Criterion B: Design

Product Development Plan:

#1 Creating a Spreadsheet class

- Create a class to store the user's spreadsheet information
- Use arrays to contain the name of the spreadsheet, date, item/description, income, outcome, and balance
- Ensure that the array is properly printed in a table format

#2 Creating a Display Spreadsheet class

- Create a class that displays the given spreadsheet in a table format
- Takes in user-input and sets the contents of the spreadsheet
- Allows the user to modify any information that is incorrectly put in
- Returns the overall balance by finding the difference between income and outcome
- Generates monthly and yearly expenses

#3 Develop a method to save the spreadsheet

- On button click, the current spreadsheet can be saved as an Excel workbook
- Spreadsheets that have already been saved can be easily loaded in the menu

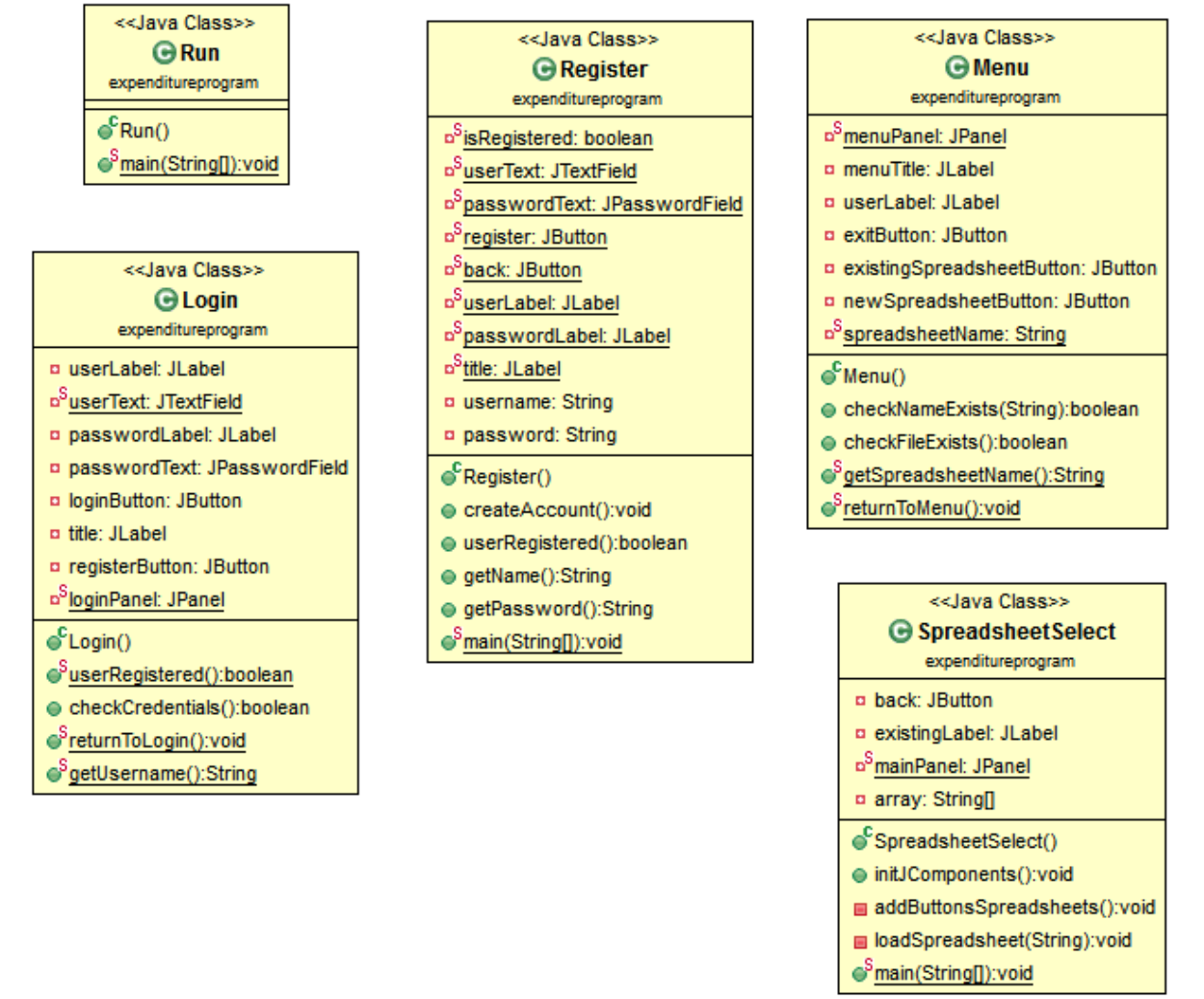
#4 Creating a Menu:

