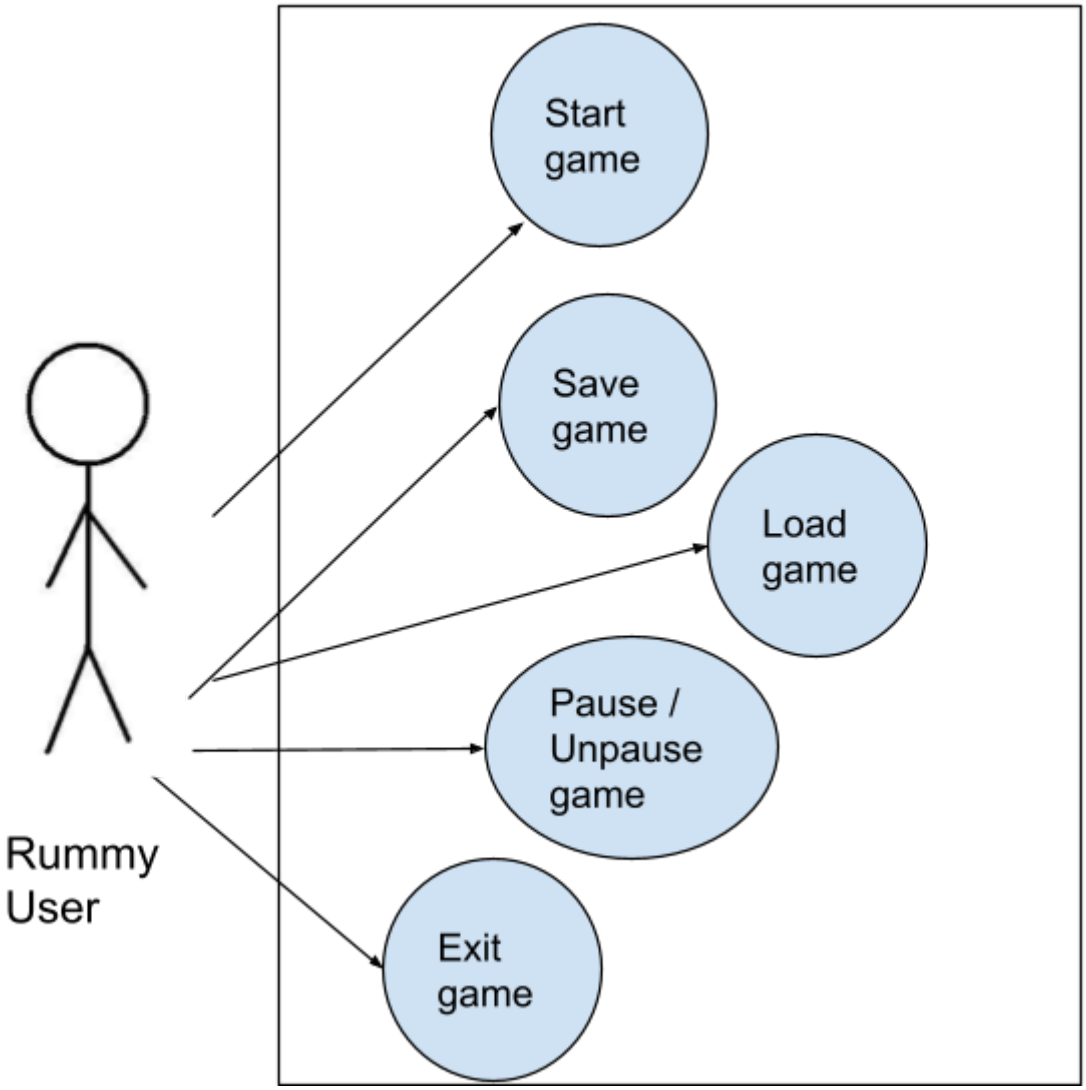
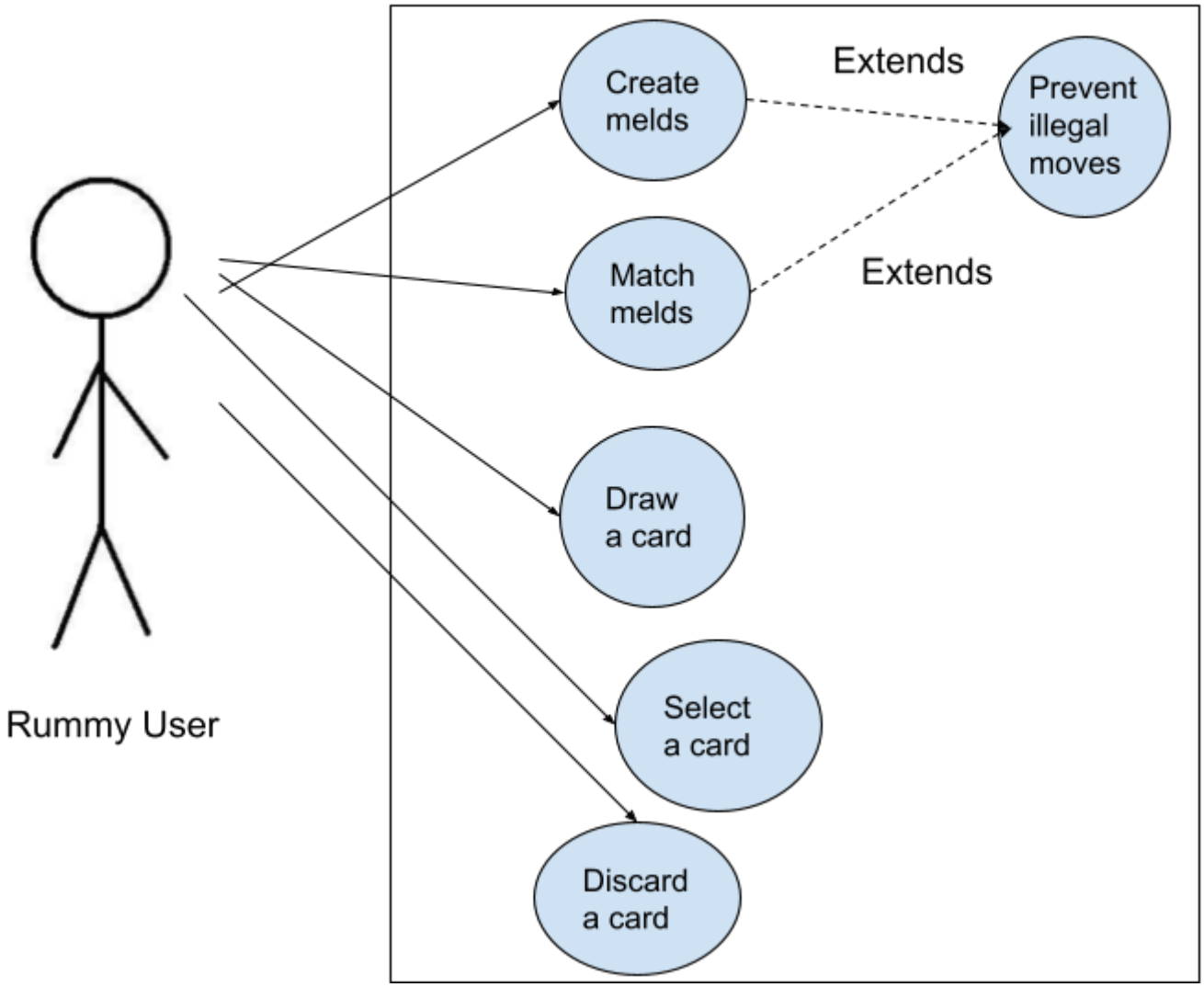
