

Final Project Documentation  
*Home-to-Numbers*

by Humam Houara

Midterm Paper  
CART 451 - Network Media Studio  
Concordia University  
December 09, 2022

To make such a project that involves maps, and i was very unfamiliar with its technologies, I had a lot of exploring and researching to do. After exploring many options I decided to go with leaflet. The Javascript library that was able to give me so much freedom and a variety of maps to choose from. Also, its documentation was very clear and easy to understand, with lots of resources online that helped me.

The base maps that i ended up using is a map called '`DarkGray`' which I really liked its aesthetics and can be impeded easily on the web, also features responsive design. Lastly, for the animation of the lines I wanted to use, to show an animation of movements and connect the dots of the dataset, I used a beautiful library called Tween.JS. as well as the basic JQuery library.

I also had to use some basic HTML and CSS to build the framework of the project. I thought about doing the project using react, but I found it much simpler when dealing with many Javascript libraries to use HTML and CSS.

For the interactive story method, I wanted refugees to post their stories in the most accessible way possible so I created a google spreadsheet that I made where they can write their stories and prepared HTML ids so they have options of uploading images, videos, and links along with the countries longitude and latitude of their host country. When users write on the google spreadsheet, and it updates automatically on the website, Using google's API.

All the data gathered and used was from the UNHCR website, the only reliable source I was able to use, they also offer a page full of stories of refugees. The data sets can be fully customized through the UNHCR website. I was able to get a query that is called One-to-many. So it is the countries and all the refugees that left from it, and where they all went to, by cities.

I also used Figma to make some wireframes and mockups to envision how the project would look.

Lastly, thankful to this learning process I was able to stumble across multiple softwares and technologies that I am definitely going to use in the process. Plotly, is a software for building and deploying data in apps built with Python, Tailwind, and CSS framework, building websites without leaving HTML. and many map APIs

Some of the features I was able to implement were able to visualize numbers of a big data set to convey the message and make it clickable with it being interactive. Another feature is the reading of the stories map view, when the user scrolls the map moves to the scrolled story. Some of the features that I wanted to implement but could not either because of time constraints or because I simply could not do them. Which was to link the stories dataset JSON file to the main refugee visualization. Another feature is having a label of what the numbers are when hover-over the dot/circle. Another feature I wanted to implement is an HTML grabber from the UNHCR website to update the stories automatically.

Not being able to have the hover over number details was a blessing in disguise. Having this broad abstract of the map, kinds of serves the message of the concept. The more I worked on this project and the more I read about the problem, I became more confident in my choice of topic. When I collected the data and saw how massive this issue is yet no one is aware of it, felt like this project is made for a purpose that is serving the greater good which is really important to me.

## Annex

```
<script>
  var map = L.map('map');

  if (L.Browser.mobile) {
    map.setView([15, -21.95], 2);
  } else {
    map.setView([0, 0], 2);
  }

  L.esri.basemapLayer('DarkGray').addTo(map);

  Papa.parse('../csv-data/Flowmap_Cities_one_to_many.csv', {
    download: true,
    header: true,
    dynamicTyping: true,
    skipEmptyLines: true,
    complete: function(results) {
      var geoJsonFeatureCollection = {
        type: 'FeatureCollection',
        features: results.data.map(function(datum) {
          return {
            type: 'Feature',
            geometry: {
              type: 'Point',
              coordinates: [datum.s_lon, datum.s_lat]
            },
            properties: datum
          }
        })
      };
    }
  });
};
```

### *Setting up the map*

```
},
pathDisplayMode: 'selection',
animationStarted: true,
animationEasingFamily: 'Cubic',
animationEasingType: 'In',
animationDuration: 2000
}).addTo(map);
```

### Animation code

```

<div id="narration">
  <div id="contents">
    <div id="top"></div>
    <a href=".../docs/main/index.html"> <button> View Map</button></a>
  </div>
</div>

<!-- <div class="loader">Loading...</div> -->
<div id="map"></div>

<script type="text/javascript">
  // Create the Leaflet map with a generic start point
  var map = L.map('map', {
    center: [0, 0],
    zoom: 1,
    scrollWheelZoom: false,
    zoomControl: false,
    tap: false
  });
</script>

