

Table Of Contents

1. Short Forms	3
2. Conventions	3
3. Introduction	4
3.1 Objective	4
3.2 Content	4
3.3 RRRT Model Description	4
3.4 Relationship between RRRT and Doc	4
4. Description of the System	5
4.1 Scope of the Project	6
5. Terms	7
6. Requirements	10
6.1 Functional Requirements	10
6.2 Non-Functional Requirements	12
7. Assumptions	13
8. Use Case Diagram	14
9. Use Cases	15
10 UC-# (Table Description)	15
10.1 UC-01 (System Start-up)	16
10.2 UC-02 (Joining the Table)	17
10.3 UC-03 (Hand Sequence)	19
10.4 UC-04 (Round of Betting)	21
10.5 UC-05 (Adding a Player to the Table from the Queue)	22
10.6 UC-06 (Player Removal)	23
10.7 UC-07 (Account Transactions)	24
10.8 UC-08 (System Shutdown)	25
10.9 UC-09 (Instant Messaging)	27
10.10 UC-10 (Seated Player Inactivity)	28
10.11 UC-11 (Administrator Changes System Mode)	29
10. Design Decisions	30
11. References	38

1. Short Forms

The following is a list of all the short forms used throughout the white-paper documentation and documentation in the Rose RT models.

<u>Short Forms</u>	<u>Full Forms</u>
CBS	Central Banking System
CS	Central Server
HHD	Hand-Held Device
UPS	Uninterruptible Power Supply
SB	Small Blind
SBB	Small Blind Bet
BB	Big Blind
BBB	Big Blind Bet
FR	Functional Requirements
NR	Non-Functional Requirements
DD	Design Decision
A	Assumptions
DG	Discussion Group
CaS	Case Study

2. Conventions

Every requirement has a unique ID, that will be used in the use cases for traceability. The numbering scheme for the IDs is as follows:

- They begin with a prefix 'FR' (functional requirements) or 'NF' (non-functional requirements) followed by a separator '-' and a unique number.
- Design decisions begin with a prefix 'DD' followed by a separator '-' and a unique number.
- Assumptions begin with a prefix 'A' followed by a separator '-' and a unique number.
- Functional requirements may come from either the 'DG' (discussion group) or 'CaS' (Case Study).
- Requirements from the DG are traced with the date and time of their posting.
 - E.g.. (DG(user)-year-month-day, time)
- Requirements from the CaS will be traced by their section number, paragraph number, line number (paragraph numbers are relative to the section).
 - E.g.. (Cas-section#-paragraph#-line#)

3. Introduction

Poker games are becoming more popular now, then ever before. Thanks to the publicity and media coverage of poker tournaments like the *World Poker Tour*, more and more people are becoming interested in being part of this fad. Casino owners are also faced with the task of keeping up with the ever growing demand for poker tables.

3.1 Objective

Juan Pablo Enterprises, a charity casino, has approached our team to help resolve the problem. They hope to convert their table based Texas Hold'em Poker, to an electronic version. The electronic version will be very closely based on the rules of the table based game. It will support seven (7) clients/players. They will each have a dedicated hand-held device(HHD) with which they will interact with the central server.

3.2 Content

This report begins by briefly describing the system and listing any assumptions that were made. Subsequent sections describe the system's functional/non-functional requirements, user case diagrams, use cases, and design decisions. Each sections contains traceability information that will enable the reader to trace all information back to the original case study (version 1.0) that was presented by the user or the discussion group.

3.3 RRRT Model Description

The Rational Rose Real Time Model contains: a Use Case Diagram, sequence diagrams corresponing to each use case main sequence and alternative, capsules and protocols. The use cases, proposed later in this document, were used to develop the Use Case Diagram and sequence diagrams. The Use Case Diagram established the relationship between each use case and the actors that interact with various components of the system. Sequence Diagrams represent signals and states derived from the flow of events as described in the Use Cases. The system model consists of capsules, ports and protocols that were a direct product of the states and messages defined in the sequence diagrams.

3.4 Relationship between RRRT and Doc

This report is supplementary to the documentation found within the Rational Rose RealTime Model of the system. For more detailed information regarding the sequence diagrams, capsule structure, state machines, communication pathways (protocols) and system architecture, please refer to the Rational Rose RealTime Model provided on the CD. This document gives in-depth explainiation of system requirements and functionality.

4. Description Of The System

When a player wishes to play Texas Hold'em Poker, the player selects an empty handheld device. If there are no handheld devices available the player is placed into a first come first served queue. Once a handheld device becomes free the player at the front of the queue is assigned the empty handheld device. Once a player has been assigned a handheld device, the player enters his/her name and banking information into the device. Upon successful validation of the player's banking information the player becomes part of the game. If there is an active round in progress the player can watch the active round, but cannot play until that round has completed. Players may leave the game at any time. If a player does not have any money at the beginning of a round, they are removed from the game so that players waiting in the queue are allowed a chance to play.

A new hand starts when the small blind bet (\$5) is placed followed by the big blind bet (\$10). After this, every player at the table is dealt out two (2) cards. They must decide whether to fold, match the big blind bet, or raise the bet. After everyone has 'called' the highest raised amount, the 'Flop' occurs. i.e. The first three (3) community cards are turned over. Another round of betting ensues starting with the player immediate to the left of the dealer. Once the highest raised amount has been matched by all players still playing, the 'Turn' card is turned over. Once again another round of betting occurs, and finally the 'River' card is turned over. A final round of betting occurs. Once all bets have been matched, the person with the highest ranking hand is awarded the game pot. The dealer button moves to the left (clockwise) and a new hand begins.

The electronic games' architecture is partitioned the following three components. The Central Server(CS), the Central Banking System (CBS) and the client. Described below is the overview of each system:

THE CENTRAL SERVER (CS)

The CS is the main control centre for the system. It regulates the main flow of the poker game. Further more, the CS provides a means for interaction between clients and allows players to interact with the CBS.

THE CENTRAL BANKING SYSTEM (CBS)

The CBS stores player specific banking information. CBS also allows players to interact with their personal bank accounts.

THE CLIENT

The client is any person who wish to play the game. He/she will pick up a HHD and log in. The HHD will not store any permanent information, but rather act as an interface between the player and the server.

4.1 Scope of the project

The overall scope of the project is to design a system that implements the Texas Hold'em poker game. The system itself is considered to contain all the functionality requirements such that the client, the server and the central banking system perform as required. The non-functional and functional requirements listed later in this document will dictate all the requirements this system needs to satisfy.

The following environmental conditions of the poker system will not be considered within the scope of this project:

CENTRAL SERVER

- Security:

The issue of security is out the scope of this project, and thus it's expected that the CS has secure connection with other system components of the system such as the HHD, and the CBS. The reason we did not consider is in the scope of our project was because of time constraints. Also the user did not outline the requirement of a system that has secure communication between its components. (Already existed in A-08).

