

CSCI 4448 - Project Part 2 - Individual

Name: Theerarun Tubnonghee (Steve)

Project: Blackjack!

Requirements:

- **UR-001** - “ As a player, I can view the rules of Blackjack.”
- **UR-002** - “ As a player, I can pick how many computer players to play with.”
- **UR-003** - “ As a player, I can be dealt cards.”
- **UR-016** - “ As a player, I can save the current state of the game and continue to play.”
- **UR-017** - “ As a player, I can continue my saved state of the game.”
- **UR-018** - “ As a player, I can start the game and play with some amount of chips.”

Use Case Documents:

Use Case ID:	UR-001
Use Case Name:	Player selects “Help” to view rules of Blackjack.
Description:	Player view the rules and how to play Blackjack in the user interface, before start to play any game.

Actor:	Players		
Pre-conditions:	Player opens Blackjack! program.		
Post-conditions:	Player views the rules of Blackjack.		
Frequency of Use:	Only when the player wants to understand on how to play Blackjack.		
Flow of Events:		Actor Action	System Response
	1	Player clicks on Help button in the first display before starting any game.	The View displays the rules to the display.
Variations:	Player does not click on Help button, instead he starts a game.		
Exceptions:			
Developer Notes:			

Use Case ID:	UR-002
Use Case Name:	Player selects how many computer players to play with.
Description:	Player selects a number of computer players to play with between 0 and 3 computer players.

Actor:	Players		
Pre-conditions:	Player selects to start a new game.		
Post-conditions:	Player entered a number of computer players to play with.		
Frequency of Use:	Every time Player starts a new game.		
Flow of Events:			
		Actor Action	System Response
	1	Player clicks on New Game button in the first display.	The View displays the Pre-Game options.
	2	Player enters a number of computer players in the “Enter Computer Box”	The game stores the selection of Player.
Variations:	Player chooses from 0 to 3 computer players.		
Exceptions:	Default selection is zero computer player.		
Developer Notes:			

Use Case ID:	UR-003
Use Case Name:	Player can be dealt cards.
Description:	Dealer distributes cards to players every new round.

Actor:	Players, Dealer
Pre-conditions:	Player placed a bet for a new round.

Post-conditions:	Player owned two cards in his hand.										
Frequency of Use:	Every time a new round starts.										
Flow of Events:	<table> <tr> <th></th><th>Actor Action</th><th>System Response</th></tr> <tr> <td>1</td><td>Player placed the amount of the bet.</td><td>The Table stores all bet on table, and automatically start a round.</td></tr> <tr> <td>2</td><td>Dealer distributes a single card to all players and himself, and another single card to all players and himself.</td><td>The Table moves those distributed cards to Players' hands, and Dealer's hand.</td></tr> </table>			Actor Action	System Response	1	Player placed the amount of the bet.	The Table stores all bet on table, and automatically start a round.	2	Dealer distributes a single card to all players and himself, and another single card to all players and himself.	The Table moves those distributed cards to Players' hands, and Dealer's hand.
	Actor Action	System Response									
1	Player placed the amount of the bet.	The Table stores all bet on table, and automatically start a round.									
2	Dealer distributes a single card to all players and himself, and another single card to all players and himself.	The Table moves those distributed cards to Players' hands, and Dealer's hand.									
Variations:	Empty seats will not be dealt cards.										
Exceptions:											
Developer Notes:											

Use Case ID:	UR-016
Use Case Name:	Player can save the current state of game and continue to play.
Description:	Player can choose to save the current state of the game during the placing bet state, and Player can navigate back to play the current game.

