



Landleren

Bellens Toon, Bogaerts Arne, Metdepenningen Roel, Van Dyck Toon

# Inhoud

Leer de landen volgens kaart, omtrek, vlag of madness. ....	2
1. Inleiding.....	2
2. HTML Structuur.....	3
3. Javascript-functionaliteit.....	4
4. Gebruiksaanwijzing .....	6
5. Conclusie .....	6
6. Verwijzingen.....	7
Leer vlaggen met een AI .....	8
1. Inleiding.....	8
2. HTML Structuur.....	8
3. JavaScript-functionaliteit.....	8
Video-opname en beeldverwerking.....	8
Interactie met de gebruikersinterface .....	9
Vlagselectie en -score .....	9
Resultaten berekenen en weergeven .....	9
4. Gebruiksaanwijzing .....	9
5. Conclusie .....	10
6. Verwijzingen.....	10

# Leer de landen volgens kaart, omtrek, vlag of madness.

## 1. Inleiding

De oefeningen genaamd 'Vlaggen, Omtrek, Kaart en Madness' zijn een interactieve webapplicatie ontwikkeld met JavaScript, HTML en CSS. Het biedt gebruikers een plezierige en educatieve ervaring waarin ze hun kennis van wereldvlaggen kunnen testen.

Deze applicatie bevat een set van vragen over vlaggen, omtrek of plek op de kaart (naarmate van de oefening) van de verschillende landen in Europese Unie. Voor elke vraag wordt een afbeelding van een vlag, omtrek of kaart getoond en moeten gebruikers het bijbehorende land identificeren uit een set van vier opties. Elke vraag en de bijbehorende opties zijn opgeslagen in een JavaScript-object, en alle vragen zijn opgeslagen in een array van deze objecten.

De quiz is dynamisch en interactief. Het houdt het aantal vragen bij dat is gesteld, het aantal correct beantwoorde vragen.

Deze documentatie biedt een gedetailleerd overzicht van de functionaliteiten van de quizen. Het verklaart hoe de applicatie is opgezet en hoe de verschillende JavaScript-functies bijdragen aan de werking van de quiz. Het is bedoeld als hulpmiddel voor ontwikkelaars die het bestaande project willen uitbreiden of aanpassen, of voor degenen die willen leren hoe ze een soortgelijke quiz applicatie kunnen bouwen.

In de volgende secties zullen we in detail treden over de verschillende aspecten van de applicatie, inclusief de vraag- en antwoordlogica, de gebruikersinterface, de resultaatverwerking, en de structuur en het formaat van de vragenlijst. We hopen dat deze documentatie nuttig en informatief zal zijn voor iedereen die geïnteresseerd is in de onderliggende code van de Vlaggen Quiz Applicatie.

De optie 'Madness' laat toe om alle drie de voorgaande quizen ('vlaggen', 'omtrek' en 'kaart') te combineren in één quiz. Zo hebt je alle mogelijk vraagstellingen in één quiz.

## 2. HTML Structuur

Deze HTML-code definieert de structuur van de webpagina voor de quizapplicatie. Hier is de functionaliteit van elk onderdeel:

1. **Doctype, html, head, en body tags:** De code begint met de standaardstructuur van een HTML-document met de doctype-declaratie en de openende en sluitende tags voor html, head en body. Alles wat wordt weergegeven op de webpagina gaat in de body tag.
2. **Meta tags:** De charset meta tag definieert de tekenset die wordt gebruikt (UTF-8), en de viewport meta tag maakt de website responsief en schaalbaar op verschillende apparaten.
3. **Link tag:** De link tag verbindt een externe CSS-bestand, style.css, dat de stijlen bevat voor de elementen op de webpagina.
4. **Start-scherm div:** Deze div toont het start-scherm van de quiz met uitleg over de quiz, het totale aantal vragen en een knop om de quiz te starten.
5. **Quiz-vak div:** Deze div is het hoofdgedeelte van de quiz, dat de vraag, de keuzemogelijkheden en de volgende-knop bevat. Het heeft de klasse 'hide' die standaard het gedeelte verbergt en zal worden verwijderd wanneer de quiz wordt gestart.
6. **Resultaat-vak div:** Deze div toont de resultaten van de quiz, waaronder het aantal pogingen, het aantal correcte antwoorden, het percentage juiste antwoorden, en de totale score. Het bevat ook knoppen om de quiz opnieuw te proberen of terug te gaan naar de startpagina. Het heeft ook de klasse 'hide' en wordt alleen weergegeven aan het einde van de quiz.
7. **Script tags:** Aan het einde van de body tag zijn er twee script tags die de JavaScript-bestanden vlaggen-gemakelijk.js en app.js laden. Deze bestanden bevatten de logica voor de werking van de quiz.

In het algemeen heeft deze HTML-code de structuur voor een interactieve quizapplicatie waarbij de gebruikers de vlaggen van verschillende landen moeten identificeren. De eigenlijke interactieve elementen en de dynamische veranderingen op de pagina worden beheerd met JavaScript.

### 3. Javascript-functionaliteit

Elke test maakt gebruik van twee javascript files, een hiervan is app.js en een ander bevat een array met de vragen voor de test.

App.js

Het app.js JavaScript-bestand is voor een interactieve quiz met meerdere keuzes. Hier is een gedetailleerde uitleg van hoe dit script precies werkt:

1. Variabelen:
  - Het script begint met het definiëren van een aantal variabelen die door de code heen worden gebruikt. Deze variabelen bevatten elementen van de webpagina (zoals het aantal vragen, de vraagtekst, keuzecontainer, etc.), informatie over de quiz (zoals het huidige vraagnummer, de beschikbare vragen en opties, enz.), en statistieken (zoals het aantal correcte antwoorden en pogingen).
2. `setAvailableQuestions()`:
  - Deze functie wordt gebruikt om de beschikbare vragen in te stellen. Het doorloopt de vragenlijst `quiz` en duwt elke vraag naar de `availableQuestions` array.
3. `getNewQuestion()`:
  - Deze functie wordt gebruikt om een nieuwe vraag te krijgen voor de gebruiker. Het haalt willekeurig een vraag uit de `availableQuestions` array, toont deze vraag, en verwijdert deze vervolgens uit de array. Het haalt ook willekeurig de keuzeopties voor de vraag uit en toont deze aan de gebruiker.
4. `getResult(element)`:
  - Deze functie wordt aangeroepen wanneer de gebruiker op een optie klikt. Het controleert of het antwoord correct is door het vergelijken van het ID van de gekozen optie met het correcte antwoord van de huidige vraag. Het geeft ook feedback aan de gebruiker door de kleur van de gekozen optie te veranderen op basis van het feit of het antwoord correct of incorrect was.
5. `unclickableOptions()`:
  - Deze functie maakt alle opties onklikbaar na het beantwoorden van de vraag, om te voorkomen dat de gebruiker op een andere optie klikt na het geven van het antwoord.
6. `antwoordenIndicator()` en `updateantwoordIndicator(markType)`:
  - Deze functies worden gebruikt om een indicator te maken en bij te werken die de gebruiker laat zien welke vragen ze correct of verkeerd hebben beantwoord.
7. `next()`:
  - Deze functie wordt aangeroepen wanneer de gebruiker op de "volgende" knop klikt. Het controleert of de quiz is afgelopen (als alle vragen zijn gesteld) en roept zo nodig de `quizOver()` functie aan. Zo niet, dan roept het `getNewQuestion()` aan om een nieuwe vraag te krijgen.
8. `quizOver()` en `quizResult()`:

- De `quizOver()` functie verbergt de quiz en toont het resultaatenscherf aan de gebruiker. De `quizResult()` functie berekent en toont de resultaten van de quiz aan de gebruiker, zoals het totaal aantal vragen, pogingen, correcte antwoorden, en de score.
9. `resetQuiz()`, `tryAgainQuiz()`, `goToHome()`, en `startQuiz()`:
- `resetQuiz()` wordt gebruikt om de quiz te resetten door alle statistieken terug te zetten naar hun originele waarden en de `availableQuestions` array te legen.
  - `tryAgainQuiz()` verbergt het resultaatenscherf, toont het quizscherf en start de quiz opnieuw door `resetQuiz()` en `startQuiz()` aan te roepen.
  - `goToHome()` verbergt het resultaatenscherf en toont het beginscherf. Het reset ook de quiz door `resetQuiz()` aan te roepen.

#### Quiz JavaScript-file:

Dit JavaScript-bestand, bijvoorbeeld `vlaggen-gemakelijk.js`, bevat een array van objecten genaamd `quiz`. Elk object in deze array vertegenwoordigt een quizvraag en de bijbehorende informatie. Laten we het in detail bekijken:

1. Array van Objecten:
  - `quiz` is een array die meerdere objecten bevat. Elk object binnen deze array representeert een enkele vraag in de quiz.
2. Een enkel vraagobject:
  - Elk vraagobject binnen de `quiz` array bevat vier eigenschappen: `q`, `options`, `antwoord` en `img`.
3. Eigenschappen van een vraagobject:
  - `q`: Dit is de vraag die aan de gebruiker wordt gesteld. In dit geval is de vraag steeds "Welk land is dit?".
  - `options`: Dit is een array van de mogelijke antwoorden die de gebruiker kan kiezen. Elk element in deze array is een string die een mogelijke keuze representeert.
  - `antwoord`: Dit is het indexnummer van het juiste antwoord in de `options` array. Het begint te tellen vanaf 0, dus als antwoord bijvoorbeeld 1 is, dan is het tweede element in de `options` array het juiste antwoord.
  - `img`: Dit is een string die het pad naar de afbeelding voor de vraag representeert. In dit geval is het de afbeelding van de vlag van het land dat de vraag betreft.

In het algemeen, wanneer de quiz wordt uitgevoerd, zal de code een vraagobject uit deze array halen, de vraag, de keuzes en de afbeelding aan de gebruiker tonen, en dan het antwoord van de gebruiker vergelijken met het juiste antwoord dat in het vraagobject is opgeslagen. Dit proces wordt herhaald tot alle vragen zijn gesteld of de gebruiker ervoor kiest om de quiz te beëindigen.

## 4. Gebruiksaanwijzing

Volg de volgende stappen om een van de quizzen te doen:

1. Klik op één van de planeten 'vlaggen', 'omtrek', 'kaart' of 'madness'.
2. Klik vervolgens op een moeilijkheidsgraad voor 'vlaggen', 'omtrek', 'kaart'.
3. Lees de introductie van de oefening en klik op de knop 'Begin oefening'.
4. Vervolgens wordt een vraag getoond met vier mogelijke antwoorden.
5. Als men een antwoord kiest wordt aangegeven of dit juist of fout is. Het is ook mogelijk om de vraag over te slaan.
6. Ga vervolgens door alle vragen. Eens je alle vragen hebt doorlopen kom je op het resultatenschermbord.
7. Op het resultatenschermbord krijg je de eindscore te zien.
8. Vanuit het resultatenschermbord kun je kiezen om de oefening opnieuw te starten of terug te gaan naar het tussen menu of een andere moeilijkheidsgraad te kiezen.

