

# Oplevering Strijp-T Hotspots

Max van Hattum – 16-6-2021

## Inhoudsopgave

|                                    |   |
|------------------------------------|---|
| Introductie .....                  | 1 |
| Huidige infrastructuur .....       | 2 |
| Git Repo en Folder Structuur ..... | 2 |
| Variabelen aanpassen .....         | 3 |
| Back-end .....                     | 3 |
| Front-end .....                    | 3 |
| Google Maps API .....              | 4 |
| Docker .....                       | 4 |

## Introductie

Voor Strijp-T is een Hotspots App gerealiseerd, een interactieve plattegrond. Deze applicatie bestaat uit een front-end gemaakt met Vue.js en een back-end. De back-end is gemaakt met ASP.NET Core en een database, MySQL.

Om de applicatie te deployen is Docker geconfigureerd, waardoor in een paar stappen de gehele applicatie gebouwd en deployed kan worden.

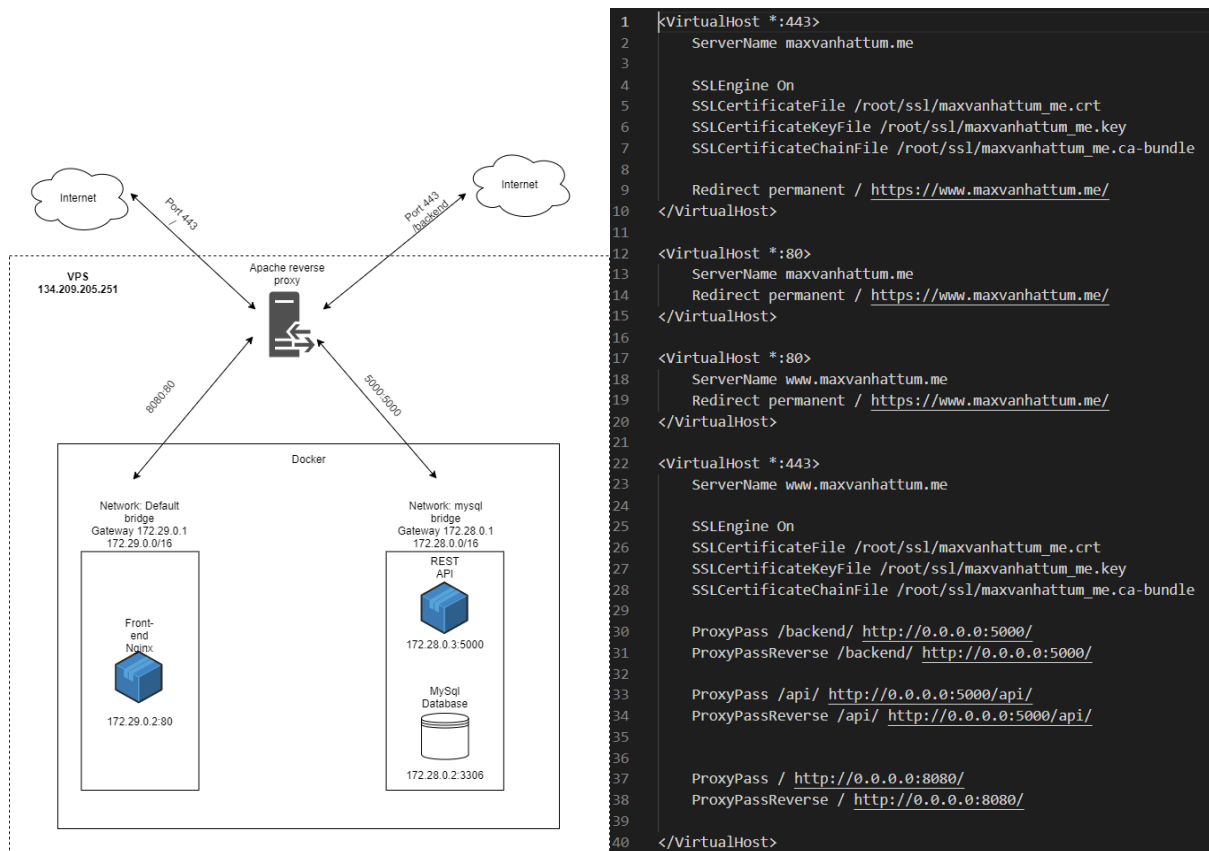
Op het moment van schrijven is de applicatie gedeployed op een VPS bereikbaar via [www.maxvanhattum.me](http://www.maxvanhattum.me), hier draaien de front- en back-end op. Een reverse proxy regelt de encryptie en het routen van de requests.

