

# Fully Dressed Use Case

## Set Up a Game

**Primary Actor:** Player

**Stakeholders and Interests:**

- *Player* - wants to play a new game or continue a previous game.

**Preconditions:**

None

**Success Guarantee (Postconditions):**

- The player is aware of their color.
- The game board is visible.
- The game directs which player makes a move.

**Main Success Story:**

1. The user requests a game session
2. The system provides the user with the opportunity to start a game.
3. The user selects to start a new game session.
4. The system begins the setup process and displays the various options (Number of players, difficulty level, color palette) the user needs to select from.
5. The user selects the various options provided by the system. *[Alt1: User tries to continue without selecting the provided options]*
6. The system provides the user with the opportunity to start the game.
7. The user confirms that they want to start the game.
8. The system provides the user the opportunity to play a turn. Use case end.

**Alternative Scenarios:**

*[Alt1: User tries to continue without selecting the provided options]*

1. The system informs the user that they have not selected the required options.
2. Flow resumes at Main success Scenario step 4.

**Exceptions:**

None

**Special Requirements:**

- Colours and sizes of text fonts used must provide - or be able to provide - for the visually impaired (e.g. colour blindness).

**Open Issues:**

None

## Take a Turn

**Primary Actor:** Player

**Stakeholders and Interests:**

- *Player* - wants to play a new game or continue a previous game.

**Preconditions:**

- The player is aware of their color.
- The game board is visible.
- The game directs which player makes a move.

**Success Guarantee (Postconditions):**

- If a player is able to make a move and the game updates the game state.

**Main Success Story:**

1. The system provides the user(s) with information (player number and the color) about the active player.
2. The user selects a stack or a piece of their color to move. *[Alt 1 - The user selects a piece or stack of a different player] [Alt 2 - The user plays a reserved piece]*
3. The system indicates the piece or stack which the user selected.
4. The user chooses where to move the selected piece or stack. *[Alt 3 - The user unselects the previously selected piece or stack]*
5. The systems check the validity of the user's move and update the game state. *[Alt4: The move is invalid.]*
6. The system provides the next player to play their turn. Use case ends. *[Alt5: The game is over]*

**Alternative Scenarios:**

*[Alt 1 - The user selects a piece or stack of a different player]*

1. The system notifies the user that it's invalid.
2. Flow resumes at Main success Scenario step 2.

*[Alt 2 - The user plays a reserved piece]*

1. Flow resumes at Main success Scenario step 4.

*[Alt 3 - The user unselects the previously selected piece or stack]*

1. Flow resumes at Main success Scenario step 2.

*[Alt4: The move is invalid.]*

1. The system informs the user that their move is invalid.
2. Flow resumes at Main success Scenario step 2.

*[Alt5: The game is over]*

1. The system informs the users about the game state (If a user won or the game is a draw). Use case ends.

**Exceptions:**

None

**Special Requirements:**

- Colours and sizes of text fonts used must provide - or be able to provide - for the visually impaired (e.g. colour blindness).

**Open Issues:**

None