Completion Report

Save button implemented for user to determine when to save.

We have completed all 6 objectives that we originally planned from sprint 1 as well as having additional features implemented in the design. We added the ability to have the manager remove employees from the system without modifying the database directly (i.e. changing the .txt file), but instead using the user interface methods available. We also added a save button that wasn’t planned from sprint 1 to let the manager decide when the employee list should be saved. The schedule will automatically be saved because in the event that taxes need to be filed, employees that worked there must be accounted for, even if they do not work there anymore. However, if there were changes done to the employees that weren’t meant to be done, the manager can close the application without saving.

Plans for next iteration

Plans for an undo and redo feature can be implemented by creating two stacks: the ‘undo’ stack contains every single user interaction with the system leading up to opening that program. The second stack, redo, contains anything popped from the ‘undo’ stack. If you are in the middle of calling ‘undo’ but then make a change that wasn’t part of the ‘redo’ stack, then the ‘redo’ stack becomes unavailable until you ‘undo’ once more. ‘Redo’ is only available immediately after calling ‘undo’.

Team Reflection

Landon and I found it difficult to make time to meet up to work on the project since we had different schedules. Given this, we had to split up work and then use the little time we had together to either put our work together and discuss what we need to change/what we have to do next. The second sprint was more interesting in the parts where we had creative liberties while at the same time, have requirements to meet from sprint 1 objectives. The annotations from the sprint 1 submission for corrections was unclear, so we made assumptions as to what Dr. Hosack would like, trying to follow to the most correct submission.