

Poker - Analysis

Group 3

Team Members:

Ryan Mc Quillan
Grace Angulu
Ethan Clarke
Daniels Aprubs
Malachy Byrne
Evelina Prosyankina

1. Refined Requirements Specification

Purpose

This is a Requirements Specification document for an online, internet-based game of Poker reliant on Object Orientation Design. Poker is a card based game that is used for gambling. Players must bet the same amount as the previous player if they want to continue playing. They can even bet to intimidate other players to fold (quit the round).

This document is an analysis of what is required to make such a game. Our project, called MyPoker, requires our group to decompose our chosen variety of poker into a set of requirements.

Document Conventions

Acronym	Full Title
OOP	Object Orientated Programming
x	X Coordinate
y	Y Coordinate

Intended Audience

All group members will have access to this document as a mark of what they hope to achieve and how they go about development. When an object or figure is ambiguous then this document will be referenced for the next course of action.

Project Scope

The game is designed to provide an exciting version of Poker online through the internet. It is designed as a variation of “Texas Hold’em” that we have dubbed “Dublin Hold’em”. (mechanics are explained in detail within our Structured Walkthrough).

The game is online so we will have an online network for players to connect to and play together without interruption. Also, as an online game, we will host our game on the web.

2. Scenarios

Game Begins (New Round)

Current System State

The system brings each player into the game, of which we will describe three. The three players, John, Alice, and Peter, are each dealt two cards. The values of these cards are not relevant to the informal description, but the values can only be viewed by their owner. The dealing is handled by the system, but a random player will be assigned as the “Dealer”. Each player’s pool of money is set to the default amount.

Informal Scenario

Peter is assigned as Dealer, so John is the first to play. John places a bet of 1, which is subtracted from his money pool. Then Alice chooses to Call this bet, and also puts 1 into the pile. John Raises the bet, and bets 2 from his money pool.

Next Scenario

The System continues with the next round, which begins with dealing The Flop.

The Flop

Current System State

There has been one round of bets made, and the pool is equal to 4, and the highest bet is 2. Each player has made one bet and it is now the second turn. Each player still possesses 2 cards, the values of which are only available to their owners.

Informal Scenario

The system deals out three cards which are displayed to all players. These cards are known as The Flop. Another round of betting occurs as described above, continuing with the current prize pool and minimum bet. John Raises the bet to 3, the prize pool increases to 7. Alice chooses to Fold, she makes no bet and cannot participate in this round any more. Peter Calls the bet, 3 is subtracted from his personal pool and the prize pool is increased to 10.

Next Scenario

The next round, The Turn, occurs. System deals one more card and prompts players for bets.

The Showdown

Current System State

The 3 turns of dealing have been completed, and betting has concluded. The prize pool is at 25. Alice has folded and will not be considered in any further actions on the System's behalf.

Informal Scenario

John and Peter's cards are revealed, they can now each see each other's cards. The System evaluates the values of each player's cards against that of the Flop, Turn, and River. This is checked against the System's database of possible winning-hands one can have in the game. John is found to have a hand with a greater point value, and the System announces him as the winner of this round. He receives the value of the prize pool as reward. The players are prompted for whether or not they would like to play another round.

Next Scenario

If at least one player chooses to play another round, a New Round scenario will occur. Otherwise there are no more scenarios.

3. Primary Class List

Card

The Card class will be used to store information about a card.

Attributes:

- suit - the suit of the card
- value - the value of the card

Methods:

- matches - checks that the card has the same value as another card
- less_than - checks that the card has a lower value than another
- greater_than - checks that the card has a higher value than another card

Player

The Player class will be used to store information about a player.

Attributes:

- hand - the cards in the player's hand
- bet - The current bet the player has made this round
- cash - The amount of money the player has

Methods:

- hand_type - The value of the player's hand
- pay - add money to the player's cash

Pot

The Pot class will be used to store information about the pot.

Attributes:

- value - amount of money in the pot
- last_raise - Last player to raise the pot

Methods:

- raise - raise the value of the pot

Deck

The Deck class will be used to store information about the deck.

Attributes:

- cards - The cards in the deck

Methods:

- shuffle - Shuffles the deck

CardDisplay

The CardDisplay class will be used to store information about the displayed cards.

Attributes:

- value - The value of the card
- suit - The suit of the card
- x - The x position of the card
- y - The y position of the card

Methods:

- draw - Draws the card to the screen

PotDisplay

The PotDisplay class will be used to display the pot.

Attributes:

- pot - The pot represented by the counter
- x - The x position of the counter
- y - The y position of the counter

Methods:

- draw - Draws the counter to the screen

PlayerDisplay

The PlayerDisplay class will be used to display the player on the screen.

