**Use Case 1:** Hosting a game.

**Context of use:** Host player creates and joins a game lobby on the server which other players can join.

**Level:** User-goal

**Primary Actor:** Host player

**Precondition:** Must have an active connection to the server.

**Success End Condition:** Game has been started, host player joined the game and has the join key, server is ready for others players to join.

**Failed End Protection:** Game will not be created, host player notified and will remain at main menu.

**Trigger:** User decides to host a game.

**Main Success Scenario**

1. Host player clicks “Host Game” button.
2. System displays text boxes for time out limit, NSID and # of Players.
3. Host player enters the information into the newly appeared text boxes.
4. Host player clicks “Create” button.
5. System connects to the server.
6. System creates a game lobby on the server using the entered information.
7. Host player receives a signature key which other players will use to join the game.
8. Host player is moved to the in-game screen.

**Extensions**

4a. Host player provided invalid input in the text boxes: Display error message pop-up and prompt for server information.

5a. Host player cannot connect to the server: Display error message pop-up and return host player to the main menu.

**Use Case 2:** Joining a game.

**Context of use:** Joining player joins an existing game lobby.

**Level:** User-goal

**Primary Actor:** Joining player

**Preconditions:** Must have an active connection to the server, must have a valid join key to an active game.

**Success End Condition:** Joining player connects to the game.

**Failed End Protection:** Game will not be joined and will remain at main menu.

**Trigger:** User decides to join a game.

**Main Success Scenario**

1. Joining player clicks “Join Game” button.
2. System displays text box for the join key.
3. Joining player enters the information into the newly appeared text box.
4. Joining player clicks “Join” button.
5. Joining player connects to the game lobby on the server using the entered information.
6. Joining player is moved to the in-game screen.

**Extensions**

3a. Joining player provided invalid input in the text box: Display error message pop-up and return joining player to the main menu.

5a. Joining player cannot connect to the server: Display error message pop-up and return joining player to the main menu.

**Use Case 3:** Reading How to Play.

**Context of use:** Player is shown the instructions and rules.

**Level:** User-goal

**Primary Actor:** Player

**Precondition:** None

**Success End Condition:** Player can read the instructions and rules.

**Failed End Protection:** None

**Trigger:** User decides to read the instructions and rules.

**Main Success Scenario**

1. Player clicks “How to Play” button.
2. System displays pop-up with clickable section headers.
3. Player clicks a specific section header.
4. Player is shown instructions pertaining to the clicked header.
5. Player clicks “Close” button.
6. System closes the pop-up instruction window.

**Extensions**

None

**Use Case 4:** Play a card.

**Context of use:** Active Player plays a card in-game.

**Level:** User-goal

**Primary Actor:** Active player

**Preconditions:** Must be player’s turn.

**Success End Condition:** Card from player’s hand is added to a firework, a new card is dealt and turn is ended.

**Failed End Protection:** Card is discarded and a red launch counter is removed.

**Trigger:** User decides to play a card.

**Main Success Scenario**

1. Active player clicks “Play a card” button.
2. Active player clicks their card of choice.
3. Active player’s card is removed from hand and placed onto the appropriate firework pile.
4. Active player is dealt another card face down.
5. Inactive players are notified of what card is played and what card is dealt.

**Extensions**

3a. Card cannot properly be played: Card is discarded and a red launch counter is removed.

4a. Deck is empty: Active player cannot draw new card.

**Use Case 5:** Discard a card.

**Context of use:** Active Player discards a card in-game.

**Level:** User-goal

**Primary Actor:** Active player

**Preconditions:** Must be player’s turn, all 8 info tokens cannot be in play.

**Success End Condition:** Card from player’s hand is added to a discard pile, a new card is dealt, an info token is returned and turn is ended.

**Failed End Protection:** None

**Trigger:** User decides to discard a card.

**Main Success Scenario**

1. Active player clicks “Discard a card” button.
2. Active player clicks their card of choice.
3. Active player’s card is removed from hand and placed onto the discard pile.
4. Active player is dealt another card face down.
5. Info token is returned to play.
6. Inactive players are notified of what card is discard and what card is dealt.

**Extensions**

6a. Deck is empty: Active player cannot draw new card.

**Use Case 6:** Give colour info.

**Context of use:** Active Player give colour info to an inactive player in-game.

**Level:** User-goal

**Primary Actor:** Active player

**Preconditions:** Must be player’s turn, must be at least 1 info token in play.

**Success End Condition:** Colour info is given to specific player, info token is spent and turn is ended.

**Failed End Protection:** None

**Trigger:** User decides to give colour info to another player.

**Main Success Scenario**

1. Active player clicks “Give colour info” button.
2. Active player hovers with the mouse over a card in any inactive players’ hand.
3. Cards in the same player's hand which have the same colour as the one moused over are raised. Cards for which that colour information has already be shared do not raise.
4. Active player clicks moused over card.
5. Inactive player whose card is clicked is given colour info about all cards that were raised.
6. Inactive players whose cards weren’t clicked are notified of what card info was given.
7. System removes one info token from play.

**Extensions**

3a. No cards raise: Colour info for moused over card or others that match it cannot be given.

**Use Case 7:** Give number info.

**Context of use:** Active Player give number info to an inactive player in-game.

**Level:** User-goal

**Primary Actor:** Active player

**Preconditions:** Must be player’s turn, must be at least 1 info token in play.

**Success End Condition:** Number info is given to specific player, info token is spent and turn is ended.

**Failed End Protection:** None

**Trigger:** User decides to give number info to another player.

**Main Success Scenario**

1. Active player clicks “Give number info” button.
2. Active player hovers with the mouse over a card in any inactive players’ hand.
3. Cards in the same player's hand which have the same number as the one moused over are raised. Cards for which that number information has already be shared do not raise.
4. Active player clicks moused over card.
5. Inactive player whose card is clicked is given number info about all cards that were raised.
6. Inactive players whose cards weren’t clicked are notified of what card info was given.
7. System removes one info token from play.

**Extensions**

3a. No cards raise: Number info for moused over card or others that match it cannot be given.

**Use Case 8:** Looking at discard pile.

**Context of use:** Player is shown all the cards in the discard pile.

**Level:** User-goal

**Primary Actor:** Player (Both active and inactive)

**Precondition:** None

**Success End Condition:** Player can look at discard pile.

**Failed End Protection:** None

**Trigger:** User decides to look at discard pile.

**Main Success Scenario**

1. Player clicks “Discard Pile” button.
2. System displays pop-up containing all cards in the discard pile.
3. Player clicks “Close” button.
4. System closes the pop-up instruction window.

**Extensions**

None

**Use Case 9:** Player drops out.

**Context of use:** Player exits program while in a game.

**Level:** User-goal

**Primary Actor:** Player (Both active and inactive)

**Precondition:** None

**Success End Condition:** Player’s program is closed, other players are returned to main menu.

**Failed End Protection:** None

**Trigger:** User decides to exit game.

**Main Success Scenario**

1. Player clicks “Close”.
2. System closes the players program.
3. Other players are moved back to main menu, game is ended, server is closed and other players are shown the score screen.
4. Other players click “Close” on score screen.

**Extensions**

None

**Use Case 10:** Player exits program

**Context of use:** Player exits program while not in a game.

**Level:** User-goal

**Primary Actor:** Player

**Precondition:** None

**Success End Condition:** Player’s program is closed.

**Failed End Protection:** None

**Trigger:** User decides to exit program.

**Main Success Scenario**

1. Player clicks “Close”.
2. System closes the players program.

**Extensions**

None