**Overview**

The purpose of this project was to take California death data and publish it to the web. Death data by county in California by retired and working deaths over 4 years was the base data for this project. The data was saved as a GEOJSON file then incorporated into a leaflet html file that was then pushed to an Amazon Web Server to make it available online.

**Methodology**

1. **Data Retrieval**
   1. Geo-processed death data for California counties was loaded into ArcGIS as the data was stored as a feature layer.
   2. The data was projected from the North American Albers Equal Area Conic Projection into the World Geographic Coordinate system 1984 as that is the coordinate system that GEOJSON files contain (Internet Engineering Task Force 2016).
   3. After projecting the layer the layer was then exported as a GEOJSON file using the conversion tool in ArcMap’s Arc-Toolbox.
   4. The GEOJSON file was then converted to a JavaScript variable file by giving the GEOJSON a variable name and saving it with the .js extension (Figure 1).

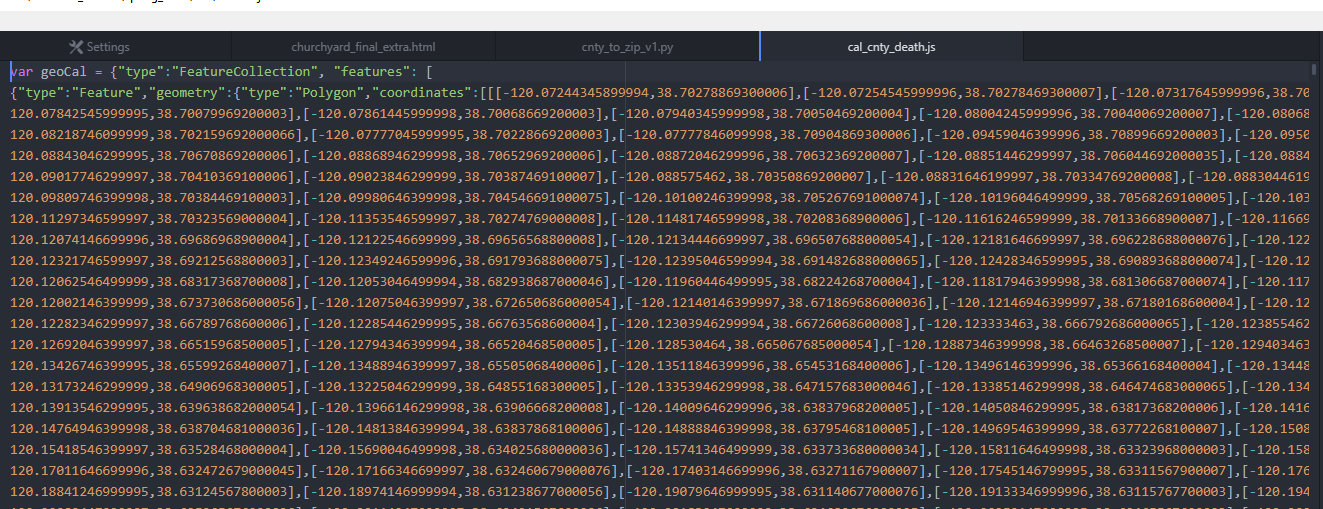


Figure 1: JavaScript GEOJSON

1. **Leaflet HTML File**
   1. Each individual segment of the Leaflet HTML code will be discussed in the following sections
   2. *Base HTML Code*
      * 1. The first section of code (Figure 2) defines the basic html template to hold the Leaflet Map as well as defining the map window’s base characteristics and style.
   3. *Defining the Working Death Attributes*
      1. For the working death portions of the JSON data (Figure 3) to retrieve the name of the counties as well as the working death data per 10K people the feature.properties function was used. The feature.properties.WRKp10K and feature.properties.NAME represented the death data and name of the county respectively.
      2. Additionally, the combination of the getColorWrk and styleWrk functions assign the different range values of the work deaths to a specific color so that a thematic map can be created.
      3. Finally the L.geoJSON function adds the work death feature to the web map

