



**HOGESCHOOL ROTTERDAM**  
**CMI**

**MODULEWIJZER WEM04**  
**“Casus: Systeemontwerp Webapplicatie.”**

4 studiepunten

Versie: 16-04-2009



## A4-Modulebeschrijving

<b>Modulecode:</b>	MEDMWM01-4
<b>Modulenaam:</b>	Casus: Systeemontwerp Webapplicatie.
<b>Belasting (aantalCP):</b>	4 studiepunten
<b>Relatie met ander Onderwijs</b>	Gaat verder op WEM02 algemene kennis en is ter voorbereiding van wem05 en WEM06
<b>Vereiste voorkennis:</b>	Een goede basis in programmeren en webdevelopment is vereist.
<b>Programmasoort</b>	Kennisgestuurd expeditiemodel
<b>Looptijd</b>	10 weken
<b>Vrijstelling:</b>	In overleg
<b>Leermiddelen:</b>	Praktijklokaal.
<b>Leerdoelen:</b>	<ul style="list-style-type: none"><li>• De student kan een casus analyseren, ontwerpen en ontwikkelen</li><li>• De student kent de mogelijkheden en beperkingen van diverse webplatformen</li><li>• De student kan zich snel een nieuw en onbekend ontwikkelplatform eigen maken.</li><li>• De student kan werkzaamheden en ontwerpbeslissingen goed documenteren en daarmee een overdraagbaar product opleveren.</li><li>• De student kan ontwerpen en implementaties daarvan gestructureerd testen en daarmee de ontwikkelaars en opdrachtgever van bruikbare feedback voorzien om de kwaliteit van het product te verbeteren.</li></ul>
<b>Inhoud:</b>	<ul style="list-style-type: none"><li>• User centered systeemontwerp</li><li>• Serverside scripttalen als:<ul style="list-style-type: none"><li>◦ ASP.net,</li><li>◦ (oo)PHP</li></ul></li><li>• Frameworks als:<ul style="list-style-type: none"><li>◦ .NET MVC</li><li>◦ Yii en of Zend</li></ul></li><li>• Gestructureerd testen</li></ul>
<b>Werkvormen:</b>	± ½ uur per week: bespreking van voortgang case. ± 2 uur per week: presentatie en workshops diverse onderwerpen klassikaal. ± 8 uur per week: zelfstandig (groepsverband) werken aan opdrachten.
<b>Toetsing:</b>	Oplevering (deel) cases (door docent en medestudenten) en presentaties
<b>Auteur(s):</b>	Mio van der Lijn
<b>VersieDatum:</b>	16-04-2009
<b>Module beheerder:</b>	Rimmert Zelle en Mio van der Lijn



## Inhoudsopgave

<b>A4-Modulebeschrijving .....</b>	<b>2</b>
<b>Inhoudsopgave .....</b>	<b>3</b>
<b>1 Algemene Omschrijving.....</b>	<b>4</b>
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> .....	6
1.7 <i>Toetsing</i> .....	7
<b>2 Programma.....</b>	<b>8</b>
2.1 <i>Inleiding</i> .....	8
2.2 <i>Weekschema</i> .....	9
<b>3 Toetsing en beoordeling .....</b>	<b>10</b>
3.1 <i>Beoordelingscriteria</i> .....	10
3.4 <i>Beoordelingsprocedure</i> .....	10
3.5 <i>Herkansingen</i> .....	11



## 1 Algemene Omschrijving

### 1.1 Inleiding

In deze module is het de bedoeling een casus uit te werken / op te lossen met behulp van een server-side scripttaal en framework, dat je (aan het begin van de module) nog niet beheerst.

De module is er niet op gericht om een specifieke taal of platform uit te leggen. Vanuit de ervaringen die je tot nu toe – o.a. in WEM01 - hebt opgedaan, wordt je in staat geacht jezelf in korte tijd een relatief onbekend platform eigen te maken en een eerste versie van een product te realiseren. Gezien de hoeveelheid documentatie die de makers van de diverse platforms aanbieden is dit zeker geen onbegonnen taak. De docent zal hierbij zoveel mogelijk een ondersteunende rol vervullen.