- Response Times:

The issue of response time for events is out of the scope of the project due to extraneous events such as, network delay and data loss. More over, none of the events in the game require quick responses from either the computer or human counter part

- Fraudulent Accounts:

The issue of fraudulent accounts is out of the scope of the project since the system has no control over truly verifying this information. This would also be the responsibility of the third party, the creators/writers of the CBS to verify card information.

HAND HELD DEVICE (CLIENT)

- Safety:

It's out of the scope of the project to ensure that the HHDs comply with industry safety standards. The project is emulating the workings of the HHD, and this it is not our concern to ensure safe hardware.

- Hardware Aspects:

HHDs hardware aspects are out of the project scope. The specifications given by the user were to create the software for the HHDs and not the hardware aspects.

CENTRAL BANKING SYSTEM

- Transactions handling:

The issue of handling transactions for the CBS are out of the scope of this project since it was designed by a third part and access to it's internal structure is restricted.

OVER ALL SYSTEM

- Cost:

Dealing with the costs of hardware and required software to create the system are out of scope since they were provide.

-Portability:

It is out of the project's scope to test whether the system works on different platforms.

5. Terms

The following is a list of terms used throughout the white-paper documentation and documentation in the Rose RT models.

NO LIMIT - The initial raise amount must be at least the amount of the big blind bet. Subsequent raises must be at least equal to previous bet/raise amount in the same round of betting. The maximum raise amount is all of the chips the player has available at the table.

DEALER BUTTON - A marker that is used to indicated the theoretical dealer of each hand. The player who has the dealer button does not deal the cards. It is simple an indicator of who the small blind and big blinds are. After each round of play the button moves clockwise (to the left of the dealer), to the next active player.

HAND - A hand is a combinations of five (5) cards that have some relevance. Starting with the highest hand: A Royal Flush, Straight Flush, Four of a kind, Full House, Flush, Straight, Three of a kind, Two pairs, One pair, and High card.

ROUND - A round is the sequence of all events starting with the placing of the small blind and ending with the determination of the winning hand.

ROUND OF BETTING - A round of betting is considered to be the sequence of actions (fold, check/call, bet/raise) by all of the players in the game, starting with the first player to the left of the dealer. There are a total of four (4) rounds of betting. The first round occurs after each player is dealt two cards. The second round occurs after the flop. The third round of betting occurs after the turn card and fourth and final round of betting ensues after the river card.

POCKET CARDS - The two cards that each player receives, and are unique to each player.

ACTIVE PLAYER - A player is considered to be active if they still hold her pocket cards. i.e. The player has not folded his/her hand.

SMALL BLIND BET - The active player to the left of the player with the dealer button is called the 'small blind'. They are the first to receive the cards but is required to post a small blind. The small blind is equal to half the lower limit bet rounded down to the nearest dollar.

BIG BLIND BET - The active player left of the small blind is called the 'big blind'. They must post a big blind, and is the second to receive the cards. The big blind is equal to the lower limit bet.

(Note: Small blind bets and big blind bets are 'blind' because the player commits some amount of money to the pot before the cards are dealt.)

FOLD - When a players feel like they cannot make a good hand from the cards that were dealt out, they has the option of giving up and 'folding' their hand. Once a player folds their hand, they are out for the hand. All money invested in the pot by them before the fold is left in the pot.

CHECK - A bet of \$0 is a check. This is only allowed if the minimum bet/raise for the current round of betting is also \$0.

CALL - If there is an outstanding bet on the table, the player may match the bet. They may bet/raise if they like.

BET/RAISE - Bet and raise are very similar. A bet is defined as a wager of a certain amount of money by a player in the game. If someone has already bet, then a person may raise. This is done by first calling the amount that was bet, then adding as much more to the pot as player likes. This means that everyone will have to match the higher amount to stay in the round.

ALL IN - 'All in' is where the player bets all the chips they have currently have available at the table.

FLOP - After the first round of betting, three (3) cards are turned over. These three cards are know as the flop.

TURN - After the flop, a round of betting occurs. When this is completed, the next card, fourth card is turned over.

RIVER - When the round of betting for the turn card is completed, the fifth and final card is turned over. This card is called the river. A final round of betting occurs after this card. At the end of the round of betting, a winner(s) is/are determined.

DECK OF CARDS - The game is played with a standard deck of 52 cards. No jokers or wild cards are used.

CHIPS - Chips are a way of representing money. Instead of having to deal with real money on a table, the game will use chips that represent money. When the player quits the game, the chip amount they have remaining will be deposited into their back account.

STATUS UPDATE - This is a server function. After every interaction(event) from the player or server, the server updates the state of all HHDs and it's internal state to reflect any changes.

CONNECTIVITY REQUEST - This is a server function. Every 15 seconds the CS will broadcast a ping and wait for a reply.

ACTION REQUEST - When a players turn arrives, the CS will send an action request. The player is given 15 seconds to take an action. If the player fails to reply within 15 seconds, their hand is automatically folded. If the player does not respond to two consecutive action requests, they are removed from the table.

6. Requirements

The next section lists all the functional and non-functional requirements of this project. Functional requirements describe what the system must do, i.e., what the user requires of the system. Non-functional requirements are guidelines for implementation. They outlines concepts such as security, performance, and UI design.

They begin with a prefix 'FR' (functional requirements) or 'NF' (non-functional requirements) followed by a separator '-' and a unique number. Functional requirements may come from either the 'DG' (discussion group) or 'CaS' (Case Study).

6.1 Functional Requirements

<u>ID</u>	<u>Functional Requirements</u>	<u>Traceability</u>
FR-01	The player is connected to the poker system using a Handheld device (HHD). That is the players are able to interact with the game through the use of a dedicated HHD.	CaS-1.1-2-3
FR-02	Player must be assigned to an empty seat to be part of the table. Once a player is seated, they are not allowed to change position.	CaS-1.2-2-2 DG(TADave)-05-01-28, 2:40pm
FR-03	Player must be placed in First-In-First-Out queue if there is no empty seat at the table.	CaS-1.2-1-3
FR-04	Player must provide login information, user name and banking information, in order to be assigned to a HHD.	CaS-1.2-2-1
FR-05	Player must provide banking information to obtain access to their banking accounts for withdrawal and deposit transactions. Players are able to access their bank account at any time. Players playing money is assigned to their HHD.	CaS-1.4-2-1
FR-06	Bets are visualized and represented in the following way: <ol style="list-style-type: none"> 1. Five dollar chips are coloured White. 2. Ten dollar chips are coloured Red. 3. Fifty dollar chips are coloured Green. 4. One-Hundred dollar chips are coloured Black. 5. Five-Hundred dollar chips are coloured Purple. 	CaS1.3.6-1-1
FR-07	A new player joining the table mid-round, will begin playing in the next round.	CaS-1.2-2-3
FR-08	Player(s) are removed from the table at the end of a round if they do not have sufficient play money (\$10).	CaS-1.2-2-4
FR-09	A round starts with a bet made from the player to the left of the dealer called small blind. Small blind must place a bet of \$5.	CaS-1.3.1-1-1 CaS-1.3.1-1-2