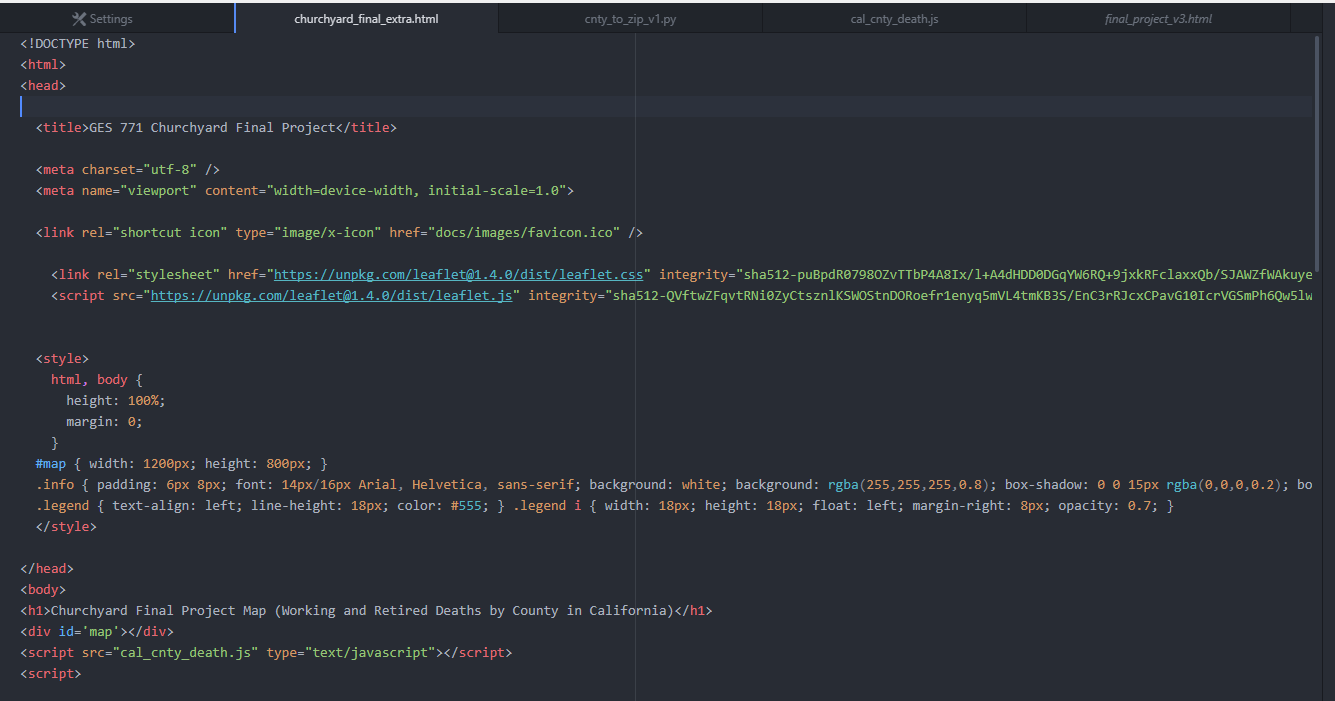


Figure 2: HTML Base Code

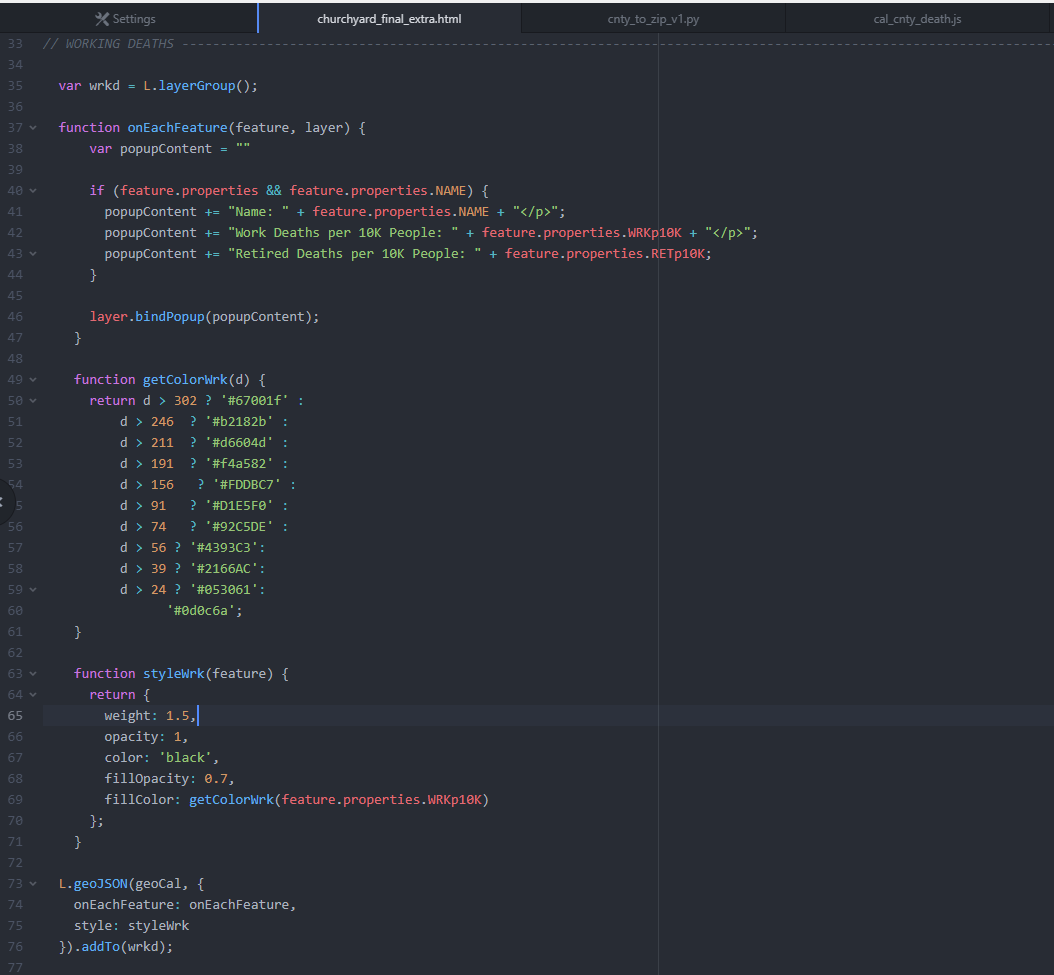


Figure 3: Working Death Code

* 1. *Defining the Retired Death Attributes*
     1. The retired death portion of the code is similar to the working death portion of the code but using the RETp10K field to assign colors to the web map (Figure 4).
  2. *Displaying the Map* 
     1. The grayscale and streets variables define the two base maps which our map data is displayed on top of.
     2. The map variable defines the initial center and zoom level of the map
     3. The overlays variable points to the two layer groups that contain the map data for the working and retired deaths that was created in parts C and D above.
     4. The two L.control.layers functions add the checkboxes which allow for swapping between which base layer is shown and which overlays are visible on the map
     5. Finally, the legend portion of the code takes the colors defined in the getColorWrk / getColorRet functions to create a legend that displays the color and the range of values the color represents (Figure 5).

1. **Displaying the Map on the Web**
   1. Once the html file was completed the file was uploaded to our Amazon Web Server so that the html file could be viewed from the internet (Figure 6 and Figure 7).

