Fall 2022 – CS203 Object Oriented Programming
Homework -Bonus-

Deadline: 11/20/2022 Sunday 11:59pm

****** This is a bonus homework, it is not required to submit it. Your lowest hw grade will be replaced with this homework grade******

Flashcard Application - Leitner System

Objectives:

- To practice Public Interface, Encapsulation, Unit Test, and Class Implementation

Flashcard Application

In this homework, we will implement a flashcard application (https://en.wikipedia.org/wiki/Flashcard) based on the Leitner system (https://en.wikipedia.org/wiki/Leitner_system).

Flashcards are frequently used to study vocabulary, symbols, and programming constructs. A `flashcard` is a card bearing a challenge on the front side and a response on the back side. A challenge (or response) can be either text, an image, or sound. A user can `create`, `modify`, and `delete` flashcards. Modify and delete functionalities are optional and bonus points will be provided(+10 points each). For each side of the flashcard, the user is able to either enter text or select an image or sound file from disk. For this homework, we will only use the **text**. In this implementation of the Leitner system, flashcards are organized in *`five boxes`* (numbered 1-5). Once a new card is created, it gets stored in Box 1.

Studying with the Leitner system: The application randomly picks a card from one of the five boxes. Note, the likelihood of a card to be picked from a lower numbered box is exponentially higher than from a higher numbered box. For example, for a card in Box 1 it is twice as likely to be picked as a card in Box 2 (four times as likely as a card in Box 3, ...). If the user knows the answer, the flashcard moves to the next box (if there is one). If the user does not know the answer, the flashcard goes back to Box 1. Once a card has been picked, the application randomly decides whether to show the front or the back of the card. Users have four choices:

(a) to flip the card,
(b) to indicate that they know the answer,
(c) to indicate that they do not know the answer,
(d) to end the study session.

Task:

- Read the problem description and identify candidates for classes.
- Assuming that among other classes you have identified **`FlashCard`** and **`Box`** as class candidates.
- Create a public interface for these two classes (including JavaDoc comments). For now, a flash card can only be **text** (no image or sound).
- Design **`unit tests`** for the two classes (using JUnit is optional).
- Implement the classes and use the unit tests to validate your implementation
- It is your task to identify the necessary classes, attributes and methods. Make sure you implement the core methods such as; create a new flash card, display the flash cards…
- Create a user interface that shows the options to the user and get input from the user

Homework Submission Instructions

- ***Deliverables:***
    - A well written project report (.pdf)
        - Explain your classes, attributes and methods. Explain how you implemented the Inheritance, and benefits of this implementation
        - Put the screenshots of your outputs (the working version of your code)
    - Project.zip file
        - Submit the working version of the whole project file
        - Do not submit only the classes

- Submit your files to Canvas (other submission methods will not be accepted)

If any of the problem statements is unclear, use the Canvas discussion board to ask for clarifications.