**Improvements that Team Orion would implement to the current Phase 1**



[**Improvement #1: GUI & data representation**](#_y94kjy96omp8)[**3**](#_y94kjy96omp8)

[**Improvement #2: Error Handling**](#_19x39pefv0c0)[**5**](#_19x39pefv0c0)

[Opening Files, The good:](#_y00fkgcnbnd7) [5](#_y00fkgcnbnd7)

[The Over-Load:](#_9so1so6z4ges) [5](#_9so1so6z4ges)

[The Concerning:](#_louh06s7z2o4) [6](#_louh06s7z2o4)

[The small:](#_udbcjvd208kf) [6](#_udbcjvd208kf)

[**Improvement #3: Sorting function**](#_qmykom7g5eg)[**7**](#_qmykom7g5eg)

[Summary:](#_8k8ncnabo48h) [7](#_8k8ncnabo48h)

[The Current Sorting Feature:](#_kdfn1ov01pje) [7](#_kdfn1ov01pje)

[What could have been improved:](#_cnp0rxmn03h3) [8](#_cnp0rxmn03h3)

# Improvement #1: GUI & data representation

We felt that for the sake of the employees at the Schulich School of Medicine and Dentistry, the usability and appearance of the Peach Galaxy software should be improved. We will redesign the graphical user interface so that it contains fewer usability redundancies. For example, a different tab for each of Teaching, Publications, Presentations and Grants and Clinical Funding (as shown below) is not particularly necessary considering that they each perform the exact same functions. The only difference is the file being operated on. Instead, we can have one unified tab that will change depending on the type of file loaded by the user.

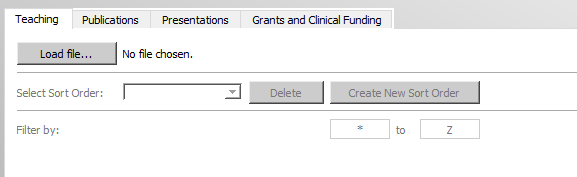


Fig 1.0

There is also a lot of wasted space that we found to be unnecessary. The Peach Galaxy logo should not take up so much space on the screen because once the user has opened the program, they aren’t really interested in what it is called anymore. They just want to get their work done. Currently it takes up about a quarter of the vertical screen space.

Besides the interactable interface, we found that the graphical representations of the data (like the pie charts) did not appear to be publication ready. Below is an example of a pie chart produced by the program. The numbers representing the subsection of the chart are awkwardly placed, and the colors are automatically assigned rather than being customizable. Administrators would not be likely to include these in their presentations, so we will revisit how the charts implemented in Phase I are drawn.

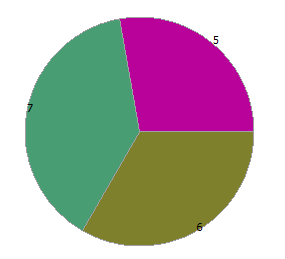


Fig 1.1

Also, the bar charts produced by the program are incorrect. Below is an example of the teaching done by Paul Malcomson. The horizontal axis is incorrectly labelled in millions rather than by the number of courses taught. It also doesn’t scale properly. Every bar extends beyond the visible area even though the actual values are only 5,5, and 7.

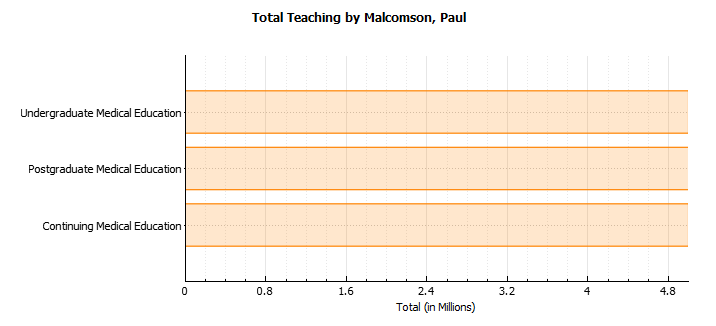


Fig 1.2

Finally, the tooltips provided by the GUI are not informative. The ‘on-hover’ information provided by every single UI element just says ‘Export to PDF’ rather than the relevant information. We will update all that information.

