

Use Cases

Use case: Collect Reward

Primary actor: Main Character (mc)

Goal in context: For the mc to collect a reward

Preconditions: Game must be started with the board map loaded in and populated with “regular” rewards (and one “bonus” award)

Trigger: Mc must move to a cell containing a reward

Scenario:

1. Mc: initially placed on the “start cell” of the board map
2. Mc: moves around the board map (2D grid) to where a reward is located and moves on the cell where it is located to claim
3. Mc: is awarded points (amount may differ) which are added to the total score (displayed on screen) and reward is removed from the map (can no longer be collected)

Exceptions:

1. There are no more rewards to collect: in that case this action can not be completed
2. Mc tries to move to a cell where there is a barrier: mc is blocked from that cell
3. Mc is not able to collect the bonus award before it disappears: bonus award cannot be collected

Priority: Essential, must be implemented

When available: Once game begins

Frequency of use: Multiple times per game

Channel to actor: Via keyboard controls

Secondary actors: Start screen user, End screen user

Channels to secondary actors: Mouse cursor

Open issues:

1. What happens if a reward cannot be reached by the mc (obstructed by barriers)?

Use case: Win Condition

Primary actor: Main Character (mc)

Goal in context: For the mc to “win” the game and finish

Preconditions: Main Character must collect all regular rewards

Trigger: Main Character moves to the exit cell

Scenario:

1. Mc: moves around the board map (2D grid) and collects/claims all the regular rewards
2. Mc: moves to the exit cell
3. Mc: “wins” the game and exit screen is displayed
4. System: End screen is displayed (with final score, replay option, and exit game option)

Exceptions:

1. Main character enters same cell as moving enemy: game ends
2. Main character score falls below zero: game ends

Priority: Essential, must be implemented

When available: Once game begins

Frequency of use: Once per game

Channel to actor: Via keyboard controls

Secondary actors: Start screen user, End screen user

Channels to secondary actors: Mouse cursor

Open issues:

1. What happens if a reward cannot be reached by the mc (obstructed by barriers)?
2. Enemy/Punishment is located on the exit cell

Use case: Collision Fixed Enemy

Primary actor: Main Character (mc)

Goal in context: To stall the main character and lower his score

Preconditions: Main character score must be greater than or equal to 0 to decrease score

Trigger: Main Character moves onto a punishment cell

Scenario:

1. Mc: Moves a different tile
2. Mc: Lands on punishment cell
3. Mc: is frozen for 2 game ticks
4. Fixed Enemy: Deducts points per game tick from the mc score
5. Mc: Thaws and is able to move to an adjacent cell

Exceptions:

1. Main character collected all regular rewards and reached the exit cell: mc wins the game
2. If punishment causes the score below zero, then game over, score cannot be deducted anymore

Priority: Essential, must be implemented

When available: Once game begins

Frequency of use: Multiple times per game

Channel to actor: Via keyboard

Secondary actors: Start screen user, End screen user

Channels to secondary actors: Mouse cursor

Open issues:

1. Punishment cell not reachable
2. Punishment cell is completely avoidable

Use case: Collision Moving Enemy

Primary actor: Main Character (mc)

Goal in context: For the mc to lose the game

Preconditions: Game must be initiated

Trigger: Main Character moves onto the same cell as moving enemy

Scenario:

1. Mc: moves around the board map (2D grid) and collects/claims all the regular rewards
2. Moving Enemy: moves around the board taking the shortest path to the mc
3. Mc: running to the exit cell
4. Moving Enemy: catches up and moves onto the same cell as mc
5. Mc: Mc loses the game
6. System: End screen is displayed (with replay option and exit game option)

Exceptions:

1. Main character successfully collects all rewards and reaches exit cell
2. Game has not been initiated
3. No moving enemies are spawned into the game

Priority: Essential, must be implemented

When available: Once game begins

Frequency of use: Once per game

Channel to actor: Via keyboard controls

Secondary actors: System, End game Screen

Channels to secondary actors: Mouse cursor

Open issues:

1. Moving enemies could make the game too difficult as they take the optimal path to the main character
2. Moving enemies are too simple and are unable to reach the main character

Use case: Re-initiate Game

Primary actor: User (mc)

Goal in context: For the user to play the game again

Preconditions: Game must in end game scene

Trigger: User clicks on the “play again” button

Scenario:

1. Mc: loses the game by either colliding with a moving enemy, or causing his score to go negative by colliding with a fixed enemy (punishment)
2. System: End screen is displayed (with replay option and exit game option)
3. User: Moves cursor onto “replay button”
4. User: Pressed left mouse cursor
5. System: Changes from title screen to gameplay screen

Exceptions:

1. User quits the game

Priority: Essential, must be implemented

When available: Once game reaches end game scene

Frequency of use: Once per win or lose

Channel to actor: Via mouse cursor

Secondary actors: System

Channels to secondary actors: Mouse cursor

Open issues:

1. Game might not reset correctly upon replaying, and player and enemy positions may not be reset

Use case: Movement

Primary actor: Main Character

Goal in context: To give user movement controls of the main character

Preconditions: Game must be initiated

Trigger: User must press any of WASD keys

Scenario:

1. Mc: initially placed on the “start cell” of the board map
2. Mc: moved around the board map (2D grid) by system using WASD

Exceptions:

1. Main Character tries to move onto a cell with a barrier on it, which is adjacent to him
2. Main Character is frozen by a freezer punishment

Priority: Essential, must be implemented, Movement in every direction must work

When available: Once game begins

Frequency of use: Once per game tick

Channel to actor: Via keyboard controls

Secondary actors: System

Channels to secondary actors: Keyboard controls

Open issues:

1. Main Character can't move to an empty cell (“open” tile)

Use case: Initiate Game

Primary actor: User

Goal in context: To start the game

Preconditions: Game file must be executed and running

Trigger: Pressing the “play” button on the title screen

Scenario:

1. User: Moves cursor onto “play button”
2. User: Pressed left mouse cursor
3. System: Changes game screen to gameplay

Exceptions:

1. User never clicks on the start button
2. User clicks on the “rules” page

Priority: Essential, must be implemented

When available: Once .exe has been executed and title screen has loaded

Frequency of use: Once before the game has begun

Channel to actor: Via mouse cursor

Secondary actors: System

Channels to secondary actors: Mouse cursor

Open issues:

1. Game file crashes