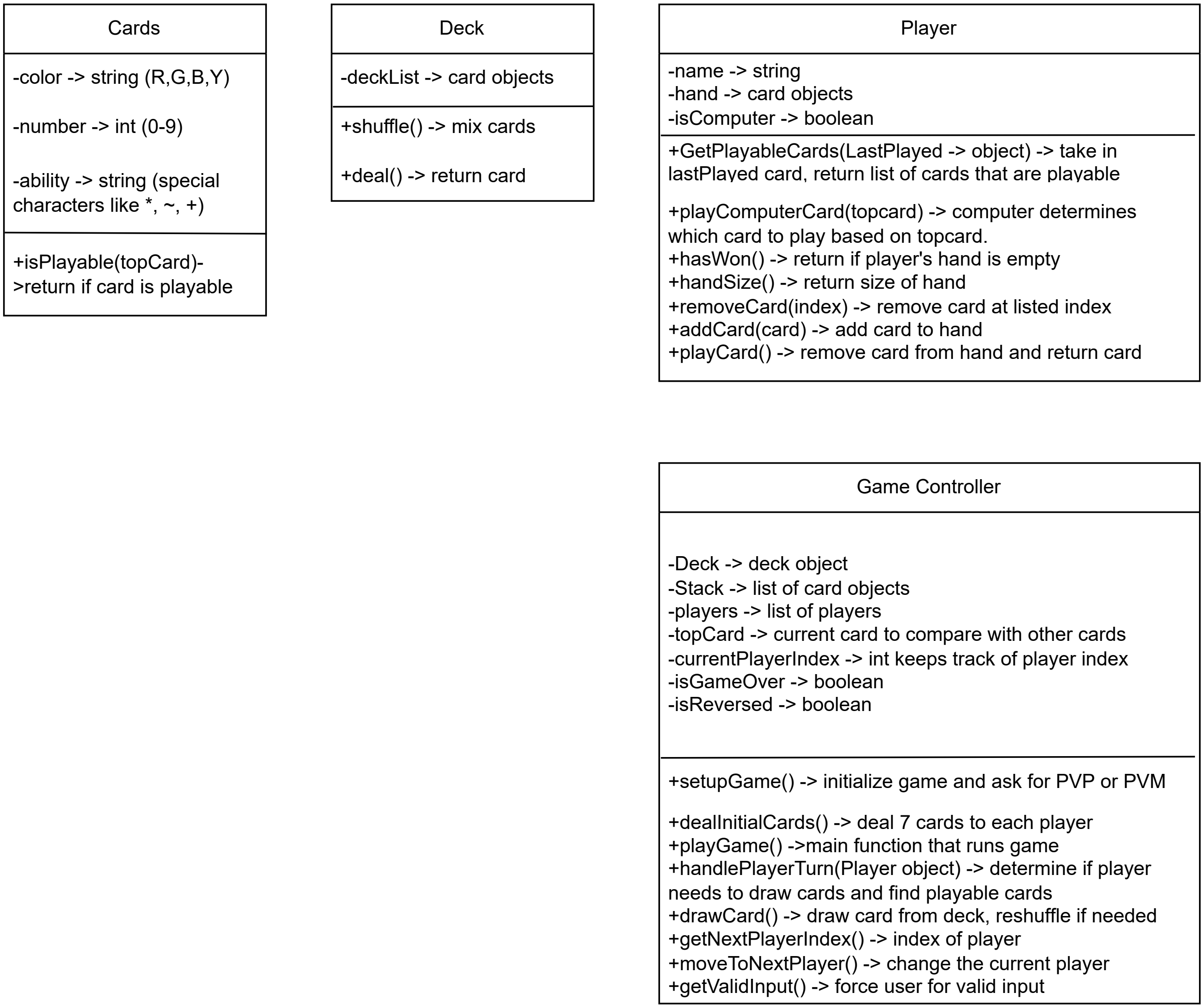

Cards

-color -> string (R,G,B,Y)

-number -> int (0-9)

-ability -> string (special characters like *, ~, +)

+isPlayable(topCard)->return if card is playable

Deck

-deckList -> card objects

+shuffle() -> mix cards

+deal() -> return card

Player

-name -> string
-hand -> card objects
-isComputer -> boolean

+GetPlayableCards(LastPlayed -> object) -> take in lastPlayed card, return list of cards that are playable

+playComputerCard(topcard) -> computer determines which card to play based on topcard.
+hasWon() -> return if player's hand is empty
+handSize() -> return size of hand
+removeCard(index) -> remove card at listed index
+addCard(card) -> add card to hand
+playCard() -> remove card from hand and return card

Game Controller

-Deck -> deck object
-Stack -> list of card objects
-players -> list of players
-topCard -> current card to compare with other cards
-currentPlayerIndex -> int keeps track of player index
-isGameOver -> boolean
-isReversed -> boolean

+setupGame() -> initialize game and ask for PVP or PVM

+dealInitialCards() -> deal 7 cards to each player
+playGame() ->main function that runs game
+handlePlayerTurn(Player object) -> determine if player needs to draw cards and find playable cards
+drawCard() -> draw card from deck, reshuffle if needed
+getNextPlayerIndex() -> index of player
+moveToNextPlayer() -> change the current player
+getValidInput() -> force user for valid input