

# Projectopgave

In dit project ontwikkel je een website aan de hand van het **Laravel-framework**. Het thema en concept kies je volledig zelf. Let er wel op dat je een website aflevert die

- een bepaald economisch/sociaal/praktisch ... nut heeft in de werkelijke wereld
- een **logisch en samenhangend geheel** vormt (een totaalconcept) m.a.w. je levert geen verzameling losstaande pagina's af en ook geen website waarbij essentiële onderdelen ontbreken

De website maakt gebruik van een databank die uit minstens **3 onderling gerelateerde tabellen** bestaat (gebruikerstabel mag hiervan deel uitmaken). Zorg ervoor dat de data in minstens een deel van de tabellen kan worden getoond in een **overzichtpagina** en in aparte **detailpagina's** (niet als een table- of recorddump; voorzie een minimum aan lay-out/groepering). Ook moet het op zijn minst mogelijk zijn om data toe te voegen en te bewerken. De voertaal van de website is vrij te kiezen.

## Technisch

Gebruik de in de theorie en in het labo geziene structuren en methodieken: URL design, Laravel-structuur, Custom Controllers, Route Groups, Blade, Query Builder / Eloquent ORM, Authentication, Forms, Validation, Pagination, enz.

Om voldoende technologische diepgang te geven aan je oplossing, werk je één van volgende concepten uit (verder beschreven als *concept naar keuze*):

- een uitgebreide filtering a.d.h.v. Sessions
- een zoekfunctie én gebruiksvriendelijke paginatie
- gebruikers met verschillende rollen en rechten (Authorization)

Je kan extra punten scoren wanneer je een *extra feature* dat de nodige zelfstudie vereist realiseert (zie 'evaluatie').

**Beveiliging** van de website staat hoog in het vaandel: wachtwoorden worden geëncrypteerd, enkel rechtmatige toegang tot de correcte pagina's wordt toegestaan, etc. Let verder op courante PHP-zaken zoals het plaatsen van commentaren en het volgen van de stijlregels.

**Databank.** Voorzie een genormaliseerde MySQL- of SQLite-databank met de nodige relaties. Deze bestaat uit minstens 3 onderling gerelateerde tabellen. Populeer deze databank met voldoende representatieve data: minstens 3 realistische gebruikers en in totaal minstens 20 realistische data-entries – geen *sfdsdqfsqq*s en *Lorem Ipsum* dus!

**Configuratie.** Voor de configuratie die je aflevert heb je twee mogelijkheden:

1. Je levert een standaard Laravel-project af voorzien van Migrations en Seeders. Je houdt je zoveel mogelijk aan de git-conventies m.b.t. een Laravel-project en specificeert de installatie-instructies in de readme.md. Na het opvolgen van de instructies werkt je project feilloos op een labo-PC (WAMP).
2. Je levert een Vagrant-omgeving af die in de provisioning o.a. de databank importeert en de Laravel-bibliotheken installeert. Je website is volledig operationeel wanneer het commando *vagrant up* wordt aangeroepen op een labo-PC.

Het is niet toegelaten om een SQLite-databank-bestand rechtstreeks in je oplossing op te nemen!

**Ontwerp.** Zorg voor een eenvoudig, overzichtelijk en ergonomisch ontwerp voor je website dat aanleunt bij het onderwerp en doelpubliek. (Geen one-page-design!) Maak gebruik van een bestaande CSS template maar zorg er voor dat deze naadloos overgaat in de eigen aanpassingen/uitbreidingen die je doorvoert. Zorg er m.a.w. voor dat het geheel coherent blijft.

Volg zo veel als mogelijk de webstandaarden (HTML5, CSS3). De te gebruiken *rich text editor* - indien nodig - is TinyMCE. Pas deze aan de noden van de gebruiker aan.

