met abstracte instructies `a`, `b`, ... gewerkt.

```

Add  $\stackrel{\text{def}}{=} L0; +x1/p; x1=x1.p;
+x2/s; x2=x2.s;
\}; x2.+s; y=new; y.+p;
y.p=x2; x2.s=y; x2=x2.s; \};

##L0;
\}; y=x2; \}$ 
```



Figuur 1:
Rekenen met moleculen.

In Week 4 wordt MPP ten slotte gebruikt om te rekenen zonder getallen: een getallenlijn is een sliert van atomen die door middel van p en s velden met elkaar verbonden zijn, tezamen met een speciaal atoom Z dat het getal 0 representeert. Voor ieder atoom (getal) verwijzen p en s naar respectievelijk de voorganger en opvolger van dat getal in de natuurlijke getallen (Z heeft dus geen voorganger, en het laatst gecreëerde atoom heeft geen s-veld). Foci x, y, . . . worden gebruikt als variabelen; hun waarde is die van het atoom waarnaar zij verwijzen. Figuur 1 toont een programma dat de getallen x1 en x2 optelt en het resultaat opslaat als y. Deze figuur toont tevens een vloeistof waarin $x1 = 1$ en $x2 = 2$ (links), en de resulterende vloeistof na het toepassen van de functie Add (rechts) met resultaat $y = 3$.

Het materiaal voor Week 3 en Week 4 is opgedeeld in zes hoofdstukken en telt tien opgaven (vijf in elke week). In totaal bevat de webklas 36 opgaven en (voor de eerste twee weken) nog tien extra facultatieve opgaven. Voor een gedetailleerde beschrijving van de inhoud van de Webklas Informatica en overige informatie verwijzen we naar [4, 3].

Quotes

De Webklas Informatica heeft drie studenten Informatica opgeleverd, waaronder Daan Staudt (zijn afstudeeropdracht met resultaat 8,5 werd gepubliceerd als [8]). Hij diende ten behoeve van toekomstige werving de volgende quote in: *‘Mijn naam is Daan. In mijn zesde jaar op het Vossius Gymnasium, in 2004, heb ik met veel plezier de Webklas Informatica gevolgd. Het was een geweldige ervaring om meer inzicht te krijgen in wat een programmeertaal nu precies is. Na het volgen van deze webklas had ik een concreter beeld bij de opleiding Informatica en wist ik zeker dat deze opleiding iets voor mij was.’*

Daan was de eerste deelnemer aan de Webklas Informatica die besloot om Informatica aan de UvA te gaan studeren, hij volgde de webklas in 2004. Ook in 2005 en 2006 leverde de webklas een toekomstige UvA student op. Hieronder volgt nog een aantal citaten waarmee we het karakter van het intensieve e-mailcontact tussen deelnemers (cursief) en de webklasdocent (gewoon schrift) willen illustreren.

Uit 2005:

Ik wil jullie heel erg bedanken voor de mogelijkheid om een beter inzicht te krijgen in Informatica, ik overweeg nu om dit als tweede studie te nemen. Echt tof dat jullie dit doen!

hoihoi, hier de laatste paar opgaven. Geniet ervan en het zou fijn zijn als je me 36 kon uitleggen, want ik heb daar net een uur met mijn zus over gebeld en die vond dat het dan zo maar moest, maar ik ben het daar nog steeds niet mee eens.

Zie hier met enige vertraging de antwoorden op de opgaven uit de vierde en tevens de laatste week van de zeer informatieve, educatieve en ronduit gerieve webklas informatica. Stephan, het was een genoegen, dank je voor toekomstige moeite en voor de moeite in het verleden en excuus voor de vertraging. Tabee!

Bij opgave 28 moet je het resulterende molecuul beschrijven, niet hoe het wordt opgebouwd. Je merkt wel op dat het programma weer naar het eerste label springt (en impliciet dus dat de creatie van het molecuul oneindig lang doorgaat), dus ik reken je antwoord halfgoed. Bij het eerste molecuul van opgave 29 gebruik je enkel velden met label f. Die zitten echter helemaal niet in het molecuul. Waar je fout gaat, is door van links naar rechts te werken. Probeer eerst eens het meest rechteratoom te maken...

Je voelt ‘m al aankomen... Je bent geslaagd! Gefeliciteerd! Het certificaat zal waarschijnlijk nog even op zich laten wachten, maar het komt er zeker aan.

Oohja: je vind het vast leuk om te weten dat je de beste deelnemer was tijdens deze webklas. Ik heb twee vragen fout en twee vragen voldoende gerekend. De rest was helemaal goed. Nogmaals gefeliciteerd!

Uit 2006:

hier zijn mijn antwoorden van week 3, sorry dat ze zo laat zijn, antwoorden van week 4 heb ik niet, ik snap het namelijk niet, heb er wel mn best op gedaan. nogmaals excuses.

Zou je iets specifiekere kunnen zijn over wat je niet snapt van de laatste opgaven? Wellicht dat ik dan wat dingen kan verduidelijken. Je krijgt van mij nog een week extra de tijd om opgave 29 en 32 t/m 36 in te leveren als je wilt.

[...]

Ik neem je antwoorden even door:

27: Goed.

28: Goed.

29: Je geeft hier de code van de verkeerde moleculen. Het is de bedoeling dat je de twee plaatjes in de inleiding (<http://staff.science.uva.nl/alban/WebKlas/Week3/index.html>) namaakt. Je kan deze opgave opnieuw inleveren als je wilt.

Uit 2007:

Mijn commentaar op je uitwerkingen voor week twee: 22. Hier gaat een aantal kleine dingetjes mis. Je programma doet hetzelfde als het oorspronkelijke, maar sommige van de labels hebben een te grote waarde gekregen: [...] 23. Hier gaat een boel mis, maar ik denk dat dat gedeeltelijk ook komt doordat je het oorspronkelijke programma niet goed gelezen hebt: sommige

instructies missen gewoon. De uitwerking is: [...] T., bovenstaande lijkt veel commentaar, maar het gaat best wel goed. Gewoon zo doorgaan dus!

Uit 2008:

*Ik wil best extra uitleg geven, en daarbij misschien zelfs een voorbeeld gebruiken... maar dan moet ik natuurlijk wel weten *wat* je niet snapt. Zou je daarover kunnen uitwijden?*

[...]

Juist! (Zo, dat was een hele bevalling :))

[...]

Beide antwoorden zijn goed! En daarmee kan ik mededelen dat je de Webklas Informatica succesvol hebt afgerond! Het begin (week 1) verliep een beetje stroef, maar daarna heb je extra inzet getoond en is het toch allemaal goed gekomen.

ERVARINGEN

Het aantal deelnemers aan de Webklas Informatica fluctueert van jaar tot jaar. Tabel 1 toont enkele statistieken sinds november 2005. Zoals eerder gezegd zijn de webklasopdrachten verspreid over vier weken. De tabel toont voor iedere editie hoeveel deelnemers ten minste één opgave van een specifieke week hebben ingeleverd. Verder toont de tabel hoeveel scholieren daadwerkelijk geslaagd zijn.

Editie	Week	Week 2	Week 3	Week 4	n	Geslaagd %
November 2005	29	26	24	22	22	76
November 2006	8	8	7	6	4	50
November 2007	3	2	2	2	2	67
November 2008	13	10	10	5	5	38

Tabel 1:

Enkele statistieken.

De cijfers in Tabel 1 behoeven enige uitleg. In november 2005 hadden we veel deelnemers omdat twee wiskundeleraars van het Barlaeus Gymnasium in Amsterdam [1] hun leerlingen de mogelijkheid boden de Webklas Informatica als een alternatieve praktische opdracht (PO) te volgen. Voor deze scholieren gold de webklas dus als officiële studielast. De functie van de webklas als officieel studiemateriaal bracht een extra verantwoordelijkheid met zich mee: we hebben één van de scholieren van het Barlaeus na twee weken op het hart moeten drukken een andere PO te kiezen omdat duidelijk werd dat hij de Webklas Informatica niet succesvol zou afronden.

De toekomst

Allereerst willen we deze webklas blijven aanbieden. De ervaring heeft geleerd dat de maand november hiervoor een geschikte periode is. Ook zijn we door de ervaring nogal soepel met onze deadlines geworden en accepteren we vrij gemakkelijk enige vertraging (soms wel van een paar weken).

In het schooljaar 2008-2009 zijn in samenwerking met de Its Academy [5] en 4 scholen in de regio Hilversum 2 webklassen, waaronder de Webklas Informatica,

bewerkt tot zogenaamde 'onderwijsmodules' voor leerlingen van 5-vwo. Het doel van dit project is het ontwikkelen van verrijkingstof naast de in gebruik zijnde leergangen Informatica. Aan deze eerste proefronde namen 3 informaticadocenten en 50 leerlingen afkomstig van 4 scholen deel, en natuurlijk de begeleiders van de webklassen. Eind 2008 is de onderwijsmodule Informatiekunde 'Social Web' succesvol afgerond. De onderwijsmodule Informatica 'Wat is een programma?' heeft vanaf mei 2009 4 weken proefgedraaid (onderbroken door een vrije week en op één school ook nog een werkweek): per school werd 1 les per week door ons gegeven, en een tweede les door de eigen docent. Daarnaast werd intensief gebruik gemaakt van e-mail en internetcontact. De ervaringen wijzen uit dat de thematiek van deze module voor de 'gemiddelde deelnemer' nogal abstract is en dat het zorgvuldig lezen (en bestuderen) van enkele enigszins formeel getinte, korte tekstfragmenten (1 of 2 pagina's A4) op zich al een moeilijke opgave is. Niettemin zijn we met klem uitgenodigd volgend schooljaar terug te komen en door te gaan met dit project.

In het najaar van 2009 is de stof van onze onderwijsmodule tot eigenstandig lesmateriaal ontwikkeld (inclusief uitgebreide toolondersteuning via een webapplicatie) en is een docentenhandleiding

gepubliceerd. Verder blijven we vanuit de UvA als ontwikkelaars uitdrukkelijk beschikbaar voor incidentele ondersteuning van de docenten die deze onderwijsmodule willen gebruiken. Ten slotte blijven we ook beschikbaar voor ondersteuning van leerlingen die hun activiteiten voor de webklas of onderwijsmodule tot profielwerkstuk willen uitbreiden.

Bij elke editie van de webklas beschouwen we het cursusmateriaal kritisch en waar nodig proberen we dit te verbeteren en/of uit te breiden. De hierboven beschreven ervaringen in Hilversum met leerlingen die wel het vak Informatica volgden, maar er niet zelf voor hadden gekozen aan onze module mee te doen, vormden aanleiding om enkele links met extra uitleg en interactieve oefeningen te maken, om op die wijze de wat moeilijkere onderdelen zoals threadextractie beter over te brengen (zie [6] en bijvoorbeeld de verwijzing hiernaar in de inleiding van Hoofdstuk 3 van de webklas [3]).

VERANTWOORDING

De auteurs van de Webklas Informatica zijn de UvA-docenten Inge Bethke en Alban Ponse. Zij zijn ook de auteurs van de leergang 'Programma-Algebra, een Inleiding tot de Programmatuur' [2] die in het eerstejaarsonderwijs in de UvA Bachelor Informatica wordt gebruikt. Het leerdoel van dit onderwijs is het overbrengen van een algebraïsche benadering van de achterliggende concepten van het imperatief, sequentieel programmeren. De webklas is een sterk verkorte, maar serieuze versie van deze leergang en beoogt voor elke goedwillende deelnemer toegankelijk te zijn: voorkennis en ervaring zijn niet nodig en moeilijke begrippen zoals uit de algebra komen niet voor in het cursusmateriaal.

Stephan Schroevens is sinds enkele jaren de docent van de Webklas Informatica. Hij heeft jarenlange ervaring als assistent in het Informaticaonderwijs en als voorlichter van de studie Informatica. Stephan Schroevens is nu student in de Master of Logic (UvA) en doceert tevens ten behoeve van de Its Academy aan leerlingen van Amsterdamse en Hilversumse scholen. Het materiaal van de Webklas Informatica is vrij beschikbaar via de site [3].



[1]

Barlaeus Gymnasium, <http://www.barlaeus.nl/>.

[2]

I. Bethke en A. Ponse. Programma-Algebra, een Inleiding tot de Programmatuur. Amsterdam University Press, 2003. ISBN 90 5629 279 X.

[3]

I. Bethke en A. Ponse. Webklas Informatica, <http://www.science.uva.nl/alban/WebKlas/>.

[4]

I. Bethke en A. Ponse. Synopsis van de Webklas Informatica, <http://www.science.uva.nl/alban/WebKlas/Docent/Synopsis.html>.

[5]

Informatica, Techniek en Science Academy. Samenwerkingsverband tussen de UvA, VU, HvA en INHOLLAND en plm. 40 scholen in de regio Noord-Holland/Flevoland, <http://www.itsacademy.nl/>.

[6]

S. Schroevers. Voorbeelden en samenvattingen van de Webklas Informatica, <http://student.science.uva.nl/sschroev/pga/extra/>.

[7]

UvA Webklassen, <http://www.studeren.uva.nl/webklassen/>.

[8]

D. Staudt. A Case Study in Software Engineering with PSF: A Domotics Application. PRG-Report PRG0811, August 2008. Beschikbaar op <http://www.science.uva.nl/research/prog/publications.html>.

13

Werken 2.0: Van vak naar vaardigheid



Peter Becker - Haagse Hogeschool,
Academie voor ICT en Media

Jan Krans - Haagse Hogeschool,
Academie voor ICT en Media

SAMENVATTING

Op de Haagse Hogeschool is een minor ontwikkeld waarin studenten kunnen kennismaken met en zich kunnen voorbereiden op het Nieuwe werken. De minor is voorbereid en wordt uitgevoerd in samenwerking met een bedrijfspartner. De eerste ervaringen zijn positief. Studenten geven aan zowel het onderwerp als de inbreng van buiten waardevol te vinden.

TREFWOORDEN

Minor, samenwerking bedrijfsleven, nieuwe leren, web 2.0, verandermanagement.

INLEIDING

De afgelopen jaren is het zogenaamde 'Nieuwe werken' omarmd door zowel overheid, bedrijfsleven als politiek. Het nieuwe werken wordt vaak in verband gebracht met ICT, aangezien de technische ontwikkelingen het nieuwe werken faciliteren, maar feitelijk gaat het om een ontwikkeling waarbij organisaties op een andere manier worden georganiseerd en geleid. Kenmerken van het nieuwe werken zijn: minder sturing, grotere verantwoordelijkheid van de medewerker en de mogelijkheid een deel van de werkzaamheden plaats- en tijdonafhankelijk uit te voeren. Maatschappelijke ontwikkelingen als de toegenomen

individualisering, de opkomst van de kenniseconomie en de noodzaak van bedrijven om talent aan zich te binden vormen belangrijke factoren die de opkomst van het nieuwe werken teweegbrengen. Toch kan niet worden ontkend dat de ICT-ontwikkelingen van de afgelopen jaren een belangrijke voorwaarden-scheppende en stimulerende rol spelen bij de opkomst van het nieuwe werken.

Als opleidingen zijn we verantwoordelijk voor de scholing van mensen die de verschillende ICT-toepassingen zullen implementeren en installeren. Daarnaast zullen onze studenten ook als werknemer in aanraking komen met het nieuwe werken als concept waarbinnen ze hun professionele werkzaamheden verrichten. Het is daarom interessant de werknemers van de toekomst te laten kennismaken met de (on)mogelijkheden van Web 2.0 en hen te laten reflecteren op de invloed die het gebruik van nieuwe tools kan hebben op de structuur van een organisatie. Zij komen daarmee niet alleen in aanraking met de technische aspecten, maar vooral ook met kritische succesfactoren als steun van het management, bereidheid van medewerkers om kennis te delen en het veranderingsvermogen van een organisatie.

De opleiding Informatiedienstverlening en Informatie-management (IDM), onderdeel van de Academie voor ICT en Media van de Haagse Hogeschool organiseert in samenwerking met Atos Origin de minor 'Werken 2.0' waarin deze kennismaking en reflectie centraal staan. De eerste ervaringen met de minor zijn zeer positief en leiden wat ons betreft tot een aantal conclusies.

DE MINOR

Web 2.0 bepaalt de vorm en de inhoud van de minor die is opgebouwd uit drie delen.

1

Ervaring opdoen met de nieuwe mogelijkheden voor communicatie en kennisdeling door te werken met een aantal gratis beschikbare tools zoals Google Docs, Calendar, LinkedIn, Remember The Milk, Blogspot, Dropbox, SlideShare enzovoort.

2

Het werken aan een managementgame, gebaseerd op de fictieve Gemeente Briksendam. Hierin werkte de studenten scenario's uit voor een aantal processen bij een gemeente. Met de game kunnen ambtenaren werken aan een verbetering van de processen door 2.0-elementen toe te voegen.

3

Het uitwerken van een adviesopdracht over de toepassing van het nieuwe werken bij de overheid. Door middel van een vragenlijst werden ambtenaren van een aantal ministeries bevraagd over hun kennis van en houding tegenover het nieuwe werken. Ook werd in kaart gebracht wat de mogelijke belemmeringen om te starten met het nieuwe werken.

Bij de opdrachten gaat het niet alleen om het verwerven van kennis en ervaring, maar ook om het veranderen van de mindset.

In het onderwijs van de minor Werken 2.0 wordt het model van Atos Origin voor verandermanagement gehanteerd door achtereenvolgens de vijf fasen van verandering te doorlopen:

1

bewustwording

2

beeldvorming

3

ervaring

4

doen

5

verzorgen

Ad 1: **Bewustwording.** In de eerste fase worden de studenten door middel van gastcolleges en demonstraties geconfronteerd met verschillende maatschappelijke en technische ontwikkelingen. Het is gebleken dat studenten niet erg op de hoogte zijn van het bestaan van veel tools, laat staan dat ze zich bewust zijn van de mogelijkheden en consequenties voor professionele omgevingen.

Ad 2: **Beeldvorming.** Na de bewustwording wordt de studenten gevraagd een visie te formuleren ten aanzien van het nieuwe werken en de rol van ICT. Daarbij wordt extra aandacht besteed aan de persoonlijke situatie van de student door middel van de vraag: 'Hoe zie jij je professionele toekomst?'

Ad 3: **Ervaring.** Zoals aangegeven bepaalt Web 2.0 niet alleen de inhoud maar ook de vorm. Studenten worden gestimuleerd de verschillende online tools te gebruiken bij het groepsonderwijs. Hierbij gaat het om communities als LinkedIn en MySpace, Online agenda en omgevingen waarin samen aan documenten kan worden gewerkt. De minor kende bij de eerste uitvoering een heuse wereldprimeur: het nieuwste samenwerkingsplatform van IBM werd door de studenten gebruikt om informatie te delen en samen te werken. De docenten konden via het platform de voortgang van de studenten monitoren. De traditionele leeromgeving die normaal binnen het onderwijs wordt ingezet voor documentdistributie door de docent wordt bij deze minor niet gebruikt.

Ad 4: **Doen.** Teneinde inzicht te krijgen in de randvoorwaarden voor een succesvolle implementatie van 2.0-tools in een professionele setting voeren de studenten een quickscan uit bij verschillende ministeries. Uit de quickscan kwam als belangrijkste conclusie naar voren dat een groot deel van de respondenten positief staat tegenover de mogelijkheden, maar dat de techniek (firewall, geen mogelijkheden applicaties te installeren) een belemmerende factor is. Hierbij moet worden aangetekend dat bij dergelijk onderzoek diegenen die enthousiast zijn over het onderwerp eerder geneigd zijn de vragenlijst in te vullen.

Ad 5: **Verzorgen.** Op basis van hun bevindingen presenteren de studenten een advies over de implementatie van nieuwe samenwerkingstools binnen de overheid. Dat de overheid hiervoor open staat, blijkt uit het succes van het initiatief 'Ambtenaar 2.0'. De initiatiefnemer hiervan, ambtenaar Davied van Berlo, was al eens gastspreker bij de minor.

De ontwikkelingen ten aanzien van het nieuwe werken gaan snel. De inhoud van de minor ligt dan ook niet vast. Nieuwe tools, ontwikkelingen en inzichten worden zoveel mogelijk verwerkt in de stof en de opdrachten en regelmatig worden nieuwe gastsprekers uitgenodigd. Bij de volgende uitvoering zullen de studenten aan de slag gaan met het nieuwe platform Google Wave dat eind 2009 wordt gelanceerd.

BEVINDINGEN

We kunnen constateren dat de studenten compleet verrast worden door datgene wat hen wordt aangeboden. Dat informaticastudenten niet zo bekend zijn met genoemde maatschappelijke ontwikkelingen is misschien nog niet zo verrassend te noemen.

Opvallend is wel dat ze ook erg onbekend blijken te zijn met het scala aan tools, laat staan dat ze hebben nagedacht over de toepassing van deze tools in een professionele omgeving en de veranderingen die dit teweeg kan brengen.

Verder hebben we waargenomen hoe de oorspronkelijke passiviteit van de studenten compleet veranderde tijdens de minor. Na aanvankelijk een afwachtende houding te hebben aangenomen gingen de studenten vanaf de ervaringsfase zeer voortvarend te werk. Hierbij is het duidelijk dat de studenten sterk worden aangesproken door zowel de inhoud als de onderwijsvorm. Het grote aantal gastpresentaties en bedrijfsbezoeken speelt daarbij een belangrijke rol.

De studenten tonen zich na afloop van de minor zeer enthousiast over het geleerde. Ze geven aan dat hun visie op het nieuwe werken en het samenwerken binnen organisaties compleet is veranderd en de meesten richten de focus ook op de eigen professionele toekomst.

Bij de afsluitende presentaties wordt aan de studenten gevraagd daar een aantal sheets aan te wijden. Soms geven studenten aan in de tien weken van de minor gericht en effectiever met de eigen toekomst bezig te zijn geweest dan in andere onderwijssettings zoals gesprekken met een studiebegeleider. Zij laten deze uitspraken ook volgen door concrete acties. Dit beperkt zich niet tot werkzaamheden in het kader van stage of afstuderen, waarbij studenten zijn betrokken bij de implementatie van het nieuwe werken in organisaties; ook op persoonlijk vlak leidt dit tot vergaande inzichten en besluiten.

CONCLUSIES EN AANBEVELINGEN

Hoewel er in de minor geen sprake is van gericht onderzoek, is de conclusie gerechtvaardigd dat de huidige generatie studenten ondanks typering als 'digital natives' en 'generatie Y' slechts in beperkte mate bekend is met de technische mogelijkheden ten behoeve van samenwerking en communicatie. Ontwikkelingen als het Nieuwe werken zijn voor de studenten geheel nieuw, terwijl zij daar in hun professionele toekomst zeker mee in aanraking zullen komen. Dat geldt overigens niet alleen voor de toekomstige professionals in de ICT. In ieder beroepenveld, van jurist tot fysiotherapeut, zal de komende jaren sprake zijn van een grotere mate van kennisuitwisseling, eigen verantwoordelijkheid en samenwerking. Wat ons betreft is onderwijs in Werken 2.0 dan ook interessant voor studenten van alle disciplines. Dat levert direct voordeel op want iedere hbo-student kan nu al profiteren van de mogelijkheden die 2.0-tools bieden voor communicatie, netwerken, organiseren en samenwerken.

Binnen De Haagse Hogeschool wordt de minor voortaan aangeboden aan de studenten van alle opleidingen. Zo ontwikkelt Werken 2.0 zich langzamerhand van vak naar vaardigheid.

Tijdens het blok worden studenten gestimuleerd om in relatie met het onderwerp kritisch na te denken over de eigen toekomst. Dit heeft een onverwacht positief effect en het lijkt ons de moeite waard om de mogelijkheden van deze alternatieve vorm van studiebegeleiding nader te onderzoeken.

Ten slotte kan het geen kwaad ook eens te kijken naar de mate waarin wij als docenten de nieuwe mogelijkheden benutten. Te denken valt bijvoorbeeld aan het gebruik van Wiki-software voor het ontwikkelen van lesmateriaal, het inrichten van een omgeving op het netwerk waar alle presentaties voor alle collega's beschikbaar zijn en het gebruik van communities om gastsprekers of mogelijke opdrachtgevers te benaderen. De mogelijkheden zijn eindeloos.



Herbouw van legacy-systemen: een onderbelicht onderwerp in het Informaticaonderwijs



Herman J. Pootjes - Open Universiteit Nederland

SAMENVATTING

In de cursus Ontwikkelp practicum van de Open Universiteit Nederland voeren studenten opdrachten uit om een bestaand, klein informatiesysteem met realistische omvang dat niet meer voldoet, te verbouwen. Onder andere door reverse engineering moeten de structuur, de bedrijfsregels en de eisen van het oude systeem bepaald worden. Samen met nieuwe eisen vormen ze de basis voor het nieuwe systeem. Via dataconversie moeten de gegevens van het oude naar het nieuwe systeem omgezet worden.

TREFWOORDEN

herbouw, legacy, reverse engineering, data conversie, model driven Development.

INLEIDING

Het is voortdurend nodig om informatiesystemen aan te passen aan nieuwe eisen of te renoveren: bedrijfsregels kunnen wijzigen, er kunnen nieuwe technologieën toegepast moeten worden, wetgeving kan veranderen, enzovoort. Daarom is het onderwerp her- en verbouw van zulke informatiesystemen, die we aanduiden met de term 'legacy-systeem', van groot belang, zowel voor studenten als voor professionals. In het hbo en wo wordt aan de her- en verbouw van informatiesystemen weinig tot geen aandacht besteed: het accent bij het ontwerpen en implementeren van informatiesystemen ligt duidelijk bij nieuwbouw. Recentelijk heeft de Open Universiteit een nieuwe

cursus uitgebracht over het herbouwen van informatiesystemen onder de titel Ontwikkelp practicum. Zoals de titel al aangeeft, gaat het niet slechts om een theoretische verhandeling maar voert de student daadwerkelijk de herbouw van een informatiesysteem uit. De cursus volgt op de cursus Databases en de cursus Model Driven Development (MDD) en vormt daarmee samen een geheel.

VAN LEGACY NAAR NIEUW SYSTEEM

Het renoveren van een informatiesysteem is enigszins te vergelijken met de renovatie van een huis. Het probleem komt dan neer op de vraag: 'Hoe kan ik mijn huis renoveren terwijl ik alle voorzieningen ervan wil blijven gebruiken?'.

U ziet de problemen al voor zich: jarenlang in de rommel van een verbouwing en daarna blijkt het nog lapwerk te zijn. Vaak zou het gemakkelijker zijn, als er ruimte was, een nieuw huis naast het oude te bouwen, de inboedel over te brengen en daarna het oude huis af te breken.

Wat bij huizen meestal niet kan, is bij informatiesystemen wel mogelijk: we bouwen een nieuw informatiesysteem naast het oude, brengen de data (inboedel) over en gooien het oude systeem weg zodra we weten dat het niet meer nodig is. Bij huizen kan zich het probleem voordoen dat de oude inboedel niet goed 'past' in het nieuwe huis. Het proces van aanpassen van de inboedel heet in geval van informatiesystemen dataconversie.

Het bouwen van het nieuwe informatiesysteem komt neer op het bouwen van een nieuwe database met bijbehorende applicatie. Voor het ontwerpen van de database gebruiken we de structuur van de oude database met bijbehorende bedrijfsregels en breiden deze uit op basis van nieuwe requirements. Eventuele fouten in de structuur worden direct gecorrigeerd. In termen van een MDD-model (her)ontwerpen we een informatiemodel (grondslag voor de database), een interfacespecificatie (schermen en menu's) en logica (implementatie van bedrijfsregels en requirements).

STRUCTUURAANPASSING EN DATACONVERSIE

Het nieuwe systeem wordt gebouwd aan de hand van een reeks van eisen. Deze eisen komen niet uit

de lucht vallen; ze zijn ondermeer gebaseerd op een grondige analyse van het bestaande systeem. Een complicerende factor kan zijn dat het nieuwe systeem op een nieuw platform ontwikkeld moet worden of voor een ander databasetype. In geval van de cursus is het oude systeem (X) gebaseerd op MySQL en moet het nieuwe (Y) gebaseerd worden op Firebird. De requirements worden uitgebreid met nieuwe eisen en wensen.