<u>ID</u>	<u>Functional Requirements</u>	<u>Traceability</u>
FR-10	The player left to the small blind must made a bet of \$10. This player is called the big blind.	CaS-3.1-1-3 CaS-1.3.1-1-4
FR-11	Two cards will be dealt out to each active player after the blinds have been placed.	CaS-1.3.1-2-1
FR-12	First round of betting must start with a bet from the player to the left of the big blind.	CaS-1.3.1-2-2
FR-13	After the first round of betting has completed, three community cards, call the flop occur.	CaS-1.3.1-3-1
FR-14	Except for the first round of betting, the player to the left of the dealer begins all other rounds of betting.	CaS-1.3.1-4-1
FR-15	The winner is determined after the last round of betting is completed.	CaS-1.3.1-5-4
FR-16	After the second round of betting is completed, the fourth card, called the turn is turned over.	CaS-1.3.1-5-1
FR-17	After the third round of betting is completed, the fifth and final card, called the river is turned over.	CaS-1.3.1-5-2
FR-18	Last round of betting must take place after the river card is faced up.	CaS-1.3.1-5-4
FR-19	Once a hand has been completed, the dealer button moves to the player left of the previous dealer. Next, the CS gets a new, complete, freshly shuffled deck.	CaS-1.3.1-5-6 CaS-1.3.5-5-3
FR-20	The game can be played by up to 7 players.	CaS-1.3.3-1-1
FR-21	Player must act within 15 seconds of their turn. This is only applicable to players that are involved in current round.	CaS-1.3.3-2-1
FR-22	Player is folded when the player does not respond within the given 15 seconds grace period for the first time.	CaS-1.3.3-2-2
FR-23	Player will be removed from the table if the player does not respond within 15 seconds for the second consecutive time.	CaS-1.3.3-2-3
FR-24	The dealer is identified by a marker known as the button.	CaS-1.3.4-1-3
FR-25	When a player folds, he/she loses any chips that were contributed to the pot. A player may fold at any time.	CaS-1.3.8-3-1
FR-26	When the player does not have enough money to cover the bets and raises, the player is said to go 'all-in', and is simply contesting that portion of the pot that the player's money covers.	CaS-1.3.8.7-1-1
FR-27	A side pot is created after a player has gone all-in and constitutes of the bets made by the remaining players on the table that are not all-in.	CaS-1.3.8.7-1-3
FR-28	At the end of a hand with side pot(s), the player who did not contribute to the side pot(s), is not eligible to win those side pot(s).	CaS-1.3.8.7-1-4
FR-29	An 'all-in' player must leave the table, if he/she loses the hand at the end of a round.	CaS-1.3.8-7-2-4

ID	Functional Requirements	Traceability
FR-30	All financial transactions, withdraw and deposit, must be performed by and take place in an external banking system known as the Central Banking System (CBS).	CaS-1.4-1-1
FR-31	Handheld devices contain a card reader. Both credit and debit cards are accepted. Players must enter a PIN in case they use a debit card to access their account.	CaS-1.4-2-2
FR-32	A player may only withdraw money in multiples of ten dollars from the CBS. The withdrawn amount gets assigned to the player.	CaS-1.4-2-4
FR-33	When the player withdraws money from his/her bank account for the first time, they must withdraw at least \$20. For any other withdrawals, a minimum withdrawal amount of \$10 is required.	DG(TADave)-05-01-18, 10:33pm
FR-34	The player is allowed to access their bank accounts and change their banking information at any time during the game.	DG(TADave)-05-02-08, 1:02pm
FR-35	The poker system must have a GUI for the “server” that allows an administrator to turn the system on or off.	DG(TADave)-05-02-08, 1:06pm
FR-36	A player can decide to leave the game at anytime. Their assets are deposited back into their respective bank accounts.	DG(TADave)-05-01-28, 12:13pm CaS-1.4-3-1
FR-37	When a player leaves the game in an active round, their cards must be folded and the player is removed from the table. A seat is now available for a new player to use. Other players at the table cannot see the change of the player having left, until the player’s turn.	DG(TADave)-05-01-21, 12:04pm CaS-1.4-3-1
FR-38	A player becomes part of the game when the system accepts the player’s login information.	CaS-1.2-2-2
FR-39	Player must either call, raise or re-raise if there is an outstanding bet made previously in this round of betting.	CaS-1.3.1-4-3
FR-40	Player must check or bet if there is no outstanding bet made previously in this round of betting.	CaS-1.3.1-4-2
FR-41	Bets made by players are deposited into a game pot.	Cas-1.3.2-1-1
FR-42	The administrator is able to set the server into testing mode.	DG(TADave)-05-01-14, 15:08
FR-43	Players are able to communicate with other players through the use of chat window.	CaS(Addendum)-1.2-1-1
FR-44	The chat window consists of two parts. First, the sending message part, which is consist of small text field and send button. Second, the received message list which consists of all message received.	CaS(Addendum)-1.2-2-2
FR-45	Each message received contains the name of the player sends the message, the time the message was sent and the actual text message. Received message will be displayed at the end of the received message list of the chat window.	CaS(Addendum)-1.2-3-2

7. Assumptions

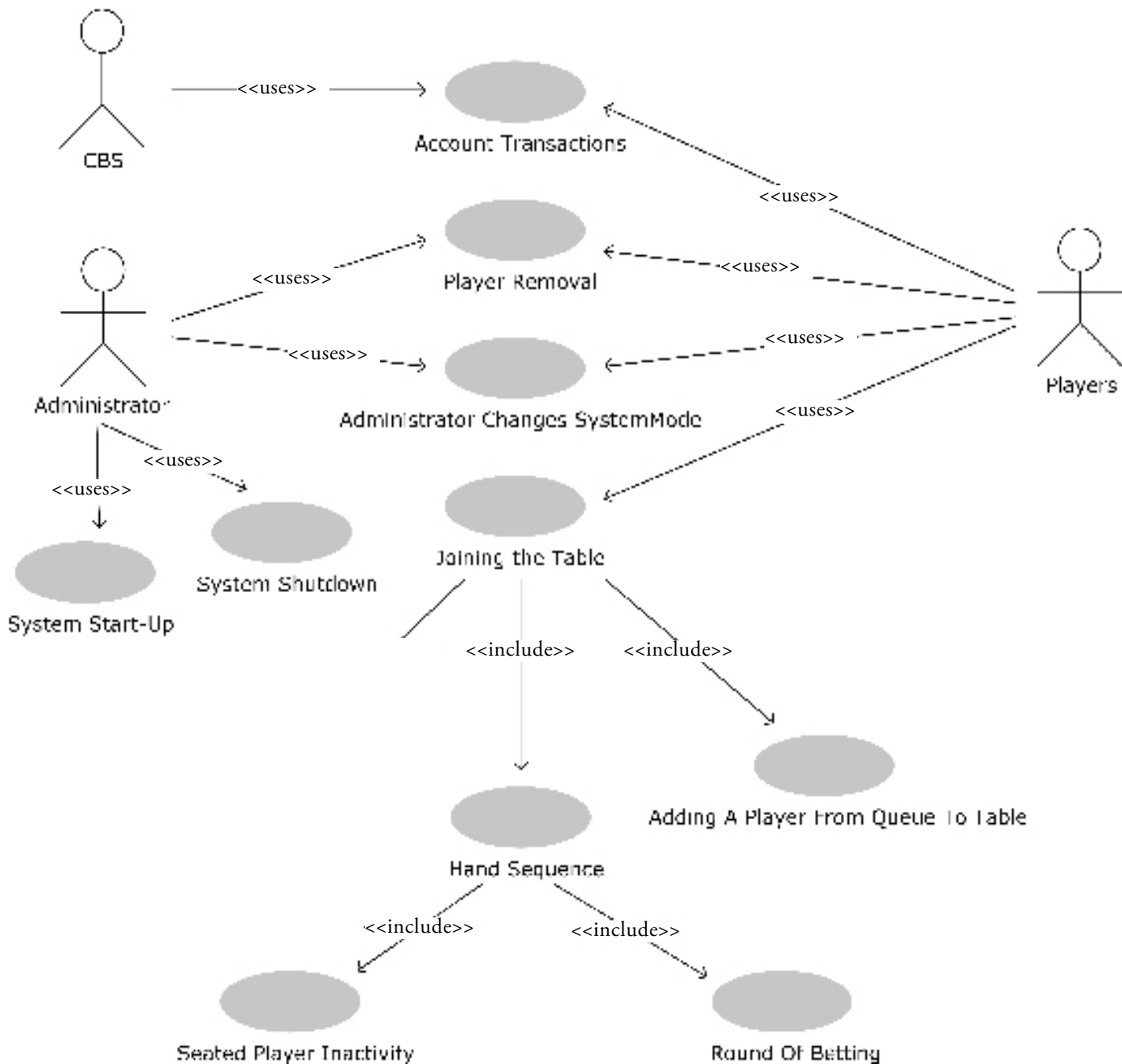
The following sections contains assumptions made based on previous knowledge of the game and further analysis of the system.

