Requirements Specification

A couple different classes are required for this program to work. You need a class for the cards to know how many cards are in the deck and the colors of the cards and the special cards if there is any. A class for the deck is also necessary so it can shuffle the cards so that the same order of the cards will not reappear each new game. And so that the cards in hand are not always the same for the player as well. There is also a class for the player so that they are given the chance to decide on what to do for when playing the game. There will be a class called Score that will keep track and state how to get a score which will be by winning the game.

The ShuffleDeck function is supposed to shuffle the total number of cards that are in the deck. This will always allow a brand new deck and hand for the new game. Special Cards will be defined as the draw 2, draw 4, reverse, skip and change the color and all of these will have different action and will run differently. In the player class, each player will have a hand which they will start off with 7 cards and will decrease or increase as the game goes based on what happens. The Score will be calculated by how many times the player has won in the game UNO and it will keep track of the score in a text file that will be created and reused as the games starts anew. Since the game is called UNO, the point of the game is that whenever a player reaches to one card they say UNO so that they are saved and can possibly win depending on their luck, if they do not say UNO then the player gets the penalty of drawing 2 more cards.

UML Diagrams

|  |
| --- |
| Deck |
| +Deck();  +Void ShuffleDeck();  +Void discardPile(); |

|  |
| --- |
| Cards |
| -String CardNames;  -String Color;  -Int CardNumber;  -String SpecialCard;  -Int TotalNumberOfCards; |
| +Cards();  +Int getTotalNumberOfCards();  +Int setTotalNumberOfCards();  +String getColor();  +String setColor();  +String getSpecialCards();  +String setSpecialCards(); |

|  |
| --- |
| Player |
| -String username;  -Int PlayersHand; |
| +Player();  +String getUsername();  +Int getHand(); |

|  |
| --- |
| Score |
| +Score();  +Void RulesForScore();  +Void TrackScore(); |