

# Use Case Descriptions Checkers

## Use Case: Checkers Game

- Iteration: 1
- Primary Actor: Player 1, Player 2
- Goal in Context: Start and manage a complete game of Checkers between two players.
- Preconditions: None.
- Trigger: A player initiates the game.
- Scenario:
  - The system initializes the game environment and displays the board.
  - The system prepares each player's pieces in starting positions.
  - The game is ready for players to begin taking turns.
- Post Conditions: The game is ready to proceed with players taking turns and making moves.
- Exceptions:
  - If the system fails to initialize, an error message is shown, and the game cannot start.
- Priority: High, as it is essential for gameplay initiation.
- When Available: Within its 1st iteration.
- Frequency of Use: Once per game session.
- Channel to Actor: Player interacts through the game's user interface.
- Secondary Actors: N/A
- Channel to Secondary Actors: N/A
- Open Issues: None

## Use Case: Setup Board

- Iteration: 1
- Primary Actor: System
- Goal in Context: Arrange all pieces on the board for the beginning of the game.
- Preconditions: The Checkers Game has been initiated.
- Trigger: The system initiates board setup as part of game start.
- Scenario:
  - The system places each player's pieces on the designated starting positions.
  - The board is visually updated to show the initial setup.
- Post Conditions: All pieces are correctly positioned on the board for the game to begin.
- Exceptions:
  - If the board setup fails, an error message is displayed, and the game does not start.
- Priority: High, as it is crucial for starting the game.
- When Available: Within its 1st iteration.
- Frequency of Use: Once per game session.
- Channel to Actor: System controls the setup directly.
- Secondary Actors: N/A
- Channel to Secondary Actors: N/A
- Open Issues: None

## Use Case: Game Controller

- Iteration: 1
- Primary Actor: System
- Goal in Context: Manage turn-taking and special game events.
- Preconditions: The game is in progress.
- Trigger: A player ends their turn.
- Scenario:
  - The system ends the current turn.
  - The system enforces any additional game rules (e.g., king creation, elimination).
- Post Conditions: The game continues in accordance with the rules.
- Exceptions:
  - System error in handling turn or rule enforcement.
- Priority: High, essential for game management.
- When Available: Within its 1st iteration.
- Frequency of Use: Continuously throughout the game.
- Channel to Actor: System-managed.
- Secondary Actors: N/A
- Channel to Secondary Actors: N/A
- Open Issues: None

## Use Case: Select Piece

**Iteration:** 1

**Primary Actor:** Player

**Goal in Context:** Allows the player to select a piece to move on their turn.

**Preconditions:** The game must be in progress, and it must be the player's turn.

**Trigger:** The player clicks on or selects a piece on the board.

**Scenario:**

1. The player selects a piece by clicking or entering its coordinates.
2. The system highlights or marks the selected piece.
3. The system verifies that the selected piece belongs to the current player.

**Post Conditions:** The piece is highlighted, and the player can now select a destination.

**Exceptions:**

- The selected piece does not belong to the player, and an error message is shown.

**Priority:** High priority, as it is essential for gameplay.

**When Available:** Within its 1st iteration.

**Frequency of Use:** Multiple times per player turn.

**Channel to Actor:** Player interacts directly with the board.

**Secondary Actors:** N/A

**Channel to Secondary Actors:** N/A

**Open Issues:** None

## Use Case: Move Piece on click

**Iteration:** 1

**Primary Actor:** Player

**Goal in Context:** Allows the player to move a selected piece to a new position on the board.

**Preconditions:** A piece has been selected by the player, and it is a valid piece to move.

**Trigger:** The player clicks on or selects a destination square on the board.

**Scenario:**

1. The player selects a destination for the selected piece.
2. The system checks if the move is valid according to the rules of Checkers. ○ If valid, the piece is moved to the new position.
3. The board is updated to show the new position of the piece.

**Post Conditions:** The piece has been successfully moved to the new position.

**Exceptions:**

- The move is invalid, and an error message is displayed.

**Priority:** High priority, as it is essential for gameplay.

**When Available:** Within its 1st iteration.

**Frequency of Use:** Multiple times per game, depending on player moves.

**Channel to Actor:** Player interacts directly with the board.

**Secondary Actors:** N/A

**Channel to Secondary Actors:** N/A

**Open Issues:** None

## Use Case: Eliminate Piece

- **Iteration:** 1
- **Primary Actor:** Player
- **Goal in Context:** Allows the system to remove an opponent's piece when a capture move is made.
- **Preconditions:** The current player has made a valid capture move.
- **Trigger:** The player completes a capture move.
- **Scenario:**
  - The system identifies the opponent's piece that is being captured.
  - The captured piece is removed from the board.
  - The board state is updated and displayed to both players.
- **Post Conditions:** The opponent's captured piece is removed from the board.
- **Exceptions:**
  - The system fails to remove the piece due to an error.
- **Priority:** High priority, as it impacts the game's progress and rules.
- **When Available:** Within its 1st iteration.
- **Frequency of Use:** Multiple times per game, depending on capture moves.
- **Channel to Actor:** Game Controller updates the board.
- **Secondary Actors:** System
- **Channel to Secondary Actors:** N/A
- **Open Issues:** None

