

2.9 Routes

Laravel zorgt voor cleane URL's. Dit wil zeggen dat de links naar webpagina's aangepast kunnen worden zonder dat de opgeroepen bestanden hiervoor van naam moeten veranderen. Daarnaast worden er dan ook geen bestandsextensies weergegeven bij het surfen naar de pagina. Dit bevordert de ervaring van de gebruiker. Door de routes logisch op te stellen, zal een gebruiker in staat zijn om de juiste pagina te vinden. Zo verliest hij geen tijd bij het zoeken in menu's. Zeker bij grote (onoverzichtelijke) websites kan dit wel degelijk nuttig zijn.

```
1  <?php
2
3  /*
4  |-----
5  | Web Routes
6  |-----
7  |
8  | Here is where you can register web routes for your application. These
9  | routes are loaded by the RouteServiceProvider within a group which
10 | contains the "web" middleware group. Now create something great!
11 |
12 */
13
14 Auth::routes();
15
16 Route::get('/', 'OverviewController@index')->name('home');
17 Route::get('/archief', 'OverviewController@archive')->name('archief');
18 Route::get('/maakproject', 'ProjectController@createProject')->name('maakproject');
```

Codefragment 9:Routes

Eén van Laravels belangrijkste hulpmiddelen zijn de routebestanden (codefragment 9). Hierin wordt bepaald welke functie uit een controller wordt opgeroepen als er naar een bepaalde URL wordt gesurft.

2.10 Authenticatie

Laravel biedt een ingebouwde authenticatie aan. Door het commando 'php artisan make:auth' uit te voeren in een terminal, worden alle nodige bestanden aangemaakt. Hierdoor zal er een eenvoudige login- en registreerpagina gevormd worden, alsook de achterliggende controllers. Standaard gebeurt authenticatie op basis van de paswoord- en e-mailkolom uit de gebruikerstabel. Dit is ook van toepassing bij het maken van de projectenwebsite.

Doordat de authenticatie automatisch gebeurt, kunnen de andere controllers makkelijk controleren indien een gebruiker daadwerkelijk ingelogd is. Met behulp van de coderegel 'Auth::user()' worden alle gegevens van de ingelogde persoon geladen. Indien de waarde 'NULL' is, betekent dit dat de gebruiker nog niet is ingelogd.

2.11 Conclusie

Laravel is een gratis, opensource maar vooral eenvoudig PHP web framework. De documentatie is zeer uitgebreid. Door de grote community is er veel overzichtelijke documentatie beschikbaar. In het framework zitten veel handige ingebouwde functionaliteiten. De structuur is overzichtelijk waardoor de code goed leesbaar is. Door Eloquent ORM is de relatie tussen de database veel eenvoudiger. Via een speciaal routebestand wordt er gezorgd dat de juiste request met de juiste methodes verbonden worden. Dit maakt het voor de developer een stuk overzichtelijker.

3. BootstrapVue

3.1 Framework Vue

Vue is een JavaScriptframework voor het bouwen van gebruikersinterfaces. In tegenstelling tot andere monolithische frameworks, is Vue vanaf de basis ontworpen om zeer eenvoudig toepasbaar te zijn. De kernbibliotheek is hoofdzakelijk gericht op de presentatie laag en is eenvoudig te integreren met andere bibliotheken of bestaande projecten. Aan de andere kant is Vue ook perfect in staat om geavanceerde Single-Page Applicaties aan te sturen in combinatie met ondersteunende bibliotheken.

Eén van de redenen waarom voor Vue.js wordt gekozen, is dat JavaScript eenvoudig kan worden toegevoegd. De broncode is zeer goed leesbaar en de documentatie is de enige tutorial die er nodig is. Er zijn geen externe bibliotheken nodig. Het is ook niet vereist om plug-ins te installeren, hoewel er veel plug-ins beschikbaar zijn. Het aansluiten van Vue.js op bestaande code is ook heel eenvoudig (codefragment 10) [9].

