Use Cases

Use case overview: Starting the game, Playing the game, Pausing the game, Exiting the game, Collision with a moving enemy, Collision with a score bomb, Collision with a health bomb, Game over, Collecting a Reward, Collecting a heart reward, Collecting a freeze reward, Unfreezing a moving enemy, Moving the hero, Colliding the wall, Validate the input, Setting a new high score, Initialize the high score, Resuming the game, Restarting the game, Entering the exit cell, Blocked exit cell.

- **Use-case**: Starting the game.
- **Primary actor**: The player.
- **Secondary actor**: The game system
- **Goal**: Initialize the game by creating a board and populating it with appropriate objects.
- **Precondition**: The game must be installed and fully configured on the target device.
- **Trigger**: The player decides to launch the game.
- **Scenario**:
    - The player launches the game.
    - The game displays the main menu windows with the following buttons: “Play a new game”, “exit” and a label showing the top score.
    - The player hits the "play a new game" button.
    - The application displays the interface of the game.
- **Exceptions**:
    - If the player clicks the "exit" button in the main menu, then the game closes.—see use-case "Exiting the game".
- **Priority**: Moderate priority.
- **When available**: Second increment.
- **Frequency of use**: Many times

- **Use-case**: Playing the game.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: To finish the game by either winning or losing
- **Precondition**: The board has been populated with appropriate objects, and is awaiting user input.
- **Trigger**: The player decides to play the game by pressing any key.
- **Scenario**:
    - The system displays "press any key to start" message.
    - The player presses any key to start the game.
    - The player uses keyboard "up", "down", "left", "right" to move, collecting all the regular rewards and getting to the end point in order to win the game.

    However, if the player-controlled main character is caught by enemies or either the score or health reduces to zero, then the player loses the game.
    - Once the game finishes, a pop-up window shows displaying statistics about the game, and gives the player the ability to return to the main menu by pressing any key.
    - The player presses any key to return to the main menu.
- **Exceptions**:
    - When the player is playing the game and closes it using the system exit button, —see use case "Exiting the game".
- **Priority**: High priority.
- **When available**: first increment.
- **Frequency of use**: Many times.

- **Use-case**: Pausing the game.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: To pause the game.
- **Precondition**: The game must be running and must be on the game interface.
- **Trigger**: The player decides to pause the game by pressing the “pause” button on the game interface.
- **Scenario**:
    - The player presses the "pause" button.
    - The game pauses and opens the pause window.
- **Exceptions**: N/A.
- **Priority**: Low priority.
- **When available**: Third increment.
- **Frequency of use**: Many times.

- **Use-case**: Exiting the game.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: Exit application.
- **Precondition**: The application/game has been launched and is running.
- **Trigger**: The player decides to close the application either by pressing an exit button in one of the menus or by using the system window exit buttons.
- **Scenario**:
    - Player clicks “exit” or uses the system close buttons.
    - The application is terminated, stopping all related services.
- **Exceptions**: N/A.
- **Priority**: Moderate priority.
- **When available**: Third increment.

- **Frequency of use**: Once in a program life.

- **Use-case**: Collision with a moving enemy.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: To kill the player.
- **Precondition**: The application is running and the game has started.
- **Trigger**: Either the player moves into a cell containing a moving enemy or a moving enemy moves into a cell containing the player.
- **Scenario**:
  - The player and enemy are in the same cell.
  - An animation or message displays indicating the player has died and the game has finished.
- **Exceptions**: N/A.
- **Priority**: Moderate priority.
- **When available**: First increment.
- **Frequency of use**: Once in a game life.

- **Use-case**: Collision with a score bomb.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: Reduce the player's score.
- **Precondition**: The application is running and the game has started.
- **Trigger**: The player moves into a cell containing a score bomb.
- **Scenario**:
  - The player and score bomb are in the same cell.
  - An animation or message displays indicating that the player's score has reduced by a certain amount.
  - The game resumes with updated player statistics and awaits user input.
- **Exceptions**:
  - The player's score drops to zero or becomes negative - see use case "Game over"
- **Priority**: Moderate priority.
- **When available**: First increment.
- **Frequency of use**: Once or many times in a game life.

- **Use-case**: Collision with a health bomb.
- **Primary actor**: The player

- **Secondary actor**: The game system.
- **Goal**: To reduce the player's health.
- **Precondition**: The application is running and the game has started.
- **Trigger**: Either the player moves into a cell containing a health bomb.
- **Scenario**:
    - The player and health bomb are in the same cell.
    - An animation or message displays indicating that the player's health has been reduced by a certain amount.
    - The game resumes with updated player statistics and awaits user input.
- **Exceptions**:
    - The player's health drops to zero or becomes negative - see use case "Game over"
- **Priority**: Moderate priority.
- **When available**: First increment.
- **Frequency of use**: Manys times in a game life.

