ihatequarantine -- Kevin Li (PM), Derek Leung, Justin Shaw, Albert Wan
SoftDev2 pd9
P04 -- Data Visualization
2020-04-23

Disease Comparison: Day by Day

Task division
- Kevin Li
  - Project manager
  - Update README, make sure devlog is up to date
  - Update design doc to reflect changes to project midway through
  - Some coding (if necessary)
- Derek Leung
- Justin Shaw
- Albert Wan

Everything about data
- The dataset we are using is a CSV file that breaks down the total amount of deaths attributed to 13 separate pandemics (i.e. Cholera, COVID-19, Ebola), per day.
  - For example, day 1 of the COVID-19 outbreak had 0 deaths. It remained the case until day 11, the first death. Day 16, the second death.
  - The data for COVID-19 is up to date as of April 1st. It is still fairly early into this pandemic compared to the others, so the data will cut off abruptly.
  - Some of the diseases appear to repeat. This is because outbreaks of that disease have happened multiple times throughout history and in different regions.
    - We could make an array with the disease AND the year + region, then "index match" it with the columns in the CSV file. Index 0 would be "Cholera - Hispaniola (2010)" and Index 1 would be "COVID-19 - World (2019)" for example.
  - Original data:
    https://docs.google.com/spreadsheets/d/1zn_pqFBv9W9Hrfe-0LcfSYdywZHe4cOig4xQZ5mVaBQ/edit#gid=1624097889
  - Took only the necessary data and exported to a CSV file for easy manipulation/access.
- We will not be using an sqlite3 database for this project, as we want to focus on sharpening our d3.js skills. An API will also be more trouble to deal with than necessary.
- We will obtain the data using Python functions in the utl folder.

- It will be passed through jinja2 variables (variable | tojson inside a <script> tag)
- The variable will then be able to be accessed in any other JS files that are linked.
- Diagram:
-

Minimum viable product
- There will only be one main page (and potentially a separate help/FAQ page if required, should be able to put it all on the main page though).
- A line graph automatically renders on the page.
  - One differently colored line is drawn per epidemic/pandemic.
  - The X-axis is the amount of days since the first day of each outbreak. For the MVP, we want to cap it. 365 days seems like a good point because the numbers for most of the epidemics have stopped increasing in "drastic" amounts. And it's a satisfying number.
    - The dataset actually goes to 801 days.
    - For the MVP, we don't want to worry about resizing it since it's capped at 365 days.
    - Since we can't show 365 tick marks, we should split it into chunks (25 days, 50 days, 75 days, etc.)
  - The Y-axis is the total amount of deaths caused by an outbreak.
  - Hovering over a point on one of the lines should show the exact number on that day (since obviously we don't have space to show 500 tick marks)
- Below the graph is a legend. Similar to Naviance's Scattergrams feature, the user should be able to click on a disease to hide the line for that disease. It is completely wiped (not just hidden, otherwise you'd be able to hover over an invisible line for data
  - Do not let the user turn off the data for every single disease. Just in case something screwy happens because of it.

Extras if we have time
- Cool animations (? anything in mind ?)
- The user being able to pick how many days are to be shown on the X-axis
  - Limit between 100 days (arbitrary limit, I know, just gets rid of edge cases in case the user tries putting 1 in or something) and of course, 801 days.
  - Use an HTML form for this.
- The X- and Y-axes scale accordingly.
  - For example, turn off the data for Swine Flu/H1N1, by and away the largest dataset that dwarfs the others in total deaths.

- ○ If the scale doesn't change, there's a massive amount of empty space in the Y-axis. It should scale so that the next largest epidemic/pandemic reaches to the top of the graph.
  - ■ Can achieve this with an array listing the diseases from most deaths to least deaths, for easy calculations/access.

Site map
A site map is not necessary for a project like this. We are planning to have only a single page. A separate help/FAQ page may be helpful, but even that can be implemented as a bunch of tooltips that get shown after hovering over a (?) button next to the appropriate section on the page.

Component map

Bootstrap or Foundation?