## Use Case: Switch Player

- **Iteration:** 1
- **Primary Actor:** Player
- **Goal in Context:** Allows the system to alternate turns between players.
- **Preconditions:** The current player has completed their turn.
- **Trigger:** The player's turn is over (either after a move or capture).
- **Scenario:**
  - The system verifies that the player's turn is complete.
  - The active player is switched to the other player.
  - The system updates the game state to reflect the new player's turn. ○ The new player is notified that it is now their turn.
- **Post Conditions:** The turn is switched to the other player.
- **Exceptions:**
  - The system fails to switch the player due to an error.
- **Priority:** High priority, as it is critical for turn-based gameplay.
- **When Available:** Within its 1st iteration.
- **Frequency of Use:** Multiple times per game as turns alternate.
- **Channel to Actor:** Game Controller handles turn management.
- **Secondary Actors:** System
- **Channel to Secondary Actors:** N/A
- **Open Issues:** None

## Use Case: End Game

- **Iteration:** 1
- **Primary Actor:** Player
- **Goal in Context:** Ends the game when a win/loss condition is met or if a player forfeits.
- **Preconditions:** One player has no remaining pieces or legal moves, or a player chooses to forfeit.
- **Trigger:** A win/loss condition is met or a player forfeits.
- **Scenario:**
  - The system checks the board state and verifies if a win/loss condition is met. ○ The game state is updated to GAME\_OVER.
  - The system determines and displays the winner.
  - The end-game screen is displayed with options to "Play Again" or "Exit."
- **Post Conditions:** The game ends, and the result is displayed.
- **Exceptions:**
  - The system fails to end the game due to an error.
- **Priority:** High priority, as it is essential for game completion.
- **When Available:** Within its 1st iteration.
- **Frequency of Use:** Once per game.
- **Channel to Actor:** Game Controller manages end-game logic.
- **Secondary Actors:** System
- **Channel to Secondary Actors:** N/A
- **Open Issues:** None



## Use Case: Create King

- **Iteration:** 1
- **Primary Actor:** Player
- **Goal in Context:** Promotes a piece to a king when it reaches the opposite end of the board.
- **Preconditions:** A piece reaches the last row on the opponent's side during a move.
- **Trigger:** The player moves a piece to the last row.
- **Scenario:**
  - The system identifies that the piece has reached the opponent's last row.
  - The piece is promoted to a king, with a visual indication.
  - The board state is updated to reflect the promotion.
- **Post Conditions:** The selected piece is promoted to a king with enhanced movement.
- **Exceptions:**
  - The system fails to promote the piece due to an error.
- **Priority:** Medium priority, as it enhances gameplay but is not essential.
- **When Available:** Within its 1st iteration.
- **Frequency of Use:** As needed during gameplay.
- **Channel to Actor:** Game Controller handles promotion.
- **Secondary Actors:** System
- **Channel to Secondary Actors:** N/A
- **Open Issues:** None

## Use Case: Play Again Button

- **Iteration:** 1
- **Primary Actor:** Player
- **Goal in Context:** Allows the player to restart the game with a fresh board after the game has ended.
- **Preconditions:** The game is over, and the "Play Again" button is available on the end-game screen.
- **Trigger:** The player clicks the "Play Again" button.
- **Scenario:**
  - The player selects the "Play Again" button.
  - The system resets the game state and initializes a new board.
  - The board is reinitialized, and the players are notified of the new game start.

**Post Conditions:** A new game begins with the board reset.

- **Exceptions:**
  - The system fails to reset the board due to an error.
- **Priority:** Medium priority, as it enhances user experience.
- **When Available:** Within its 1st iteration.
- **Frequency of Use:** Once per session, if the player wants to replay.
- **Channel to Actor:** User interface for player interaction.
- **Secondary Actors:** N/A
- **Channel to Secondary Actors:** N/A
- **Open Issues:** None

## Use Case: Exit

- **Iteration:** 1
- **Primary Actor:** Player
- **Goal in Context:** Allows the player to exit the game after it has ended.
- **Preconditions:** The game is over, and the "Exit" button is available.
- **Trigger:** The player clicks the "Exit" button.
- **Scenario:**
  - The player selects the "Exit" button.
  - The application closes or returns to the main menu.
- **Post Conditions:** The game session is terminated.
- **Exceptions:**
  - The application fails to exit properly.
- **Priority:** Low priority, as it provides a way to end the session.
- **When Available:** Within its 1st iteration.
- **Frequency of Use:** Once per session, if the player wants to end the game.
- **Channel to Actor:** User interface for player interaction.
- **Secondary Actors:** N/A
- **Channel to Secondary Actors:** N/A
- **Open Issues:** None