Figuur 1 toont de gang van zaken in grote lijn: via een aantal stappen wordt het nieuwe systeem gebouwd (1), waarna de dataconversie (2) kan plaatsvinden. Bij het bouwen van het nieuwe systeem wordt gebruik gemaakt van de MDD-tool Cathedron. Studenten moeten op basis van de tabelstructuur van het

oude systeem een informatiemodel (PIM) opstellen (a). Door de studenten te laten werken met het bestaande informatiesysteem, moeten ze achter de eisen en de bedrijfsregels van het bestaande systeem zien te komen. Ze gebruiken deze gegevens om tot een verbeterd informatiemodel met bijbehorende bedrijfsregels te komen (b). Vervolgens passen ze het model aan en breiden het uit op basis van nieuwe eisen en bedrijfsregels. Ook moet de userinterface aangepast en uitgebreid worden (c). Daarna worden de bedrijfsregels en de eisen omgezet in interface definities en bijbehorende logica. Deze laatste stap is specifiek door het gebruik van de tool Cathedron (d). Ten slotte moeten de gegevens van het oude systeem omgezet worden naar het nieuwe. Natuurlijk zijn deze gegevens voor een deel vervuld. Studenten moeten de opschoning en de conversie uitvoeren. Hierbij kunnen ze kiezen uit SQL-scripts of een programma geschreven in Java of een andere derdegeneratietaal. Ook moeten de studenten een plan maken voor het uitvoeren van de dataconversie waarna het nieuwe systeem in gebruik genomen kan worden.

BELESTE TRANSPORT

De opdrachten van de cursus zijn gebaseerd op een fictief bedrijf, Belesté Transport. Het betreft een koeriers- en transportbedrijf dat brieven, pakketten en kleine vrachten vervoert. Het bedrijf is te snel gegroeid; het informatiesysteem is door de oprichters ooit zelf in elkaar geflanst. Vanaf 1988 wordt gebruikgemaakt van deze applicatie. Vanaf 2004 is door Botman Automatisering een nieuwe applicatie Betis gemaakt, gebaseerd op MySQL. De applicatie functioneert niet goed en Botman doet daar nauwelijks wat aan. Door problemen met Botman is er, afgezien van een summiere handleiding, geen verdere informatie over Betis beschikbaar. De database van **Betis** bevat ruim 10000 klanten en 1.5 miljoen opdrachten. Voortdurend raken gegevens zoek, het bedrijf loopt orders

mis en er is geen betrouwbare managementinformatie. De nood is hoog, een verbeterd systeem is hard nodig. Bij de cursus krijgt de student een werkende versie van Betis ter beschikking.

VOORBEELD VAN EEN PROBLEEM IN BETIS

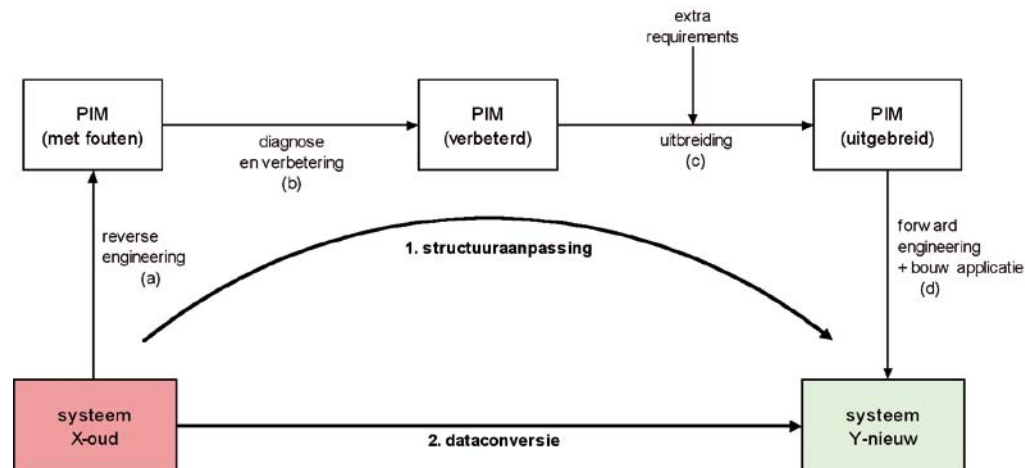
In Betis is het niet mogelijk een transport te annuleren; deze eis was niet door de opdrachtgever geformuleerd. In de loop der tijd heeft het personeel daar wat op gevonden: als datum voor het transport wordt dan 31-1-2099 gekozen. Het nieuwe systeem moet wel een mogelijkheid krijgen een transport te annuleren. De annuleringskosten bedragen 25 euro, wat verdisconteerd moet worden in de prijs voor het transport. Bij de conversie van de oude gegevens moet een student de records met de afwijkende transportdatum aanpassen: er moet een andere datum gekozen worden en het transport moet als geannuleerd aangemerkt worden en de prijs moet opnieuw berekend worden.

DE CURSUS ONTWIKKELPRACTICUM

De hoofdstukken van de cursus beschrijven de verschillende stappen van figuur 1 in detail. Waar nodig wordt extra theorie behandeld en wordt uitleg gegeven. Ieder hoofdstuk wordt afgesloten met een aantal opdrachten dat de student moet uitvoeren. Voor het uitvoeren van de opdrachten krijgt de student de beschikking over gestructureerde antwoordbladen. Per opdracht krijgt de student feedback op zijn ingeleverde werk en een beoordeling. Daarnaast ontvangt hij bij bepaalde opdrachten een Cathedron-model dat hij dient te gebruiken voor het uitvoeren van de volgende opdracht.



www.ou.nl, kijk bij cursussen van de Faculteit Informatica. Cursuscode: T51211 of T51711



Figuur 1
Structuuraanpassing en conversie

Grootschalig gebruik van versiebeheer in informaticaonderwijsprojecten

Jan Derriks - Domein Media, Creatie en Informatie, Hogeschool van Amsterdam

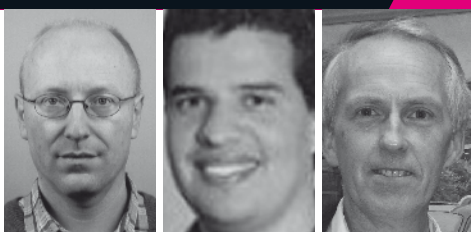
Ahmed Nait Aicha - Domein Media, Creatie en Informatie, Hogeschool van Amsterdam

Jacob Brunekreef - Domein Media, Creatie en Informatie, Hogeschool van Amsterdam

fj.derriks, a.nait.aicha, j.j.brunekreef@hva.nl

SAMENVATTING
In 2007 is bij de informaticaopleidingen van de Hogeschool van Amsterdam het gebruik van versiebeheer ingevoerd bij studentprojecten in de eerste twee studiejaar. De doelstelling was drieledig: het (leren) omgaan met een professionele werkomgeving, het bevorderen van het teamwerk in de projecten en het verhogen van het zicht van docenten op de prestaties van projectteams. We kunnen concluderen dat het eerste doel geslaagd is, het tweede doel deels geslaagd is en het laatste doel beperkt geslaagd.

TREFWOORDEN
Versiebeheer, Subversion, StatSVN, software-ontwikkeling, teamwork, teamprestaties



INLEIDING
In de IT-wereld wordt versiebeheer al langere tijd gezien als een onmisbaar instrument voor het professioneel in teamverband ontwikkelen en beheren van software. Vanaf het studiejaar 2007-2008 is aan alle teams binnen studentprojecten van de informatica-opleidingen van de Hogeschool van Amsterdam (HvA) een Subversion-repository aangeboden. De studenten werden verplicht deze versiebeheertool te gebruiken, in ieder geval bij de oplevering van de resultaten. In deze bijdrage beschrijven we onze ervaringen hiermee. We analyseren het gebruik van Subversion (svn) door teams en door individuele teamleden. De geanalyseerde data komen uit de svn-logbestanden. Voor het genereren van de svn-logbestanden hebben we gebruikgemaakt van de tool StatSVN. Met de introductie van versiebeheer hebben we drie doelstellingen voor ogen:

- 1 het leren omgaan met een professionele werkomgeving;
- 2 het ondersteunen van studenten bij (het vormgeven van) teamwork binnen onderwijsprojecten;
- 3 het verschaffen van inzicht aan docenten in het verloop van dat teamwork.

STUDENTPROJECTEN BIJ INFORMATICA
Sinds 1999 is het onderwijs bij de opleidingen Informatica en Technische Informatica van de HvA in de eerste twee studiejaar voor bijna de helft gebaseerd op grote projecten. Een project duurt een half jaar en is begroot per teamlid op circa 336 manuur (12 ECTS). Studenten werken in zelfsturende teams van circa vijf leden. Een project kent deliverables per team in de vorm van software en documenten. Een team wordt beoordeeld op deze deliverables, en op het doorlopen proces. Van ieder teamlid wordt verwacht dat hij een bijdrage levert aan de software-ontwikkeling en dat hij alle opgeleverde code kan begrijpen en toelichten.

De studentprojecten van de informaticaopleidingen kennen een aantal problemen:

- Het werk wordt vaak niet gelijk onder teamleden verdeeld. Soms kan een enkele student nagenoeg alle software hebben geschreven en zijn zijn teamgenoten niet staat om vragen over de code te beantwoorden.
- Er is vaak een te strikte scheiding in taakverdeling tussen teamleden, waarbij iedereen vooral doet waar hij al goed in is en dit niet altijd deelt met teamgenoten.
- Er wordt, ondanks een tijdig opgesteld plan van aanpak, vaak te lang gewacht met de realisatie van (deel)producten, waardoor er een piekbelasting ontstaat rond het moment van oplevering.
- De opleiding (docenten/beoordelaars) heeft onvoldoende zicht op het gevolgde software-ontwikkelproces.

Door gebruik te maken van een versiebeheersysteem onder beheer van de opleiding kan een aantal van deze problemen eerder worden gesignaleerd door zowel het team als de opleiding. Daarmee kan ook eerder aan een oplossing worden gewerkt.

VERSIEBEHEER
Sinds september 2007 wordt elke student en elk projectteam de mogelijkheid geboden om vanaf de start van het project gebruik te maken van Subversion-versiebeheer. Het onder beheer van de opleiding aanbieden van een versiebeheeromgeving biedt in onze ogen een aantal voordelen:

- 1 Veiligheid van gegevens. Een team wordt niet meer benadeeld door een teamlid dat een USB-stick verliest.
- 2 Kennisdeling wordt gefaciliteerd vanuit de opleiding en niet door bijvoorbeeld MSN.
- 3 Er wordt ervaring opgedaan met een tool die in het werkveld veel gebruikt wordt.
- 4 Versiebeheer komt de kwaliteit van de ontwikkelde software ten goede, zeker bij het werken in teamverband.
- 5 Binnen de opleidingen worden agile-ontwikkelmethoden onderwezen. Het toepassen van bijvoorbeeld test-driven development en/of daily build-and-smoke tests is nauwelijks mogelijk zonder goed ingericht versiebeheer.
- 6 Een studententeam wordt zich er meer van bewust dat het product een teamprestatie moet zijn en dat elk teamlid de code van een ander kan en mag verbeteren.
- 7 Het is mogelijk om Subversion-statistieken te verzamelen en te vergelijken voor wat betreft teams binnen één project, of over verschillende projecten. Daarmee krijgen zowel de docenten als de studenten inzage in de groei van het product en het binnen een team doorlopen ontwikkelproces.

De voordelen 1 t/m 4 hebben een algemeen inhoudelijk karakter. Ze sluiten aan bij ons eerste doel, het leren omgaan met een professionele werkomgeving. De laatste (5 t/m 7) zijn specifiek educatieve voordelen waar een opleiding gebruik van kan maken. Die dekken vooral de twee andere doelen af: studenten helpen bij het vormgeven van teamwerk en als docenten een beter zicht krijgen op dat teamwerk.

ERVARINGEN

In het studiejaar 2007-2008 zijn vanuit de opleiding 94 repositories voor teams binnen 7 projecten aangemaakt. Daarvan zijn er 12 niet gebruikt, waarvan 8 binnen één project. Hieruit concluderen we dat, uitgezonderd het ene project, er door de teams in grote meerderheid gebruikgemaakt is van de geboden faciliteit. Of er *goed* gebruik van gemaakt is, is de volgende vraag. Om een beeld te krijgen van het gebruik van Subversion door studenten is het open source tool StatSVN gebruikt. Met behulp van deze tool kunnen statistieken uit Subversion logbestanden worden gegenereerd. De statistieken hebben de vorm

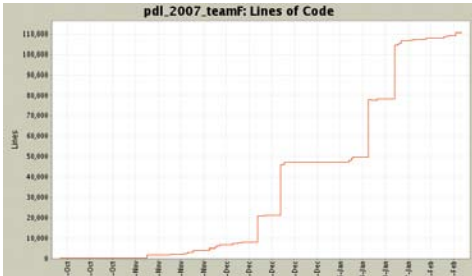


Diagram 1:
lines of code

van tabellen en grafieken. Deze statistieken kunnen gegenereerd worden per team, per project, per user en zelfs per bestand. Hieronder volgt een aantal voorbeelden van de gegenereerde statistieken.

Lines of code (LOC)

Het diagram toont het aantal ‘lines of code’ dat is geproduceerd door twee teams, gedurende de looptijd van een project. We zien dat team F (linkerdiagram) geleidelijk code heeft toegevoegd aan de repository, met enkele flinke sprongen. Team R (rechterdiagram) heeft daarentegen pas op het laatste moment (vlak voor de oplevering) de code in de repository gezet. Dit hoeft overigens niet te betekenen dat team R pas op het laatste moment is begonnen met coderen: het team kan er bewust voor gekozen hebben om pas in een zeer laat stadium gebruik te maken van Subversion. Daarmee is duidelijk dat de nodige voorzichtigheid in acht moet worden genomen bij het trekken van conclusies uit dit soort diagrammen.

Author activity

Diagram 2 geeft de verhouding weer tussen door één gebruiker nieuw toegevoegde bestanden (linkerdeel van de balk) en gewijzigde bestanden (rechterdeel van de balk). Dit diagram geeft deze verhouding weer voor vijf teamleden. We zien dat de voorlaatste gebruiker vooral wijzigingen heeft aangebracht. Dit wil niet zeggen dat hij/zij het meest heeft gewijzigd ten opzichte van andere teamgenoten; een teamlid dat slechts een keer iets wijzigt en nooit iets anders heeft gedaan zal 100% wijzigingen hebben.

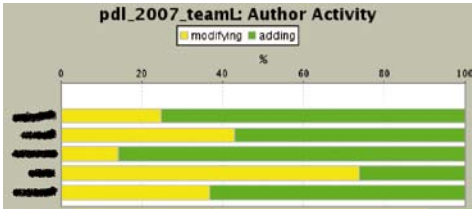


Diagram 2:
wijzigingen versus toevoegingen per teamlid

Activity by hour of day

Diagram 3 geeft de totale activiteit, uitgedrukt in het aantal commits, van het team per uur van de dag weer. We zien dat de studenten niet alleen in de geroosterde uren aan het project werken. Soms gaat het werk door tot in de kleine uurtjes.

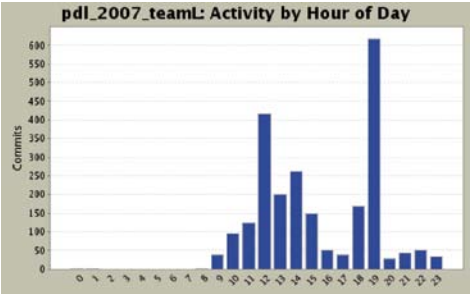


Diagram 3:
commitactiviteit per uur van de dag

Activity by day of week

In het diagram 4 zien we dat het werk aan het project niet op één dag van de week gebeurt. Sterker nog: de ingeroosterde projectdag van het betreffende project was dinsdag, terwijl de grafiek een piek op de donderdag toont.

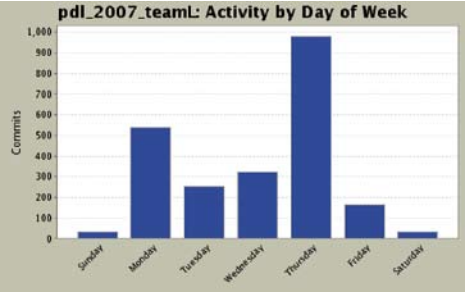


Diagram 4:
commitactiviteiten per dag van de week

ANALYSE VAN DE DATA

Om een overallbeeld te krijgen van de teamprestaties zijn deze op een aantal fronten ondergebracht in verschillende categorieën. Zo zijn de diagrammen van de groei van het aantal regels in relatie tot de tijd ondergebracht in drie categorieën: goed, matig en slecht. De onderstaande paragrafen tonen de verdeling van 82 projectteams over drie categorieën voor wat betreft verschillende aspecten van het teamwerk. Daarbij is een uitsplitsing gemaakt naar projecten uit het eerste studiejaar en projecten uit het tweede studiejaar.

Spreiding LOC over de looptijd van een project

Voor onderstaande tabel is gekeken naar de LOC-diagrammen (zie figuur 1) van alle 82 teams.

	Goed	Matig	Slecht
Teams eerste studiejaar	11	15	15
Teams tweede studiejaar	18	15	8
Totaal aantal teams	29	30	23

Wenselijk is een goede spreiding in de tijd van de werkzaamheden, zoals af te leiden uit het aantal ingecheckte regels code. De verdeling goed, matig en slecht is gebaseerd op de spreiding van activiteiten over de 21 weken van een project. De verdeling is goed als er binnen 7 weken is begonnen; slecht als er pas in de laatste 7 weken activiteit zichtbaar is en matig daartussenin.

Observaties:

- Nog niet de helft van de teams begint tijdig, dat valt niet echt mee.
- Tweedejaars doen dit beter dan eerstejaars, maar niet heel veel beter.

We concluderen dat (te) veel teams (te) laat beginnen met het opslaan van code in Subversion.

Wijzigingen ten opzichte van nieuw

Hiervoor is gebruikgemaakt van de diagrammen zoals getoond in figuur 2.

	>20%	10-20%	0-10%
Teams eerste studiejaar	21	6	14
Teams tweede studiejaar	12	12	17
Totaal aantal teams	33	18	31

Wenselijk is een significant aantal (>20%) wijzigingen t.o.v. commits van nieuwe regels code. Dat geeft aan dat een applicatie tijdens het bouwen getest en aangepast wordt.

Bij het scoren is niet alleen gekeken naar de percentage-diagrammen uit Subversion, maar ook naar de absolute aantallen activiteiten: als een teamlid 90% van alle commits voor zijn rekening neemt is het percentage in de balk van die persoon in hoge mate bepalend voor de score op dit onderdeel.

Observaties:

- Minder dan de helft van het aantal teams vertoont het gewenste gedrag, dat valt tegen.
- Een substantieel aantal blijft daar zelfs ver vandaan.
- Dit doen eerstejaars beter dan tweedejaars. Hiervoor is nog geen verklaring.

We concluderen dat (te) weinig teams Subversion gebruiken als tussenopslag voor code waaraan nog volop gewerkt wordt.

Spreiding van werkzaamheden over de week

Hiervoor is gebruikgemaakt van de diagrammen zoals getoond in figuur 4.

	Veel	Gemiddeld	Weinig
Teams eerste studiejaar	22	7	12
Teams tweede studiejaar	21	9	11
Totaal aantal teams	41	16	23

Hier is niet echt wenselijk gedrag aan te geven. De volgende verdeling is toegepast:

- Veel: minimaal drie dagen in de week met activiteit boven een denkbeeldige 30% lijn, spreiding in het uren-diagram is dan niet relevant.
- Gemiddeld: minimaal twee dagen in de week met activiteit boven een denkbeeldige 30% lijn, met duidelijke spreiding in het uren-diagram.
- Weinig of niet: overig (d.w.z. één dag in de week die eruit springt, weinig of geen spreiding in het uren-diagram.

Observaties:

- De helft van de teams werkt (redelijk) gespreid over de week.
- Er is nauwelijks verschil tussen eerstejaars- en tweedejaarsteams waarneembaar. Men zou kunnen verwachten dat tweedejaars meer spreiding vertonen omdat ze meer zelfstandiger te werk gaan en eigen tijd indelen.

Hieruit kunnen we concluderen dat veel teams min of meer zelfstandig gedrag vertonen, waarbij niet alleen gedurende de geroosterde projectdagen wordt gewerkt.

Spreiding aantal commits over teamleden

Dit onderdeel is gescoord op basis van de berekende standaard deviatie van de in Subversion getoonde percentages van het aantal commits van de afzonderlijke teamleden. Hoe kleiner de standaarddeviatie, hoe kleiner (beter) de spreiding.

- Goed: standaarddeviatie is kleiner dan of gelijk aan 15.
- Matig: de standaarddeviatie ligt tussen de 15 en 30.
- Slecht: de standaarddeviatie is groter dan 30.

	Goed	Matig	Slecht
Teams eerste studiejaar	2	9	30
Teams tweede studiejaar	6	24	11
Totaal aantal teams	8	33	41

Wenselijk is dat de spreiding over de teamleden maximaal is: iedereen doet evenveel. Dit komt overeen met de categorie ‘goed’.

Observaties:

- De helft van de teams vertoont weinig spreiding (categorie ‘slecht’): één of twee teamleden doen het werk, de rest vertoont minimale activiteiten in Subversion.
- Tweedejaars doen het gemiddeld wel duidelijk beter dan eerstejaars.
- Er zijn maar zeer weinig teams met veel spreiding van werkzaamheden, zowel in het eerste als in het tweede jaar.

Uit deze observaties kunnen we concluderen dat de spreiding van het werk (aantal commits) over teamleden erg slecht is, voor zover te zien in Subversion. Spreiding aantal regels code over teamleden Dit onderdeel is op dezelfde manier gescoord als het vorige onderdeel, met dezelfde categorieën:

	Goed	Matig	Slecht
Teams eerste studiejaar	2	9	30
Teams tweede studiejaar	3	19	19
Totaal aantal teams	5	28	49

Wenselijk is dat de spreiding over de teamleden maximaal is: iedereen doet evenveel. Dit komt overeen met de categorie ‘goed’. Voor de tweedejaars is het wenselijk dat zij het nog beter doen.

Observaties:

- De helft van de teams vertoont weinig spreiding (categorie ‘slecht’): één of twee teamleden doen het werk, de rest vertoont minimale activiteiten in Subversion.
- Tweedejaars doen het gemiddeld minder slecht dan eerstejaars.
- Er zijn maar zeer weinig teams met veel spreiding van werkzaamheden, zowel in het eerste als in het tweede jaar.

Uit deze observaties kunnen we concluderen dat de spreiding van het aantal regels code over teamleden erg slecht is, voor zover te zien in Subversion.

CONCLUSIES

Op basis van onze ervaringen met het gebruik van Subversion in studentprojecten komen we tot de volgende conclusies:

1

Subversion wordt door het overgrote deel van de teams gebruikt. Van de geboden faciliteit wordt op grote schaal gebruik gemaakt. Daarbij speelt enige dwang (de verplichting om projectresultaten op te leveren via Subversion) natuurlijk wel een rol.

2

Binnen projecten hebben teamleden het idee van ‘eigen code’ voor een flink deel losgelaten. Ze spreken meer over teamcode dan voorheen.

3

De intensiteit en professionaliteit van gebruik zijn zeer wisselend. We zien teams die een ‘ideaal profiel’ laten zien (regelmatige commits, goede spreiding over teamleden, collectief werken op vaste dagen/uren, goede indeling repository), maar we zien ook de nodige teams met het tegenovergestelde profiel. Hier ligt nog duidelijk een educatieve taak!

4

De statistieken die met behulp van StatSVN verzameld kunnen worden zijn niet zonder meer te gebruiken als objectief oordeel over het teamwerk.

5

Maar, de statistieken uit StatSVN hebben op zijn minst wel een grote anekdotische waarde: bij terugkoppeling van resultaten naar studenten is direct duidelijk te maken wat het verschil in werken tussen diverse teams lijkt te zijn. Voor studenten is dit zeer herkenbaar, en daarmee is het een goed startpunt voor discussies over het inrichten van projectwerkzaamheden.

Samenvattend mogen we stellen dat ons eerste doel -het kennismaken en leren omgaan met een professionele werkomgeving- geslaagd is. Studenten hebben voor het overgrote deel gebruik gemaakt van de geboden Subversionfaciliteiten.

Het tweede doel -het bevorderen van het teamwerk- is deels geslaagd. Een aantal studenten geeft aan zonder meer baat gehad te hebben bij het gebruik van Subversion. Aan de andere kant suggereren bepaalde statistieken slecht teamwerk.

Het derde doel -docenten inzicht verschaffen in het teamwerk- is beperkt geslaagd. Verkregen statistieken zijn niet betrouwbaar en kunnen dan ook niet zonder aanvullend onderzoek gebruikt worden bij de beoordeling van teamwerk. Aan de andere kant leveren de statistieken wel waardevol anekdotisch onderwijsmateriaal.

GERELATEERD WERK

Draheim en Pekacki ([3]) beschrijven een tool, Bloof, die op basis van gegevens uit het versiebeheersysteem CVS de evolutie van een software product analyseert. Deze tool vertoont een duidelijke gelijkenis met het ten behoeve van ons onderzoek gebruikte StatSVN. Draheim en Pekacki geven geen meetresultaten, maar geven al wel aan dat de verkregen gegevens met de nodige terughoudendheid bekeken moeten worden. Beck ([1]) motiveert het gebruik van CVS in een cursus software-engineering (SE). Doel is primair om studenten vertrouwd te maken met een professionele ontwikkelomgeving met features voor parallel ontwikkelen, gebruik van meerdere versies, goed ingericht change

management. Hij beschrijft geen ervaringen. Glassy ([4]) motiveert het gebruik van Subversion in plaats van CVS: veiligheid, robuustheid, aantal features. Hij beschrijft ook meetresultaten verkregen bij het gebruik van Subversion in SE-cursussen. StatSVN is niet gebruikt, wellicht mede hierdoor is de analyse van de meetresultaten tamelijk elementair. Burn [2] beschrijft een vergelijkbaar onderzoek als het onze: analyse van Subversion repositories in groepsprojecten op een tweetal universiteiten. Ook hij is terughoudend voor wat betreft een statistische analyse en benadrukt de anekdotische waarde van de verkregen informatie.



Frans Vodegel (links), algemeen voorzitter NIOC programmaraad met collega Jeroen Weber.



[1]

Jon Beck: Using the CVS version management system in a software engineering course. *Journal of Computing Sciences in Colleges*, vol. 20(6), pp 57-65 (2005)

[2]

Andy Burn: Analysis and Visualisation of Subversion Repositories in Group Projects. 9th Annual Conference of the Subject Centre for Information and Computer Sciences, Higher Education Academy, UK (2008)

[3]

Dirk Draheim, Lucasz Pekacki: Analytical Processing of Version Control Data: Towards a Process-Centric Viewpoint. Technical Report B-03-07, Institute of Computer Science, Free University, Berlin (2003)

[4]

Louis Glassy: Using version control to observe student software development processes. *Journal of Computing Sciences in Colleges*, vol. 21(3), pp 99-106 (2006)

16

Kids in Control as creative designers

Een onderzoek naar de bruikbaarheid en inzet van professionele simulatieontwerpssoftware als cognitieve tool in de bovenbouw van het basisonderwijs.



Nardie Fanchamp - Hogeschooldocent Lectoraat Science- en Techniekeducatie Fontys Pabo Limburg)

Nienke Valkhoff - Incontrol Simulation Solutions

TREFWOORDEN

Simulatie Ontwerp Software, Serious gaming

SAMENVATTING

'Kids in Control': een inspirerend onderzoek naar talenten en leren van leerlingen uit de bovenbouw van

het basisonderwijs. De ervaringen en opbrengsten van een pilotstudie waarin leerlingen met professionele ontwerpssoftware simulaties bouwen van fabricageprocessen. Het onderzoek belicht de bruikbaarheid en inzet van professionele simulatie ontwerpssoftware als cognitieve tool. De volgende fase van het onderzoek, uitgevoerd door het Lectoraat Science- en Techniek-educatie in het primair onderwijs, komt eveneens in beeld.