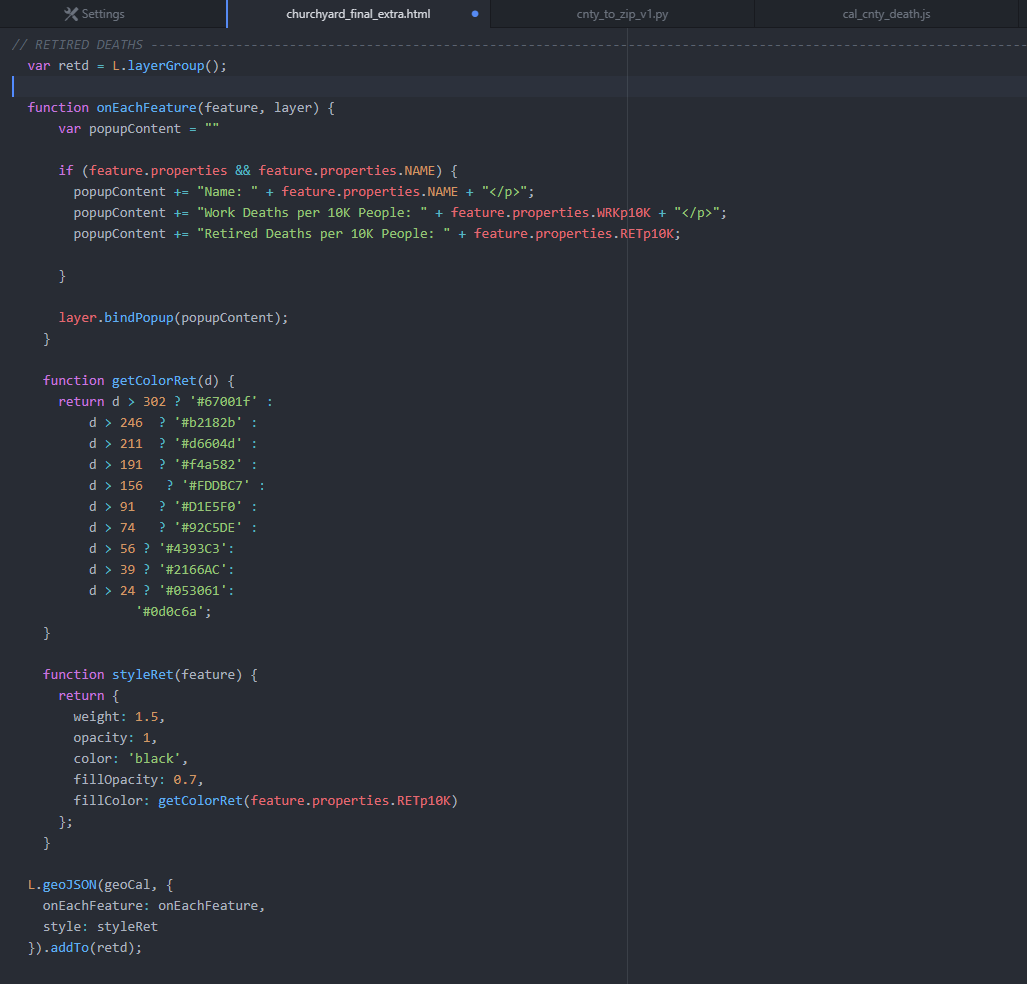


Figure : Retired Death Code

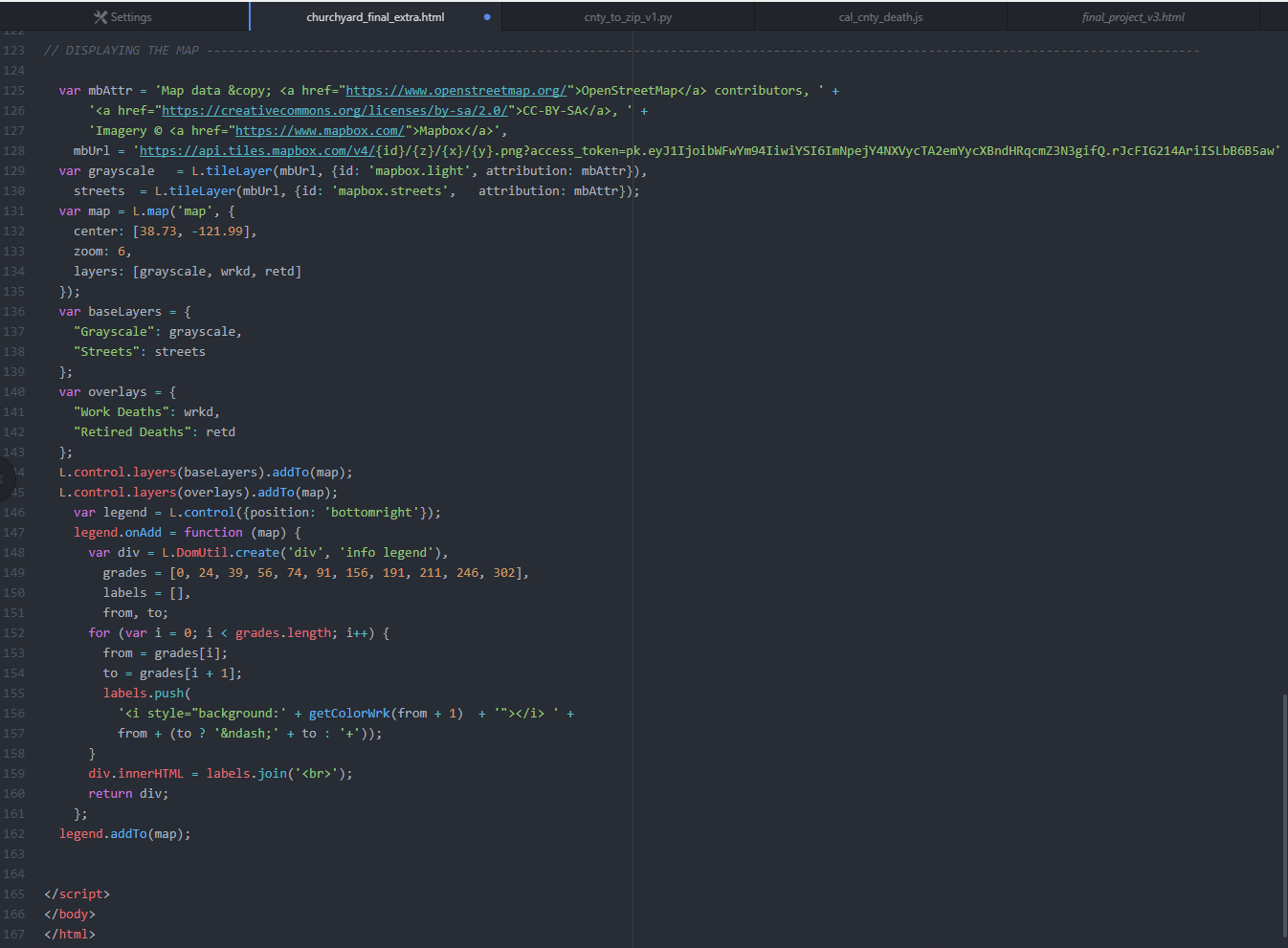


Figure : Map Display Code

