Nicole Schneider

BlackJack Design:

The UML diagram below is an outline of the classes I have for the Java part of the project. The MainActivities class controls the game based on the user input from activity_main. The Game class represents a game of blackjack, and has 2 Players: one is a Dealer and one is the User. Each of those inherit from the abstract Player class, since they share attributes and methods. Each Player has a Hand which consists of an array of Cards, with some limits as specified by the requirements (2-5 cards, can't go over 21, etc.). Cards are objects that have a Suit (diamonds, clubs, spades, hearts) which is an enumerated type, and an integer value (numerical value in context of blackjack- king = 10, etc.). The deck is a Stack of 52 Cards and has the ability to shuffle itself when the shuffle() method is called by the Game. The Ace card is a special case where if the Hand value is over 21, the Ace card's value becomes 1.

Methods

Card:
+ Card(int value, String name, Suit suit)
+ getValue(): int
+ toString(): String
+ getName(): String
+ setValue(int newValue): Void
  * used for Ace being 1 or 11

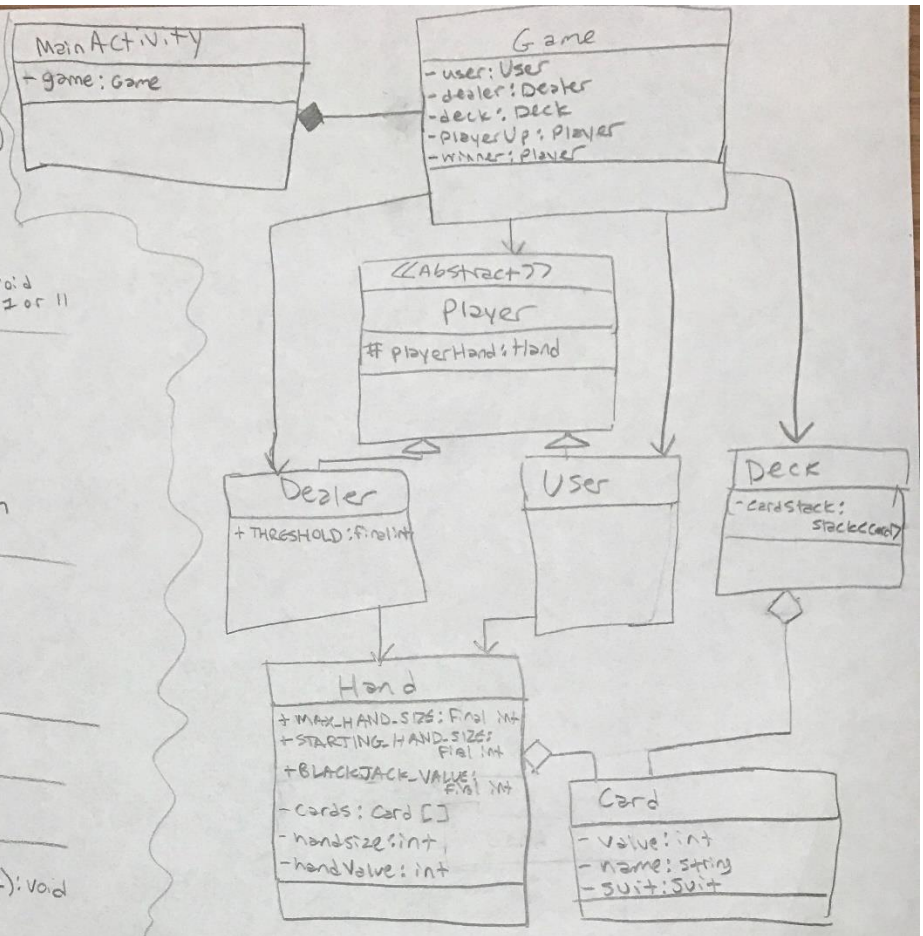
Hand:
+ Hand(Card c0, Card c1)
+ updateValue(): void
+ getValue(): int
+ getCards(): Card []
+ getHandSize(): int
+ addCard(Card c): boolean
+ toString(): String

Deck:
+ Deck()
+ toString(): String
+ shuffle(): void
+ drawTopCard(): Card

User:
+ toString(): String

Dealer:
+ toString(): String

Player:
+ createHand(Card c1, Card c2): Void
+ isBust(): boolean
+ hit(Deck d): boolean
+ hasBlackJack(): boolean
+ getHand(): Hand



Game:
+ Game()
+ switchPlayers(): void
+ executeHitTurn(): boolean
+ executeStandTurn(): boolean
+ getWinner(): player
+ pickWinner(): void
+ getUserHand(): Hand
+ getDealerHand(): Hand
+ dealerMove(): void

MainActivity:
onCreate(Bundle savedInstanceState): void
+ updateUserHandViews(): void
+ updateDealerHandViews(): void
+ displayWinner(): void
+ clearViews(): void
+ disableButtons(): void
+ userHit(View v): void
+ userStand(View v): void
+ newGame(View v): void