Om een voorbeeld te geven van wat er bedoeld wordt:

```
<div id="journal">
  <input type="text" v-model="message">
  <div>{{message}}</div>
</div>
```

Codefragment 10: Voorbeeld van Vue [9]

Voor iemand die al ervaring heeft met JavaScript-bibliotheken, is codefragment 11 vrij eenvoudig te verstaan. De “new Vue” zorgt ervoor dat er een nieuwe component kan worden aangemaakt met daarin de nodige data. Deze relatief korte en simpele code is een van de sterktepunten van Vue.js in tegenstelling tot andere frameworks.

```
var journal = new Vue({
  el: '#journal',
  data: {
    message: 'Your first entry'
  }
});
```

Codefragment 11: JavaScript in Vue

3.2 Componenten

De meeste frontendframeworks zorgen ervoor dat een website kan worden opgebouwd uit componenten. Elk belangrijk afzonderlijk onderdeel van een site kan worden omgezet in zo een component. Elke component heeft dus zijn eigen HTML-, JavaScript- en CSS-code indien nodig.

Elke webpagina bestaat uit één grote component die opgebouwd is uit één of meerdere kleinere componenten. Elke component kan een parent- en childcomponent bevatten. Parent A is de component waarin de huidige component B zich bevindt en B is dan de childcomponent van component A. Zo is alles heel overzichtelijk en kan je elk deel perfectioneren naar eigen belang en zin, zonder dat je met de rest rekening moet houden. Alles is mooi geordend en als er een fout is, kan je ze makkelijk opsporen en verhelpen.

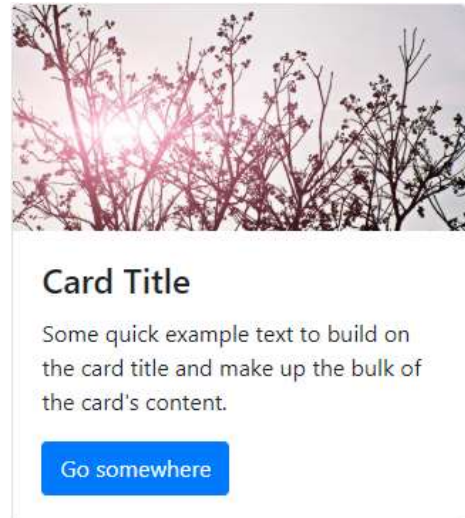
3.3 BootstrapVue

BootstrapVue is het framework waarmee te werk is gegaan, exact hetzelfde als Vue maar Bootstrap (toolkit om HTML, CSS, JavaScript makkelijker te implementeren) wordt hier mee in geïntegreerd. Dit maakt het mogelijk zo weinig mogelijk code te moeten schrijven voor wat men wil bereiken. Bootstrap werkt met voorgemaakte klassen, onderdelen, afkortingen, binnenwegen, ... om zo efficiënter te kunnen programmeren.

```
<div>
  <b-card
    title="Card Title"
    img-src="https://picsum.photos/600/300/?image=25"
    img-alt="Image"
    img-top
    tag="article"
    style="max-width: 20rem;"
    class="mb-2"
  >
    <b-card-text>
      Some quick example text to build on the card ti
    </b-card-text>

    <b-button href="#" variant="primary">Go somewhere
  </b-card>
</div>
```

Codefragment 12: Card in Vue



Figuur 3: Card in Vue

In codefragment 12 wordt dit alles duidelijk, waar een “card” voorgesteld wordt. De tag “b-card” is een bestaand onderdeel in BootstrapVue. In deze tag kunnen extra opties zoals border, afbeeldingen, stijl... gewijzigd, ingevoegd worden. De voorgemaakte klassen kan men ook terugvinden in de optie class.

Een klasse is een groep van stijlen die toegepast zullen worden op meerdere elementen. Maar Bootstrap heeft echter ook voorgemaakte klassen, die de meest voorkomende stijlen bevatten. Met enkele letters worden deze zo toegevoegd aan het object. Zo verwijst ‘mb-2’ naar “margin-bottom = 2rem;” (codefragment 12) [10].

3.4 HTML

Elke component heeft dus zijn eigen HTML-, JavaScript- en CSS-gedeelte. HTML wordt geschreven in een “<template>”-tag, Vue heeft een ingebouwd grid- en lay-outsysteem wat alles zeer overzichtelijk en gebruiksvriendelijk maakt (figuur 4). Daarnaast zorgt dit systeem ervoor dat alle pagina’s ook responsive⁴ gemaakt worden.

⁴ Ervoor zorgen dat een website zich automatisch aanpast naargelang de grootte van het scherm van de gebruiker. [12]

col-sm-8		col-sm-4
col-sm	col-sm	col-sm

Figuur 4: Grid- en lay-outsysteem

```

<b-container class="bv-example-row">
  <b-row>
    <b-col sm="8">col-sm-8</b-col>
    <b-col sm="4">col-sm-4</b-col>
  </b-row>

  <b-row>
    <b-col sm>col-sm</b-col>
    <b-col sm>col-sm</b-col>
    <b-col sm>col-sm</b-col>
  </b-row>
</b-container>

```

Codefragment 13: Grid- en lay-outsysteem in Vue

De pagina wordt bekeken als een tabel van rijen en kolommen. Elke rij en/of kolom kan worden aangepast naar eigen wil: de breedte, hoogte, responsive, enz.

