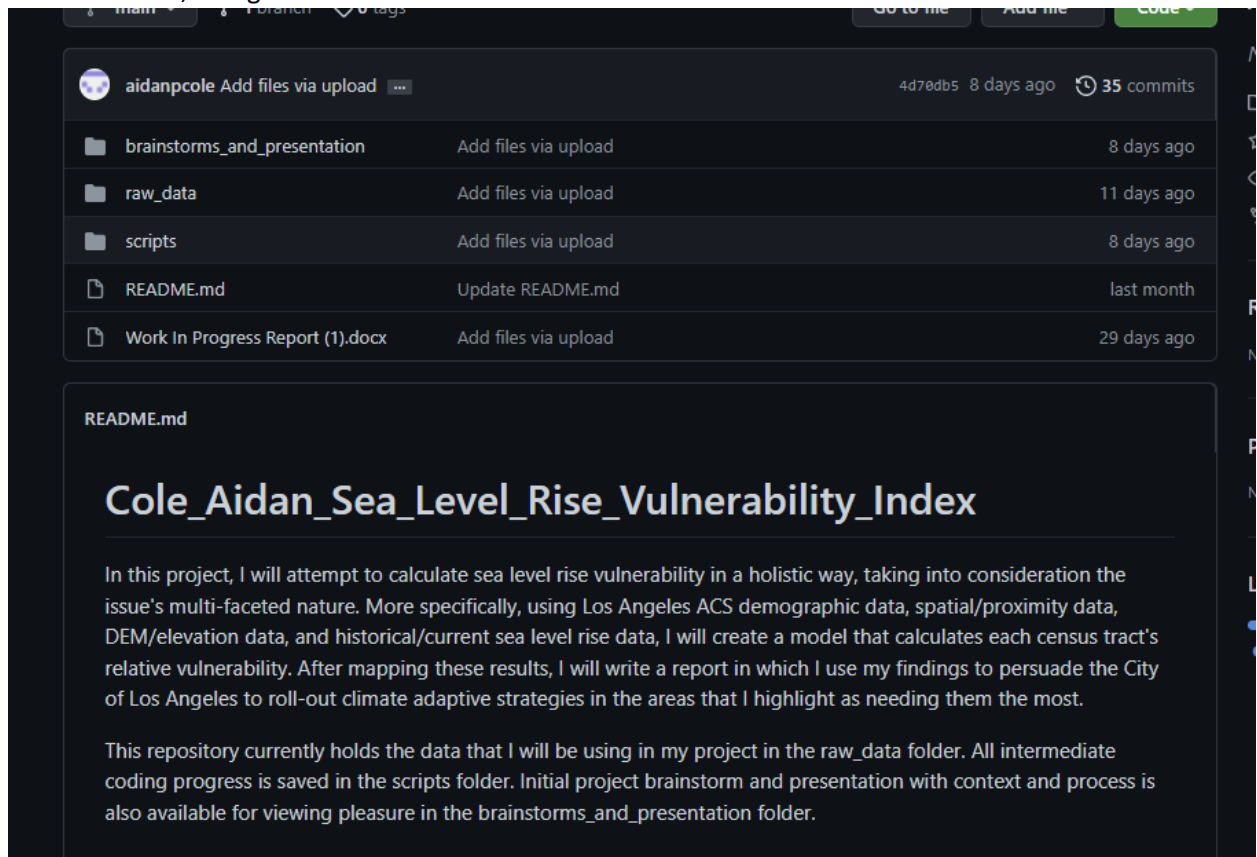**File Organization**

1. Is it easy to find what you are looking for?
   a. Yes, the files are neatly organized, with each file understandable as to what is happening. I can see where Aiden brainstormed, his raw data, and his code scripts are stored, along with his documents.

2. Are the files well-named?
   a. The files are well named. For example, in the scripts, I can see how Aiden has two code files, one is from ArcGIS, the other is his R code. Its nice to see all the code in one file.
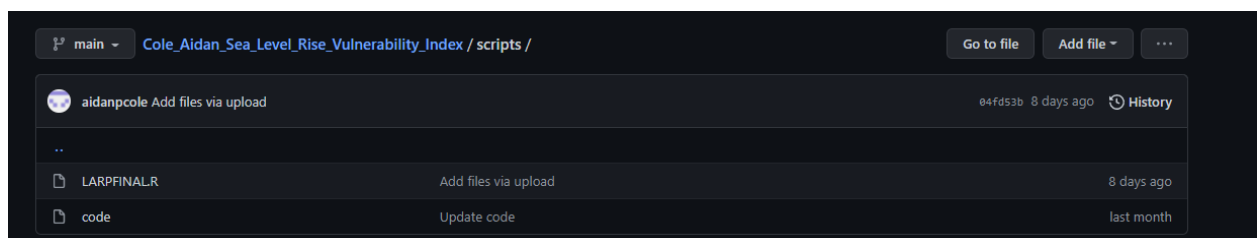
3. Is the README helpful?
   a. This is a clear and concise readMe file, that will be updated I believe when the deliverable is finalized. The only thing is to update it to the current project that looks at heat vulnerability instead of sea level rise.

