New Fully Dressed Use Case: Select a piece

Primary Actor:

The human playing the game

Stakeholders and Interests:

The user

Preconditions:

The human player must be the current player to take a turn

Postconditions:

The user's piece must be selected

Main Success Scenario:

1. The system ensures the current player has pieces left **(Exception: Player has run out of pieces, and is now out of the game)**
2. The system tells the user it is their turn to make a move
3. The user selects a piece to lay from the 'Select piece' drop-down menu
4. The system updates the human player's piece choice selection
5. The user clicks a block on the board
6. The system shows an outline of where the piece is currently ready to be placed, indicating which blocks will be filled.
7. The user moves and rotates the piece, to find a suitable location to place it
8. The user clicks the "place piece" button
9. The system checks if the piece is in the correct place(**Alternative flow: Selected piece did not fit**)
10. The system places the piece on the board

Alternative Flows:

Selected piece did not fit

1. The user selects a new piece type from a drop-down menu
2. Continue from step 7 on the main scenario

Exceptions:

Player has run out of pieces, and is now out of the game

(This exception starts from step 1 of the main scenario)

1. The system has detected the player has run out of pieces
2. The system marks this player as “out of the game” so they cannot take more turns

Special Requirements

Should the pieces be changeable by pressing keys?

Open Issues

Will the user select pieces by only text names, or will there be reference images?

Should there be a side panel of images to click, or just a simple drop-down menu?

Fully Dressed Use Case: Change color-blind settings

Primary Actor:

The player

Stakeholders and Interests:

The human player

The AI players

Preconditions:

The user must be on the main menu screen.

Postconditions:

The player successfully chooses an option that is best suited for their eyesight.

Main Success Scenario:

1. The user starts the program
2. The system shows the user the main page, including radio buttons that shows two color-blind settings, Yes and No **(Alternative flow: User doesn't turn color-blind mode on)**
3. The user selects the "Yes" radio button to turn color-blind mode on
4. The systems changes the color of the blocks in the 'Select your color' menu (**Exception: Choose a different color-blind setting**)

Alternative Flow:

User doesn't turn color-blind mode on

(*The alternative flow starts at line 2 in the main scenario)*

1. The system selects the "NO" radio button by default for color-blind setting
2. The system makes no changes to the 'Select your color' menu, and allows the user to continue setting their game up like normal.

Exceptions

Choose a different color-blind setting

(*The exception starts at line 4 in the main scenario)*

1. The user decides it is hard to tell the difference between the colors in the ‘Select a color’ drop-down box
2. The user uses radio buttons to change the color-blind settings
3. The systems changes the color of the blocks in the ‘Select color’ menu

Special Requirements

Should this option be changeable at any time?

Open Issues

Should this cover all kinds of color blindness?
Should it change the piece colors, change the pieces to symbols, or possibly change all the colors to grey-scale?