```

Adding zoom-in/out buttons controls

```

1  body {
2      margin: 0;
3      padding: 0;
4  }
5
6  #map {
7      position: absolute;
8      top: 0;
9      bottom: 0;
10     right: 0;
11     left: 0;
12 }
13 #side-bar {
14     display: inline-block;
15     position: absolute;
16     height: 10000px;
17     width: 25%;
18     z-index: 314159;
19     background-color: ■ rgba(255,255,255, 0.86);
20     opacity: 50%;
21 }
22 h1 {
23     font-family: PT Sans Narrow, Arial;
24     font-weight: normal;
25     margin-bottom: 0;
26     color: ■ gray;
27     font-size: 30px;
28 }
29 h2{
30     font-family: PT Sans Narrow, Arial;
31     font-weight: normal;
32     margin-bottom: 0;
33     color: □ black;
34     font-size: 40px;
35 }
36
37

```

## CSS

```

1  //google sheets
2  var googleDocURL = 'https://docs.google.com/spreadsheets/d/18vDYVbiweir0SCTNfwl20X-K7xSweApWe0kv0DoXpc/edit#gid=0';
3
4  //google api
5  var googleApiKey = 'AIzaSyAtJarj0kmrj8KxD8kHrTjwhsEMiCTwJ94';
6

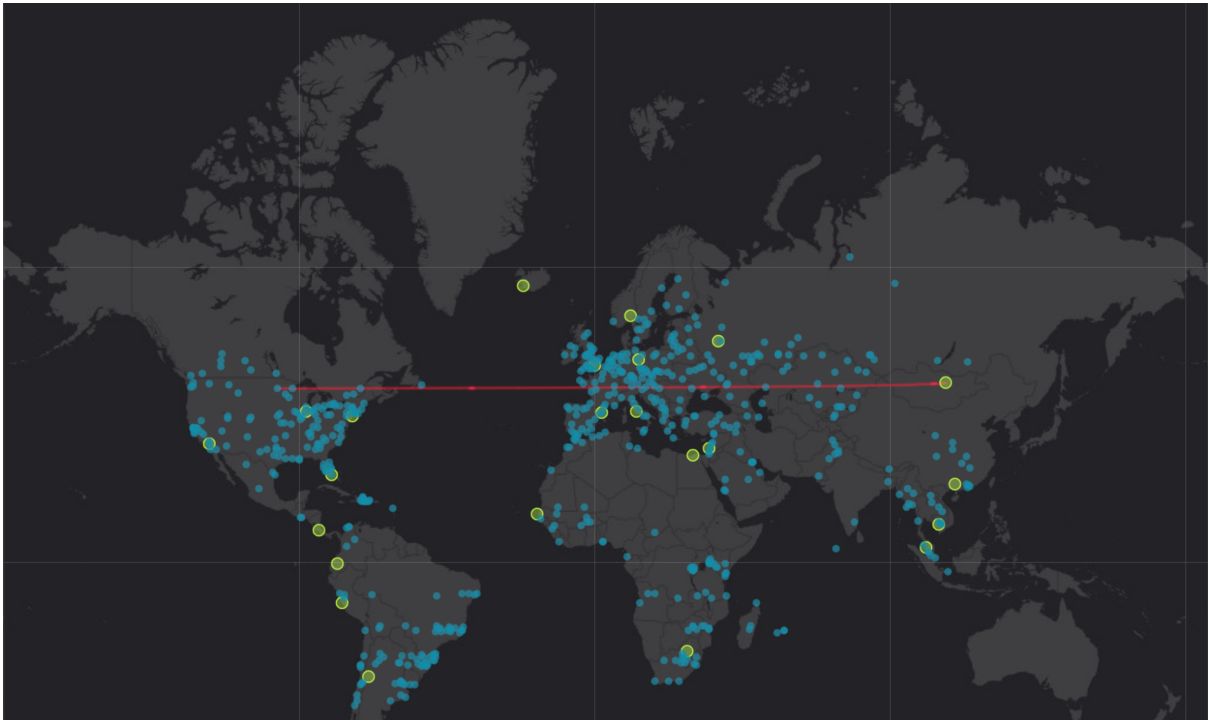
```

