



HOGESCHOOL ROTTERDAM

CMI

MODULEWIJZER WEM04

“Casus: Systemontwerp Webapplicatie.”

4 credit points

Module eigenaar: Media aan de Maas

Versie: 28-03-2008



A4-Modulebeschrijving

Modulecode:	WEM04
Modulenaam:	Casus: Systeemontwerp Webapplicatie.
Belasting (aantalCP):	4 CP
Relatie met ander Onderwijs	Gaat verder op WEM02 algemene kennis en is ter voorbereiding van wem05 en WEM06
Vereiste voorkennis:	Een goede basis in programmeren en webdevelopment is vereist.
Programmasoort	Kennisgestuurd expeditiemodel
Looptijd	10 weken
Vrijstelling:	In overleg
Leermiddelen:	Praktijklokaal.
Leerdoelen:	<ul style="list-style-type: none">• De student kan een casus analyseren, gestructureerd ontwerpen en ontwikkelen• De student kent de mogelijkheden en beperkingen van diverse webplatformen• De student kan zich snel een nieuw en onbekend ontwikkelplatform eigen maken.• De student kan een goed gedocumenteerd project aan anderen overdragen, opdat deze anderen snel aan de slag kunnen.• De student kan snel aan de slag in een lopend project en hier uitbreidingen of aanpassingen op maken.
Inhoud:	<ul style="list-style-type: none">• Gestructureerd systeemontwerpen• Serverside technologieën als:<ul style="list-style-type: none">○ ASP,○ JSP,○ PHP,○ RubyOnRails○ ColdFusion• Architecturen voor webapplicaties zoals:<ul style="list-style-type: none">○ Template engines○ MVC “frameworks”○ Database abstractie
Werkvormen:	± ½ uur per week: bespreking van voortgang case. ± 1½ uur per week: presentatie diverse onderwerpen klassikaal. ± 10 uur per week: zelfstandig (groepsverband) werken aan opdrachten.
Toetsing:	Oplevering (deel) cases (door docent en medestudenten) en presentaties
Auteur(s):	Michel van Gooswilligen en Mio van der Lijn
VersieDatum:	28-03-2008
Module beheerder:	CMI



Inhoudsopgave

A4-Modulebeschrijving.....	2
Inhoudsopgave.....	3
1 Algemene Omschrijving.....	4
1.1 <i>Inleiding</i>	4
1.2 <i>Leerdoelen</i>	4
1.3 <i>Inhoud</i>	5
1.4 <i>Relatie met andere onderwijseenheden</i>	6
1.5 <i>Werkvormen</i>	6
1.6 <i>Keuzeruimte</i>	7
1.7 <i>Toetsing</i>	7
2 Programma	8
2.1 <i>Inleiding</i>	8
2.2 <i>Weekschema</i>	9
3 Toetsing en beoordeling	10
3.1 <i>Procedure</i>	10
3.4 <i>Herkansingen</i>	10



1 Algemene Omschrijving

1.1 Inleiding

Bij het realiseren van een webapplicatie is de keuze voor het uiteindelijke platform grotendeels bepalend voor het verdere verloop. Vaak genoeg is de keuze meer een emotionele beslissing dan een objectieve en weloverwogen beslissing. Mensen zijn vertrouwd met een bepaald platform en hebben geen zin of durf om een ander platform in te zetten.

Door inzicht te hebben in meerdere talen en diens mogelijkheden kan door de juiste keuze echter de doorlooptijd van een project worden gehalveerd (goede keuze) of verdubbeld (slechte keuze).

Sommige platformen bieden bijvoorbeeld 'out of the box' ondersteuning voor zaken waar een andere taal in eerste instantie helemaal niet voor bedacht was. Via deze module krijg je de mogelijkheid om inzicht te verkrijgen in andere platformen wat kan bijdragen om in de toekomst een weloverwogen keuze te kunnen maken op basis van objectieve feiten.

Hoewel het mogelijk is om een taal te leren vanuit boeken en tutorials leert de ervaring dat het gericht werken aan een specifieke opdracht het beste resultaat in de kortste tijd oplevert. Op deze manier zal je direct op zoek moeten gaan naar oplossingen voor problemen die vanuit de opdracht naar voren komen waardoor je een heel andere beleving krijgt dan door voorbeelden en kleine deelopdrachten uit te werken.

1.2 Leerdoelen

- De student kan een casus analyseren, gestructureerd ontwerpen en ontwikkelen
- De student kent de mogelijkheden en beperkingen van diverse webplatformen
- De student kan zich snel een nieuw en onbekend ontwikkelplatform eigen maken.
- De student kan een goed gedocumenteerd project aan anderen overdragen, opdat deze anderen snel aan de slag kunnen.
- De student kan snel aan de slag in een lopend project en hier uitbreidingen of aanpassingen op maken.