De focus binnen deze module is er meer op gericht op het ontwikkelproces. Hoe kan dat meer user-centered? Hoe betrek ik de opdrachtgever bij het proces en hoe werk ik doelgericht aan een beter (voldoend) product?

### 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 werkzaamheden en ontwerpbeslissingen goed documenteren en daarmee een overdraagbaar product opleveren.
- De student kan ontwerpen en implementaties daarvan gestructureerd testen en daarmee de ontwikkelaars en opdrachtgever van bruikbare feedback voorzien om de kwaliteit van het product te verbeteren.



### 1.3 Inhoud

In deze module ga je in groepsverband aan de slag met een zelf gekozen server-side platform (scripttaal + framework), om één van de voorgedefinieerde casussen te ontwikkelen. Doel is uiteraard om een product te realiseren dat zo goed mogelijk aansluit bij de wensen en behoeften van de gebruikers. Je bent samen met je ontwikkelteam verantwoordelijk voor een goede analyse en prioritering van de requirements en neemt deze 'use cases' als uitgangspunt voor je ontwerp en implementatie. (use case driven development).

Ontwerpdocumentatie dient

1. om voor jezelf (je ontwikkelteam) om na te denken over de verschillende ontwerpkeuzes,
2. om de opdrachtgever (en het testteam) in het ontwerpproces te betrekken (zodat deze vroegtijdig feedback kan geven),
3. om de beheerafdeling van de opdrachtgever (of anderen die aan dit project verder gaan werken) inzicht te geven in de applicatie en de gemaakte ontwerpkeuzes (documentatie)

In deze module werkt ieder in twee teams tegelijk: een ontwikkelteam en een testteam. Doel is uiteraard om een product te realiseren dat zo goed mogelijk aansluit bij de wensen en behoeften van de gebruikers. Je bent samen met je testteam verantwoordelijk het geven van goede feedback op de ontwerpdocumentatie en de implementatie van het ontwikkelteam en neemt deze 'test scripts' als uitgangspunt voor de implementatie. (test driven development).

Testdocumentatie dient

1. om het ontwikkelteam feedback te geven op ontwerpdocumenten en handvaten te geven om kwalitatief goede software te schrijven,
2. om de opdrachtgever inzicht te geven in de kwaliteit van de applicatie en te helpen prioriteiten te stellen in requirements voor volgende increments

Om de casus uit te werken zal je verder onderzoek doen naar het gekozen platform en de mogelijkheden die dit platform biedt voor jouw casus. Tijdens de module zullen de studenten samen met de docent een lijst van onderwerpen opstellen waarop de verschillende platformen kunnen worden onderzocht. Om de onderzochte aspecten ook voor de andere teams inzichtelijk te maken, zullen de verschillende teams de resultaten van hun onderzoek presenteren in een korte workshop, waarbij het dus van belang is de kern van de zaak duidelijk vast te stellen om het binnen het korte tijdsbestek over te dragen.

Een overzicht van onderwerpen die minimaal gedurende de module aan bod moeten komen zijn:

- de implementatie van architectuur design patterns (zoals MVC) in het gekozen platform (taal + framework)



- faciliteiten voor authenticatie en autorisatie in het gekozen platform (taal + framework)
- faciliteiten voor database abstractie (zoals ORM) in het gekozen platform (taal + framework)
- faciliteiten voor (unit) testing in het gekozen platform (taal + framework)

Daarnaast zijn eigen onderwerpen zeer welkom. Je zou kunnen denken aan:

- het werken met template( engine)s
- het bieden van een webservice voor een client side applicatie
- het gebruik van componenten
  - voor externe communicatie (Mail, sms, etc)
  - voor het gebruik van webservices
  - etc.
- internationalisering
- Faciliteiten om te ontwikkelen volgens een bepaald software development proces (agile, XP, RUP) of bepaalde onderdelen daarvan te ondersteunen

De volgende server-side platformen kunnen worden gekozen om in te ontwikkelen:

- ASP.NET MVC
- PHP met het Yii en/of Zend framework

Andere server-side technologieën (zoals JSP, Ruby on Rails of ColdFusion) zijn alleen mogelijk als minimaal twee teams zich op dit platform willen richten.