Google's doc URL and the API key

96	Mar del Plata	-38.00002033	-57.57998438	554916	Argentina	20	842	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
97	Marbella	36.51661989	-4.88333012	153069.5	Spain	27	968	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
98	Marietta	33.95561342	-84.54324813	61360	United States of America	23	733	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
99	Maseru	-29.31667438	27.48327307	239839.5	Lesotho	36	678	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
100	Masvingo	-20.05961668	30.8200203	76300.5	Zimbabwe	15	505	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
101	Mataro	41.53995668	2.45002071	149826	Spain	4	815	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
102	Matsue	35.46699404	133.0666475	150527	Japan	14	844	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
103	Matsumoto	36.2404352	137.9700175	217796.5	Japan	4	629	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
104	Matsuyama	33.84554262	132.765839	525089	Japan	36	673	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
105	Medenine	33.399999	10.41669956	61705	Tunisia	11	695	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
106	Medina	24.49998903	39.5800024	1010000	Saudi Arabia	14	787	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
107	Melbourne	28.08331036	-80.60832035	170870	United States of America	24	602	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
108	Melbourne	-37.82003131	144.9750162	2131812.5	Australia	11	740	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
109	Metairie	29.98386619	-90.15277653	270171	United States of America	20	506	579	Rome	41.89595563	12.48325842	1687226	Italy	800446	12
110	Tacoma	47.21131594	-122.5150131	460273	United States of America	29	885	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
111	Vancouver	49.27341658	-123.1216442	1458415	Canada	24	987	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
112	Columbia	38.95207847	-92.33390955	244754	United States of America	25	852	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
113	Adapazari	40.79997601	30.4150321	260109	Turkey	19	619	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
114	Adelaide	-34.93498777	138.6000048	990677	Australia	26	970	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
115	Aguascalientes	21.87945992	-102.2904135	763589.5	Mexico	9	897	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
116	Akron	41.07039878	-81.51999597	451155	United States of America	19	551	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
117	Al Ahmadi	29.0769448	48.08377274	68763	Kuwait	31	948	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
118	Al Jahra	29.33747154	47.6580623	194193	Kuwait	20	767	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
119	Albacete	39.00034426	-1.869999839	127597	Spain	20	909	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
120	Albany	42.67001691	-73.81994918	484286	United States of America	35	945	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
1	Alexandria	38.82043276	-77.09998153	127273	United States of America	4	745	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
122	Algeciras	36.12671215	-5.466530363	106687.5	Spain	8	806	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
123	Algiers	36.7630648	3.05055253	2665831.5	Algeria	10	788	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
124	Allentown	40.59998822	-75.50002751	300980.5	United States of America	36	702	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
125	Gisenyi	-1.684665915	29.26290605	83623	Rwanda	6	626	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36
126	Gitarama	-2.069603659	29.75998165	87613	Rwanda	32	530	103	Berlin	52.52181866	13.40154862	3250007	Germany	904561	36



Screenshot of the project



"I tried to take advantage of the summer time to improve myself even more – as an athlete and as a person."

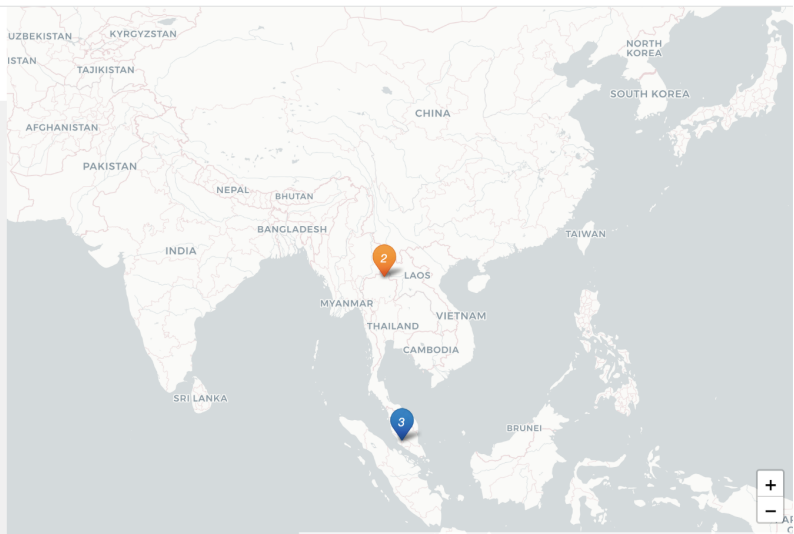
'I felt like I was being reborn'



Source: UNHCR

"Before receiving my nationality, I lived in a state of constant worry."

