Project 6

Milestone 5 – CSE 4904

William Sanville
**Final Product**

The finished product of the JokerStars system performs the basic functionality of an online card room. It allows multiple players to play Texas Hold'em over the Internet against one another. The system is broken down into two separate executable files, one for the server, and another for the end users. The server executable runs on a dedicated machine. Its job is to maintain connections from the users, broadcast messages to the users, and automate the flow of the card game. In a traditional game of Texas Hold'em played at a casino, the casino provides a dedicated dealer to shuffle and deal the cards, keep track of the bets made, evaluate the winning hands, award money to players, ensure each player plays by the rules, and more. In the JokerStars system, the server executable performs the duties of the dealer. The server executable allows system operators to create virtual card tables at run time for users to play on. For example, an administrator can create a Texas Hold'em table and specify the amount of money needed to play at the table, how many players are allowed to sit at the table, and what the blinds are for the table. Currently, the only card game implemented is Texas Hold'em. However, the system is designed to support future development of games, such as Omaha. System operators will have the ability to choose which card game the virtual card table plays if more games are developed in the future.

The end user's application requires the user to either create an account for the system, or log into an existing account. This is needed in order to keep track of how much “money” each user has (the system uses play money, not real money). Upon logging in to the system, the user is greeted and presented with the amount of money in

---

1 Blinds are forced bets at the beginning of a poker hand. These types of bets are common among games that do not require an ante at the beginning of a hand.
their account. The user is also presented a list of card tables that are running on the server. From this point, the user can choose which card tables to play on. In popular card playing systems, users can play at multiple card tables at the same time, a feature that is not available in a traditional card room. Users of the JokerStars system also have this feature available. Users can play at multiple card tables at the same time. In addition, users can also watch a card table instead of playing the game. Once joining a card table, the user will be presented with a graphical representation of all elements in a game of Texas Hold'em, such as the community cards, chips, players, and more.

The original product specifications were for a single card table to run at a time. The actual implementation of the system allows for multiple card tables to run at the same time on both the server and the end user's machine. Also, the original specifications did not include an interface for the end user to choose which table he or she wishes to play on. However, due to the ability for multiple tables to run, an interface for selecting the desired card table was added.

The specifications called for card tables with a maximum of nine players. However, that number was decreased to six in order to shift the focus from implementing the user interface to implementing more critical aspects, such as the game engine.

**Card Table Features**

Once a user has selected a card table to play at, the user must first select a seat that is not occupied. A player will only be able to sit down at a table if there are less than six users playing and if the user has the amount of money required to play at the table. Upon sitting down at a table, the user must wait until he or she can post the big
blind in order to begin playing a hand of poker. The big blind is a forced bet by the
player who is two seats to the left of the current dealer. The style of Texas Hold'em that
is implemented is a ring game, which is a game where players can enter and leave at any
time. A hand of poker will begin once two players are seated at a table and will continue
while there are at least two players at the table who have money. In the event of a player
losing all of his or her money, the player will be marked as “Sitting Out.” The card table
includes a chat box so that players and spectators can communicate with each other. The
chat box also serves as a way for the virtual dealer to make announcements about the
game. For example, the dealer announces which player wins the hand in the event of a
showdown².

The card table interface allows players to decide what actions they wish to perform when it is their turn to act. It also displays the other users playing at the table, along with poker chips to represent the money that is in the pot, as well as any money players have bet in the round of betting. When a showdown occurs, the card table displays the cards for all users involved in the showdown. In a traditional card game, the showdown process is slightly different. Instead of showing all hands, hands are shown in the same order as the players act in a round of betting. For example, if three players are involved in a showdown, the first player who would bet in a round of betting shows his hand. If the second player has a weaker hand than the first player, he has the option to “muck” or to discard his hand without showing it. Likewise with the third player, if he has a stronger hand than the previous hands shown, he shows his hand, otherwise, he can discard the hand. This behavior was not implemented in the JokerStars system. It was

---

² A showdown occurs in all poker games. If more than one player remains after the last betting round, remaining players expose and compare their hands to determine the winner or winners.
felt that this was not an essential element of a poker game and could be left out.

**System Requirements**

In order for an end user to run the JokerStars system, the user must have a computer that supports the .NET Framework 3.5 SP1 (the system will not run on any version less) and an Internet connection. For Windows users, this requires Windows XP SP2 or higher. For Mac or Linux users, third party software is available to run .NET applications. However, JokerStars was not tested on these operating systems and there are currently no plans to test support for non-Microsoft operating systems in the future.

The server executable has the same requirements as the client, with the exception that the server requires the .NET Framework 2.0 or higher. This requires Windows 2000 or higher. It is recommended that the server has a strong Internet connection in order to handle many users of the system concurrently.

**Limitations**

JokerStars was designed to deliver the bare minimum features for an online Texas Hold'em game, due to the short time frame of development (only 1 semester). Because of this, the system is not as secure or robust as a commercial system. One of the most important aspects of any online application is security, especially an application that handles money. Despite the fact that JokerStars does not use real money, security is still important to preserve the integrity of the game. Malicious users could ruin the game experience for legitimate users, resulting in a useless system. The JokerStars system does not encrypt any information traveling between the server and the user. This leaves the
potential for users to discover what cards their opponents are holding in their hand. In addition, JokerStars does not verify the integrity of data sent from users. This leaves a potential for security threats, such as man in the middle attacks.

The JokerStars networking components are not ideal, as they do not perform any redundancy checking on the data that is received. As a result, JokerStars performs poorly and is prone to failure when using unreliable Internet connections. An important future enhancement to the system would be to increase the robustness of the communication protocol by implementing some type of checksum on all data that is sent.

The server executable runs on a single physical machine. As a result, the number of users who can access the system at the same time is limited. Also, the uptime of the system is greatly limited due to the lack of redundancy and load balancing.

**Known Bugs**

- Players who are holding cards in their hand are not displayed properly to a user who has just joined a card table.
  - When the cards are dealt, each client receives a message that tells them to display card graphics.
  - If a player joins the card table after this message has been sent, he or she will view the game without a visual representation of the cards in other players' hands.
- The Game Engine does not properly tally the number of active players at a given game table.
  - When counting the number of users at the table who are eligible to play, the
Game Engine looks at each player's status. A new player status was added late in the development phase to represent a player who is waiting for the big blind. The Holdem Table was not updated to check for this added player status when counting the number of eligible players when starting a hand of poker.

- The system does not perform well when using an unreliable Internet connection.
  - Because the networking components do not perform any redundancy checking on data, the application is prone to runtime errors when using an unreliable connection. This will force the end user to close the application and in some cases may require the server to restart the game.
  - The Game Engine is event driven, it does not continuously loop. When the engine receives a piece of data from a client signifying that an event has occurred (such as a player making a bet or leaving the table), the engine responds to the event and then waits for a new event. If the engine is waiting on Player 1 to make a bet, and Player 1’s application has crashed, the engine will keep waiting forever.
  - A simple fix would be to enforce a time limit on all user actions. Many commercial systems allow players a maximum of 30 seconds to decide on their action. This would improve the quality of the game for the other players and take care of the event of a user's application crashing or losing connection. After the 30 seconds has passed, the Game Engine would declare the user's hand as folded.