Actor:	Players				
Pre-conditions:	Player is in the state of placing the bet, and the game will show the "SAVE" button on the bottom right corner of the display.				
Post-conditions:	Player entered the saving name and saved the game.				
Frequency of Use:	During the placing bet state of each new round.				
Flow of Events:	<table> <tr> <th></th><th>Actor Action</th><th>System Response</th></tr> </table>			Actor Action	System Response
	Actor Action	System Response			

	1	Player is in the placing bet state	The View displays a SAVE button on the bottom right corner of the interface.
	2	Player clicks SAVE button.	The View brings up Saving Interface to let Player specify the name of the saving state.
	3	Player enters the name of saving state, and hit "Save Game".	The system stores the state of MVC and players' information and display the saved information.
	4	Player hits "Go Back"	The system brings Player back to the game and continue to play.
Variations:	Player does not click save button during the placing bet state.		
Exceptions:	Player placed a new bet to start a new round.		
Developer Notes:			

Use Case ID:	UR-017
Use Case Name:	Player can continue the saved game.
Description:	Player can select to load a saved game, and continue to play the game.

Actor:	Players		
Pre-conditions:	Player is in the first display of the program.		
Post-conditions:	Player continues the saved game, and start a new round.		
Frequency of Use:	Every time Player decides to load a saved game.		
Flow of Events:			
		Actor Action	System Response
	1	Player clicks on “Load Game”	The View brings up the saved

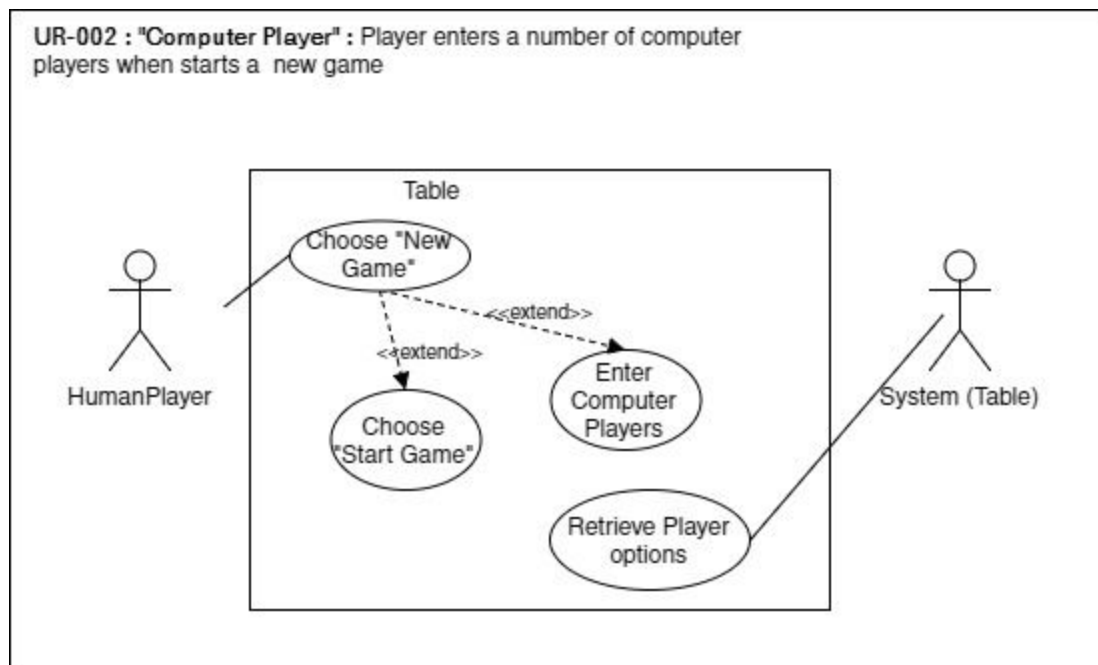
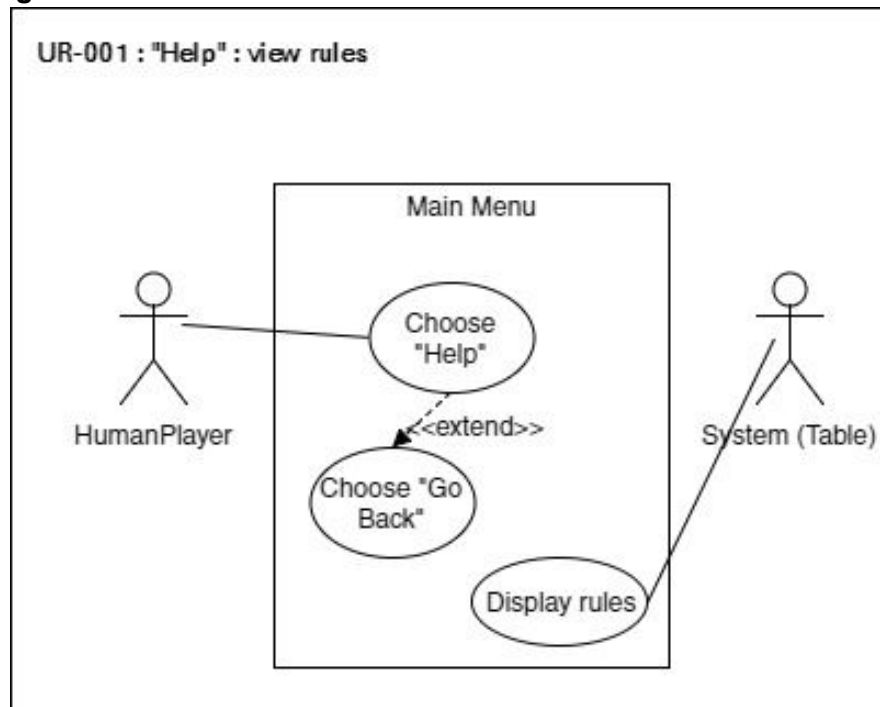
			game lists.
	2	Player selects the saved game.	The View highlighted on the saved game that player clicked on.
	3	Player clicks "Load".	The system loads up the current table MVC of the saved game to the display.
Variations:	Player does not select to load a saved game.		
Exceptions:	Player selects a new game or help buttons.		
Developer Notes:			