Fontys PABO Limburg

INCONTROL
Simulation Solutions

vtb

Regionaal VTB steunpunt
Wetenschap en techniek
Midden Limburg

Simulatie is een aansprekende manier om in aanraking te komen met de mogelijkheden van ICT. Tijdens het eerste Kids in Controlproject hebben we gedemonstreerd dat ICT ook het probleemoplossend denkvermogen en het hogere-ordedenken bij kinderen kan activeren. Nu onderzoeken we verbetering van de toepasbaarheid en de meerwaarde voor het onderwijs/ de ontwikkeling van leerlingen.

MOTIVATIE

Met Simulaties en Serious Gaming komen kinderen op een aansprekende manier in aanraking met de mogelijkheden en toepassingen van ICT. Tijdens de pilotstudie waren kinderen van 12 jaar met de juiste begeleiding goed in staat hun eigen ontwerp van een zelfbedachte fabriek te bouwen binnen het professionele simulatiepakket Enterprise Dynamics. Dit pakket is oorspronkelijk bedoeld voor het ontwerpen van bedrijfsprocessen en logistieke afhandeling. We willen laten zien dat ICT niet alleen als onderwerp op zich interessant is, maar ook het probleemoplossend denkvermogen en het hogere-ordedenken bij kinderen kan activeren.

Incontrol Simulation Solutions heeft al jaren ervaring met het toepassen van simulatie binnen de onderwijsprogramma's in het hbo en wo, maar wil graag verder discussiëren over welke rol simulatie ook in het basisonderwijs kan spelen en wat simulatie kan betekenen voor de ontwikkeling van kinderen. Dit in een nauwe samenwerking met het Lectoraat Science- en techniekeducatie binnen het primair onderwijs van de Fontys Pabo Limburg.

INLEIDING

Binnen het basisonderwijs in Nederland krijgen wetenschap en techniek met diverse (na)scholings- en stimuleringsregelingen (o.a. VTB-professional) steeds meer aandacht. ICT als een expliciet aandachtsgebied binnen het domein 'Wetenschap en techniek' krijgt echter nauwelijks aandacht. In de gangbare opvattingen en praktijken is ICT in het basisonderwijs eerder een hulpmiddel dan een doel van onderwijs. In de Amerikaanse Standaard voor Technische Geletterdheid (Dugger & Gilberti, 2002; ITEA, 2002) wordt ICT wel als een specifiek domein ('wereld') van technologie gedefinieerd. ICT is daar een domein waaruit, waarover en waarmee kinderen kennis, inzicht en vaardigheden dienen te ontwikkelen. Daarmee is ICT uitdrukkelijk wel onderwerp van en middel voor het techniekleren.



Incontrol Enterprise Dynamics software is professionele simulatie ontwerpsoftware die bedoeld is om gebruikers te ondersteunen bij het ontwerpen, analyseren, simplificeren, met andere woorden het modelleren en schematiseren van bedrijfsprocessen en logistieke afhandelingen. Het kan hierbij gaan om simulaties die een afspiegeling van de werkelijkheid vormen of simulaties die een nog te bouwen werkelijkheid virtueel representeren. Dit met de mogelijkheid de omgeving uitvoerig te testen alvorens deze fysiek gerealiseerd wordt. De resultaten, de computersimulaties van een veronderstelde werkelijkheid, zijn

gebaseerd op onderliggende mathematische modellen die dicteren hoe de simulatie zich gedraagt wanneer de variabelen gemanipuleerd worden (Rieber, 2005).



Recentelijk heeft Jonassen (2006) Direct Manipulation Environments (DME's) als een specifiek type van mindtools geïntroduceerd. DME's zijn computertoepassingen die het mogelijk maken om met behulp van aangeleverde of zelf te construeren bouwstenen (objecten, attributen, functies, formules en parameters) nieuwe werkende fysieke en/of virtuele computermodellen te bouwen. Kenmerkend is dat bij het uitproberen van het model direct ervaren kan worden wat de effecten van het handelen zijn. Incontrol ED-software kan tot het type DME gerekend worden. Het bouwen van simulaties met ED vraagt onderzoekend, construerend, experimenterend en redenerend leren. Met ED-software kunnen



simulaties ontwikkeld worden voor diverse toepassingen; bijvoorbeeld het oplossen van logistieke problemen in productie- of transportprocessen van vliegvelden en magazijnen, het ontwerpen van nieuwe productielijnen in een fabriek of het optimaliseren van deze processen.

In een eerdere casestudie naar de inzet en het gebruik van Incontrol Enterprise Dynamics software (Slangen & Fanchamps, 2008) is de educatieve waarde van deze specifieke software, waarmee technische computersimulaties gebouwd kunnen worden, onderzocht. Het accent is daarbij gelegd op het gebruik van ICT als:

- 1 concreet onderwerp van techniek;
- 2 hulpmiddel om techniek beter te leren begrijpen;
- 3 middel om het hogere-ordedenken in het domein techniek te stimuleren (mindtools).

In de uitgevoerde studie is gebleken dat kinderen met de leeftijd van circa 12 jaar gemakkelijk en snel met deze software leren werken. Daarbij dient men wel bepaalde werkwijzen en procedures in acht te nemen. Ook dient de leeromgeving enigszins gestructureerd voorbereid te zijn. De kinderen zijn dan met juiste begeleiding en materialen in staat om werkbare simulaties te ontwikkelen voor zelfgestelde of -ontworpen problemen, zoals de opdracht: 'Bouw een goed werkende chocoladefabriek'. De onderzoeksresultaten bieden interessante leerervaringen voor de leerlingen en tenderen naar positieve antwoorden als het gaat om de vragen: is ED-simulatiesoftware een:

- 1 hulpmiddel om techniek beter te leren begrijpen?
- 2 middel om het hogere-ordedenken in het domein techniek te stimuleren (mindtools)?

Verduidelijkingsvragen

Uit deze studie kwamen echter ook enkele constatering en aan het oppervlak die om verdere studie vragen. Specifieke onderzoekspunten daarbij zijn:

- Welke kennis dienen de docenten te hebben van het programma?
- Is het programma in de huidige vorm bruikbaar?
- In hoeverre zijn er al voorbereide opdrachten nodig?
- welke kennis dienen leerlingen te hebben en wat is de rol van de docent?
- Welk soort opdrachten leiden in welke mate tot welk soort cognitieve opbrengsten? (zoals: mathematiseren).

Tijdens het werken met deze software werden de onderzoekers ook geconfronteerd met het feit dat het materiaal niet zomaar in de klas te gebruiken is.



Enkele punten vragen daarom om nader onderzoek en ontwikkeling:

- 1** De software vraagt op punten een vertaling of vereenvoudiging waardoor leerlingen optimaal zelfstandig gebruik kunnen maken van het materiaal.
- 2** Een instructieve handleiding/werkboek met de verschillende onderdelen is nodig om leerlingen zelfstandig met het materiaal te laten werken.
- 3** Leerkrachten die leerlingen met het materiaal laten werken hebben een korte scholing nodig om leerlingen goed te kunnen begeleiden. Daarvoor zal een onderwijsleerpakket ontwikkeld moeten worden.

ALGEMENE EERSTE CONCLUSIES N.A.V. DE UITVOERING PILOT

- 1** Het blijkt mogelijk om met een aangepaste versie een instructiehandleiding en goede leerkracht begeleiding kinderen met ED te laten werken. Handleiding en programma zullen verdere verbeteringen vragen. Kennis, vaardigheden en attitude ten aanzien van leren van de leerkracht zijn erg belangrijk. De leerkrachten zullen vaardigheden moeten opdoen om kinderen te begeleiden in meer open probleemsituaties, moeten zelf processen kunnen analyseren en deze kunnen vertalen (samen met leerlingen) naar boeiende leermomenten.
- 2** Het blijkt dat leerlingen die met ED werken in een sterke interactieve leerdialoog komen. Zulke dialogen zijn vaak diepgaand en werkelijk probleemoplossend. Het antwoord is niet bekend bij de leerkracht. Dit geeft vaak spannende en zeer interessante leerervaringen.

- 3** Het is interessant om binnen een door het lectoraat aangezette vervolgstudie meer te analyseren op de cognitieve opbrengsten. Daarbij willen we o.a. ook verder onderzoeken welk soort opdrachten leidt tot welk soort cognitieve opbrengsten.
- 4** Bijkomend interessant blijkt te zijn dat schematisering en een benadering op basis van stroomdiagrammen en blokschema's interessante benaderingen zijn die vaak binnen het vo met moeite en weinig betekenisvol gerealiseerd kunnen worden.

DOEL VERVOLGSTUDIE

Het doel van de vervolgstudie is het op basis van de hiervoor beschreven onderzoeks- en ontwikkelpunten verder bruikbaar maken van het ED materiaal voor zelfstandig gebruik door leerlingen in de eigen school. Dit betekent het onderzoeken van de noodzakelijke wijzigingen, het volgens de specificaties aanpassen van het pakket, het ontwikkelen van een handleiding en werkboek met ondersteunende leermiddelen, dit materiaal testen en optimaliseren en het instrueren/ scholen van enkele (ICT-minded) leerkrachten om de haalbaarheid in de praktijk te testen. Het vervolgonderzoek wordt ingebed en uitgevoerd binnen het Lectoraat Science- en Techniekeducatie in het primair onderwijs van de Fontys Pabo Limburg; dit in een nauwe samenwerking met Incontrol Enterprise Dynamics.



[1] Dugger, W. E., & Gilberti, A. F. (2002). Standards for technological literacy: content for the study of technology. Virginia: International Technology Education Association (ITEA). ITEA. (2002). Standards for technological literacy: content for the study of technology. Virginia: International Technology education Association.

[2] Jonassen, D. H. (2006). Modeling with technology, Mindtools for conceptual change (3 ed.). Columbus, Ohio: Pearson Prentice Hall.

[3] Rieber, L. P. (2005). Multimedia Learning in games, Simulations, and Microworlds. In R. E. Mayer (Ed.), The Cambridge Handbook of Multimedia Learning (pp. 549-567). Cambridge: Cambridge University Press.

[4] Slangen, L., & Fanchamps, L. (2008). Eindrapportage van het pilotproject Kids in Control Midden-Limburg. Roermond: Fontys Pabo Limburg.

Kansen voor het e-book in het hoger onderwijs



Frans J.M. Mofers - Universitair Hoofddocent Faculteit Informatica Open Universiteit Nederland en Projectleider e-book project binnen de OU

DEELTHEMA'S

Het onderwerp heeft raakvlakken met de thema's:

- relaties tussen media & ICT (gebruik e-book als nieuw medium voor informatieoverdracht en communicatie);
- onderwijskundige aspecten (nieuwe onderwijskundige toepassingen die mogelijk worden met een e-book);
- nieuwe technologische ontwikkelingen (een nieuw soort apparaat dat op de markt komt en hoe dan ook een rol zal gaan spelen in het onderwijs).

SAMENVATTING

De komende jaren komen allerlei varianten van e-booklezers (dat wil zeggen apparaten die geoptimaliseerd zijn voor het lezen van teksten zoals kranten of boeken) op de markt. Deze apparaten gaan voor een deel traditionele boeken en andere papieren documenten vervangen. Onderwijsorganisaties zullen beleid moeten ontwikkelen over de manier waarop zij van de mogelijkheden van dit medium gebruik maken in hun onderwijsprocessen.

INLEIDING

Binnen de OU is onderzoek uitgevoerd rond het gebruik van e-books in het onderwijs. Studenten hebben met behulp van e-bookreaders cursussen bestudeerd. Ook elders in Nederland worden in het onderwijs experimenten met e-books en e-bookreaders uitgevoerd. De verwachting is dat de e-bookreader naast de pc en de telefoon tot het standaard-instrumentarium van de scholier en de student zal gaan behoren.

HET E-BOOK

De markt van e-booklezers is op dit ogenblik nog volop in ontwikkeling. In Japan (Sony Reader, 30.000 boeken beschikbaar) en de VS (Amazon Kindle 160.000, boeken beschikbaar) ligt de verkoop van e-boeken in aantallen van honderdduizenden. In Nederland is de iLiad populair, mede als gevolg van het e-paper-abonnement op NRC Handelsblad waarmee de krant via deze reader gelezen kan worden.

De omzet aan software (boeken) komt wereldwijd al op aantallen van tientallen miljoenen. Naar verwachting wordt een groot percentage van de boeken over enkele jaren in elektronische vorm uitgegeven en dit zal ook de afzet van readers een krachtige impuls geven. Ondertussen wordt er ook gewerkt aan nieuwe hardwareconcepten waarbij de elektronische inkt waarmee de huidige apparaten werken, mogelijk vervangen of aangevuld zal worden door andere technieken zoals flexibele plastic folies waarop elektronische componenten aangebracht zijn (Plastic Logic).

ONDERWIJS

Het e-book zou over een aantal jaren ook bij de (informatica)student een plaats kunnen veroveren tussen de laptop en de telefoon. Immers zelfs de almaar doorklikkende student zal naar verwachting af en toe sequentieel geordende informatie tot zich willen nemen; los van een internetverbinding en Web2.0-sociale-interactiesites. Met een e-bookreader staat een grote hoeveelheid materiaal ter beschikking op een makkelijk hanteerbaar apparaat met handzame afmetingen.

Ook de uitgevers van tekstboeken in het onderwijs spelen een belangrijke rol. Momenteel investeert men nog vooral in websites die naast een elektronische versie van het tekstboek, complete leeromgevingen bieden waarin docentmateriaal en aanvullend multimediaal leermateriaal voor studenten beschikbaar gesteld wordt. Daarnaast kunnen complete toetsbanken gebruikt en aangevuld worden voor gebruik in de eigen onderwijsinstelling. Men is echter ook gericht op zoek naar mogelijkheden om het materiaal niet allen online maar ook offline aan te bieden, bijvoorbeeld voor gebruik op e-bookreaders.

E-books blijken niet alleen traditionele boeken te kunnen vervanger maar bieden ook mogelijkheden om boeken te verrijken met indexen of audio met voor-gelezen teksten.

De onderwijskundige meerwaarde van deze vorm is echter nog beperkt, het betreft vooral de vervanging van de ene technologie (een boek op papier) door een boek in elektronische vorm.

De e-reader zou mogelijk kunnen helpen het probleem van de aanschaf en distributie studieboeken in het middelbaar onderwijs op te lossen; brugklassers zouden letterlijk ontlast kunnen worden van hun loodzware rugzakken.

Ook de beweging naar Open Educational Resources (vrije distributie van leermaterialen zoals in het Wikiwijsproject van minister Plasterk) kan profiteren van het e-book. Distributie op papier is dan immers niet meer het vanzelfsprekende alternatief voor het moeilijk lezen van materiaal vanaf het scherm van de pc.

Met e-books waarop invoer mogelijk is ontstaan ook voor specifieke onderwijsprocessen nieuwe

mogelijkheden. Het Nederlandse bedrijf iRex Technologies levert een aantal apparaten met zowel goede mogelijkheden om teksten met een pen in te voeren en daarnaast ook goede communicatiemogelijkheden. Deze technologie kan de terugkeer van de handgeschreven invoer betekenen. Op die manier ingevoerde tekst kan als afbeelding, of omgezet naar ASCII-code, vervolgens uitgewisseld worden met medestudenten of met docenten. Vooral bij opleidingen zoals informatica, wiskunde en scheikunde, waar schematische afbeeldingen en formules een belangrijke rol spelen kan een dergelijk invoerapparaat voordelen bieden. Ook annotaties op bestaande documenten kunnen uitgewisseld worden, waardoor nieuwe manieren ontstaan om informatie te delen binnen groepen van studenten en docenten.

EXPERIMENTEN

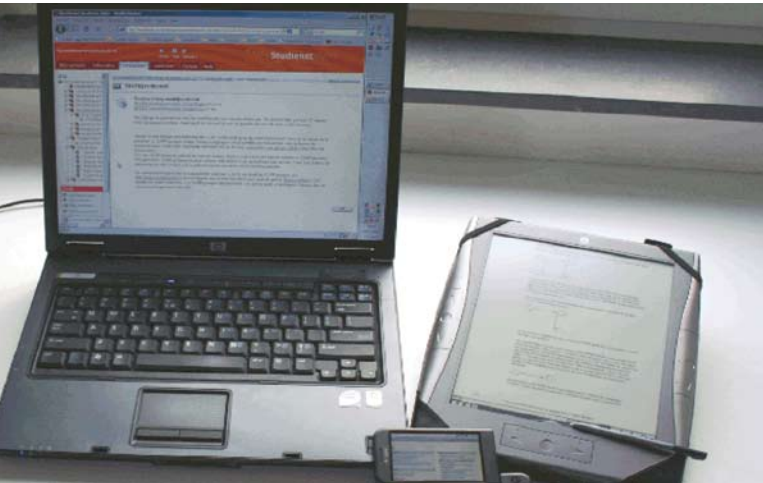
Binnen de Open Universiteit Nederland is een aantal experimenten uitgevoerd waarbij complete (papieren) cursusboeken omgezet zijn naar een digitale versie (PDF). Studenten hebben een aantal vakken op deze manier bestudeerd met een e-bookreader die nog geen invoermogelijkheden bezat. Zij waren te spreken over het gebruiksgemak. Omdat het een beperkt aantal cursussen betrof vond men het omschakelen tussen papieren materiaal en het elektronisch materiaal omslachtig en zou men liever alle teksten in één variant ter beschikking willen hebben.

Ook is er een duidelijke impact op het productieproces van teksten bij een instelling als de OU die ervaring heeft met het grootschalig ontwikkelen van zelfsturend onderwijsmateriaal. Apparaten die documenten op A4-formaat prettig leesbaar weer kunnen geven komen nu langzaam op de markt en kunnen extra conversiewerk om papieren documenten voor een e-book beschikbaar te stellen helpen reduceren.

Ook zijn experimenten doorgevoerd waarbij docenten handgeschreven opmerkingen plaatsden op PDF's of Worddocumenten van studenten. Dit handgeschreven

commentaar kan vervolgens, al dan niet omgezet naar ASCII-code, in elektronische vorm worden teruggestuurd naar studenten.

De belangrijkste belemmering voor een grootscheeps gebruik op korte termijn lijkt de prijsstelling van deze apparaten. Een model met alle gewenste functies heeft nog steeds een prijskaartje van boven de € 500. Er is op dit ogenblik nog moeilijk een businesscase te maken waarbij elke leerling of student naast zijn mobiele telefoon en zijn pc ook over dit apparaat zou moeten beschikken om te kunnen studeren.



Rechts: een e-book

- [1] Pilot E-readers in het onderwijs SURFnet/Kennisnet innovatieprogramma [http://www.surfnetkennisnetproject.nl/attachments/1899748/Pilot_E-reader_in_het_onderwijs.pdf]
- [2] onderzoek e-book project bij Open Universiteit Nederland [http://dspace.ou.nl/handle/1820/1935]
- [3] gratis toegankelijke OU-cursus Logica in e-bookformaat [http://www.spinoza.ou.nl/johan-van-benthem/logica-in-actie]



Hans Frederik, voorzitter van
Stichting NIOOC

Onderwijssamenwerking in Business
Process Management

Johan Versendaal - Hogeschool Utrecht,
Universiteit Utrecht

Pascal Ravesteyn - Hogeschool Utrecht,
Universiteit Utrecht

{ johan.versendaal, pascal.ravesteijn}@hu.nl
{ j.versendaal, pascalr }@cs.uu.nl
Hogeschool Utrecht, Faculteit Natuur en Techniek,
Nijenoord 1, 3552 AS, Utrecht, NL
Universiteit Utrecht, Bètafaculteit, Padualaan 14,
3584 CH, Utrecht, NL

SAMENVATTING

In een unieke samenwerking tussen Hogeschool Utrecht, Universiteit Utrecht en bedrijfsleven hebben studenten van de HU (informaticaopleiding) en studenten van de UU (masteropleiding Informatiekunde) in projectvorm samengewerkt in een BPM-cursus. In het theoriedeel werden onder andere succes- en situationele factoren van BPM-implementaties uit de wetenschappelijke literatuur geïdentificeerd en gebruikt om er implementatiemethoden mee te definiëren: de UU-studenten namen hier het voortouw en betrokken HU-studenten in hun onderzoek. In het praktische deel werden, met beide benen op de grond, interorganisationale bedrijfsprocessen met behulp van BPM-systemen gemodelleerd, geconfigureerd en geautomatiseerd. De samenwerking kan succesvol worden genoemd omdat de studentenevaluaties positief kritisch waren, en omdat de samenwerking in wetenschappelijke publicaties heeft geresulteerd.

TREFWOORDEN

BPM, Samenwerking, HU, UU, Bachelor, Master

INLEIDING

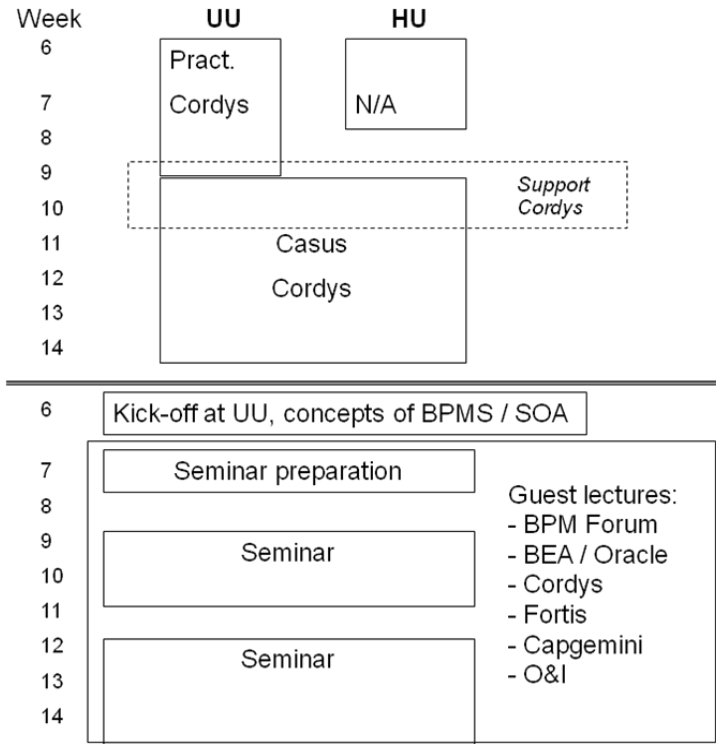
Zowel in de private als publieke sector is samenwerking in de keten van groot belang. Ondanks dit belang is de traditionele IT ter ondersteuning van ketensamenwerking zowel in aanschaf als qua maatwerk kostbaar. Daarnaast leveren implementaties niet wat men er van te voren van verwacht (Ravesteyn & Versendaal, 2009). Echter, Service-Oriented Architectures (SOA's) en Business Process Management (BPM-) suites dragen bij aan het modelleren van publieke en private processen, en maken het mogelijk om heterogene applicaties gemakkelijker te koppelen (Weske, 2007; Hiemstra et al, 2009).

Als docenten van Hogeschool Utrecht en Universiteit Utrecht vinden we het belangrijk dat studenten worden opgeleid in en bekend worden met SOA's en BPM-suites. En dit niet alleen in theorie maar ook door het daadwerkelijk modelleren, configureren en implementeren van een BPM-suite in een nagebootste werkelijkheid. Daartoe is er een cursus ontwikkeld voor masterstudenten van de universiteit en bachelorstudenten van de hogeschool, waarin de theorie en praktijk van BPM worden behandeld. Deze cursus draait in de huidige vorm vanaf voorjaar 2006 en wordt jaarlijks gegeven: de cursus is op moment van schrijven vier keer gegeven.

De cursus heet BPM, en beslaat 6 of 7,5 ECTS, afhankelijk of je student van de hogeschool of

universiteit bent. Het theoriedeel (een seminar met een wetenschappelijk karakter) wordt op de universiteit gegeven, en het praktijkdeel (het werken met en configureren van de Cordys BPM omgeving) op de hogeschool. Hogeschoolstudenten en universitaire studenten zijn in groepjes aan elkaar gekoppeld,

waarbij de universitaire student de verantwoordelijkheid heeft in het theoriedeel. In de achtereenvolgende jaren dat de cursus is gegeven is telkens een specifiek wetenschappelijke vraagstelling in het theoriedeel aangepakt en empirisch uitgewerkt. In figuur 1 wordt een schematische weergave van de cursus gegeven.



Figuur 1:
Schematische weergave van de BPM-cursus

Verticaal staan de weeknummers: merk op dat het praktische en theoretische deel op de Y-as gescheiden zijn in het plaatje, maar in werkelijkheid parallel plaatshebben. Horizontaal is aangegeven hoe de universitaire studenten en hogeschoolstudenten het vak doorlopen. Ook is aangegeven welke organisaties een bijdrage in de vorm van gastcolleges hebben gegeven: we vinden het als docenten belangrijk dat de studenten voortdurend voeling houden met de praktijk, of het nu om adviesbureaus, softwareontwikkelaars, BPM-suitegebruikers of (BPM-)verenigingen gaat.

In de volgende paragrafen behandelen we achtereenvolgens het praktische deel en het theoretische deel van de BPM-cursus. Daarna geven we conclusies, en identificeren we mogelijkheden voor de cursus in volgende jaren.

PRAKTISCH DEEL VAN DE BPM-CURSUS

De beoogde competenties in het praktische deel zijn gerubriceerd naar een drietal categorieën:

- 1
- Business Administration: de student is in staat om de relaties tussen de verschillende businessfuncties en -afdelingen te herkennen binnen en tussen organisaties.
- 2
- Information Architecture and Technology: de student begrijpt de basis van SOA's en BPM-suites en kan ermee werken. De student is in staat om de BPM-suite te configureren, en legacy-systemen in een keten te integreren. De student is in staat om een systeemarchitectuur voor interorganisationale samenwerking te definiëren.
- 3
- Influence and Alignment: de student is in staat om zowel op bedrijfskundig niveau als op ICT-niveau oplossingen te bedenken en te implementeren. De student is in staat in onderling overleg, en

met onderhandeling met andere studenten tot oplossingen te komen. Met behulp van de Cordys BPM-suite worden drie organisaties verbonden: 'retailer', 'wholesaler' en 'manufacturer'. Bij elk van deze organisaties is van te voren al een ERP-systeem aanwezig. De gemixte studentgroepen worden gekoppeld aan een van deze organisaties. Tussen de organisaties (=studentgroepen) wordt vervolgens onderhandeld over input- en outputformaten, protocollen, integraties et cetera, die vervolgens in Cordys worden geïmplementeerd.

Concreet ontwerpen en implementeren de studenten met elkaar de procesmodellen, soapnodes (integratie-interfaces), webservices en keyperformance-indicators welke gebruikt worden in management-dashboards. Ze zijn daarbij afhankelijk van elkaar, zo moet bijvoorbeeld de retailer afspraken maken met de wholesaler, die op zijn beurt keuzes maakt met de manufacturer.

- Bij elke studentevaluatie van de cursus komen de volgende zaken rondom het praktische deel terug:
- BPM hands-on wordt als complex en tijdrovend ervaren.
 - Binnen een team is samenwerken tussen studenten van hogeschool en universiteit complex, vanwege de verschillende competenties van studenten bachelor en master.
 - Inhoudelijk samenwerken tussen de verschillende studententeams (die een organisatie vertegenwoordigen) is complex: je moet echt overeenstemming hebben over processen en formaten van berichten.

THEORETISCH DEEL VAN DE BPM-CURSUS

In de vier jaar dat de cursus wordt gegeven hebben we telkens een ander deelthema behandeld.

Achtereenvolgens waren dat:

- 1
- Business Process Management raamwerk
- 2
- Validatie van succesfactoren van BPM-suite implementaties
- 3
- Identificatie van situationele factoren en gerelateerde implementatie methodefragmenten
- 4
- Human Interaction Management en BPM. De laatste drie thema's worden hieronder kort toegelicht.