In elke rij en kolom die nodig is, wordt dan de nodige html geïmplementeerd met zijn eigen stijl die onderaan het bestand wordt toegevoegd in het CSS-gedeelte.

3.5 JavaScript

In het <script> gedeelte wordt de JavaScriptcode geschreven: de functionaliteit. Hier wordt de data beschreven en bewerkt.

```

data: {
  name: 'Vue.js'
},
// define methods under the 'methods' object
methods: {
  greet: function (event) {
    // 'this' inside methods point to the Vue instance
    alert('Hello ' + this.name + '!')
    // 'event' is the native DOM event
    alert(event.target.tagName)
  }
}
})

```

Codefragment 14: JavaScript methoden in Vue

In het scriptgedeelte kan alles meegegeven, gedeclareerd en bewerkt worden. De parentcomponent kan “props” doorgeven. Dit zijn data die kunnen doorgegeven worden. Daarnaast blijven uiteraard nog de huidige data aanwezig. Deze kunnen bewerkt worden in het gedeelte van de methoden. Er zijn ook lifecyclemethoden. Deze worden altijd, één voor één, opgeroepen bij elk deel van het laadproces van een component.

Voorbeeld: De “props” worden doorgegeven van de parentcomponent. Deze zijn nog niet beschikbaar als de website ingeladen wordt. Dit gebeurt pas als de pagina met de huidige component ingeladen wordt. Eén van de vele lifecyclemethoden houdt in dat deze pas opgeroepen wordt nadat de component ingeladen is, nl. mounted. In deze methode zijn dus de “props” al beschikbaar en kunnen ze bewerkt worden.

3.6 CSS

Voor traditionele websites zijn er veel manieren waarop CSS kan worden geïmplementeerd. De meest voorkomende manier is om de opmaak van elke webpagina in een eigen CSS-bestand te zetten. Voor kleinere pagina's is er vaak ook CSS binnen de HTML-pagina (meestal in het <head> gedeelte). Dit resulteert in grote CSS-bestanden, die moeilijk te beheren zijn.

```
1 <template>
2   <div class="success-message">this will be green</div>
3 </template>
4
5 <style scoped>
6   .success-message{
7     color: green;
8   }
9 </style>
```

Codefragment 15: CSS in Vue

In Vue kan elk onderdeel zijn eigen styling hebben (codefragment 15). Het scoped attribuut zorgt ervoor dat deze CSS-code enkel lokaal toegepast wordt. Anders bestaat het risico dat andere delen van de site die dezelfde klassenaam hebben ook aangepast worden.

3.7 Besluit

Vue is zeer handig voor het opsplitsen van lange codeteksten. Door alles op te bouwen in kleine componenten wordt overzicht bewaard op zowel HTML, CSS als JavaScript. Per component is elk codeblok dus vrij beperkt in lengte. Voor de projectenwebsite die relatief complex is zorgt dit ervoor dat er efficiënt kan worden gewerkt.

4. Webtoepassing hosten

Met Microsoft Azure kunnen er snel en efficiënt cloudoplossingen worden ontwikkeld, geïmplementeerd en beveiligd. Het implementatiemodel van Azure is flexibel. Zo kan er in Visual Studio code worden ontwikkeld. Deze kan worden doorgevoerd naar GitHub, gebouwd met Azure DevOps en vervolgens in Azure geïmplementeerd. Er kunnen ook een andere IDE (Integrated Development Environment) gebruikt worden. Ook is het mogelijk om rechtstreeks vanaf de ontwikkelcomputer code in de cloud te implementeren.

Er is een groot gamma aan programmeertalen, van C# tot Java en Node.js. De gebruiker kiest zelf het ontwikkelbesturingssysteem. Microsoft biedt en ondersteunt een breed scala aan ontwikkelprogramma's voor Windows, Linux en macOS.[11]

4.1 Azure portal

De eerste stap bij het hosten van de webtoepassing is het maken van een application service binnen het Azure-abonnement.

Er zijn verschillende manieren waarop dit kan worden gemaakt. Er kan Azure Portal, de Azure-CLI, en script of een IDE worden gebruikt.