Use Case ID:	UR-018
Use Case Name:	Player can start the game with some specified amount of chips.
Description:	Player selects amount of chips to start the game with by entering the amount of money.

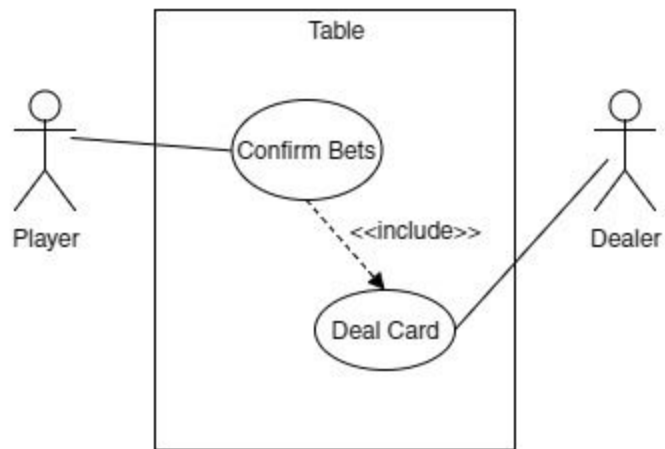
Actor:	Players		
Pre-conditions:	Player selects to start a new game.		
Post-conditions:	Player entered some amount of chips to play.		
Frequency of Use:	Every time Player selects to start a new game.		
Flow of Events:			
		Actor Action	System Response
	1	Player clicks on “New Game” button.	The View displays the Pre-Game options.
	2	Player enters the amount of money to start the game with.	The game stores the selection of Player.
Variations:			

Exceptions:	
Developer Notes:	

