Milestone

Nicholas Bills

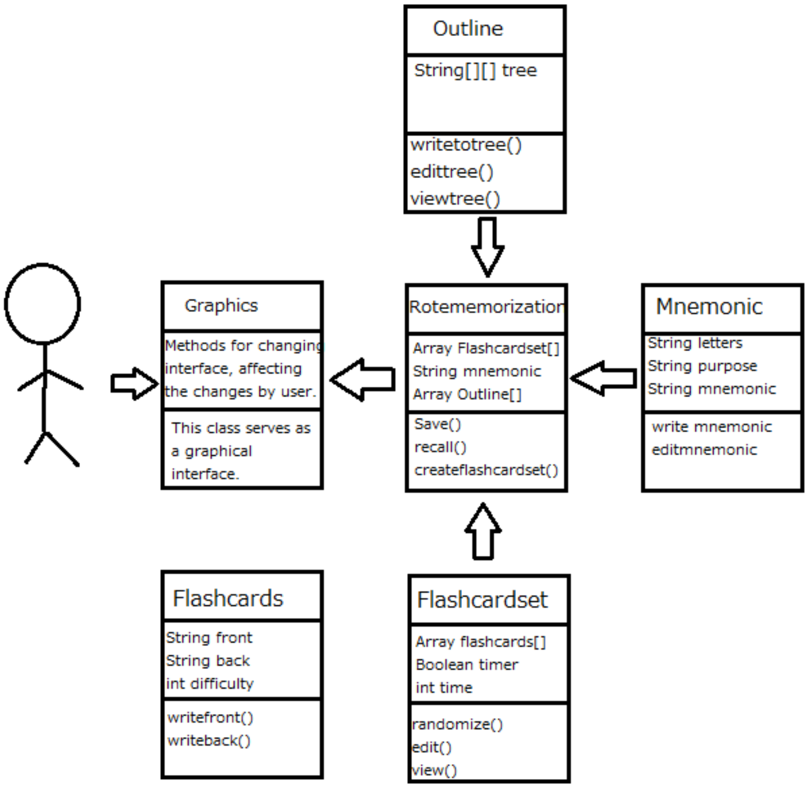
Abstract

Memorization techniques are integral to students to succeed in their coursework. When I came up with this project my idea was that I needed a way to memorize information that forced me to study in smaller bits than the usual binge studying I do. This program is meant to utilize many techniques of rote memorization as well as other techniques in order to develop memorization of topics. This is still very much a work in progress but thus far these are the avenues I have decided to approach. I have decided to just make a very good program instead of attempting to link it to another area of the project. s

Introduction

Rote memorization is a memorization technique based on repetition, and is a common form of memorization across the world. Techniques for rote memorization include the use of flashcards, outlines, and mnemonics. This program aims to utilize all of these techniques to offer students the best ability to learn. This will be done through three distinct sections of the program a flashcard section, an outline section, and a mnemonic section. The flashcard section of the program will possess multi-sided cards that can contain string values or pictures and will be able to be viewed in normal flashcard manner, but will also have the option to be set for a subset of the cards to pop up at timed intervals in order to drive rote memory by randomized recollection. The outline aspect will be structured by the student but will be able to be viewed and edited. The mnemonic aspect will possess a generator that creates a mnemonic from the letters put into it by the user. Each of these techniques is useful for creating successful memory links in the brain and develop the abilities of rote memorization. I have foregone some of the other ideas in my proposal in favor of a more integrated memorization program.

Detailed System Description



This program will possess a GUI class, Graphics, that will control the interface that the user interacts with. The interface will open upon starting the program and will possess buttons detailing the various options the user can undergo. As of this point, I have developed a primitive interface from the JAVAFX framework that can react to some specific user input. I have also had to move to Eclipse to run my program as notepad++ would not display the desired graphics. I want to develop the interface so as to streamline the use of the program as well as develop more complex operations. The program will have a masterclass, Rotememorization, that connects the GUI class to the other classes controlling the various aspects of the program. The Rotememorization class will be important because it will possess the ability to login to a profile, view profile statistics, save the various data files created by the user, to recall the saved files, display a list of the saved files, and to create any of the various memorization techniques. It should also have a method that reads through flashcard sets to determine if they possess the variable to be opened at time intervals, and should open the subset of the flashcard set to be questioned. Thus far, this class has been created and possesses the ability to save variables found within through the use of serialization. The way I have developed the serialization is weak at the moment, and can only create ".dat" files which has been having difficulties being changed upon recall and manipulation.

Flashcard

This will be the primary memorization technique utilized by this program. The flashcards will be created as objects with few variables outside of the information they are meant to relate to the user. One of the variables they will possess is an integer containing the difficulty of the flashcard. These flashcards will be held in an object, Flashcardset, which will contain them in an array. The flashcardset will have a Boolean timer which determines whether the user wants this set of flashcards to be opened at intervals. It will also possess an integer outlining the number of minutes the user wants between when the flashcard set will be opened. There will also be a Boolean random which creates a randomized amount of time between the flashcard launches. Along with these variables it will contain the methods to view cards, edit flashcardsets, functions to create subsets that can be viewed. I also want to make an algorithm that edits the cards added to the subset overtime based on the users understanding of the material and their input into the flashcard difficulty. I have created the flashcard objects and connected them to the flashcardset objects. Currently the ability to create and view them must be done within the console but I will shortly be able to control them through the GUI interface.

Outline

The outline memorization technique is useful for memorizing things that are related to each other. It is going to be held in an object outline. The manner in which the outline will be created and stored is still under deliberation. One manner could be a two dimensional array to create a tree but then the relationship may become mixed. More objects or variables may have to be created. This will have the ability to view the outline as well as save and recall much like the other classes.

Mnemonic

The mnemonic is an important memory technique for objects grouped together. This will be in an object that takes a string of first letters of the objects needed to be turned into a mnemonic. The object will have a method that writes a mnemonic for the letters given, however good it is. The user will be able to recall the randomized mnemonic if they do not like the outcome or they will be able to personally edit individual values if they like some of the input values. The mnemonic can be saved and recalled much like the other objects found in this program.