Succesfactoren van BPM-implementatie

In 2007 kregen groepjes van twee studenten (een hogeschoolstudent en een universitaire student) de opdracht om in het veld (bij organisaties waar bijvoorbeeld BPM-implementaties actueel waren) van te voren uit wetenschappelijke literatuur geïdentificeerde succesfactoren (Ravesteyn, 2007) te toetsen en te valideren. Daarbij werden drie methoden van valideren gebruikt:

- Full factorial: de lijst van succesfactoren werd aan de respondent voorgelegd en daarbij werd de vraag gesteld of de succesfactoren correct en compleet waren.
- Scale construction: de respondent werd per succesfactor een set van afgeleide vragen voorgelegd. De antwoorden op de vragen zouden daarbij terug te voeren zijn naar de correctheid van de succesfactoren.
- Open interviews: de respondent werd niet geconfronteerd met een lijst van succesfactoren: in plaats daarvan werd in een open interview de vraag gesteld welke succesfactoren een rol speelden bij BPM implementatie.

Op deze manier kwamen de studenten in aanraking met een academische vaardigheid: het valideren van theorie. De ervaringen van de studenten waren in het algemeen positief: ze vonden het leuk om bij te dragen aan wetenschappelijk onderzoek. De resultaten van de validatie zijn na de cursus in een wetenschappelijk paper opgenomen (zie figuur 2).

Figuur 2:
paper gevalideerde succesfactoren van BPM implementatie (Ravesteyn & Versendaal, 2007)

Ravesteyn et al

A Situational Implementation Method for BPMS

A Situational Implementation Method for Business Process Management Systems

Pascal Ravesteyn
HU University of Applied Sciences Utrecht,
Nijenoord 1, 3552 AS, Utrecht, The Netherlands
pascal.ravesteijn@hu.nl

Slinger Jansen
Utrecht University, Padualaan 14, 3584 CH ,
Utrecht, The Netherlands
slinger@cs.uu.nl

ABSTRACT
For the integrated implementation of Business Process Management and supporting information systems many methods are available. Most of these methods, however, apply a one-size fits all approach and do not take into account the specific situation of the organization in which an information system is to be implemented. These situational factors, however, strongly determine the success of any implementation project. In this paper a method is provided that establishes situational factors of and their influence on implementation methods. The provided method enables a more successful implementation project, because the project team can create a more suitable implementation method for business process management system implementation projects.

SITUATIONELE FACTOREN EN METHODE-
FRAGMENTEN VAN BPM-IMPLEMENTATIE

In 2008 werd aan groepjes studenten gevraagd om, gegeven de geïdentificeerde en gevalideerde succesfactoren (Ravesteyn & Versendaal, 2007), situationele factoren te vinden en daarbij methodefragmenten voor BPM-suite implementatie te definiëren. Gebruikmakend van in de literatuur bekende IS-methodieken

werden door de groepen studenten per succesfactor verschillende situationele factoren en methodefragmenten gedefinieerd. De resultaten van de cursus uit 2008 leidden opnieuw tot een wetenschappelijke publicatie (zie figuur 3), waarmee we opnieuw hebben aangetoond op een wetenschappelijke manier studenten mee te nemen in (toegepast) onderzoek.

Ravesteyn et al.

A Situational Implementation Method for BPMS

A Situational Implementation Method for Business
Process Management Systems

Pascal Ravesteyn
HU University of Applied Sciences Utrecht,
Nijenoord 1, 3552 AS, Utrecht, The Netherlands
pascal.ravesteijn@hu.nl

Slinger Jansen
Utrecht University, Padualaan 14, 3584 CH
Utrecht, The Netherlands
slinger@cs.uu.nl

ABSTRACT

For the integrated implementation of Business Process Management and supporting information systems many methods are available. Most of these methods, however, apply a one-size fits all approach and do not take into account the specific situation of the organization in which an information system is to be implemented. These situational factors, however, strongly determine the success of any implementation project. In this paper a method is provided that establishes situational factors of and their influence on implementation methods. The provided method enables a more successful implementation project, because the project team can create a more suitable implementation method for business process management system implementation projects.

Figuur 3:
paper situationele factoren en methodefragmenten van BPM
implementatie (Ravesteyn & Jansen, 2009)

Human Interaction management en BPM

In het voorjaar van 2009 hebben we als docenten niet een specifieke onderzoeksvraag aan de studenten voorgelegd. In een aantal introductiecolleges werd de BPM-trend rondom Human Interaction Management (HIM) toegelicht. Elk studentengroepje werd nu gevraagd om binnen het kader van HIM een eigen unieke onderzoeksvraag te definiëren. Vervolgens moesten de studenten een korte, maar volledige,

onderzoekscyclus doorlopen, inclusief theorievorming, validatie en disseminatie. Ieder groepje heeft aan het eind van de cursus zijn ‘paper’ gepresenteerd. Twee ‘papers’ (van twee studentgroepen) waren van dusdanig niveau dat we ze in aangepaste vorm hebben ingestuurd naar een wetenschappelijk congres. Een van de papers is geaccepteerd op het IIMA (www.iima.org) congres in de Verenigde Staten. Figuur 5 laat een snapshot van het paper zien.

Applying Human Interaction Management Concepts to E-
Mailing: A Visualized Conceptual Model

Baldur Kristjansson; Patrick Mikalef; Johan Versendaal
Padualaan 14,
3584 CH Utrecht, Utrecht University, Netherlands
T +31 30 253 1454
F +31 30 251 3791
bkristja@cs.uu.nl

Pascal Ravesteyn
Nijenoord 1
3552 AS Utrecht, Hogeschool Utrecht University of Applied Sciences and Utrecht
University, Netherlands
T +31 30 238 8819
F +31 30 238 8448
pascal.ravesteijn@hu.nl; pascalr@cs.uu.nl

ABSTRACT

Electronic mail (e-mail) is one of the dominant IT applications used by knowledge workers and managers. Its key functionality remains the same despite being used for much more than simple messaging: document sharing and archiving for example. At the same time, e-mail is one of the main sources of information overload, which threatens the efficiency, effectiveness and health of knowledge workers. In this paper we address this problem by taking a human-driven and collaborative perspective going beyond traditional Business Process Management: we apply Human Interaction Management (HIM) concepts in constructing a conceptual model for the processing of e-mails. We subsequently use the model to build an exploratory prototype of an add-in for a popular e-mail client hence making the model more tangible and understandable. The prototype shows the implementability of the model and can serve to gain more feedback on its validity, and inspire new ideas about its other uses, such as linking e-mail to collaborative workspaces.

Figuur 4:
paper Human Interaction Management in (inter-)
organisatiele communicatie (Kristjansson et al., 2009)

CONCLUSIES EN VOORUITBLIK

In vier achtereenvolgende jaren wordt nu de cursus BPM gegeven. In deze vier jaar zijn we in staat geweest om studenten op een hele praktische manier met de voeten in de klei bezig te laten zijn met BPM, terwijl tegelijkertijd een theoretische en academische basis is geven op het gebied van BPM. Opvallend is dat er op het eerste jaar na elk jaar één of meerdere wetenschappelijke publicaties zijn voortgekomen uit het werk van de studenten.

De BPM-module zoals deze door Hogeschool Utrecht en Universiteit Utrecht is ontwikkeld biedt een brede basis in zowel theoretische als praktische BPM-kennis en -vaardigheden. We stellen verder dat we goede ervaringen hebben met het verbinden van onderwijs met onderzoek: het een versterkt het ander, een echte win-wincombinatie. Het jaarlijks bedenken van een nieuw theoretisch thema is een succesfactor van de aantrekkelijkheid van de cursus, zowel voor studenten als docenten. Studenten kunnen in de state of the art van BPM worden onderwezen, terwijl er voor docenten ook een voortdurende uitdaging is, om weer met 'iets nieuws te komen'. Tot slot heeft de module tot op heden vijf publicaties opgeleverd en daarmee ook een duidelijke bijdrage geleverd aan het wetenschappelijke domein.

Voor 2010 staat de cursus weer op het programma bij beide instituten. Op dit moment zijn wij als docenten bezig om een nieuw onderzoeksthema te bedenken. Hierbij zoeken we ook steeds meer samenwerking met het bedrijfsleven, zo is het HIM-thema van afgelopen jaar bijvoorbeeld voortgekomen uit een samenwerking met Fortis Nederland en Capgemini, een samenwerking die na afloop van de module is voortgezet. Ook in het praktische deel van de module staan de ontwikkelingen niet stil. De afgelopen jaren is met wisselend succes gebruik gemaakt van de Cordys software. Vooral het beheer van het systeem is complex en tijdrovend en daarom zijn we nu de mogelijkheid aan het bestuderen om de cloud-oplossing van Cordys te gebruiken (www.theprocessfactory.com). Niet alleen kan hiermee het beheer worden versimpeld maar de oplossing die geboden wordt door Cordys in samenwerking met Google biedt ook veel nieuwe mogelijkheden rondom het ontwikkelen van MashApps.

Kortom, de module Business Process Management van de Hogeschool en Universiteit Utrecht heeft vier succesvolle jaren achter de rug en wij kijken met veel plezier uit naar de komende jaren.



[1] Hiemstra, A., Ravesteyn, J.P.P. & Versendaal, J.M. (2009). An Alignment Model for Business Process Management and Service Oriented Architecture. In Proceedings of the 6th International Conference on Enterprise Systems, Accounting, and Logistics (ICESAL 2009). Thessaloniki, Greece.

[2] Kristjansson, B., Mikalef, P., Versendaal, J.M. & Ravesteyn, J.P.P. (2009). Applying Human Interaction Management Concepts to E-Mailing: A Visualized Conceptual Model. To be published in The Communications of the IIMA. International Information Management Association (IIMA 2009). Houston, TX, United States.

[3] Ravesteyn, P. 2007. A Study into the Critical Success Factors when Implementing Business Process Management Systems. Published in the Conference Proceedings of the International Resources Management Association, Vancouver, Canada.

[4] Ravesteyn, P. and Jansen, S. (2009). 'A Situational Implementation Method for Business Process Management Systems', Published in Proceedings of the American Conference on Information Systems 2009, San Francisco, United States.

[5] Ravesteyn, P. and Versendaal, J. (2007). 'Success Factors of Business Process Management Systems Implementation', In Proceedings of the Australasian Conference on Information System (Tolman, M. Ed.), p. 396-406. Toowoomba, Australia, University of Southern Queensland.

[6] Ravesteyn, P. and Versendaal, J., 'Constructing a Situation Sensitive Methodology for Business Process Management Systems Implementation' (2009). PACIS 2009 Proceedings. Paper 70.

[7] Weske, M. (2007). 'Business Process Management: Concepts, Languages, Architectures'. Springer-Verlag, New York.

Het Soccer-Funproject: functioneel programmeren in het onderwijs met behulp van voetbal



Peter Achten - Model Based System Development, Informatica, Radboud Universiteit Nijmegen
P.Achten@cs.ru.nl

SAMENVATTING

In het Soccer-Funproject leren scholieren en studenten op een stimulerende manier functionele programmeertaalconcepten. Soccer-Fun is geïnspireerd op een uitspraak van Johan Cruijff, waarin hij stelt dat het brein van een voetballer een zuivere functie is. In Soccer-Fun maken scholieren en studenten een breinfunctie en proberen daarmee andere teams te verslaan.

TREFWOORDEN

Functioneel programmeren, strategieën, simulatie, kunstmatige intelligentie

INLEIDING

Het Soccer-Funproject is ontwikkeld aan de Radboud Universiteit Nijmegen om informaticastudenten op een stimulerende manier te laten oefenen met functioneel programmeren en redeneren over software. Hierbij gaat het om software die bestaat uit verschillende componenten die tegelijkertijd beslissingen nemen. Soccer-Fun is geïnspireerd op de volgende uitspraak van Johan Cruijff:

'Als ik een bal aan de voet heb die ik wil afspelen, dan moet ik rekening houden met mijn bewaker, de wind, het gras, de snelheid waarmee de spelers lopen. Wij berekenen de kracht waarmee je moet schoppen en de richting waarin in ééntiende seconde. De computer doet daar twee minuten over!' (De Tijd, 2 mei 1987)

Hiermee zegt Johan Cruijff in feite dat zijn brein een functie is die een aantal parameters krijgt (bewaker, wind, gras, loopsnelheden) en die twee waarden berekent (kracht en richting). Dit karakteriseert met een het doel van de opdrachten waarmee studenten zich bezighouden met Soccer-Fun: maak je eigen breinfunctie, stop die in een elftal van voetballers, en kijk of je andere teams kunt verslaan.

SOCCER-FUN

In het bachelorinformaticacurriculum aan de Radboud Universiteit Nijmegen worden de *imperatieve, objectgeoriënteerde en functionele* programmeerparadigma's onderwezen. Het functionele programmeerparadigma stimuleert het modelleren van probleemdomen in termen van data types, abstractie en compositie in programmeren met behulp van (*hogere-orde*)functies, en het bewerken van gegevensverzamelingen met *lijsten*. Functionele talen zijn uitermate geschikt voor abstractie en dat komt sterk tot uiting in de bijbehorende practicumopdrachten. Dit geldt ook voor het door ons ontwikkelde practicum voor de tweedejaarscursus *'Abstractie en Compositie in Programmeren'* (2005-2008)⁴. De abstracte aard van de practicumopdrachten wordt echter niet door alle studenten gewaardeerd. Om een (nieuw) programmeerparadigma onder de knie te krijgen is actieve deelname aan het practicum een bijna noodzakelijke voorwaarde. Het is dus belangrijk om een practicum aan te bieden dat studenten

⁴ Inmiddels is deze cursus opgevolgd door "Functioneel Programmeren" voor de studies informatica en kunstmatige intelligentie.

stimuleert hun taalvaardigheid en probleemoplossend vermogen te vergroten. Bovendien willen wij als docenten graag de *fun* van functioneel programmeren overbrengen op onze studenten. Het Soccer-Fun-raamwerk is ontwikkeld met deze twee doelen voor ogen. Voor de volledigheid willen we benadrukken dat het Soccer-Funraamwerk een *aanvullende* rol heeft in het pakket practicumopdrachten dat we studenten aanbieden.

Soccer-Fun is geïnspireerd op de hierboven geciteerde uitspraak van Johan Cruijff waarin hij stelt dat het brein van een voetballer een functie is. In het citaat wordt alleen de situatie beschreven waarin een voetballer in balbezit is en de bal wil spelen. In een echte wedstrijd treden meer situaties op zoals in balbezit komen, (strategische) posities innemen, verdedigen en aanvallen. Merk op dat dit niet alleen in voetbal voorkomt, maar in iedere teambalsport zoals hockey, basketbal en handbal. Dit draagt ertoe bij dat

het ontwikkelen van breinfuncties ook interessant is voor studenten die geen affiniteit hebben met voetbal. Uiteraard moeten studenten rekening houden met specifieke voetbalsituaties zoals de buitenspelregel. Het bedenken en implementeren van succesvolle strategieën is uitdagend voor de informatica- en kunstmatige intelligentiestudenten. Studenten kunnen de kwaliteit van hun implementatie toetsen door de door hen ontwikkelde teams tegen elkaar te laten spelen, zoals getoond wordt in figuur 1. *Soccer-Fun* voegt op deze manier een *competitie-element* toe aan het practicum, wat als stimulerend ervaren wordt. Dit is niet vrijblijvend voor de studenten: de cursus wordt afgesloten met een toernooi waarin alle teams die voldoen aan vooropgestelde criteria tegen elkaar uitkomen. We loven twee prijzen uit: één voor de toernooiwinnaar en één voor de code die in onze ogen het best gebruik maakt van functionele programmeertaalconcepten.



Figuur 1:
Het Soccer-Funraamwerk in actie.

Soccer-Fun is geïmplementeerd in de in Nijmegen ontwikkelde zuivere functionele programmeertaal Clean. Deze is gratis beschikbaar en kan gedownload worden via <http://clean.cs.ru.nl/>. Het *Soccer-Funproject* zelf is een open project, d.w.z. dat iedereen, na aanmelding, bijdragen kan leveren in de vorm van code, oefeningen, voorbeeldteams, documentatie, en dergelijke. De *Soccer-Funsite* is: <http://www.cs.ru.nl/P.Achten/SoccerFun/SoccerFun.html> Om met *Soccer-Fun* te kunnen werken moeten studenten kennisgemaakt hebben met Clean taalelementen die ook in andere programmeertalen te vinden zijn zoals basistypes en records. Kenmerkende functionele taalelementen die ook nodig zijn, zijn algebräische types, lijsten en lijstcomprehensions. In eerste instantie kunnen taalconcepten als hogere-ordefuncties en recursief programmeren ‘vermeden’ worden. Deze komen aan bod naarmate de brein-functies uitdagender worden. Op deze wijze kan functioneel programmeren stapsgewijs worden aangeboden en geïntegreerd in een bestaand of een te ontwikkelen programmeerpracticum.

Het voetbalbrein

In zijn eenvoudigste vorm is het brein van de voetballer een functie (*FootballerAI*) van brein invoer (*BrainInput*) naar brein uitvoer (*BrainOutput*):
`:: FootballerAI' ::= BrainInput → BrainOutput`
BrainInput is een record waarin de breinfunctie de beslissingen van de scheidsrechter aantreft, de toestand van de voetbal (positie en snelheid), alle overige spelers in het veld en de speler zelf. Het doel van de breinfunctie is een passende actie te berekenen van type *BrainOutput*. De belangrijkste acties zijn het bewegen van de speler, het spelen van de bal met been of hoofd en het in bezit nemen van de bal. Het Soccer-Funraamwerk interpreteert de gewenste acties en zet deze om in realistische acties. In het Soccer-Funraamwerk kan de student er voor kiezen of er middels randomwaarden afwijkingen

toegevoegd worden aan deze acties.

Zodra het voetbalbrein een echte wedstrijd moet spelen, is het van belang dat deze naar de scheidsrechter luistert. Deze geeft namelijk aan wanneer de eerste helft van de wedstrijd voorbij is, en er dus een helftwissel plaatsvindt. Om dit te realiseren heeft het brein een *geheugen* nodig. Dit is de normale vorm van de breinfunctie (*FootballerAI*):
`:: FootballerAI memory ::= (BrainInput, memory) → (BrainOutput, memory)`
Dit is een voorbeeld van een geparametriseerd functie type, waarin *memory* het type geheugen is van de voetballer. De student ontwerpt een geschikt geheugen (meestal een record). Dit geheugen wordt als argument meegegeven bij aanroep van de breinfunctie, die naast een voetballeractie van type *BrainOutput* ook een nieuwe waarde van type *memory* oplevert.

Breinschets

In het informaticaprogrammeeronderwijs vinden we het belangrijk dat studenten een ontwerp maken voordat de codeerfase begint. In *Soccer-Fun* kun je al op een informele, maar toch effectieve, manier nadenken over je breinfunctie. We noteren dat met *breinschets*. Hier is een voorbeeld dat het spelgedrag van mini-F-voetballertjes⁵ beschrijft:

mini-effie voetbalveld doel invoer

if ik kan tegen de bal schoppen
==> schop bal richting doel tegenstander
otherwise ==> ren en draai richting bal

⁵ Kinderen die graag willen voetballen, maar nog geen 7 jaar zijn kunnen als ‘mini-F’ lid worden van een voetbalvereniging en meedoen aan competitie.

Een breinschets heeft een *naam* (*mini-effie*) en als argument ten minste de *BrainInput* (*invoer*). De uitvoer is ten minste een *BrainOutput*-waarde (*schop*, *ren*, *draai*). Met behulp van beslissingsregels wordt de werking van het brein vastgelegd. Op deze manier wordt het voor de studenten snel duidelijk of ze voldoende informatie kunnen vinden in de *BrainInput*, of dat ze extra informatie nodig hebben. In dit geval zijn dat de afmetingen van het voetbalveld en de positie van het te verdedigen doel. Deze twee argumenten krijgt een voetballer mee aan het begin van een wedstrijd. Alle benodigde extra informatie wordt als argument meegegeven aan de breinschets vóór het verplichte invoerargument⁶. Een breinfunctie die geen geheugen nodig heeft hoeft dit ook niet in zijn breinschets op te nemen, en kan als een *FootballerAI*’ functie gerealiseerd worden. Een breinfunctie die wel een geheugen nodig heeft, neemt dit bij zijn invoer argument op en levert dit ook op als resultaat. Een dergelijk brein is uiteraard van type (*FootballerAI memory*).

Compositioneel programmeren is een belangrijke functionele programmeerstijl die ook toegepast kan worden in *Soccer-Fun*. Hierin worden breinfuncties samengesteld middels eenvoudigere breinfuncties. Als voorbeeld geven we de breinschetsen van drie elementaire breinfuncties: *halt*, *fix* en *kick*.

halt invoer

==> beweeg niet en draai niet

fix positie d invoer

if de afstand tussen mij en positie is minder dan d meter
==> halt invoer
otherwise ==> ren en draai richting positie

⁶ De technische reden hiervoor is dat in functionele talen currying het mogelijk maakt om extra informatie toe te voegen aan een functie vóór de ‘normale’ argumenten.

kick positie invoer

if ik kan tegen de bal schoppen
==> schop bal richting positie
otherwise ==> halt invoer

Het *halt*-brein besluit onder alle omstandigheden stil te staan en niet te roteren. Het (*fix p d*)-brein beweegt een speler naar positie *p* op het voetbalveld en staat stil zodra de speler positie *p* op afstand *d* genaderd is. Het (*kick p*)-brein laat de speler de bal naar positie *p* schoppen indien de bal binnen trapbereik ligt. Met deze elementaire breinfuncties kan het *mini-effie*-brein eenvoudig gerealiseerd worden:

mini-effie voetbalveld doel invoer

if ik kan tegen de bal schoppen
==> kick (centrum doel tegenstander) invoer
otherwise ==> fix balpositie nul invoer

Het gebruik van breinschetsen is een nuttig hulpmiddel voor docent en student om ideeën over mogelijke breinfuncties te communiceren, bekritisieren, en verder te ontwikkelen. Door het kiezen van de mate van detail kan een concrete programmeeropdracht gegeven worden en wordt tegelijkertijd veel keuzevrijheid geboden om deze te realiseren.

Scheidsrechterbeslissingen

In *Soccer-Fun* is een scheidsrechter geïmplementeerd die de concrete speelacties van alle spelers beoordeelt, de spelregels hanteert, en op grond hiervan een reeks *scheidsrechterbeslissingen* genereert. Deze beslissingen worden aan alle spelers bekend gemaakt middels hun *BrainInput*. Net als het brein van de voetballers kan het brein van een scheidsrechter met een functie gemodelleerd worden. In tegenstelling tot voetballers die een actie berekenen, is het doel van de scheidsrechter het *monitoren* van de wedstrijd en indien nodig spelers te sanctioneren. De scheidsrechter

kan dus ook ingezet worden om voetballers bepaalde trainingen te laten doen. In *Soccer-Fun* is een aantal scheidsrechters ontwikkeld die de voortgang van de volgende trainingen in de gaten houden en van commentaar voorzien:

- *Slalomtraining*: implementeer een voetbalbrein dat een voetballer van de ene kant van het voetbalveld naar de andere kant laat lopen zodanig dat deze om de achtereenvolgende tegenstanders slalomt. Aan de andere kant ligt de voetbal klaar om in de goal gespeeld te worden om de opdracht succesvol af te ronden.
- *Overspeeltraining*: implementeer een voetbalbrein dat een voetballer de bal laat spelen naar de eerstvolgende teamgenoot die dicht bij het doel van de tegenstander staat. De teamgenoot mag slechts een beperkt aantal meter naar de bal toelopen om deze in bezit te nemen. De laatste voetballer moet de bal in de goal spelen om de opdracht succesvol te beëindigen.
- *Vrij-overspelentraining*: implementeer een voetbalbrein dat in staat is het goede moment te kiezen om de bal te spelen naar een medespeler zodanig dat geen tegenstander in staat is de bal te onderscheppen. De medespeler moet in balbezit komen zonder ver te lopen om de opdracht succesvol te beëindigen.
- *Keepertraining*: implementeer het voetbalbrein van een keeper die het midden van zijn goal verdedigt tegen een aantal tegenstanders die de bal naar elkaar overspelen. De keeper moet dit een bepaalde periode lang doen om te slagen voor deze opdracht.

De hierboven beschreven trainingsopdrachten lenen zich goed voor het werken met lijsten (de verzameling van spelers op het veld) en lijstcomprehensions. Voorbeelden van functies die met lijsten werken zijn: bepaal wie je teamgenoten zijn, waar deze staan, welke teamgenoot het dichtst bij je staat, welke teamgenoten niet aanspeelbaar zijn omdat ze zich in buitenspelpositie bevinden of omringd zijn door te veel

tegenstanders. Dit bereidt studenten tevens voor op het maken van breinfuncties die een echte wedstrijd gaan spelen.

De effectiviteit van een team wordt voor een groot deel bepaald door de geïmplementeerde strategie. Een team dat de hierboven beschreven strategie van de mini-F-spelers hanteert is normaal gesproken eenvoudig te verslaan door het innemen van veldposities en het overspelen van de bal. De student moet dus nadenken over een geschikte strategie en deze implementeren. Voorbeelden van functies die in dit kader ontwikkeld kunnen worden zijn: hoe zorg ik ervoor dat mijn team in balbezit komt? wat is een geschikte veldverdeling voor mijn spelers? wat is een effectieve aanval? wat is een effectieve verdediging?

In de cursus ontwikkelen studenten een voetbalteam waarin de spelers uitgerust worden met de door hen ontwikkelde voetbalbreinen. Dit voetbalteam komt uiteindelijk in actie in een toernooi dat de cursus afsluit. Om hiervoor in aanmerking te komen moet het team aan een aantal kwaliteitseisen voldoen:

- De keeper mag het strafschopgebied niet verlaten (hij mag zich niet als een *mini-F-voetballer* gedragen) en moet de bal onderscheppen als dat redelijkerwijs mogelijk is (hij moet dus actief deelnemen aan het spel).
- Veldspelers moeten een redelijke veldverdeling innemen (om te voorkomen dat ze allemaal achter de voetbal aanhollen). Veldspelers moeten de bal afspelen als dat redelijkerwijs zinvol is (om te voorkomen dat men een team van solisten ontwerpt).
- Alle voetballers dienen de spelregels in acht te nemen en de beslissingen van de scheidsrechter op te volgen. Dit is vooral bedoeld om te voorkomen dat een spel in livelock komt omdat voortdurend het verkeerde team de bal wil inwerpen of aftrappen en de scheidsrechter hierop moet ingrijpen.
- De breinfuncties moeten voldoende efficiënt zijn. Dit kan eenvoudig bepaald worden door een team tegen zichzelf te laten spelen en te eisen dat de

framerate niet lager wordt dan 20 frames per seconde. In de Soccer-Fun-GUI wordt deze teller continu getoond.

Soccer-Fun voor vwo'ers

Naast de inzet in het tot nu toe beschreven academisch onderwijs, is Soccer-Fun ook gebruikt tijdens wervingsactiviteiten voor vwo-scholieren om informatica te gaan studeren aan de Radboud Universiteit Nijmegen. Uiteraard ontbreekt bij deze voorlichtingsactiviteiten de tijd en voorkennis bij de deelnemers om geheel zelfstandig breinfuncties te ontwikkelen. Om hen hiermee te helpen krijgen ze de beschikking over een aantal voorgemaakte breinfuncties waaraan een aantal kleine aanpassingen gemaakt moeten worden. Het gebruik van brein-schetsen blijkt een goed hulpmiddel te zijn om vwo-scholieren uit te leggen hoe deze breinfuncties werken, en hoe ze de gewenste verandering kunnen implementeren. Vwo'ers reageren enthousiast op Soccer-Fun, en zijn veelal in staat om op het niveau van breinschetsen strategieën te begrijpen en aan te passen. Voor een daadwerkelijke realisatie in Clean ontbreekt vaak de tijd. We willen onderzoeken op welke manier het abstractieniveau van de huidige Soccer-Fun-API verhoogd kan worden.