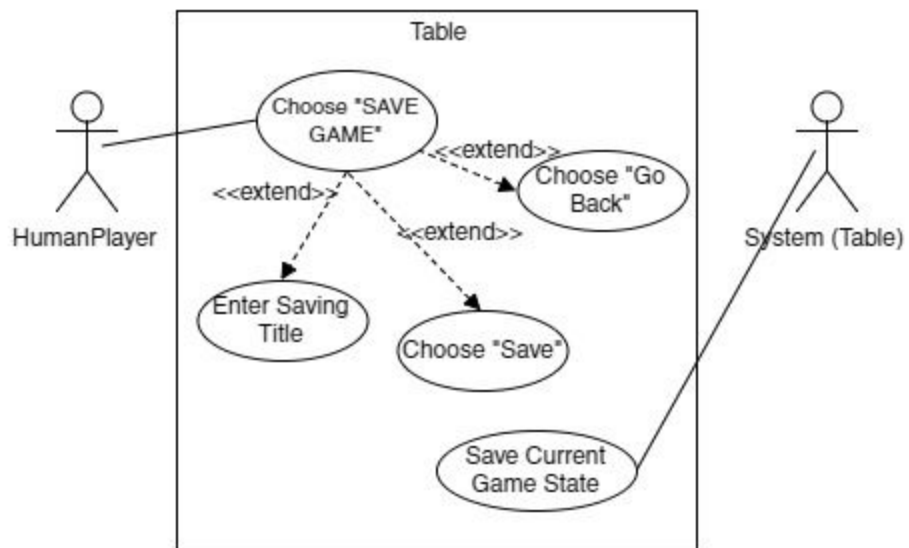
Use-Case Diagram:



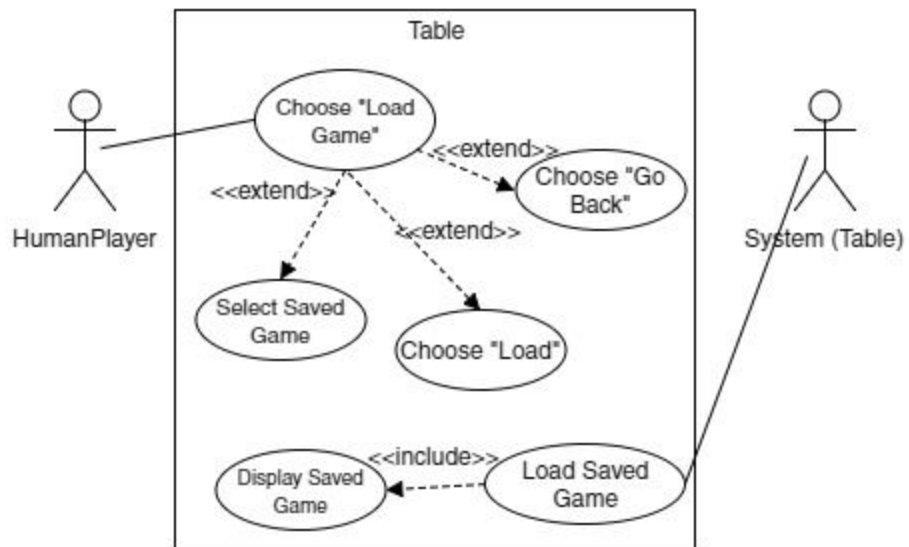
UR-003 : "Deal" : Player is dealt cards after placed bet.



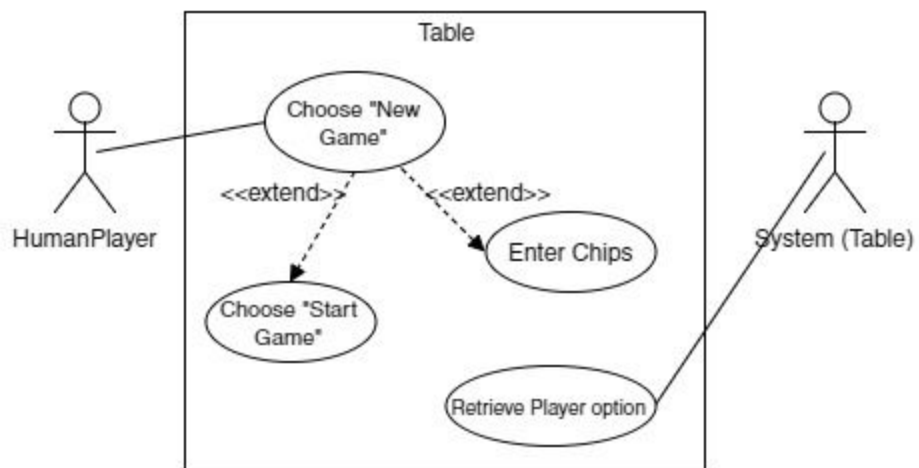
UR-016 : "Save Game" : Player saves game or go back and play current state



