Covid-19 Dashboard Exercise [Bernardo Martelli]

1. Introduction

Since the outburst of Corona, back in January and February 2020, I had in mind the realization, together with a friend, of a dashboard showing some data referring to COVID-19. We made a simplified one, when COVID-API's were not yet available, using data from the World Health Organization. The information displayed was not updated automatically. Later many websites have appeared showing the number of deaths and the number of infections per country on a map, and API's are now available. So I want to make a simplified Dashboard to show the number of victims of COVID, divided per country. I do understand that the idea is not novel. Nor it is meant to be a tool to help the community. My only reason is exclusively to learn to work with React/Redux, to learn how to present data graphically and to learn how to fetch information from an API.

2. Design and Implementation

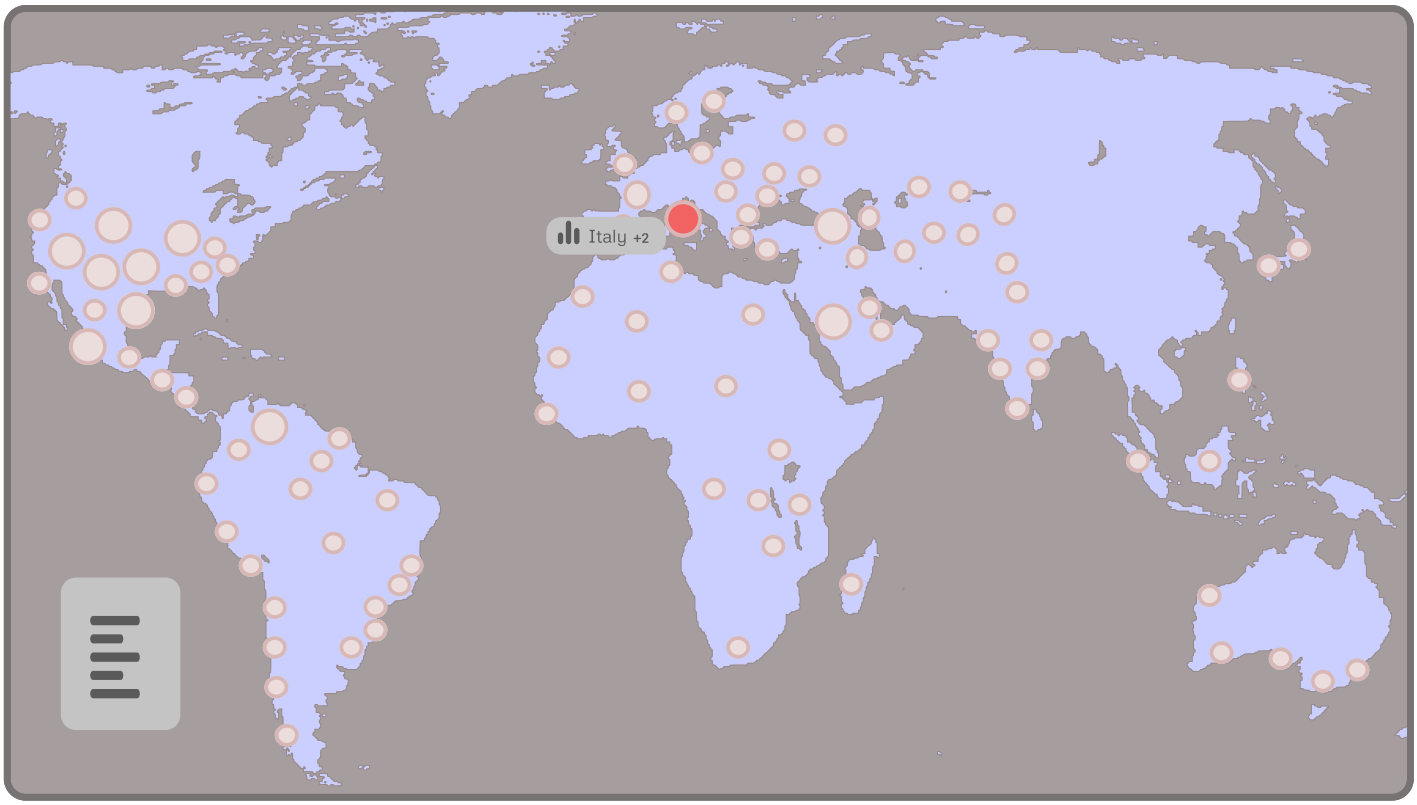
– Expected List of Features and inital design

A map of countries with overlapping symbols that shows in real-time, how many new victims, infections and recovered there are. The user should have a clear overview of the current updated situation.