## 5. Conclusie

De applicatie stelt gebruikers in staat om een reeks vragen te beantwoorden en hun resultaten te bekijken. Hieronder volgen de belangrijkste punten:

1. We hebben de structuur van de JavaScript-code besproken, die verantwoordelijk is voor het beheren van de spellogica, waaronder het initialiseren van de quiz, het genereren van nieuwe vragen, het controleren van de antwoorden, en het bijhouden en weergeven van de resultaten.
2. We hebben de structuur en het doel van de quizdata zoals bijvoorbeeld in 'vlaggen-gemakkelijk.js' uitgelegd. Deze data bestaat uit een array van objecten, waarbij elk object een vraag, de bijbehorende opties, het correcte antwoord en een afbeelding vertegenwoordigt.
3. We hebben de structuur van de HTML-pagina onderzocht, die fungeert als de visuele interface van de quiz. Deze pagina omvat verschillende elementen die inhoud dynamisch bijwerken op basis van de acties van de gebruiker en de spellogica.

In de kern is deze applicatie een interactieve ervaring die gebruikers in staat stelt om hun kennis te testen en direct feedback te krijgen. Het illustreert ook hoe HTML, CSS, en JavaScript kunnen samenwerken om dynamische en interactieve webapplicaties te creëren.

## 6. Verwijzingen

Hier zijn enkele referenties die nuttig kunnen zijn voor verdere verkenning of begrip van de technologieën die worden gebruikt in de oefeningen:

- TensorFlow.js: <https://www.tensorflow.org/js>
- Leerbare Machine Afbeelding:  
<https://teachablemachine.withgoogle.com/train/image>
- HTML: <https://developer.mozilla.org/en-US/docs/Web/HTML>
- JavaScript: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- CSS: <https://developer.mozilla.org/en-US/docs/Web/CSS>



# Leer vlaggen met een AI

## 1. Inleiding

De oefening "Learn Flags with AI" is een webgebaseerde applicatie waarmee gebruikers hun kennis van Europese vlaggen kunnen testen. Door gebruik te maken van computervision en machine learning-technieken, kunnen gebruikers de camera van hun apparaat gebruiken om beelden van vlaggen vast te leggen, die vervolgens worden verwerkt door een vooraf getraind model om de weergegeven vlag te voorspellen. De applicatie biedt interactieve functies en feedback om de leerervaring te verbeteren.

## 2. HTML Structuur

De HTML-code definieert de structuur en lay-out van de oefenpagina. Het bestaat uit de volgende belangrijke elementen:

- Koptekstsectie: bevat de paginatitel, tekencodering en viewport-instellingen.
- Externe bronkoppelingen: importeert vereiste CSS- en JavaScript-bestanden voor Bootstrap, Font Awesome, TensorFlow.js en Teachable Machine Image.
- Hoofdgedeelte: Bevat de inhoud van de oefening en de componenten van de gebruikersinterface.
  - Introductie Div: Geeft een inleiding tot de oefening weer, inclusief een beschrijving en een knop "Begin oefening".
  - Oefening Div: Bevat de belangrijkste oefeningselementen, zoals het vlagweergavegebied, videostream, opname- en herkansingsknoppen, spiegelvakje en feedback over resultaten.
  - Resultaten Div: toont de trainingsresultaten, inclusief statistieken en opties om opnieuw op te starten of terug te gaan.
- JavaScript-code: Ingebed in <script>-tags, verwerkt de JavaScript-code de functionaliteit en interactiviteit van de oefening.

## 3. JavaScript-functionaliteit

De JavaScript-code biedt de kernfunctionaliteit van de oefening 'Vlaggen leren met AI'. Het omvat de volgende belangrijke aspecten:

### Video-opname en beeldverwerking

- De camera initialiseren: Wanneer op de knop "Oefening starten" wordt geklikt, vraagt de JavaScript-code toegang tot de camera van het apparaat en initialiseert de videostream.

- **Frames vastleggen:** De gebruiker kan een afbeelding uit de videostream vastleggen door op de opnameknop te klikken. Het vastgelegde frame wordt weergegeven en verwerkt voor vlagvoorspelling.
- **De video spiegelen:** Met het selectievakje mirror kan de gebruiker het spiegelen van de videostream in-/uitschakelen.
- **Beeldvoorspelling:** De vastgelegde afbeelding wordt doorgegeven aan een vooraf getraind machine learning-model (geladen met behulp van Teachable Machine Image) om de weergegeven vlag te voorspellen. De voorspelde vlagklasse wordt vervolgens weergegeven.