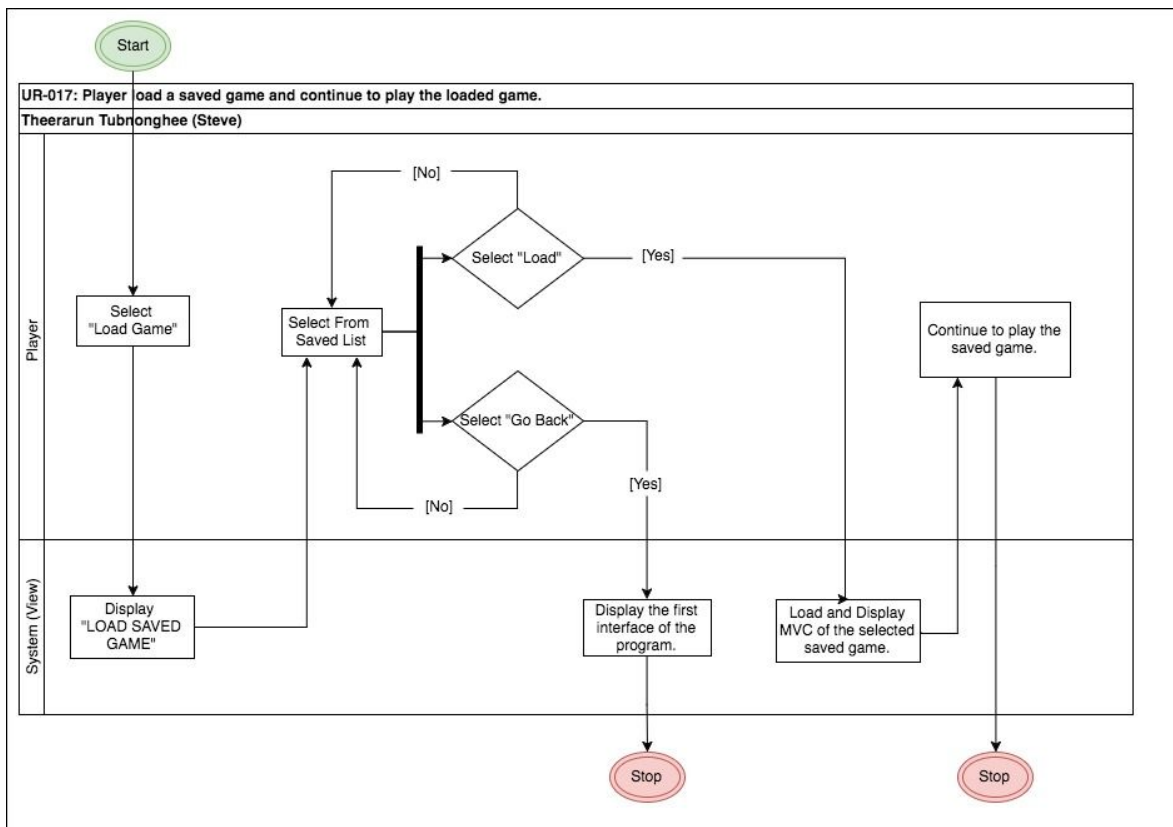
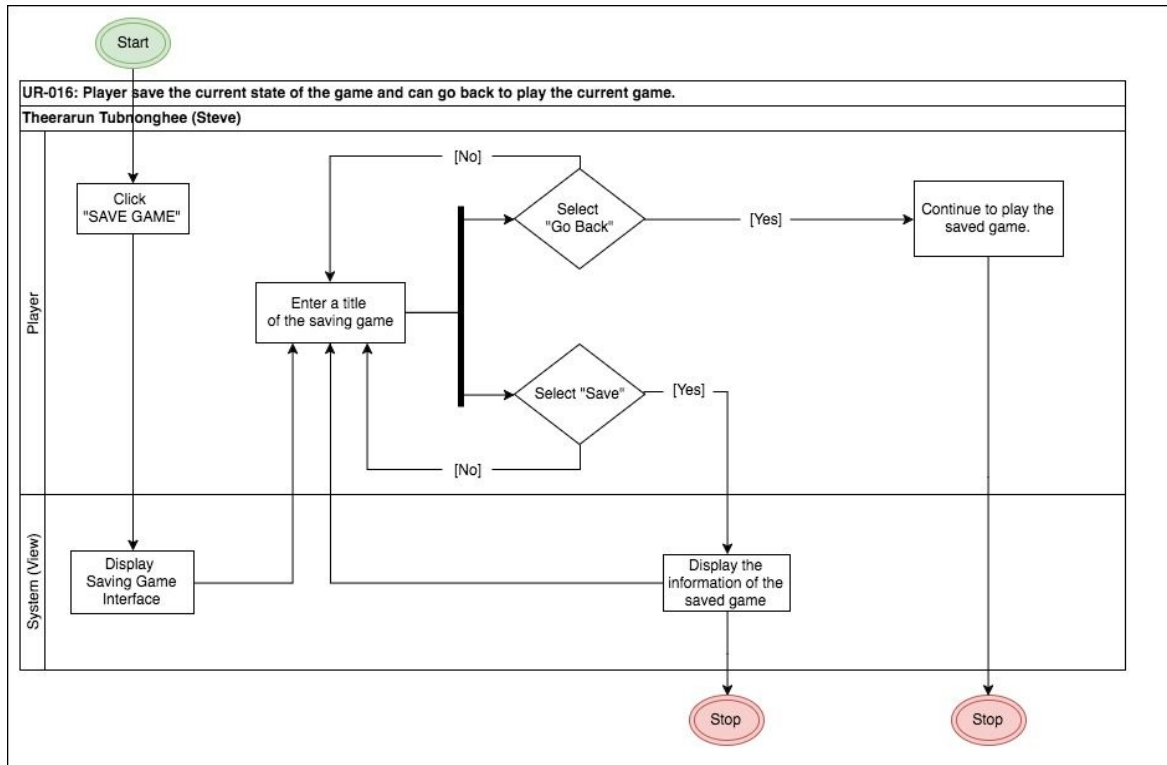
UR-017 : "Load Game" : Player loads saved game



