Kathleen Levi

Dr. Tucker

CS-172

19 June 2015

Requirements Specification

*A clear problem definition. State the requirements – what is required of the system? What must it accomplish or provide? Do you make any assumptions?*

First the system must introduce the game asking the user if they know how to play and their name. The system must also show the rules(possibly using switch statements to navigate). In order for this program to work the system must shuffle the Uno cards and deal the cards to the player and the computer. It also must dedicate the numbers and colors to each card so that there are no duplicate cards in the deck and then it will take the top card off of each stack as the cards continue to be used and put onto the discard pile. The system must also test whether the card that the user or computer chooses to play matches at least one of the variables of the card in the discard pile- either in number, color or symbol. The system must also use the random cards like the add two and skip cards to deal the correct function of each of those cards to each player. Also, when a player wins the game by running out of cards the game must end and congratulate the winner while also adding up the scores and saving them to a text file for the player and for the high score sheet as well. The game must also ask if the user would like to play again and if so, start the game over and play another round while re-shuffling the cards and beginning a new game for player and computer.

One definite assumption that I am making is that I know how to code all of these things and put them together in order to make the game work. I don’t know if I know enough or will be able to code enough to make the functions of the wild cards work but hopefully I can learn and make the program work at least on the most basic level. I am also assuming that I will be able to use vectors properly so that the user can choose which card to play and then it will take that card out of their hand and play it on the deck. I am also assuming that I can easily decide which cards are which and I will be able to assign them properly because unlike a normal 52 card deck there are something like 102 cards in an Uno deck with only a few numbers and 4 colors, so that will also be a challenge, but I’m assuming that it will be similar to a 52 card deck… praying its not much different. I am probably making other assumptions as well but I think a lot of them have to do with my ability to code and see whether or not I will have the knowledge required to code this project. Shouldn’t be too bad.

*The design of your project. For instance, what classes, what properties, what behaviors, etc. will be required?*

For the design of my project I am thinking a lot of the code will be in main besides the person class that I will have in my project. The person class is the only class that I can think of that would work in my code since my deck of cards is going to be a vector and/or stack functions. The person class will have name, score and cards as private variables and then the functions will set and get those variables, and for the cards it will add and subtract a card from the players hand. That way I can eventually have a multi-player game in my code and will be able to save multiple players as well.

I also think that I will need to have a way to shuffle and deal cards. Which I will use vectors and stacks for. I also will need to have switch statements for the rules and a lot of loops for the game and continuing the game until someone wins!

Kathleen Levi

Dr. Tucker

CS- 172

23 June 2015

Updated Requirements Specification

*Did you change the design along the way? Did you add/ remove features?*

I definitely changed my design along the way as I have been making changes and adding classes and adding/ removing other features. I realized as I was creating my program that I had a lot of functions, some that could be implemented into a header file like for the welcome section that defines the rules so now in the main function its all a short few words rather than a whole bunch of functions.

I also switched the cards vector form a private to public so that I could clear it later in the code, which mad some of the code easier as well. I also added the Welcome class that has all of the functions for the rules involved. Lastly I added the Stack class so that I could use stacks in the code as the draw pile by just popping the first element off of the stack in the shuffled deck.

I also wasn’t thinking about the functions I would use in my code, it was silly of me to think that I would put all of my code in main, that would have been a lot of code. I put a lot of the code in functions as well and using loops was extremely helpful as well so that I wouldn’t have dead ends in my code. I also got rid of the high scores text file and just had one being saved each time someone starts a new game.

|  |
| --- |
| **Player** |
| * name string * score int |
| * cards vector<int> * Player (string name, int score) * setName (string name) void * getName() string * getScore() int * changeScore (int NewScore) void * addCard (int cards) void * subCard (int cards) void’ * getCards() vector<int> |

|  |
| --- |
| **Welcome** |
| * Intro() void * ObjectOfTheGame() void * HowToPlay() void * FunctionsOfActionCards() void * GoingOut() void * Scoring() void * WinningTheGame() void |

|  |
| --- |
| **Stack** |
| * size int * elements[200] T |
| * Stack() * empty () const bool * peek() const T * push (T value) void * pop() T * getSize() const int |