This document is individually submitted by Paarth Kashyap. It contains both Technical and Collaborative problems I faced during the completion of the ICS Final Project.

Technical Problems

1. **Problem**- Initially used a modified version of MoreMovieData.csv thinking it would be easier to process→ It was too complicated to effectively read into arrays
   Solution- Used the original MoreMovieData.csv and used a series of for loops to store information from lines into Linked Lists That is later added to an object linked list

2. **Problem**- The login file would not update after creating a new user and was throwing Null errors when searching for users
   Solution- Each new user created would be directly added to the loginFile.txt file and would call Sorting.java file (that is responsible for updating the obj-list). When a user is not found, an if statement would handle it by breaking the loop and returning false.

3. **Problem-** To build the interface as the original plan, the movies needed to be displayed in a panel. Placing each movie with its information onto 1 panel was not efficient and difficult to place with proper spacing.
   Solution- Each movie is assigned to 1 panel that holds the ratings, image and name. To display the movies, an array of panels is created that hold the panels of each movie. Each panel is created using a for loop and added to the main display. I also decided it would look neater to display all movies initially as it was too time-consuming to implement a scroll bar for the main display while maintaining only 6 movies on the screen (think online shopping scroll system).

4. **Problem**- The original rating system involved a JComboBox on the movie panel which would be created in the loop. The combo box was re-written into memory causing all the previous panels to not have a functioning combo box. Only the last panel would work
   Solution- Initially, attempted to create an array of JComboBoxes to assign while creating the individual panels, after implementing, the action listener would call code from different sections of the class and was causing the program to load N number of times after an action.
   Solution 2- Created a separate panel consisting of 2 combo boxes, movie names and ratings to give. This way I was able to pick what movie and what rating to assign based on the logged-in user and user input. The current Rating system also updates the rating in Real-Time to the CSV files that they are stored in (MoreMovieData.csv)

5. **Problem**- When receiving the rating input from the rating panel, the ratings would update in the wrong index (wrong column) and even in the wrong lines completely (for the wrong movie), causing the entire program to give IO, Null Pointer, and StringIndexOutOfBoundsException errors.
   Solution- Saved the position of the User from the obj-list and manually count the line it was on in the CSV file (position+1). Then, parse the line and keep the name of the movie + any reviews before the actual location of the changed review (substring from 0 to (2*movie_position +1) due to commas), imputed the rating and got the substring from everything after that index (substring from movie name +(2*movie_position +2) due to commas). Write this updated line to only the movie chosen line.

Collaborative problems
- TJ was in easy contact with the team but did not attempt to work on any of his assigned tasks, I tried to assign the easier tasks such as creating the help button, which he refused. No contribution from his side

- Meekail had been active at the beginning of the stage of project (first week of assignment), As I completed my initial tasks of Object classes and sort class, I noticed that Meekail would only "Attempt" his tasks of building the rating system, and recommendation system. He was not able to complete either of them and ultimately I had to complete them myself
- He created the login check file which worked well until the first milestone of the project. That class was later modified by me to be usable by the program in the final stages of development.
- Essentially, the entire project's code has been developed by me as I have spent the most hours building it.
- The documentation was a collaborative component that was almost evenly split between Meekail and Paarth