Texas Hold’Em - Programmer’s Manual

By Team ETCetera

Table Of Contents

Introduction

Scope of the Product

Our project will be to implement and document a standalone Texas Hold'Em Poker game. The software will implement the game for two players: one being the user, the other being the computer. Standard rules and hand evaluation for the game will be strictly followed. The twist to our game will be that relevant statistics and predictions will be shown during play. These will take into account the cards the user and computer have in their hands as well as the cards that are on the table, assuming a standard 52-card deck. These calculations will help the user determine the best plays to make next.

The interface will accommodate a single user with a computer competitor, and will likely be a simple command-line interface at first. As time allows, we hope to expand this to a graphical user interface (GUI) to make it easy and more intuitive to use. The computer competitor will initially be relatively simple, foregoing machine learning algorithms for simple odds-based responses. This will allow us to focus more on the core implementation of the game. However, if time allows, we would like to expand the computer competitor to be based on relevant machine learning algorithms in order to provide a more interesting challenge (both for the players and the creators).

This project will result in a standalone application without an Internet requirement and will not interact with any any other software systems. The game will work on the Windows platform at minimum, specifically Windows versions 7 through 10. The program will be written entirely in Java. If we are able to implement a GUI, we are considering the use of JavaFX, which is incorporated into Java versions 1.7 and higher. We may consider the use of other available software packages if they fit into our design and help reduce the overall workload or improve functionality. However, it is still our goal to have our only dependency be a minimum version of Java.

Definitions

IDE – Integrated Development Environment
JRE – Java Runtime Environment
JDK – Java Development Kit
CLI – Command Line Interface
GUI - Graphical User Interface

References

"Fisher-Yates Shuffle." *Programming-Algorithms.net*. N.p., n.d. Web. March & april 2017. <http://www.programming-algorithms.net/article/43676/Fisher-Yates-shuffle>.

Jake Sellers, Graphic Designer

Richard D. P. (1998). *Dealing with Imperfect Information in Poker.* Retrieved from http://poker.cs.ualberta.ca/publications/papp.msc.pdf

"Playing Card Image Assets (SVG + PNG)" (2014). *Github.com.* Web. April 2017. Retreived from https://github.com/hayeah/playing-cards-assets

PokerListings.com. "Official Texas Holdem Rules | How to Play Texas Hold'em." *Pokerlistings*. N.p., 27 Nov. 2016. Web. 02 May 2017. Retrieved from http://www.pokerlistings.com/poker-rules-texas-holdem

General Description

Product Perspective

Texas Hold'Em Poker is the most popular forms of poker to this day, and the game is complicated and nearly impossible to perfect. Professional poker players spend decades honing their techniques and strategies for winning the game. Beginner and even average poker players can easily be overwhelmed with the rules and overall goal of the game that they can not begin to wrap their head around possible strategies to increase their odds of winning the game. This application will help players better understand the mathematics of the game, with the goal of making them more aware, intelligent Texas Hold'Em Poker players.

Product Functions

At the highest level, the application will be a standalone Texas Hold'em Poker game with one human player who is playing against the computer. The game includes a simple, easy to use GUI with images of playing cards, buttons for each game option, a print out description of the games' play-by-play, and popups for the user's input as needed.

To satisfy our goal of helping poker players become more aware of the mathematical side of the game, relevant statistics and predictions will be shown during play. These will take into account the cards the user and computer have in their hands as well as the cards that are on the table, assuming a standard 52-card deck. These calculations will help the nn user determine the best plays to make next.

User Characteristics

The users of the program will be beginner or moderate Texas Hold'Em poker players who would like to better understand the mathematics and statistics that are involved in the game. These expected user will be interested in playing one-on-one games against a simple computer opponent.

General Constraints

- PC with Windows Operating System, versions 7 though 10.
- Java version 1.7 or newer installed

Assumptions and Dependencies

The game will work on the Windows platform, specifically Windows versions 7 through 10. Our only dependency is Java version 1.7 or newer.

Specific Requirements

There following represents the specific requirements of this software, organized by the version in which in which the requirement will be met. A simple three-digit format will be used, with the first digit representing major functionality, the second will represent intermediate functionality, and the third will represent more detailed, specific requirements.