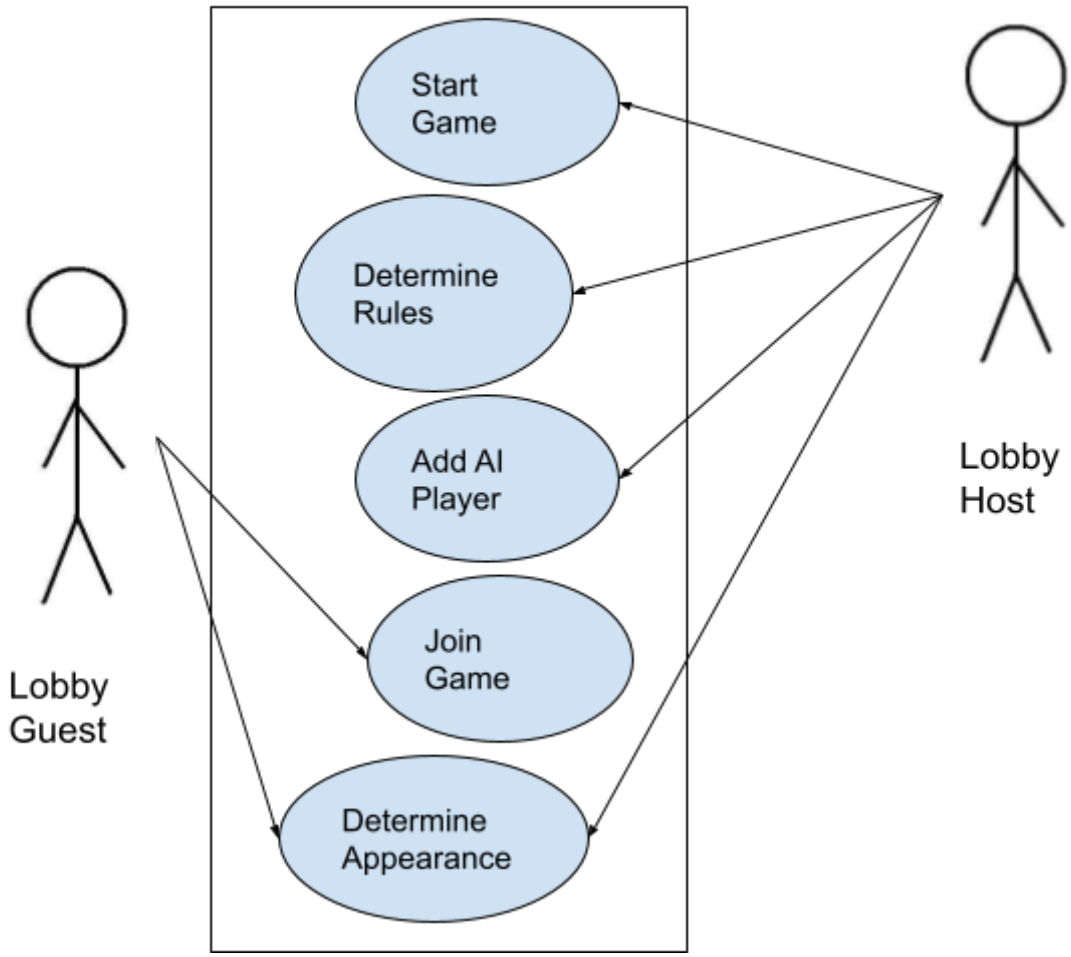
Super Rummy Application: Managing Game State