CONCLUSIE

We stellen vast dat *Soccer-Fun* een welkome aanvulling is op bestaande practica functioneel programmeren. Studenten leren functionele taalconcepten toepassen zoals algebraïsche types, (hogere-orde)functies, en lijst(comprehensions) die normaal gesproken als abstract ervaren worden, maar die geconcretiseerd worden binnen Soccer-Fun. De meerderheid van de studenten reageert enthousiast op Soccer-Fun. Vanwege onze positieve ervaringen bij het inzetten van Soccer-Fun in voorlichtingsactiviteiten voor vwo-scholieren, willen we onderzoeken of het mogelijk is om onderwijsmateriaal te ontwikkelen voor

het vwo om leerlingen met functionele taalconcepten te laten werken.

VERANTWOORDING

In het verleden hebben we ervaring opgedaan in het verzorgen van voorlichtingsactiviteiten met behulp van Mathew Nelsons Robocodeproject (<http://robocode.sourceforge.net/>). Dit is een simulatiepakket om het objectgeoriënteerde paradigma uit te leggen en om ermee te werken. Sleutelementen die we hebben overgenomen in *Soccer-Fun* zijn de grafische visualisatie van de voetballers en het competitie-element (in *Robocode* strijden tanks tegen elkaar). Om het abstractieniveau van *Soccer-Fun* te verhogen willen we onderzoeken in hoeverre de methoden in het objectgeoriënteerde Alice-raamwerk [Conway 2000] van toepassing zijn op *Soccer-Fun*. In Alice kunnen gebruikers complexe acties ontwikkelen in een combinatorische stijl die erg lijkt op de functionele stijl. Een ander sleutelement van Alice is dat code tijdens de ontwikkelfase direct geëxecuteerd kan worden. Op deze manier kunnen vooral beginnende gebruikers direct het effect van hun code zien. In het simulatiepakket NetLogo [Wilensky 1999] wordt door middel van scripts die op turtles toegepast worden complexe, dynamische, systemen beschreven. Soccer-Fun onderscheidt zich hiervan door het brein als een pure functie te zien, en het gebruik van functionele taalconcepten om de breinfunctie te implementeren.



[1] Een referencemanual is te vinden in de Soccer-Fun distributie.

[2] Peter Achten. Teaching Functional Programming with Soccer-Fun. In Frank Huch, Adam Parkin, Eds. Proceedings of the 2008 ACM SIGPLAN Workshop on Functional and Declarative Programming in Education (FDPE'08), Victoria (BC) Canada, September 21, 2008, ISBN:978-1-60558-068-5, pp. 61-72, 2008.

[3] Matthew Conway et al. Alice: Lessons from building a 3D system for novices. In Thea Turner, Gerd Szwillus, Eds. CHI'00: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, ACM, pp. 486-493, 2000.

[4] Uri Wilensky. NetLogo. <http://ccl.northwestern.edu/netlogo/>. Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL, 1999.

Een module over modelchecking voor het vwo



Frits Vaandrager - Radboud Universiteit Nijmegen

David N. Jansen - Radboud Universiteit Nijmegen

Els Koopmans - Olympus College Arnhem

SAMENVATTING

We beschrijven achtergrond, leerdoelen en opzet van een lesmodule over modelchecking voor klas 5-vwo, die dit jaar voor de derde keer verzorgd wordt in een samenwerking tussen de Radboud Universiteit Nijmegen en het Olympus College te Arnhem. Tevens doen we verslag van onze ervaringen bij het geven van deze module.

TREFWOORDEN

lesmodule over modelchecking, samenwerking vo-wo.

WAAROM DEZE MODULE?

- De Radboud Universiteit Nijmegen en het Olympus College participeren, samen met de HAN en een 30-tal andere middelbare scholen uit de regio Arnhem-Nijmegen, in het project Sprint-UP. Doelen van dit samenwerkingsproject zijn onder andere:
- inhoudelijke vernieuwing bètaonderwijs;
 - leerlingen een aantrekkelijker en vollediger beeld geven van de wereld van bèta en techniek;
 - studenten en junioronderzoekers kennis laten maken met leraarschap en hen mogelijk interesseren voor een loopbaan in het onderwijs.

Een aanvulling specifiek in dit project voor informatica:

- Leerlingen kennis laten maken met een vervolgopleiding Informatica binnen het vwo.

Sprint-UP wordt gesubsidieerd door het Platform Bèta Techniek. Binnen de informatica bestaat er bij uitstek behoefte aan dit soort initiatieven. Docenten Informatica op middelbare scholen hebben vrijwel nooit Informatica gestudeerd (een wezenlijk verschil met de situatie bij andere bètavakken), het wetenschapsgebied van de informatica verandert zeer snel, en de instroom/uitstroom van ICT-opleidingen bij het vwo is structureel volstrekt onvoldoende gelet op de behoefte vanuit de samenleving. Wij vermoeden dat één van de redenen waarom scholieren zo weinig voor ICT-opleidingen kiezen is dat ze geen idee hebben van wat het wetenschapsgebied van de informatica inhoudt en naar welke fascinerende fundamentele vragen er momenteel onderzoek wordt verricht. Al met al zijn er veel goede redenen voor een nauwere samenwerking tussen vo en wo op het gebied van de Informatica!

WAAROM MODELCHECKING?

Bij het informaticaonderwijs op middelbare scholen wordt relatief veel aandacht besteed aan programmeren. In de universitaire wereld en het bedrijfsleven is er echter sprake van een duidelijke trend waarbij modellen het voornaamste product van systeemontwikkeling worden, denk bijvoorbeeld aan de opkomst van de Unified Modeling Language (UML) en Model Driven Architecture (MDA). Hierbij worden systeemeisen, gedrag en functionaliteit allemaal

beschreven in termen van (grafische) modellen. Idealiter worden deze modellen gebruikt voor de communicatie met klanten en belanghebbenden, het analyseren van het ontwerp, het genereren van code en het testen van de uiteindelijke implementatie. De beschikbaarheid van modellen vereenvoudigt tevens onderhoud en hergebruik. Wij menen dat het wenselijk is om in het middelbaar onderwijs meer aandacht te besteden aan deze ontwikkelingen, en in het bijzonder aan het *modelleren* van systemen (op een hoog niveau van abstractie) en het *analyseren* van modellen. Binnen de door ons ontwikkelde module gebruiken leerlingen het *modelchecking* pakket Uppaal voor het modelleren en analyseren van systemen (ontwikkeld door de universiteiten van Uppsala en Aalborg met bijdragen van de Radboud Universiteit Nijmegen, en gratis beschikbaar voor niet-commercieel gebruik). Binnen de universitaire wereld is er recent veel aandacht voor modelchecking. Zo is in 2007 de ACM Turing Award, vaak aangeduid als de ‘Nobelprijs voor de informatica’, uitgereikt aan de onderzoekers Clarke, Emerson en Sifakis voor hun baanbrekend onderzoek op dit terrein. Modelchecking is een succesvolle technologie om aan te tonen dat een ontwerp van een computersysteem voldoet aan de eisen die er aan gesteld worden. Met behulp van efficiënte algoritmen doorloopt een model checker automatisch de toestandsruimte van een systeemmodel en bepaalt of het systeem voldoet aan zijn specificatie. De modelchecker Uppaal beschikt over een uitstekende grafische user interface en na een korte training van één les kunnen leerlingen er al mee uit de voeten. Ze hebben dan geen weet van de (prachtige) achterliggende theorie en algoritmen, maar dat hoeft ook helemaal niet.

DOELSTELLINGEN EN OPZET

De door ons ontwikkelde module omvat twaalf lessen en wordt afgesloten met een toets. Doel van de module is om leerlingen te leren werken met een modelchecker, in het bijzonder aanpassingen maken aan bestaande modellen, zelf eenvoudige modellen construeren, en gewenste eigenschappen van systemen formaliseren en uitrekenen met behulp van een model checker. De meeste lessen (acht van de twaalf) zijn practica waarbij de leerlingen zelf aan de slag gaan met Uppaal om diverse systemen te modelleren en analyseren:

- een geldautomaat;
- een verkeerskruising;
- een spoorwegovergang;
- logische puzzels zoals ‘De wolf, de geit en de kool’ en ‘De kannibalen en de missionarissen’
- een mutual-exclusion-algoritme (voorkomt dat deelprogramma’s elkaar verstoren door tegelijkertijd een gemeenschappelijke resource te gebruiken, bijvoorbeeld een printer of gedeeld bestand);
- een algoritme voor het kiezen van een leider in een ring;
- het plannen van activiteiten in een werkplaats.

Bij de meeste practicumopdrachten geven we de leerlingen een model, dat ze vervolgens moeten aanpassen. De resterende vier lessen worden gebruikt voor inleidingen over

- betrouwbaarheid van software;
- modelchecking met Uppaal;
- mutual-exclusion-algoritmen;
- algoritmen voor het kiezen van een leider.

De keuze van de diverse toepassingen is vrij willekeurig en varieert van jaar tot jaar. Belangrijk is dat de problemen zich eenvoudig laten uitleggen, relevant zijn vanuit praktisch oogpunt, maar het gedrag toch zo complex is dat correctheid van voorgestelde oplossingen moeilijk aangetoond kan worden zonder

ondersteuning door een model checker. We proberen de leerlingen kennis te laten maken met de elegantie/schoonheid van gedistribueerde algoritmen.

ERVARINGEN

Wij zijn nog steeds bezig met de finetuning van de module en met de verdere ontwikkeling en verbetering van het lesmateriaal. Een aantal voorlopige indrukken/conclusies zijn:

- Verreweg de meeste leerlingen vinden het onderwerp van de module interessant en het werken met Uppaal leuk. De groep leerlingen die informatica volgt is nogal divers, met behalve leerlingen met profiel Natuur en techniek bijvoorbeeld ook leerlingen met profiel Cultuur en maatschappij. Dit maakt het moeilijk om de gehele populatie aan te spreken.
- De goede grafische interface van Uppaal en de robuustheid van het gereedschap zijn essentieel voor het succes van de module.
- Nauwe samenwerking tussen vo en wo was uiterst belangrijk bij de ontwikkeling van de module. De didactische aanpak die nodig is bij Informatica-leerlingen uit 5-vwo verschilt wezenlijk van de aanpak bij studenten Informatica op een universiteit. Zo is voor vwo-5-leerlingen 50 minuten een verhaal aanhoren eigenlijk te lang, terwijl dit voor studenten meestal geen probleem is. Wanneer enkele leerlingen met vragen en opmerkingen laten blijken dat ze de stof interessant vinden en goed snappen, dan is dit geen indicatie dat dit ook geldt voor de rest. Een ervaren vwo-docent pakt signalen van de leerlingen veel eerder op, ziet sneller waar het tempo lager of hoger moet, bij welke leerlingen het kwartje wel of niet gevallen is, welke vaardigheden extra training behoeven, enzovoort.
- Leerlingen zijn vaak verrassend goed in het meedenken over oplossingen van concrete problemen. Zo bedacht één leerling ter plekke het Le Lann-Chang-Roberts-algoritme voor het kiezen van een leider in een unidirectionele ring. Na één

of twee voorbeelden zien leerlingen ook razendsnel waarom allerlei mutual-exclusion-algoritmen incorrect zijn.

- Leerlingen hebben verrassend veel moeite met het omzetten van een informeel idee naar een formeel model. Modelleren is moeilijk. Dit inzicht heeft ons ertoe gebracht om meer cursustijd te gebruiken om de leerlingen zelf te laten modelleren. Modelleren, zo blijkt, leer je toch vooral door het te doen.

Ons streven is om het komende jaar voldoende documentatie en cursusmateriaal te ontwikkelen, zodat de module ook beschikbaar gesteld kan worden aan andere geïnteresseerde docenten.



REFERENTIES

- [1] De cursuswebsite, <http://www.cs.ru.nl/~fvaan/Masterclass/olympus.html>
- [2] De model checker Uppaal, <http://www.uppaal.com/>
- [3] Het Sprint-UP project, <http://www.platformbetatechniek.nl/sprintup>

Universeel systeem voor bedrijfsvoering

Dit omvat het algehele bedrijfsgebeuren; integreert uitvoering-, organisatie-, domeingegevens; geeft ruimte aan innovatieve bedrijfsontwikkelingen; verwerkt veranderingen doorlopend; wordt zonder ontwikkeling geïmplementeerd.



Arie Schinkel - Hogeschool Arnhem en Nijmegen

Deny Smeets - Hogeschool Arnhem en Nijmegen

SAMENVATTING

Veranderingen zijn hét probleem voor alle informatie-systemen. Effectieve en snelle afstemming op dynamische markten, zich wijzigende doelgroepen, nieuwe producten, ketenvorming en reorganisatie is noodzakelijk. Een afdoende oplossing kan worden gevonden in volledige flexibiliteit van alle systeem-elementen en -gegevens. Het universeel systeem voor bedrijfsvoering biedt dat perspectief. Zowel de systeem- als de informatiecomponenten zijn volledig modulair en generiek, hét uitgangspunt voor werkelijk flexibele systemen. Voor het ICT-onderwijs biedt het systeem inzicht in uniforme ICT-faciliteiten voor de bedrijfsvoering.

TREFWOORDEN

Globalisering, procesgericht, integratie, modellering, flexibel, generiek, universeel

INLEIDING

Door toenemende variatie in activiteiten en een veelheid aan specifieke systemen worden organisaties omvangrijker en complexer. Dit resulteert in herhalingen, ondoelmatigheden, misverstanden en andere ongemakken en problemen. Het maakt organisaties

star en vooral moeilijk aanpasbaar aan de versneld toenemende interne en externe veranderingen. Het universeel systeem voor bedrijfsvoering (kortweg 'bedrijfssysteem') ondervangt deze onvolkomenheden voor een groot deel door alle activiteiten onder te brengen in één 'overall' componentensystematiek. Deze is volledig gemodelleerd en alle starre specifieke systemen zijn vervangen door één flexibele generieke toepassing. Het systeem is beperkt tot één bedrijfsproces en één bedrijfsbestand, plus een generieke toepassingengenerator. Het bedrijfsproces is de virtuele versie van de werkelijke procesgeoriënteerde organisatie. Het fungeert als wegwijzer en gebruiksaanwijzing voor de werkers. Het geeft aan wat, waar, wanneer en hoe iets moet gebeuren en waar de nodige gegevens vandaan komen. Geavanceerde toepassing van informatie- en communicatietechnologie ondersteunt het bedrijfsproces. Door de grote flexibiliteit is het vooral een systeem voor werkorganisaties in veranderlijke en innoverende omgevingen. Het is toegerust voor snelle en doeltreffende aanpassing aan de beweeglijke mondiale markt.

BEPERKINGEN GEBRUIKELIJKE SYSTEMEN

Zowel de gebruikelijke conventionele systemen als de geïntroduceerde vernieuwde systemen lossen het probleem niet op.

Tekortkomingen conventionele systemen

Achterhaalde functionele taakverdeling

Tijdens de industriële revolutie in de negentiende eeuw werd de oude, nog uit de gildentijd stammende organisatie vervangen door de functionele werk-organisatie. Deze nieuwe taakverdeling had, naast voordelen, ook nadelen. Er ontstonden onderbrekingen van het werkverloop bij de overgang van de ene soort handelingen naar de andere. Dit leidde niet alleen tot langere doorlooptijden, maar ook tot tussenvoorraden. Gevolg: verstoringen van de soepele doorstroom, extra planning en registratie, animositeit tussen de afdelingen en de noodzaak tot complexe coördinatie op hoger niveau. Dat leidt tot hiërarchische organisaties, die moeilijk zijn aan te passen. Toffler (1970) zette ruim dertig jaar geleden al vraagtekens bij de bureaucratische functionele organisatie. Meer recent onderstrepen ook andere auteurs die vraagtekens (Naisbitt 1985, Morton 1991, Hammer 1993).

Moelijk aan te passen systemen

ICT-systemen en toepassingen ondersteunen sinds meer dan een halve eeuw in toenemende mate organisaties in de bedrijfsvoering. Onderhoud en aanpassing zijn vanaf het allereerste begin en ook nu nog een probleem. Daardoor gebeurde vervanging veelal te laat met daarmee gepaard gaande extra problemen.

Omgeving organisaties steeds veranderlijker

Wereldwijde ontwikkelingen in goederen- en voedselproductie, transportmiddelen, evenals technologische ontwikkelingen e.d. hebben grote veranderingen tot gevolg. Deze veranderingen dwingen tot aanpassing van organisaties op velerlei gebieden. Voor bedrijfs- en informatiekundigen vooral op het terrein van organisatie-, informatie- en kennissystemen.

Ontoereikendheid vernieuwde systemen

In ICT-vakbladen verschijnen al enige jaren berichten over onvoldoende afstemming van informatiesystemen op de nieuwe behoefte van bedrijven en organisaties. De systemen zijn verouderd en achterhaald, te omvangrijk en complex en vooral te star. De aangereikte oplossingen lijken niet afdoende.

Uitkomsten van onderzoeken

Naast bovenstaande constatering, stellen de vakmensen tevens dat er geen vervangende, volledig op de nieuwe situatie afgestemde systemen beschikbaar zijn. Ook is ingrijpend wijzigen van bestaande systemen door de koppeling aan de organisatie, dikwijls niet goed mogelijk. Deze situatie belemmert bedrijven steeds meer in de noodzakelijke aanpassing van hun organisatie en systemen aan de veranderingen.

Beperkingen van geboden oplossingen

In de ICT-vakliteratuur verschijnen al een aantal jaar steeds meer ideeën voor een betere afstemming van systemen op de ontwikkelingen. Componententechnologie, integratieamwerken, Enterprise Application Integration, bedrijfsnetwerken, Business proces management. Vrijwel alle ideeën zijn gericht op wijzigingen in de ICT-systemen. De gelanceerde ideeën hebben echter niet of onvoldoende geleid tot een conceptueel nieuw en concreet operationeel systeem. Pogingen daartoe waren voor een belangrijk deel symptoombestrijding.

Aan nieuwe systemen te stellen eisen

Uitgaande van voorgaande analyse, uitkomsten van de onderzoeken en ideeën van de deskundigen, moeten nieuwe systemen aan een groot aantal eisen voldoen (tabel 1). Realisatie is volgens de deskundigen een moeilijke opgave.

Ontkoppeling van functionele organisatie
Overkoepelend geïntegreerd
Meer voorzien in kennisproducten
Beter aanpasbaar aan veranderingen
Eenvoudig van opzet en toepassing
Component georiënteerd
Gestandaardiseerd door modellering
Eenvoudig te onderhouden
Gebruiksvriendelijk

Tabel 1
Aan nieuwe systemen te stellen eisen

EEN NIEUWE SYSTEEMBENADERING

Bedrijfsproces als uitgangspunt

In het bedrijfsproces wordt de toegevoegde waarde gegenereerd: de belangrijkste doelstelling van het bedrijf. Het systeem dient daarom primair te zijn gericht op ondersteuning van dit proces. Vervang de functieorganisatie door een procesorganisatie. Dat reduceert werkstroomonderbrekingen en tussenvoorraden met extra werkzaamheden, kosten en communicatie. Integreer de procesorganisatie en de informatievoorziening.

Integratie van systemen en gegevens

Systemen en toepassingen dienen zoveel mogelijk geïntegreerd ter voorkoming van inconsistenties en dupliceringen. Ook integratie ter bevordering van effectiviteit en efficiency dient te worden nagestreefd. Integratie van gegevens betreft uitvoering-, organisatie- en vakgebiedgegevens (werkinformatie, werkaanwijzingen en werkkennis). Een actieve, gerichte vergaring beschouwt alle in de organisatie aanwezige gegevens als een samenhangend geheel. Een toegankelijke, stimulerende beschikbaarstelling is

mogelijk, door alle categorieën gegevens direct te relateren aan de activiteiten waarbij ze nodig zijn. Deze activiteiten vormen het bedrijfsgebeuren, zodat de bijbehorende gegevens altijd eenvoudig toegankelijk en direct beschikbaar zijn. Dat maakt werkers minder afhankelijk van anderen en resulteert in een juiste en snelle afhandeling van transacties.

Doorlopende aanpassing aan veranderingen

Zowel de afzet- als voorzieningenmarkt is door een veelheid aan ontwikkelingen voortdurend aan veranderingen onderhevig. Bedrijven dienen daar op straffe van marktverlies snel op in te spelen. Dit vereist een marktgerichte, dat wil zeggen procesgeoriënteerde organisatie, die snel kan worden aangepast aan de veranderingen. Deze wordt verkregen door een modulaire opbouw, waarin alle onderdelen uitwisselbaar zijn. Modellering is daarvoor het aangewezen instrument.

Flexibiliteit is ook van belang voor de voorziening in maatwerk en kwaliteit. De organisatie stemt de producten af op specifieke wensen van de afnemers: zowel op groepen van afnemers (confectie) als op individuele afnemers (maatwerk). Daarbij gaat het niet alleen om de producten zelf, maar ook om de inschakeling van de afnemer bij de totstandkoming. Dit past in het streven naar continuïteit en beperking van onzekerheid.

Geavanceerde ICT-ondersteuning van bedrijfsvoering

De nieuwe systemen maken optimaal gebruik van de mogelijkheden van ICT. De snelheid en capaciteit van de apparatuur en de mogelijkheden van de programmatuur maken een geavanceerde ondersteuning van de bedrijfsvoering mogelijk. Specifieke werkwijzen kunnen generiek worden gemaakt met behoud van specifieke producten. Dit geldt eveneens voor specifieke middelen, handelingen, regels en gegevens. Voordelen zijn:



Bezoeker van het NIOC

- beperking van min of meer grote aantallen specifieke processen tot één generieke
 - grote consistentie in alle middelen, handelingen, regels en gegevens
 - uitwisselbaarheid, dus flexibiliteit bij opbouw, wijziging, onderhoud
- Geavanceerde toepassing van ICT maakt het tevens mogelijk de onderlinge toelevering gedetailleerd te regelen en te plannen, zodat tussenvorraden met alle bijkomende handelingen tot een minimum worden teruggebracht. Tot slot: door verdere automatisering van routinewerkzaamheden krijgen werkers meer tijd voor innoverende activiteiten.

OPZET VAN BEDRIJFSSYSTEEM

Het 'Universeel systeem voor bedrijfsvoering' bestaat uit een bedrijfsproces, een bedrijfsbestand en een applicatiegenerator. De structuur van het bedrijfsproces vervangt de gebruikelijke functionele structuur. De verzameling bestandelementen van het bedrijfsbestand komt in de plaats van de gezamenlijke bestanden. Niet alleen gegevenseenheden (records) worden als elementen beschouwd, ook alle componenten van activiteiten. Dat zijn alle middelen, handelingen en regels. De applicatiegenerator komt in de plaats van de gebruikelijke informatieverwerkende systemen. Hij vervangt alle ontwikkelde, aangeschafte en gehuurde specifieke en halfspecifieke systemen en toepassingen.

Organisatie van systeem

Toelichting op proces, bestand en generator.

Systeemdeel bedrijfsproces

Het bedrijfsproces bevat alle activiteiten die zich in de organisatie voordoen; technische en commerciële, verzorgende en administratieve, primaire en ondersteunende, uitvoerende en leidinggevende. Ze kunnen variëren van karakter en omvang, eenvoudig en moeilijk, beperkt en omvangrijk, samengesteld en enkelvoudig. Ze vormen de kern van de organisatie en

daarmee ook van het systeem.

De structuur van het bedrijfsproces is één, alle activiteiten bevattende rubricering, die het werkverloop volgt als een opeenvolging van bewerkingen, logisch en chronologisch geordend. Alle activiteiten zijn ondergebracht in dit ene proces, vanaf het bepalen van het bedrijfsbeleid tot en met het afsluiten van het bedrijfsgebeuren.

De gegevens van de activiteiten zijn vastgelegd in een activiteitspecificering; een werkaanwijzing voor de uitvoering van de activiteit.

Systeemdeel bedrijfsbestand

Het bedrijfsbestand bevat alle elementen nodig voor het uitvoeren van activiteiten. Ze worden onderscheiden in componenten (middelen, handelingen en regels), objecten (personele, materiële en immateriële) en resultaten (informatie-, organisatie- en kennisproducten). De componenten voeren de activiteit uit, de objecten ondergaan de uitvoering en de resultaten zijn het gevolg van de uitvoering.

De structuur van het bedrijfsbestand is één, alle elementen bevattende rubricering naar functie. Alle elementen die zich in de organisatie voordoen zijn ondergebracht in dit ene bestand.

De elementen zijn verkregen door ontleding van bestaande activiteiten en zijn door modellering generiek gemaakt. Hierdoor kunnen ze in verschillende activiteiten worden toegepast, terwijl ze slechts eenmaal in het bestand voorkomen.

Toepassingengenerator

Het systeem bewerkt de gegevens voor het vervaardigen van een informatieproduct, zodanig, dat de verwerking kan worden uitgevoerd met een 'generieke' applicatiegenerator in plaats van een 'specifiek' informatie-systeem (Bers, 2004). Door de gegevens vooraf op een bepaalde wijze te modelleren, selecteren en ordenen worden met deze generieke verwerking specifieke producten verkregen. De generator kan, op basis van instellingen door de ontwikkelaar en de ingevoerde gegevens van de gebruiker.

Inhoud van systeem

De inhoud van het systeem bestaat uit basisgegevens en gegevensproducten. Beide categorieën zijn onderverdeeld in: uitvoering-, organisatie- en domeingegevens/producten, ofwel werkinformatie, werkaanwijzingen, werkkennis.

Basisgegevens

Uitvoeringgegevens zijn de basis voor informatieproducten en zijn gewoonlijk in bestanden van informatiesystemen opgenomen (technische gegevens, uitgevoerde acties, voorraden, gebruik, actiedatums e.d.). Organisatiegegevens vormen de bouwstenen voor inrichtingproducten (taken, functies, rollen, hoedanigheden, relaties e.d.). Domeingegevens geven inhoud aan kennisproducten (vak kennis, materiaalkennis, bewerking kennis, risicofactoren, veiligheidsaspecten e.d.)

Gegevensproducten

Uitvoeringproducten zijn de resultaten van de gebruikelijke informatiesystemen, zoals bestellingen, facturen, salaris-specificaties, saldiijsten, brieven, verslagen e.d. Organisatieproducten zijn de uitkomsten van het ICT-ondersteunde werk van organisatiedeskundigen (werkaanwijzingen, inwerkprogramma's, taak/functiebeschrijvingen, functieprofielen, levensloopoverzichten, relatiepatronen, organisatie-structuren e.d.). Domeinproducten zijn de producten van het ICT-ondersteunde werk van opleiders (syllabussen, casusuitwerkingen, presentaties, lesprogramma's, cursusprogramma's e.d.).

MODELLERING VAN SYSTEEM

Het systeem is in alle opzichten gemodelleerd en genormeerd. Daardoor is het ook mogelijk de gehele organisatie als model te presenteren. Onderscheiden worden modellering van onderverdelingen (structuren), onderdelen (uitvoeringseenheden) en gegevens.

Modellering van onderverdelingen

De onderverdeling betreft het bedrijfsproces en het bedrijfsbestand. De modellering van de onderverdeling gebeurt bij beide op dezelfde wijze: een voortgezette 'onderverdeling in drieën' op drie maal drie niveaus. De onderdelen moeten het geheel omvatten, elkaar uitsluiten en evenwichtig zijn. De driedeling bij het bedrijfsproces is gebaseerd op fase (voorbereiding, uitvoering en afwerking). De driedeling bij het bedrijfsbestand is gebaseerd op functie.