## Interactie met de gebruikersinterface

- **Oefenschermen in-/uitschakelen:** De code verbergt of toont verschillende secties van de oefeninterface, zoals de introductie-, oefenings- en resultatendivs, op basis van de voortgang van de gebruiker.
- **Feedback en smiley-indicatoren:** De code biedt realtime feedback aan de gebruiker door "Correcte" of "Onjuiste" berichten weer te geven en smiley-indicatoren bij te werken voor elke beantwoorde vraag.

## Vlagselectie en -score

- **Willekeurige vlagselectie:** De code selecteert willekeurig een vlag uit de opgegeven landenarray om weer te geven als de huidige vlag om te identificeren.
- **Scoren:** Elke keer dat de gebruiker een vraag beantwoordt, vergelijkt de code de voorspelde vlag met het huidige land en werkt de score van de gebruiker dienovereenkomstig bij.
- **Scoreweergave:** de huidige score van de gebruiker wordt op de pagina weergegeven.

## Resultaten berekenen en weergeven

- **Resultaatberekening:** Nadat de gebruiker alle vlaggen heeft geïdentificeerd, berekent de JavaScript-code de eindscore van de gebruiker door het aantal juiste antwoorden te vergelijken met het totale aantal vragen.
- **Resultaatweergave:** De trainingsinterface wordt bijgewerkt om de eindscore van de gebruiker weer te geven, samen met een bericht dat zijn prestaties aangeeft (bijv. "Geweldig werk!" of "Blijf oefenen!"). De resultaten bevatten ook opties om de oefening opnieuw te starten of terug te gaan naar het begin.

## 4. Gebruiksaanwijzing

Volg deze stappen om de oefening 'Vlaggen leren met AI' te gebruiken:

1. Open het HTML-bestand in een webbrowser die JavaScript ondersteunt.

2. Verleen toestemming voor toegang tot de camera van uw apparaat wanneer daarom wordt gevraagd.
3. Lees de introductie van de oefening en klik op de knop "Begin oefening".
4. De oefeninterface wordt weergegeven, met een vlag en de videostream.
5. Plaats de vlag in het zicht van de camera en klik op de opnameknop om de afbeelding te verwerken.
6. De voorspelde vlag wordt weergegeven, samen met feedback over of het antwoord juist of onjuist was.
7. Ga door met het vastleggen van afbeeldingen van vlaggen totdat alle landen zijn bedekt.
8. Zodra alle vlaggen zijn weergegeven, worden de trainingsresultaten weergegeven, inclusief de eindscore en het prestatiebericht.
9. Vanuit het resultatenschermbord kun je ervoor kiezen om de oefening opnieuw te starten of terug te gaan naar het begin.

## 5. Conclusie

De oefening "Learn Flags with AI" laat zien hoe computervisie en machine learning-technieken kunnen worden gebruikt om een interactieve en boeiende leerervaring te creëren. Door videooverwerking te combineren met een vooraf getraind model, kunnen gebruikers hun vlagidentificatievaardigheden oefenen met behulp van de camera van hun apparaat. De oefening biedt realtime feedback, het bijhouden van scores en een uitgebreide weergave van de resultaten.

Deze documentatie biedt een overzicht van de HTML- en JavaScript-code voor de oefening en legt de structuur en functionaliteit ervan uit. Gebruikers kunnen deze documentatie raadplegen om te begrijpen hoe de code werkt en de nodige wijzigingen of verbeteringen aan te brengen.

