# Onderzoek Heatmap | Ontdekstation 013

# Map Description automatically generated

*Afbeelding: Leaflet-Heatmap-Layer-Example.*

Contents

[Onderzoek Heatmap | Ontdekstation 013 1](#_Toc115697694)

[1- Context 2](#_Toc115697695)

[2- Mogelijkheden 2](#_Toc115697696)

[3- Voorbeeld 3](#_Toc115697697)

[3.1- intro 3](#_Toc115697698)

[3.2- Eerste Uitwerking 3](#_Toc115697699)

[3.3- Conclusie Eerste Uitwerking 3](#_Toc115697700)

[3.4- Combinatie informatiebronnen 3](#_Toc115697701)

[3.5- Privacy 4](#_Toc115697702)

[3.6- Implementatie bestaande frontend 6](#_Toc115697703)

[4- Conclusie 6](#_Toc115697704)

[References 7](#_Toc115697705)

## Context

Functioneel requirement 10 van het project geeft aan dat de gebruiker gegevensdistributie middels een heatmap moet kunnen inzien. In dit document zal ik onderzoeken hoe dat binnen de kaders van dit project gerealiseerd kan worden.

De hoofdvraag voor dit onderzoek luidt dan ook:

* *Hoe kan een Heatmap gerealiseerd worden binnen de applicatie?*

Hiervoor is de vorige groep al begonnen met het gebruik van Leaflet en OpenStreetMap. OpenStreetMap is een Open Source wereldkaart. Leaflet betreft een open source Javascript library voor het bewerken van de kaart geleverd door OSM.

Dit roept de volgende deelvraag op:

* *Kan de heatmap geïntegreerd worden in het bestaande kaartensysteem van de applicatie? (React Leaflet)*

De vorige projectgroep heeft met deze bronnen al een systeem opgezet voor het weergeven van losse datapunten op de kaart. Wij gaan deze sprint onderzoek doen naar de uitbreiding van deze functionaliteit met een *Heatmap*.

Hier volgt dan ook de tweede deelvraag:

* *Kunnen meerdere informatiebronnen gecombineerd worden in één kaart voor een volledig beeld van de weerssituatie?*

Verder speelt privacy in het huidige digitale tijdperk een steeds grotere rol. Het moet voor de gebruikers van het systeem dan ook mogelijk zijn om wel data bij te dragen, terwijl hun privacy nog gewaarborgd wordt. Een belangrijke deelvraag luidt dan ook:

* *Kan de Heatmap zo ingesteld worden dat de exacte locatie van meetstations verborgen blijft?*

## Mogelijkheden

Een optie die redelijk eenvoudig is te implementeren in het bestaande systeem is de *react-leaflet-heatmap-layer.* (Js, 2018)Deze library voegt een extra ‘laag’ toe aan een bestaande react-leaflet kaart. De heatmap layer neemt een lijst met coördinaten en waarden en zet deze om naar een heatmap.

