At this stage in the project, our code satisfies our original design and specifications. As per our design, or code is split into front and back end systems while passing references to data between them.

Included in the back-end system is our code to process data into a processed object and storing it. Our IOUtility handles this functionality, parsing the raw csv files and reading out the specific fields that we are then able to store in one of processed record objects (ProcessedPublicationRecord, ProcessedTeachingRecord etc). These objects are then able to be stored in our database object, which houses vectors of each type of record. While this uses up more memory than simply using data that has been queried, this cuts down the amount of processing needed for each command or query that the customer inputs. Overall, this makes it faster for the user and system, using up less resources.

This efficiency is further improved by passing references to record and filters between front end and back-end. By passing references only, we cut down on the amount of data being sent back and forth between the two systems and making it faster for the user. This also allows for better control over data, making sure we only have one instance of a record and are working with/changing only that record. This keeps all data consistent across queries and changes. To implement this, we used vectors of vectors of pointers when passing data. The vectors allow us to pass entire lists (vectors) of data between parts of the system and minimize the amount of data being passed.

For our front-end, our design revolved around a straight forward and easy to use interface. As per our original GUI design, the interface revolves around allowing the user to select a file to load into the system or selecting a file to view. Due to our backend design, we can have multiple processed files ready to view by user. This list of processed files allows the user to change quickly between data sets that they may want to view or see summaries of.

This mentality for easy-to-use interfaces was kept in the summary that comes up when selecting a processed file: users can resize and move around the graph or summary to better view what they are interested in. Summaries are also clearly displayed, showing fields of interest and allowing for the user to easily break the data down into more detail.