1.3 Inhoud

Als leidraad voor de module dient het volgende scenario: “Bedrijf Wow Solutions heeft de afgelopen tien jaar vele “thin client” webapplicaties ontwikkeld voor uiteenlopende klanten. De laatste jaren komen er echter weinig nieuwe klanten bij en komen de inkomsten vooral van lopende onderhoudscontracten en aanpassingen op bestaande systemen.

De directie is van mening dat dit komt omdat alle systemen eigenlijk nog op dezelfde manier worden ontwikkeld als 10 jaar geleden. Het team heeft in de loop der jaren middels CGI alle systemen gerealiseerd door in C diverse libraries te schrijven. Klanten worden echter steeds veeleisender en doen de systemen van Wow Solutions af als ouderwets, lelijk, en niet gebruiksvriendelijk. Hoewel deze zaken aan te pakken zijn binnen de huidige structuur van vele C custom libraries heeft de directie het plan om over te stappen naar een ‘moderne’ internettaal. Voornamelijk zodat ze functionaliteit niet steeds zelf van de grond af hoeven op te bouwen en grondig moeten doortesten.

Onderzoek laten uitvoeren door de dure medewerkers van Wow Solutions is echter geen optie. Het bedrijf en de leden van het technisch team willen graag een vooronderzoek laten uitvoeren zodat ze op basis daarvan kunnen kiezen op welke nieuwe technologie ze zich moeten gaan storten.

Zodoende heeft de directie een aantal oude relaties opgevijzeld middels oude vriendschappen met personen binnen het onderwijs. Uiteindelijk is overeen gekomen dat de opleidingen GMT en CMD en hun creatieve en innovatieve denkers (de studenten) de opdracht op zich zullen nemen om dit onderzoek voor Wow Solutions uit te voeren.

Hiertoe heeft Wow Solutions een tweetal van hun bestaande systemen uitgeschreven die als leidraad (cases) zullen dienen bij het onderzoek. De studenten zullen voor beide cases een zogenaamde reference build ontwikkelen. Dat wil zeggen een versie van het systeem wat als voorbeeld kan dienen om zo snel bepaalde functionaliteit, en eigenschappen van een platform te doorgronden.“

In deze module ga je in groepsverband aan de slag met een zelf gekozen platform om één van de voorgedefinieerde case te ontwikkelen. Hierbij ben je samen met je groep verantwoordelijk om de aangedragen case te splitsen in prioriteiten en deze incrementeel te gaan ontwikkelen naar eigen inzicht. Hierbij zal je zowel rekening houden met de op te leveren functionaliteit als zaken die nodig zijn voor een goede doorgronding van het gekozen, onbekende platform.

Naast het uitwerken van de case zal je op een tweede manier verder onderzoek doen naar het gekozen platform. Tijdens de module zullen de studenten gezamenlijk met de docent criteria opstellen waarop de verschillende platformen gericht kunnen worden vergeleken. Voorbeeld hierbij is het gebruik of de mogelijkheid tot template-engines, database abstractie en/of installatie/ontwikkelkosten (licenties). Deze aspecten wordt door elk groepje onderzocht en zullen gedurende het verloop van module worden gepresenteerd in presentaties van 5 a 10 minuten, waarbij het dus van belang is de kern van het verhaal duidelijk vast te stellen om het binnen het korte tijdsbestek duidelijk te maken.



Als extra aspect om de diverse platformen te testen (alsmede jouw vaardigheden met betrekking tot documentatie opstellen, en gestructureerd werken) zal er in week 6 een overdracht plaats vinden van de cases. Er worden zoals gezegd twee cases ingebracht, waarbij een aantal groepen aan case 1 werkt, en de overige aan case 2.

In week 6 zal een oplevering plaatsvinden van het product. Vervolgens worden de cases en producten overgedragen aan de andere groepen. Deze zullen een aantal uitbreidingen en aanpassingen moeten maken op de bestaande code van de andere groep.

Een overzicht van elementen die gedurende de module aan bod kunnen komen zijn:

- Het software development process, de bijbehorende methodieken en ontwerpdocumenten
- Kennismaking met een van de volgende server-side technologieën:
 - ASP.net
 - JSP
 - PHP
 - Ruby On Rails
 - ColdFusion

De Keuze voor PHP dient goed gemotiveerd te worden. Doel is immers het kennismaken met nieuwe server-side technieken en niet het doen wat je gewend bent te doen.

