

# CPSC 224 Final Project

## Project Plan

November 9, 2024

Poker - Texas Hold'em

GOAT TEAM

Prepared by:

Keyu Chen

Daisy Barajas-Manzo

Nick Maximiano

## **1. Project Description**

We plan to design a GUI-based poker game based on the classic Texas Hold'em structure, with additional features to make the game accessible and fun. The game starts with allowing 2-6 players to participate. At the start of each game, players receive two hidden cards and are given the option to choose how many chips they wish to play with, with a minimum starting amount of 500 chips by default.

The game begins with the big blind (first player) and little blind (last player) positions. The big blind player must contribute 2 chips, while the little blind player contributes 1 chip to the pot. Once the big blind player bets, the players following will take turns during a series of table rounds. Each player will have four main actions displayed and only 3 available during their turn: raise, check, call(match), or fold. Once it's the last player's turn, they will bet their 1 chip and another betting round will begin, but this time, the big blind player (player 1) will have the opportunity to raise, check, or fold based on his cards that he can now look at.

As each round progresses, the dealer reveals the first three cards, then the turn (a fourth card), and finally, the river (a fifth card). The goal of each betting round is to either have the best five-card hand or to force other players to fold, ultimately winning the pot for that game round. If there's a tie, the pot is split evenly between the tied players, with any leftover chips given to the dealer. Players who run out of chips or quit the game will be eliminated. Also, to be the "winning player" you would have to have made the most profit from playing, which a leader board will be displayed at the end of all game rounds when the player quits.

To make the game accessible and enjoyable, especially for new players, we will implement clear and timely notifications about the actions of the previous player (e.g., "Player 1 raised 10 chips" or "Player 2 folded" and "Player + ", you're up"). This way, players won't feel lost or confused about the game's progress and will better understand the ongoing table rounds. The notifications will also include details such as the current pot size.

The game will track the progress of each player and provide an in-game leaderboard, allowing players to compare their chip count and profit at any given time. Players can also access a rules guide and view hand rankings to better understand how to make decisions during the game.

The goal of Texas Hold’em is to finish with more chips than the other players by consistently winning pots and outlasting the other players. The game continues with multiple rounds of play until only one player remains. Players can individually exit the game which will fold their current hand if it occurs during the middle of a turn, and upon exiting or being the final player there is a profit screen (which acts as our win/lose screen since poker doesn’t technically have winners and losers).

## 2. Functional Requirements

### a. Splash Screen / Intro Screen

- Upon launch, the application shall display an intro screen featuring a background image related to poker (cards, chips, or a poker table). The splash screen shall also show the game title and a button to start the game. There will be options to set the number of players, player icons, and starting number of chips

### b. Win/Loss Screen with Final Scores

- At the end of each game round, a profit/loss retro screen shall appear, showing the list of players and the player’s name with the most profit on top of the leaderboard, the total chips they have, the profit they made, and the status of all other players (whether they are still in or out). If there is a tie, the pot distribution shall be shown.

### c. Player Information

- Players shall be able to set their name and select an icon before starting the game. If no name is provided, the default name shall be ‘Player 1’ for the first player, ‘Player 2’ for the second, and so on. And if no icon is selected a random one will be selected for them as well.

- Players shall be able to choose how many chips to start with, with the minimum being 500 by default

- Players’ information, including their name, icon, current chip count, and status shall be updated during the game

### d. Game Configuration Settings:

- The game shall allow players to configure at least two settings before starting:
  - o Number of players: players can choose to play with 2-6 players
  - o Number of chips: players can choose the number of chips they would like to start with (at least 500 chips)
  - o Game Length: The game shall allow the option to set the maximum number of game rounds or to play until only one player remains with chips.