Assumptions begin with a prefix 'A' followed by a separator '-' and a unique number.

<u>ID</u>	<u>Assumptions</u>	<u>Traceability</u>
A-01	In a regular FIFO queue, items can only be removed from the front. However in our case, we are implementing a FIFO queue that allows removal from the middle. Furthermore this implementation still maintains the general properties of a queue.	Group Meeting (05/02/20)
A-02	The system is ready for use whenever it has successfully turned on.	Group Meeting (05/02/10)
A-03	A round of betting is completed when all active players match the highest bet.	Group Meeting (05/02/03)
A-04	The server has a UPS and is running on a stable machine. In the event that an administrator shuts down, the server will complete the hand, distribute the winnings, and give all HHDs a message, before turning off.	Group Meeting (05/02/10) (info from TA Dave)
A-05	The HHDs will not run out of batteries or crash. They might experience network disconnectivity or a player may voluntarily exit the table or exit the queue.	Group Meeting (05/02/03)
A-06	A BB is considered a bet. Although it is required and the player does not have a choice, all other players must match that bet.	Group Meeting (05/02/10)
A-07	After a hand, each player must have ten dollars (\$10) or more to continue to the next hand. If the player has less than that amount, he/she is removed from the table.	Group Meeting (05/02/10)
A-08	The issue of security is out of the scope of this project, and thus we will assume that there is a secure connection between all HHDs and the CS, and also between the CS and the CBS.	Group Meeting (05/02/20)
A-09	A check is a bet of \$0.	Group Meeting (05/02/03)
A-10	A call performed on a bet of \$0 is also known as a check.	Group Meeting (05/02/03)
A-11	A round ends if there is only one player in the hand.	Group Meeting (05/02/03)
A-12	The CS will only change states if it is not in an active hand sequence. That is 'testing' mode can only be enabled when there is no active hand sequence in progress.	Group Meeting (05/02/02)

8. Use Case Diagram

This section contains the use case diagram that describes the relationships between actors and use cases. It also relates the use cases to use cases in this system.



9. Use Cases

Use Cases are a simple way of describing a series of events/interactions. They capture the high-level objectives of the application. The following sections contains use cases that describe the functionality of the system.

<u>Use Case: UC-# - Name</u>	<u>Traceability</u>
DESCRIPTION: It's a short summary of the use case.	
ACTORS: Entities which interact with the system	
PRECONDITION: Contains that must be true before a use can begin.	
TRIGGERING EVENT: The event which starts the use case.	
MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES: Events or responsibilities that may occur during the described scenario.	
POST-CONDITION: Condition that will be true on successful completion of the use case.	
RESULTING EVENT: The event that results in the successful completion of the use case.	
ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES: Forks that may occur in the main sequence of steps.	
NON-FUNCTIONAL REQUIREMENTS: Non functional requirements that are relevant to the use case.	
COMMENTS: A free format of section to comment on the user case.	

Use Case: UC-01 - System Start-up	Traceability
DESCRIPTION: The system is currently off and the administrator wishes to allow players to join the game system.	
ACTORS: Administrator	
PRECONDITION: The system must be off.	
TRIGGERING EVENT: Administrator turns the system on.	FR-35
MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES: 1. CS interfaces with the CBS. 2. CS opens an empty table with seven (7) seats available.	FR-30,DD-09 FR-20
POST-CONDITION: CS is ready to accept HHD connections. CS now has access to the CBS.	
RESULTING EVENT: System has started up and is ready for service.	A-02
ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES:	
NON-FUNCTIONAL REQUIREMENTS: Server interface is easy to use.	NF-01
COMMENTS: 1. It is assumed that only the Administrator may power up the system. 2. There are no power failures. 3. There is a secure connection between CS, HHD(s) and CBS.	A-04

Use Case: UC-02 - Joining the table	Traceability
DESCRIPTION: This use case describes the sequence of steps for a player to join the table.	
ACTORS: Player	
PRECONDITION: UC-01 successfully completed.	
TRIGGERING EVENT: Player requests to join the table by providing login information.	FR-04
MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES: <ol style="list-style-type: none"> 1. CS verifies login information and withdrawal amount of a multiple of 10 and greater than 20 dollars. 2. CS validates withdraw amount and assign the withdrawn amount to the HHD (player). 3. Player is assigned to an empty seat at the table by the CS. 4. CS ensures that there isn't an active hand in progress and adds the Player to the players ready for the current round of Poker. 5. CS informs the HHD (player) of the current status of the table. 	FR-32,FR-04, FR-38, FR-31,FR-33,DD-16 FR-34 FR-02,DD-10 FR-07 FR-01
POST-CONDITION: Player's account balance is updated in the CBS.	
RESULTING EVENT: The player is entered in the next hand of Texas Hold'em Poker.	FR-07

