Project 2 Requirements Document

By: Katie S. Johnston

# Introduction

For Project 2, I had originally planned to do a Spotify clone, but because the API was having some problems, I instead opted to build out a Covid tracker. I decided to choose this project because it would revolve around using charts, something I’ll need to do for my work at NM. Since there is a Covid-19 API and a couple of React libraries related to charts, I figured this would be a great challenge to try out.

# Repository

My Github Repo for this project can be found here: <https://github.com/itdev161-fa2022/katie-johnston-p2>

# Stacks

This project is mainly focused on APIs and React. The React dependencies used include Axios, Countup and React Chart.js.

There are 2 APIs used in this project. The first is for US data and can be found here: <https://api.covidtracking.com/v1/us/daily.json> . This data was part of a project collecting information on Covid cases up until March 7th, 2021. The second API used is for worldwide counts and can be found here: <https://covid19.mathdro.id/api> . This data is updated daily and takes the overall number of cases and deaths by country.

# Part 1: The Line Chart

The first element of my React App involves a dynamic line graph. This graph will show Covid cases against deaths in the United States, and when clicking on the infections or deaths titles it will default to showing either infections or deaths on the chart. While the API limitations only go up to March 2021, the API shows for deaths and cases by day in the United States.

Images:

Chart, line chart

Description automatically generated

Chart

Description automatically generated

Chart

Description automatically generated

# Part 2: Cards of Data

The next element of this app are the cards that show the total numbers of cases and confirmed deaths from Covid-19. At launch, the tracker displays the Global counts for Covid-19 cases and deaths, and when switching to other countries it goes to that country’s numbers of Covid cases and deaths. This data is current as of the day listed on the cards. Because the US API does not list current case numbers, when the US is selected the cards will instead show global case numbers.

Images:

Graphical user interface

Description automatically generated with low confidence

A picture containing text

Description automatically generated

# Part 3: The Bar Charts

For each non-US country, the Bar Charts will display current case numbers and deaths for each country. The country is selected by dropdown item, which is populated using data from the Global case API.

Images:

Graphical user interface, application

Description automatically generated

# Part 4: Optimized for Mobile

Since I will be taking a mobile development course next semester, I decided to try some optimizations for my app to work in a mobile browser. This required some slight changes to the CSS for each respective App page. This step was surprisingly pretty straightforward as all it required was to add *item xs={12} md={3}* to the project which allows for a graceful mobile display. Using Developer Tools inside of the Google Chrome Browser I was able to test out the display on mobile phones to confirm it’s functionality.

Images:

Graphical user interface, application

Description automatically generated

Application

Description automatically generated with medium confidence