# 2a. Iteration – Revised Use Case Description

**Change Game Color Theme:**

| | |
|---|---|
| Primary Actors | Human players |
| Stakeholders and Interests | Human players, Developers, Testers |
| Preconditions | Preconditions for changing the game color theme are:<br>1. The user is able to run and start the program as intended. This means some sort of menu or user interface is displayed.<br>2. The user interface shown to the player must be functional. If all is shown is a window and buttons that don't work it wouldn’t be possible to change the theme.<br>3. The user interface must have a dedicated and appropriate spot for this option. An appropriate spot would be something like the settings menu. But also have some sort of interaction method such as a button or a drop down menu that will actually change the theme when it's implemented and the user provides the action.<br>4. Therefore another precondition is that the user is able to navigate through the menus to that interaction method.<br>5. You can't change the theme if there is only one theme to choose from. So a precondition is that there must be more than one themes to choose from. |
| Postconditions | The successful implementation of this use case comes from the player able to change the appearance of the user interface by selecting a desired theme.<br>1. After the user has selected a desired theme the postcondition is that the color background for the buttons and background of the frame all implemented in the main menu is changed according to the users selection of the theme.<br>2. The selection of the theme also properly changes the appearance of the board and the pieces when the game is initiated.<br>3. The user is still on the main menu screens and ready to start a game or change setting options again. |

| | |
|---|---|
| Main Success Scenario | 1. The player runs the program and it starts without any complications. A clear and functional main menu interface is available to the user.<br>2. The main menu offers a settings button that when pressed navigates the user to another page where the option to change theme is presented.<br>3. A drop down menu shows the available options for the user to decide which one they would like to use.<br>4. The selection is made and the appropriate option is chosen by the user. After the action is performed the appearance of the menu and the board is changed according to the users selection. |
| Alternative Flows | 1. An alternative flow happens if the user is okay with the default theme and does not want to change the theme. This avoids changing the theme path and puts the user on being ready to imitate the game |
| Exceptions | An exception could occur when the user selected a theme for example yellow but instead of the expected yellow it's grey.<br>Another exception is when the theme is selected it does not apply to the whole interface for some reason.<br>The game could crash while trying to update colors for frame backgrounds. |
| Special Requirements | The human player may choose the color theme again before they start the game. |

**<u>Return to Initial Interface(Main Menu)/Implementing a back button:</u>**

| | |
|---|---|
| Primary Actors | Human players |
| Stakeholders and Interests | Human players, Developers, Testers |
| Preconditions | Preconditions for implementing a back button for navigation are:<br>1. The program must start and show an initial screen or the main menu. The main menu will not have a back button.<br>2. Functions such as the settings and initiating a game have different pages/panels  from the main menu.<br>3. Before navigating backwards the going forward has to be logical and intuitive. For example the new game button should not go to settings. |

| | |
|---|---|
| Postconditions | The successful implementation of this use case comes from the player able to navigate forward and backwards through the user interface.<br>1. New Game button is pressed from the main menu.<br>2. More settings appear like difficulty and number of players.<br>3. Instead of initiating the game the player has forgotten to turn hints off. Player does not have to restart the whole program but can use the back button to go back to main menu.<br>4. Similar when players has entered the settings page the user can go back to menu and initiate the game with the settings applied. |
| Main Success Scenario | 1.User has started the game and main menu is displayed with different options for the player to interact with.<br>2. Player is able to press the settings button or the New game and it should lead to the correct pages.<br>3. The player is able to change the settings according to their preferences and is able to use the back button to the main menu.<br>4. The new game page features other options like difficulty and player numbers which is implemented later. From here the player can decide to use the back button that goes back to the main menu  or initiate the game that leads to another page which is also implement later. |
| Alternative Flows | 1. The player has already started the game and almost halfway through the game and wants to access the settings to change something. This flow will include saving the game feature which hopefully  is implemented in the coming iterations. |
| Exceptions | 1. The game crashes or the back button freezes. |
| Special Requirements | 1. The human player can quit the game even when it is the other player’s turn. |