Super Rummy Application: Gameplay Loop



Super Rummy Application: Starting the Game



Title: **Use Case ID: 1**	Start Game
Description:	This use case describes the steps required to begin a game of Super Rummy.
System Under Design:	Super Rummy
Primary Actor:	Lobby Host
Participants:	Lobby Guest
Goal:	Begin a game.
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes the user has accessed the Super Rummy website and is on the main selection menu
Success Postcondition:	The game has begun.

Actor:	**System Responses:**
1.) Host has arrived at the Super Rummy site.	
	2.) Super Rummy displays the start screen, including a button labeled “Start Game”
3.) Host clicks the button.	
	4.) Super Rummy changes from the start screen to the game screen.
5.) Host sees that the display has changed.	
	6.) Super Rummy deals cards to each player. The number of cards is determined by customizable rules.
7.) Host and Guests can see the cards in their hand face-up, and see cards in opponent’s hands face down.	
	8.) Super Rummy moves the top card of the deck into the discard pile and turns it face-up.
9.) Host and Guests can see the top card of the discard pile.	

Title: **Use Case ID: 2**	Join Game
Description:	User will enter a lobby hosted by another player.
System Under Design:	Super Rummy
Primary Actor:	Lobby Guest
Participants:	Lobby Host
Goal:	Multiple human players will be in the same lobby.
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	The Host has shared the room code with the Guest.
Success Postcondition:	The Guest has entered the lobby.