## Minimumvereisten

Indien aan één van de volgende voorwaarden niet voldaan is, is het resultaat automatisch 0:

- De gebruikte scriptingtaal is PHP (5.4 of hoger)
- Het gebruikte framework is Laravel (5.2)
- De gebruikte template engine is Blade
- Het gebruikte DBMS is MySQL of SQLite
- De data wordt effectief uit de databank opgehaald

Indien er zich grote beveiligingsproblemen voordoen (*SQL Injectie, Cross-Site Scripting (XSS), onrechtmatige toegang al dan niet door Parameter Tampering (!), niet-geëncrypteerde wachtwoorden in de databank, foutboodschappen met wachtwoorden*) worden er 5 punten van het totaalcijfer afgetrokken.

### Timing en Upload

Je oplossing bewaar je in een Git-repository die je eveneens naar de Ikdoeict GitLab Server (<https://git.ikdoeict.be/>) pusht. Noem deze repo SWS2\_1516\_project en deel deze met jouw labodocent (kevin.picalausa of joris.maervoet) die je master rechten toekent – verder is deze online repo niet publiek. **Afwijkingen op deze naamgeving of niet gedeelde repos worden als niet ingediend beschouwd.**

Volg één van de twee hierboven beschreven configuratie-mogelijkheden strikt op!

De deadline voor pushen is **zondag 29/05/2016 om 22u**

## Evaluatie

Jouw inzending zal tijdens examenperiode 2 samen met jou en een labodocent getoetst worden aan volgende functionele criteria.

<b>FUNCTIONEEL</b>	<b>/10</b>
Totaalconcept en logische samenhang	/1
Gebruiksgemak en informatie-architectuur <ul style="list-style-type: none"> <li>• Breadcrumbs</li> <li>• Vermijd dead ends</li> <li>• Paginering (indien van toepassing)</li> </ul>	/1
Design: is er een functioneel, coherent en uniform design? <ul style="list-style-type: none"> <li>• Navigatie</li> <li>• Header, footer en toolbar</li> <li>• Ook uniform design voor login en registratie</li> </ul>	/1
URL design	/1
Databank ontwerp: is het achterliggend db-model relevant, voldoende groot en worden de tabellen effectief gebruikt?	/1
Realisatie <i>concept naar keuze</i>	/2
Pagina's: wordt je informatie op een heldere en krachtige manier weergegeven? Enkel <u>niet-statische</u> pagina's worden in rekening gebracht. Een tabeldump van alle records is geen handig overzicht. <ul style="list-style-type: none"> <li>• Homepagina</li> <li>• Overzichtspagina</li> <li>• Detail pagina</li> <li>• Case-specifieke pagina's</li> </ul>	/2
Gebruikerssysteem <ul style="list-style-type: none"> <li>• Werk een correct gebruikerssysteem uit dat voldoet voor jouw case. Denk hierbij aan het al dan niet nuttig zijn van een profiel- of registratiepagina. Onrechtmatige toegang tot de registratie blok je af!</li> </ul>	/1

Tevens wordt jouw inzending ook technisch nagekeken. Dit gebeurt waar mogelijk op het vlak van de efficiëntie, kwaliteit en structuur van de code.