### e. Card Dealing and Betting Rounds:

- Each round shall consist of the following stages:

1. Two cards are dealt to each player. Each player can choose to raise, (check or call), or fold

2. Three cards are revealed. Players again choose to raise, check, call, or fold based on their hand
  3. Turn: A fourth card is revealed, followed by another round of decisions
  4. River: The fifth and final card is revealed, followed by the final round of betting for that game round
- After each round, the game shall automatically evaluate the hands and determine the current winner of that game round
  - The dealer shall manage the dealing of cards and ensure that only valid actions are allowed
    - f. *Pot and Chip Management:*
  - A pot shall be created for each game round, accumulating all players' bets in that game round.
  - If a player folds, they forfeit their chips in the pot for that game round.
  - Chips will be displayed clearly for each player, and the total amount in the pot shall be updated with each turn
- g. *Poker Hand Ranking and Winner Determination*
- The player with the best hand at the end of the river shall win the pot. In case of a tie, the pot shall be divided equally among the tied players. Any leftover chips in a split pot shall be awarded to the dealer.
  - The game shall include a system to evaluate hands based on the official hand rankings (e.g., Royal Flush, Straight, Full House, etc.).
  - The system shall handle edge cases, such as when players have equal hand ranks but different values of cards
- h. *Player Elimination, Exit, and Re-entry + Leaderboard:*
- If the player runs out of chips or quits the game early, they shall be eliminated from the game and shall no longer participate in the game, displaying their current chip count and profit/loss of all the players and where they were in the leaderboard.
  - If the player folds after the game round is over, the game will restart with a new game round, with the big blind and little blind positions rotating and the players reentering the game again.
- i. *Player Action Notification:*
- When it's the player's turn to act, the game shall notify them of the actions taken by the previous player and the current pot size.
  - The game shall provide guidance on their available actions (raise, (call or check), fold)
- j. *Raise Rule:*
- If a player raises during their turn, they shall not be allowed to raise again during the immediate next round of betting, even if it's their turn again.

- If a player raises, and the rest of the players call (match) the raise, the player will be skipped for the next round and cannot raise again until the table round is completed, and new cards are revealed.

*k. UI Elements:*

- The interface shall include:
  - o Card Display: An image representation of each of the player's cards and the river cards
  - o Action Buttons: Button for each player's actions (raise, check, call, fold, quit)
  - o Chips and Pot Display: The current chip balance for each player and the total amount in the pot
  - o Player Information: A section showing player names, icons, and chip amount
  - o Action History: A list showing the most recent actions taken by the previous player during the round
  - o Leaderboard: A display showing the players' current and overall ranking based on chip amount and profit

*l. Visual Effects*

- The game shall provide retro vibe visual effects for actions, and the status of the icon, such as when a player wins the game round, when a player is eliminated or folds, and when the player raises the bet.