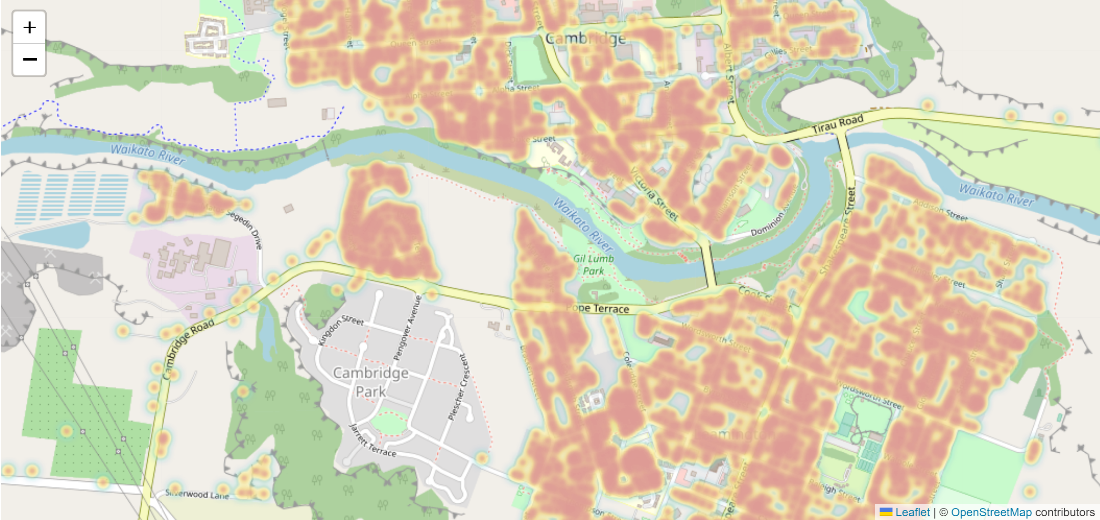
## Voorbeeld

### 3.1- intro

Om te onderzoeken of de Heatmap Layer geschikt is voor onze toepassing heb ik het voorbeeldproject van de Leaflet Heatmap Layer Github getest. Hiervoor diende slechts de repository gecloned en het command ‘npm install’ gerund te worden. Hierbij kwam er al wel een nadeel van de library om de hoek kijken: het project is voor het laatst 4 jaar geleden geüpdate, waardoor er nog dependencies op oudere React-versies inzitten. Hierdoor worden een aantal onderdelen als depricated aangegeven. Hier ben ik echter nog geen verdere praktische problemen mee tegengekomen.

### 3.2- Eerste Uitwerking

De uitwerking van de library lijkt perfect voor het doel dat wij er mee voor ogen hebben: het visualiseren van weerdata. Op basis van de ingevoerde datapunten kan er een heatmap geproduceerd worden. Bij in-en uitzoomen beweegt deze dynamisch mee. Tevens zijn er meerdere belangrijke parameters die aangepast kunnen worden, waardoor de user experience gefinetuned kan worden.



*Afbeelding 1: Overzichtsfoto van de example heatmap.*

### 3.3- Conclusie Eerste Uitwerking

Bovenstaande implementatie geeft antwoord op de eerste deelvraag: het voorbeeld geeft namelijk duidelijk weer dat een Heatmap implementatie met Leaflet en OpenStreetMap mogelijk is.

### 3.4- Combinatie informatiebronnen

Voor het vormen van een volledig beeld is het van belang dat meerdere manieren van informatiecommunicatie in de kaart verwerkt kunnen worden. Zo moet een gebruiker op de heatmap nog altijd de individuele stations kunnen inzien.

De vorige groep heeft voor het weergeven van de meetpunten gebruik gemaakt van ‘Markers’ en ‘Popups’ uit React Leaflet. Om te onderzoeken of dit gecombineerd kan worden met de Heatmap heb ik een enkele marker toegevoegd aan de Heatmap kaart.

Text

Description automatically generated

*Afbeelding 2: Toegevoegde Marker/Popup combinatie aan de Heatmap example.*

Dit geeft het volgende resultaat:

Map

Description automatically generated

*Afbeelding 3: Uitwerking van Marker/Popup op de example Heatmap kaart.*

Hoewel de bijpassende afbeelding uiteraard ontbreekt, geeft dit wel duidelijk weer dat het mogelijk is om informatiecommunicatietechnieken te combineren in React Leaflet.