Het is mogelijk om de applicatie op de zelfde manier te installeren, maar ook los van elkaar, door de individuele Dockerfiles te gebruiken, dit vereist wel andere configuratie.

Er moet een Google Maps API key aangevraagd worden waarin de Maps JavaScript API en Directions API bruikbaar zijn.

## Huidige infrastructuur

Onderstaand een schets van de huidige opzet. Ook de apache configuratie.



## Git Repo en Folder Structuur

De code is beschikbaar op: <https://github.com/Maxvanhattum/strijp-t-hotspots>

De folder structuur spreekt voor zich, in de root folder staan twee folders waarin de back-end en front-end staan. Hierin staat ook een docker-compose.yml en een .env file. In de .env file staan de variabelen die Docker gebruikt voor het configureren van de back-end.

| Naam               | Gewijzigd op    | Type             | Grootte |
|--------------------|-----------------|------------------|---------|
| back-end           | 16-6-2021 09:39 | Bestandsmap      |         |
| front-end          | 16-6-2021 09:39 | Bestandsmap      |         |
| .env               | 19-3-2021 15:39 | ENV-bestand      | 1 kB    |
| docker-compose.yml | 19-3-2021 14:52 | Yaml Source File | 2 kB    |

## Variabelen aanpassen

Voor het deployen moeten er een aantal variabelen aangepast worden, voor de back-end staan deze in een `.env` file. Voor de front-end moeten deze in twee files aangepast worden.

### Back-end

Onderstaand alle variabelen die Docker gebruikt.

```
1 FRONTEND_URL=https://www.maxvanhattum.me
2 MYSQL_ROOT_PASSWORD=rand!om!pass123
3 MYSQL_USER=restapi
4 MYSQL_PASSWORD=pass0909!@
5 MYSQL_DATABASE=hotspot
6 BACKEND_URL=www.maxvanhattum.me/backend/
7 SWAGGER_PREFIX=/backend/
```

- De `FRONTEND_URL` moet matchen met de URL waarop de front-end te bereiken is.
- `BACKEND_URL` moet matchen met de URL waarop de back-end te bereiken is.
- `SWAGGER_PREFIX` moet ingesteld staan als de back-end niet op de root te bereiken is, anders moet dit `"/` zijn.

De MySQL user, password en root password kunnen gerandomized worden met oog op veiligheid.

### Front-end

In de front-end moeten twee waardes aangepast worden in files. Ten eerste de URL waarop de back-end te bereiken is. Deze staat in `front-end->src->services->API->API.js` op regel 4.

```
front-end > src > services > API > JS API.js > axiosConfig > baseUrl
1 import axios from "axios";
2
3 const axiosConfig = {
4   baseUrl: "https://newWebsiteURLHere.me/api/",
5   timeout: 30000
6 };
```

Ten tweede de Google Maps API key ingevoerd worden, in `front-end->src->utils->googlemap.js` op regel 1.

```
front-end > src > utils > JS googlemap.js
```

```
1 const API_KEY = "HIERDENIEUWEAPIKEY";
```

## Google Maps API

De Google Maps API heeft een maandelijks gratis krediet van 200\$, ruimschoots genoeg voor deze applicatie. Hiervoor worden twee API's gebruikt, de Google Maps Javascript API en de Directions API.

Eerst moet er op Google Cloud console een project aangemaakt worden:

<https://developers.google.com/maps/documentation/javascript/cloud-setup?hl=nl>

Hierna kunnen de API's enabled worden:

<https://console.cloud.google.com/apis/library/maps-backend.googleapis.com>

<https://console.cloud.google.com/apis/library/directions-backend.googleapis.com>

## Docker

Om beide applicaties tegelijkertijd op te starten kan in de root folder met de *docker-compose.yml* de volgende command uitvoerd worden:

```
docker-compose up
```

Nu is de applicatie draaiend. De back-end is bereikbaar op port 5000, de front-end op port 8080.