#### **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 zijn er verschillende presentaties en workshops. Ook kan er tijd worden vrijgemaakt om de voortgang en de deliverables van de projectteams te bespreken met de docent. 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..

Studenten kiezen zelf over welk onderwerp zij een workshop zouden willen geven tijdens de lessen en kunnen hier ook onderwerpen voor aandragen. De docent bepaald uiteindelijk welk team welke onderwerp uitwerkt, om te waarborgen dat verschillende onderwerpen aan bod komen.



## 1.7 Toetsing

De volgende deliverables zullen worden getoetst tijdens deze module:

1. De ontwerpdocumentatie
2. De testscripts en het testrapport
3. De gegeven workshops
4. Het opgeleverde product



## 2 Programma

### 2.1 Inleiding

De lesinhoud is gericht op het ondersteunen van de verschillende teams tijdens hun werk aan de casussen. Tijdens deze module werk je met twee teams aan een project: een ontwikkelteam en een testteam. Tegelijkertijd werkt ieder team ook aan twee projecten mee: bij het ene project als ontwikkelteam, bij het andere als testteam.

Om te voorkomen dat je twee platforms tegelijk moet leren verdient het de voorkeur als de beide projecten waar je aan werkt hetzelfde platform gebruiken. Om te zorgen dat het testteam echt onafhankelijk is van het ontwikkelteam en beide projecten niet verwart, verdient het de voorkeur als beide projecten zich richten op een andere casus.

Teams bestaan uit drie personen. Doel hierbij is efficiënt samen te werken en van elkaar te leren. Ieder teamlid dient gelijkelijk verantwoordelijk te zijn voor alle deliverables; als een teamlid uit zou vallen zou zijn/haar aandeel door de overblijver(s) zonder problemen moeten kunnen worden voortgezet.

In het begin van de module zal het theoretische gedeelte van de les gevuld worden met presentaties van docenten over onderwerpen die betrekking hebben op systeemontwerp in het algemeen.

In de loop van de module zal meer tijd worden vrijgemaakt voor workshops van studenten over specifieke implementatiekwesties op het gekozen platform of framework. Studenten worden geacht minimaal aan 1 presentatie mee te werken.





## 2.2 Wekschema

Week	Les	Development team	Test Team
<b>10 (Q3)</b> <b>(18)</b>	<ul style="list-style-type: none"> <li>• Intro cases,</li> <li>• Indelingen groepen,</li> <li>• Kiezen platformen</li> <li>• Les valt uit ivm 4mei</li> </ul>		
<b>1</b> <b>(19)</b>	Geen les ivm 4 mei		
<b>2</b> <b>(20)</b>	<ul style="list-style-type: none"> <li>• visie en 'use case driven development'</li> </ul>	<ul style="list-style-type: none"> <li>• Oplevering visiedocument</li> </ul>	
<b>3</b> <b>(21)</b>	<ul style="list-style-type: none"> <li>• Use cases uitwerken in Activities</li> </ul>	<ul style="list-style-type: none"> <li>• Oplevering Functioneel Ontwerp</li> </ul>	<ul style="list-style-type: none"> <li>• Feedback op visiedocument</li> </ul>
<b>4</b> <b>(22)</b>	<ul style="list-style-type: none"> <li>• Testing of 'test driven development'</li> </ul>	<ul style="list-style-type: none"> <li>• realiseren Proof of Concept 1</li> </ul>	<ul style="list-style-type: none"> <li>• Feedback op FO</li> <li>• Oplevering Test Scripts</li> </ul>
<b>5</b> <b>(23)</b>	Les valt uit ivm pinksteren	<ul style="list-style-type: none"> <li>• Realiseren PoC 2</li> <li>• Voorbereiding presentatie/workshop over onderwerp binnen casus en platform</li> </ul>	
<b>6</b> <b>(24)</b>	<ul style="list-style-type: none"> <li>• Domeinmodel</li> <li>• Presentaties/workshops (architectuur en design patterns)</li> </ul>	<ul style="list-style-type: none"> <li>• Realiseren PoC 3</li> </ul>	
<b>7</b> <b>(25)</b>	<ul style="list-style-type: none"> <li>• Technisch ontwerp en architecturen</li> <li>• Presentaties/workshops (database abstractie en template engines)</li> </ul>	<ul style="list-style-type: none"> <li>• Oplevering Technisch Ontwerp</li> </ul>	
<b>8</b> <b>(26)</b>	<ul style="list-style-type: none"> <li>• Presentaties/workshops (Unit Testing)</li> </ul>	<ul style="list-style-type: none"> <li>• Oplevering Unit Tests (adhv Test Scripts)</li> </ul>	<ul style="list-style-type: none"> <li>• Feedback op TO</li> </ul>
<b>9</b> <b>(27)</b> <b>10</b> <b>(28)</b>	<ul style="list-style-type: none"> <li>• Presentaties/workshops (authenticatie en autorisatie)</li> <li>• Eindpresentaties Teams</li> </ul>	<ul style="list-style-type: none"> <li>• Oplevering code en implementatieplan</li> </ul>	<ul style="list-style-type: none"> <li>• Oplevering testrapport met verbeterpunten</li> </ul>