When one hovers over the bubbles on the map, data is displayed.

When the user clicks one of the countries, a panel opens on the left, showing the total amount of casualties, infections and recoveries for that country. To close it, click on it.

* Figma and Giphy were used to make the initial UI design and the animation Link



– Choices

During the design phase, I had in mind to change the size of the bubbles according to the data. Like in Johns Hopkins original dashboard, the bigger the number of infected people, the bigger the size of the bubble on the map. But finally, I have discarded this option, because it is more important to me to be able to see the smaller countries, not being covered by the bubbles; this is also why the bubbles are semi-transparent. Like in John Hopkins dashboard, the graphics would clutter the map unnecessarily because Covid-19 is spread all over the world. If the map used had a zoom function, which showed the individual countries better, it would make more sense to change the size of the bubbles.

I thought about making the panel on the left draggable, but I finally discarded the idea. It is a feature to be later implemented, possibly.

I initially thought of using Redux as an exercise, but it was not essential. I decided to go deeper with Components, and passing state as props from parent Component to the child, instead.

A useful feature would be to have a local JSON file as a data source to use if no API or Internet connection is available.

This project could be worth being further explored and expanded, for example, by comparing the data on different days, and therefore showing a trend of the spread of the virus.

– Design

The interface is minimalistic. I chose a light-colours palette like in my design, avoiding strong combinations. I have used a grey background and some blue to highlight the maps when hovered. I may change the colour of the bubbles to bring them into prominence, and maybe animate them.

– Implementation

I made the backbone of the project using a Javascript library called Datamap.

Datamap uses a popular library called D3, that makes use of Scalable Vector Graphics to produce dynamic, interactive data visualizations. The data source originates from a free API, available at https://covid19api.com, that collects data from Johns Hopkins CSSE. Covid19api also provide the documentation for Postman.

The library collects all the data inside an array of objects; for each country, there is an object that represents it.