### 3.5- Privacy

Voor de laatste deelvraag staat de privacy van de meetstation-eigenaren centraal: *kan de Heatmap zo ingesteld worden dat de exacte locatie van meetstations verborgen blijft?*

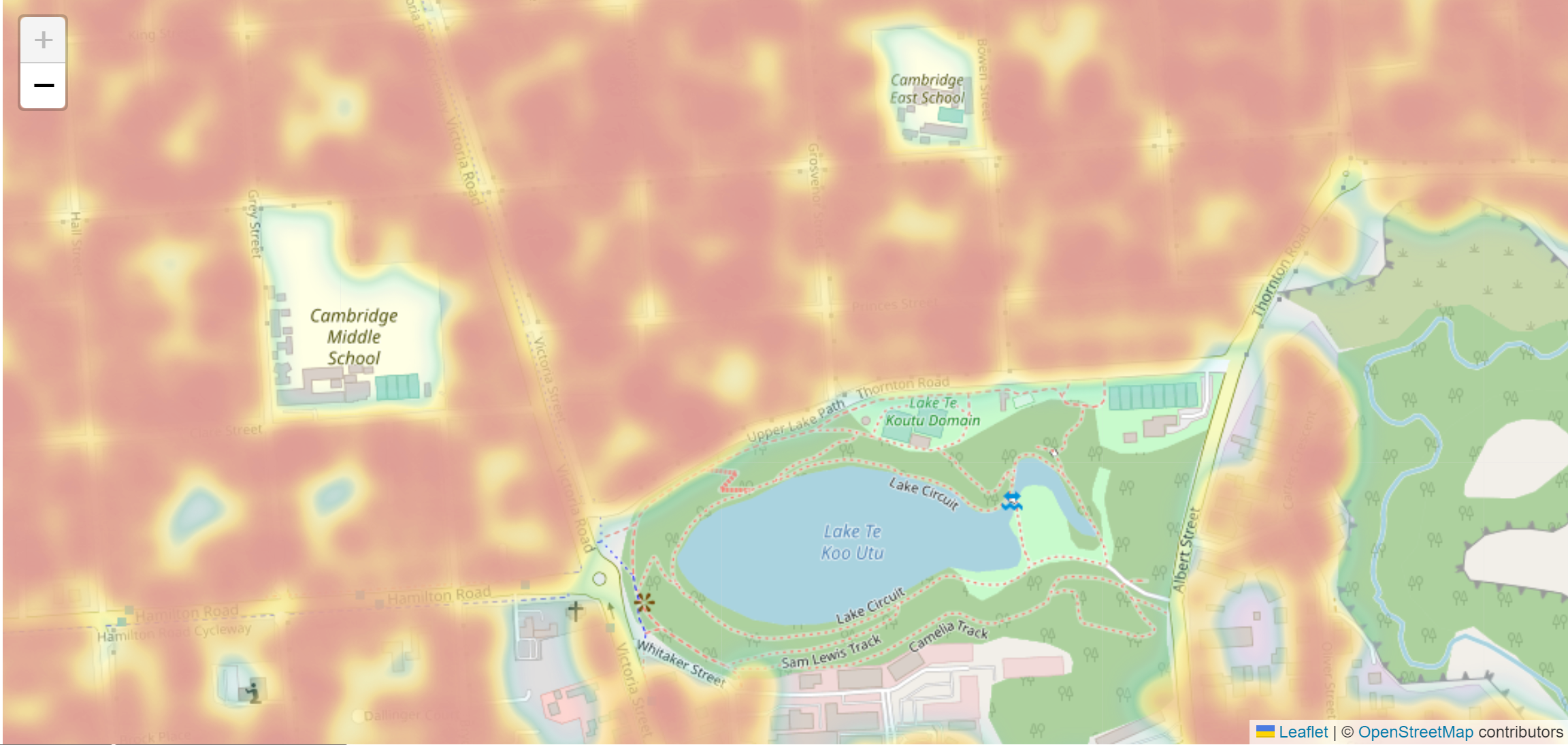
Bij het onderzoeken van deze vraag kwam een minpunt van de leaflet-heatmap-layer om de hoek kijken: het gedrag van de heatmap bij zoomen is niet aan te passen. Hoewel de exacte locatie van stations redelijk goed gemaskeerd kan worden door gebruik te maken van *blur* en *radius* opties, worden ze nog altijd duidelijk zichtbaar als er ver genoeg wordt ingezoomd.