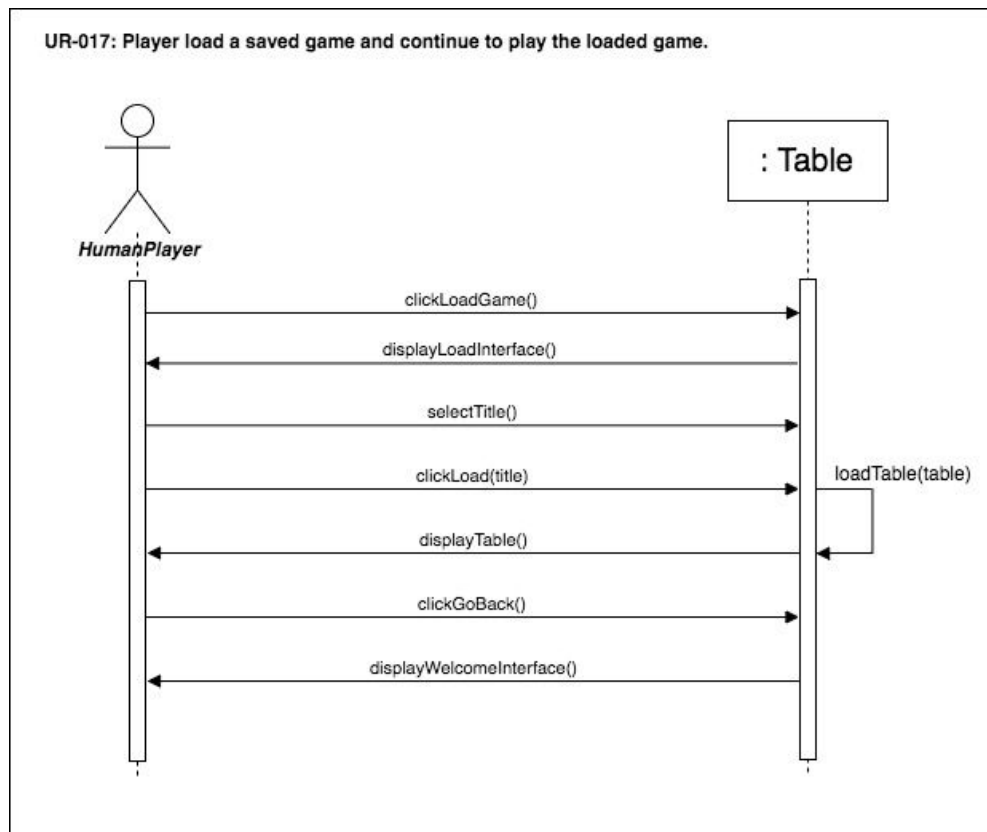
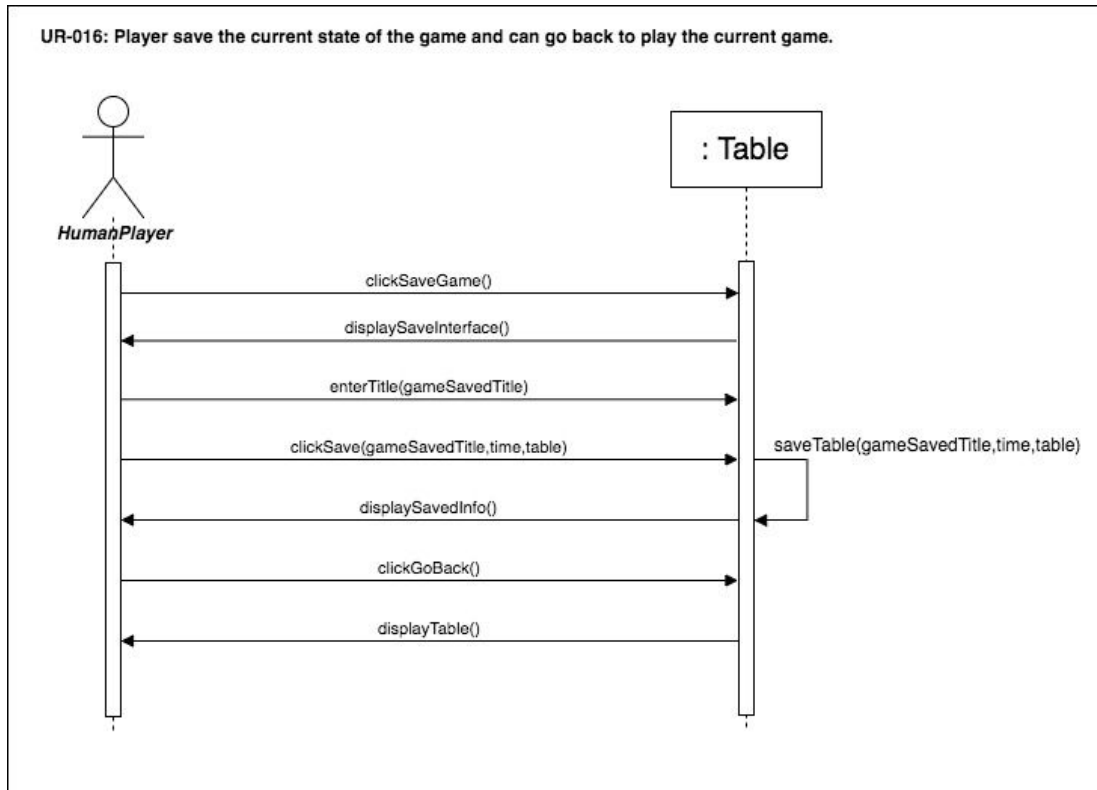
UR-018 : "Starting Chips" : Player specifies amount of starting chips to play.