1.0.0 The software should provide an initial, playable version of Heads Up, No Limits, Texas Hold 'Em
- 1.1.0 The software should provide the basic features of Hands Up, No Limits, Texas Hold 'Em
  - 1.1.1 The software should allow the user to play multiple hands in sequence until only one player has any remaining chips
  - 1.1.2 Each hand should be played according to standard Texas Hold 'Em rules
  - 1.1.3 Upon showdown, or when only one player remains in the hand, the contents of the pot should be distributed according to standard Texas Hold 'Em rules
  - 1.1.4 The software should allow the user to play multiple games in sequence

1.2.0 User should be able to perform all betting actions according to the game's rules
1.2.1 User should be able to Fold
1.2.2 User should be able to Raise when appropriate
1.2.3 User should be able to Call when appropriate
1.2.4 User should be able to Check when appropriate
1.2.5 User should be able to Bet when appropriate
1.2.6 User should be able to go All In when appropriate

1.3.0 A rudimentary computer opponent should be provided
1.3.1 Provide betting strategy implementation for pre-flop hands
1.3.2 Provide betting strategy implementation for post-flop hands
1.3.3 Provide interface through which bot can communicate its moves and get information about the current game state

1.4.0 User should be able to interact with the game through a simple CLI
1.4.1 Allow user to quit or reset the game at any point
1.4.2 Allow user to perform all available moves through the CLI
1.4.3 Provide all necessary information about the game to the user through the CLI Including the board cards, pot size.

1.5.0 Maintain a persistent game state
1.5.1 Game state should hold the pot, small blind and big blind amounts, and the current board cards, along with any other relevant data not allocated to another class.
1.5.2 Game state should remain consistent throughout play

1.6.0 Provide a simple Deck class to maintain a single deck of 52 cards
1.6.1 The Deck class should allow Cards to be drawn one at a time
1.6.2 The Deck should be capable of shuffling itself internally
1.6.3 The Deck should be capable of being reset by restoring any drawn cards to the deck and shuffling it.
1.6.4 The Deck class should provide an implementation of the Fisher-Yates shuffling algorithm

1.7.0 Provide a simple Player class for storing relevant information for a particular player
1.7.1 The Player class should store and provide access to a String containing the name of the player
1.7.2 The Player class should maintain the number of chips available to the player and the number of chips the player has committed to the pot
1.7.3 The Player class should provide the means to modify the number of chips available to the player
1.7.4 The Player class should provide the means to modify the number of chips the player has committed to the pot
1.7.5 The Player class should provide the means to access the current number of chips the player has committed to the pot and the number of chips currently available to the player
1.7.6 The Player class should store two hole cards for the player
1.7.7 The Player class should provide the means to set and retrieve the hole cards

1.8.0 Provide a system for ranking pre and post-flop hands

1.8.1 Provide hand ranker for post-flop hands
1.8.2 Provide hand ranker for pre-flop hands
1.8.3 Provide a system to convert a hand ranking to a meaningful string so the player can see what their best hand is

1.9.0 Provide a simple Card class for holding information about a particular card
1.9.1 The Card class should be capable of representing all 52 possible cards
1.9.2 The Card class should provide a numeric system to represent the cards
1.9.3 The Card class should provide String identifiers for all possible cards

2.0.0 The software should provide a more refined version of Heads Up, No Limits Texas Hold 'Em
2.1.0 Provide user with additional control in setting certain values
2.1.1 User should be able to set the amount of starting chips for players
2.1.2 User should be able to set the number of rounds per blind increase
2.2.0 Bot should make smarter betting decisions based on hand potential
2.2.1 Provide hand potential calculator
2.2.2 Bot strategy incorporates hand potential into its betting strategy
2.3.0 Improved game realism
2.3.1 Increase blinds based on the number of hands played
2.4.0 Provide user with some statistics based on their hand
2.4.1 Provide user with pre-flop estimated win rate
2.4.2 Provide user with post-flop hand strength
2.4.3 Provide user with post-flop effective hand strength

3.0.0 The software should provide a GUI through which the user can play
3.1.0 The GUI should allow the user to select appropriate options at each stage in play
3.1.1 The GUI should allow the user to quit at any point
3.1.2 The GUI should allow the user to perform all the betting actions outlined in requirement 1.1.0
3.2.0 A simple text area should be provided to show the user key information
3.2.1 The text area should be able to display user hand stats, bot actions, and any other key information
3.2.2 The text area should be capable of being reset to a blank state
3.3.0 The GUI should provide current information about the game state
3.3.1 The GUI should show any visible cards, including hole cards for any player whose cards are visible, and any visible board cards
3.3.2 The GUI should show the user the current state of the pot
3.3.3 The GUI should show the user the number of chips required to call
3.3.4 The GUI should show the user the number of chips available to each player along with the number of chips each player has committed to the pot
3.3.5 The GUI should indicate when it is the user's turn

4.0.0 Adds additional features and refines the GUI and bot behavior
4.1.0 Allow user to save and resume their current game
4.1.1 Allow user to save their game as a file on their local computer
4.1.2 Allow user to resume their saved game by loading game info from local file

4.2.0 Allow user to reset their game at any time

4.3.0 Present only the relevant betting action buttons at any given time

4.4.0 Improve bot's betting strategy to incorporate other player's actions

5.0.0 Provide support for a single human user to play against multiple computer opponents

5.1.0 Allow one user to play against 1-7 bots (at minimum)