- **Use-case**: Game over.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: To complete the game by either winning or losing the game.
- **Precondition**: The application is running and the game has started.
- **Trigger**: The player wins or loses the game.
- **Scenario**:
    - The player wins by successfully making it to the end point or loses by taking a lot of damage from enemies.
    - A window shows up showing that the player has lost as well as statistics for that particular game.
- **Exceptions**: N/A.
- **Priority**: Moderate priority.
- **When available**: Second increment.
- **Frequency of use**: Once in a game life.

- **Use-case**: Collecting a Reward.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: To collect one of the required rewards to complete the unlock the exit and increase player score.
- **Precondition**: The application is running and the game has started.
- **Trigger**: The player moves to a cell containing a reward.
- **Scenario**:
    - The player and the reward are in the same cell.

    - The reward disappears and the player is left in the cell.
    - An animation or message indicating that the players score has increased.
    - The players score updates.
    - The rewards counter updates.
    - The game resumes.
- **Exceptions**:
    - The player and a moving enemy enter a reward space at the same time -- see use-case “Collision with a moving enemy”
- **Priority**: Moderate priority.
- **When available**: First increment.
- **Frequency of use**: Zero or more times in a game life.

- **Use-case**: Collecting a heart reward.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: To increase the players health count by one (1).
- **Precondition**: The application is running and the game has started.
- **Trigger**: The player moves to a cell containing a heart reward.
- **Scenario**:
    - The player and the heart reward are in the same cell.
    - The heart reward disappears and the player is left in the cell.
    - An animation or message is displayed indicating that the players health count has increased.
    - The players health count increases.
    - The game resumes.
- **Exceptions**:
    - The player and a moving enemy enter a heart reward space at the same time -- see use-case “Collision with a moving enemy”
- **Priority**: Moderate priority.
- **When available**: First increment.
- **Frequency of use**: Zero or more times in a game life.

- **Use-case**: Collecting a freeze reward.
- **Primary actor**: The player
- **Secondary actor**: The game system.
- **Goal**: To stop the moving enemy’s movement for two (2) ticks.
- **Precondition**: The application is running and the game has started.
- **Trigger**: The player moves to a cell containing a freezereward.
- **Scenario**:
    - The player and the freeze reward are in the same cell.
    - The freeze reward disappears and the player is left in the cell.

    - An animation or message is displayed indicating that the moving enemy can no longer move.
    - The moving enemy stops moving.
    - The game resumes.
- **Exceptions**:
    - The player and a moving enemy enter a freeze reward space at the same time -- see use-case “Collision with a moving enemy”
- **Priority**: Moderate priority.
- **When available**: First increment.
- **Frequency of use**: Zero or more times in a game life.

- **Use-case**: Unfreezing a moving enemy.
- **Primary actor**: The game system.
- **Secondary actor**: The player.
- **Goal**: To resume the moving enemy’s movements..
- **Precondition**: The application is running, the game has started and the moving enemy is frozen.
- **Trigger**: Two (2) ticks have elapsed since the moving enemy was frozen.
- **Scenario**:
    - An animation or message is displayed indicating that the moving enemy will resume movement.
    - The moving enemy resumes movement.
    - The game resumes.
- **Exceptions**:
    - The player collects a freeze reward -- see use-case “Collecting a freeze reward”
- **Priority**: Moderate priority.
- **When available**: First increment.
- **Frequency of use**: Zero or more times in a game life.

- **Use-case**: Moving the hero.
- **Primary actor**: The player.
- **Secondary actor**: The game system.
- **Goal**: To move the main character in the game board.
- **Precondition**: The game interface is launched and the game is started.
- **Trigger**:
    - The player decides to move the main character in the board and press “Up”, “Down”, “Left” or “Right”.
- **Scenario**:
    - The game has started.
    - The player presses “Up”, “Down”, “Left” or “Right” to move.

    - The main character moves in the game board based on the player pressed button.
- **Exceptions**:
    - If the main character faces a wall, barrier or boundary based on player entered direction, the main character will not move. --see use case”Colliding the wall”
    - If the player doesn’t press a valid key, the main character will not move. --see use case “validate the input”.
- **Priority**: Moderate priority.
- **When available**: First increment.
- **Frequency of use**: Many times in game life.

- **Use-case**: Colliding the wall
- **Primary actor**: The player.
- **Secondary actor**: The game system.
- **Goal**: Let the main character move.
- **Precondition**: The game interface is launched and the game is started.
- **Trigger**:
    - The main character collides with a wall, barrier or boundary.
- **Scenario**:
    - The player controlled the main character collides with a wall, barrier or boundary.
    - If there is no wall, barrier or boundary on the player entered direction.
    - After the player enters a direction,the main character moves.
- **Exceptions**:
    - If the player doesn’t press a valid key, the main character will not move. --see use case “validate the input”.
- **Priority**: Moderate priority.
- **When available**: second increment.
- **Frequency of use**: Many times in game life.