- Allows the user to create a new spreadsheet or choose from an existing one
- Allows the user to name their spreadsheets accordingly
- The user can log out of the program here

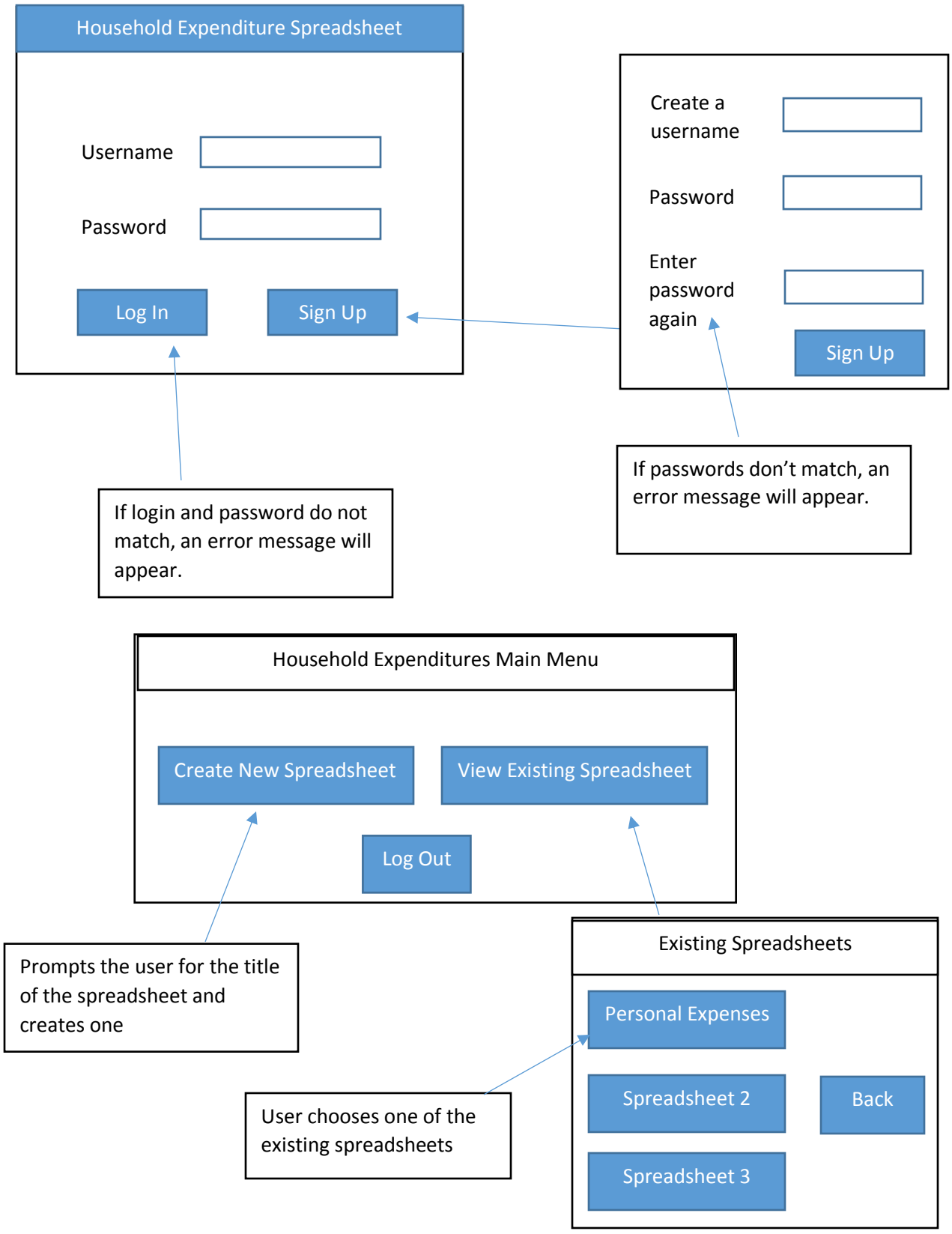
#5 Creating a Login/Register Class:

- Prevents others from stealing personal/financial information
- Allows user to input corresponding username and password to access the program
- Allows user to create a new account if it's their first time using the program