Actor:	**System Responses:**
1.) Guest types a room code into the “Join Lobby” space, then hits enter.	
	2.) Super Rummy determines whether an open lobby exists with that code, and if there is, it lets the Guest into the lobby. Otherwise, it will either display a message that the lobby is full or that there is no lobby with that code.
3.) Guest sees the error message, or Guest sees that they have entered the lobby, Host sees that the Guest has entered their lobby.	

Title: **Use Case ID: 3**	Determine rules
Description:	This use case describes the steps taken to select specific rules for playing Rummy
System Under Design:	Super Rummy
Primary Actor:	Lobby Host
Participants:	The user has a device with internet access, and has accessed the Super Rummy website.
Goal:	Make selections about the available rules.
Following Use Cases:	
Invariant:	The user has a mobile device
Precondition:	This use case assumes the user has navigated to the “edit rules” screen from the main menu.
Success Postcondition:	The user has selected the rules from a specific ruleset and is ready to start the game

Actor:	**System Responses:**
1.) Host has navigated to the rules menu.	
	2.) Super Rummy displays rules, and sliders with which to change what they’re set to. i.e. “How many decks of cards will be used” with the slider ranging from 1-3.
3.) User adjusts a slider by clicking it and dragging it to the side.	
	4.) Super Rummy displays that the slider has moved and associated rules have been changed.
5.) User sees these changes.	

Title: **Use Case ID: 4**	Add AI Player
Description:	This use case describes the steps for adding AI players to lobbies.
System Under Design:	Super Rummy
Primary Actor:	Lobby Host
Participants:	Lobby Guest
Goal:	Have an additional AI player in the lobby.
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	The Host has entered the Super Rummy website.
Success Postcondition:	An additional AI player will be in the lobby.

Actor:	**System Responses:**
1.) Host clicks the “Add AI Player” button.	
	2.) Super Rummy adds an AI player to the displayed list of players in the lobby.
3.) Host and Guests can see the additional AI player in the list of players.	

Title: **Use Case ID: 5**	Load Game
Description:	This use case describes the steps for loading a game.
System Under Design:	Super Rummy
Primary Actor:	Lobby Host
Participants:	Lobby Guest
Goal:	Load a previously saved game.
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.

Precondition:	Host and Guests have entered the website, all desired Guests are in the lobby, and the Host knows the Save Code for a previously saved game.
Success Postcondition:	An additional AI player will be in the lobby.

Actor:	**System Responses:**
1.) Host enters the Save Code into the “Load Save Code” field, and hits enter.	
	2.) Super Rummy checks that the Save Code is valid, and that there aren’t more people in the lobby than were in the saved game. If valid, it displays the “Choose Seats” screen.
3.) Host and Guests see their username as well as the usernames of the players in the saved game, and can click the username of a previous player.	
	4.) The system will display that user’s username beneath the selected previous player’s username.
5.) Host and Guests will see, beneath each previous player’s username, the username of the player who has chosen that previous player.	
	6.) Once the Host and each Guest has chosen a previous player, the system will display the Resume Game button to the host.
7.) Host clicks the button.	
	8.) Super Rummy loads the game, with the Host and each Guest having the hand and score of their chosen previous player.
9.) Host and Guests will see this change and can continue play as normal.	

Title: **Use Case ID: 6**	Determine Appearance
Description:	This use case describes the steps for changing the appearance of Super Rummy.
System Under Design:	Super Rummy
Primary Actor:	Super Rummy User
Participants:	
Goal:	Allow for UI customization.

Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	User has navigated to the Appearance menu, either from the pause menu or the main menu.
Success Postcondition:	The UI will have changed.

Actor:	**System Responses:**
1.) User navigates to the Appearance menu.	
	2.) Super Rummy displays various display options, organized into Card Back, Card Front, and Board, as well as a button to return to the menu from which they arrived.
3.) User clicks an option from one or more categories.	
	4.) System displays that the selected option has been selected.
5.) User exits the menu.	
	6.) User sees the menu from which they arrived, which the UI changed to the settings they chose, if applicable.