Use Case: UC-02 - Joining the table	Traceability
<p>ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES:</p> <p>Alternate-1 (Step #1)</p> <ol style="list-style-type: none"> 1. The CS fails to validate login information. 2. CS informs player of invalid login information. <p>Alternate-2 (Step #2)</p> <ol style="list-style-type: none"> 1. The Player fails to request a withdrawal amount of a multiple of 10 and larger or equal to \$20. 2. CS informs the Player of invalid withdrawal request. 3. CS returns to Main Sequence Step #01. <p>Alternate-3 (Step #2)</p> <ol style="list-style-type: none"> 1. If CBS returns insufficient funds for withdraw amount, that is the player has less than \$20 in their bank account (credit or debit). The CS requests the player for new banking information (credit or debit). 2. Player enters new account information. 3. CS returns to Main Sequence Step #01. <p>Alternate-4 (Step #3)</p> <ol style="list-style-type: none"> 1. If the CS fails to find a seat, CS places the HHD (player) in a waiting queue. 2. CS informs the player of their rank in the queue. <p>Alternate-5 (Step #4)</p> <ol style="list-style-type: none"> 1. If there is an active round in progress, then the player has to wait until he/she can play in the next round. 	<p>FR-04 DD-01</p> <p>FR-33, FR-32 DD-02</p> <p>DD-02, FR-34</p> <p>FR-03 DD-05</p> <p>FR-07</p>
<p>NON-FUNCTIONAL REQUIREMENTS:</p> <p>Player must be able to use the HHD device without any previous knowledge of it. This means that appropriate instructions must be given to the player on the HHD at all times.</p>	<p>NR-01</p>
<p>COMMENTS:</p> <p>In the Main Sequence step 3, we assume that the player is given infinite time to make the withdrawal request. However, the player must provide the withdrawal request in order to proceed.</p> <p>In Alternate 1 step 2, the player is only informed of incorrect banking information. In this case, the player is not part of the game.</p>	

Use Case: UC-03 - Hand Sequence	Traceability
<p>DESCRIPTION: This use case describes the sequence of steps occurring during each hand.</p>	
<p>ACTORS: Player(s)</p>	
<p>PRECONDITION: UC-01 has completed successfully. There are at least two active players seated at the table. The active players have at least \$20. No active hand sequence.</p>	
<p>TRIGGERING EVENT: A 15 second countdown(occurs before every hand) is over and there are at least two active players.</p>	DD-07
<p>MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES:</p> <ol style="list-style-type: none"> 1. CS determines the new dealer. 2. CS informs all players of the dealer position. 3. CS deducts \$5 from the small blind. 4. CS updates the pot and performs a status update. 5. CS deducts \$10 from the big blind. 6. CS updates the pot and performs a status update. 7. CS deals out the pocket cards, starting from the small blind, in clock-wise order. 8. First round of betting begins with input from the first active player to the left of the big blind. 9. After the first round of betting is completed, the flop occurs. 10. The second round of betting begins with input from the first active player to the left of the dealer. 11. After the second round of betting is completed, the Turn card is shown. 12. Third round of betting begins with input from first active player to left of the dealer. 13. After the third round of betting is completed, the River card is shown. 14. The last round of betting begins with input from the first active player to the left of the dealer. 15. After the betting is completed, the winner(s) of the pot(s) (main and/or side pots) are determined. 16. CS will remove player(s) who have less than \$10 at the end of the round. 17. CS gets a new, complete, freshly shuffled deck. 	FR-19, FR-24,DD-08 FR-19 FR-09 FR-41 FR-10 FR-41 FR-11 FR-12 FR-13 FR-14 FR-16 FR-14 FR-17 FR-14,FR-18 FR-15, FR-28 DD-04,FR-08,FR-29 FR-19
<p>POST-CONDITION: CS is not in active round sequence. The 15 second countdown begins again.</p>	
<p>RESULTING EVENT: There is no active hand sequence.</p>	DD-07

Use Case: UC-03 - Hand Sequence	Traceability
<p>ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES:</p> <p>Alternate-01 (Step #07)</p> <ol style="list-style-type: none"> 1. If CS's "testing" mode was turned on by the Administrator, go to UC-11 Main Sequence Step #1. 2. Go back to Main Sequence Step-08. <p>Alternate-02 (Step #9)</p> <ol style="list-style-type: none"> 1. If the CS is in "testing" mode, the Administrator specify the Flop card. 2. CS informs all players of Flop card. 3. Go back to Main Sequence Step-10. <p>Alternate-03 (Step #11)</p> <ol style="list-style-type: none"> 1. If the CS is in "testing" mode, the Administrator specify the Turn card. 2. CS informs all players of Turn card. 2. Go back to Main Sequence Step-12. <p>Alternate-04 (Step #13)</p> <ol style="list-style-type: none"> 1. If the CS is in "testing" mode, the Administrator for the River card. 2. CS informs all players of River card. 2. Go back to Main Sequence Step-14. 	<p></p> <p>FR-42,DD-??</p> <p>FR-13</p> <p>FR-42,DD-??</p> <p>FR-16</p> <p>FR-42,DD-??</p> <p>FR-17</p>
NON-FUNCTIONAL REQUIREMENTS:	
COMMENTS:	

Use Case: UC-04 - Round of Betting	Traceability
<p>DESCRIPTION: This use case describes the steps performed by the system during each of betting.</p>	
<p>ACTORS: Player(s)</p>	
<p>PRECONDITION: UC-03 is in progress.</p>	
<p>TRIGGERING EVENT: CS has requested betting input from player X.</p>	FR-14
<p>MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES:</p> <ol style="list-style-type: none"> 1. CS asks player X to bet 2. CS adds the bet to the current round pot and informs all players of the pot-status. 3. All remaining active players decide to call the highest bet amount starting clockwise from player X. 	FR-39,FR-40 FR-06 FR-39,FR-40,A-11
<p>POST-CONDITION:</p>	
<p>RESULTING EVENT: The round of betting has successfully completed</p>	FR-13,FR-16,FR-17
<p>ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES:</p> <p>Alternate-1 (Step #1)</p> <ol style="list-style-type: none"> 1. If player X decides to fold, the player's cards are removed. 2. If there is only one player left on the table. Go to UC-03 - Main Sequence Step #15. <p>Alternate-2(Step #1)</p> <ol style="list-style-type: none"> 1. If player X decides to fold, the player's cards are removed. 2. If there is more then one player, go back to main sequence #1 and request input from the next player. . <p>Alternate-3 (Step #3)</p> <ol style="list-style-type: none"> 1. If player X decides to raise, CS verifies that the player has sufficient funds to meet the raise amount and that the raise is made as a multiple of 10. 2. Go back to Main Sequence Step #2 with new player X being the player that decided to raise. <p>Alternate-4 (Step #3)</p> <ol style="list-style-type: none"> 1. If a player has insufficient funds to call the highest bet amount, CS denotes that the player is now "all-in". 2. CS creates a side pot with the amount that the all-in player could not meet. 3. Go back to Main Sequence Step #3 for the remaining players. 	FR-25 FR-25 FR-39 FR-26 FR-27

Use Case: UC-04 - Round of Betting	Traceability
NON-FUNCTIONAL REQUIREMENTS:	
COMMENTS: 1. Player X at the triggering event is considered to be the first player to play in each round. 2. If player X bets \$0 then this is also know as a check. 3. If a call is performed on a bet of \$0, then this is also known as a check.	A-10 A-09