Attributes:

- card_count - The number of cards held by the player
- x - The x position of the player
- y - The y position of the player

Methods:

- draw - Draws the player to the screen

3a. Class Responsibility Collaboration Descriptions

Card	
suit value matches less_than greater_than	card

Pot	
value last_raise raise	pot player

Player	
hand bet cash hand_type pay	card pot player

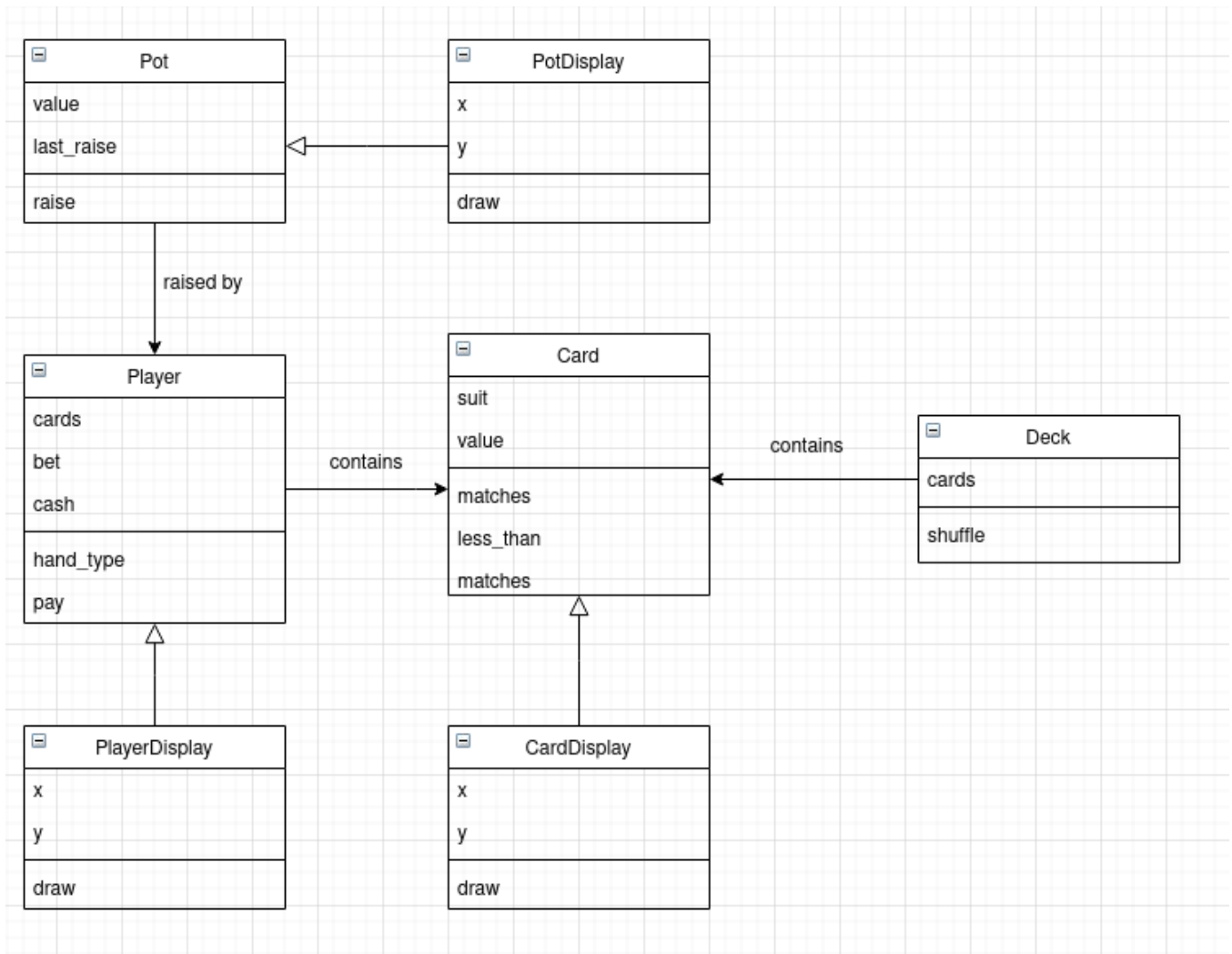
Deck	
cards shuffle	deck card

CardDisplay	
value suit x y draw	card

PotDisplay	
amount x y draw	pot

PlayerDisplay	
card_count x y draw	player card

4. Class Diagrams



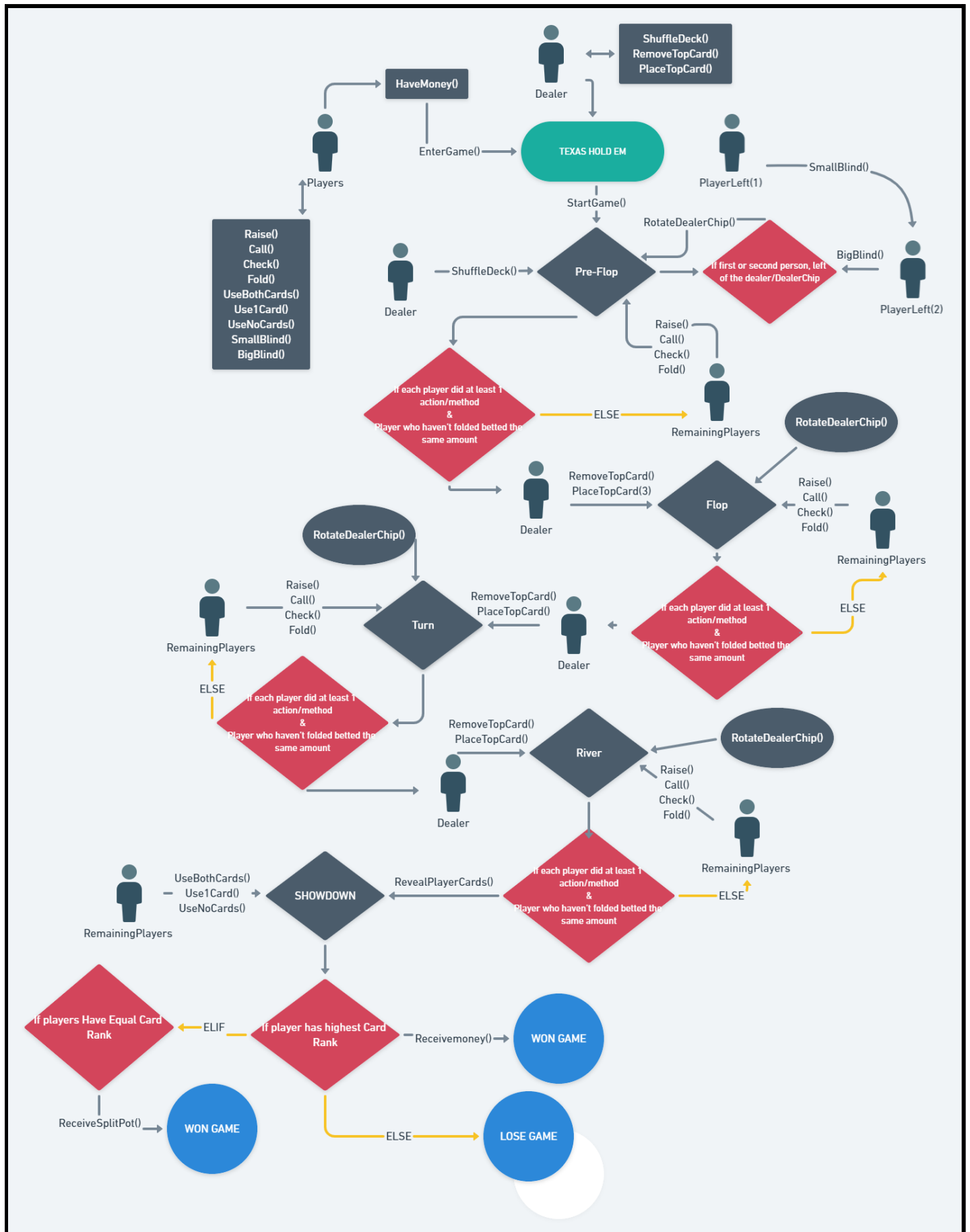
5. Use Case Templates + Diagram

Use Case 1: Win Game

USE CASE		Win Game
GOAL IN CONTEXT		Player "X" Bets an ambiguous amount of money and receives the highest value for cards, winning the round.
Scope & Level		Players, Dealer, Rules
Preconditions		Player has Money
Success End Condition		Players wins the final round and receives all the accumulated bets.
Failed End Condition		Player loses at or before the final round and loses the amount they bet.
Primary, Secondary Actors		Players, Dealer
Trigger		Player bets on the initial round.
DESCRIPTION	Step 1 2 3 4 5 6	ACTION Dealer shuffles deck and passes 2 cards to each player face down. Player left of dealer places Small Blind. 2 nd player left of dealer plays Big Blind. This rotates left by 1 person every round. Players either Raise, Call, Check or Fold. Dealer discards top card and reveals top 3 cards. This state is known as "The Flop". Players either Raise, Call, Check or Fold. Dealers removes top card and reveals the 4 th card to get to what is known as "The Turn". Players either Raise, Call, Check or Fold. Dealers removes the top card and reveals the final 5 th card to get to the final state known as "The River". Final round of betting occurs. Players either Raise, Call, Check or Fold. Players then reveal their hand and whoever possesses the highest ranked order wins the round and obtains the bet amount that accumulated.
EXTENSIONS	Step	BRANCHING ACTION If player participates in the small or big blind they won't bet in the first round, affecting the betting amount and overall prize amount. When players reveal their hand, they may choose both, one <u>or</u> none of their cards with the community cards, which can affect the outcome of who wins the round. The player can prematurely fold and lose without getting to the final round.
SUBVARIATIONS	Step 1	BRANCHING ACTION Players may not bet more money than they currently have on hand. Players cannot Raise if another Player goes all in.

