PHASE 3: DOCUMENTATION

**Programming Problems**

Problem # 1: GUI Images Layout Manager

Inserting images in JPanel would stack them up against one another when the GridBagLayout manger was used. This did not produce the desired results as the pictures of the movies were supposed to have space in between them and spread evenly through the space. This was achieved by inserting blank pictures called Capture.jpg and Capture2.jpg that were blue in colour (similar to that of background). The size was the pictures were set (54 X 151 and 38 X 151 pixels respectively) so that all eight pictures could fit inside the screen along with the text field.

Problem # 2:

The actionPerformed method in the Handler class proved to very problematic as it could not be passed a value. Moreover it could also not return a value. The actionPerformed method was called to respond to user’s interaction with buttons and textfields. As the variables could not be passed down a value, most of the actions were limited in what they could do. This problem was primarily dealt by using it to call methods from either the same or different classes. By doing so, I did not have to declare or initialize a variable and the actionPerformed would find it and easily pass it to other methods where these variables could be modified for intended purpose.

Problem # 3: Merging classes

During Milestone C, I had to merge the classes with the GUI so that it all worked together and corresponded to each other as one functional unit. Initially, we had three different classes called Choose, MainVideo and Login. Login was responsible for getting input from user and searching the database to get the information for any returning customer. This would be then redirected to either MainVideo or Choose depending on who user interaction. MainVideo would be responsible for renting and Choose would be responsible for both returning and renting.

However, these caused problems as MainVideo and Choose were both trying to rent movies and this caused problems as they would both rent different Strings to be inputted in the external file. This was resolved by grouping them in a single class along with Login into our main class called MovieRentalMain. This resolved problems in two ways: only a single String (newRec) was used to write the data on to our external file and that the String from text field was directly passed to the methods that would call the picture methods (pictures, pictures3 and pictures4).

Problem # 4: Version incompatibility

All the team members had jGRASP version 2.02\_01 whereas the school ones were jGRASP 2.00\_07. This caused compatibility issues as initializing and declaring rules in in actionPerformerd method were stricter in the previous version comparatively. For one of the actionPerformed methods declaring a String inside the Handler class caused an error that asked the variable for the String to be declared final.

The problem was resolved by declaring it instead outside of the Handler class and in the beginning of the MovieRentalMain class. There was no need to declare the variables inside the class and the class recognized the variable as String and modified it as required by the program. Using the word final in front of the declaration statement did not correct the problem as the String was a variable and not a constant and had to change in that method.

**TESTING NOTES**

During the testing phases, a lot of errors were encountered and resolved. Most common error was NullPointer Exception when a comparison was made between two arrays and one of them was not initialized. Most of the time this had been due to problems with passing the data through the actionPerfomed method. This only became a problem during later testing phases when the classes were being merged. This then lead to combining four different classes into one which prevented the problem.

The testing phases also revealed errors when invalid input was entered and this was taken care of by making the methods making the comparison return a boolean. If the boolean was false, the Handler would return a dialog box letting the user know of the error. In this way, the program did not crash and allowed user to try again without restarting the program.