Title: **Use Case ID: 7**	Draw a card
Description:	This use case describes the steps taken for the user to draw a card from the top of the deck
System Under Design:	Super Rummy
Primary Actor:	Rummy users
Participants:	
Goal:	Draw a card from the deck or discard pile.
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes that the user must draw a card from the top of the deck in order to begin their turn

Success Postcondition:	The user successfully drew a card and is ready to create melds, match, or discard

Actor:	**System Responses:**
1.) Before having drawn a card this turn, the user taps either the deck or the discard pile.	2.) Super Rummy moves the top card of the tapped pile to the user's hand. If the discard pile was tapped, all users can see what is now the top of the discard pile.
3.) User sees that the card has been moved to their hand.	

Title: **Use Case ID: 8**	Prevent illegal moves
Description:	This use case describes the steps for creating illegal moves for the rummy game
System Under Design:	Super Rummy
Primary Actor:	Rummy user
Participants:	
Goal:	To prevent players from making certain moves
Related Use Cases:	Create Melds Match Melds
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes the player has drawn a card to begin their turn and is ready to make a move
Success Postcondition:	To deny the user from making a move that would be considered "illegal"

Actor: 1.) The user attempts to perform an illegal move during their turn 3.) User sees the message that their move was illegal.	**System Responses:** 2.) Super Rummy prevents the user from performing this move, providing a description of why the move was illegal, if possible.

Title: **Use Case ID: 9**	Create melds
Description:	This use case describes the steps necessary for creating a meld
System Under Design:	Super Rummy
Primary Actor:	Rummy users
Participants:	
Goal:	Create a meld as either a set or a run, sets and runs are defined as follows but are subject to change based on selected rules: ● **Set:** 3 or more of the same card based on its value (2-10, jack, queen, king, ace). ○ Suit and color DO NOT MATTER when making a set ● **Run:** 3 or more cards in ascending/descending order of the same color/suit. These cards must be in order, one after the next, by value! ○ This order cannot wrap back around from the highest to the lowest card (cannot create a run with a high ace that goes ace, 2, 3 and vice versa)
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes that the user has drawn a card to begin their turn, and has at least three cards selected.
Success Postcondition:	The user is successfully able to create a meld in the form of either a set or a run

Actor:	**System Responses:**
1.) User selects at least three cards.	2.) Super Rummy presents a “Create Meld” button.
3.) User selects the “Create Meld” button.	4.) Super Rummy checks that the move is legal, if it is, it moves the selected cards onto the board. If not, it displays a message that the move was illegal, with details why if possible.
5.) User sees the cards move onto the board or sees the message stating that the move was illegal.	

Title: **Use Case ID: 10**	Match melds
Description:	This use case outlines the steps required for matching cards to other users melds
System Under Design:	Super Rummy
Primary Actor:	Rummy users
Participants:	
Goal:	When it is the users turn, they should be able to match or “append to” to other players melds as long as they have placed down a meld of their own
Related Use Cases:	Prevent Illegal Moves (extends)
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes the user has one or more cards selected and at least one meld is present on the board.
Success Postcondition:	The user is able to match a card to another users meld

Actor: 1.) The user, with at least one card selected, taps a meld on the board. 3.) User views updated hand and sees all users view matched card	**System Responses:** 2.) Super Rummy checks that the move is legal, and if it is, moves the selected cards to be part of the selected meld.

Title: **Use Case ID: 11**	Select Card
Description:	This use case outlines the steps of a user selecting cards in their hand.
System Under Design:	Super Rummy
Primary Actor:	Rummy users
Participants:	
Goal:	Allow the user to select cards so that they can match and create melds.
Related Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes the user has one or more cards in their hand.
Success Postcondition:	The user has selected one or more cards.

Actor: 1.) The user clicks a card. 3.) User views that the card has moved.	**System Responses:** 2.) Super Rummy moves that card up slightly in the UI, so that it stands out from the others. It is registered as selected.