Use Case 2: Player Wins Split Pot

USE CASE		Player receives split pot
GOAL IN CONTEXT		Player gets to the final round and splits the accumulated bet amount with at least 1 or more players.
Scope & Level		Players, Dealer, Rules
Preconditions		Player has Money
Success End Condition		Player gets to the final round and wins the round along with at least 1 other Player
Failed End Condition		Players loses at or before the final round and loses the amount they had bet.
Primary, Secondary Actors		Players, Dealer
Trigger		Player bets on the initial round.
DESCRIPTION	<div>Step 1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> <div>6</div>	<p>ACTION</p> <p>Dealer shuffles deck and passes 2 cards to each player face down. Players either Raise, Call, Check or Fold. Player left of dealer places Small Blind. 2nd player left of dealer plays Big Blind. This rotates left by 1 person every round.</p> <p>Dealer discards top card and reveals top 3 cards. This state is known as "The Flop".</p> <p>Players either Raise, Call, Check or Fold.</p> <p>Dealers removes top card and reveals the 4th card to get to what is known as "The Turn".</p> <p>Players either Raise, Call, Check or Fold.</p> <p>Dealers removes the top card and reveals the final 5th card to get to the final state known as "The River". Final round of betting occurs. Players either Raise, Call, Check or Fold.</p> <p>Players then reveal their hand. If at least 2 players have the same rank of cards that are the highest in the round they split the prize amount by the number of players that have that same rank.</p>
EXTENSIONS	STEP 1	<p>BRANCHING ACTION</p> <p>The amount a player receives from the split pot is equal to the amount betted by everyone divided by the number of players who received that same rank. This will affect the amount outcome amount.</p> <p>A rare occurrence can happen where the community cards are the highest rank which means all players still in the round will have the amount split between them, if they choose none of their cards and stick to the community cards.</p>
SUBVARIATIONS	Step 1	<p>BRANCHING ACTION</p> <p>Players may not bet more money than they currently have on hand.</p> <p>Players cannot Raise if another Player goes all in.</p>



6. Result of “Structured Walk-Through”

Our poker project will be a play on Texas Hold'em called 'Dublin Hold'em'.

Dublin Hold'em is a simple poker game when you understand the rules, however, when the game begins, we hope to have a tutorial game, which will slowly teach new players how to play the game. The goal of Dublin Hold'em is to use the two cards you are dealt at the beginning of the round, as well as the cards dealt on the table by the dealer, and make the best possible five card hand.

The rules are just as simple and are as follows:

- Each player gets two cards, face down to avoid other players seeing them
 - These are called 'hole cards'
- There will be several betting rounds, where each player gets to bet a certain amount of chips.
- Eventually, five cards are dealt, face up on the table, in the middle for everyone to see
 - These face up cards are the community cards. And the players use these community cards to build a 5 card hand with their own two card hole.

The mission is to make the best five card poker hand with the cards you have and the community cards (this is 7 cards in total).

We are hoping to make the Dublin Hold'em card game as accessible as possible for new players and players well versed in the game. There will be a playable tutorial, as well as written rules for those that may just want to brush up on them before playing.

The two classes that we will be using will be the Player and the Pot. These two classes will hold everything related to the game, keeping code clear and concise.

The player class will hold all information about the person playing the game, those that are playing with the main player, their level, and anything else that may be related to the player.

The pot class will hold everything else, from the player's current hand, the money and chips on the table, the minimum bet and anything related to the more currency based aspect of the game.

7. Minutes

- Organised meetups between classes to discuss what we did and what we will do.
- Decided variation of poker.
- Simplified rule set slightly.
- Tutorial will be implemented.
- Distributed tasks to team members, based on strengths and weaknesses.
- Assigned tasks such as which part of analysis will be done by whom, other general roles such as “backend dev”, “visual designer” and “frontend dev.”