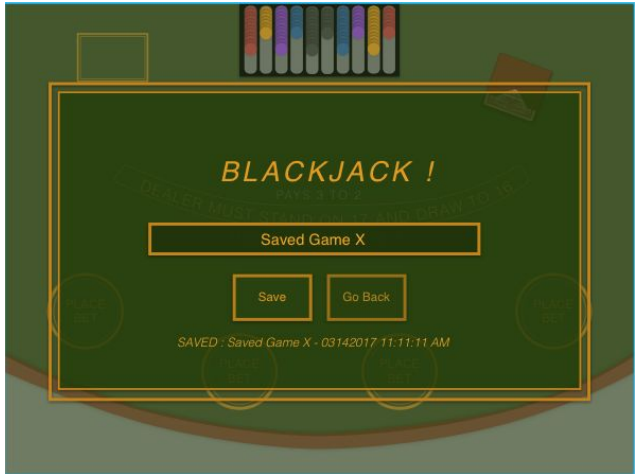
Activity Diagrams:




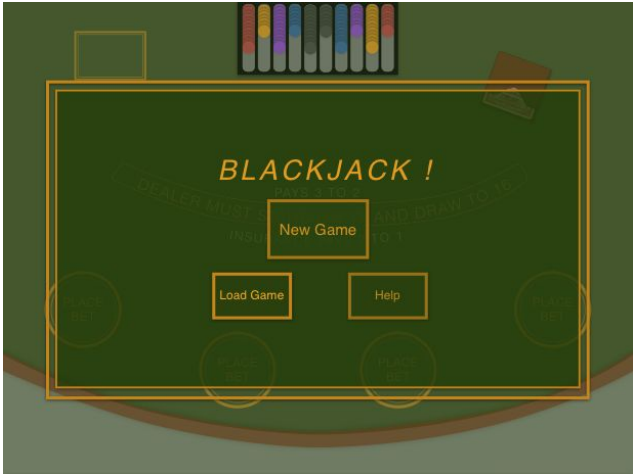
User Interactions:

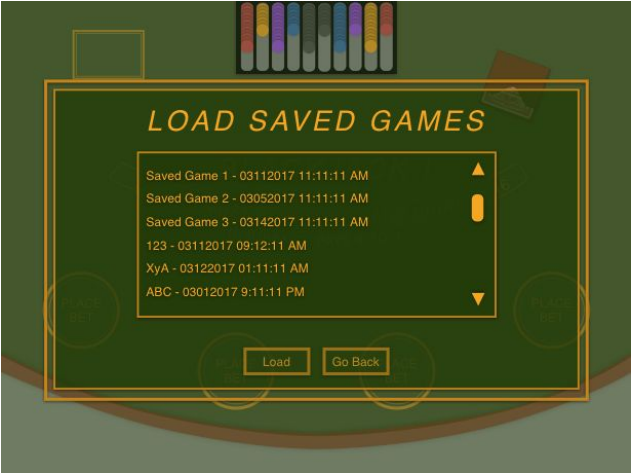



UI Mockup:

Use Case ID:	UR-016		
Use Case Name:	Player can save the current state of game and continue to play.		
Description:	Player can choose to save the current state of the game during the placing bet state, and Player can navigate back to play the current game.		
Flow of Interactions:			
		Scenario	User Interface Response
	1	Player is in the placing bet state	
	2	Player clicks SAVE button.	
	3	Table display Saving Interface	
	4	Player enters the name of saving state, and hit "Save Game".	

	5	Player hits “Go Back”	
--	---	-----------------------	--

Use Case ID:	UR-017		
Use Case Name:	Player can continue the saved game.		
Description:	Player can select to load a saved game, and continue to play the game..		
Flow of Interactions:			
		Scenario	User Interface Response
	1	Player is in the welcome interface.	

	3	Table display Load Saved Game Interface	
	4	Player selects a saved game from the saved list.	
	5	Player clicked “Load”	
	6	Player clicked “Go Back”	