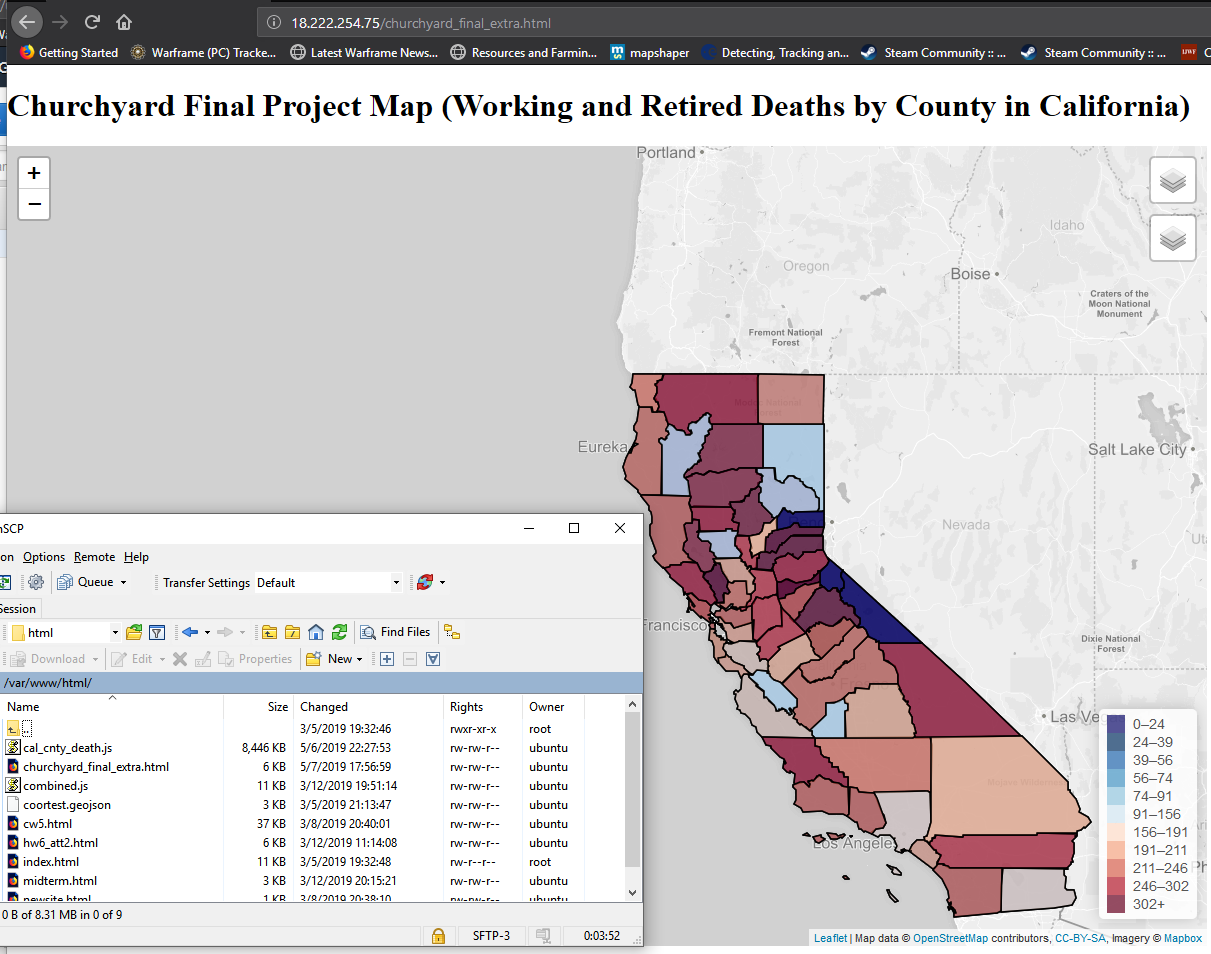


Figure : Final Output I

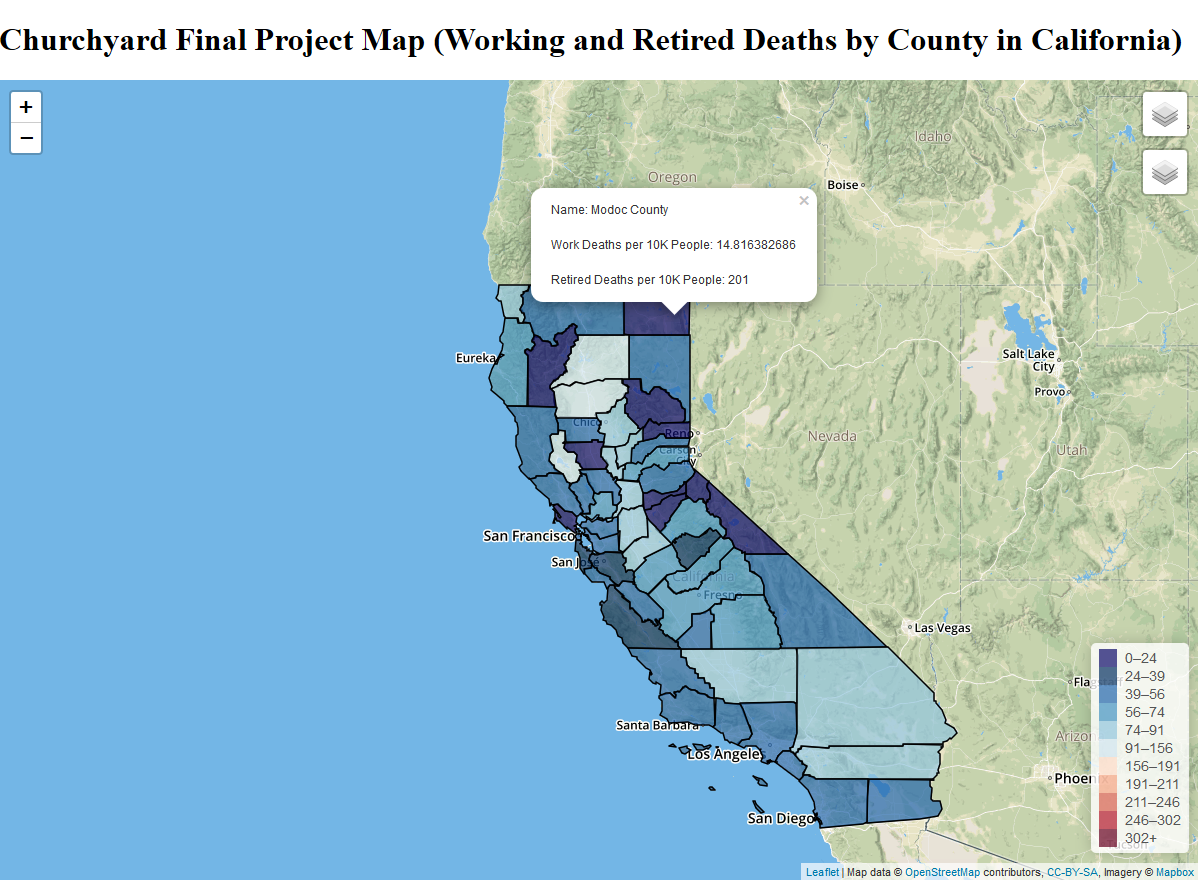


Figure : Final Output II

**References**

1. Internet Engineering Task Force (2016) *The GeoJSON Format* [Online] Available from: <https://tools.ietf.org/html/rfc7946>