### 3 Toetsing en beoordeling

#### 3.1 Beoordelingscriteria

De verschillende deliverables zullen volgens onderstaande weging worden getoetst op de volgende beoordelingscriteria:

1. De ontwerpdocumentatie (30%):
  - a. Is de ontwerpdocumentatie tijdig opgeleverd (aan docent en testteam)?
  - b. Zijn de relevante ontwerpkeuzes uitgewerkt?
  - c. Zijn de ontwerpbeslissingen goed onderbouwd?
  - d. Zijn de ontwerpbeslissingen helder gecommuniceerd?
  - e. Zijn standaard modellen correct toegepast?
2. De testscripts en het testrapport (20%):
  - a. Zijn de testscripts en het testrapport tijdig opgeleverd (aan docent en ontwikkelteam)?
  - b. Zijn voor relevante risico's testscripts uitgewerkt?
  - c. Zijn de testscripts bruikbaar?
  - d. Doet het testrapport goed onderbouwde uitspraken over de kwaliteit van het product?
  - e. Biedt de gegeven feedback en het testrapport bruikbare adviezen voor kwaliteitsverbetering?
3. De gegeven workshops (20%):
  - a. Is het onderwerp goed onderzocht en begrepen?
  - b. Bevat de workshop voor dit onderwerp de relevante inhoud?
  - c. Zit de workshop goed in elkaar?  
Is de workshop voor de studenten goed te volgen?
4. Het opgeleverde product (30%):
  - a. Is het product tijdig opgeleverd?
  - b. Voldoet het product in redelijke mate aan de gestelde requirements?
  - c. Is het product goed gebouwd?
    - i. Is het systeem voldoende opgedeeld in deelstukken om bijvoorbeeld goed aanpassingen te kunnen maken of duidelijkheid te bieden bij de algehele werking?
    - ii. 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').
    - iii. Is de code helder georganiseerd, leesbaar en van zinvol commentaar voorzien?
  - d. Is helder wat precies is opgeleverd en wat de klant er nog mee moet?
    - i. Voorzien van een helder opleverdocument
    - ii. Voorzien van een implementatieplan

#### 3.4 Beoordelingsprocedure

Studenten krijgen direct na het geven van een workshop te horen of deze voldoende was of niet.



Aan het einde van de module stelt de docent op basis van de bovenstaande criteria het groepscijfer vast.

Studenten beoordelen elkaars participatie in en bijdrage aan het team. Dit peerassessment gebruikt de docent samen met de eigen waarneming en indruk van de student om tot een individueel eindcijfer te komen.

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

### **3.5 Herkansingen**

Bij onvoldoende resultaat worden er met de groepen afspraken gemaakt om dit resultaat om te vormen tot een voldoende resultaat. Workshops kunnen niet na afloop van de module worden herkanst. In de meeste gevallen zal de herkansing bestaan uit het realiseren van een extra increment, waarvoor (op basis van het testrapport) extra requirements gelden.

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.