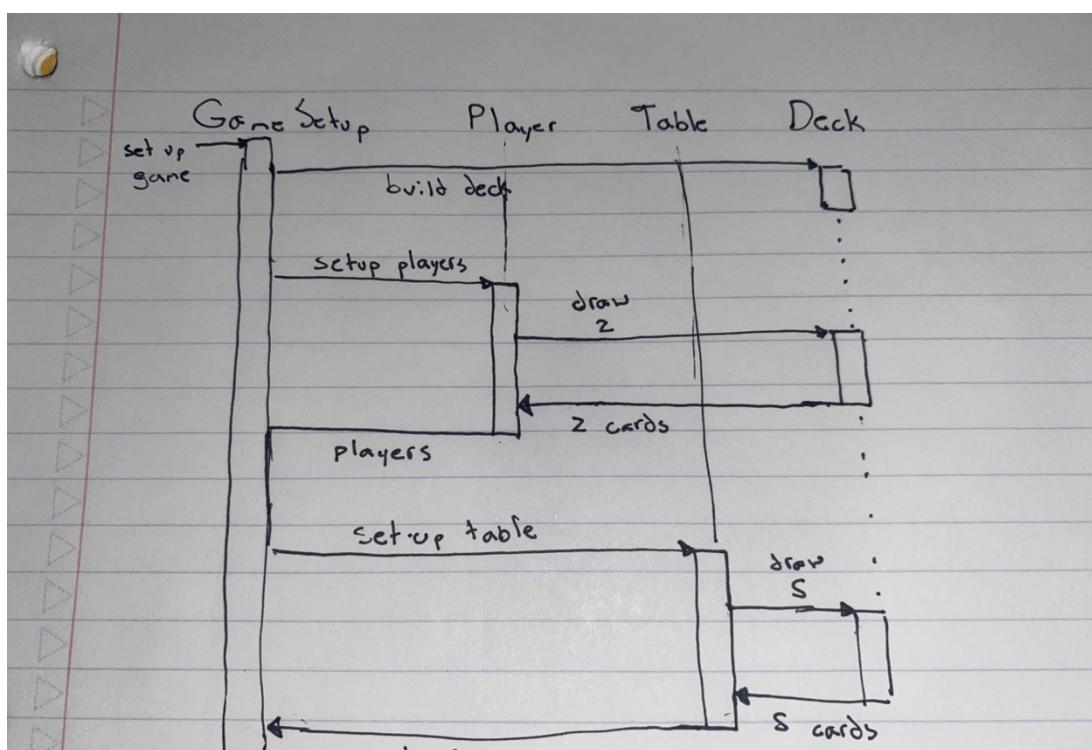
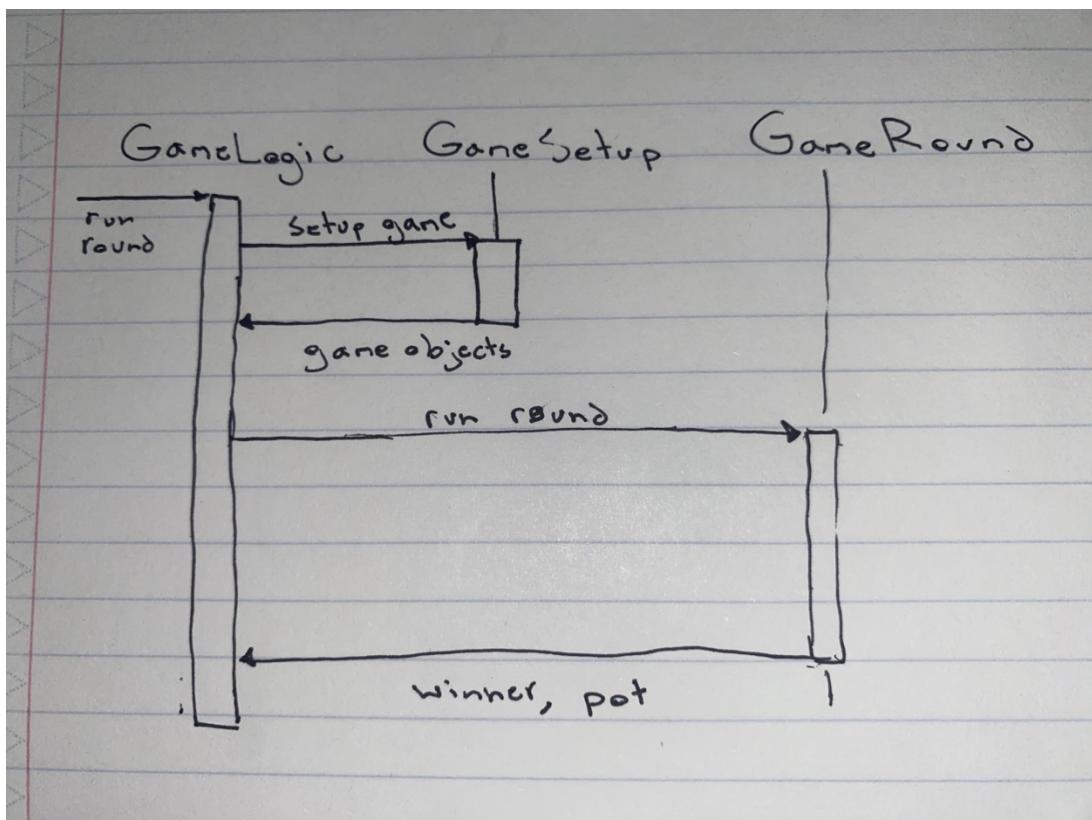
### **3. UML Objects and Sequence Diagram**

