CS 1302 – HW 10

*Gradebook GUI Application*

Contents

[1 Introduction 1](#_Toc164088565)

[2 Problem Statement 1](#_Toc164088566)

[3 Development Requirements 2](#_Toc164088567)

[4 Rubric & Status 2](#_Toc164088568)

[5 Time Log 3](#_Toc164088569)

[6 Submission Requirements 3](#_Toc164088570)

# Introduction

For this HW you will develop a GUI based application to manage a gradebook for a course. This HW counts as 3 homework assignments.

# Problem Statement

A student has a name and a 4-digit id as well as any number of test scores. A course has a name (*e.g.* CS 1302, *etc.*) and any number of students.

Develop a GUI[[1]](#footnote-1) based system to implement the user stories below.

User Stories: As a professor, I want to…

1. Add a course
2. Add a student to the course
3. Display all students.
4. Save all students to a text file
5. Read students from a text file when the system first starts.
6. Display the grand average test score (the average of each student’s average)
7. Display the student with the largest average test score
8. Delete a student
9. Change the name of a course
10. Change a test score(s) for a student

You have liberty to develop these use cases and required classes any way you like. There are different ways to implement these – use your own judgment – you do no need to ask me.

# Development Requirements

These are general development requirements.

|  |  |
| --- | --- |
| **Requirement** | **Penalty[[2]](#footnote-2)** |
| 1. The system must be coded in Java, object-oriented, and utilize best practices we have learned in class and the text.
 | 50 |
| 1. The code must be developed solely by you. You may use any of the GUI examples in the code download for Chapter 11.
 | 100 |
| 1. The Status of each User Story must be stated on the provided spreadsheet
 | 10 |
| 1. The Time Log on the provided spreadsheet must be accurate and complete
 | 2-15 |

# Rubric & Status

Accompanying this document is a spreadsheet, *hw10\_timelog.xlsx*. On the Status tab, you’ll see data as shown below that shows the rubric. You will complete the “Status” for each of the User Stories.

|  |
| --- |
| **Provide the Status of each user story** |
| C=Complete, I=Incomplete, empty=not attempted |  |
|  |  |  |
| **OO Design** | **Points** | **Status** |
| Implementation of classes | 20 |  |
| Class diagram | 3 |  |
| **Testing** |  |  |
| Unit Tests | 10 |  |
| **Implementation of User Stories** |  |  |
| Add a course | 8 |   |
| Add a student to the course | 8 |   |
| Display all students ordered on name | 8 |   |
| Save all students to a text file | 8 |   |
| Read students from a text file | 8 |   |
| Display the grand average test score | 5 |   |
| Display student with largest average | 5 |   |
| Delete a student | 5 |   |
| Change the name of a course | 5 |   |
| Change a test score for a student | 5 |   |
| **WOW Factor[[3]](#footnote-3)** | 5 |  |
| **Total** | 103 |  |

# Time Log

Accompanying this document is a spreadsheet, *hw10\_timelog.xlsx*. On the Time Log tab, you will record all time you send on this HW in a timely and accurate manner. A sample of the Time Log is shown below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | Automatically Calculated |  |
| **Beg Date/Time** | **End Date/Time** | **Int.Time** | **Tot Time** | **Run Total** | **Brief Description of Activity** |
| 6/10/20 12:30 PM | 6/10/20 1:25 PM | 0:15 | 0:40 | 0:40 | Sketched initial class diagram |
|  |  |  |   |   |  |
|  |  |  |   |   |  |
|  |  |  |   |   |  |

# Submission Requirements

Checklist:

|  |  |  |
| --- | --- | --- |
|  | **Complete?** | **Requirement** |
| 1. |  | The class diagram is included as a *jpg* file, oriented correctly, or placed on a Word or PDF document. It is placed in the package folder. |
| 2. |  | The completed (status and time log) spreadsheet, *hw10\_timelog.xlsx* is placed in the package folder and is readable (you should test that your spreadsheet opens correctly on a windows machine to avoid losing 25 points). |
| 3. |  | There is a text file with sample data that can be read into your system located in the package folder. |
| 4. |  | All java files (including test classes) are in thepackage folder. |
| 5. |  | Your *package* folder is zipped into a file name: *hw10\_yourLastName.zip.* * See Lab 2, Stage 9 for exact instructions.
* Do not zip your workspace folder, do not zip your src folder.
 |
| 6. |  | Submit in the *hw10* dropbox on Blazeview by the deadline. |

1. A menu-driven system is allowed for the students who previously responded to my email about having difficulties with JavaFX [↑](#footnote-ref-1)
2. If applicable, the penalty is subtracted from you score on the project. [↑](#footnote-ref-2)
3. How well does the GUI pop? How does it flow? Easy to use? Designed well? [↑](#footnote-ref-3)