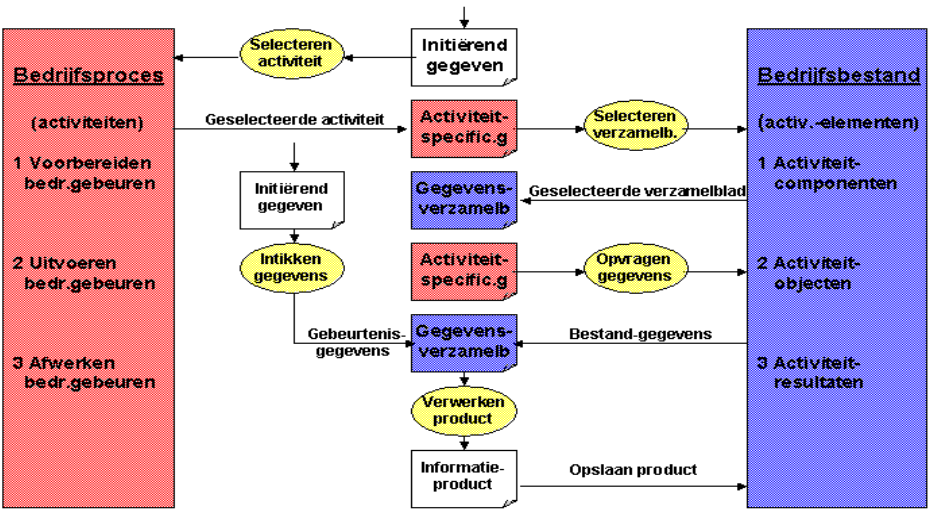
Modellering van uitvoeringseenheden

De modellering van de uitvoeringseenheden betreffen de inhoud en de indeling van de specificeringen. Onderscheiden worden activiteitspecificeringen en elementspecificeringen. De modellering van de specificeringen bestaat bij alle twee uit een driedeling van de inhoud. De activiteitspecificeringen onderscheiden functie, te nemen stappen en te raadplegen bestandselementen. De elementspecificeringen onderscheiden een driedeling naar categorieën gegevens.

Modellering van gegevens

Bij de gegevensmodellering wordt gebruik gemaakt van een in de praktijk beproefde methode van feittype georiënteerde informatiemodellering (Bakema, 1996). Bij deze methode worden 'feiten' uit de bestaande (ongemodelleerde) informatieverwerking verwoord, vastgelegd, geclassificeerd en verwerkt tot de nieuwe (gemodelleerde) informatieverwerking. Gereedschappen ondersteunen bepaalde bewerkingen van de modellering (modelconversie, databasegeneratie, e.d.). Deze gegevensmodellering zorgt er niet alleen voor, dat alle elementen slechts eenmaal in het systeem voorkomen, het normeert tevens de aanduidingen van de elementen. Modelleringmethode en applicatiegenerator zijn op elkaar afgestemd (Bakema, 2004). Diverse andere modelleringsmethoden, als RM, ER, NIAM, Infomod, ORM, UML kunnen worden toegepast (Lek, 2005).

Werkwijze activiteit



Figuur 1
Principewerking van systeem

Bovenstaande gegevensmodellering sluit aan op de opzet en modellering van het bedrijfssysteem. Evenals de organisatiecomponenten komen de gegevenselementen door de modellering slechts eenmaal in het systeem voor.

GEBRUIK EN INVOERING VAN SYSTEEM

Informatieverwerking zonder toepassingen

Het bedrijfssysteem beschikt niet over de gebruikelijke specifieke toepassingen. De gebruiker componeert zelf de toepassing op het moment van uitvoering. In plaats van een pakket specifieke toepassingen beschikken de gebruikers over één generieke toepassingengenerator. De specifieke elementen brengt de gebruiker zelf in,

daarbij geholpen door de activiteitspecificering die alle aanwijzingen daartoe bevat. Ook een onervaren gebruiker kan de activiteit uitvoeren, omdat het bedrijfssysteem de uitvoering van elke activiteit volledig ondersteunt. Voor principewerking zie figuur 1, voor uitwerking zie hoofdstuk 6).

Invoering door overnemen model

Het systeem is in alle opzichten gemodelleerd en dus generiek. Dit maakt de invoering een stuk eenvoudiger, omdat gebruikgemaakt kan worden van model- of voorbeeldsystemen. Overname en aanpassing aan de eigen situatie is zondermeer mogelijk. Er is een uitgewerkt generiek model beschikbaar, waarin alle normaliter voorkomende activiteiten zijn opgenomen. Voor branches worden semispecifieke modellen opgezet. Het systeem kan geleidelijk worden ingevoerd zonder schoksgewijze implementatie.

VOORBEELDTOEPASSING VAN SYSTEEM

Het voorbeeld betreft een toepassing bij kernactiviteiten (drijven van handel) en gaat over het opstellen van een offerteaanvraag bij een rijwielfabriek. De activiteit wordt uitgevoerd door een offertebehandelaar van de afdeling 'Kopen productiematerialen'. De werkwijze van zowel de werker als de computer is generiek, doch specifiek benoemd.

Installeren van activiteit

Het installeren van de activiteit begint met het kennisnemen van het initiërend gegeven, het 'Koopverzoek'. De werker beoordeelt of het verzoek duidelijk en volledig is en of het past binnen de gebruikelijke wijze van werken. Vervolgens wordt de activiteitspecificatie 'Opstellen offerteaanvraag' geactiveerd. Dit gebeurt door het intikken van de activiteitcode (het systeem selecteert de activiteit uit het bedrijfsproces). De activiteitspecificatie geeft een volledig beeld van de te nemen stappen en de uit het bedrijfsbestand op te halen gegevens. Tot slot wordt het in de specificatie aangegeven gegevensverzamelblad 'Actie-documenten' geactiveerd. Hiermee is de activiteit geïnstalleerd. (Bij het opstellen van een aantal 'aanvragen', zoals gebruikelijk, worden deze handelingen slechts eenmaal uitgevoerd.)

Inbrengen van gegevens

Het inbrengen van gegevens (invullen van het verzamelblad) betreft identificerende gegevens, vaste bestandgegevens en variabele gebeurtenisgegevens. De identificerende 'aanvraag'-gegevens worden overgenomen van de activiteitspecificering (benoeming e.d.) en gegenereerd door het systeem (volgnummer, datum e.d.). Vaste bestandgegevens betreffen te offrenen materialen, potentiële leveranciers, te stellen condities e.d. Deze worden volgens aanwijzingen op de specificering opgehaald uit het bedrijfsbestand. De variabele gebeurtenisgegevens zijn, te offrenen aantallen, offerte- en leverdata, bijzonderheden e.d.

Ze worden overgenomen (ingetikt) van het 'Verzoek'. De inhoud van het verzamelblad wordt gecontroleerd, waarmee het inbrengen van de gegevens is voltooid.

Verwerken van product

Het verwerken van het product bestaat uit:

- het opstellen van de aanvraag
- het vastleggen van de uitvoering
- het verzenden van de aanvraag.

De aanvraag wordt opgesteld door het activeren van de verwerker (Universele Toepassingen Generator), die de inhoud van het verzamelblad omzet in het product. Na verschijning van de aanvraag op het scherm wordt deze gecontroleerd en in geval van postverzending afgedrukt. De uitvoering van de activiteit wordt als 'lopende actie' vermeld in het bestand (records van de betreffende materialen en leveranciers. Daarnaast in het voortgangscntrole systeem. De ingevulde gegevensverzamelplaat doet dienst als archiefexemplaar van de aanvraag en wordt onder resultaten in het bestand opgeborgen. Tot slot wordt de aanvraag verzonden aan de potentiële leveranciers en wordt, indien gewenst, de verzoeker in kennis gesteld. De verzending vindt plaats per e-mail, fax of post.

Vergelijking

Activiteitspecificatie, initiërend gegeven, verzamelblad, evenals alle bestandselementen vormen een integrerend deel van het systeem, zodat bovenstaande handelingen voor een belangrijk deel automatisch worden uitgevoerd. De generieke toepassing genereert een specifiek product. De door de gebruiker uit te voeren handelingen zijn niet wezenlijk anders dan bij gebruik van een specifieke toepassing. Het ontwerpen en onderhouden van een soms groot aantal specifieke toepassingen is overbodig.

BIJZONDER SYSTEEM IN MEER OPZICHTEN

Wat is er bijzonder aan het systeem? Welke kenmerken bevorderen de flexibiliteit die in veel opzichten zo belangrijk is? In hoeverre verschilt het van de conventionele systemen?

Systeem is wezenlijk anders

Het ‘Universeel systeem voor bedrijfsvoering’ is een ‘alles omvattend’ systeem, ondergebracht in één systematiek. Het omvat alle werkactiviteiten (primaire, ondersteunende en sturende) en alle werkgegevens (werkinformatie, werkaanwijzingen en werkkennis). Het is de virtuele versie van de werkelijke organisatie. Het geeft aan wat, waar, wanneer, waarmee en hoe iets moet gebeuren. Het is door z’n flexibiliteit vooral een systeem voor werkorganisaties in veranderlijke omgevingen. Het ‘Universeel bedrijfssysteem’ komt tegemoet aan de door de deskundigen en managers geformuleerde eisen (zie paragraaf 1).

Universeel bedrijfssysteem	Conventionele systemen
Eén algemeen bedrijfssysteem	Verschillende soorten systemen en toepassingen
Systeem met ruimte voor bedrijfsinnovatie	Innovatiebelemmerende starre toepassingen
Logische (natuurlijke) processtructuur	Complexe (onnatuurlijke) functionele structuur
Eén genormeerde gegevensbank	Aantal verschillende gegevensbanken
Algehele modellering van systeem	Grote differentiatie van systeemelementen
Eén uniforme werkwijze	Meer uiteenlopende werkwijzen
Toepasbaar in alle bedrijven/organisaties	Alleen toepasbaar in zelfde specifieke omgeving
Doorlopende aanpassing aan veranderingen	Periodieke, dikwijls omslachtige aanpassing
Inrichting door overneming en aanpassing	Inrichting door langdurige, kostbare ontwikkeling
Implementatie geleidelijk	Implementatie schoksgewijs

Tabel 2

Verschillen tussen bedrijfssysteem en conventionele systemen

Vergelijking met conventionele systemen

Volgens deskundigen op ICT-gebied, zijn de conventionele informatiesystemen niet meer opgewassen tegen de eisen die de veranderlijke omgeving stelt. Wensen en ideeën zijn er wel, maar geschikte nieuwe systemen niet. Het universeel bedrijfssysteem kan in de nieuwe behoefte voorzien. Tabel 2 toont een vergelijking met de conventionele systemen, waarbij het bedrijfssysteem als uitgangspunt is genomen.



[1] Bakema, B., J.P. Zwart en H. van der Lek (1996). Volledig Communicatiegeoriënteerde Informatiemodellering FCO-IM. Academic Service, Den Haag.

[2] Bakema, G. (2004). Metadata management en applicatiegeneratie; van visie naar toepassing. TINFON 4, p. 100-105.

[3] Bers, A.C. van (2004). Organisatiemodellering?, Applicatie Generatie. NIII, Radboud Universiteit, Nijmegen.

[4] BKIB, Stichting (1995). Hoe nu verder? Afstemming van informaticaonderwijs op behoefte van bedrijfsleven. Amsterdam, Stichting BKIB.

[5] CINOP, (1997). Scholing van werkenden in ICT-ondersteunde werkuitvoering. Amsterdam, Stichting BKIB.

[6] Hammer, M. en Champy J. (1993). Reengineering? the corporation, A manifesto for Business Revolution. Hammer and Company.

[7] Lek, H. v.d. (2005). Informatieanalyse is niet uit te besteden. Automatisering Gids 25, p. 15.

[8] Morton, S. (1991). The corporation of the 1990s, Information Technology and Organizational Transformation. Oxford University Press.

[9] Naisbitt, J. (1985). Reinventing the corporation. Warner Books. Profound, (1995). Informaticaonderwijs en de toekomstige behoefte van het bedrijfsleven; verslag van een onderzoek. Amsterdam, Stichting BKIB.

[10] Schinkel, A. (1995). Vraag en aanbod informatica-onderwijs; nu ook marktmechanisme in onderwijs? TINFON 4, p.132-134.

[11] Schinkel, A. en Van der Kamp P. (1998). Methode voor bepaling werk- en leerkennis; ict-kennis in een veranderende wereld. TINFON 1, p. 25-29.

[12] Schinkel, A. en Van der Kamp P. (1999a). Kennis ICT-ondersteunde werkuitvoering. TINFON 1, p. 9-14.

[13] Schinkel, A. (1999b). De methodische aanpak van BKIB. Telewerken 3, p. 6-7.

[14] Schinkel, A. (2002). Bank voor werkkennis. TINFON 4, p. 171-173.

[15] Smith, A. (1776). An Inquiry into the Nature and Causes of the Wealth of Nations. Londen, W. Strahan & T. Cadell.

Toffler, A. (1970). Future shock. New York, Random House.

VERANTWOORDING

Het ‘Universeel systeem voor bedrijfsvoering’ is het resultaat van een evolutionair ontwikkelproces. In 1993 is op verzoek van de Stichting BKIB (Bevordering Kwaliteit Informatica Beroepsopleiding) door de auteur een onderzoek opgezet om ICT-kennis te integreren in vakkennis (Schinkel, 1995). Dit onderzoek werd bij bekende Nederlandse bedrijven uitgevoerd door Profound (1995). Het resulteerde in een congres en een aantal aanbevelingen aan de Minister van Onderwijs waarin o.a. de suggestie werd overgenomen om IT te vervangen door ICT (BKIB, 1995). Aansluitend is een methode ontwikkeld om de nieuwe geïntegreerde kennis vanuit de praktijk te bepalen (Schinkel, 1998 en 1999b). Deze methode werd getoetst in een door CINOP (1997) uitgevoerd onderzoek.

De Hogeschool van Arnhem en Nijmegen (HAN) stelde de methode beschikbaar via internet en ontwikkelde daarvoor programmatuur. Vervolgens werd ook het product van de methode (Werkkennis) op internet geplaatst (Schinkel, 1999a). Zo ontstond een Bank voor werkkennis (Schinkel, 2002), die de ontwikkelde uitbouwden tot een ‘overall’, volkomen flexibel, gegevensverwerkend systeem, met als producten: werkinformatie, werkaanwijzingen, werkkennis. Dit systeem sluit aan op de geavanceerde, al eerder door HAN ontwikkelde, programmatuur.

Hoe denken we er over? Informatica in het voortgezet onderwijs



Nico van Diepen - Universiteit Twente

SAMENVATTING

De auteur heeft onderzoek gedaan naar de verschillende inzichten en opvattingen over het vak informatica in het vo. Daarbij heeft hij informatici uit vo, hbo, wo, lerarenopleiding en auteurssteams gevraagd om zevenendertig stellingen op een specifieke wijze te sorteren, volgens de regels van de Q-methodologie. Deze stellingen betroffen het vak informatica, de plaats van het vak en de rol van de docent. Negenenvijftig aangeschrevenen hebben de Q-sort ingevuld. Door clusteranalyse is gezocht naar punten van overeenstemming en van verschil, naar groepen gelijkgezinden en naar thema's waarover groepen het eens zijn. De antwoorden clusterden tot vier factoren die ruwweg te benoemen zijn als de wetenschappelijke kijk, de docentvisie, de non-sense visie en de vernieuwersvisie. Bij iedere factor zijn kenmerkende en onderscheidende stellingen aangegeven.

INLEIDING

Er leven veel opvattingen over het vak Informatica in het voortgezet onderwijs. In de media, in het werkveld, op het NIOC en op het congres van I&I vinden hevige discussies plaats. Uit het vakdossier Informatica [1] uit andere publicaties rijzen vragen over Informatica op. In 2008 heeft er een onderzoek plaats gevonden om de opinies over het vak in kaart te brengen. In deze bijdrage worden resultaten van dit onderzoek gepubliceerd.

Om uitspraken te kunnen doen over en verklaringen te kunnen geven van subjectieve zaken als persoonlijke gezichtspunten, meningen, houdingen, gevoelens en dergelijke is de Q-methodologie [2] een krachtig instrument. In dit onderzoek is daarvoor gekozen. Na interviews met betrokkenen en na bestudering van relevante bronnen zijn stellingen geformuleerd over informatica in het voortgezet onderwijs. Deze stellingen hebben betrekking op de inhoud van het vak, de plaats van het vak in de school en de rol van de docent. Na overleg met deskundigen zijn zevenendertig stellingen geselecteerd:

1. In het examenprogramma wordt onvoldoende onderscheid gemaakt tussen havo en vwo.
2. De feitelijke inhoud van het vak Informatica is te veel afhankelijk van de docent.
3. De eindtermen van het vak Informatica zijn te globaal.
4. Een centraal (schriftelijk of praktisch) eindexamen voor Informatica is gewenst.
5. Projectwerk is een werkvorm en hoort daarom niet in het examenprogramma Informatica thuis.
6. In het programma voor Informatica ontbreekt een funderende theoriemodule, met bijvoorbeeld logica en automaten theorie.
7. Voor informatiemodellering is ERM voor havo de beste keuze.
8. De concepten uit de objectoriëntatie zijn te complex voor leerlingen in het vo.
9. De huidige beschikbare lesmethoden voor Informatica voldoen.
10. De breedte van het vak is de kracht van het vak.

11. Het vak Informatica is nog onvoldoende uitontwikkeld.
12. De leerlingen van alle profielen moeten hetzelfde informaticaprogramma aangeboden krijgen.
13. De site www.informaticavo.nl is een onmisbaar hulpmiddel voor informaticadocenten in het vo.
14. Het vak Informatica is goed te toetsen zonder Schriftelijk deel, dus met alleen een Praktisch deel en een Projectdeel.
15. Er moet meer aandacht besteed worden aan de rol van ICT in de maatschappij, vooral aan de ethische en juridische aspecten.
16. Een centraal (schriftelijk of praktisch) eindexamen is nodig voor de borging van het niveau en de kwaliteit.
17. Voor informatiemodellering is FCO-IM voor vwo de beste keuze.
18. Een module Kunstmatige intelligentie moet onderdeel uitmaken van het verplichte deel van Informatica.
19. Onderscheid tussen havo en vwo ligt niet in de domeinen maar in de werkvormen.
20. Docenten in het vo kiezen voor de onderwerpen robotica en gaming voornamelijk omdat de leerlingen het zo leuk vinden.
21. Er moet meer aandacht besteed worden aan de rol van ICT in de persoonlijke levenssfeer, vooral communicatie, information-overload en privacy.
22. Als leerlingen bij het bouwen van een website veel HTML coderen voldoen ze aan de eisen die het examenprogramma aan het onderdeel 'programmeren' stelt.
23. Vanwege de collegiale bijsturing moet een informaticadocent die een eenpersoonssectie vormt, een koppel vormen met een collega van een andere school.
24. Docenten die bevoegd zijn voor Informatica moeten ook bevoegd verklaard worden voor NLT.
25. Het is de plicht van docenten informatica het vak zo aantrekkelijk mogelijk te maken voor meisjes.

26. Informatica moet een verplicht vak worden voor alle profielen.
27. I&I, de beroepsvereniging van informaticadocenten, moet een stelsel van certificering opzetten
28. Informatica moet zich meer profileren als bètavak.
29. Er moeten modules ontwikkeld worden, gericht op het profiel Economie & Maatschappij.
30. De vakinhoudelijke kennis van CODI-geschoolde docenten is voldoende.
31. Informatiebeveiliging moet verplicht onderdeel van het vak zijn.
32. Schoolleiders en schooldecanen weten te weinig over de inhoud van het vak Informatica.
33. Iedere havo/vwo-school dient het keuzevak Informatica aan te bieden.
34. Er moet een Informaticamodule voor de onderbouw komen om de leerlingen voor te bereiden op de keuze van het vak.
35. Binnen de klas moet er gedifferentieerd worden voor de leerlingen van de verschillende profielen.
36. De functiecombinatie Informaticadocent/Systeembeheerder geeft een verkeerd signaal af over het vak.
37. De informaticadocent heeft een voorbeeldrol voor zijn collega's op het gebied van gebruik van ICT in het onderwijs.

Via een webgebaseerde tool zijn deze stellingen voor-gelegd aan betrokkenen uit het veld met de opdracht deze stellingen te ordenen in zeven categorieën. De genodigde respondenten zijn docenten informatica uit het voortgezet onderwijs en het hoger onderwijs, auteurs van informaticamethoden, studenten van de lerarenopleidingen informatica, vakdidactici van de lerarenopleidingen informatica en enkele personen die zich met het beleid rond het vak informatica bezig houden. Negenenvijftig personen hebben de stellingen op de voorgeschreven wijze gesorteerd, waarmee een schat aan gegevens is verworven.

ANALYSE VAN DE FREQUENTIES

De frequenties van de antwoorden laten zien dat vrijwel alle keuzes voorkomen. Bij de meeste stellingen variëren de gegeven antwoorden van -3 tot +3. Dit onderbouwt de eerste zin van de inleiding dat er veel opvattingen leven.

De hoogst positief scorende stellingen zijn de volgende.

- +85: item 33. Iedere havo/vwo-school dient het keuzevak Informatica aan te bieden.
- +60: item 32. Schoolleiders en schooldecanen weten te weinig over de inhoud van het vak Informatica
- +59: item 11. Het vak Informatica is nog onvoldoende uitontwikkeld.
- +56: item 34. Er moet een Informaticamodule voor de onderbouw komen om de leerlingen voor te bereiden op de keuze van het vak.
- +44: item 36. De functiecombinatie Informatica-docent/Systeembeheerder geeft een verkeerd signaal af over het vak.

Hoogst negatief scoren de volgende stellingen.

- 120: item 22. Als leerlingen bij het bouwen van een website veel HTML coderen voldoen ze aan de eisen die het examenprogramma aan het onderdeel ‘programmeren’ stelt.
- 72: item 14. Het vak Informatica is goed te toetsen zonder schriftelijk deel, dus met alleen een praktisch deel en een projectdeel.

- 56: item 5. Projectwerk is een werkvorm en hoort daarom niet in het examenprogramma Informatica thuis.
- 52: item 30. De vakinhoudelijke kennis van CODI-geschoolde docenten is voldoende.
- 50: item 12. De leerlingen van alle profielen moeten hetzelfde informaticaprogramma aangeboden krijgen.

FACTORANALYSE

Factoranalyse met behulp van het programma PQMethod2.11 [3] leverde vier factoren op.

Factor 1: Er lijken twijfels naar voren te komen over de kwaliteit en de organisatie van het huidige onderwijs. Het oordeel over de CODI-opgeleide docent is hard. Het totaalbeeld lijkt negatief kritisch. Er zijn zestien respondenten die hoog laden op deze factor. Hiervan laadt één persoon hoog negatief. De groep bestaat uit elf docenten uit het wo, een uit het hbo, twee studenten uit de universitaire lerarenopleiding, één vo-docent en één uit een andere categorie. De vo-docent is degene die negatief laadt. Deze factor kun je karakteriseren als de ho-kijk, of misschien zelfs de wo-kijk.

Factor 2: Opvallend is de weerstand tegen een centraal eindexamen en de wens voor een informatica-module in de onderbouw. Het totaalbeeld lijkt positief, realistisch en tevreden. Er zijn twaalf respondenten die hoog scoren op deze factor. Zeven zijn afkomstig uit het vo en vijf uit het ho. Deze factor lijkt de docentvisie te representeren.

Factor 3: OO-concepten zijn niet te complex, geen bètaprofilering, breedte is kracht, methoden voldoen: dit zijn wat opvallende stellingen. Deze factor straalt een no-nonsense geest uit; ga aan de slag met wat er is, de mogelijkheden liggen er.

Henk Plessius, voorzitter van de NIOC-stuurgroep bedankt John Diedonks, gespreksleider van de openingssessie.



Twaalf respondenten laden hoog op deze factor. De meesten hiervan zijn vo-docenten en een flink deel is tevens medeauteur van een informaticamethode. Factor drie is te beschouwen als de no-nonsense visie.

Factor 4: Zeven respondenten laden hoog op deze factor. Deze groep is het meest kritisch over de huidige situatie. Bij een positieve interpretatie kan deze groep de vernieuwers genoemd kunnen worden. Een negatieve benaming zou ontevredenen kunnen zijn. Van deze zeven volgden ten tijde van het onderzoek vier respondenten de universitaire lerarenopleiding en waren er twee als docent verbonden aan een lerarenopleiding. Factor 4 is wordt verder benoemd als de vernieuwersvisie.

ONDERSCHEIDENDE STELLINGEN
TUSSEN GROEPEN

In tabel, op pagina 150, staan de stellingen genoemd die bij de vergelijking van twee groepen het grootste verschil opleveren. Een positief verschil wil zeggen dat de eerstgenoemde groep de stelling hoger waardeert dan de tweede. Bij een negatief verschil is dat precies andersom.

Groep 1	Groep 2	telling	Verschil
Wo-kijk	Docentvisie	4	3.151
		16	3.045
		9	-1.739
		13	-2.154
Wo-kijk	No-nonsense visie	28	3.121
		16	1.730
		9	-2.261
		10	-2.288
Wo-kijk	Vernieuwersvisie	16	2.609
		4	2.160
		36	2.041
		6	-2.064
		24	-2.806
		28	2.174
Docentvisie	No-nonsense visie	34	1.838
		32	-1.706
		4	-1.845
		13	2.050
Docentvisie	Vernieuwersvisie	36	1.758
		5	-2.041
		24	-2.977
		6	-3.176
		9	2.207
		13	1.694
No-nonsense visie	Vernieuwersvisie	11	-2.199
		6	-3.187

Tabel 1:
Stellingen die de grootste verschillen opleveren bij het vergelijken van twee groepen

CONSENSUS STELLINGEN EN KENMERKENDE STELLINGEN

Er is maar één stelling waarover tussen de groepen consensus bestaat, namelijk stelling 19. Alle groepen plaatsen deze in categorie -1.

De tien stellingen waarover de meeste consensus tussen de groepen bestaat zijn, in afnemende volgorde 19, 2, 29, 14, 15, 31, 26, 20, 12 en 35. De gemiddelde scores voor deze stellingen zijn bij benadering -1, 1, 0, -2, 1, 0, -1, 0, -2 en 1.

Minste consensus bestaat over de stellingen 6, 24, 16, 4, 28, 9, 13, 10, 36 en 11. Stelling 6 wordt bijvoorbeeld door groep 2 en 3 in categorie -2 geplaatst, door groep 1 in categorie 0 en door groep 4 in categorie 3.

Kenmerkende stellingen voor een groep zijn stellingen die door die groep in een categorie zijn geplaatst die duidelijk afwijkt van de drie overige. Stelling 34 is bijvoorbeeld kenmerkend voor de docentvisie omdat die groep deze stelling gemiddeld op 2.12 indeelde, terwijl groepen 1, 3 en 4 de waarden 0.53, 0.29 en 1.06 gaven.

Voor groep 1, de wo-kijk, zijn de kenmerkende stellingen 16, 4, 28, 6 en 10. Kenmerkend voor groep 2, de docentvisie, zijn stellingen 34, 25, 21, 9, 3, 4 en 5. De no-nonsense groep, groep 3, heeft als kenmerkende stellingen 10, 9, 4, 16, 36, 11, 24, 5, 18, 8, 28. Voor de vernieuwers, groep 4, zijn 11, 6, 24, 4, 36, 7 en 22 kenmerkend.

CONCLUSIE EN AANBEVELING

Er zijn veel verschillende opvattingen over informatica in het vo. In dit onderzoek komen vier clusters naar voren die ruw zijn te karakteriseren als wo-kijk, docentvisie, no-nonsense visie en vernieuwersvisie. Punten van verschil en van overeenstemming zijn in kaart gebracht. Commissies, die zich bezig houden met vernieuwing van het vak informatica in het vo doen er verstandig aan zich deze verschillen in opvatting te realiseren en ernaar te handelen.



REFERENTIES

[1]
http://www.slo.nl/downloads/archief/341330003_Vakdossier_2007_informatica.pdf/

[2]
<http://www.lrz-muenchen.de/~schmolck/qmethod/srbqhc.htm>

[3]
<http://www.lrz-muenchen.de/~schmolck/qmethod/>

Taal- en spraaktechnologie

Marijke Loots - St. Bonifatiuscollege Utrecht

SAMENVATTING

In samenwerking met Studio Taalwetenschap, en ingebed in een projectgroep Taal - en Spraak-technologie heeft het nationaal expertisecentrum leerplanontwikkeling (de SLO) een module Spraak-synthese ontwikkeld en op het web beschikbaar gesteld. In de presentatie zijn de belangrijkste onderdelen van de module getoond. De uitwerking van enkele opdrachten is gedemonstreerd met behulp van de in de module genoemde syntheseprogramma's. Ten slotte zijn enkele mogelijke aanvullingen genoemd op de stof van de module.