- Software architecturen voor webapplicaties zoals:
 - Template engines
 - “Frameworks”
 - Database abstractie
- Geïnteresseerden kunnen kiezen voor een combinatie met rich internet technologieën als Adobe Flex, Microsoft Silverlight of JavaFX maar moeten zich dan goed realiseren dat ze een dataconnectie moeten realiseren met de server middels een webservice of een remoting protocol.

1.4 Relatie met andere onderwijseenheden

Deze module gaat verder op een deel van de kennis uit WEM02 en dient als opstap voor de eindmodulen WEM05/WEM06 waarin een goede kennis van webtechnologieën vereist is.

1.5 Werkvormen

Tijdens de lessen zal er vooral overleg zijn over de voortgang van de cases en is er ruimte om de gekozen platformen kort en bondig aan de andere groepen te presenteren. Studenten zullen het grootste gedeelte van de werkzaamheden buiten de les om moeten doen.



1.6 Keuzeruimte

Studenten kunnen aangeven met welk platform zij willen gaan werken. De eerste keuze zal zoveel mogelijk worden gehonoreerd. De docent bepaald uiteindelijk welk team met welke taal werkt, om te waarborgen dat voldoende verschillende talen en aspecten worden onderzocht.

Studenten dragen zelf onderwerpen aan voor en participeren in de presentaties tijdens de lessen.

1.7 Toetsing

Er zijn drie aspecten waarop wordt getoetst tijdens deze module:

1. De eerste oplevering van het product (week 6) en de daarbij behorende documentatie.
2. Vervolgens de tweede oplevering van het overgedragen product (week 9). Hierbij zullen tevens de studenten die aanpassingen in het opgeleverde product van week 6 hebben gemaakt het werk van de vorige groep beoordelen. Hierbij is ruimte voor de andere groep om hun werk te verdedigen (klassikaal).
3. Uiteindelijk zullen de periodieke presentaties worden meegenomen in de beoordeling. Deze presentaties worden beoordeeld met een voldoende of onvoldoende en worden gebruikt als wegingsfactor bij het gemiddelde cijfer van de beoordeling van de cases.

Voor de cases gelden de volgende criteria:

- Is het systeem voldoende opgedeeld in deelstukken om bijvoorbeeld goed aanpassingen te kunnen maken of duidelijkheid te bieden bij de algehele werking?
- Heldere verantwoording van gemaakte ontwerpkeuzes in de documentatie
- Is optimaal gebruik gemaakt van de mogelijkheden van het gekozen platform (zijn er bijvoorbeeld standaarden gebruikt voor oplossingen in plaats van zelf 'het wiel opnieuw uitgevonden te hebben').
- Is de code helder georganiseerd, leesbaar en van zinvol commentaar voorzien?
- De performance optimalisatie van je eindproduct.



2 Programma

2.1 Inleiding

De module is er niet op gericht om een specifieke taal of platform uit te leggen. Gezien het feit dat elke groep weer met een ander platform aan de slag gaat is het sowieso niet mogelijk om gericht een platform te behandelen.

Vanuit de ervaringen die je tot nu toe hebt opgedaan wordt geacht dat je in staat bent om een relatief onbekend platform in korte tijd te doorgronden en een (beginnend) product kan realiseren. Gezien de grote hoeveelheid starters documentatie die de makers van de diverse platformen aanbieden is dit zeker geen onbegonnen taak.

De docent zal hier verder uiteraard zoveel mogelijk een ondersteunende rol vervullen, maar ook kan je terecht bij je medestudenten.

Het doorgronden van de platformen zal geschieden aan de hand van de realisatie van twee cases. De studenten gaan hier in groepjes van 2 studenten aan werken. Hierdoor kan er direct gericht worden gewerkt om uit te zoeken hoe een specifiek platform werkt.

Aan het begin worden studenten in groepen gedeeld en kiezen ze een platform en een case om aan te werken. In week 6 zal er een eerste oplevering van het product plaats gaan vinden. De groep zal zelf verdelingen maken over welke functionaliteit wel of niet handig is. Het plan hiervoor dient te worden overlegd aan de docent.

Vanaf week 6 zal een groep zijn project overdragen aan een andere groep (die in principe in dezelfde taal werkt, maar hoeft niet, als je bijvoorbeeld van een uitdaging houdt). En zal de groep op zijn beurt het project van een andere groep krijgen. Vervolgens zal er in week 9 een tweede oplevering plaats vinden met extra functionaliteit (vanuit het oorspronkelijke plan).

