Criterion E - Evaluation

Evaluation of Product: Referring to Success Criterias

Criteria	V / X	Notes
- Barcode generator for dates	V	This works well and the barcode encodes date correctly and fast.
- Number of barcodes to make can be specified	V	However, if the user requires more than 21 barcodes, must fill in the information again because only 21 fits onto the A4 page with how it's coded.
- Can print barcode	X	The program saves the barcodes to a pdf file, so the client can then choose to print that pdf. Instead of connecting to the printer directly, this suffices.
- Can manually input items' data	V	The program differentiates between Lost items and Found items well, updating the displays as appropriate to collect more information.
- Error handling: program will not allow the user to add items with information of the wrong format. An error message in the form of a jLabel will be displayed.	V/X	Error handling is good for the most part. However, it still does not account for some cases. For example, the program does not detect when a date inputted is one in the future.
- Lost items will have information about whether it has been lost for more than four weeks	V	
- Found items will have information about the collector (e.g. collector name and student ID)	V	
- The user can add items and it will show in display table	V	The display table works well. The fourth column, status, displays multiple data attributes on each line using concatenated strings.
- Sort items by name, type, date and status	V	Sort works very well with all data types.
- Search for items by name, type and date	V	
- The data can be saved as an excel spreadsheet	X	Requires using an extra library to deal with excel spreadsheets. Ran out of time to code.

- Be able to store appropriate number of items	V	Stores 100 items maximum since the data structure used is an array. This is appropriate for my client’s use.

(Success Criteria not included in word count since table/bulleted list)

Recommendations from Client:

Recommendations for improvement	**Benefit(s)**	**Why/How realistic**
Make the display table bigger so that more items can be seen at once.[1]	Less scrolling has to be done.	Easy to implement since can just change the ‘properties’ of the jTable.
Have an option for searching collector name.[1]	This is useful for the client since if a student takes an item that is not theirs, the Lost&Found team will be able to quickly find who did it since the collector name is already logged into the database.	This is feasible since the collector name information is stored as a data attribute in FoundItem objects. An algorithm can be made to select FoundItem objects from the array and search from there.
Keep count of how many Lost Items there are (instead of total of both lost and collected).[2]	This information is useful since the client will be able to plan to accommodate for more lost items.	This can be easily implemented. The program just has to differentiate between a LostItem object and a FoundItem object.
Connect the program to some sort of database so information that has already been inputted is saved when the program closes.[1]	The client does not have to worry about information being lost due to losing power, having to close the program, etc. It will be extra useful if the data can be saved online.	This will be harder to implement but is still feasible. An extra library will have to be imported in order to do this.

Other extensions:

- Add a signature option when manually inputting collector information for FoundItems.
    - Increases security since signatures authenticate collectors well.
    - However, this is less feasible since this means the program must be connected to some sort of drawing program.
- Connect barcodes to the ‘Item’ objects themselves
    - After the user manually inputs information about the item, there can be an option for selecting that item and creating a barcode for it (the barcode will contain information like ‘type,’ ‘name,’ etc. along with ‘date’).
    - The barcode can also be made a data attribute for each ‘Item’ object.

Word count: 338

[1] Dunke Tostevin, interview by author, Bangkok, April 5, 2020, transcript question #1, Appendix C

[2] Dunke Tostevin, interview by author, Bangkok, April 5, 2020, transcript question #2, Appendix C

Total word count: 1991