Blackjack Class Diagram

ClientHandler

+ clientSocket : Socket

+ ClientHandler(socket : Socket)
+ run() : void
- getLoggedInPlayer(username : String) : Player
- getLoggedInDealer(username : String) : Dealer
+ handleMessage(Message incomingMessage): void
+ handleClearHands(Message incomingMessage): void
+ handleHit(Message incomingMessage): void
+ handleRequestHit(Message incomingMessage): void
+ handleLogin(Message incomingMessage): void
+ handleLogout(Message incomingMessage): void
+ handleJoinTable(Message incomingMessage): void
+ handleLeaveTable(Message incomingMessage): void
+ handleCreateTable(Message incomingMessage): void
+ handleCloseTable(Message incomingMessage): void

Server

- loggedInPlayers : List<Player>
- loggedInDealers : List<Dealer>
- tables : List<Table>

+ main(args : String[]) : void

Client

- Socket: socket
- inStream: ObjectinputStream
- outStram: ObjectOutputStream
- loggedIn : Boolean
- Player: player
- Game: game

+ connectToServer(ipAddress : String, port : int) : Boolean
+ sendLoginMessage(String username, String password) : void
+ sendLogOutMessage(String username, String password): void
+ sendRequestHitMessage(String username, int tableID): void
+ sendHitMessage(String playerUsername, Card card, int tableID): void
+ sendStayMessage(String username): void
+ sendJoinTableMessage(String username, int tableID): void
+ sendLeaveTableMessage(String username, int tableID): void
+ sendCreateTableMessage(String username): void
+ sendCloseTableMessage(String username, int tableID): void
+ sendClearHandMessage(String username, int tableID): void
+ sendDepositMessage(String username, int balance): void
+ sendWithdrawalMessage(String username, int balance): void
+ getTableCountMessage() : void
+ disconnect() : void
+ startListening() : void
+ handleServerMessage(Message msg) : void

Message

- serialVersionUID : long
- idCount : int
- id : int
- type : MessageType
- text : String
- from : String
- username : String
- password : String
- date : Date
- card : Card
- balance : Int
- tableID : int

+ Message(type : MessageType, username : String, password : String, text : String, from : String, card : Card)
+ getID : int
+ getUsername() : String
+ getPassword() : String
+ getType() : MessageType
+ getText() : String
+ getDate() : Date
+ getFrom() : String
+ getCard() : Card
+ getBalance(): Balance
+ getTableID(): tableID

<<enumeration>> Suit

Club
Spades
Diamond
Heart

<<enumeration>> Type

HIT
REQUEST HIT
STAY
JOIN_TABLE
LEAVE_TABLE
CREATE_TABLE
CLOSE_TABLE
LOGIN
LOGOUT
WITHDRAWAL
DEPOSIT
ERROR
WIN
LOSE
BUSTED
TABLE_COUNT
TABLE_FULL
NO_DEALER
RECEIVE_HIT
CLEAR_HANDS

<<enumeration>> Value

1
2
3
4
5
6
7
8
9
10
11

Lobby

- tables: ArrayList<tables>
- maxTables: int

+ addTable(): void
+ removeTable(): void
+ getAvaliableTable(): ArrayList<Tables>
+ isTableAvaliable(): bool

Game

- card : Card
- players : List<Player>
- dealer : Dealer
- shoe : Shoe
- currentPlayer : int
- timer : Timer
- turnTimeoutTask : TimerTask
- maxPlayers : int

- startTurnTimer() : void
+ nextPlayerTurn() : void
+ dealerTurn() : void
+ roundEnd() : void
+ hit() : void
+ stay() : void

Shoe

- decks : ArrayList<Deck>
- numDecks : int
- numCardsInShoe : int

+ Shoe(numDecks : int)
+ getNumDecks() : int
+ getCard() : Card
+ reset() : void

Dealer

- id : int
- username : String
- password : String
- hand : Hand
- handValue : int

+ Dealer(username : String, password : String)
+ getDealer() : Dealer
+ getID() : int
+ getUsername() : String
+ getPassword() : String
+ getHand() : hand
+ getHandValue() : int
+ addCardToHand(newCard : Card) : void
+ getHandValue(): int
+ startGame(): void
+ hitPlayer(): void
+ dealStartingCards(): void
+ clearDealerHand(): void

Player

- id : int
- idCount : int
- username : String
- password : String
- balance : int
- hand : Hand
- currBet : int
- stayed : boolean

+ Player(username : String, password : String, balance : int)
+ getID() : int
+ getUsername() : String
+ getPassword() : String
+ getBalance() : String
+ getHand() : Hand
+ getHandValue() : int
+ getCurrBet() : int
+ setBalance(amount : int) : void
+ setHand(card1 : Card, card2 : Card) : void
+ setCurrBet(amount : int)
+ withdrawalMoney(amount : int) : void
+ depositMoney(amount : int) : void
+ addCardToHand(newCard : Card) : void
+ hasStayed() : boolean
+ stay() : void
+ clearPlayerHand() : void

Table

- players : ArrayList<Player>
- dealer : Dealer
- maxPlayers : int
- maxDealers : int
- game : Game

+ Table(dealer : Dealer)
+ getPlayers() : ArrayList<Player>
+ getNumPlayers : int
+ getDealer() : Dealer
+ getMaxPlayers() : int
+ getMaxDealers() : int
+ getGame() : Game
+ isEmpty() : Boolean

Deck

cards : List<Card>

+ Deck()
+ getNumCards() : int
+ isEmpty() : boolean
# fillDeack() : void

Card

- value : int
- suit: String

+ Card()
+ getValue() : int
+ getSuit() : Suit
+ getSymbol() : Symbol

Hand

- value : int
- cards : List<Card>
- handTotal : int
- Busted : boolean
- hasBlackJack : boolean
- aceasEleven : int

+ Hand()
+ Hand(firstCard : Card, secondCard : Card)
+ getHandTotal() : int
+ getCards() : List<Card>
+ addCard(newCard : Card) : void
+ isBusted() : Boolean
+ isBlackJack() : boolean
+ clearHand() : void