Use Case: UC-05 - Adding a Player to the Table from the Queue	Traceability
DESCRIPTION: This use case describes the process of removing a player from the queue and adding the player to the table.	
ACTORS: Player	
PRECONDITION: The queue is not empty.	
TRIGGERING EVENT: A seat is available.	FR-02
MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES: 1. Remove the first player from the queue. 2. Assign the seat to the player. 3. CS informs the player that he/she has been added to the table. 4. Go to UC-02 - Main Sequence Step #3.	FR-02 FR-03
POST-CONDITION:	
RESULTING EVENT: The player is seated at the table and ready to play.	FR-02
ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES:	
NON-FUNCTIONAL REQUIREMENTS:	
COMMENTS:	

Use Case: UC-06 - Player Removal	Traceability
DESCRIPTION: Steps taken by the CS in the event where a player decides to leave the game, or is being forced to leave the game.	
ACTORS: Player	
PRECONDITION: Player is logged in.	
TRIGGERING EVENT: Player requests to leave or is being removed from the game.	FR-36
MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES: <ol style="list-style-type: none"> 1. CS ensures the player is seated at the table and playing in the current hand. 2. The CS folds the player's hand. 3. CS deposits the player's assets to their bank account. 4. Player is removed from the table. 5. When this player's turn arrives, all other players are informed of the removal. 6. CS ensures at least two players are left on the table. 	FR-37 FR-36 FR-37 FR-37 A-12
POST-CONDITION: There is an empty seat at the table.	
RESULTING EVENT: The player is removed.	FR-37
ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES: <p>Alternative-01 (Step #1)</p> <ol style="list-style-type: none"> 1. If the player is in the queue, he/she is removed from the queue. 2. CS deposits the player's assets to their account. 3. CS moves all players, behind the player that was just removed, forward one position in the queue. <p>Alternative-02 (Step #1)</p> <ol style="list-style-type: none"> 1. If the player is on the table, but inactive, his/her assets are deposited into their account. 2. Remove the player from the table and inform all other players of the removal. <p>Alternative-03 (Step #6)</p> <ol style="list-style-type: none"> 1. If there is only one player left on the table, go to UC-03 - Main Sequence Step #15. 	FR-03 FR-36 FR-03 FR-36 FR-01 A-12
NON-FUNCTIONAL REQUIREMENTS:	
COMMENTS:	

Use Case: UC-07 - Account Transactions	Traceability
DESCRIPTION: Description of the interactions that occur when the CS accesses the CBS in order to perform a transaction on the player's account.	
ACTORS: Player, CBS	
PRECONDITION: Player logged in successfully.	
TRIGGERING EVENT: Player requests access to their account.	FR-05
MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES: 1. CS ensures that the CBS is connected. 2. CS ensures that the player is not in 'all in' status. 3. CS requests the player to specify the amount they would like to withdraw. 4. CS performs updates on the players bank account. 5. CS updates the player's playing money.	FR-30 DD-06 FR-05,DD-17 FR-30 FR-01
POST-CONDITION: CBS is available to perform other transactions.	
RESULTING EVENT: The player is informed that the transaction was successful.	FR-01

<u>Use Case: UC-07 - Account Transactions</u>	<u>Traceability</u>
<p>ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES:</p> <p>Alternative-01 (Step #1)</p> <p>1. If the CBS is disconnected, all account transactions are denied except the critical ones. (Refer to DD-18).</p> <p>Alternative-02 (Step #2)</p> <p>1. The CS denies account access if the players status is 'all in'.</p> <p>Alternative-03 (Step #4)</p> <p>1. The withdrawal request is denied and the player is informed of the failure.</p> <p>Alternative-04 (Step #2)</p> <p>1. The account access was a request to deposit assets.</p> <p>2. The CS ensures that the player is seated at the table and not participating in a hand.</p> <p>3. CS requests a deposit amount for the transaction.</p> <p>4. CS verifies that the player has provided valid account information.</p> <p>5. Go to Main Sequence Step #3.</p> <p>Alternative-05 (Step #2)</p> <p>1. The deposit request came from a player who was not seated, then the request is denied.</p> <p>Alternative-05 (Step #3)</p> <p>1. If the player has insufficient funds, they is informed of the error.</p>	<p>DD-18</p> <p>FR-26, DD-06</p> <p>DD-02</p> <p>DD-06 FR-05</p> <p>FR-05 DD-06,DD-17</p> <p>DD-06</p> <p>DD-02</p>
COMMENTS:	

<u>Use Case: UC-08 - System Shutdown</u>	<u>Traceability</u>
<p>DESCRIPTION:</p> <p>Description of the sequence of the steps that occur when the Administrator decides to shutdown the system.</p>	
<p>ACTORS:</p> <p>Administrator</p>	
<p>PRECONDITION:</p> <p>The system is on.</p>	
<p>TRIGGERING EVENT:</p> <p>The Administrator has chosen to turn off the system.</p>	FR-35

Use Case: UC-08 - System Shutdown	Traceability
<p>MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES:</p> <ol style="list-style-type: none"> 1. Inform all the players in the game (queue and table) that the system will shut down. 2. CS waits until the current hand is completed. 3. Remove the players from the queue. (Refer to UC-06) 4. Remove the players seated at the table. (Refer to UC-06) 5. Disconnect from the CBS. 	<p>DD-03 DD-03 DD-03 DD-03 FR-30</p>
<p>POST-CONDITION: Table is empty. Queue is empty.</p>	
<p>RESULTING EVENT: System is off.</p>	FR-35
<p>ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES: Alternative-01 (Step #1) 1. The table and queue are empty, go to Main Sequence Step #5.</p>	DD-03
<p>NON-FUNCTIONAL REQUIREMENTS:</p>	
<p>COMMENTS: A player removal will trigger UC-06</p>	

Use Case: UC-09 - Instant Messaging	Traceability
DESCRIPTION: This use case describes the steps of events when a player decided to open social conversation with other players on the table.	
ACTORS: Player(s)	
PRECONDITION: The system is on. Player seated at the table.	
TRIGGERING EVENT: Player has decided to send an instance message (IM) to all other players on the table by writing the message in the sending part of the chat window.	FR-42, FR-43
MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES: 1. Upon player's send action, the message is passed to the CS. 2. The CS broadcast the message to all players including the originator.	FR-43 FR-42
POST-CONDITION:	
RESULTING EVENT: All other players in the system receive the message.	FR-44
ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES:	
NON-FUNCTIONAL REQUIREMENTS: The GUI is intuitive to use.	
COMMENTS:	

