

User Manual

For our final project, our group had the goal of fully designing and implementing the classic poker game Texas Holdem. We wanted to create a simulation of the game where a user can be immersed in the poker experience, playing against a customizable number of CPU players with a CPU dealer regulating the game. Although poker is an incredibly popular game, it does usually involve betting real money on games, which isn't necessarily something that everyone is willing or even able to do. Instead, we wanted to create a simulation of the experience where you can feel free to bet as much fake money as you like, completely free of financial consequences.

One of our primary focuses within this project was to create a clean, user friendly graphical user interface. Familiarity and aptitude for GUI was a common strength among several of our team members, so we decided for this to be an area within our project where we could really expand and make something not only unique but also particularly user friendly. One of the ways we wanted to achieve this was through customization. Often times a great user experience stems from the user's ability to customize that experience. It is naturally difficult to please everyone, as most people have individualized preferences for their respective experiences or interfaces that they choose to use. Therefore we went out of our way to find aspects of our GUI that were feasibly customizable, and we collectively feel that the result is a successfully accomplished these goals. The GUI features five customizable options, some of which are merely aesthetic shifts while others have a tangible impact on the gameplay. These

options include the color of the user's player, the color of the poker table itself, the number of CPU opponents that the user will be playing against, as well as the options to show CPU opponent chips as well as their respective hands after each round.

A lot of the background and motivation for this program and the problem we tackled comes from the popularity of poker itself. Poker is easily one of the most well known, recognizable, and widely played card games in existence. So much so, in fact, that many online adaptations of the game already exist on the internet. Although I was personally unfamiliar with the game's rules prior to starting this project, one of our group members has a distinct passion for the game and loves playing it, hence the initial suggestion when we were brainstorming ideas for this project. Upon further research, we came across many different online poker games all of which successfully simulate the poker experience on the computer. Many of these games had interesting ways in which the user could interact with the game, as well as intriguing and immersive GUIs. Being exposed to these computerized poker games, as well as the rules of the game for the first time(some of us) helped inspire and motivate us to take on this project and create our very own version of poker, this one of course in Java.

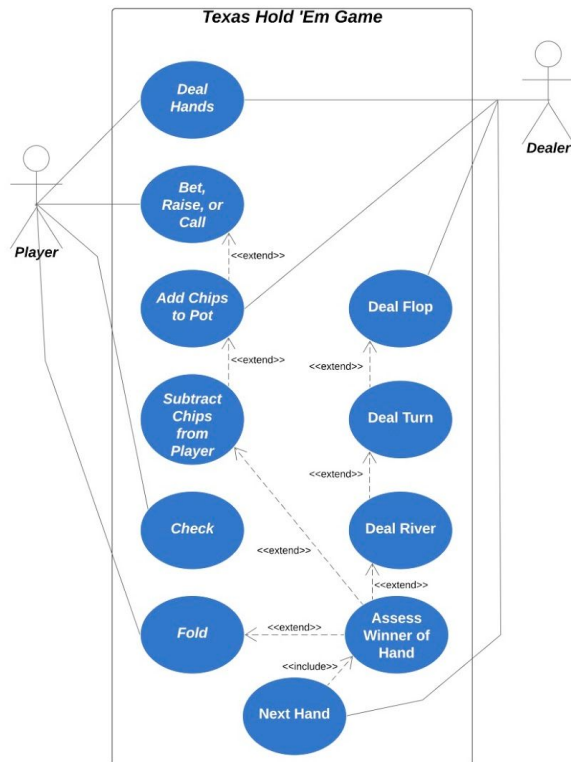
As far as specific mechanics we wanted our program to have, we made many user stories to accurately reflect the needed functions of your standard game of Texas Holdem poker. The two obvious types of people taking part in a poker game are the players and the dealer. In our case, we want the user to of course take part in the game as one of the players. The other players, as well as the dealer, are to be controlled by

the CPU. However as mentioned earlier, the user is able to customize exactly how many CPU players he or she wants partaking in the game.

The dealer alone has several responsibilities that needed to be implemented in order for the game to be properly managed by the CPU. As one might be able to assume, the dealer's primary responsibility is to deal cards throughout the game. In poker terminology, this includes dealing each player's initial two cards, the river, the flop, and the turn. The dealer also must be able to determine which player has actually won each round by comparing the players hands. The player/user, of course, has its own set of responsibilities that we knew had to be implemented off the bat. Players have the classic actions of being able to bet, check, and call, raise, or fold. This gave us a lot to manage as far as implementation just for the logic of the game itself. Overall, the dealer and players basically need to be able to follow the rules and appropriate steps that are generally taken in your generic game of Texas Holdem poker. As you can see from our use case diagram, there is a clear relationship between the roles of the dealer and player, regardless of whether it is a CPU player or the user. The series of moves by either entity accurately reflects your typical game structure of Texas Holdem.

Texas Hold 'Em Use Case Diagram

Shane Staret | December 10, 2019



The instructions for playing our game are quite simple and straightforward. Having the game be easy to play was one of our top priorities, to go hand in hand with our user friendly GUI. Upon launching the game, the user is greeted with a box with some brief text describing the project, authors, and what the game is. There are three button options the user can press, those being labeled play, settings, and exit. From the beginning, we wanted to make sure to not overwhelm the user and instead present them with a short and clean set of game options. Because of this we know to only present the user with what they needed and nothing extraneous. The exit button has a rather self explanatory function. Upon pressing the settings button, however, the user is

presented with a new set of customizable features. There are a total of six drop down menus which allow the user to customize the number of players in the game, the color of both the poker table and the player, the starting chip amount, as well as the option to show each opponent's chips and their hands after each round. The save button here returns the user to the main screen, where they can go ahead and play the game. The game then proceeds as one would predict with Texas Holdem, with the CPU dealer dealing cards to each of the players, starting bets being placed, and then moving from player to player as each is presented with the opportunity to call, raise, or fold each turn.

In summary, we collectively feel as though we did a good job of accomplishing our goals, those being to create an aesthetically pleasing and user friendly GUI to be paired with a customizable and easy to play poker experience, free of the confines of real life financial restriction.