TREFWOORDEN

taaltechnologie, voortgezet onderwijs, spraak, taal



INLEIDING

In het voortgezet onderwijs wordt weinig aandacht besteed aan de formele eigenschappen van taal en spraak. De module Spraaksynthese is een eerste poging om interesse te wekken voor onderwerpen als spraaksynthese, spraakperceptie en automatisch vertalen. De module is in het voorjaar van 2009 beschikbaar gesteld op de website van de SLO. Het leek een goed idee om docenten in het voortgezet onderwijs naar hun reactie te vragen. Daarom is in de presentatie een demonstratie gegeven van de inhoud van de module. Daarbij is ook aangegeven wat aan extra materiaal zou kunnen worden behandeld. Deze tekst is, net als de presentatie, bedoeld om een beeld te geven van de module en de context waarin deze kan worden aangeboden.

OPBOUW VAN DE MODULE



De module van 20 studiepunten moet leerlingen vertrouwd maken met het begrip text-to-speech: het laten uitspreken van zinnen door een computer. De module is deel van een serie die mogelijk nog een module spraaksynthese en automatisch vertalen zal bevatten.

Het doel is leerlingen

- te laten kennismaken met een aantal manieren om spraaksynthese te realiseren, en hun de voor- en nadelen te laten zien.
- te leren een eenvoudig spraaksynthesesysteem met een beperkte woordenschat te maken.

De module vereist enige programmeerkennis, omdat om een programma in Visual Basic moet worden begrepen en aangepast. Bovendien moet de leerling enige kennis hebben van relationele databases.

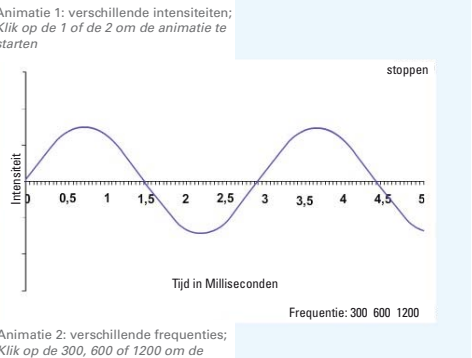
Dat laatste bepaalt ook de benodigdheden voor de module: een Windows XP machine uitgerust met MS Access 2003. De opbouw van de module zal duidelijk worden in de bespreking van de verschillende onderdelen.



In *Concepten en voorbeelden* wordt de communicatie tussen zender en ontvanger schematisch en dynamisch in beeld gebracht. Ook wordt de aandacht gevestigd op de functionaliteit van Spraak in Windows XP. Met Spraak kunnen commando's in gesproken taal worden omgezet.

De werking

Bij de werking wordt uiteengezet hoe de fysieke eigenschappen van geluid zich verhouden tot het menselijk spraakkanaal. Begrippen als frequentie en intensiteit worden toegelicht (o.a. aan de hand van Flash-filmpjes die geluidsgolven met wisselende intensiteit en frequentie laten zien en horen).



In de rechtermarge van de website worden de begrippen kort omschreven. Ook wordt kort ingegaan op compressietechnieken. In de context van het menselijk spraakkanaal worden begrippen als stemloos en stemhebbend besproken.

Om duidelijk te maken hoe de verschillende generaties spraaksynthesesystemen zijn ontwikkeld, wordt een parallel getrokken met de manier waarop kinderen leren lezen. Letters worden omgezet in klanken, en klanken worden aaneengeregeng tot woorden.

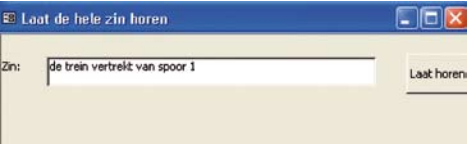
Op deze manier worden drie generaties systemen besproken en toegelicht: Grapheme-To-Phoneme, Difoonspraak en Unit-selectie. Er wordt een link gelegd naar demoversies van twee spraaksystemen. Bij het maken van de module maakte één van de

systemen gebruik van difoonspraak: de aaneenrijging van tweeklanken, zodat overgangen tussen afzonderlijke klanken natuurlijk klinken. Inmiddels maken beide systemen gebruik van Unit-selectie, zodat een verschil tussen deze twee technieken helaas niet meer waarneembaar is via de gegevens links. Bij Unit-selectie worden bestaande fragmenten gehaald uit een database, en aan elkaar geplakt tot de gewenste uiting. In de module wordt het aan elkaar plakken van woorden in MS Access ook gezien als Unit-selectie.

Het begrip foneem wordt hier geïntroduceerd: een betekenisonderscheidende klank. Een foneem wordt opgenomen in het klanksysteem van een taal als twee woorden die onderling alleen verschillen in deze klank, in deze taal ook in betekenis verschillen.

Fonemen hebben hun eigen notatiesysteem. Klinkers en medeklinkers hebben hun eigen wijze van indeling, afhankelijk van de wijze waarop de verschillende articulatoren worden gebruikt.

Als laatste component wordt de aandacht gevestigd op de bijgeleverde database. Daarin zijn afzonderlijke woorden opgenomen die in een bepaalde volgorde een omroepbericht van de NS zouden kunnen imiteren. Er zijn voorbeeldfilmpjes die laten zien hoe de formulieren in MS Access kunnen worden gemaakt. Met de formulieren kunnen de afzonderlijke woorden worden afgespeeld.



Speciale functies

Onder Speciale functies wordt het meest opvallende verschil tussen een neutrale mededeling, een vraag en een boze uitspraak uiteengezet, met de opdracht om de database op dit punt uit te breiden.

De stationsomroepinstallatie

Eindopdracht

De laatste onderdelen behelzen nadere uitleg van de stationsomroepinstallatie en de eindopdracht van de module. Als voorbeeld van de opdracht volgt een klein stukje code waarmee een zin uit de database kan worden afgespeeld:

```
PositieSpatie = InStr(BeginPositie, Zin, ' ')  
Do While PositieSpatie > 0  
    Woord = Mid(Zin, BeginPositie, PositieSpatie -  
        BeginPositie)  
    DoCmd.FindRecord Woord  
    GeluidsfragmentAfspele  
    BeginPositie = PositieSpatie + 1  
    PositieSpatie = InStr(BeginPositie, Zin, ' ')  
Loop  
Woord = Mid(Zin, BeginPositie, Len(Zin) - BeginPositie + 1)
```

In de eindopdracht wordt de leerlingen opgedragen zelf een spraaksysteem op te bouwen. Daartoe moet een database worden gevuld met opnamen van losse woorden. Daarna moet de code voor de formulieren worden geschreven of aangepast.

UITWERKING VAN DE MODULE EN TOEVOEGINGEN

In de presentatie op het NIOC zijn enkele vragen en opdrachten nader toegelicht. Daarbij zijn geluidsfragmenten ten gehore gebracht als uitwerking van de volgende opdrachten:

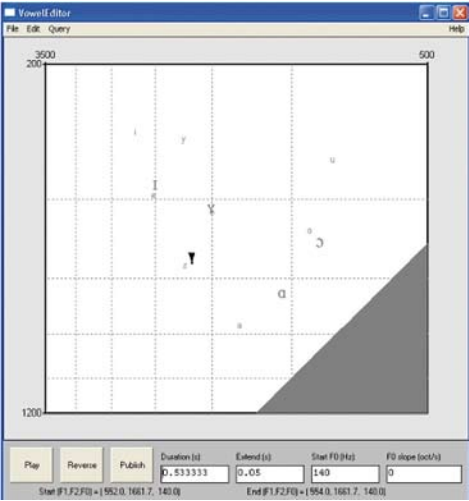
- Laat een zin horen in één van de spraaksynthesesystemen uit deze paragraaf
- Voer ook een woord in dat niet bestaat in de betreffende taal en luister hoe het voorbeeldsysteem dit woord uitsprekt
- Voer in een Nederlandstalig systeem de zinnen 'Vandaag regent het' en 'De regent vervangt de koning' in. Wordt het woord 'regent' op verschillende manieren door het systeem uitgesproken?

De stof die wordt behandeld in de module is vrij summier. Daarom is een aantal suggesties gedaan ter aanvulling:

- Leg een relatie tussen de fysieke eigenschappen van geluid en spraakklanken. In de module wordt alleen gesproken over geluidsgolven met regelmatige herhalingen, waaraan een toonhoogte kan worden toegekend. Laat zien en horen dat sommige medeklinkers uit (gemoduleerde) ruis bestaan.
- Illustreer het begrip foneem met de zogenaamde klinkerdriehoek. Laat zien hoe foneemstelsel van verschillende talen zich tot elkaar verhouden. Bespreek bijvoorbeeld het verschil tussen het Engelse bed (/bed/) en bad (/bæd/) enerzijds en de plaats van de Nederlandse klinker in het Nederlandse woord bed anderzijds.
- Gebruik een bestaand systeem als Praat van de Universiteit van Amsterdam om de invloed van de articulatoren op spraak te demonstreren. Met dit programma kan een klank worden geanalyseerd in termen van een grondtoon en de zogenaamde formanten. Het samenspel van formanten kan hoorbaar worden gemaakt door te klikken op een bepaalde plaats in de klinkerdriehoek (zie afbeelding).

Zo kan het begrip foneem verder worden toegelicht.

- Maak gebruik van bestaande mogelijkheden op internet. Er zijn goede definities en audiobestanden te vinden van de begrippen die in de module worden behandeld. Daartoe behoren ook voorbeelden van difoonspraak.



VERANTWOORDING

De module is een initiatief van de Studio Taalwetenschap en is ontwikkeld in samenwerking met de SLO, de Hogeschool van Arnhem en Nijmegen en de Universiteit Twente. De projectgroep bestond uit Ton Ammerlaan, Jos Canton, Karijn Helsloot, Arjan van Hessen, de hoofdauteur van de module, Victor Schmidt, en de auteur van deze bijdrage.

De module is te vinden op
http://www.slo.nl/voortgezet/tweedefase/vakken/informatica/taalenspraak/bestanden_taal

Het programma Praat is te vinden op
<http://www.fon.hum.uva.nl/praat/>



Instructional design for Java enterprise component technology

extended abstract



Marco Marcellis - Open University of the Netherlands,
Faculty of Informatics
m.marcellis@telfort.nl

Ella Roubtsova - Open University of the Netherlands,
Faculty of Informatics
ella.roubtsova@ou.nl

Bert Hoogveld - Open University of the Netherlands,
Faculty of Informatics
bert.hoogveld@ou.nl

INTRODUCTION

Nowadays, almost any business role needs an enterprise application and the growing request for creating of such applications is supported by component technologies. Those technologies provide typical services used by enterprise applications and prefabricated component types for application design. For example, the Java EE technology offers several related component models, like the EJB 3.0 [1] model for the server side components with the corresponding application servers, and the servlet technology [2] for the server side components. There are also Integrated Development Environments (IDE) like NetBeans [3] providing the infrastructure and wizards for custom deployment of components from given component types.

INTERRELATED LAYERS

The process of development of an enterprise application consists of steps for creating three closely interrelated layers of an enterprise

application, namely, a layer connecting the system with the database, a layer that presents the business logic of an enterprise and the client layer (Figure 1). Filling in those layers with the components and the glue code, so that the components communicate and their composition as a whole has the required functionality, is the task of the developer. Although IDEs offer automation for design of different layers, the choice of necessary components and the glue code is to be done by the designer who should take into account the requirements to the complete enterprise system, use his knowledge of the component model and interface based component paradigm. This process of choices may become difficult because the available tutorials in this area usually focus on parts of the whole process, train isolated skills and do not help to reach integrated objectives. The situation when partial knowledge does not help in solving the complete task is usually called the 'transfer paradox' [4].

4C/ID-MODEL

From the teaching perspective the component based development of an enterprise application is considered as a complex task that has to be taught as a whole. One of the most successful approaches for development of effective learning material for complex tasks is the approach called Four Component Instructional Design model (for short, 4C/ID-model) [5]. The 4D/ID approach proposes to choose the criteria and classify a set of complete complex learning tasks of different complexity. Each learning task can have several exercises. Some of the exercises have to

be used as demonstration, other - as partial demonstration and should be finished by the students. The final exercises should be fulfilled by the students independently.

TASK CLASSES

The criteria of separation of learning tasks depend on the domain of complex tasks. For the development of enterprise applications the requirements to the user access and to the back-end systems serve as a natural means for the choice the learning tasks. We have fixed the back-end system and separated the task classes based on the requirements to the user access: a web-browser and an application client (Figure 2).

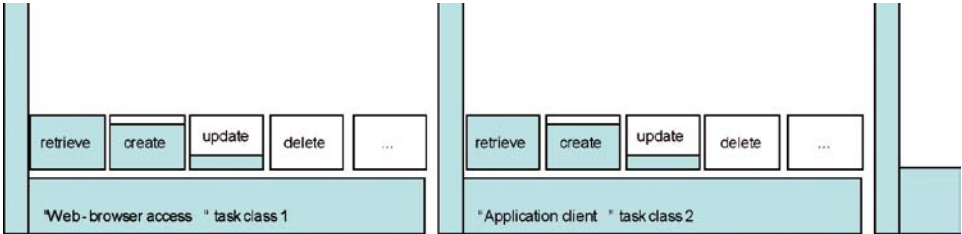


Figure 2. Learning task classes and exercises

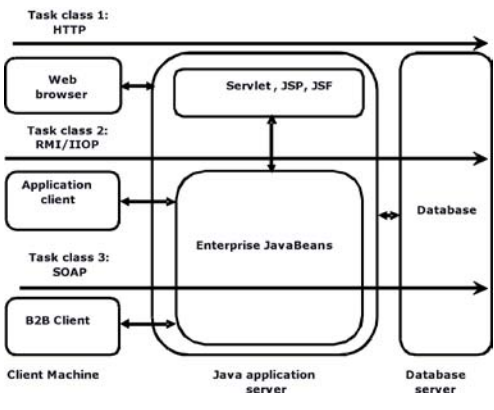


Figure 1. Layers of enterprise applications and learning task classes

Inside of each of those task classes the requirements are decomposed into 'create', 'retrieve', 'update' and 'remove' groups of functionality. Each of these functionalities can be seen as a simple enterprise application. Each of the functionalities can be of a different level of complexity and may be implemented with local and remote clients and different types of Enterprise Java Beans (stateless and stateful session beans or message beans). The complexity of the business logic component and the number of middleware services and classes used for connection with the database grows if the implementation goes from the 'retrieve' to 'update', 'create' and 'delete' functionality. So, the implementations of standard functionality are the natural candidates for the exercises within one task class.

The exercise for implementation of the 'retrieve' functionality is used as demonstration. It is shown as a grey box in Figure 2. The creative processes of relating requirements to the underlying business model and databases and modeling an application in terms of the component model are demonstrated to the students as supportive material. The implementation of the 'update' and 'create' is partially supported and has to be finished by students. These exercises are shown as partially grey boxes in Figure 2. The 'delete' functionality is fulfilled by the students independently. The box representing this exercise in Figure 2 is not colored. From the outside, the complexity of a learning task is measured by the complexity of the functionality provided by the enterprise as demanded. From the inside, the complexity is defined by the number of services and techniques used to implement this functionality. So, the learning of component paradigm is built into the exercises of different complexity in each of the tasks of development of an enterprise application.

RESULTS

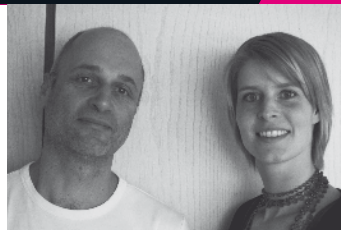
We have applied this method of instructional design to the course Component based development taught to regular and commercial students at Open University of the Netherlands. The application has shown that 70% of the students achieve the necessary skills fulfilling the three mentioned exercises and can independently implement the fourth exercise. After the training those students implement another enterprise application independently. For the 30% of students who need more training, an open electronic course may improve their training. As future work we are going to extend the set of learning tasks including the applications with business-to-business communication shown as Task 3 at Figure 1.



REFERENCES

[1] R.P. Sriganesh, G Brose, M. Silverman Mastering Enterprise JavaBeans 3.0. Wiley, 2006.
[2] 2EE. Java Servlet Technology, <http://java.sun.com/products/servlet/>.
[3] NetBeans IDE. <http://www.netbeans.org/>.
[4] Merrill, M. D. First Principles of instruction. Educational Technology, Research and Development 50(3), pages 43-59., 2002.
[5] J.J.G.van Merriënboer, P.A. Kirchner. Ten Steps to Complex Learning. Lawrence Erlbaum Associates, Publishers, London, 2007.

Personal branding. 'Do hyperego's dream of eternity'



Judith Gaffert - Fontys Hogeschool ICT

Eric Slaats - Associate lector bij Fontys Hogeschool ICT

SAMENVATTING

Door met studenten bewust te werken aan personal branding via digitale media, ontstaan 'bewuste' hyperego's. Op deze wijze wordt het onbewust personal branden omgebogen naar bewust personal branden. Extra spin-off in dit proces is een stevige oriëntatie van de student op zijn heden en toekomst binnen het gekozen vakgebied. Bovendien levert de gekozen aanpak een aantal interessante portfolio-objecten voor de student op.

TREFWOORDEN

Hyperego, Me Media, Personal branding

INLEIDING

Vrijwel iedereen googlet zijn eigen naam wel eens. Steeds vaker zal dit leiden tot verbazing over de persoonlijke voetafdruk, de sporen die je bewust en onbewust op het web hebt achtergelaten. Ontwikkelingen in social networking en de vrijwel onbeperkte toegankelijkheid van ICT en Webtechnologie slingeren dit effect verder aan. In ons reële leven denken we bewust na over de gevolgen van ons handelen en onze presentatie. Daarnaast zijn we door onze opvoeding en sociale interactie geconditioneerd wat betreft onze gedragingen. We kleden en gedragen ons veelal op een bewuste wijze en creëren daarmee een imago. Over

het imago van onze virtuele alter ego('s) maken we ons schijnbaar veel minder zorgen.

Binnen bovenstaand kader behandelt dit artikel het fenomeen hyperego's en de persistence (onuitwisbaarheid) van data (footprint) die hiermee bewust en onbewust op het web achtergelaten worden. Vooral studenten op weg naar de start in hun carrière zou geleerd moeten worden hoe ze hyperego's kunnen inzetten ten faveure van hun imago. Voor ICT&Media studenten van Fontys Hogeschool ICT wordt in het curriculum hier speciaal aandacht aan besteed onder de noemer personal branding.

MEDIAREVOLUTIES

De mondelinge oervorm van informatieoverdracht is traag en onderhevig aan vervorming en foutintroductie. De komst van schrift verandert dit. De lezer krijgt immers precies te lezen wat de schrijver vastlegt. De boekdrukunst verbetert de verspreiding doordat verveelvoudiging van teksten eenvoudiger wordt. De transformatie van geschreven naar gedrukte tekst wordt de eerste mediarevolutie genoemd. De huidige kranten en magazines zijn de laatste ontwikkeling binnen deze eerste mediarevolutie. De opkomst van elektronische media naast de gedrukte media zorgt ervoor dat het informatieaanbod een multimediaal karakter krijgt. Beeld en geluid manifesteren zich in de media als televisie en radio. Dit wordt de tweede mediarevolutie genoemd. Samen met gedrukte media vormen ze een krachtig middel om de massa te bereiken (vandaar de term massamedia).

Een kenmerk van massamedia is dat ze een one-to-many karakter hebben en in feite gecontroleerd worden door degenen die ze exploiteren of bezitten.

De derde mediarevolutie kenmerkt zich door liberalisering van de media. Door een breed beschikbare drager (internet) en betaalbare middelen om digitale content te maken, kan iedereen naast een ontvanger ook aanbieder van informatie worden. De media-consumer maakt hiermee een transitie naar media-prosumer. Deze ontwikkeling legt voorheen exclusieve diensten, zoals het uitzenden van filmisch materiaal, in handen van het generale publiek.

TRENDS?

Voor degenen die actief zijn op internet hebben recent veel ontwikkelingen plaatsgevonden die snel heel normaal geworden zijn. Als enkele van deze ontwikkelingen op de tijdslijn van de afgelopen 10 jaar geplaatst worden, dan is het bijna schokkend om te zien in welk tempo ze hun intrede gedaan hebben.

Enkele voorbeelden:

- 1998 Google start, de eerste blogging site online.
- 2001 Wikipedia wordt opgericht.
- 2003 Second life start.
- 2004 Flickr, Hyves, Facebook zien het licht.
- 2005 YouTube komt online.

Websites zijn het belangrijkste middel geworden om content te verspreiden en de kwaliteit van zoekmachines zoals Google zorgt voor zeer toegankelijke ontsluiting. Het zoeken (of beter: het vinden) van informatie (content) is nog nooit zo eenvoudig geweest. Het maken van een site, nodig om content ter beschikking te stellen, was tien jaar geleden zeker niet voor iedereen weggelegd. Distributie van persoonlijke content was exclusief voor degenen die webtechnologie in de vingers hadden.

De komst van Flickr, een vriendelijke methode voor het publiceren van fotomateriaal, verandert dit. Hiermee is het prosumer-worden voor beeldmateriaal gefaciliteerd. Meer impact heeft de introductie van YouTube een jaar later (2005). Op dit moment serveert YouTube meer dan 100 miljoen video's per dag uit. Sites zoals YouTube en Flickr faciliteren het publiceren van content in de vorm van foto en video. Het koppelen van dit soort content aan persoonlijke informatie wordt mogelijk bij de introductie van social networking sites. Deze maken het mogelijk om een virtueel sociaal netwerk op te bouwen en om personal content te koppelen aan je digitale alter ego. Internationaal gezien zijn Facebook en MySpace de meest populaire uitingen hiervan. In Nederland zijn dat o.a. Hyves, LinkedIn en Plaxo.

Blogging wordt, ondanks dat het een relatief oud fenomeen is, in deze context enorm populair. De hoge verwachtingen die de industrie heeft van dergelijke ontwikkelingen komt tot uiting in de enorme bedragen die ervoor betaald zijn. News Corporation van Rupert Murdoch betaalt in 2005 580 miljoen dollar voor MySpace, Google betaalt in 2006 1,65 miljard dollar voor YouTube.

HYPEREGO'S

Door alle vormen van content die op persoonlijke pagina's van een sociaal netwerk site geplaatst kunnen worden, transformeren dit soort sites tot ego-etages. Dit resoneert vooral in de jeugdcultuur waar het heel normaal is je hele persoonlijke hebben en houden online te plaatsen. Dit fenomeen wordt ook wel Me-Media genoemd. Naast informatie die bewust online geplaatst wordt, creëren we vaak onbewust veel meer identiteitsbepalende elementen op het web. Hierbij is te denken aan accounts voor e-mail, betalingsverkeer, chat, gaming, webcambroadcasts en online shops. Onze 'digitale ik' kent dus een flink aantal verschillende componenten. De wijze waarop een



Volle aandacht voor een keynote-spreker.

entiteit (persoon, bedrijf) zich via deze uitingen op het web profileert, wordt ook wel een hyperego genoemd. Het aantal componenten dat betrokken is bij de vorming van hyperego's neemt nog steeds toe. Een uitbreiding, waarvan de contouren nu zichtbaar beginnen te worden, is het 'internet der dingen' (spime). Hierbij valt de grens tussen fysiek en virtueel verder weg doordat objecten ook een virtueel alter ego hebben en verbonden worden met andere digitale entiteiten. De virtuele identiteit van een persoon zal zich dan uitbreiden met de virtuele alterego's van fysieke objecten waarmee die persoon in het reële leven een verband heeft. Het gaat dan bijvoorbeeld niet alleen om een foto of een filmpje van je auto, huis, gitaar e.d., maar tevens om hoe dat object zich gedraagt in het reële leven gekoppeld aan een digitale identiteit.

Ontwikkelingen rondom hyperego's zorgen voor een nieuwe reeks vraagstukken. In hoeverre kunnen we invloed uitoefenen op ons hyperego? Hoe persistent zijn de data die we achterlaten en hoe groot is deze footprint en wie heeft toegang tot onze data? Er zijn internetdiensten die het web afzoeken en alle informatie gekoppeld aan een persoon bij elkaar zoeken en presenteren. Om na te gaan welke informatie over jezelf te vinden is, kan een check uitgevoerd worden via diensten die bijvoorbeeld te vinden zijn op www.pipl.com of www.wieowie.nl. Voor een aantal studenten die deze actie uitvoerden leverde dit een wat ontvullende reality check op.

Wat betekent dit? In relatie tot gedrag, dat vooral jongeren (studenten) vertonen, is het antwoord in feite relatief simpel. Ben je bewust van je hyperego! Geconstateerd kan worden dat in een aantal gevallen de digitale barrière een schijnbaar anonimiteitschild biedt dat ander sociaal gedrag uitlokt. Realisatie dat anonimiteit eigenlijk niet bestaat zou een eerste stap kunnen zijn.

Een tweede interessante les is dat data die je publiceert aanwezig (persistent!) blijft. Je laat een enorm digitaal spoor na als je op internet actief bent. Geruchten dat Google elke zoekactie onthoudt en zo gebruikersprofielen opstelt, zijn hardnekkig. Zoekgedrag is hiermee in feite onderdeel van een hyperego. Een ander voorbeeld is dat het verwijderen van Facebookprofielen onmogelijk lijkt te worden. Ondanks het feit dat Facebook dit ontkent, is een aangemaakt profiel blijvend. Het verwijderen van profieldata geeft niet de garantie dat deze data ook echt verdwenen is. Op content die gekopieerd is en elders door anderen gebruikt wordt, kan nauwelijks meer invloed uitgeoefend worden.

Gezien het belang van hyperego's in ons dagelijkse en professionele leven, lijkt het nodig dat er bewust mee omgegaan wordt. In deze zin moeten hyperego's een onderdeel vormen van onze educatie en zijn ze een vast onderdeel geworden van onze personal branding.

PERSONAL BRANDING EN HYPEREGO'S

Personal branding is in Nederland een vrij jong fenomeen. Het is een begrip dat in Amerika aan het einde van de jaren 90 beschreven is door de visionair Tom Peters (www.tompeters.com). Personal branding betekent over jezelf denken als een merk, inclusief toekomstbeelden, onderscheidend vermogen, kernwaarden etc.

Het belang van personal branding ligt deels ten grondslag aan trends zoals feminisering en individualisering. Feminisering vertaalt zich naar de behoefte van de maatschappij aan warmte, emotie en meer dan alleen harde feiten. Als we dit koppelen aan personal branding dan betekent dit in professionele sfeer dat we tegenwoordig meer moeten laten zien dan alleen vooropleiding met bijbehorende vakken. Bedrijven willen zien wie de persoon achter de naam is. Kortom: wat zijn persoonlijke eigenschappen en kwaliteiten? Naast feminisering is ook individualisering een indicator voor personal branding. Je onderscheiden van de rest om boven het maaiveld uit te komen wordt ook verwacht. Er worden geen meelopers meer geaccepteerd.

Daarnaast is een van de drijvende krachten achter personal branding de enorme hoeveelheid producten en diensten die appelleren aan personalisering en co-creation. Door bepaalde keuzes/combinaties te maken kun je aangeven wie je bent en waar je voor staat. Het is duidelijk de tijd van customizen en co-creation i.p.v. massaconsumptie.

Als personal branden gekoppeld wordt aan nieuwe digitale media, dan sluit dit naadloos aan op het verschijnsel hyperego. Deze vaak onbewuste manier van personal branden wordt steeds belangrijker gezien het feit dat real life zich meer en meer verplaatst naar het virtuele. De bewustwording hiervan is laag, terwijl de gevolgen hiervan groot en nog niet te overzien zijn gezien de onuitwisbaarheid van data. Welke werkgever googlet een potentiële werknemer niet even, of bekijkt een Hyvesprofiel? De scheidslijn tussen privé en zakelijk is hiermee feitelijk weggefallen, je bent permanent zichtbaar geworden. Je hyperego is dus een zeer effectief onderdeel van je personal brand geworden.

