CSE 1325-001 Homework #8-12 – Project

Due Dates:

Sprint 1 – Friday, March 30th, 11:59 PM
Sprint 2 – Sunday, April 8th, 11:59 PM (delayed because of Test 2)
Sprint 3 – Sunday, April 15th, 11:59 PM
Sprint 4 – Sunday, April 22nd, 11:59 PM
Sprint 5 – Sunday, April 29th, 11:59 PM

Introduction

In this project, you will use the skills you have learned throughout the first 2 parts of the class, as well as incorporate skills from the 3rd part of the class. The goal of this project is to create a Media Management System for a Library. This project will consist of both a Command Line Interface (CLI) and a Graphical User Interface (GUI). You must use C++, and show mastery of the key Object Oriented Concepts (Encapsulation, Inheritance, and Polymorphism).

There are 5 kinds of media in the library: Books, Seasons of Television Shows, Video Games, Music Albums, and Movies. There are also three different kinds of users: Product Manager (PM), Librarian (LB), and Customer (CU).

Roles

The Product Owner has identified the needs of each of the 3 actors that will interact with the system, and has assembled these into a prioritized Product Backlog in the Scrum Spreadsheet, which is included in the documentation on Blackboard.

Product Manager (PM)

The PM needs to be able to add new instances of media in the system. For each media, they will need to specify an id number, call number, title, and genre. In addition, each book will need an author and copyright year. Each movie will require a release year, producer, director and a list of the leading actors. Each video game will require a release year and the studio that made it. Each music album will require a release year, artist, and a list of the tracks. Finally, each television show season will have a release year, producer, a list of directors for that season, a list of leading actors, and a season number.

The PM also needs to create bundles by selecting one or more already existing media. Only one copy of each media can be added (no duplicate books or movies). For each bundle, the PM needs to define the name, ID number, and the list of media that it contains. In order to create the bundle, they would need to see the list of media first, then type in the ID number of those media.

Lastly, the PM needs to be able to create librarians. When creating librarians, a name and id is required.

Librarian (LB)

A LB needs to check in and out books and bundles. To check out a book, they would see a list of books and bundles. Then the librarian would type in the media or Bundle ID number and the Customer's ID number. To check in, the LB would check the media ID number or customer ID number to see if there

are any media that are currently checked matching those conditions. The LB would then modify the transaction to say the media was checked in. The fee would then be calculated ($0.50 / day) and would be put on the Customer's balance.

Customer (CU)

CU needs to be able to browse a catalog of media in the library with pictures (in extra credit versions) and see whether or not the media is checked out. Each CU will also need to see what they have checked out and if they have any fees to pay. A CU may not check out books if they have a balance.

Process

Each student will follow a simplified Scrum process. The Product (or Feature) Backlog covers the full anticipated scope of the project – all 5 weeks.

The developer (you) is free to negotiate changes to the Product Backlog and priority with the Product Owner (professor). If a change is approved, an email from the Product Owner clearly stating the change in the Product Backlog is acceptable.

You must provide your own updated UML models and design your own menus. This is part of the assignment. A basic, incomplete UML class diagram is provided to assist your understanding, but you must provide a complete diagram yourself. Update your UML design before you start coding. Neither the professor nor TA will discuss your project with you at all unless you bring your own UML models and up-to-date Scrum Spreadsheet with you.

Every class documented in the standard C++ library remains available for your use on this project. When it comes to the GUI, you just use the package discussed in class (TBD). You may consider third party libraries that have been approved in advance by the Product Owner in writing, as long as you comply with all license agreements.

The product baseline should be built frequently. All automated tests that were created to identify any breakage needs to be added to the Sprint Backlog of tasks, prioritized, and addressed. Build your program as a small set of functionality components that can be tested, then compile it and test it. Note any bugs found in the Sprint Backlog (task list) for prioritization and eventual resolution. Do not write all the code for the entire project then ask us for help getting it to compile. It should be compiled incrementally. Then if something breaks, you know if it was something you just wrote, instead of it being anywhere in your code. This is how it is done in the real world.

In lieu of weekly demos, students will submit weekly via Blackboard at the end of each sprint (see deadlines section at the top). You are expected to complete at least the features indicated in the "Expected Sprint" Column on the Product Backlog.

Final work projects are due in Blackboard on Sunday April 29th at 11:59 pm. No extensions will be given. If you develop incrementally, you will have a working program on that date. You will demo this project for extra credit during the second to last class.

Bonus

Unlike previous homework assignments, your opportunity for bonus credit is primarily rooted in completing additional features within the 5 sprint project limit. Thus, it is to your advantage to complete features ahead of the sprint schedule to allow time for additional features to be completed by the end of the scheduled project.

Each additional sprint completed is worth an additional 100 points. All items in expected sprint 6 must be completed before any items in expected sprint 7. If you wish to implement features out of priority order, be sure to receive written approval from the Product owner in advance.

Fine Print

As mentioned before, the Product Backlog will change during the project. Agile software development is all about effective response to change. You will have the opportunity to experience this over the next 5 weeks.

Deliverables

At the end of each sprint, the following must be submitted:

Code for the entire project up to the point of where you are (zip file).

UML Class Diagram properly reflecting the state of your project. (xmi file)

UML Use Case Diagrams properly reflecting the state of your project (starting on sprint 2) (zip file containing all use cases)

Scrum Spreadsheet representing the total project. (xls)

User's manual (end of sprint 5 only). Only has to be a few paragraphs. (pdf)

Demo (in class after sprint 5). Up to 25 points bonus.

Be sure to follow the naming conventions from previous homework assignments on files. For UML files, feel free to use any program you want. If you use something other than Umbrello, you must submit either a jpg or pdf.