Het product van week 6 zal door docent worden beoordeeld en in week 9 ook door de groep die het project heeft gekregen. Het product van week 9 zal ook door de docent worden beoordeeld in samenspraak met de beide groepen.

Binnen elke les is ruimte om extra les te ontvangen over specifieke onderwerpen, zoals ontwerpmethodiek, database ontwerp, etc. mits deze een les van tevoren zijn aangevraagd.

Gedurende de module zullen studenten presentaties geven over een specifiek onderwerp, toegespitst op het gekozen platform en/of software architecturen. De studenten dragen deze onderwerpen zelf aan via een klassikale discussie (op diverse moment in de periode). Studenten worden geacht minimaal aan 1 presentatie mee te werken.



2.2 Weekschema

Week	in Les	Huiswerk
1 (19)	<ul style="list-style-type: none"> • Intro cases, • Indelingen groepen, • Kiezen platformen • Vaststellen lesinhoud en presentatieonderwerpen week 3 	<ul style="list-style-type: none"> • Opstellen planning en PVE met prioritering • Oriëntatie op platform • Voorbereiding presentatie
2 (20)	<ul style="list-style-type: none"> • Les valt uit i.v.m pinksteren • Inleveren planning en PVE (eerste increment) 	<ul style="list-style-type: none"> • Uitwerken case • Voorbereiding presentatie
3 (21)	<ul style="list-style-type: none"> • Studenten presentaties • Vaststellen lesinhoud en presentatieonderwerpen volgende les • Overleg en voortgang 	<ul style="list-style-type: none"> • Uitwerken case • Voorbereiding presentatie
4 (22)	<ul style="list-style-type: none"> • Studenten presentaties • Vaststellen lesinhoud en presentatieonderwerpen volgende les • Overleg en voortgang 	<ul style="list-style-type: none"> • Uitwerken case • Voorbereiding presentatie
5 (23)	<ul style="list-style-type: none"> • Studenten presentaties • Vaststellen lesinhoud en presentatieonderwerpen week 7 en 8 • Overleg en voortgang 	<ul style="list-style-type: none"> • Uitwerken case • Voorbereiding presentatie increment
6 (24)	<ul style="list-style-type: none"> • Oplevering increment 1 met originele PVE waarin is aangegeven wat is gerealiseerd.(naar docent) • overdracht increment 1 (naar andere team) • presentatie increment 1 (andere teams) 	<ul style="list-style-type: none"> • Uitwerken case • Voorbereiding presentatie
7 (25)	<ul style="list-style-type: none"> • Inleveren planning en PVE tweede increment (aangepaste prioritering) • Inleveren beoordeling ontvangen increment • Studenten presentaties • Overleg en voortgang 	<ul style="list-style-type: none"> • Uitwerken case
8 (26)	<ul style="list-style-type: none"> • Overleg en voortgang • Studenten presentaties 	<ul style="list-style-type: none"> • Uitwerken case • Voorbereiden eindpresentatie
9 (27)	<ul style="list-style-type: none"> • Oplevering increment 2 met (2^e) PVE waarin is aangegeven wat is gerealiseerd en reflectieverslag over beide deelprojecten (over technieken, werkwijze, overdracht, etc) (aan docent) • presentatie increment 2 (andere teams) 	
10 (28)	<ul style="list-style-type: none"> • beoordelingsgesprekken 	



3 Toetsing en beoordeling

3.1 Procedure

De opleveringen van week 6 en week 9 worden beoordeeld met een cijfer. De presentaties worden beoordeeld met een voldoende of onvoldoende.

Het eindcijfer is een weging van:

- Oplevering week 6: 65%
- Oplevering week 9: 35%
- Gegeven presentaties kan het eindcijfer positief of negatief beïnvloeden door een halve punt toe te voegen of af te trekken. (5 kan dus 5,5 worden, maar 5,5 kan ook een 5 worden)

Peer assessments worden meegewogen in het cijfer voor increment 1

Het reflectieverslag wordt meegewogen bij het tweede increment.

De 4CP worden toegekend bij een voldoende score (minimaal 5.5) van het gewogen eindcijfer.

3.4 Herkansingen

Bij onvoldoende resultaat van de praktijk opdracht worden er individuele afspraken gemaakt met de groepen om dit resultaat om te vormen tot een voldoende resultaat. Presentaties kunnen niet worden herkanst. Voor de herkansing zal gelden dat het eigen werk (week 6) zal dienen als leidraad waarop extra functionaliteit wordt verwacht.

In alle gevallen dient er rekening mee te worden gehouden dat de beoordelingscriteria bij herkansing zwaarder zullen zijn, dan wanneer de opdrachten binnen het beoogde tijdspad worden afgerond.