Topic: Design alternatives

The challenge(s):

Last week I was talking about to visualize data about the Coronavirus with the help of the data from the Johns Hopkins University. Besides that, I was running first tests with the React framework and the React Vis library for visualization. During the last week some challenges have occurred:

- This data source is not enough. If I want it to make the visualization as interactive and appealing as possible, more different sources are needed. Especially the latitude and longitude of the data is not really helpful to make choropleth maps. Furthermore, to present different graphs than all the other corona pages, other data is needed as well. So alternative sources are needed.

- How to make things interactive? React Vis is a nice library which provides many possibilities but making bar and radial charts interactive in an appealing way, is not their strongest asset. Alternative libraries are needed.

- Preprocessing and reading the data is an ongoing operation. Using different libraries makes it tricky to preprocesses the data once and then leave it like that. Also, what I didn't mention last week was the problem of how to read that preprocessed data in React?

The methods you tried to address the challenge(s):

- The original assumption, that I can make geographical maps with the latitude and longitude data might be still right but there are easier and simpler alternatives. GeoJSON or TOPOJSON is a file format, which provides simple geographical features for Java Script. This is perfect for React and there are many libraries which work with this file format. I chose "React Simple Maps" ([1]) to create a choropleth map together with a data source they suggested to use. The data source uses this GeoJSON file format and provides not just geographical information but also information about every country ([2]), such as the total population. This additional information might help to build graphs about e.g.: infected people per country in percentage. This would relativize the high numbers of certain countries and put a different point of view on the whole topic. Especially, this is a visualization I truly miss in the daily debates.

- React Vis provides possibilities to implement features such as swipe, zoom, drag, crosshair and so on but all these features are not really helpful when it comes to a bar or radial charts. I tried to make a single bar appear differently (e.g.: get highlighted), when a mouse hovers over it but there is no provided solution for this. Selfmade tries lead to cost-intensive computations because of iterating through the data over and over again and change details when the mouse moves a single pixel. I implemented a feature that a bar delivers more

information after a mouse click on it but without highlighting, it appears more like a hidden feature for now.

Luckily, the choropleth map and line charts work fine interactively, which both are going to be part of the webpage. Although, the choropleth map does not provide any possibilities to add a legend. A self-build interactive legend, which displays the level of density has been made.

- To read the data, which gets preprocessed by python, I use the D3 library ([3]). The preprocessing changed a few times already. By now, the scripts generate 9 different CSV files about global information; detailed information per country split up in confirmed, recovered, deaths and active cases, and time-series files to see daily changes per country.

A slight problem was the freshly added GeoJSON file ([2]). To pair it with my already preprocessed data, I just keep checking for equal country names in my Java Script code, whenever it is needed. Although, the names do not always match. Special cases for countries such as Eswatini (former Swaziland), Sudan (now 2 countries) or South Korea (written as Korea, South) forced me to edit the GeoJSON file by hand. I simply edited the names of around 10 countries to match them with the data from the Johns Hopkins University. This way is better than the other way around because the data from the Johns Hopkins University gets updated every day, while the GeoJSON file will not change anymore.

Lessons learned, if any:

- It is hard to plan a project and start with working on the notation and data when you don't really know where to go with it. Especially, when you are unclear about what framework you are going to use. I chose React and React Vis but I did not really know how it works, which lead to the problem, that I did not know what data React Vis expects. So, in the long run, I think such things are hard to plan ahead and somehow pointless.

- I never heard about the GeoJSON format before. This type of format is completely new to me.

Plan of next week:

- Keep working on the project to make it more appealing and likeable.