Title: **Use Case ID: 12**	Discard a card

Description:	This use case describes the steps taken to discard one card and add it to the end of the discard pile
System Under Design:	Super Rummy
Primary Actor:	Rummy users
Participants:	
Goal:	The user must be able to discard one card and add it to the discard pile in order to end their turn
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes that the user has exactly one card selected.
Success Postcondition:	The user successfully discards one card and adds it to the end of the discard pile

Actor:	**System Responses:**
1.) User, with one card selected, clicks the discard pile.	
	2.) Super Rummy moves that card to the discard pile, displaying it for all users, and ends User's turn.
3.) User sees that the card has moved and that their turn has ended.	

Title: **Use Case ID: 13**	Assign card point values
Description:	This use case describes the steps for assigning point values to cards
System Under Design:	Super Rummy
Primary Actor:	Rummy Users
Participants:	
Goal:	Give a specific point value to each card as follows: ● 2-9 = 5 points ● 10-King = 10 points ● Ace = 15 points(high), 5 points(low)
Following Use Cases:	

Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes that the user has created a meld/has current melds displayed or has matched a card to a meld
Success Postcondition:	Each card successfully has its own point value which is used to determine user scores at the end of each round

Actor:	**System Responses:**
1.) User has matched cards or melds displayed on the board	
	2.) Super Rummy assigns total point value to the meld or matched card and displays it to the user. follows the point values given in the “Goal” section
3.) Users see displayed point values and can determine the current score of the game	

Title: **Use Case ID: 14**	Pause / Unpause game
Description:	This use case describes the steps taken in order to pause the current progression of a game
System Under Design:	Super Rummy
Primary Actor:	Rummy users
Participants:	
Goal:	To provide the ability to halt the current progression of the game as well as providing a “pause menu” which outlines specific options such as “save game”, “exit game”, “help”, etc.
Following Use Cases:	

Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes that the user has begun a game
Success Postcondition:	The user is able to successfully pause the game and has access to the pause menu and its options

Actor:	**System Responses:**
1.) User is in game and selects the pause option in order to select option from pause menu	
	2.) Super Rummy receives pause request, halting the current game progression and displays the pause menu with its multiple options to the user
3.) User sees pause menu and decides if they would like to select one of the options given or to resume play	
	4.) Super Rummy's next actions depend on users choice, however, base case would be to provide the user with a "resume play" input display
5.) User selects to "resume play"	
	6.) Super Rummy unpauses the game and game progression commences again

Title: **Use Case ID: 15**	Save game
Description:	This use case describes the steps taken to save the progress of an ongoing game
System Under Design:	Super Rummy
Primary Actor:	Rummy users
Participants:	
Goal:	To allow users to save the current state of their game if they want to come back to play later
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.

Precondition:	This use case assumes the user has begun playing a game
Success Postcondition:	The user can save the state of the game and come back to it to play at a different time

Actor:	**System Responses:**
1.) User has started a game and would like to save the game progress in order to resume play at a different time	
	2.) Super Rummy shows display which asks user to “save game” and invokes the user to tap on it through some form of vibrant UI
3.) User sees “save game” display and taps the option to save the progress	
	4.) Super Rummy stores the current game state (melds, hands, total points, etc.) and displays a message that the game state has been saved.
5.) User sees the message that the save has gone through successfully.	

Title: **Use Case ID: 16**	Exit game
Description:	This use case describes the steps taken when a user wants to exit the game
System Under Design:	Super Rummy
Primary Actor:	Rummy user
Participants:	
Goal:	To be able to exit the current game either closing the or bringing the user back to the main menu
Following Use Cases:	
Invariant:	The user has a device with internet access, and has accessed the Super Rummy website.
Precondition:	This use case assumes the user is ready to stop playing the current game they are in and have accessed the “pause menu”
Success Postcondition:	The user successfully exits the game bringing them to either the main menu or exiting the

Actor:	**System Responses:**
1.) User is ready to exit game and is looking at the “pause menu” display	
	2.) Super Rummy displays “exit game” option in “pause menu” and allows the user to select it through UI
3.) User sees “exit game” display option and selects it	
	4.) Super Rummy shows new display message stating that all unsaved progress will be lost, and provides an option to “Exit to Main Menu”
5.) User selects the option to return to Main Menu.	
	6.) Super Rummy returns user to the main menu.
7.) User sees that they have been returned to the main menu.	