De grafische interface van de Azure-portal is zeer gebruiksvriendelijk. In de portal kunnen er extra resources toegevoegd worden en bestaande resources kunnen worden aangepast.

4.2 Azure App service

Azure App Service is een volledig beheerd computerplatform in de Azure-omgeving, geoptimaliseerd voor het hosten van web-apps, REST-API's en back-ends.

Met dit PaaS (Platform as a Service) aangeboden door Microsoft Azure kan men zich concentreren op het ontwikkelen, terwijl Azure zorgt voor de infrastructuur die nodig is om de toepassingen uit te voeren en uit te breiden.

4.3 Web-app

Wanneer er een app moet worden gehost, kan er in de portal een nieuwe web-app worden aangevraagd. Door het maken van een web-app in Azure Portal, worden er een aantal hostingresources in App Service gemaakt. Deze kunnen gebruikt worden voor het hosten van een applicatie, ondersteund door Azure, ongeacht of deze is geschreven in ASP.NET Core, Node.js, PHP, enzovoort. In figuur 5 wordt een configuratie van een framework of programmeertaal van een app weergegeven.

Home > Alle resources > BestBike1234 - Toepassingsinstellingen wijzigen

BestBike1234 - Toepassingsinstellingen wijzigen
App Service

Zoeken (Ctrl+/,)

Opslaan Negeren

Instellingen

- Toepassingsinstellingen wijzigen...
- Authentication / Authorizati
- Application Insights
- Beheerde service-identiteit
- Back-ups
- Aangepaste domeinen
- SSL-instellingen
- Networking
- Opschalen (App Service-plan)
- Uitschalen (App Service-plan)
- WebJobs
- Push
- MySQL In-app
- Eigenschappen

Algemene instellingen

.NET Framework version v4.7

PHP version 5.6

App Service ondersteunt de installatie van nieuwere Python-versies. Klik hier voor me...

Python version Uit

Java version Uit

Java minor version

Java web container

Platform 32-bit 64-bits

Web sockets Uit Aan

Always On Uit Aan

Figuur 5: Instellingen in Azure

Dankzij de vele beschikbare hulpprogramma's in Azure Portal kan er volledig bepaald worden hoe de applicatie wordt beheerd en beveiligd.

4.4 Implementatiesites

Met behulp van Azure Portal kunnen er eenvoudig implementatiesites aan een App Service-web-app worden toegevoegd. Er kan bijvoorbeeld een implementatiesite voor fasering worden gemaakt. Hier kan code naar gepusht worden om deze te testen in Azure. Als alles in orde is, kan de staging-implementatiesite eenvoudig worden gewisseld met de productiesite. In Azure Portal gebeurt dit allemaal met een paar muisklikken (figuur 6).

Start > Alle resources > BestBike1234 - Implementatiesites

BestBike1234 - Implementatiesites
App Service

Zoeken (Ctrl+/,)

+ Site toevoegen Wisselen

Implementatie

- Snelstart
- Implementatiereferenties
- Implementatiesites
- Implementatie-opties
- Implementatiecenter (preview)

NAAM	STATUS	APP SERVICE-PAN
bestbike1234-staging	Wordt uitgevoerd	BestBike1234-app-service-plan

Figuur 6: Voorbeeld van Implementatiesite.

4.5 Ondersteuning continue integratie en implementatie

Azure Portal biedt onmiddellijk na aanschaf continue integratie en implementatie met Azure DevOps, GitHub, Bitbucket, Dropbox, OneDrive en een lokale Git-opslagplaats voor de ontwikkelcomputer. Wanneer de applicatie verbonden wordt met een bron, doet de App Service de rest. De code en toekomstige wijzigingen hiervan zullen automatisch synchroniseren met de applicatie. Bovendien kan er met Azure DevOps een eigen build- en releaseproces worden gedefinieerd voor het compileren van de broncode, het uitvoeren van de tests, het bouwen van een release en ten slotte het implementeren van de release naar de applicatie, telkens wanneer de code wordt doorgevoerd. Dit gebeurt allemaal zonder tussenkomst van de gebruiker.

4.6 Samenvatting

Azure App Service vereenvoudigt het beheren en controleren van de applicatie in vergelijking met traditionele hostingopties. Met App Service kost het minder tijd en moeite om de applicatie uit te voeren en te beheren. Het biedt bovendien geavanceerde Cloudfuncties, zoals automatisch schalen en Git-integratie.

