

Use Case 1: Start the Game

Primary Actor: Player

Goal: Enter the player name and begin the game

Preconditions:

- The game is launched and displayed on the screen

Trigger:

- The player enters their name and selects "Start Game"

Scenario:

1. The player opens the game
2. The system prompts the player to enter a name
3. The player enters a name and clicks "Start Game"
4. The system initializes level 1 and spawns the main character, enemies, and rewards
5. The game begins

Exceptions:

- The player does not enter a name → Default name is assigned
- The game does not initialize → Display an error message

Priority: Essential

When Available: At the start of the game

Frequency of Use: Once per game session

Use Case 2: Move Character

Primary Actor: Player

Goal: Move the main character around the board

Preconditions:

- The game is running
- The player is in control of the character

Trigger:

- The player presses an arrow key

Scenario:

1. The player presses a movement key (Up, Down, Left, Right)
2. The system checks if the move is valid (not a wall or barrier)
3. If valid, the character moves to the new cell
4. If the new cell contains a reward, punishment, or enemy, respective effects are applied

Exceptions:

- The player tries to move into a wall → The character does not move
- The player moves into an enemy → Game over screen is triggered
- The player moves into a punishment → Score decreases

Priority: Essential

When Available: Throughout the game

Frequency of Use: Every time the player moves

Use Case 3: Collect Rewards

Primary Actor: Player

Goal: Increase score by collecting rewards

Preconditions:

- The game is running
- Rewards are present on the board

Trigger:

- The player moves into a cell containing a reward

Scenario:

1. The player moves to a cell containing a reward
2. The system removes the reward from the board
3. The player's score increases
4. After a set duration, the reward respawns

Exceptions:

- Rewards do not disappear after collection → Display an error
- Rewards do not increase score → Display an error
- The reward doesn't respawn → Display an error

Priority: Essential

When Available: Throughout the game

Frequency of Use: Multiple times per level

Use Case 4: Activate Switch

Primary Actor: Player

Goal: Open the exit gate to the next level

Preconditions:

- The switch is placed on the map
- The player has access to the switch
- The player has a certain score

Trigger:

- The player moves into the switch's cell

Scenario:

1. The player moves onto the switch's cell
2. The system registers the switch activation
3. The exit gate for the level opens

Exceptions:

- The switch does not activate → Display an error

Priority: Essential

When Available: Once per level

Frequency of Use: Once per level

Use Case 5: Complete Level

Primary Actor: Player

Goal: Exit the current level and advance to the next one

Preconditions:

- The player has activated the switch
- The exit gate is open

Trigger:

- The player moves to the exit gate

Scenario:

1. The player reaches the exit
2. The system checks if the switch is activated
3. If yes, the player advances to the next level
4. The system loads the next level with increased difficulty

Exceptions:

- The player reaches the exit but hasn't activated the switch → The system prevents level completion
- The player has finished all three levels → 'Congratulations!' is displayed

Priority: Essential

When Available: At the end of each level

Frequency of Use: Once per level

Use Case 6: Encounter Enemy

Primary Actor: Player

Goal: Avoid enemies to stay in the game

Preconditions:

- The game is running
- Enemies are on the board
- Enemies are chasing the player

Trigger:

- The enemy catches up to the player and collides with the player's cell or vice versa.

Scenario:

1. The enemy moves into the player's cell or vice versa.
2. The system registers a collision
3. The game immediately ends
4. The system displays 'Game over'

Exceptions:

- The game does not register the collision → Display an error

Priority: Essential

When Available: Throughout the game

Frequency of Use: As many times as the player encounters an enemy

Use Case 7: Receive Punishment

Primary Actor: Player

Goal: Lose points upon encountering static punishments

Preconditions:

- The game is running
- Punishments exist on the board

Trigger:

- The player moves into a punishment cell

Scenario:

1. The player moves onto a punishment cell
2. The system decreases the player's score
3. The punishment disappears from the board
4. If the score drops below 0, the game ends

Exceptions:

- The punishment is not removed from the board → Display an error
- The score does not decrease → Display an error

Priority: Essential

When Available: Throughout the game

Frequency of Use: Multiple times per level

Use Case 8: Use Super Points

Primary Actor: Player

Goal: Temporarily double movement speed or eat moving enemies

Preconditions:

- The game is running
- Super points are present on the board

Trigger:

- The player moves onto a super point cell

Scenario 1:

1. The player moves onto a super point cell
2. The system registers the super point collection
3. The player's movement speed doubles temporarily
4. After a certain duration, the speed returns to normal

Scenario 2:

1. The player moves onto a super point cell
2. The system registers the super point collection
5. The player can temporarily eat the enemies.
6. If an enemy is eaten, then the enemy respawns in the respawn area
7. After a certain duration, enemies will be able to kill you again if a collision occurs

Exceptions:

- The super point does not activate → Display an error
- The speed boost does not end → Display an error
- The super point visual remains on board after collection → Remove visual and display warning
- Multiple super points activate simultaneously → Only apply one super point effect
- Enemy respawn fails after being eaten → Force respawn at default location
- Super point timer fails to start → Force end effect after maximum duration
- Player-enemy collision detection fails during super mode → Force correct collision state
- Enemy respawns in invalid location → Move to default respawn position

Priority: Optional (Enhances gameplay)

When Available: Throughout the game

Frequency of Use: Limited times per level

Use Case 9: Game Over

Primary Actor: Player

Goal: Display the ‘Game Over’ screen

Preconditions:

- The player’s score drops below 0 OR
- The player collides with a moving enemy

Trigger:

- The game registers the player’s loss

Scenario:

1. The player either gets caught by an enemy or their score drops below 0
2. The system registers the game over state
3. The system displays a ‘Game Over’ screen
4. The player can choose to restart or exit

Exceptions:

- The game over screen does not display → Display an error

Priority: Essential

When Available: Whenever the player loses

Frequency of Use: Once per game session if the player loses

Use Case 10: Save Game State

Primary Actor: Player

Goal: Store current progress

Preconditions:

- Game is running

Triggers:

- The player activates the save function

Scenario:

1. Player triggers save
2. The system captures the current state:
 - Level progress
 - Score
 - Player position
 - Goblin positions
 - Power-up states
3. The system writes a save file
4. The system confirms save success

Exceptions:

- Save file write fails → Retry save
- Corrupt save state → Maintain the previous save

Priority: Optional

When Available: Throughout gameplay

Frequency of Use: Player determined

Use Case 11: Configure Controls

Primary Actor: Player

Goal: Customize keyboard mapping

Preconditions:

- The settings menu is open

Triggers:

- The player selects the control configuration

Scenario:

1. Player opens settings
2. The system displays the current mapping
3. The player selects the key to remap
4. The player inputs a new key
5. The system validates and saves mapping

Exceptions:

- Invalid key combination → Request new input
- Save configuration fails → Restore defaults

Priority: Optional

When Available: In the settings menu

Frequency of Use: As needed