Hiervoor is een oplossing redelijk voor de hand liggend: het maximale zoom-niveau beperken. Ver genoeg dat de gebruiker specifieke informatie over zijn buurt/wijk kan inzien, maar nog voldoende afstand voor het waarborgen van de privacy van meetstation houders. Deze opties zijn ingebouwd in Leaflet en dus goed toe te passen.

Chart, bubble chart

Description automatically generated

*Afbeelding 4: Heatmap zonder maximale zoom ingesteld. De losse meetpunten zijn duidelijk zichtbaar op hun exacte locatie.*



*Afbeelding 5: Heatmap met een maximum zoom ingesteld (zie zoom-knop linksboven).*

### 3.6- Implementatie bestaande frontend

Om de Heatmap-layer werkend te krijgen met de huidige frontend heb ik een aantal zaken aangepast:

* Van React-Leaflet moet versie 2.0.0 gebruikt worden in plaats van 3.2.0.

*De versie downgrade is een gevolg van compatibility-problemen tussen pre- en post React-Leaflet 3.0.0. Versie 3.0.0 is een complete re-write, waarin veel functionaliteiten fundamenteel veranderd zijn.* (PaulLeCam, 2020)

*In het eerste opzicht lijken er voor de huidige functionaliteit geen problemen te ontstaan met de downgrade van React-Leaflet, en zorgt het ervoor dat de Heatmap, gebaseerd op React-Leaflet 2.0.0, gebruikt kan worden. Tevens zijn er tot op heden geen beveiligingsproblemen bij het downgraden van React-Leaflet.*

* De data voor de heatmap moet beschikbaar zijn in een Array, in de vorm [longitude, latitude, value].
* In plaats van ‘MapContainer’ wordt ‘Map’ gebruikt uit Leaflet voor het weergeven van de leaflet. Dit is een gevolg van de versie-downgrade.
* De HeatMapLayer moet aan de ‘Map’ toegevoegd worden.

