CS-235 Assignment 2: Complete Software Implementation

February 19, 2019

Number of Credits: 50% of 15 credit module.
Deadlines:
Week 1's Minutes and Contribution Breakdown:
Monday 25th February at 11:00am
Week 2's Minutes and Contribution Breakdown:
Monday 4th March at 11:00am
Week 3's Minutes and Contribution Breakdown:
Monday 11th March at 11:00am
Week 4's Minutes and Contribution Breakdown:
Monday 18th March at 11:00am.
Final Submission Deadline:
Monday 18th March.
Learning Outcome: To gain further experience of implementing larger software as part of a team.

1 Problem Statement and Overview

In this assignment, each group will finish the implementation of an application, Tawe-Lib, as described in the previous assignment A1 of CS-230: Software Engineering I. A detailed list of the functionality and requirements specifications for Tawe-Lib are given in the A1 document from CS-230.

For this assignment, you are expected to complete the implementation that was partially completed in Assignment 2 of CS-230.

For this assignment, you have a choice of either using a code base that your group (or group member) has previously worked on for CS-230, or using the example project given in Assignment 1 of this module.

2 Functional Specification

2.1 Statistics

As previously stated in CS-230 Assignment 1, a statistics page is required in the eventual release version of the product and is to be developed at this time. The system should present users with information about how many items a user borrows per day, per week and per month (on a rolling basis). This information will be displayed both textually and graphically to a user using appropriate formats and should be accessible from the user dashboard.

Additionally, a librarian will also require a statistics page to view information about most popular resources (Books, DVDs, Laptops or Video Games) over a given time period (such as the past day, week or all time). A librarian will also be able to see statistical information about the fines that are issued to users of the system.

2.2 Resource Cap

As Tawe-Lib has grown in popularity, there is a need to limit the number of items that a user can check out at any one time. Each user may only have 5 items checked out at any one time and must return an item before a new item can be issued.

Due to the expensive nature of laptops, a checked laptop resource counts as 3 of the 5 allowed items in the system.

2.3 Video Game Resource

In addition to the existing resources available in the system, a new resource in the system named video games is required. The resource video game has:

- Publisher
- Genre
- Certificate Rating
- Multiplayer Support

As in CS-230, you will also need to build a user interface for allowing Librarians to manage the video game resource and copies. Librarians shall be able to create new resources and edit existing ones. When a Librarian creates a new resource (or edits an existing one) they must provide all the data stated above.

2.4 Ratings & Reviews

The Tawe-Lib system should now also support a method of ratings and reviews for all types of resources in a system. The system should provide an appropriate method of rating a resource, along with a place to leave a review.

These ratings and reviews should also then be viewable on the detailed view of a particular resource.

A user may only rate / review a resource that they have previously loaned from the Tawe-Lib system. A user can leave a rating without a review, but a review must always be accompanied by a rating.

2.5 New Additions

Each time a user logs onto the Tawe-Lib system, they will be able to view a list of all new resource additions to the library. This information should be displayed on the user dashboard page.

Here, new resources that a Librarian has added to the Tawe-Lib system should be tracked against each user login date, and new resources since the last login date of a user should be displayed the next time a user logs in.

2.6 Trailers

To give a clearer sense to users of the content that they are interested in, appropriate resources will offer trailers when they are browsed. Here, relevant video media should be played in an appropriate window with the typical toolbar controls available to a user.

2.7 Events

The library is now offering events (such as author book signings) that can be created and registered on through the Tawe-Lib system. Only a Librarian can create an event, which consists of:

- A title of the event
- The date and time of the event
- Maximum number of attendees
- Description of the event

Users will be able to browse events and register on events that they are interested, unless the event is fully booked.

Once an event has passed, it should be removed from the list of upcoming events. However, a user will be able to view events that they have attended previously.

3 Deliverables

3.1 Weekly Contribution Breakdowns and Minutes

You must submit weekly Contribution Breakdowns (see CS-230 Assignment Overview document). These will include the Contribution Breakdown itself and the minutes of the weekly meeting where you create the Contribution Breakdown. These will help document the contribution of members and the progress over the duration of the assignment.

Unlike CS-230, the Contribution Breakdown is now a digital submission via Blackboard, along with the minutes that accompany the report.

3.2 Implementation and Group Report

This section describes which files need to be submitted for the assignment and how they are submitted.

One member of each group, the Planning and Quality Manager, creates a folder named *"GroupGN-A2"* where *"GN"* is replaced by your group number. This folder will contain all files and folders that you submit (see below). Once this folder is ready, zip it up to create a file named *"GroupGNA2.zip"* and submit this single file to Blackboard.

3.2.1 Implementation Source Code

40% of A2. Your application source code (the .java files you wrote) is to be submitted. The source files should be placed within a folder named *"source"* within your *"GroupGN-A2"* folder. Please also include within the "source" folder a README file (named *"README.txt"* with basic instructions on how to compile and run your code.

The main method must be implemented in a class named *"Main.java"*. This class should contain no other method.

Your code must follow the coding conventions from CS-230 Lecture 13 (Coding Conventions). You must also comment your methods appropriately using normal Java comments (this is in addition to Javadoc). If your code is written well (and is self documenting) then you may need only a few uses of normal (i.e., non-Javadoc) Java comments.

3.2.2 Documentation

10% of A2. Your application should be fully documented with Javadoc as per the coding conventions in CS-230 Lecture 13 (Coding Conventions). Please also see CS-230 Lecture 15 (Javadoc).