I became a Thai citizen as a teenager, when I was in Grade 8 or





## Marker

L.Marker is used to display clickable/draggable icons on the map. Extends [Layer](#).

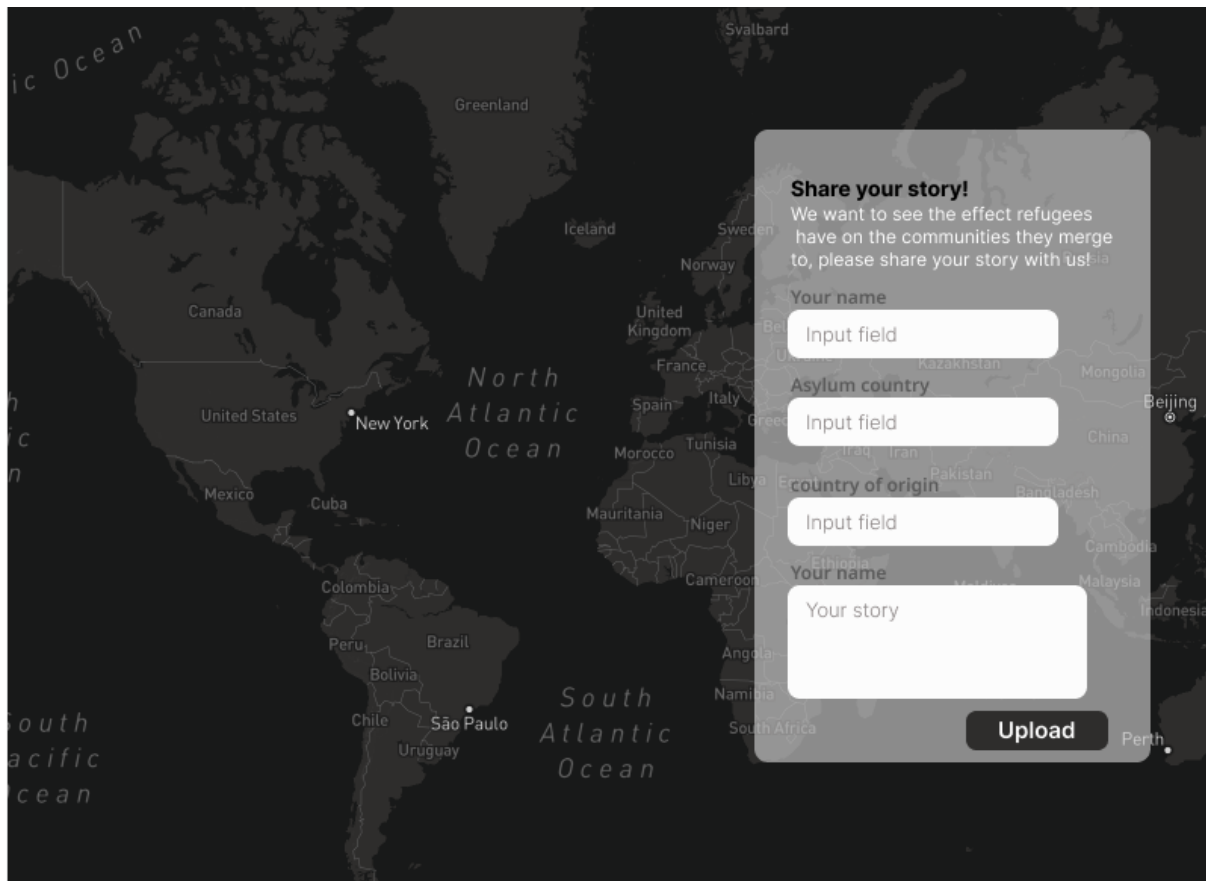
### Usage example

```
L.marker([50.5, 30.5]).addTo(map);
```

### Creation

Factory	Description
<code>L.marker(&lt;LatLng&gt; latlng, &lt;Marker options&gt; options?)</code>	Instantiates a Marker object given a geographical point and optionally an options object.

Marker example.



### Future form

The inspiration behind this project was simply the people. The refugees. I wanted them to have a voice on the internet. A community.

The future of the project is limitless. I want to have filtered data sets where the user could choose countries, asylum countries, and specific years, and learn more about a particular country dealing with refugees. Also, a place where refugees can have outside resources. Make the website multilingual for better coverage.

A \*live action \* screen recording of your project

<https://vimeo.com/779972065>