HO CHI MINH UNIVERSITY OF SCIENCE
FACULTY OF INFORMATION TECHNOLOGY
SOFTWARE ENGINEERING DEPARTMENT
APCS – CLC
COURSE: **CS162 – KTLT**
LECTURER: Dr. ĐINH BÁ TIẾN

FINAL PROJECT

A LIBRARY MANAGEMENT SYSTEM

- TRƯƠNG PHƯỚC LỘC
- HỒ TUẤN THANH

HCMC, February 25, 2016

Table of contents

Each group is optional to choose one of two following projects.

1 Project 1

1.1 Overview

You are asked to implement a program to help the teachers in manage the library of multiple-choice questions. The library including a set of multiple-choice question from several subjects aims to help teacher in creating examination for students and help students to study.

Each mulltiple-choice question includes the question, a list of options, the right option(s). The questions belong to different subjects, e.g. mathematics, data structure, database. Each question has an author (the teacher contributed it). Each teacher is correspond to one or more subjects. Be noted that some questiones are view-able to students while the others are not. The teacher should indicate that feature while creating a new question. The student can view only view-able questions.

There are three types of users including Administrators, Teacher, and Student. All users need to login to the system before using any features. Common functionalities for all 3 types of users include

- Login.
- Logout.
- Change the password.
- Update the user information.

1.1.1 Administrators

Administrators are able to see the list of all accounts in the system. From this list, he/she can reset the password of any account, remove or ban that account. Since the number of accounts can be very large, your program should have a feature of filtering the list. The list can be filtered by the folowing criteria:

- User types (show a specific type of user only),
- Account status (show accounts whose this status is set on),
- Username (show all accounts with this user name),

Your program should allow admins to sort that list according to the above criteria. Besides, admins should be able to create a new admin account or a new teacher account. The program should also have the feature to output the list of users as an XML file. For security, the passwords should not be included in this file. Finally, since banning or

deleting accounts may cause serious impact to related users, the program must warn the admin before executing this operation.

1.1.2 Teacher

Each teacher correspond to one or more subjects. He/she can view/alter/delete/add all questions in his/her subjects. The teacher can search the question by key words, authors.

1.1.3 Students

A student could view the view-able questions. They can search question by subjects, keywords, and author. The student could choose view or not view the answers belong with the questions.

1.2 Note

1.2.1 User information

- Username (if the user is student, username must be student id)
- Password
- Name
- Date of Birth
- Address
- Sex
- Type of user
- Status (activated/deactivated)

1.2.2 Question information

- Subject
- Author
- Viewable or not
- Multi answer or single answer
- List of options
- Answer

1.2.3 Technical requirements

- Use the linked-list as your main data structure,
- Should include comments for all functions,
- Put each class in two separate files: a header file (.h) and a implementation file (.cpp)
- All data should be stored in files.
- Password must be encrypted.
- The data files' structure is self-defined and should be documented.

1.2.4 Policy on submitting the project

- The project can be done by a group of at most 2 students.
- You should submit only one project compressed and named by this format: <StudentID1_StudentID2>.zip or <StudentID1_StudentID2>.rar.
- Delete the folder Debug before submitting your project.
- You will receive 0pts if we cannot compile your source code.
- Your project should already contain a data file containing some accounts, questions, etc.
- Give the percentage of your project completion in a checklist file (to be sent).
- Attach the checklist in your project.
    - No checklist or empty checklist → 0pts.
    - Implemented features which are not indicated in the checklist → 0pts
- Your rar/zip file should contain the folder Source and the checklist file.
- We have zero tolerance for cheaters.
- We have zero tolerance for submitting the solution after the deadline.

2 Project 2

You are asked to implement a program to demonstrate how a singly linked list works.

1. Initialize: empty list, random list, load from file
2. Random a number/Allow user enter a number to insert/find/remove…
3. Insert: head, tail, before a node, after a node
4. Find: a specific number, in range, less than, greater than.
5. Traverse list and output the result
6. Remove: head, tail, a specific node
7. Save list to file

Notes:

- Run-step-by-step mode, Run-once mode
- Groups choosing this project should take a look here.
    - Link: https://www.cs.usfca.edu/~galles/visualization/AVLtree.html