



HOGESCHOOL ROTTERDAM / CMI

**Toelichting praktijkopdracht**

# **Ontwikkeling applicatie en mondelinge toets**

**CMTPRG02-5**



**70% eindcijfer**

**Toetsvorm: mondelinge toets**

**Individueel beoordeling**

**Deadline is zondag 00:00 week 8**

Voor het praktijkdeel is het de bedoeling om een zelfgekozen webapplicatie te ontwikkelen welke qua complexiteit vergelijkbaar is met de [Photor](#) eindopdracht ([online versie](#)).

De leerdoelen worden getoetst in de vorm van een mondelinge toets. Steekproefsgewijs worden onderwerpen (zie modulewijzer programma hoofdstuk 2), die behandeld zijn in de cursus, aangehaald. Het is aan jou de taak om toe te lichten hoe dit onderwerp is teruggekomen in de praktijkopdracht en welke keuzes je hierin hebt gemaakt. Met name de onderbouwing is belangrijk tijdens het mondeling.

De feedback die volgt uit de mondelinge toets wordt door de docent in Cumlaude geplaatst, bij de inleveropdracht van de praktijkopdracht.

## Praktijkopdracht

- **Beveiliging**
  - **Inloggen:** bepaalde functionaliteit is afgeschermd voor alleen ingelogde gebruikers, maakt niet uit welke gebruiker, als je maar bent ingelogd als rol 'gebruiker'
  - **Validatie:** bij invoervelden wordt gebruik gemaakt van verplichte velden, en/of velden met controle op inhoud (denk aan een postcode of emailadres), gebruiker krijgt foutmeldingen te zien (*deze zijn NIET te omzeilen door bijv. javascript uit te zetten...*)
- **Verschillende rollen** (dus meerdere soorten gebruikers), bijvoorbeeld
  - Admin rol
  - Gebruiker rol
- **Verschillende rechten**
  - Admin gedeelte alleen toegankelijk voor ingelogde admin
  - Gebruiker heeft eigen afgeschermd deel waarin hij zijn gegevens kan aanpassen. Een andere gebruik kan hier niet bij (door er bijvoorbeeld naar te dieplinken).
- **Zoeken & Filteren**
  - **Vrije tekst:** zoekveld met vrije tekst, zoekt in meerdere kolommen (bijvoorbeeld in naam en 'bericht'), hoeft niet in het hele systeem te zoeken, mag ook voor een specifieke lijst met data (bijv. nieuwsberichten).
  - **filtermogelijkheid:** dmv dropdownlist. Bijvoorbeeld alle nieuwsberichten van categorie 1 of 2 zien
  - Bonus: combinatie van zoeken EN filteren, of filteren op meerdere categorieën
- **Beveiliging**
  - OWASP top 10
  - Weet de impact, oorzaak en oplossingen van de belangrijkste beveiligingsrisico's en kan ze uitleggen binnen je framework/project
- **OTAP**
  - Weet wat komt kijken bij deployment naar een live/productie omgeving
  - Snap waar al je configuraties binnen jouw framework staan
  - Kan uitleggen hoe en wat moet worden aangepast voor een 'publish' naar productie
  - Bonus: Heb de O & P uit de OTAP verwerkt in je project



- **Schakelen van status met button in lijst**

bijvoorbeeld aan/uit, actief/niet actief

- MOET via een post
- MOET naar aparte action in een controller
- mag ook via Ajax

Alle nieuwsberichten

Nieuwsbericht 1

|                                      |          |           |
|--------------------------------------|----------|-----------|
| <input type="checkbox"/> Niet actief | Bewerken | Verwijder |
|--------------------------------------|----------|-----------|

Nieuwsbericht 2

|  |          |           |
|--|----------|-----------|
| <input checked="" type="checkbox"/> Actief | Bewerken | Verwijder |
|--|----------|-----------|

Nieuwsbericht 3

|                                      |          |           |
|--------------------------------------|----------|-----------|
| <input type="checkbox"/> Niet actief | Bewerken | Verwijder |
|--------------------------------------|----------|-----------|

Nieuwsbericht 4

|  |          |           |
|--|----------|-----------|
| <input checked="" type="checkbox"/> Actief | Bewerken | Verwijder |
|--|----------|-----------|

[nieuw bericht toevoegen](#)

- **Diepere validatie:**

Voer voor een actie naar keuze een extra validatie toe, waarbij de gebruiker deze actie pas mag uitvoeren nadat hij iets anders een x aantal keer heeft gedaan. (licht dit toe in je assessment)

a. *Bijvoorbeeld bij een forum:*

een gebruiker mag pas een topic openen als hij minimaal 5 reacties heeft gegeven.

b. *Bijvoorbeeld bij review systeem:*

een gebruiker mag een review plaatsen als hij minimaal op 5 verschillende dagen is ingelogd.



