

Team: Sri Sai Manjunadh Valluru, Chansovannary Sar, Mohith PrakashReddy,
VenkataSrinivasaReddy Gade

Professor: Ryan Culpepper

Date: 12/13/2023

Course: CS410

Specification Report for Basic Rummy Game

Purpose of the Document:

The specification report for this Basic Rummy game will provide a detailed overview of our outline and design layout for this project. It will include an outline of the functionality of the game, like rules, gameplay, strategy, and overall player interactions. This report is intended to inform and plan out the game's development according to the specifications and feedback given by the instructor.

Game Components to Consider:

- All NPCs with a player count between 2-7
- Kings = High & Aces = Low
- The game can be consisted of 1-2 standard playing card decks

Full Game Overview:

- All players are placed in a circle just like UNO
- Each player draws a card, and the player with the lowest value card is the first dealer. If there is a tie, all players will redraw again.
- Dealer shuffles and deals cards to players which are:
 - 10 cards for 2 players
 - 7 cards for 3-4 players

- 6 cards for 5-7 players
- Players can look at their hand, but is hidden from opponents
- The remaining cards will become the draw pile, and discard the first card on top of the draw pile to create the discard pile.
- The rotation of turns starts from the left of the dealer going clockwise.
- The player can take one card from the draw pile, which is hidden, or the discard pile, which is shown.
- If at any point the draw pile runs out, shuffle the discard pile to create a new draw pile. The last card on the discard pile before being shuffled becomes the first card in the new discard pile.
- After a player draws a card they can either meld or lay off:
 - Melding is having 3 or more of a kind (3 king cards no matter their suite) or having 3 or more of a run (Cards of the same suite which are in an increasing or decreasing order: 2, 3, 4, 5 or A, K, Q, J) (Ace can be used as 1 or 11 for melding but can't wrap around meaning: K, A, 2 is not allowed)
 - After making a meld, you must lay the cards face up on the table.
 - Melding is optional and can be only done once a turn.
 - Laying off is adding 1 or more cards from your hand to an existing meld on the table.
 - Laying off is optional and there is no limit for the number of cards to lay off per turn.
 - After melding, laying off, or not taking your turn, you must discard one card from hand, face up, to the discard pile.

- If you drew a card from the discard pile, you can't discard the same card on the current turn.
- Next player, clockwise goes, and if a player gets rid of all cards from their hands, they get points based on the value which is accumulated from all the cards in the other player's hands.
- Point system:
 - Face cards = 10 points each
 - Ace = 1
 - Number = Points based on their value
- The player goes "Rummy" if they get rid of all their cards in their first turn without previously melding or laying off. In this event, they get twice as many points.
- Once a player has an empty hand and gets their points, the role of the dealer is shifted to the next player, going clockwise by 1, and the new hands are dealt.
- This whole process repeats if, in the game, a player receives a pre-determined amount of points or plays a pre-determined amount of hands (which can be counting the amount of different dealers).
- The player with the most points or reaches the pre-determined amount of points first wins.

Question About Game Implementation:

Regular Way:

- Rotating the dealer and determining the game's end based on a set number of points or hands is the common way to play multiple rounds.

Simple implementation:

- There are no rounds or set amounts of points to reach in a game. Instead, a player wins by either going Rummy first or emptying their hand, to make the game shorter.

Question for Professor:

- Which implementation would you prefer our team to proceed with? The implementation to count rounds and accumulate points, or the simple implementation of where a player gets an empty hand first to win.

Special Situations and Exception Handling

- If we take the route of creating a point-based system, we need to take into consideration breaking ties after the set amount of hands, or the set amount of rounds. We could allow ties, but we need to consider some alternate approaches. The game can either allow a tie for the outcome, continue the game for another set of rounds, run a tiebreaker round, or even score based on tiebreakers. That is TBD.

Team Interaction:

Since our team consists of four individual programmers for this project, we are using Discord for communication and GitLab for control, all of us need to create a systematic approach and communication to avoid deadlocks, and disagreements, and create an overall smooth development. These are some of the practices we are trying to incorporate for this project:

- Regular meetings and check-ins
- Dividing and assigning tasks accordingly
- Using Gitlab appropriately:
 - Making sure everyone understands how to use Git for branching, merging, pulling requests, and overall how to solve conflicts.

- We also want everyone to frequently commit and push to the repository to keep the code for the project up-to-date.
- Review code and merge requests frequently
- Clear communication via Discord
- Documenting overall progress:
 - This can be done via a shared online document, in the GitLab repository, or even in the code with frequent comments.
- Frequent code testing
- Respect individuals and team

References:

Here is where we referred the rules for Rummy:

[How to play Rummy](#)