## 6. Verwijzingen

Hier zijn enkele referenties die nuttig kunnen zijn voor verdere verkenning of begrip van de technologieën die worden gebruikt in de oefening "Vlag leren met AI":

- TensorFlow.js: <https://www.tensorflow.org/js>
- Leerbare Machine Afbeelding: <https://teachablemachine.withgoogle.com/train/image>
- HTML: <https://developer.mozilla.org/en-US/docs/Web/HTML>
- JavaScript: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- CSS: <https://developer.mozilla.org/en-US/docs/Web/CSS>

# Leer de vlaggen met Geolocator

## 1. Inleiding

De oefening "geolocator" is een webgebaseerde applicatie waarmee gebruikers hun kennis van Europese vlaggen en landen kunnen testen. Door gebruik te maken van een interactieve kaart kunnen gebruikers de landen van Europa leren op basis van zowel namen als vlaggen afhankelijk van de gekozen moeilijkheidsgraad. Er is een unieke leerervaring op de applicatie door de simpliciteit van de gebruikersinterface.

## 2. HTML Structuur

De HTML-code definieert de structuur en lay-out van de applicatie. Deze is opgebouwd uit deze kernelementen:

- Header tags: hier staat de titel van de pagina en de imports voor de styling en functionaliteitsdocumenten
- SVG: dit is de afbeelding van de kaart van europa die interactief wordt gemaakt
- Oefening: bevat alle deelcomponenten om de pagina te laten werken
  - Start scherm: dit is het scherm met de basisuitleg over de werking van de oefening
  - Hoofd scherm: dit is het scherm dat tijdens de oefening te zien is en bestuurt het grootste deel van de applicatie
  - Eind scherm: toont de resultaten van de oefening met de optie om naar het startscherm terug te gaan of de oefening opnieuw te proberen.
- Overige script tags: deze geven de pagina de nodige functionaliteit en verwijzen naar de bestanden die de desbetreffende code bevatten.

## 3. JavaScript-functionaliteit

Alle JavaScript bestanden zijn vereist voor de functionaliteit van de pagina, voornamelijk de interactieve kaart en de quiz-applicatie

### Interactieve kaart

- Instellingen van de kleur van elk land: dit gebeurt in een apart bestand genaamd map-config
- Interactiviteit toevoegen: dit word gedaan door het map-interact bestand en de SVG op de webpagina en geeft de gebruiker een kleurverandering als de muis over een land beweegt en natuurlijk de klik functie
- Geselecteerd land doorgeven: dit gebeurt door een combinatie van zowel map-interact en quiz-app-flag of quiz-app-name

## Interactie met de gebruikersinterface

- Dit gebeurt door een versie van quiz-app en map-interact, waarbij quiz-app de vragen en antwoorden, net als de juiste menu's te tonen, en map-interact zorgt voor het kunnen selecteren van een land

## Configuratie

Elk land krijgt een eigen item met 7 verschillende eigenschappen.

- Hover: hierin staat de landnaam, deze wordt meegegeven naar de quiz applicatie als het land geselecteerd wordt
- Url: hierin is er een optie om een url te plaatsen naar een website, maar dit staat uit
- Target: dit is deel van het url systeem, en bepaald of bij het openen van een link al dan niet een nieuw tabblad wordt gebruikt
- UpColor: deze variabele bepaald de standaard kleur van het land
- OverColor: bepaald de kleur van het land als er een muis over beweegt
- DownColor: bepaald de kleur van het land als er op is geklikt

## Resultaten berekenen en weergeven

Op het einde van de quiz worden de resultaten berekend door de quiz-app.