## Mondelinge toets

In het mondeling moet duidelijk worden of je zelfstandig in staat bent geweest om een framework onder de knie te krijgen, de richtlijnen van het framework hebt kunnen volgen en daar uiteindelijk een applicatie in hebt kunnen bouwen. Steekproefsgewijs zullen er vragen worden gesteld uit onderwerpen van de les (hoe komt het terug in het framework en waar heb jij het geïmplementeerd). Ook de bovengenoemde verplichte onderdelen zullen daarbij aan bod komen.

## Fases van het mondeling

### Fase 1: Zelf presenteren in 5 minuten

Vertel iets over het proces en hoe de planning die je in week 1 gemaakt hebt zich verhoudt tot het uiteindelijke verloop.

Laat de voortgang middels GitHub zien. Zorg dus dat deze pagina open staat.

Doorloop zelf de belangrijkste punten en licht dit toe aan de hand van de live applicatie en de code. Neem hierin de verplichte onderdelen van de opdracht mee.

### Fase 2: Vragen

- specifieke oplossingen vanuit jouw framework (ken je ze, waarom evt niet gekozen?)
  - bijvoorbeeld: hoe gaat je framework om met beveiliging?
  - bijvoorbeeld: hoe gaat je framework om met databases?
  - bijvoorbeeld: aan welke eisen moet een 'live' server voldoen?
  - bijvoorbeeld: toelichten waarom je iets niet op de 'correcte' manier hebt gedaan
- algemene MVC vragen (ken je ze, en hoe zit dat in jouw framework)
  - bijvoorbeeld: wat zijn layouts/partials, en hoe zit dat in jouw framework?
  - bijvoorbeeld: wat is de taak van een controller?
  - bijvoorbeeld: wat is een sql injection, en hoe beveilig je hiertegen?
- specifieke vragen m.b.t. hoe je iets hebt gebouwd (technisch in je code)
  - bijvoorbeeld: hoe heb je verschillende rollen geïmplementeerd?



## Beoordeling praktijkopdracht (70% eindcijfer)

Zie ook modulewijzer

|   |   | punten |   | punten |  | punten |
|---|---|--------|---|--------|--|--------|
| De opgeleverde webapplicatie is voldoende uitdagend | Webapplicatie is complexer dan de voorbeeldopdracht   | 3      | Webapplicatie is gelijk aan de voorbeeldopdracht  | 1,5    | Webapplicatie is minder complex  | 0      |
| Gemaakte keuzes in de uitwerking zijn onderbouwd    | Je kunt veruit de meeste gemaakte keuzes onderbouwen en aangeven hoe deze tot stand zijn gekomen  | 3      | In de uitleg van de werking is duidelijk geworden dat je met name hulp hebt gehad (bijv dmv een tutorial) van externen  | 1,5    | Je kunt de werking niet onderbouwen en voor de uitwerking ben je vrijwel volledig afhankelijk geweest van hulpmiddelen   | 0      |
| Projectvoortgang en Git voortgang                   | De voortgang op GIT laat een evenredige werkbelasting over de periode zien en dit heeft een match met de planning en user stories   | 2      | De voortgang laat een evenredige werkbelasting zien maar wel met een piekbelasting aan het einde.<br>Er is niet gehandeld naar het feit dat de projectplanning en/of user stories hebben afgeweken. | 1      | De werkbelasting zit met name aan het einde van het project of GIT is niet gebruikt.<br>De user stories beschrijven niet het project of de planning is totaal niet realistisch | 0      |
| OWASP top 10  | de top 10 onderdelen zijn zeer consequent toegepast door de applicatie  | 1      | de top 10 onderdelen veelal toegepast door de applicatie  | 0,5    | de top 10 onderdelen zijn beperkt toegepast  | 0      |
| OTAP  | De ontwikkel en productieomgeving zijn voorzien van een eigen configuratie en nergens in code wordt gecontroleerd op HTTP_HOST en configuratie staat op de juiste manier in Git | 1      | Er is op 1 plaats verkeerd rekening gehouden met de scheiding tussen ontwikkel en productieomgeving   | 0,5    | Er is op meer dan 1 plaats verkeerd rekening gehouden met de scheiding tussen ontwikkel en productieomgeving   | 0      |