

CAB301 Project:

Development of a Software Application for a Community Library to Manage Its Movie DVDs

Due Date: 25 May 2020

Weighting: 60%

Group or Individual: Individual

In this project you need to develop a software application for solving a real-world problem. In the development of this software application, you need to use some specified data structures to store and manipulate some data, and need to apply some algorithms that you have learnt in the lectures to solve some computational problems in the software application, and analyse the time complexity of the algorithms.

1. Preamble

You are hired to develop a software application for a community library to manage its movie DVDs.

The community library lends movie DVDs to its members. The information about a movie includes title, starring, director, duration, genre, classification, and release date. The genre of a movie may be Drama, Adventure, Family, Action, Sci-Fi, Comedy, Animated, Thriller, or Other. A movie is classified as General (G), Parental Guidance (PG), Mature (M15+), or Mature Accompanied (MA15+). The community library may have multiple DVDs of the same movie.

In order to borrow movie DVDs, members must register with the system through a librarian. When members are registered with the system, their full name, residential address, and a contact phone number are recorded in the system. The system also keeps the information about the movie DVDs that are currently being borrowed by each of the registered members.

When a registered member borrows a movie DVD, the movie DVD is added to the member's record; and when the member returns the movie DVD to the community library, the movie DVD is removed from the member's record. A registered member can borrow up to 10 movies at any time. However, none can borrow multiple copies of the same movie at the same time. It is assumed that both the full names of members and the titles of movies are unique.

The system analysis has now been completed, and a high-level object-oriented design has also been finished. Your task is to follow the following iterative and incremental software development process to implement the object-oriented design.

2. The Software Development Process and Tasks

- 1) Design and implement a class *Movie* to model a movie DVD. In this software application, each movie DVD is represented by an object of the class *Movie*.
- 2) Design and implement a class *MovieCollection* to represent a collection of movie DVDs (a collection of the objects of the class *Movie*). The class *MovieCollection* must use a binary search tree as a class member to store the movie DVDs.
- 3) Design and implement a class *Member* to model a library member. In this software application, each registered member is represented by an object of the class *Member*.
- 4) Design and implement a class *MemberCollection* to represent a collection of registered members (a collection of the objects of the class *Member*). The class *MemberCollection* must use an array as a class member to store the members.
- 5) Make use of the above reusable classes to build a software application that meet the following functional and non-functional requirements:
 - Both the staff and all the registered members can use the software application. The software application has a main menu and two sub-menus. The identity of a user must be verified in the main menu before he/she can use the system as a registered member or a staff member. It is assumed that the username and the password for the staff are 'staff' and 'today123', respectively, and that the username of a registered member is the member's name in one word. For example, if a member's surname and given name are 'Smith' and 'John', respectively, then the member's username is 'SmithJohn'. The password is a four-digit integer of the member's choice.
 - The software application should have a staff menu that allows the staff to do the following:
 - Add DVDs of a new movie to the software application
 - Remove a movie DVD from the software application
 - Register a member with the software application
 - Find a member's contact phone number, given the member's full name
 - The software application should have a member menu that allows a registered member to do the following:
 - Display all the information about all the movie DVDs in alphabetical order of the movie title, including the number of the movie DVDs currently in the community library
 - Borrow a movie DVD from the community library, given the title of the movie DVD
 - Return a movie DVD to the community library, given the title of the movie DVD
 - List current movie DVDs that are currently being held by the registered member
 - Display the top 10 most frequently borrowed movie DVDs by the members in the descending order of the frequency. The display should include the

title of the movie and frequency (the number of times that the movie DVDs have been borrowed by registered members so far).

- The main menu and the two sub-menus should look like:

```
Welcome to the Community Library
=====Main Menu=====
 1. Staff Login
 2. Member Login
 0. Exit
=====

Please make a selection (1-2, or 0 to exit):
```

```
=====Staff Menu=====
1. Add a new movie DVD
2. Remove a movie DVD
3. Register a new Member
4. Find a registered member's phone number
0. Return to main menu
=====

Please make a selection (1-4 or 0 to return to main menu):
```

```
=====Member Menu=====
1. Display all movies
2. Borrow a movie DVD
3. Return a movie DVD
4. List current borrowed movie DVDs
5. Display top 10 most popular movies
0. Return to main menu
=====

Please make a selection (1-5 or 0 to return to main menu):
```

3. Assignment Requirements

- The software application must be implemented in Microsoft Visual C#, C++, Java, or C. Having said that, you are not encouraged to use C as the design is an object-oriented one while C is not an object-oriented programming language.
- You may create and use other classes in the software application, if you wish.
- You may not use any third-party class libraries.
- You do not have to store members' information or movie DVDs' information into files.

4. Submissions

Your submission should be a single zip file named by your_student_number.zip, and comprises of a complete console application project and a report in PDF covering: (1) an algorithm that you designed for implementing the function “Display the top 10 most frequently

borrowed movie DVDs by the members in the descending order of the frequency”; (2) analysis of the time complexity of the algorithm; (3) your functional testing results - screenshots for each of the functional tests.

Re-submission is allowed as long as it is before the deadline. If you submit multiple versions of your assignment, we will only mark the last one before the deadline.