<b>TECHNISCH</b>	<b>/10</b>
------------------	------------

Routing en controllers <ul style="list-style-type: none"> <li>• groepeer routes waar nodig</li> <li>• plaats de logica in controllers die je indeelt per entiteit</li> </ul>	/1
Views (hiërarchisch) <ul style="list-style-type: none"> <li>• logische mappenstructuur en template-overerving</li> <li>• gebruik de juiste helper-functies</li> <li>• overweeg het gebruik van Laravel Collective's HTML&amp;Forms</li> </ul>	/2
Eloquent <ul style="list-style-type: none"> <li>• vermijd dat een for-loop in Blade leidt tot veelvuldig query'en</li> <li>• plaats herhaaldelijke complexe queries in repositories</li> <li>• gebruik create/save zoals aangewezen</li> </ul>	/1
Configuratie (migrations en seeders OF vagrant-omgeving) <ul style="list-style-type: none"> <li>• Zie technisch gedeelte opgave</li> </ul>	/1
Authentication <ul style="list-style-type: none"> <li>• Gebruik het ingebouwde mechanisme, maar pas het aan aan de meest voor de hand liggende oplossing voor jouw systeem bv. geen registratie, uitgebreide registratie of enkel registratie door admins</li> <li>• integratie in het DB-design</li> </ul>	/1
Forms en validatie <ul style="list-style-type: none"> <li>• pas de correctie validatie toe op jouw forms</li> <li>• ondersteun persistentie</li> </ul>	/1
Realisatie <i>concept naar keuze</i> <ul style="list-style-type: none"> <li>• Zie technisch gedeelte opgave</li> </ul>	/2
Codedocumentatie	/1

Je kan extra punten scoren wanneer je een *extra feature* dat de nodige zelfstudie vereist realiseert (**optioneel**):

EXTRA FEATURE	/2
Realiseer één van volgende zaken (evaluatie functioneel en technisch) <ul style="list-style-type: none"> <li>• e-mail met aparte configuratie voor testen en productie</li> <li>• tracking cookie met opslag gedrag bezoekers</li> <li>• ondersteuning meerdere talen (Localization)</li> <li>• je werkt met afbeeldingen die efficiënt op schijf moeten worden opgeslagen en waarvan ook thumbnails op een efficiënte manier moeten worden voorzien</li> <li>• je gebruikt een JS-bibliotheek zoals jquery die gebruik maakt van JSON-gegevens</li> </ul>	/2

gegenereerd vanuit Laravel (bv. voor de filtering) <ul style="list-style-type: none"> <li>• je voorziet SEO-gerelateerde features die zich dynamisch aanpassen wanneer de data wijzigt zoals een correcte en volledige sitemap.xml én rich snippets of gelijkaardig</li> <li>• je gebruikt <i>memcached</i> voor een relevante feature</li> </ul>	
---	--

### **Bepaling eindcijfer**

Het eindcijfer wordt, mits aanwezigheid op het evaluatiemoment, berekend als het gemiddelde van het meetkundig en rekenkundig gemiddelde van de functionaliteit F (cijfer op /10) en de uitvoering U (cijfer op /10), vermeerderd met het extra feature E (cijfer op /2):

$$\text{cijfer} = ( ((F*U)/5) + (F+U) ) / 2 + E$$

Heb je - het extra feature buiten beschouwing gelaten - bijvoorbeeld op functionaliteit 7/10 en op uitvoering 8/10, dan is het cijfer 13/20. Heb je op functionaliteit 5/10 en op uitvoering 5/10, dan is je cijfer 7,5/20. De redenering achter deze formule is dat je nooit geslaagd kunt zijn als je maar de helft van de functionaliteit hebt, en dat ook nog eens voor de helft verkeerd uitvoert. Uiteraard kan je nooit meer dan 20/20 krijgen als totaalcijfer voor dit vak.

### **Huishoudelijk reglement en modaliteiten**

Het huishoudelijk reglement is van toepassing. Jouw oplossing moet individueel tot stand komen. Kopiëren of laten kopiëren resulteert automatisch in minstens een 0 voor het ganse project. Maak voldoende backups: een crash van welke aard ook is geen excuus voor laattijdig indienen. Studenten die niets of niet tijdig ingediend hebben worden standaard niet verwacht op het evaluatiemoment en automatisch een NA toegekend.

**Let op: om te kunnen slagen voor de OPO Webscripten 2 moet je minstens 8/20 op het project behalen.**

### **Disclaimer EP3**

De docenten behouden zich het recht om de projectopgave voor EP3 te herformuleren en andere vereisten en beperkingen op te leggen, zowel van technologische als van thematische aard.

Veel succes ... en laat het niet te lang liggen!

Kevin Picalausa en Joris Maervoet