A copy of the website produced by Javadoc (all files produced) must be included in a folder named *"doc"* within your *"GroupGN-A2" folder.*

3.2.3 Demo via Screen Capture

10% of A2. You must create a video that demonstrates the features of your application. You could use screen capturing software to do this. The audio or subtitles should clearly explain what is happening in the video. All marks for feature implementation will be awarded based on the video submission, as in CS-230.

You should submit one video that captures all the features and this should not exceed 10 minutes in total length. Any content beyond the 10 minute time will not be marked. Your videos should be placed in a folder named *"demo"* within your *"GroupGN-A2"* folder

3.2.4 Meeting Minutes

5% of A2. A copy of the minutes of group meeting for a minimum of three meetings held before the assignment deadline. The minutes should follow Bob's Minutes of Meeting Protocol. See blackboard for a copy of this document. You must combine all the minutes into a single file named *"minutes.pdf"* that should be placed directly within your *"GroupGN-A2"* folder.

3.2.5 Contributions Report

5% of A2. A description of what and how each group member contributed to the project is submitted. The format and content is identical to the submissions in CS-230 A1 and A2. The file should be named *"contributions-report.pdf"* and should be placed directly within your *"GroupGN-A2"* folder.

The Contributions Report, no more than 5 pages, describes who contributed to classes, implementations, and other contributions, e.g., the minutes etc. The Contributions Report is also written collectively. In other words, each group member describes their respective contribution to the project. The report also informs the reader if any unexpected problems arose during the course of the assignment. For example, if a group member skipped too many lectures, and as a result didn't have the background necessary to contribute, this should be stated in the group report. Likewise, if the group experienced success in some areas, both expected and unexpected, this is included in the report.

As with the CS-230 assignments, a confidential members contribution report can be submitted by any team member. Please refer to CS-230 A1 for further details.

Finally, this document should summarise any changes you made to the design during this implementation phase. It is normal to change the design during implementation, but please write a few sentences to document how you have changed it.

3.2.6 Summary of What to Submit and How to Submit it

When completing the submission of the assignment, the Planning and Quality Manager has a directory structure in their submission folder that looks like this (GN is your group number):

```
GroupGN-A2/
    demo/
    doc/
    source/
    contributions-report.pdf
    minutes.pdf
```

All the files and folders are zipped together in a single zip file and are submitted to Blackboard with the name *"GroupGN-A2.zip"* where "GN" is replaced by your group number. Marks will be deducted for those submissions that do not follow the file naming conventions and required file formats.

3.3 Individual Assessment and Viva

The individual assessment component is worth **30% of A2**. It is primarily there to evaluate the understanding of the project by each group member.

Viva Dates:

- Your viva will take place between **Tuesday 19th March and Friday 22nd March 2019** (see Blackboard for exact date and time). Please do not book any travel before your viva date. If you miss your viva, at best you will get a heavy deduction and at worst a zero mark on A2.

The viva will take place as a group, but its purpose is to evaluate each team member's (individual) contribution to the group project. This assessment begins with a short and informal five to ten minute presentation of the group's implementation. It should demonstrate how the group's implementation satisfies all the requirements of the implementation. Any group member, or combination of group members, can give this presentation.

After the summary presentation, each group member will be asked to individually define their contribution to both the design and implementation of the software product. For each group member, this contribution will include some software design and some coding in Java. Questions will also be asked to evaluate individual's knowledge of the product.

More specifically, you (as an individual) will have no more than 5 minutes to clearly explain two things:

1. How you designed/implemented your technical part of the system. You may explain using diagrams in your submitted reports or the source code of your system.

2. How your part of the system fits into the larger software product. You must spend at least a minute talking about the code implemented by another group member and demonstrate to us that you understand it.

Students are responsible for the full course material during their viva. We can ask any question from any lecture of the term. You must demonstrate sufficient knowledge of these materials in the viva.

4 Assignment 3 Preview

Assignment 3 is a re-engineering task. The feature specification given in Assignment 1 (Design) will change. Each individual will then implement those changes. The better the teams are at designing and implementing assignment 2, the easier this task will be. Assignment 3 also encourages each group member to be highly involved in the group work in Assignments 1 and 2. The more familiar you are with the previous work, the easier Assignment 3 will be.

5 Project Hints

The implementation must adhere to some rules:

1. Coding Conventions: Your team is required to follow the Coding Conventions presented in CS-230 Lecture 13 (Coding Conventions).
2. Code Documentation: Your team required to use Javadoc to document your source code. See Lectures 13 (Coding Conventions) and Lecture 15 (Javadoc) from CS-230.

Teams very often lose marks because they do not follow the coding conventions or they do not submit fully completed Javadoc documentation.

1. This coursework is very time consuming. Some teams usually lose marks because they don't manage to finish. Allow extra time for teamwork.
2. All group members are expected to contribute to both design and implementation. Group members who are weak at programming may get help from other members who are better at it. However, all group members are responsible for understanding the full submission. During the vivas, this will be tested.
3. It is best to have a robust and stable application with fewer features than an unstable program with all the features.
4. Test your application thoroughly. Your program should be able to handle random input without crashing. Strange inputs will be tested.
5. Allow your group time to learn how to use Javadoc and git (should you decide to use it).
6. Ask questions during the problem help sessions.
7. Why not use Skype for some of your group meetings?