#### OBJECTS

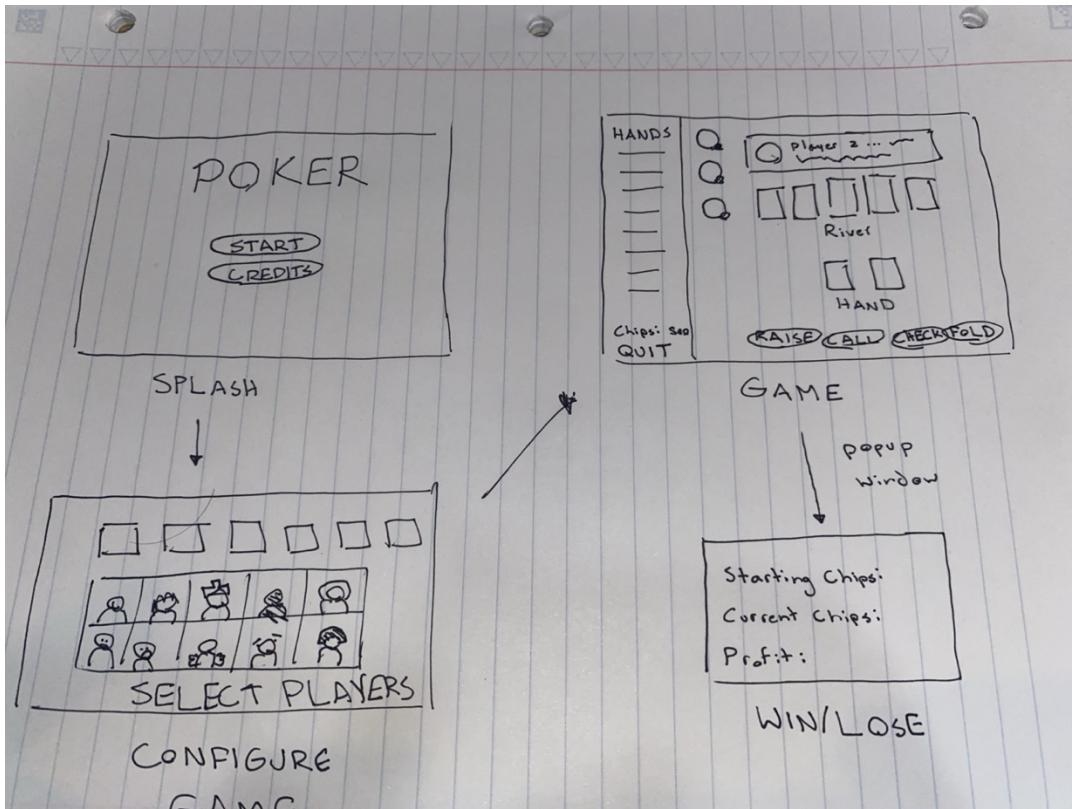
- Main: This object is used to host the various features of the game like the different areas you can be in (splash screen, game), and initiates the GameLogic object.
- GameLogic: This object will be used once the actual poker game is being played. It will lead all other actions, which include game setup, game turns, and logic.
- GameSetup: This object is used to setup the game and fills the game logic with its appropriate components, such as the deck, player, and table. It will be called once at the beginning of every round.
- GameRound: This object is used to initiate a round of poker. This object will consist of turns and all the steps to have a full round of poker from drawing two cards to declaring who has the best hand.
- Deck: This object is used as the “deck” for the game. It contains all 52 cards and ensures duplicate cards don't exist for valid game odds.
- Player: This object represents a player. This will usually be interacted with as a list of players where the list is looped through to interact with the player and allow them to make decisions.

- Table: This object represents the table. It will have its river (the 5 cards it contains) as well as a pot, which is the amount of chips available in the current round. It isn't very interacted with after setup besides the amount of money in the pot.
- Card: This object is the card object which is contained by the deck, player, and table. This object will have a suit, number, and an image for the GUI to display.