Use Case: UC-10 - Seated player inactivity	Traceability
<p>DESCRIPTION: This use case describes the steps taken by the CS if/when a player is unresponsive during their turn.</p>	
<p>ACTORS: Player</p>	
<p>PRECONDITION: The player is seated at the table and involved in the hand. The CS requests input from the player.</p>	
<p>TRIGGERING EVENT: Player hasn't responded for 15 seconds to a CS action request.</p>	FR-21
<p>MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES:</p> <ol style="list-style-type: none"> 1. CS ensures that the player has failed to respond to two consecutive action requests. 2. CS folds the player's hand. 3. CS removes the player from the table. 4. Go to UC-06 - Step #5. 	FR-23 FR-22 FR-23
<p>POST-CONDITION: The table has one less player.</p>	
<p>RESULTING EVENT: There is now a seat available.</p>	FR-23
<p>ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES: Alternative-01 (Step #3)</p> <ol style="list-style-type: none"> 1. If a player has failed to respond only once, CS records this information and folds their hand. 	FR-22
<p>NON-FUNCTIONAL REQUIREMENTS:</p>	
<p>COMMENTS: The 15 seconds that is given to each player is only applicable to players who are involved in the current hand and their turn has arrived.</p>	

Use Case: UC-11 - Administrator Changes System Mode	Traceability
DESCRIPTION: Description of the sequence of steps that occur when the Administrator decides to enable/disable the 'Testing' mode.	
ACTORS: Administrator	
PRECONDITION: The system is on.	
TRIGGERING EVENT: Administrator decides to change the state of the system.	FR-42
MAIN SEQUENCE OF STEPS AND/OR RESPONSIBILITIES: 1. CS ensures that there are no active hands in progress. 2. CS ensures that the system is in 'online' mode. 3. CS turns on 'testing' mode. 4. Administrator specifies the cards to be dealt.	A-12 FR-42 FR-42 DD-20
POST-CONDITION:	
RESULTING EVENT: The system is in 'testing' mode.	FR-42
ALTERNATIVE SEQUENCE OF STEP(S) AND/OR RESPONSIBILITIES: Alternative-01 (Step #1) 1. The CS is in an active hand sequence and informs manager of inability to comply with request. Alternate-02 (Step #2) 1. The CS is already in 'testing' mode, and it switches back to 'online' mode.	A-12 FR-42
NON-FUNCTIONAL REQUIREMENTS:	
COMMENTS: 'Testing' mode may only be activated or deactivated while the system is not in an active hand sequence.	A-12

10. Design Decisions

The following section contains design decision made by the group that were not explicitly specified by the user.

Design decisions begin with a prefix 'DD' followed by a separator '-' and a unique number.

<u>ID</u>	<u>Design Decissions</u>	<u>Traceability</u>
DD-01	<p>The CS will inform the player of any erroneous occurrences.</p> <p>PROS: This design decision helps meet the NR-01 by providing a user-friendly environment.</p> <p>CONS: The drawback of this implementation is that there is a communication overhead.</p>	Group Meeting (05/02/20)
DD-02	<p>The player will be notified if there are any transactions failures. Transaction failures include the following:</p> <ul style="list-style-type: none"> - Invalid account information. - Invalid withdrawal amount. - Insufficient funds to perform the withdrawal. <p>PROS: This design has been chosen since we chose a non-functional requirement of creating a system that is easy to use and intuitive. By informing the player of any erroneous transactions, the player is kept to up to date.</p> <p>CONS: The drawback of this implement is that there exists extra overhead of communication between client and server.</p>	Group Meeting (05/02/20)
DD-03	<p>The CS will shutdown gracefully. This includes the following steps:</p> <ul style="list-style-type: none"> - Informing all players that shutdown is imminent. - Depositing the assets of all players to respective accounts. - Disconnecting all players from the system. - Disconnecting from the CBS <p>PROS: By performing a graceful shutdown, all information will be kept valid, consistent and up to date with the most recent records. Informing all players of shutdown is consistent with our intent to design an use-friendly system (NR-01).</p> <p>CONS: The drawback of this implementation is that there is a delay in shutting down since the system must wait for all the components to properly shutdown.</p>	Group Meeting (05/02/10)

ID	Design Decisions	Traceability
DD-04	<p>Before the start of each round, the CS will verify that all participating players have at least \$10 to play with.</p> <p>PROS: By setting a minimum amount of playing money, the CS can easily deduct small and big blind bets without having to consider conditions of insufficient funds to meet the blinds.</p> <p>CONS: The drawback of this implement is that it leads to removal of valid players who have sufficient funds to play a hand, as long as they are not responsible for the blinds.</p>	Group Meeting (05/02/10)
DD-05	<p>A player is notified by the CS of their relative rank within the queue.</p> <p>PROS: By allowing players to view their position in the queue, they are kept informed of when it is most likely they will be seated at the table.</p> <p>CONS: The drawback of this implementation is that it leads to the CS broadcasting updates to all players in the queue every time the queue changes. This leads to extra communication overhead between the players and the CS.</p>	Group Meeting (05/02/10)
DD-06	<p>Following FR-34, players can access the bank at any time. However, if a player meets one of these criteria, the possible transactions they can perform will be limited.</p> <ul style="list-style-type: none"> - An “All-In” player is not allowed to access their bank account during the round. - Players in the queue are not allowed to deposit money into their accounts. <p>PROS: These restrictions are imposed due to conflicting requirements. For example, a player who is “all-in” should not have any funds after they went “all-in”. If they were able to access their account, this would defeat the logic of having an “all-in” player. Furthermore, this goes against the Case Study’s statement: “Players are not permitted to leave the game in mid-hand, go get the deed to the ranch, and use it to cover a bet.” (CaS-1.3.8.7-2-1)</p> <p>CONS: The drawback of this is that the player with assets in their bank accounts are removed from the game, after going ‘all-in’. By no allowing deposits in the queue, we can guarantee that the player has at least \$20 before they enter the table. The drawback of this is that if the player withdraws too much money, they will not be able to deposit, unless they leave the queue, or get seated at the table.</p>	Group Meeting (05/02/20)

<u>ID</u>	<u>Design Decissions</u>	<u>Traceability</u>
DD-07	<p>Between each hand, the CS will wait 15s.</p> <p>PROS: By providing a 15s grace period, players have the ability to either access their account or leave the table.</p> <p>CONS: The drawback of this implementation is that a poker game may not immediately begin when there are two or more players ready to play; the CS must wait for the timer to expire before starting a new game.</p>	Group Meeting (05/02/20)
DD-08	<p>The dealer is represneted as the player being highlighted in yellow.</p> <p>PROS: It is a bright yellow square that is clearly visible to all players.</p> <p>CONS: The player must read the legend in order to understand the notation.</p>	Group Meeting (05/03/20)
DD-09	<p>The administrator will be informed if the CS is unable to connect to the CBS. The CS will halt the startup procedure until the administrator informs the CS to reattempt a connection to the CBS.</p> <p>PROS: This design decision forces the CS to verify that the external component, CBS, is properly functioning before the CS attempts to perform any transactions.</p> <p>CONS: The drawback of this implementation is that the CS will not be available until the CS can connect to the CBS.</p>	Group Meeting (05/02/20)
DD-10	<p>The CS will assign players an empty seat at the table. Players are not allowed to choose their own seat.</p> <p>PROS: This simplifies the implementation of the system.</p> <p>CONS: The drawback of this implementation is that it does not allow players to pick their own seat.</p>	Group Meeting (05/02/20)