- **Use-case**: Validate the input
- **Primary actor**: The player.
- **Secondary actor**: The game system.
- **Goal**: Waiting for valid input.
- **Precondition**: The game interface is launched and the game is started.
- **Trigger**:
    - The player entered the key is not valid.
- **Scenario**:
    - When the game starts, if the player doesn't enter a key or entered key is not valid for a tick.

  - The main character will not move.
- **Exceptions**: N/A.
- **Priority**: Moderate priority.
- **When available**: Third increment.
- **Frequency of use**: Many times in program life.

- **Use-case**: Setting a new high score.
- **Primary actor**: The game system.
- **Secondary actor**: The player.
- **Goal**: Set the highest score to record.
- **Precondition**: The highest score is not zero.
- **Trigger**:
  - After a new game is over.
- **Scenario**:
  - The player loses or wins a game.
  - If the player's score is higher than the current highest score, set the highest score to the value of the player's current score.
- **Exceptions**:
  - If the highest score doesn't exist, then set the highest score to be zero.--see use case "Initialize the high score".
- **Priority**: Moderate priority.
- **When available**: Third increment.
- **Frequency of use**: Many times.

- **Use-case**: Initialize the high score.
- **Primary actor**: The game system.
- **Goal**: Set the highest score to zero.
- **Precondition**: The game hasn't laughed before.
- **Trigger**:
  - The first time the player opens the application.
- **Scenario**:
  - The first time the player opens the application.
  - Set the highest score to zero.
- **Exceptions**: N/A.
- **Priority**: low priority.
- **When available**: Third increment.
- **Frequency of use**: One time.

- **Use-case**: Resuming the game.
- **Primary actor**: The player.
- **Secondary actor**: The game system
- **Goal**: Go back to playing the game after the user paused it for some reason.
- **Precondition**: The application is running, the game has already started and the user decides to pause the game for some reason.
- **Trigger**: The player decides to resume the game.
- **Scenario**:
    - The player paused the game.
    - The player decides to go back to playing the game and the player hits the "Resume Game" button.
    - The pause menu disappears and the game interface shows up again with the game from where they left off.
- **Exceptions**:
    - If the player clicks the "exit" button in the main menu, then the game closes.—see use-case "Exiting the game".
- **Priority**: Moderate priority.
- **When available**: Second increment.
- **Frequency of use**: Many times

- **Use-case**: Restarting the game.
- **Primary actor**: The player.
- **Secondary actor**: The game system
- **Goal**: Initialize the game by creating a board and populating it with appropriate objects.
- **Precondition**: The game is launched and the player pauses the game.
- **Trigger**: The player decides to restart the game.
- **Scenario**:
    - The player pauses the game.
    - The game displays the pause menu.
    - The player hits the "Restart Game" button.
    - It resets the game and takes the player to a new game with a fresh board.
- **Exceptions**:
    - If the player clicks the "exit" button in the main menu, then the game closes.—see use-case "Exiting the game".
- **Priority**: Moderate priority.
- **When available**: Second increment.
- **Frequency of use**: Many times

- **Use-case**: Entering the exit cell.
- **Primary actor**: The player.
- **Secondary actor**: The game system
- **Goal**: Finish the game by entering the exit cell.

- **Precondition**: The game interface is launched and the game is started.
- **Trigger**:
  - The player should have collected all the keys (regular rewards).
  - The player enters the exit cell.
- **Scenario**:
  - The player has collected all the keys (regular rewards) and has decided to enter the exit block and finish the game.
  - The game will be over.
- **Exceptions**:
  - If the player clicks the "exit" button in the main menu, then the game closes.—see use-case "Exiting the game".
  - The player hasn’t collected all the keys. - see use case “Blocked exit cell”.
- **Priority**: Moderate priority.
- **When available**: Third increment.
- **Frequency of use**: One time.

- **Use-case**: Blocked exit cell.
- **Primary actor**: The player.
- **Secondary actor**: The game system.
- **Goal**: Preventing the main character from entering exit cell without collecting all keys.
- **Precondition**: The game interface is launched and the game is started.
- **Trigger**: The player tries to enter into the exit block without collecting all keys.
- **Scenario**:
  - The player tries to move into the exit cell without collecting all keys.
  - Due to incomplete collection of the keys (regular rewards) the hero can’t enter the block and stays in the same block.
- **Exceptions**:
  - If the player clicks the "exit" button in the main menu, then the game closes.—see use-case "Exiting the game".
- **Priority**: Moderate priority.
- **When available**: Second increment.
- **Frequency of use**: Many times.