TCS3344 – Advanced Programming
Jan – May 2024

Assignment 2 Brief and Marking Schemes

This assignment contributes **100%** of the total coursework mark. Please refer to the following assignment marking rubrics.

THE FOLLOWING LEARNING OUTCOMES WILL BE ASSESSED:

Skills

- Apply appropriate OOP techniques in developing solutions to problems.
- Develop a computer program by applying OOP techniques in data source-based applications.

General Instructions

Use the following format for the preparation of the *assignment submission*.

- Commentaries are required in the code.
- Spacing and indentation are required.
- Source code and all related sources (source code, executable file and resource file)

WARNING

- Assignments submitted after the due date will be considered late.
- Assignments submitted not later due date will not be entertained.
- SEGI University takes allegations of plagiarism very seriously. Submissions involving plagiarism will be marked but given zero marks. Plagiarism is the attempt to pass off the work of another as your own. Information taken from the work of others should be acknowledged by reference to obviate the charge of copying.
- Collusion is an academic irregularity within the SEGI University assessment regulations. Any student found colluding in the production of any assessment will be subject to an investigation with the imposition of any penalty deemed appropriate. Students must ensure they are familiar with the definition of collusion.

I. Assignment Requirements

Introduction

A Student Management System (SMS) is a software application designed to manage various aspects of student information within an educational institution.

This typically includes features like managing courses, calculating marks, and keeping track of student details.

the basic functionalities and components you might want to consider when developing a Student Management System that handles course marks and student information:

User Authentication:

In the authentication section, you only need to demonstrate a hard coded validation of user and switch them to dashboard. There will be only one username with one password. Anyone with the username and password can login to dashboard.

Dashboard:

This will be a lecturer dashboard. Lecturer is allowed to perform the following tasks:

a) Manage students.
b) Manage courses.
c) Manage mark and grade.

Mange Students:

Add, edit, and delete student records.

Store personal information (name, contact details, address, etc.).

Assign unique identifiers (like student ID).

Course Management:

Add, edit, and delete courses.

Assign courses to specific semesters or terms.

Define prerequisites and credit hours.

Mark and grade Management:

Add, edit, and delete marks for each module.

Show the grade (GPA and/or CGPA)

Tasks

Your application should contain the following functionalities:

a) Design and implement classes that Student class. In addition, use array list to store course class within the student class as well as array list to store information for multiple students.

b) Users require to use multiple forms to simulate authentication process. Users will be presented with a dashboard after login to the system.

ÿÿ c) ÿÿAll information must be saved in persistence storage using text, binary file or database.

Submission Requirements

Upon submission, please submit the following items:

- All source code files of the program. A compressed folder of the project and a separate jar file should be submitted to Blackboard.
- And executable file in the jar format.

Note

Plagiarized work will result in a ‘FAIL GRADE’ Major Computer Software List – NetBeans

Minor Computer Software List – GIMP, Ink Scape, or any necessary tools.

II. Assessment Marking Criteria

Topic/Area of concern	80 – 100 marks (A)	65 -79 marks (B, B+)	50 - 64 marks (C, C+)	40 -49 marks (D)	0 – 39 marks (F)	Grade/ Comment
Multimedia Creation						
Layout and User Interface design.	Excellent layout and UI design with proper form and dialog box implementation. In addition, it contains proper event handling that makes the navigation work smoothly.	Good layout and UI design with proper form and dialog box implementation. In addition, it contains proper event handling that makes the navigation work smoothly.	Proper layout and UI design with proper form and dialog box implementation.	Limited layout and UI design.	Bad layout and UI design.	/10
User Management System	Well designed UI and easy to use learn and use functionality of the sub system.	Proper UI interface design and user-friendly functionality.	Proper CRUD functionalities for UMS.	Limited function for UMS.	No and limited function.	/20
Course Management System	Well-designed UI and easy to use learn and use functionality of the sub system.	Proper UI interface design and user-friendly functionality.	Proper CRUD functionalities for CMS.	Limited function for CMS.	No and limited function.	/20
Student Management System	Well-designed UI and easy to use learn and use functionality of the sub system.	Proper UI interface design and user-friendly functionality.	Proper CRUD functionalities for SMS.	Limited function for SMS.	No and limited function.	/20
Persistence storage must use to store data and information in the application	Excellence plan on combination of persistence storage is used with proper design of serialization in the file system.	Proper persistence storage is used with proper design of data structure store in the file system.	Proper persistence storage is used.	Limited persistence storage is used.	No persistence storage implemented.	/20
Presentation	Excellent presentation and rich information provided about the application.	Well-presented all necessary information and easy to understand.	Proper presentation with all necessary information about the application.	Presented only with a little information about the application.	No presentation or provided limited information.	/10
FINAL TOTAL:						**/100**