- Make bar and radial charts interactive

- Focus on details for proper data visualization

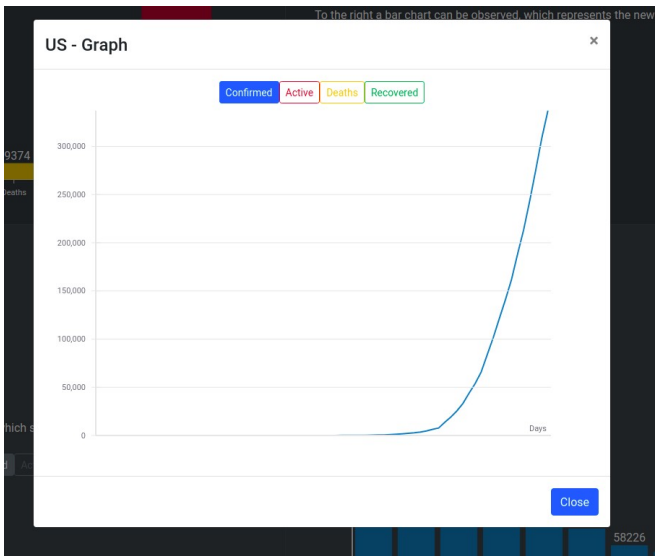
- Try to display data a bit different not like all other Corona web pages.

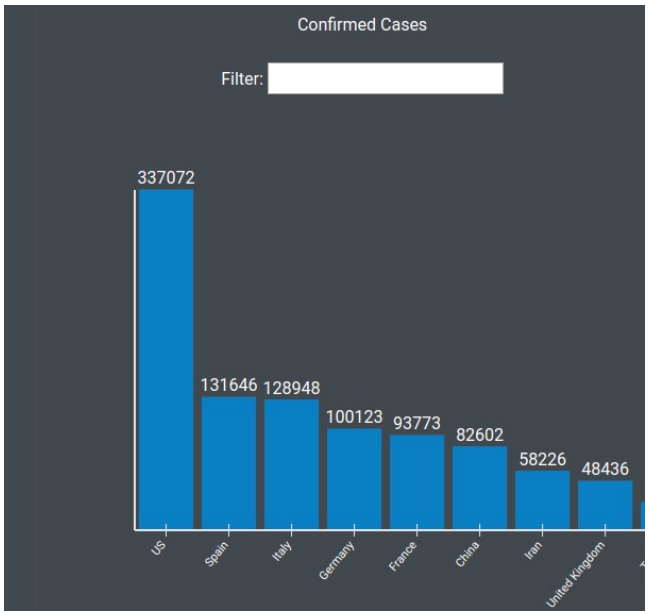
References if any:

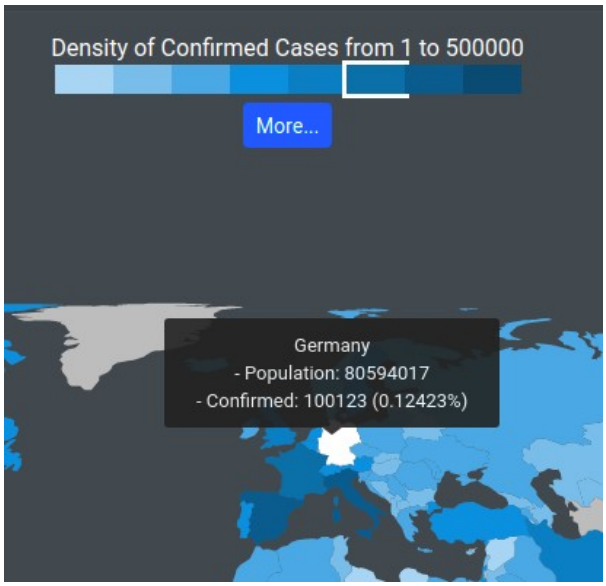
[1] https://www.react-simple-maps.io/
[2] https://github.com/zcreativelabs/react-simple-maps/tree/master/topojson-maps
[3] https://react-d3-library.github.io/

Progress:

Pictures sometimes say more than words. Here are three screenshots to display the progress.






Density of Confirmed Cases from 1 to 500000
More...
Germany
- Population: 80594017
- Confirmed: 100123 (0.12423%)