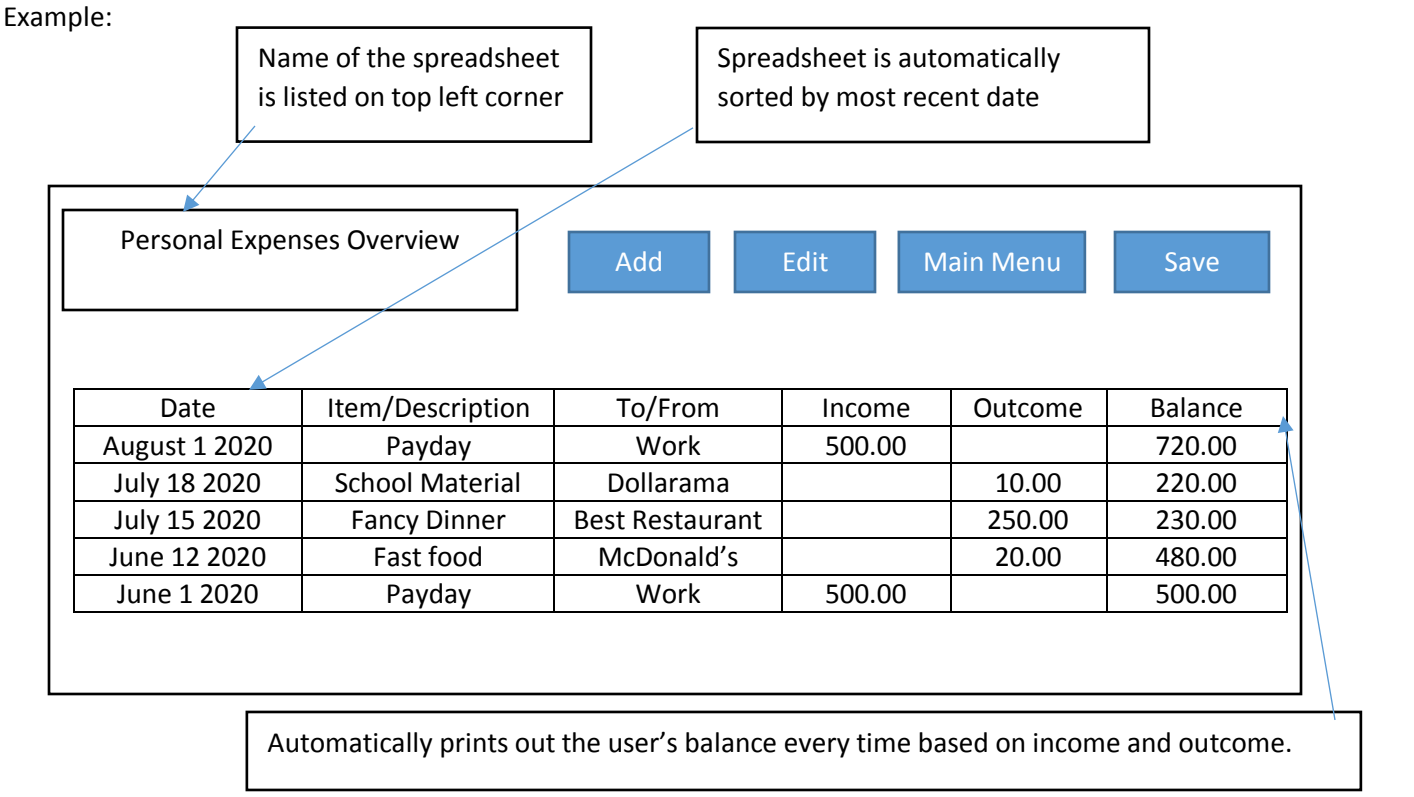
Current UML Diagram: (Full Updated UML Diagram found in Appendix B)

