Texas Hold’Em Poker – Part 2

**By:**

* Amit Dahan (304824170) – [amitdahan11@gmail.com](mailto:amitdahan11@gmail.com)
* Margarita Bazak (318110376) – [rita.bazak@gmail.com](mailto:rita.bazak@gmail.com)

**Remarks:**

* Max bet amount and max raise amount are both defined as the following:

The minimum between:

The minimum value of current chips + current bet of all players

and

Pot value + sum of all current bets

minus the current bet on the table.

This means that this calculation will work for both max bet and max raise because we’re taking into consideration the current bet on the table, which will be 0 if no bets were made (in the case of the max bet).

* Blinds increase once N ( = number of players) hands were played. Regardless of whether there’s less players (for example if a player lost all of his money and is no longer in the game).
* If the current max raise or bet equal 1, then the appropriate button will immediately bet/raise 1 without confirmation.
* If all computer players are either folded or non-existent, then when all humans fold, existing pot disappears.
* During replay mode, the table to the right of the screen does not get updated accordingly – it is related to the state of the game, not the hand.
* In order for a player to see his cards during his turn, he must hover over his cards with his mouse.
* If all human players retire / only 1 player remains – the game automatically ends without prompt and the game flow starts from the beginning.
* Only showing one winner at the end of the game, even if there are multiple players with the same amount of chips.

**Project Hierarchy:**

javafx-client module:

* JavaFXPokerApplication
* RootController (root.fxml)
* GameInfoPaneController (gameInfoPane.fxml)
* Menus:
  + GameMenuController (gameMenu.fxml)
  + HandMenuController (handMenu.fxml)
  + BetweenHandsMenuController (betweenHandsMenu.fxml)
  + ReplayMenuController (replayMenu.fxml)
* GameBoardController (gameBoard.fxml)
* PlayerHandController (playerHand.fxml)
* CardController (card.fxml)

poker-engine module:

* exceptions package
  + BadFileExtensionException
  + DuplicatePlayerIdException
  + IllegalMoveException – not in use yet
  + InvalidBlindsException
  + InvalidHandsCountException
* immutables package – classes that are OK for use in the client because they don’t expose any “actions”, only a window into API values
  + Card
  + PlayerInfo
    - PlayerGameInfo – info sent to the client about players’ game status
    - PlayerHandInfo – info sent to the client about players’ hand status
* internals package
  + Game
    - subclass BasicGame
    - subclass MultiplayerGame
    - future subclasses
  + Deck
  + GameConfig – where the XML values gets parsed into
  + Player
    - subclass GamePlayer
    - subclass HandPlayer
  + Hand
* xml\_game\_config package – JAXB generated classes
* PokerEngine class – is the main API the client uses to do everything in the game