Jarratt Schleehauf

CS 1061

Project Proppsal

Flash Card C++ Application

The purpose of this application will be to allow a user to view and create notecards/flashcards for study. This is a text-based application so much of the functionality will be centered around an interface that allows the user to navigate from a main list of options. From this list the user can create a new deck of cards, edit an existing deck, review/ study a deck.

1. Algorithms

The algorithms for this project will be broken up into subproblems following a structured design to solve the overall problem of making this application functional.

As described above, the user will enter a main menu with a list of options. From this list the user will be able to build a deck, edit and existing deck, or review a deck. I will consider adding the ability to create a quiz from a deck but will be dependent upon the complexity after further investigation. From these main options there are sub options, such as when editing an existing deck, the user has the option of creating a new card or deleting cards. Also, under the review/study option, the user can toggle between being presented the front or back of a card first. They will also have the option of hiding cards they already know in that deck.

1. UML diagrams

I will need to become more familiar with UML diagrams to be able to effectively translate what is happening with the program. Essentially, the diagram will visually depict the components of the program and the data that will be created and how the components interact.

1. Inputs and Output (I/O)

At this phase of the exploration of the project, it seems that the majority of the data will be in the form of strings. Through prompt lines and user input the program will store the data. The building of flashcards will require significant user input, although not in an overly complex way.

1. Variables

I will need to research some options of how to implement front and back of a flashcard. This could be a place to use variables to store the string for the front, such as flashCard1Back, flashCard1Front.

1. Arrays

From the last point about the variables for front and back, each deck could be stored as an area consisting of the variables representing the front and back of each card.

1. File I/O

It is not clear whether the program would need to read or write to a file to function as intended. Also, it will be worth further study to determine if it will improve or hinder the performance of the program to do so. Also

1. Iteration (loops)

I can see that in the phase of flashcard study, the program will continue to cycle through the deck until the user prompts to exit to main menu. I would like to build a previous and next card option into the study component, which would affect the loop.

1. Interaction

One of the tricky functions is how to hide a card from a deck after the user has indicated they wish to do so. My assumption is this will need to be built into a loop for the card cycling and where that card in the array is passed over the next cycle through the deck. As indicated in other sections, the user will be given prompts, with sufficient guidance on available options and constraints, while they are navigating the application.

1. Control

Most of the application is menu driven, and when the user is in a particular function, there will be a lot of switching between display of output and user input. For instance, as a user is studying flashcards, the front of the card will be displayed, then the program will switch to input from user. When in a deck and cycling through, the user has the option to go to next card, hide card, go to previous, quit deck and select another, shuffle deck, and exit to main menu.