Chris Foremny

IT3500

**Capstone Project**

**Blackjack**

**Requirements**

Functional:

1. I want to play the game of blackjack.
2. I want to have one or two players.
3. I want to have a dealer.
4. I want to have 4 decks of cards in the “boot.”
5. I want to shuffle the cards and resume the game when there are 52 cards left in the “boot.”
6. I want each player and the dealer to get one card hidden then get one card showing.
7. I want to be able to “hit” (receive another card from boot) until I bust (over 21) or stay (less than or equal to 21).
8. I want the dealer to “hit” if their total is less than 17.
9. I want the dealer to “stay” if their total is greater than or equal to 17.
10. I want the player with the highest card value less than or equal to 21 to be the winner.
11. I want to have a tie if the player and dealer have the same value (no winner).

Non-functional:

1. Validate if an Ace is going to be a 1 or an 11.
2. If the hand has an Ace and score is greater than 21, minus ten and reevaluate if the hand is greater than or equal to 21. If less, then give player option to “hit” or “stay”. If greater than 21 “bust.”

**Features**

1. Create a class for the array of cards. (four decks)
2. Create a method to keep count of the cards left in the boot. (208 total cards)
3. Create a shuffle method to shuffle at beginning of game and to shuffle when there are 52 cards remaining in the “boot.”
4. Create a class with methods for displaying messages for players and the dealer.
5. Create a method for displaying the scores for each player and the dealer.
6. Create a method for displaying player 1, player 2, and dealer names.
7. Create a method for displaying the winner, loser, or tie.
8. Create a method to “hit” to receive another card.
9. Create a method for player to “stay” and not receive another card.
10. Create a method to add the total of the card’s values foreach player and the dealer.
11. Create a method to decide the winner, loser, or tie.
12. Create a method to control the “boot” so it can determine when to reshuffle.
13. Create a method to compare the scores of the players and dealer.