CS 180 New Design

File Format: JSON

- 4 different JSON file
- Relational Database
- To have a better time complexity and space complexity

Backend Code: C++, RapidJSON

Frontend Code: React

Developer Function:

- Query (ex. All movies with the "Drama" genre)
- Search (ex. Typing in Actors/Movie name)
- Add/Remove (ex. Adding new entries, or new fields)
- Sort (ex. Sort movies by chronologic order)

Data Structure:

- Movies (default fields)
  - ID
  - Title
  - Year
  - Genre
  - Director
- People (default fields)
  - ID
  - Name
  - Movies
- Act_In (default fields)
  - ID (People): Primary key
  - ID (Movie)
- Act_By (default fields)
  - ID (Movie): Primary key
  - ID (People)

Things to Consider:

- How are we going to handle the director, are we going to move it to people and add in a type of director and actor
- If a new field is created, what should the default value be
- When do we read/update the file using what's stored in memory?

    - Ex. Does calling getMovie(string movieName) read the JSON if it’s already in memory?
- Possibly:
    - Convert the JSON to a memory location → only write to JSON if something is modified
    - Race conditions?

Reasons for the new change:

1. This will be a relational DB where Act_In is tied to People, and Act_by is tied to Movies, easier for specific searching, cause we can just search by Act_In or Act_by. Or if we want to see what movies or people are in the DB file.
2. 8. Easier to update, because if we want to add movies or people, we can just add them into their own JSON file rather than updating the whole thing. (This is avoided if one actor shows up in multiple movies, if we need to update it we will need to update the whole thing). We could just basically change the Act_in or Act_by JSON file. And same goes for the Movies and People JSON file.
3. 9. Faster process (time complexity), cause we don’t need to do lots of changing. I think the implantation will be easier also.
4. 10. The GUI will read in 4 different JSON files, where the web can do all sorts of things.