<u>ID</u>	<u>Design Decissions</u>	<u>Traceability</u>
DD-11	<p>The CS is responsible to lock the seat for a removed player, while a round is in progress and the removed player was playing in that round. That is the CS will not assign any players from the queue, incase if any exist, to that seat. It is part of the requirement to show the change of player removal upon player turn. Therefore, the CS will unlock the seat when the removed player turn comes to act.</p> <p>PROS: The advantage of this design decision to prevent sudden update/change on the table for that seat on other player Hand held devises. That's the seated players will see the change when it is that player's turn, the removed player, by seeing an empty seat and then a new player assigned to that seat from the queue, if any exist.</p> <p>CONS: The drawback of this design decision is that it delays the flow of the game from player perspective. That's newly assigned players can not perform banking transactions such as; first, withdrawing sufficient funds that allow them to compete with other players in the next round, assuming that some players of this round are experts and more likely to play in the next round.</p>	Group Meeting (12/03/25)
DD-12	<p>The player model does not do any parsing, rather is pass on the signal data to the CS, and it's the responsibility of the CS to appropriately use this information.</p> <p>PROS: The reason for this design decision is to meet the project deadline, and therefore, we have forwarded all the work load to the CS side, instead of splitting up the work-load.</p> <p>CONS: The drawback of this design decision is that it puts most of the work overload on the CS side instead of handling some of functionality on the player side. Increasing the overload on the CS might reduce the efficiency of the CS to response to simultaneous requests from CS internal components and/or player side.</p>	Group Meeting (19/03/2005)

ID	Design Decisions	Traceability
DD-13	<p>The GUI has been mainly built from scratch instead of using an open source GUI.</p> <p>PROS: By building the GUI from scratch we are able to ensure that the specified requirement of the system as described in the Case Study are fulfilled. The GUI has been built in accordance with the responsibilities/actions players may perform while playing Texas Hold'em game. As well as reflects the CS responsibilities in terms of informing player(s) of certain events and displaying the appropriate changes on player hand held device. In short, this design decision based on building a GUI that handles the flow of information from the Central Server to players and vice versa.</p> <p>CONS: The drawback of this design decision, first, it is time consuming to handle each transition that arrive from the CS and perform appropriate update on the GUI or handling each player event than having a ready GUI that satisfy our own design. In that case we could have made a good use of the workload on the GUI and contribute that workload on the actual model. The workload also includes the time takes to produce appropriate GUI with nice look that adds publicity to the game aiming to increase the profitability of Juan Pablo poker game.</p>	Group Meeting (22/03/2005)
DD-14	<p>The main model, the Central Server, uses the mediator pattern to implement it's functionality.</p> <p>PROS: The pattern simplifies the work and distribution done by the CS. It also enhances the reusability and maintainability of the system.</p> <p>CONS: The pattern can be difficult to change the system's behaviour in major ways, since a lot of co-ordination is required between all other objects connected to the CS.</p>	Group Meeting (30/03/2005)

ID	Design Decissions	Traceability
DD-15	<p>In a tie winning game the pot going will be divided equally (to the nearest dollar) between all of the winning players. If after division of the pot, the amount that each person receives contains cents, then the first winner (person with the lowest seat number) is given one extra dollar, and all of the other players receive the winning amount rounded down to the nearest dollar.</p> <p>PROS: This decision was made to ensure that when money is divided between players, all of the money ends up going back to the players and rounded dollar amounts are not lost.</p> <p>CONS: Due to our design decision, players that tie and win the same pot can potentially not receive equal amounts of winnings. However, in a case when there is fraction amount in the division of the winning amount the amount will be distributed to closet integer value between players.</p> <p>For example, when two players tie and the pot is 125. The first player (seated at seat #1) receives 63 and the second player (seated at seat #2) receives 62.</p>	Group Meeting (01/04/2005)
DD-16	<p>Players aren't allowed to join the game using names that consist of symbols, except the “_” underscore.</p> <p>PROS: The reason for this design decision is expanded from the idea that we have prevented player from using “ ” symbol as their name, since the parsing design in the model and GUI's code is based on extracting incoming data between the pipe symbol. To make our model more safe to use and simplistic all other symbols have been prevented from use except the underscore symbol since it's frequently used as any other characters from “a-z” or “A-Z”, and digits.</p> <p>CONS: The drawback of this design decision is that the player might not be able to pick their favorite name to play with.</p>	Group Meeting (01/04/2005)

ID	Design Decissions	Traceability
DD-17	<p>Players are not allowed to deposit or withdraw a negative or zero amount.</p> <p>PROS: The reason for this design decision is expanded from the idea that we have prevented player from using “ ” symbol as their name, since the parsing design in the model and GUI’s code is based on extracting incoming data between the pipe symbol. To make our model more safe to use and simplistic all other symbols have been prevented from use except the underscore symbol since it’s frequently used as any other characters from “a-z” or “A-Z”, and digits.</p> <p>CONS: The drawback of this design decision is that the player might not be able to pick their favorite name to play with.</p>	Group Meeting (01/04/2005)
DD-18	<p>Players aren’t allowed to perform any account transaction in case if the CBS is disconnected. However the CS keeps track of critical transations. Crititcal transactions are the ones where the player leaves the table and their assets are waiting to be transfered into their bank account.</p> <p>PROS: This is done to reduce the workload on the CS, trans- actions are not stocked by the casino system. The system main goal is to operate Texas Hold’em game if all other ex- ternal components that are out of the game system control are good to go.</p> <p>CONS: They player might not be able to withdraw or deposit money at any time. If some players prefer to deposit all their money before leaving, they might not be able to do so.</p>	Group Meeting (02/04/2005)
DD-19	<p>The Java GUI follows a model/view/controller pattern.</p> <p>PROS: The controller the RoseRTCController.java is the me- diator between the GUI window, and the Player Model. The code is cleaner and moduler.</p> <p>CONS: Splitting up the application requires more classes and more code. It also requires more co-ordination between all the entities.</p>	Group Meeting (02/04/2005)

<u>ID</u>	<u>Design Decissions</u>	<u>Traceability</u>
DD-20	<p>Following FR-42, when the system is in Testing Mode, the first two selected cards goto the first player, the next two selected cards are given to the second player and so on.</p> <p>If the admin selected less cards than are required by the game, the GUI fills in the rest of the card sequentially(Clubs followed by Diamonds, followed by Hearts, and finally Spades).</p> <p>PROS: This makes it easy to test the functionality of the model. The admin does not have to calculate how many players are playing and how to distribute the cards.</p> <p>CONS: It adds more workload and overhead to the CS and the GUI.</p>	Group Meeting (02/04/2005)

References

Discussion Group:

<http://groups.msn.com/COMP3004Winter2005/general.msnw>

Case Study - Version 1.0 (provided by user)

Case Study - Addendum (provided by user)

PokerStars.com

The background, chips, and cards were used from the PokerStars program.