## 4. Gebruiksaanwijzing

Het gebruik van deze oefening is vrij eenvoudig, maar hierbij een korte uitleg:

1. Open het de website in een webbrowser (zoals chrome)
2. Eenmaal op de thuispagina aangekomen, selecteer de planeet waar geolocator op staat
3. Nu kan de keuze van moeilijkheidsgraad gemaakt worden, daarna gaat het programma naar het beginscherm
4. Klik op begin oefening
5. Het programma geeft nu een landnaam of vlag, selecteer dit land op de kaart, hierna gaat de applicatie automatisch naar de volgende vraag
6. Eenmaal alle vragen beantwoord of overgeslagen zijn komt het resultatscherm tevoorschijn
7. Op het eindscherm kun je ervoor kiezen om de oefening opnieuw te starten of terug te gaan naar het menu van de moeilijkheidsgraad

## 5. Verwijzingen

Enkele referenties voor meer informatie over de gebruikte bestanden

- HTML: <https://developer.mozilla.org/en-US/docs/Web/HTML>
- JavaScript: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- CSS: <https://developer.mozilla.org/en-US/docs/Web/CSS>

# Hosting van de applicatie

## Inleiding

De website draait op een domeinnaam van Vimexx, wordt dan naar Cloudflare doorgestuurd en draait dan op een eigen server, maar in de handleiding wordt ook uitgelegd hoe de website op netlify gehost kan worden.

## Domeinnaam

De domeinnaam is aangekocht op Vimexx, en vanaf hen wordt deze doorverwezen naar Cloudflare. Dit gebeurt door de nameservers van de domeinnaam aan te passen naar deze die Cloudflare opgeeft bij het toevoegen van een website.

## DNS

De DNS voor landleren is Cloudflare. Hiervoor is een account vereist, en eenmaal dat dit is aangemaakt kan een website worden toegevoegd. Klik hiervoor eerst op “Add a Site”, voer vervolgens de domeinnaam in en klik opnieuw op “Add site”. Vervolgens wordt er gevraagd om een plan te kiezen, en hiervoor is normaal een gratis plan goed genoeg. Hierna is het domein toegevoegd, en kunnen de dns instellingen aangepast worden om de domeinnaam aan de servers te koppelen. Dit kan wel pas gedaan worden als de hosting is ingesteld.

## Hosting

### Windows Server

In dit project werd origineel een IIS server op Windows Server 2022, omdat het hierbij makkelijker was om de website live aan te passen tijdens ontwikkeling. Als uw hosting of de eventuele eigen servers geen windows server draaien, ga dan verder naar NGINX.

Om de website in IIS te draaien moet een nieuwe website toegevoegd worden, of de huidige worden aangepast. Het model van deze website gebruikt 2 aparte websites om op grotere schaal de laadtijden te verminderen door de static pagina op een CDN of een snellere server te hosten. Als dit niet per se vereist is of de bandbreedte van de server groot genoeg is, kan de website met enkele aanpassingen weer samengevoegd worden. Voor de aparte websites moeten 2 webserver aangemaakt worden, anders is 1 voldoende.

Het instellen van de webserver is afhankelijk van het scenario waarin de server draait. In het netwerk waar de demo draait is er nog een nginx proxy, dus worden de domeinnamen en SSL daar ingesteld, en werd op de Windows server alleen 2 websites op poort 81 en 82 aangezet. Het is echter mogelijk om zowel de domeinnamen als SSL in te stellen met IIS, daarover meer informatie op <https://enterprise.arcgis.com/en/web-adaptor/10.3/install/iis/enable-https-on-your-web-server-server-.htm>.

## NGINX

In de demo setup werd nginx gebruikt als reverse proxy om de domeinen naar de juiste webserver te sturen en SSL toe te voegen (https), maar kan volledig ingesteld worden om alles rond de website te hosten. Dit deel is heel specifiek in te stellen, dus gaat dit document hier niet verder op ingaan. Meer informatie hierrond staat op de documentatie van nginx zelf: <https://nginx.org/en/docs/>

## Afronden

Als de hosting is ingesteld, moet de domeinnaam gekoppeld worden. Hiervoor moet je het IP van de hostende server hebben - dit is op de meeste online hostings makkelijk terug te vinden op het webportal, en voor eigen servers kan dit terug gevonden worden door naar <https://whatismyip.com> te gaan. Dit IP moet dan ingevuld worden in de dns records (type A) met als naam @, www en static.