5. Functionaliteiten

5.1. Integratie van de Belbin test

Na het aanmaken van een account moeten de gebruikers een Belbin test invullen. Via deze test, die is gebaseerd op dr. Belbins ideeën en is ontwikkeld voor de Nederlandse markt, kan er bepaald worden in welke mate, vanuit de persoonlijkheid, de verschillende teamrollen er worden aangesproken.

Een team bestaande uit creatievelingen loopt het risico te verzanden in ideeën. Een team bestaande uit uitvoerders loopt het risico veel werkzaamheden te verzetten zonder bevredigend resultaat. Een team dat zich van dit alles bewust van is, kan hierdoor de verdeling van taken en functies organiseren met het oog op een betere werking van het team met gepast resultaat.

5.2. Project toevoegen

Projecten of ideeën kunnen door iedere gebruiker, zowel docent als student worden toegevoegd. Bij het aanmaken van een project moeten de SMART-criteria worden opgegeven. Deze worden gebruikt om meetbare en realistische doelstellingen te formuleren. Het vaststellen van meetbare doelstellingen levert betere resultaten op en maakt het achteraf eenvoudiger om de realisatie ervan te evalueren.

5.3. Goedkeuren project

In de oude versie worden projecten door docenten aan de database toegevoegd en voorstellen goedgekeurd. Studenten kunnen enkel een projectvoorstel indienen en zich hiervoor inschrijven. Bovendien hebben ze geen toegang tot afgelopen projecten om inspiratie op te doen. In de nieuwe versie kunnen projecten op een eenvoudige manier worden goed- of afgekeurd. Gebruikers kunnen ook projecten van de voorbije jaren bekijken om hen te helpen bij het ontwikkelen van een idee.

5.4. Klassensysteem

Op de nieuwe versie van de projectenwebsite is er een duidelijk verschil tussen docent en student. Studenten kunnen meerdere voorstellen indienen die op hun beurt worden goed- of afgekeurd. Indien een voorstel wordt afgekeurd kunnen studenten dit alsnog aanpassen naar de gevraagde criteria.

6. Planning

In het project zijn er 4 grote onderdelen, namelijk:

- Wireframe, usability, design
- HTML & CSS
- Web scripting
- Extra's (Responsive, Belbin test, chat, ...)

Einddatum	Mijlpaal	Wie?
20 februari 2019	Vergadering #1 Eerste brainstorm website	Groep
21 februari 2019	Afspraak #1 begeleider	Groep
Krokusvakantie	Vergadering #2 Tweede brainstorm website	Groep
11 maart 2019	User Experience	Groep
14 maart 2019	Vergadering #3 en Afspraak #2 begeleider Brainstorm design en database- ontwerp, feedback wireframes	Groep
21 maart 2019	HTML	Groep
28 maart 2019	CSS	Groep
4 april 2019	Afspraak#3 begeleider	Groep
22 april 2019	Laravel, MySQL, Blade	Andreas en Lukas
25 april 2019	Vergadering #4 en Feedback Laravel, MySQL, Blade	Groep
25 april 2019	Afspraak #4 begeleider	Groep
2 mei 2019	Vue	Arijn en Arno
8 mei 2019	Vergadering #5 Feedback Vue	Groep
9 mei 2019	Afspraak #5 begeleider	Groep
15 mei 2019	Extra's	Groep
16 mei 2019	Vergadering #6 Overlopen eindproduct, documentatie afwerken.	Groep
22 mei 2019	Eindproduct indienen	Groep

Conclusie

De nieuwe website heeft een inlogsysteem voor studenten en docenten. Studenten moeten verplicht de Belbin test afleggen. Aan de hand daarvan krijgen ze voorgestelde groepen waarin ze terecht zouden passen. Ze beslissen wel nog zelf of ze bij de groep willen horen of niet, daarnaast kunnen ze een projectvoorstel toevoegen die dan door een docent goed- of afgekeurd kan worden. Verder kunnen de docenten opmerkingen meegeven.

Verder bestaat de website uit de nodige pagina's zoals het archief om inspiratie op te doen uit projecten van vorige jaren. De studentenpagina om informatie op te zoeken over een bepaalde student, groep, projectvoorstel of om te sorteren op aantal groepsleden, op achternaam, ...