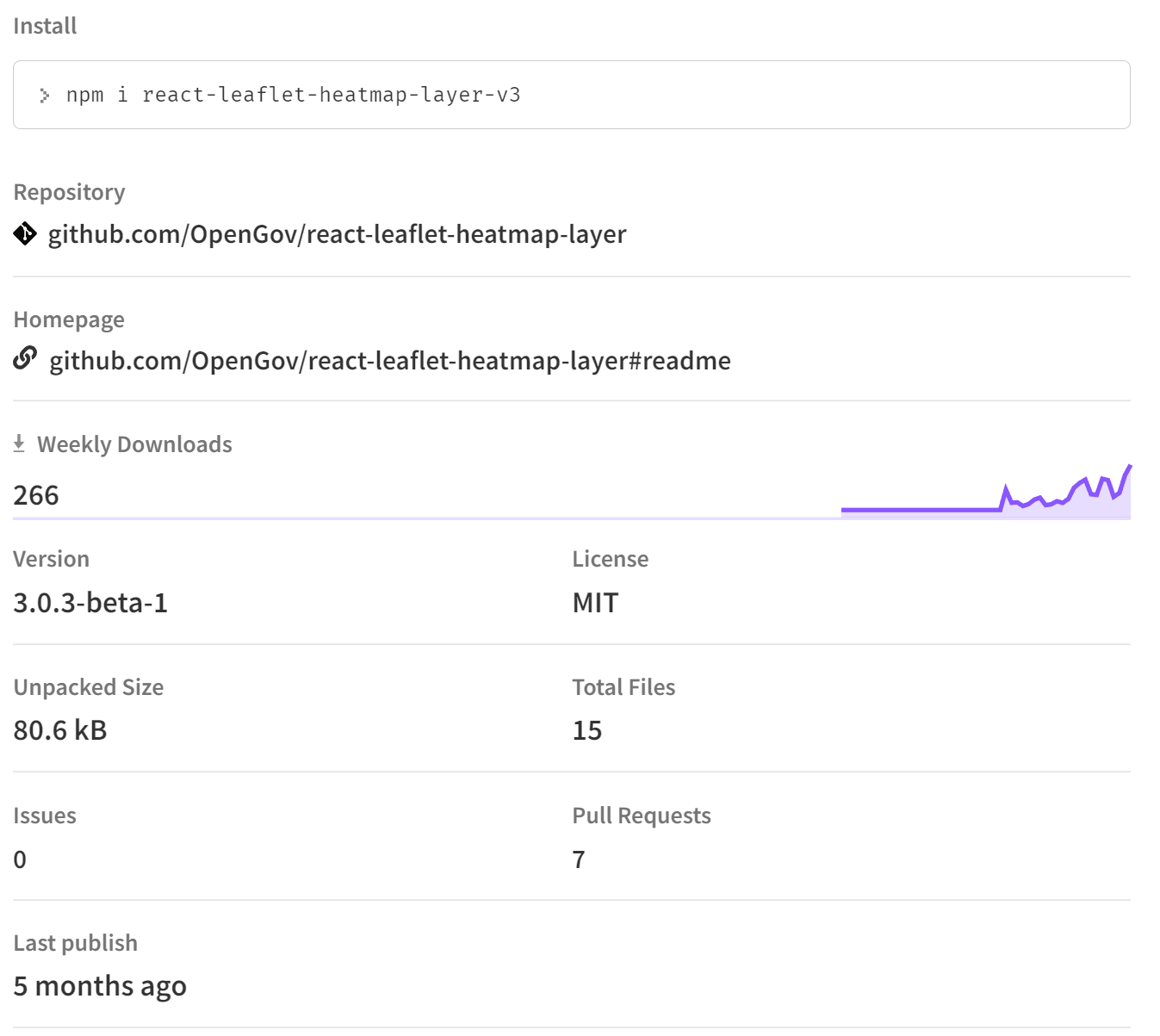
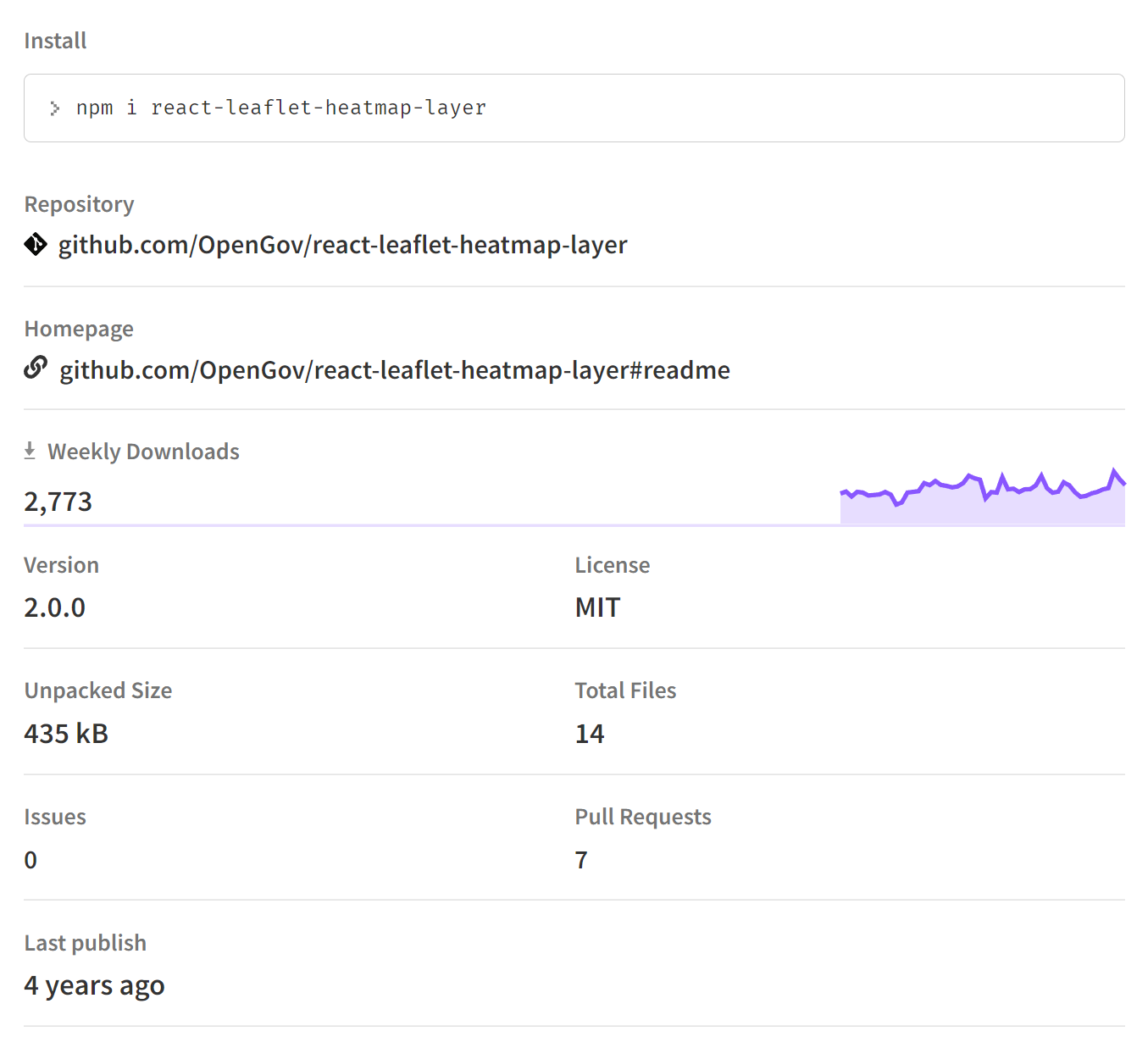
Alle stappen zijn reeds getest, met uitzondering van de weergave van actuele data ontvangen uit de backend. Dit vereist meer onderzoek naar de werking van de communicatie tussen de front-en backend.