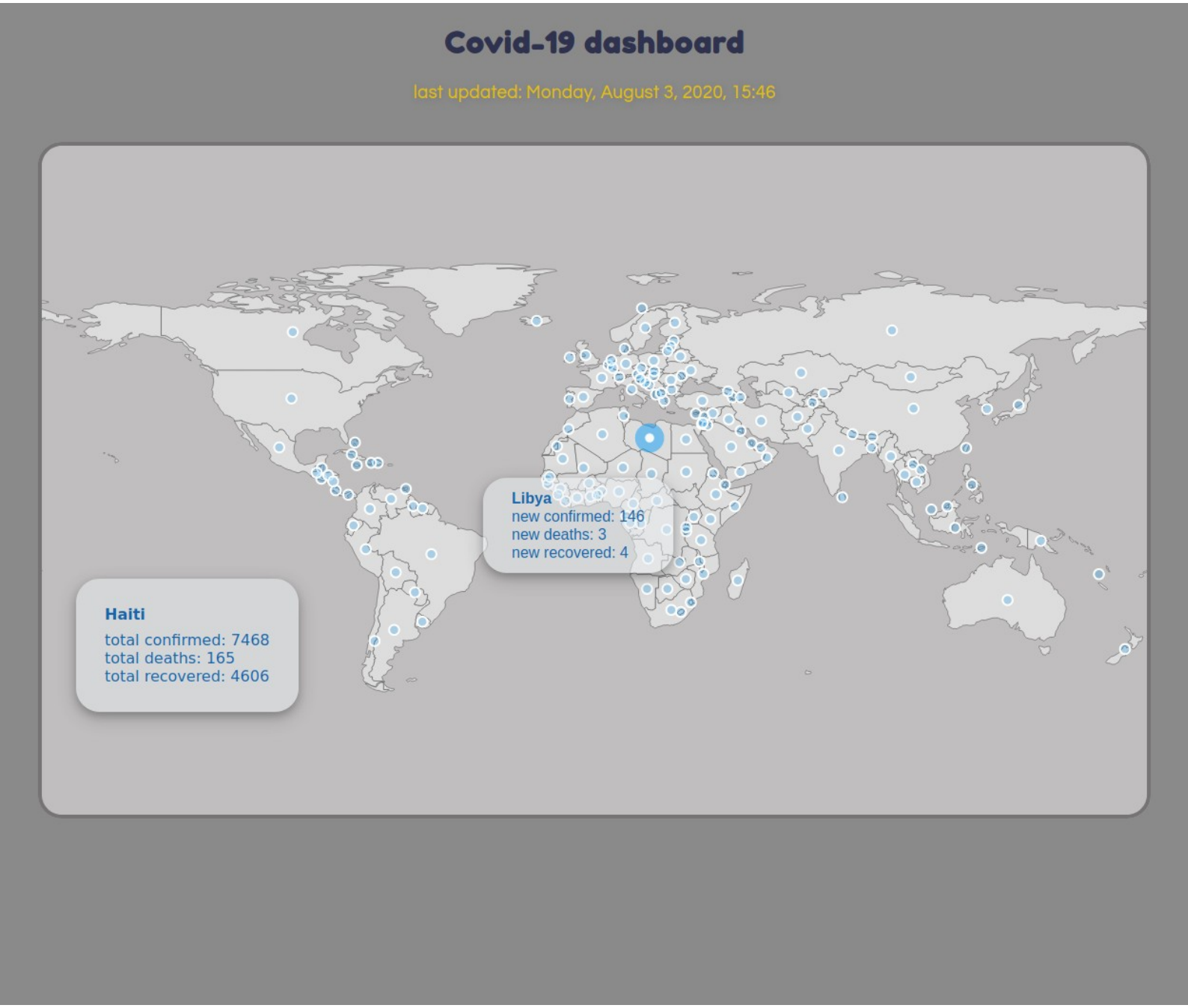
A conversion routine was necessary to be able to interchange data from the API to the library, mapping codes from ISO 3166-1 alpha-3 to ISO 3166. So, for example, the code 'AFG', used in the API to select the country Afghanistan, would match the entry 'AF' in Datamap.

I decided to use Datamap's existing data structure, trying to add other extra properties from the API to its array, like the number of deaths and alike.

To avoid errors coming from the delay caused when fetching the data from the API, I have written a conditional operator within the render function of the Data Component. Only when the loading of the JSON object from the API is completed, all the properties are then injected into an array of objects which is then passed to Datamap to build the bubbles.

Link to the project deployed

Link to the repository



3. Conclusion

I already discussed choices that I might have made differently and the one I made. I also wrote about possible improvements and further developments. I enjoyed learning to work with D3, React and APIs enough to be able to build this dashboard, but I am aware that I need much more experience and knowledge.

4. Market Survey

Compared to these projects, mine is more simple and compact.

- https://coronavirus.jhu.edu/map.html
- https://covid19map.org/
- https://covid19.health/
- https://github.com/stevenliuyi/covid19
- https://the2019ncov.com/
- https://covidtracking.app/
- https://covid19map.io/

5. References

- COVID-19 Dashboard by Johns Hopkins University [https://coronavirus.jhu.edu/map.html]
- Tool to track the spread of Coronavirus [https://covidly.com/]
- Popular Coronavirus tracking site [https://ncov2019.live/]
- COVID-19: WHO page [https://www.who.int/health-topics/coronavirus#tab=tab_1]
- Redux Crash Course with React [https://www.youtube.com/watch?v=93p3LxR9xfM]
- Free API data for Coronavirus [https://covid19api.com/]