List of All Remaining Tasks to Complete by the End of the Semester

Roughly speaking, I believe that the order that these features are listed in are the approximate priority level for getting these features done in. So us trying to get this week’s requirements done first, then getting our old features back online, then implementing our new feature ideas, then finally hopefully get a PACMAN working in there for the extra credit. If you disagree with this ordering, let me know.

# New Tasks from Week 11’s Assignment 9

Each of theses are probably going to require Professor Carmon’s videos on them to be posted to work on them. If they are not posted in time of when we would work on these features, we’ll have to work on one of the later features until these videos go up.

## Migrate the logging system to use the BOOST library.

For this task, both our group and the professor is having problems with getting MinGW and the Boost library to link-up and like each other. Until we can get Boost working in QT, we’ll have to hold off on this task for now.

## Use QFile and a separate dialog box to create a notes dialog. Export the note to a .txt file.

DONE

# Finish Reimplementing Old Features That Were Disabled

## Purchase the Items in the Shopping List

Make sure to check the Quantity in the database to ensure that we don't accidentally sell what we don't have.

## Add a New User to the Database

For admins only.

## Change a User’s Password

For admins only.

## Import a .CSV File into the Database

Already partially started on this.

Is similar to the 2nd required feature listed for this week.

## Add a Book to the User’s Book List from the Inventory

This functionality is started, but it currently behaives more like the shopping list than the original functionality, listed below. We still need to add the extra stuff of what is listed here.

When a book is added to the Book List from the inventory, remove that record from the database.

If the user closes the program while the Book List is populated, then add those books back into the database.

May be combined with our 2nd new feature listed down below.

## Print the User’s Book List to the Screen

Old functionality for this is DONE

May be combined with our 2nd new feature listed down below.

## Export the User’s Book list to a .CSV file and Empty the Book List

May be combined with our 2nd new feature listed down below.

# Begin Developing 3 New Features

## Dedicated Login Screen (Mike)

So it’ll go Splash Screen, then Log In Dialog Box, and if they can log in it’ll bring them to the Main Menu Window. If they can’t sign in, then the program closes.

## Favorite Books List(s) (Chris)

Allowing the user to be able to add a book to their Favorites List. We can even save this Favorites List into a file so that the program will “remember” the user’s favorites after the program closes and reopens. If this is not enough for Professor Carmon, then we could shift to having it so that the user can create any number of their own Book Lists! (This has the potential to combine with our previous Book List feature and may require changes to that previous feature if we do combine it with this. As in, for example, saving all Book Lists as .CSV files instead of inside a 1 or more variables.)

## Admins Being Able to Edit Book Records (Jacob)

So, this is technically possible already with what tools the admins have right now with Exporting and Importing Books, but if we made it possible for them to edit the Book Records in the Database more directly without having to go through all those extra steps it’ll make it so much easier on them.

# Potential Extra Credit Features

## Pacman!

We probably won’t have enough time to make this ourselves, so if we do this it will probably be by porting someone else’s C++ PACMAN code into our QT Project as an easter egg.

I’ve already found at least 1 GitHub Repo online that is C++ PACMAN that we’ll probably be able to use, but I have not looked into it at all, let alone test it, to see if it will work for us.

One notable requirement is that this is meant to be an easter egg for ONLY the admins! So this must be a feature that only the admins will have access to, which means it’ll have to be hidden somewhere in the admin menu. That, or at the very least the easter egg will only trigger if the user is flagged as an admin.

# Clean Up Unnecessary Items in Our Application

This is mostly here because over half of our current top menu bar is filled with menu buttons that don’t do anything. We either need to clean them out or give them purpose. This can be extended to also remind us to clean up our code whenever we can, mainly commented out code.