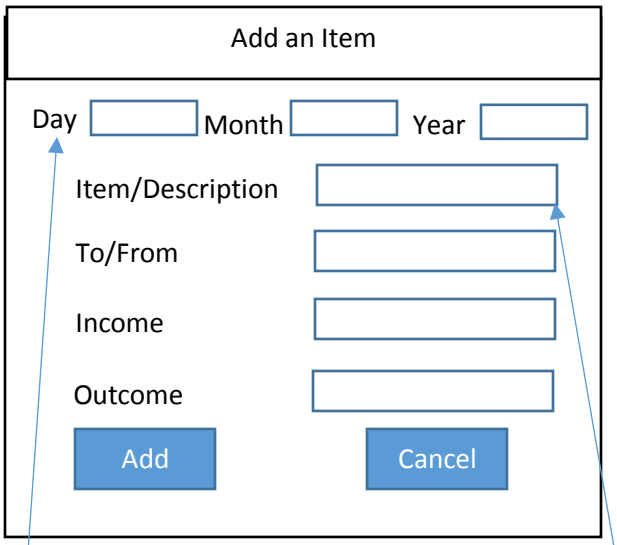


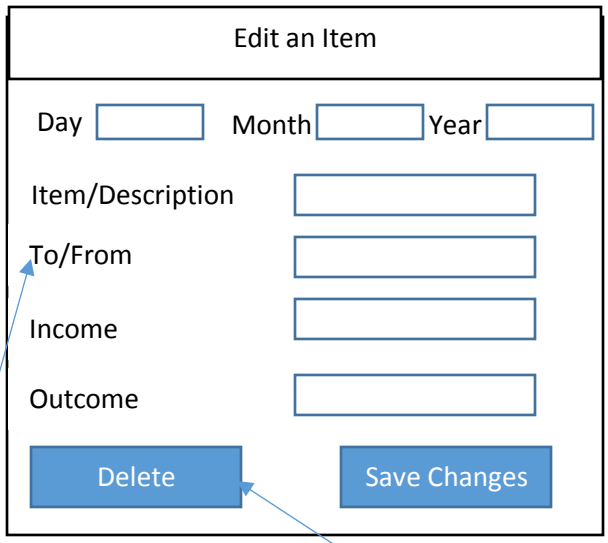
Software Design:



Example:





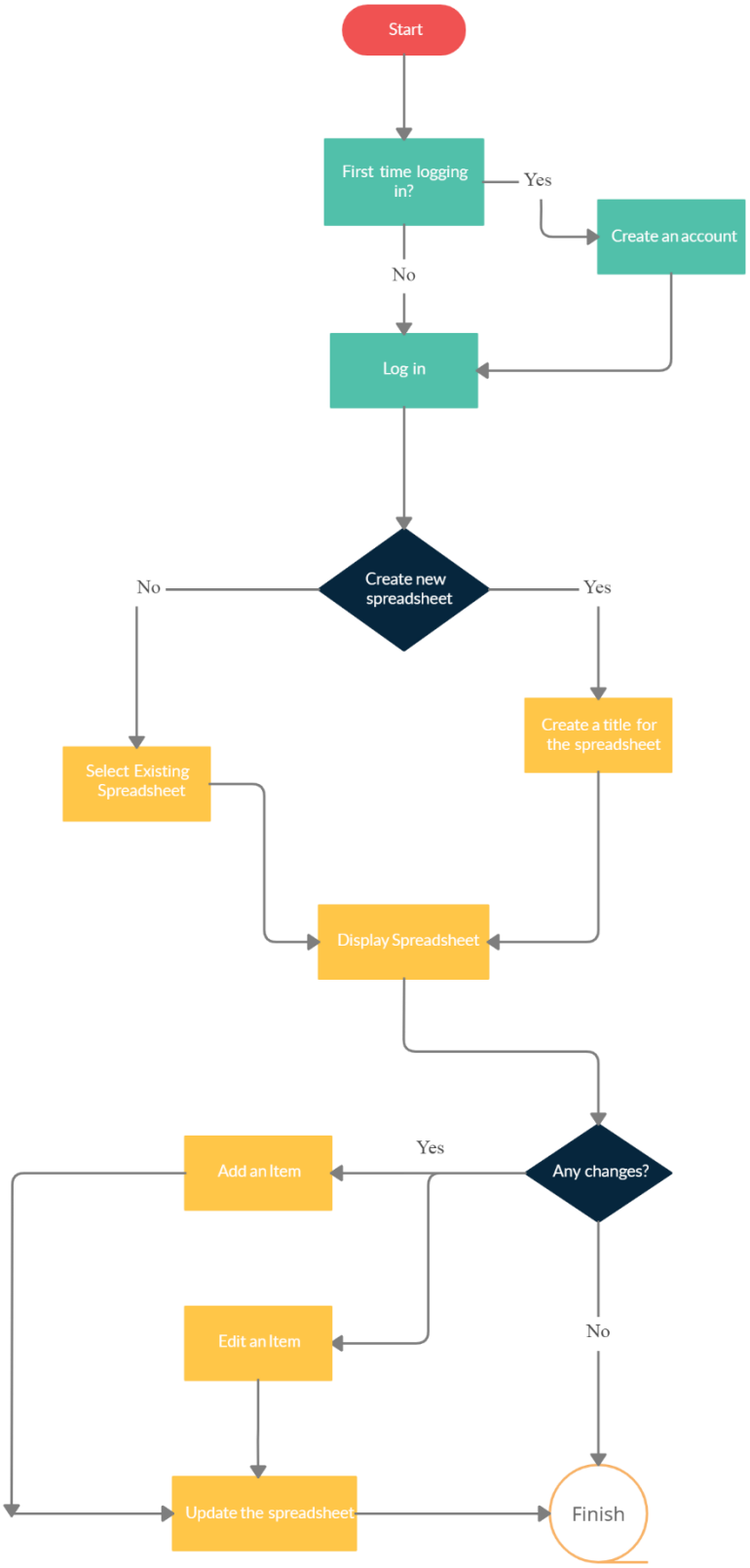


Year, Month, and Day are selected using a drop-down menu, limiting user error

If the user entered an Item/Description and names in To/From in the spreadsheet, these entries are saved and can be easily chosen again by the user

User can choose to delete an entire row

Program's Working Plan (process description):



Input Data:

Data 1 Client selects “Sign Up”, proceeds to create a username and password. Last field requires the client to re-enter the password. Afterwards, the client uses the credentials that they just entered to use the program. Example: Username: MyFamily Password: expenses	Comment: The contents in the password text fields are compared with each other to determine whether they are the same. If not, an error message will appear. Location: Once the username and password is verified, a text file will be created storing the username and password.
Data 2 Client chooses to create a new spreadsheet and is prompted to enter the name that he/she wishes to call it.	Location: A text file with the name of the spreadsheet is created. Comment: The program automatically creates a new item with the description stating the spreadsheet has been created.
Data 3 Client chooses to add a new item into the spreadsheet. Example: (assume balance is set to $10) Day: 12 Month: June Year: 2020 Item/Description: Pizza Lunch To/From: John’s Pizza Income: 0 Outcome: 10.00 Output: June 12 2020 – Pizza Lunch – John’s Pizza – 0 – 10.00 – 0.00 (as shown in the example above)	Location: Text file will be modified to include the user’s entry Comment: If the item/description was previously stored in the database, the user can choose to select it again from a drop-down menu. This function would work for the “To/From” text field as well.

Output Data

1. The specific spreadsheet is displayed, sorted by most recent date.

2. Error messages will be outputted if there are conflicting pieces of information (username and password).

Test Number	Description	Instructions	Expected Results
1	Logging into the program	To use the program, the user must log in with their credentials.	The user is able to log into the main menu if credentials are correct. Otherwise, they are stuck on the security page.
2	Selecting main menu options	The user can choose one of two options: 1. Create a new spreadsheet 2. View existing spreadsheet	If the user chooses option 1, the program prompts the user for the name of the spreadsheet and displays it. If the user selects option 2, a tab will open with a menu of existing spreadsheets. The user selects a spreadsheet and displays it.
3	Displaying the spreadsheet 1. Add an item 2. Edit an item.	1. By using this button, an "add an item" window will appear and allow the user to fill in the required fields 2. By selecting this button, an "edit an item" button will appear and allow the user to modify their existing items	The expected result after using the two buttons is that the spreadsheet will be updated with the given information.
4	Checking if monthly expense and yearly expenses are calculated	User updates the given spreadsheet	The monthly total and yearly total of the expense should change and reflect on the user's new input.
5	Income and outcome values are not doubles	User inputs an invalid datatype, for example enters a string instead of a double	The program will prompt the user to enter a positive double value for both input and output.
6	Check "log out" function	Run the program from the main menu or the spreadsheet window and click the "log out" button.	The program should be terminated and all windows should be closed.
7	Check "save" function	User clicks the "save" button in the spreadsheet	The contents of the spreadsheet will be saved as an Excel file.

8	Check if previously stored spreadsheets can be called	The user enters a previously stored item or sender/receiver.	The program should output a drop down menu of all the names of previous items and the user can choose from the list provided.
9	Check “delete” function	The user deletes one of the rows in the spreadsheet.	The program should delete an entire row of the spreadsheet.

Word count: 261 words