PERSONAL BRANDING IN HET ONDERWIJS

Personal branding is een proces waaraan binnen Fontys Hogeschool ICT bewust aandacht wordt besteed wordt. P-fasestudenten die nu de opleiding starten, zijn gezien de snelle opkomst van de Me-Media al een paar jaar aan hun digitale alterego aan het bouwen. Door de recentheid van dit fenomeen is het een effect dat met name voor de jongere groep studenten geldt. Sinds 2 jaar heeft personal branding bij het Mediaprofiel van Fontys Hogeschool ICT aandacht in het curriculum in de vorm van 'het merk IK'. Studenten worden geprikkeld na te denken over wie zij zijn, wat hun kwaliteiten zijn en wat hen beter of uniek maakt ten opzichte van andere studenten. Zoals besproken komt dit enerzijds voort uit maatschappelijke trends en behoeften; anderzijds hangt de ontwikkeling van personal branding nauw samen met het competentiegericht leren.

Competentiegericht leren is gericht op het vergroten van de persoonlijke vermogens. Een van de uitgangspunten is dat mede uitgegaan wordt van de vermogens waar iemand al over beschikt en die hij verder kan ontwikkelen. Personal branden richt zich mede op het herkennen van de eigen capaciteiten en deze verder ontwikkelen om daarmee jezelf in positieve zin te onderscheiden. In die zin draagt Personal branden mede bij aan de competentiebewustheid van studenten.

Op het moment van afstuderen moet de student een goed beeld moeten hebben van zijn of haar kunnen met betrekking tot de vastgestelde competenties. In de praktijk blijkt echter dat studenten de koppeling tussen onderwijs en competenties een lastig proces vinden. Door personal branding in het onderwijs te integreren, wordt de overgang van het volgen van vakken naar het kijken naar eigen kwaliteiten en competenties vergemakkelijkt. Studenten krijgen meer zicht op eigen kwaliteiten en brengen deze bewust naar buiten. Bij het ontwikkelen van personal branden binnen

FHICT is gezocht naar een onderwijsvorm die dit bovengenoemde effect optimaliseert. Een van de eisen hierbij is dat studenten hun personal brand moeten visualiseren om het voor hen zo concreet mogelijk te maken. Daarnaast is het in deze vorm direct mogelijk het te communiceren via de voor hen gebruikelijke kanalen. Uiteindelijk zijn intensieve 'personal-brandinglessen' gecombineerd met twee tastbare eindproducten die de studenten moeten realiseren: een personal magazine (glossy) en een portfoliosite. In beide producten moet het personal brand duidelijk zichtbaar zijn en herkenbaar in stijl en vorm. Daarnaast leren studenten hun personal brand te verweven in hun hyperego. Hiermee wordt onbewust online gedrag omgezet in een bewust proces waarbij wordt gekeken of je zichtbaarheid op internet overeenkomt met je personal brand.

CONCLUSIE

We leven in exponentiële tijden! Ontwikkelingen in ICT lijken zich nog steeds via de wet van Moore te ontwikkelen. Deze 'wet', die Gordon Moore (Mede oprichter Intel) in 1965 op basis van observatie vastlegde, beschrijft een trend voor de ontwikkeling computerchips. Grofweg komt deze trend erop neer dat de capaciteit elke 18 maanden verdubbelt. Dat betekent dat ook de mogelijkheden exponentieel blijven groeien. Hyperego's zullen dus ongetwijfeld een toenemende complexiteit krijgen. Dit biedt kansen en bedreigingen. Het is zaak onze jonge professionals zo op te leiden dat ze de kansen zien en pakken, en zich zeer bewust zijn van de bedreigingen en die ook weten te ontlopen.

De eerste generatie studenten die in de propedeusefase met (digitaal) personal branding geconfronteerd is gaat nu op stage. Het is nog te vroeg om te beoordelen of het bewustwordingsproces omtrent hun hyperego effect zal hebben op hun latere professionele leven. Het is wel al zichtbaar dat studenten gebruik maken van hun digitale alter ego bij het verwerven van stage plaatsen. Spannend is het voor deze groep om nu al na te denken hoe deze middelen in te zetten bij het afstuderen en het vinden van een eerste baan.



Europese samenwerking in het hoger ict-onderwijs

Erasmusproject Intensive Program - ISTAR-DOT

Boudewijn Bouwense - hogeschooldocent
Bedrijfskundige informatica, Hogeschool Utrecht

SAMENVATTING

- De EU kent subsidie toe aan initiatieven die
- de Europese activiteiten van hogeronderwijsinstellingen ondersteunen en
 - de mobiliteit en uitwisseling van studenten, docenten en stafleden bevorderen.

De opleiding Bedrijfskundige Informatica van Hogeschool Utrecht neemt deel aan zo'n initiatief. Onder de naam ISTAR-DOT nemen studenten en docenten uit zeven verschillende EU-landen deel aan een gezamenlijk project. Naast Nederland zijn dat Portugal, Hongarije, Finland, Cyprus, Letland en België.

ISTAR-DOT heeft een looptijd van drie jaar, met als beoogd resultaat het opleveren van een business game. In maart 2008 vond de eerste werksessie plaats, in Geel, België. Op basis van bestaande business games zijn requirements vastgesteld voor het ontwikkelen van een serious game: het exploiteren van een vliegmaatschappij. Begin april 2009 is de tweede sessie in Lissabon afgerond. Hier zijn voornamelijk grafische onderdelen ontworpen en gebouwd. Eind maart 2010 zal in Utrecht de functionaliteit gerealiseerd moeten worden. Belangrijker dan het product als deliverable, is de realisatie van internationale samenwerking tussen Noord-, Zuid- en Oost-Europese studenten.

TREFWOORDEN

Intensive Program, Erasmus Mundus, Internationalisering, Probleem Gestuurd Onderwijs, Onderwijsvernieuwing

Inleiding

Studenten van de opleiding Bedrijfskundige Informatica van Hogeschool Utrecht nemen nu voor het derde jaar deel aan een Intensive Program (IP) onder het Europees Erasmusprogramma, met de naam ISTAR-DOT.

Een Intensive Program (IP)⁷ is een kort studieprogramma (variërend van 2 weken of 10 opeenvolgende volle dagen tot 6 weken of 30 werkdagen). Studenten en medewerkers van instellingen voor hoger onderwijs uit ten minste drie deelnemende landen werken samen om:

- efficiënt en internationaal onderwijs te stimuleren met betrekking tot gespecialiseerde thema's die anders niet of slechts bij een zeer beperkt aantal hogescholen aan bod zouden komen;
- studenten en docenten in staat te stellen om in multiculturele groepen samen te werken en op die manier te profiteren van speciale leer- en lesomstandigheden. In de afzonderlijke instellingen is deze context niet beschikbaar, de blik ten aanzien van het studieonderwerp wordt hierdoor verruimd;

⁷ bron: Nuffic

- onderwijzend personeel de mogelijkheid te bieden om met buitenlandse collega's van gedachten te wisselen over vakinhoudelijke kwesties, de curriculumontwikkeling en om nieuwe onderwijsmethoden in een internationale onderwijsomgeving te testen.

Een IP kan een eenmalig event zijn maar kan ook een beperkt aantal jaren herhaald worden (de maximumduur van de financiering is drie opeenvolgende jaren, met indiening van een jaarlijkse aanvraag).

HET PROGRAMMA VAN ISTAR-DOT

ISTAR-DOT staat voor:

- I** = Make an Inventory of the existing business games on the market (free + some others by contacting business game companies)
- S** = Selecting a workable number (max 10)
- T** = Testing the games by playing them
- A** = Analyzing strengths and weaknesses
- R** = Requirements: gathering (new, additional) requirements (including non-functional IT related)
- D** = Developing a new game for encouraging entrepreneurship in our institutions
- O** = Optimizing and adapting to curricula
- T** = Training and integrating in curricula

In dit IP wordt over een periode van 3 jaar een business game voor informaticaopleidingen ontwikkeld. Coördinator van het programma is de Katholieke Hogeschool Kempen. Naast België en Nederland participeren hogescholen uit Portugal, Cyprus, Finland, Hongarije en Letland. In 2010 zal Hogeschool Utrecht optreden als gastheer.

ISTAR is in het jaar 2008 gelanceerd. De D voor Developing was gepland voor het tweede jaar. De invoering van het spel in de verschillende curricula van de landen moest in jaar 3 volbracht worden. Het is iets anders gelopen.

In 2008 is in België gestart met het opstellen van de requirements voor een geschikte business game op basis van een 5-tal bestaande games. Namens Utrecht hebben hieraan 10 studenten deelgenomen.

In 2009 is vanaf 23 maart tot 2 april in Lissabon, met behulp van de animatietool Blender, het grafische gedeelte ontwikkeld. Vanwege korting op de subsidie deden vorig jaar per hogeschool 4 studenten mee. Vanwege de benodigde specialistische kennis werden dat studenten van de opleiding Information Engineering (IE).

Dezelfde periode in 2010 zal in Utrecht de aandacht vooral zijn gericht op verdere invulling van de functionaliteit. Hiervoor worden in Utrecht studenten Informatica gerekruteerd. Hoewel de deelname aan het programma vanuit de opleiding Bedrijfskundige Informatica is gestart, hebben dus het tweede jaar studenten van de opleiding IE meegedaan en zal volgend jaar een bijdrage van studenten Informatica worden gevraagd.

ERVARINGEN

Zoals gezegd is het programma vorig jaar iets anders gelopen dan gepland. De ontwikkeling van de game is beperkt gebleven tot de grafische interface. De functionaliteit zal het komend jaar in Utrecht afgemaakt moeten worden. Hiervoor is een aantal oorzaken aan te wijzen. Ten eerste beoogt het programma niet alleen een samenwerking tussen studenten uit diverse Europese landen, maar ook tussen verschillende vakgebieden zoals BI, IE en I. Onder de deelnemende scholen bevinden zich enkele business schools. Dit laatste leverde het tweede jaar een probleem op. Immers de requirements waren al opgesteld. De studenten hadden 10 werkdagen de tijd om volgens die requirements een game in elkaar te zetten. En niet alleen dat, voor het ontwikkelen moesten ze zich ook bekwalen in

twee voor hen nog onbekende tools: Blender en Unity. Dit was het gevolg van de op het laatste moment genomen beslissing om af te zien van de oorspronkelijke opzet: de presentatie van de game op Second Life. De studenten hadden zich daar wel op voorbereid. Omdat hierdoor de ontwikkeling van de functionaliteit in het gedrang kwam, werd de rol van de business studenten marginaal. Dat uitte zich in een geringere betrokkenheid bij de noodzakelijke deliverables van het project. Naast deze inhoudelijke kwesties waren er ook wat organisatorische problemen. Door omstandigheden kon de coördinator van het programma deze keer niet aanwezig zijn. De samenwerking tussen de docenten verliep daardoor wat stroever dan de eerste keer. Bovendien was de ontvangende school nauwelijks voorbereid op de ontvangst. De dag begon nogal eens met het zoeken naar een lokaal. Kortom, zaken lopen niet altijd zoals gepland. Dat is niet erg, want daar leren we van. En er is ook veel goeds tot stand gebracht. De studenten hebben veel geleerd en zijn enthousiast huiswaarts gekeerd.

EVALUATIE

De doelstellingen van een Intensive Program zijn:

- internationale dimensie voor het curriculum;
- ontwikkeling van onderwijs;
- bevordering uitwisseling van docenten en studenten;
- intensivering van contacten met partners;
- verdieping van bepaalde onderwijsgebieden;
- speciaal onderwerp in breed Europees perspectief onderzoeken;
- instroom buitenlandse studenten bevorderen (vakgebied).

Zonder in detail in te gaan op bovenstaande punten, kan wel gezegd worden dat de meeste doelstellingen ruimschoots zijn behaald. Ook als het resultaat niet verder komt dan de D van DOT, heeft het programma nieuwe impulsen gegeven voor verdere ontwikkeling

van het onderwijs. Zo zal de opleiding Informatica van Hogeschool Utrecht het IP aangrijpen om het projectmatig werken een meer prominente, doordachte plaats in het curriculum te geven. Het heeft er ook al toe geleid dat studenten stage in het buitenland zijn gaan doen.

De reacties van de studenten waren positief:

‘I really liked the 2 weeks! During the day I learned some new programs to develop games in and afterwards we had a great time having fun. I think the content of the project was very interesting. I would definitely do this again if I’d get another chance.’

‘I really liked the project. My interest in informatics is increased. I also have a better perspective on working with business-students.’

‘Though we failed to develop all that was asked of us in the time limit provided, working with new software really motivated me to research and improve my skills on my own in the subject, as all the tools show a lot of potential in bringing innovation.’

AANBEVELINGEN

Het in een beperkte tijdspanne intensief samenwerken met studenten en collega’s uit andere landen met allemaal eigen gebruiken en gewoonten vraagt veel van een organiserende instelling. Daar zijn de volgende lessen uit geleerd:

- 1 Maak vooraf bindende afspraken met de deelnemende scholen over hun bijdragen zoals het verzorgen van gastcolleges en participerende docenten.
- 2 Vindt een goede balans tussen ongeorganiseerde en georganiseerde vrije tijd. Studenten moeten de mogelijkheid hebben op eigen wijze het bezochte land te verkennen. Maar ze willen ook geïnspireerd worden.
- 3 Zorg voor een speciale digitale infrastructuur. Er moet veel materiaal uitgewisseld worden. Belangrijke mededelingen moeten op tijd naar iedereen verspreid worden.



REFERENTIES

- [1] Intensive Programs 2009 2010. <http://www.nuffic.nl/nederlandse-organisaties/services/beursprogrammas/III-erasmus/intensive-programmes/intensive-programmes-2009-2010/intensive-programmes-2009-2010>.
- [2] Hoe manage ik een internationaal project? <http://www.socrates.at/survivalkit/>
- [3] Virtueel vergaderen: <http://www.hexa2.com/>

Augmented reality voor gebruik bij toerisme



Mohamed Abdelghany en Adrie van der Padt -
Hogeschool Rotterdam

Glenn Mosdall - Logica Rotterdam

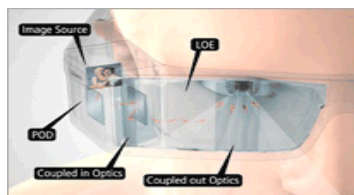
SAMENVATTING

Augmented reality is een term die al sinds begin jaren negentig bestaat. Deze term is voortgekomen uit virtual reality: augmented reality vult de realiteit aan, terwijl bij virtual reality de realiteit compleet wordt vervangen. Er zijn sinds kort brillen beschikbaar waar normaal doorheen gekeken kan worden, terwijl tegelijkertijd ook de virtuele objecten gezien kunnen worden die op deze manier de realiteit aanvullen. Er is een prototype dat een toepassing van augmented reality demonstreert in stedelijk gebied, met als doelgroep: toeristen. Met het prototype kan een toerist navigeren naar interessante punten, en eenmaal aangekomen ook informatie ontvangen over deze objecten.

INTRODUCTIE

Augmented reality is een Engelse term, en betekent letterlijk aangevulde realiteit [1]. De mens is al eeuwen bezig om de realiteit aan te vullen met 'virtuele' objecten. Zo zijn er muurbeschilderingen gevonden bij Lascaux die vermoedelijk dateren uit 15.000 voor Christus. Deze beschilderingen kunnen gezien worden als de eerste virtuele aanvulling op de realiteit gemaakt door de mens.

Aan het prototype lag een opdracht ten grondslag voor het ontwikkelen van een technologisch hulpmiddel voor toeristen in stedelijk gebied die plekken willen bezichtigen die voor hen interessant zijn. Het is dus de bedoeling om met dit project een hulpmiddel voor reizigers te maken, door het vergaren van informatie eenvoudiger te maken. Om dit te bereiken is er gebruikgemaakt van een speciale bril waar normaal doorheen gekeken kan worden, maar waarmee je ook naar virtuele objecten kunt kijken die door een computer zijn gegenereerd, zoals bij deze figuur wordt weergegeven.



Op deze manier kan een reiziger die interessante plekken bezoekt informatie ontvangen zonder zijn blik te hoeven afwenden. De informatie is dus direct zichtbaar voor de gebruiker.

Er zijn drie componenten nodig:

1 Programmeerbare computer. Dit kan elke vorm van rekenkracht zijn, variërend van een telefoon tot een mainframe.

2

Visualisatie. Hiermee wordt een systeem bedoeld waardoor de werkelijke wereld met de virtuele wereld wordt gecombineerd.

3

Tracking systeem. Om de virtuele objecten correct in de reële wereld weer te geven moet er een systeem zijn dat deze positionering bijhoudt.

Tijdens dit project worden deze drie componenten samengevoegd om een systeem te ontwikkelen dat zich richt op een toerist in stedelijk gebied. De toerist krijgt tijdens het bezichtigen van objecten die hem interessant zijn extra informatie te zien, via een beeldscherm of videobril, over het object dat wordt bekeken. Deze informatie kan de statische informatie van het object zijn (bijv. bouwjaar), maar het kan ook dynamische informatie zijn zoals afstand tot het object, of richting naar het object. Op deze manier kan het systeem de toerist ook naar de interessante objecten leiden.

TECHNISCH ONTWERP

1 Hardware

Om een prototype te realiseren is hardware nodig. De benodigde hardware kan grofweg worden ingedeeld in drie componenten:

- natuurlijk programmeerbare logica;
- een device om de plaatsbepaling af te handelen;
- een device om de augmented reality te visualiseren.

2 Programmeerbare logica

Omdat het prototype door een toerist gebruikt moet kunnen worden, moet het geheel wel draagbaar blijven. Er is voor gekozen om een laptop te gebruiken. De programmatuur kan hierdoor complexere berekeningen maken. De aansluitmogelijkheden, met oog op het device dat de augmented reality moet tonen, zijn ook wat groter.

3 Visualisatie

De wens is om een zo simpel en gebruiksvriendelijk mogelijk prototype te ontwikkelen, waarbij de toerist zijn handen zo veel mogelijk vrij kan houden. Een HMD (Head Mounted Display) met augmented reality mogelijkheden is hierdoor wenselijk. Uiteindelijk is gebleken dat de Monocular M3 van Trivision de meest geschikte bril is voor dit project. Een nadeel van deze bril is dat ze niet stereoscoop is. De virtuele objecten zullen dus voor slechts één oog zichtbaar zijn.

4 Tracking systeem

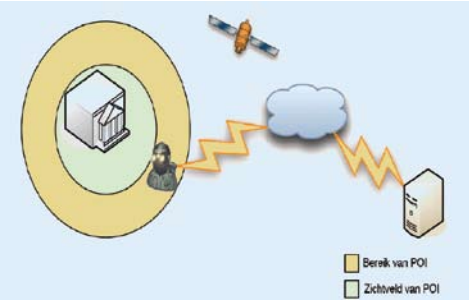
Bij dit project is het van belang om te weten waar de gebruiker zich bevindt en in welke richting er wordt gekeken. Er is een plaatsbepalingssysteem nodig dat een brede dekking heeft en dat werkt in de buitenlucht. GPS (Global Positioning System) is hiervoor uitermate geschikt. Om te weten wat de kijkrichting is van de gebruiker moet er ook een device worden gebruikt dat dit meet. Er is gekozen om een digitaal kompas te gebruiken om de kijkrichting te bepalen. Na wat onderzoek is gebleken dat er ook GPS-ontvangers bestaan met een ingebouwd digitaal kompas. In onderstaande figuur is een afbeelding te zien van de WBT-100 van Wintec.

Prototype



5 Systeemoverzicht

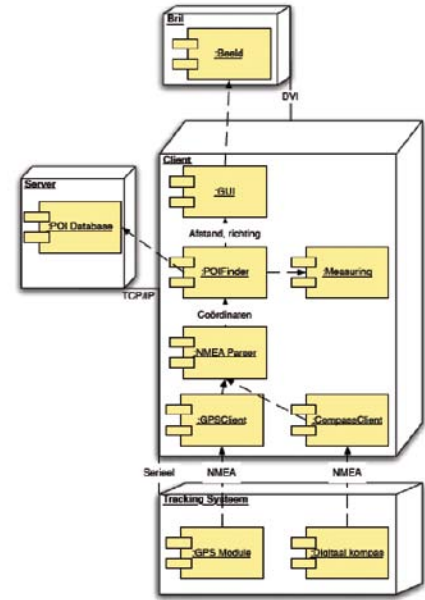
Als een toerist met dit systeem door een stad loopt dan zal het systeem de gebruiker alert kunnen maken op interessante plekken die binnen het aangegeven bereik zijn. In deze paragraaf zal een overzicht van het systeem worden gegeven op functioneel niveau.



In deze figuur is een gebruiker te zien bij een interessant object (POI). Om de interessante objecten zijn gekleurde cirkels te zien: 'Bereik van POI' en 'Zichtveld van POI'. Belangrijke objecten hebben een groter bereik dan minder belangrijke objecten. Er wordt gemeten of de gebruiker zich binnen dit bereik bevindt, door middel van GPS-locaties van de gebruiker en het object. De locatie van het object wordt door de client opgevraagd bij de server, waarna er wordt berekend wat de afstand is tot dit object, en of de gebruiker zich binnen het bereik van dit object bevindt. Als dit het geval is dan zal de gebruiker op de hoogte worden gesteld van het interessante object in de buurt. Als het de gebruiker interesseert dan kan de gebruiker een pijl volgen die de richting aanwijst van het interessante object dat in de buurt te vinden is.

6 Softwareontwerp [2]

Het ontwerp is in onderstaande figuur te zien. Onderaan het ontwerp is de GPS-module en het digitale kompas te vinden. De verbinding tussen de coördinatieapparatuur en de rest van het systeem wordt in stand gehouden door de NMEA Client in de vorm van een GPSClief en een CompassClient. Zowel de coördinaten afkomstig van de GPS-module als de richtingswaarden van het digitale kompas worden volgens het NMEA-0183-protocol verstuurd. Zo is in het ontwerp te zien dat de POIFinder toegang heeft tot de database waar alle interessante plekken in staan, en ook gebruikmaakt van een klas genaamd 'Measuring'. In deze klas staan de benodigde berekeningen om afstand en richting te bepalen [3]. Hierna wordt, afhankelijk van de instellingen, de informatie van een interessante plek getoond aan de gebruiker via een Graphical User Interface.



CONCLUSIE

Het ontwikkelde prototype is getest met een usabilitytest. De conclusie: de techniek is nog niet klaar voor gebruik de consumentenmarkt. De resultaten van het GPS vielen tegen. In de praktijk verspringt de GPS-positie nog te vaak over te grote afstanden om duidelijk een richting naar een object aan te wijzen. Dit effect was minder als de deelnemers in een open ruimte zochten naar een groot object. Ook werd er vastgesteld dat de virtuele objecten niet duidelijk zichtbaar waren in de buitenlucht, en zelfs bijna onzichtbaar werden bij blootstelling aan direct zonlicht. Daarnaast heeft ook een groot deel van de deelnemers aangegeven geen gebruik te willen maken van het huidige systeem, maar als de bril kleiner zou zijn en er geen laptop in een rugtas nodig zou zijn dan zou men wel gebruikmaken van het systeem. De deelnemers van de test hebben ook aangegeven wel geld aan een dergelijk systeem te willen uitgeven. De verwachting is echter dat de minpunten van het huidige prototype in de nabije toekomst zullen veranderen. Lumus [4], een fabrikant van augmentedreality-brillen, heeft zelfs een bril aangekondigd die voor een betaalbare prijs op de consumentenmarkt zou moeten komen. Daarnaast is een instituut van de Amerikaanse defensie (DARPA) bezig met het ontwikkelen van een lens waarmee augmented reality mogelijk gemaakt kan worden [5]. DARPA verwacht deze technologie tussen 3 en 5 jaar ontwikkeld te hebben. Daarnaast wordt binnen enkele jaren het Europese navigatiesysteem Galileo verwacht. Dit systeem zou de positie met een hogere precisie kunnen weergeven dan GPS. Hierdoor zou het probleem van de incorrecte afstand en richting verholpen kunnen worden.



REFERENTIES

[1] Stephen Cawood, Augmented Reality A Practical guide, 2008, ISBN 978-1-934356-03-6.
[2] Alan Cooper, The Inmates Are Running the Asylum, 2004, ISBN 978-0-672-32614-1.
[3] <http://mathforum.org/library/drmath/view/51879.html>
[4] <http://www.lumus-optical.com/>
[5] http://nl.wikipedia.org/wiki/Joint_Strike_Fighter-programma



STICHTING NIOC

Hans Frederik, voorzitter
 Kees van Loon, penningmeester
 Rein Smedinga, secretaris
 Henk van Leeuwen
 Jaap Zwaan
 Joke Jansen

STUURGROEP NIOC 2009

Henk Plessius, voorzitter
 Marja ter Wal, I & I
 Boudewijn Dominicus, Hogeschool Utrecht
 Marc Willemsen, ROC Midden Nederland
 Jan Kruizinga, ROC ASA
 Johan Jeuring, Universiteit Utrecht
 Karel Lemmen, Open Universiteit
 Herman Koster, Capgemini
 Hans Frederik, stichting NIOC

DIT CONGRES WORDT MEDE MOGELIJK GEMAAKT DOOR:



PROGRAMMACOMMISSIE NIOC 2009

Frans Vodegel, Hogeschool Utrecht, algemeen voorzitter

Jan Friso Groote, TU Eindhoven, voorzitter sector WO
 Mir Wermuth, Hogeschool InHolland, voorzitter sector HBO
 Wim Bos, ROC Midden Nederland, voorzitter sector VO
 Marijke Loots, St. Bonifaciuscollege Utrecht, voorzitter sector VO
 Roland Bensen, Capgemini Academy, voorzitter sector bedrijfsopleidingen
 Jos Baeten, TU Eindhoven
 Erik Barendsen, Radboud Universiteit Nijmegen
 Jan Bergstra, Universiteit van Amsterdam
 Twan Maintz, Universiteit Utrecht
 Frank Wester, Open Universiteit Heerlen
 Marga Jacobs, Hogeschool Avans
 Pascal Ravesteijn, Hogeschool Utrecht, Kenniscentrum voor procesinnovatie
 Eugène van Roden, Windesheim Zwolle
 Tanja Ubert, Hogeschool InHolland
 Harry van Vliet, Hogeschool Utrecht, Kenniscentrum Communicatie en Journalistiek
 Henk van Eijk, ROC A12
 Janus Hermse, ROC Midden Nederland
 Jan Rozema, ROC van Amsterdam
 Pieter van der Hoeven, TU Delft
 Stephan van Keulen, student lerarenopleiding TU Delft
 Johan Korten, Pieter Zandt scholengemeenschap Kampen
 Marja ter Wal-Flipse, Universiteit Utrecht
 Lyda Folkerts, SOD opleidingen
 Erik Mols, Erlen Automatisering



Colofon

- UITGAVE
Hogeschool Utrecht
- TEKST
deelnemers NIOC
- REDACTIE
Frans Vodegel en Marijke Loots
- CORRECTIE
Lisette Blankestijn Communicatie, Almere
- ONTWERP EN OPMAAK
Studio Marise Knegtmans, Amsterdam
- FOTOGRAFIE
Majid Hossainy
- DRUK
Drukkerij
- INFORMATIE
Stichting NIOC
Secretariaat
Erasmuslaan 5
3584 AZ Utrecht
T. 030 252 0315
- INTERNET
www.hu.nl, www.nioc.nl

Sorry...vuurtje

Hoe zeg je dat iemand
nummer één voor je is
als je zelf een nul bent?

Nullen en enen zeggen
evenveel als a's en o's of
piep en miauw ook sorry
- diepe blik - vuurtje?

In de ronde ogen zie je jezelf al
smullen eerst de nul daarna de één
dan ik pak jou ofwel piep en miauw

