

## **Use Cases**

### **Use Case 1**

**Use case:** StartGame

**Primary actor:** Game player (user)

**Goal in context:** To start playing the game.

**Preconditions:** The game has not yet been started, and the game has loaded in all required data and assets.

**Trigger:** The user decides to play the game by pressing the “play/start” button.

**Scenario:**

1. The user opens the game and views the main menu of the game.
2. The user presses on the “play/start” button.
3. The game screen is displayed and the user starts playing.

**Exceptions:**

1. Game is paused: User presses “continue”.
2. Game is over: User presses “restart”.

**Priority:** Essential, must be implemented in order to start the game.

**When available:** First increment.

**Frequency of use:** Frequent, whenever the user wants to play the game.

**Channel to actor:** Through a window on the computer screen.

**Secondary actors:** None

**Channels to secondary actors:** None

**Open issues:** None

## **Use Case 2**

**Use case:** ExitGame

**Primary actor:** Game player (user)

**Goal in context:** To quit gameplay and return to the home screen/main menu.

**Preconditions:** The game is currently in play and is not paused.

**Trigger:** The user decides to return to the main menu.

**Scenario:**

1. The user presses the “escape” key or “pause” button to pause the game to bring up the pause menu.
2. The user views the pause menu, where the “quit” button is.
3. The user presses the “quit” button and returns to the main menu.

**Exceptions:** None.

**Priority:** Medium, must be implemented for the feature to work, but the game is still able to run without it. The user could always go back to the main menu by completely closing the game and starting it up again.

**When available:** Second increment, while the user is playing the game.

**Frequency of use:** Frequent, whenever the user has started the game.

**Channel to actor:** Through a window on the computer screen.

**Secondary actors:** None

**Channels to secondary actors:** None

**Open issues:** None

### **Use Case 3**

**Use case:** MovePlayer

**Primary actor:** Game player (user)

**Goal in context:** To move the main character when the game is in play, while also getting rewards and avoiding enemies or punishments.

**Preconditions:** Game is currently being played, and enemies, rewards, and barriers are in place.

**Trigger:** The user decides to move on the board in a given direction.

**Scenario:**

1. The user presses an “arrow” key or one of the “WASD” keys on the keyboard.
2. The user’s character moves in that associated direction.

**Exceptions:**

1. User reaches a reward: the user receives points that contribute towards the overall score.
2. User hits an enemy: the game is over, and the user is returned to the main menu of the game.
3. User hits a punishment: the user loses points from the overall score and continues to try and complete the game.
4. User hits a barrier: the user must reverse and move in a path that is available, and continues to complete the game.

**Priority:** Essential, must be implemented in order to play and win/lose the game.

**When available:** Second increment.

**Frequency of use:** Extremely frequent, whenever the user is playing the game. If the user continually holds down a key to move in a certain direction, this use case will continually trigger 60 times per second (the FPS of the game).

**Channel to actor:** Through a window on the computer screen.

**Secondary actors:** None

**Channels to secondary actors:** None

**Open issues:** None

#### **Use Case 4**

**Use case:** GameOver

**Primary actor:** Game player (user)

**Goal in context:** Stop the gameplay and display “play again” and “return to home screen” buttons.

**Preconditions:** The game is currently being played.

**Trigger:** The user’s score goes below zero from an enemy or punishment, or the user completes the final level.

**Scenario:**

1. The game stops the current game session being played, destroying the board and all entities such as the player character, rewards, punishments, and enemies.
2. The game brings up a menu with two buttons, “play again”, and “return to home screen”.
3. The game waits for the user’s input.

**Exceptions:** None.

**Priority:** Essential, must be implemented in order to end the game.

**When available:** Third increment.

**Frequency of use:** Frequent, whenever the user is playing the game.

**Channel to actor:** Through a window on the computer screen.

**Secondary actors:** None

**Channels to secondary actors:** None

**Open issues:** None

### **Use Case 5**

**Use case:** FlipWorld

**Primary actor:** Game player (user)

**Goal in context:** To take the main character go to the “flipped” map and continue playing.

**Preconditions:** User is playing the game.

**Trigger:** The user presses the “F” key on their keyboard to flip the map.

**Scenario:**

1. The game loads in the data for the alternate map for the “flipped” world and generates the new game board.
2. The world “flips” and the user continues playing. The location on the screen of the user, enemies, punishments, and rewards will remain the same. However, the locations of walls/barriers, and hazards such as holes or lava pools will change.

**Exceptions:**

1. The corresponding location of the player on the “flipped” map is a wall/barrier. The player is not allowed to be in a wall/barrier, so the player is not allowed to flip and displays a notification informing the user of this.
2. If the player flips onto a hazard such as a lava pool or hole, the player will respawn back at the start door of the current level, and the game will return them to the original state when they started the level.

**Priority:** Medium, must be implemented for the feature to work, but the game is still able to run without it.

**When available:** When the user is playing the game.

**Frequency of use:** Sometimes, when the user triggers a world-flip.

**Channel to actor:** Through a window on the computer screen.

**Secondary actors:** None

**Channels to secondary actors:** None

**Open issues:** None

### **Use Case 6**

**Use case:** ChangeCharacter

**Primary actor:** Game player (user)

**Goal in context:** To change the main character icon from the one currently set.

**Preconditions:** User has opened the game and is on the main menu.

**Trigger:** The user decides to select a new character from the settings page.

**Scenario:**

1. The game has been opened and is on the main menu page.
2. The user selects the “settings” button on the screen, and the settings page opens up.
3. The user then selects the “change player” button, and different main character options appear.
4. The user selects the character they want, then confirms their decision with the “confirm changes” button.

**Exceptions:** None

**Priority:** Low, great interactive feature to have but does not impact game’s functionality whatsoever.

**When available:** When the user is on the main menu.

**Frequency of use:** Sometimes, when the user decides to change their character.

**Channel to actor:** Through a window on the computer screen.

**Secondary actors:** None

**Channels to secondary actors:** None

**Open issues:** None

### **Use Case 7**

**Use case:** ChangeVolume

**Primary actor:** Game player (user)

**Goal in context:** To increase or decrease the volume of the game.

**Preconditions:** User has opened the game and is on the main menu.

**Trigger:** The user decides to select a new character from the settings page.

**Scenario:**

1. The game has been opened and is on the main menu page.
2. The user selects the “settings” button on the screen, and the settings page opens up.
3. The user then selects the “change volume” button, and the volume slider appears.
4. The user changes the volume to their desired level, then confirms their decision with the “confirm changes” button.

**Exceptions:** None

**Priority:** Medium, must be implemented for the feature to work, but the game is still able to run without it.

**When available:** When the user is on the main menu.

**Frequency of use:** Sometimes, when the user decides to change their volume.

**Channel to actor:** Through a window on the computer screen.

**Secondary actors:** None

**Channels to secondary actors:** None

**Open issues:** None