Final Project Report

For our final project, we aimed to create an interactable shopping list where a user could add, remove, and view items on their list. Starting off from execution, a window pops up incorporating elements of a GUI interface for the user to interact with our program. The user begins by entering in their budget into the budget field, and an item into the input field. From there, the user may decide to either add that item, or remove it by pressing the respective button. This is done through action listeners and action events that allow the code to recognize when the user interacts with the buttons, and which buttons were pressed. If the user chooses to add the item, the cost is increased and updated using data that is read from a file through a file input stream that includes names of items in a stores inventory and their respective prices; the item is then added to an array list. If they would like to see their item added, they need only to press the “View List” button and it will pop up on the JTextArea. Using an area instead of a field allows more room for output, especially multiline output. This is due to a file output stream and a print writer that edits our list.txt file when items are added/removed from the list. The user can continue to add as many items as they wish, however, if the cost exceeds the inputted budget, a message will pop up warning the user that they have gone over their desired spending amount. Once they acknowledge the message, they can view their list once again, and decide which items they would like to remove in order to stay within budget. In order to do this, type the item into input and select the remove button. The cost will be decreased and updated as well as the list once the user decides to view their list. Let’s say that the user attempts to add an item to the list that is not in the stores inventory. In this case, an error message will pop up once again to indicate that the item they have inputted is not in inventory. The cost will not be updated, and the item will not be added to the array list. The user must acknowledge the message and try a different item. This prohibits any unexpected inputs. Another significant feature of this project is its ability to save the list between runs. That way, the user can create a basic list of items that they buy every week and adjust it based on their needs for that specific shopping experience. If they want to see their exact same list the following week, all they need to do is press “View List” after running the code. If they decide they want to start from scratch, they can press the “Clear” button to clear the list –remember to press the view button to see the changes. Lastly, when you are satisfied with your list, you may exit the program by clicking on the X in the top right corner of the window. This will close it out and stop the program from running.