Code

1. Is it easy to understand what the code is doing?
   Yes, 100$ it is easy to understand. Aiden has commented what he is performing and working on, as laid out in his code.

```
# Homeless Count by Census tract
HomelessCount <-
  st_read("/Users/Aidan/Desktop/code2/LARPFINAL/Homeless_Count_Los_Angeles_County_2019.geojson") %>%
  st_transform('EPSG:3498')

# remove tracts not in city of LA
filteredHomeless = st_filter(HomelessCount, LosAngelesBoundary, .pred = st_within)

# combine homeless count with tracts
Homeless_and_Tracts <- merge(filteredTracts, st_drop_geometry(filteredHomeless), by="Tract_N", all.filteredTracts=TRUE) %>%
  dplyr::select(-FID, -SPA, -CD, -Detailed_1, -Detailed_N, -SD, -Tract_1, -Year_1, -unincorpor,
                -SHAPE_Length, -SHAPE_Area, -totUnshe_1, -totShelt_1, -u_dens, -s_dens)


# Waste Site Locations
PollutionLocations <- read.csv("/Users/Aidan/Desktop/code2/LARPFINAL/CalEnviroScreen_4.0Excel_ADA_D1_2021.csv")

# Convert lat/long to a sf
Pollution_sf <- PollutionLocations %>%
  st_as_sf(coords = c("Longitude","Latitude"), crs=4326)%>%
  st_transform('EPSG:3498')


# filter to only LA
filteredPollution = st_filter(Pollution_sf, Homeless_and_Tracts, .pred = st_within)

Homeless_and_Tracts$GEOID <- gsub("^0+", "", Homeless_and_Tracts$GEOID)
# combine homeless count with tracts
Pollution_and_Tracts <- merge(Homeless_and_Tracts, st_drop_geometry(filteredPollution), by.x="GEOID", by.y="Census.Tract", all.Homeless_and_Tracts=TRUE) %>%
  dplyr::select(-Total.Population, -ZIP, -California.County, -Nearby.City...to.help.approximate.location.only.,
                -DRAFT.CES.4.0.Percentile.Range, -Ozone, -Ozone.Pctl, -PM2.5, -Diesel.PM, -Drinking.Water,
                -Lead, -Lead.Pctl, -Pesticides, -Pesticides.Pctl, -Tox..Release, -Traffic, -Traffic.Pctl, -Cleanup.Sites,
                -Cleanup.Sites.Pctl, -Groundwater.Threats, -Haz..Waste, -Imp..Water.Bodies, -Solid.Waste,
                -Low.Birth.Weight, -Low.Birth.Weight.Pctl, -Cardiovascular.Disease, -Cardiovascular.Disease.Pctl,
                -TotalUnemployed, -TotalWorkers, -Institutionalized, -Total18To65)


names(Pollution_and_Tracts)[names(Pollution_and_Tracts) == 'totPeopl_1'] <- "TotalHomeless"
names(Pollution_and_Tracts)[names(Pollution_and_Tracts) == 't_dens'] <- "HomelessDensity"

SocialTracts <- na.omit(Pollution_and_Tracts)
```

2. Is it modular? Do they repeat code?
   a. It's a bit repetitive in the census data wrangling, maybe a function could be made, but I think the difficulty here is Aiden finding the necessary census codes that he needs, and I think he needs to do this at least once in the beginning to know what he wants. Once he finds that, maybe he can streamline the census API calling.

3. Are things well-named? Overall "Code-Smells"
   a. Everything is well named for what is relevant to LA county. I noticed that there is code for boulder county CO, but that is example code that Aiden is using from last semester from our MUSA 508 class.

4. Are you confident this code is error-free? Is it hard to tell?
   a. Honestly, this code is well written and looks error free, not much more to say here.

5. If you inherited this project tomorrow, would you be able to succeed with it?
   a. In terms of how the code is written and organized, I believe that yes, I will be able to succeed with. It is well laid out, neat, and structured, committed to only one file of code.