Het is een klein maar krachtig platform, voor één bepaald doeleinde, nl. de groepsvorming van de projecten niet meer ten laste te leggen van de docenten. Maar eerder een taak van de studenten om zo snel mogelijk groepen te vormen en hun voorstel indienen zodat de docenten niet meer face to face of via e-mailberichten de voorstellen kunnen keuren en de groepen kunnen vaststellen in hun bestand. Dit platform bespaart dus een hoop tijd en typ- en administratiewerk voor de docenten.

Overzicht bijeenkomsten

Donderdag 14 februari:

Tijdens de eerste bijeenkomst wordt er beslist hoe het project wordt uitgewerkt: technieken, programmeertalen en databankmodel. Meerdere ideeën komen naar boven. Uiteindelijk wordt er gekozen om het frontendgedeelte met Vue te maken en het backend met Laravel, aangezien de deze technieken zo aangeleerd krijgen.

Donderdag 21 februari:

De initiële ideeën worden besproken met de docent. Het databankmodel en -type komen hier ook aan bod. Uiteindelijk wordt er gekozen voor MySQL.

De taken hebben we verdeeld naargelang ieders voorkeur: twee frontend, twee backend. Het userexperienceonderzoek wordt gezamenlijk uitgevoerd. Daar wordt vandaag al aan begonnen. Er zijn 2 mails gestuurd naar docenten om hun visie van de nieuwe projectenwebsite na te vragen. Ook wordt er nagedacht over hoe er kan onderzocht worden wat studenten van de site verwachten.

Donderdag 28 februari:

De groep heeft zijn research en werk over het userexperiencegedeelte gedaan. Nu worden de wireframes afgewerkt samen met de persona's. De planning van de komende weken wordt overlopen. Ieder werkt aan zijn deel. Het is de bedoeling om voor de paasvakantie de frontend klaar te hebben en de backend opgesteld te hebben zodat we in de paasvakantie dit kunnen omzetten.

Vrijdag 19 april:

Na onze afspraak met meneer Van Peteghem zien we de achterstand van het project in. We bespreken meteen de stand van zaken om de backend te kunnen laten samensmelten met de frontend.

Vrijdag 17 mei:

In deze bijeenkomst zijn 2 van de 5 hoofdpagina's omgezet en de filters werkende gemaakt.

Dinsdag 21 mei:

De frontend- en backenddevelopers hebben opnieuw samen gezeten om de code op elkaar af te stemmen. De website is nu bijna volledig aangepast. De stijl is zoals we het willen. Het rapport is klaar op door gestuurd te worden.

Literatuurlijst

Online handleidingen

- [1] Laravel, "Database: Getting Started," Laravel Documentatie, [Online]. Available: <https://laravel.com/docs/5.8/database>. [Accessed 5 mei 2019].
- [2] Laravel, "Database: Migrations," Laravel Documentatie, [Online]. Available: <https://laravel.com/docs/5.8/migrations>. [Accessed 5 mei 2019].
- [3] Laravel, "Database: Seeding," Laravel Documentatie, [Online]. Available: <https://laravel.com/docs/5.8/seeding>. [Accessed 5 mei 2019].
- [4] Laravel, "Eloquent: Relations," Laravel Documentatie, [Online]. Available: <https://laravel.com/docs/5.8/eloquent-relationships>. [Accessed 5 mei 2019].
- [5] Laravel, "Controllers," Laravel Documentatie, [Online]. Available: <https://laravel.com/docs/5.8/controllers>. [Accessed 5 mei 2019].
- [6] Laravel, "Blade Templates," Laravel Documentatie, [Online]. Available: <https://laravel.com/docs/5.8/blade>. [Accessed 5 mei 2019].
- [7] Laravel, "Routing," Laravel Documentatie, [Online]. Available: <https://laravel.com/docs/5.8/routing>. [Accessed 5 mei 2019].
- [8] Laravel, "Authentication," Laravel Documentatie, [Online]. Available: <https://laravel.com/docs/5.8/authentication>. [Accessed 5 mei 2019].
- [9] Vue.js, "Introduction," Vue Documentatie, [Online]. Available: <https://vuejs.org/v2/guide>. [Accessed 5 mei 2019].
- [10] Bootstrap-vue.js, "Getting started," [Online]. Available: <https://bootstrap-vue.js.org/>. [Accessed 5 mei 2019].
- [11] S. Sanders, *sven.sanders@odisee.be*, *Vragen omtrent project projectenwebsite*, 24 februari 2019.
- [12] w3schools, "HTML Responsive Web Design," [Online]. Available: https://www.w3schools.com/html/html_responsive.asp. [Accessed 5 mei 2019].