### 3.7- Verschillende versies

Bij het opzoeken van de react leaflet heatmap layer komen er twee pagina’s naarboven: de originele, en een die recent nog geüpdate is. Dit geeft het idee dat er toch een nieuwere versie van de layer is, maar dit is helaas niet het geval.

Hoewel de ‘nieuwere’ versie claimt geüpdate te zijn, verwijzen beide pagina's naar dezelfde Git: die van de originele, oude versie.

Aangezien er dus geen source code van de nieuwe versie bekend is gaan we hier niet mee werken. Overigens blijkt uit de NpmJs pagina dat er slechts van functies veranderd zouden zijn, maar dat de oudere react-leaflet versie nog altijd vereist is.



*Afbeeldingen:*

*links: de oude, originele versie van de react-leaflet-heatmap-layer*

*Rechts:de nieuwere, ‘beta’ versie van de react-leaflet-heatmap-layer*

*Zoals te zien is bij ‘repository’ verwijzen ze beide naar de originele git.*

## Conclusie

Uit dit onderzoek is gebleken dat er een bestaande, open source techniek beschikbaar is die gecombineerd kan worden met de huidige informatiecommunicatietechniek (‘Markers’) in het reeds opgezette kaartensysteem, namelijk React Leaflet en OpenStreetMap. Tevens is het duidelijk geworden dat de privacy van meetstationhouders gewaarborgd kan worden. Hoewel de exacte implementatie in de bestaande front-end nog wat werk vereist, is het duidelijk geworden dat het wel degelijk mogelijk is.

# References

Js, N. (2018). *react-leaflet-heatmap-layer*. Retrieved from NPM Js: https://www.npmjs.com/package/react-leaflet-heatmap-layer

PaulLeCam. (2020, october 31). *React Leaflet Releases*. Retrieved from Github: https://github.com/PaulLeCam/react-leaflet/releases?page=2