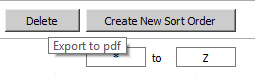


Fig 1.3

# Improvement #2: Error Handling

The software project of Phase 1 proved to be very strong when it came to the functionality of the program. Fetching the correct files, displaying the information and forming, what seemed like, pie/bar charts to accurate scale and measurements was done efficiently and effectively. However, no matter how strong and efficient it may seem to the user, it is the inner workings of capturing errors behind closed doors where it falls apart. Errors are to be handled with care in any system, solve the error itself or notify the user on what exactly is the problem so manual configuration can be easily executed.

### Opening Files, The good:

This program’s best error handling comes to us in the form of selecting files for display. In this program there are four sections and each section has a particular set of files to be loaded. Loading these files is imperative so that there is no mismatch of files, data going into the wrong file or even that of any file can be loaded in. The program recognizes which section it is in and from that determines what files need to be loaded. If the user decides to load a spreadsheet not of that section, the program will recognize it as an error then displaying to the user, in a new dialog box, that a mislabelled file has been chosen. The user is then prompted to choose another file if he/she would like. This handling the program does saves time and computation in the event a wrong file is chosen which may likely cause a bug or crashing of the program.

### The Over-Load:

As mentioned above, this program handles a user choosing a file for the correct section. Although this may be the case for one section, what happens if you load files to all sections? This is where the program has a hard time. A user may input one file to a section for displaying and editing, but once the user starts inputting files for all or some of the sections, the program may have a hard time holding these files. The result of this can cause the program to unexpectedly “stop working” and let windows find a solution, freeze or even crash the program sometimes. Any of these three would most likely make the user close the program and start it back up again. This can cause several inconveniences to the user such as the delay of time.

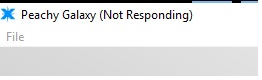
Fig 2.0

Figure 2.0 shows a sample of what could happen. This problem is usually invisible when it comes to computers with stronger processing capabilities. Error handling in this case would inform the user that the program crashed due to too many files being open and the computer the user is on may not be able to handle it.

### The Concerning:

This program does something rather annoying for the user and that is, when a file is picked, the program will detect if mandatory fields are missing to be filled by the user. Although helpful at first, the program forces the user to fill in all mandatory fields. This process can be quite tedious to accomplish, especially if there are many fields missing or, the data is not with the user at the moment. Rather, this feature would be more helpful to the user if they were allowed to type in some fields with data that they may already have on their person and then save the file with the other fields left out. This way, some data can be added on the spot rather than add more data later on.

### The small:

The program does not pick this up but when the user inputs a name for a particular sorting function they wish to implement, they are allowed to use any name they wish and that includes names that have already been taken for sorting functions. This should not be the case as a sorting order will be unique to the other ones and thus requires a particular and unique name.

# 

# Improvement #3: Sorting function

### Summary:

The sort function in its current state is lacking polish. The GUI aspect of the program was already discussed earlier but to re-iterate, display of the sorting function dialog box could see a more modern day style accompanied with a larger font to help the user read more clearly.

### The Current Sorting Feature:

The current sorting function has a solid functionality aspect to it but at the time is a little awkward to use. The user must first input a name for the particular sort they want, then is provided with a certain amount of boxes to choose which particular item is sorted and in what way. Depending on the section that the user is in, will determine what fields can be sorted. Currently the user does not have to input a field for every box but there are fields that have certain constraints. As an example, see the error message below in figure 3.0. If the program does not allow you to sort by date first, then the drop down menu should not offer the user the option to select that value in the first place.

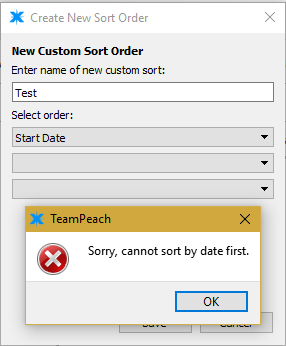


Fig 3.0

### What could have been improved:

Besides the previously mentioned fixes, another feature that would be useful for users would be to add the option to sort a column of data in alphabetical or reverse alphabetical by clicking on the header. For example, clicking on Member name once will make sure it is ordered properly, then clicking it again will reverse the order. Clicking on another column after will add this sort to the rows if possible.

Additionally there may be other fields, in the spreadsheets, that would be more representative of certain data. We would hope to accomplish the inclusion of these fields so that they can be incorporated into the sorting function as well as the current ones. This gives more flexibility and options to the user when certain other data needs to be visualized.

As stated earlier, the names of sorting functions can be anything the user choses including that of names chosen from before. This is something that should be fixed so that every sorting item can be unique and differentiable from one another.