### UML Sequence Diagrams



## GUI Template



## 4. Project Planning

### Milestone 1: Initial setup and core classes

- Issue 1: Set up initial GUI layout (game screen, buttons, and menu, pop-up notifications)
- Issue 2: Implement Deck class
- Issue 3: Implement Player class

### Milestone 2: Game Mechanics and Core Logic

- Issue 1: Implement Card dealing and betting rounds
- Issue 2: Implement raise rule
- Issue 3: Implement poker hand ranking and winner determination

### Milestone 3: Features and User Interface

- Issue 1: Implement Win/Loss screen with final scores
- Issue 2: Implement player action notification
- Issue 3: Implement leaderboard

#### Milestone 4: Testing and Debugging

- Issue 1: Perform testing on player class and game setup
- Issue 2: Perform testing on game rounds and hand evaluation
- Issue 3: Test edge cases (chip loss, tie-breaking)

#### Milestone 5: Finalization and Documentation

- Issue 1: Finalize UI for splash screen, menus, and end screen
- Issue 2: Finalize game settings (number of players, chips, and game length)
- Issue 3: Final testing

Task	Estimated Effort (Hours)	Due Date	Description
Set up initial GUI layout	1:30	11/4/2024	Create a simple mockup of the game screen and basic UI border layout style using GUI framework
Draft game rules and Class methods	3:30	11/4/2024	Define game rules, which belongs in each class, and how they will interact with each other
Implement Card Class	0:30	11/5/2024	Create a Card class that defines the cards suits, values, and display images of each card
Implement Deck Class	1:00	11/5/2024	Create a Deck class to store all 52 cards and handle shuffling and dealing
Implement testDeckOfCards	0:30	11/7/2024	Test if the deck can shuffle and disperse cards to the players and the table
Implement Player Class	1:30	11/5/2024	Create a Player class to store players information (chips, name, icon), manage hand, handle actions (fold, match, check, call, raise and drawCard, quit)

Implement testPlayerName and test testPlayerHand	1:00	11/7/2024	Test the Player class to ensure the correct display of player names, hand of cards, and actions
Implement Game Setup Class	2:00	11/5/2024	Create a Game Setup Class to store players, table, startChips, and cardDeck
Implement testGameSetup(chips, name)	1:00	11/11/2024	Test Game Setup to verify correct player setup, including chips, names, and the deck
Implement Table Class	1:30	11/11/2024	Create a Table class to store pot, card river and handle potAdd, potWin, flipCard, and drawRiver
Implement testTable(riverCount, pot)	0:30	11/11/2024	Test card accessibility functionality and the pot management (total chips in the pot)
Implement Game Round Class	4:00	11/12/2024	Implement the Game Round class to store the riverCount and handle startRound, tableTurn, playerTurn, winRound, and the Quit option
Implement testGameRound	1:00	11/12/2024	Test Game Round to ensure rounds are correctly handled, the card in the table is updated correctly
Implement Point Checker Class	4:00	11/12/2024	Implement a Point Checker class to handle pointGet of the game winRound and pointCompare to evaluate and determine the winner based on hand rankings
Implement testPointChecker	1:30	11/12/2024	Test Point Checker to ensure correct hand evaluations, such as

			flush, straight, etc. and determine correct winner
Implement Main Class	0:20	11/13/2024	Create the Main Class which calls the menu which has gameSetup() and starts the game by calling gameRound()
Refine and finalize GUI layout	3:00	11/14/2024	Refine UI design elements like buttons, pot display, leaderboard, and ensure smooth screen transitions
Implement leaderboard and win/loss screen	2:00	11/15/2024	Create a win/loss screen that we plan to add a panel inside of the splash screen frame showing player rankings and final chip counts
Handle Edge Cases	2:00	11/15/2024	Handle edge cases like players running out of chips, multiple players folding, ties
Code clean-up and documentation	2:00	11/16/2024	Perform final code clean-up, add comments, and ensure the game is well-documented