Use Cases:

Use cases: **Moving Character**
Primary actor: Player.
Goal in context: To allow the character in game the ability to move.
Preconditions: Player must have an input device (keyboard).
Trigger: Player wants to move the character either north, south, east, or west.
Scenario:
1. Player observes game board.
2. Player press keyboard keycaps to move character up, down, left, or right.
3. Player observes character's attempt to move to desired direction.
4. Player observes the result of the attempt and the new position of enemies.

Exceptions:
1. Player attempts to walk into a wall (trees): character will not move while enemies will.
2. Player enters an invalid input: character will not move while enemies will.
3. Player attempts to walk to unblocked tiles (tiles that are not trees): character and enemies will move.

Priority: Essential, must be implemented.
When available: First implement.
Frequency of use: Many times per game.
Channel to actor: Via keyboard**.**
Secondary actors: n/a.
Channels to secondary actors: n/a.
Open issues: n/a.

Use cases: **Colliding with enemy**
Primary actor: Player.
Goal in context: To cause the character to lose health
Preconditions: Player enters a valid input.
Trigger: Player moves into a tile containing enemy or enemies walk into character.
Scenario:
1. Player observes game board.
2. Player press keyboard keycaps to move character.
3. Character moves into tile containing an enemy (hunters or snakes).
4. Character loses health.

Exceptions:
1. Player enters an invalid input: character will not move, and if an enemy moves into the character, the character will lose health.
2. Player enters valid input: character moves into an enemy and loses health.

Priority: Essential, must be implemented.

When available: first implement.

Frequency of use: Up to 1 – 3 times per game (depends on the amount of health given to character).

Channel to actor: Via keyboard.

Secondary actors: Enemies.

Channels to secondary actors: Game board.

Open issues:
1. How much health should a character have?
2. If a character moves into a tile containing an enemy and at the same time, that enemy moves away from that tile, what interaction should it have?

Use cases: **Colliding with rewards**

Primary actor: Player.

Goal in context: To give bonus points to player.

Preconditions: Player enters a valid input.

Trigger: Player wants to, or coincidentally, move into tile containing a reward (chicken or egg).

Scenario:
1. Player observes game board.
2. Player press keyboard keycaps to move character.
3. Character moves into tile containing a reward.
4. Player's score will be updated and increased.
5. Player observes new score on screen.

Exceptions:
1. Character moves into a chicken tile: Score is increased by a set amount of value.
2. Character moves into an egg tile: Score is increased by a set amount of value.

Priority: Essential, must be implemented.

When available: first implement.

Frequency of use: up to the amount of reward tiles in game board.

Channel to actor: Via keyboard.

Secondary actors: Reward tiles.

Channels to secondary actors: Game board.

Open issues: n/a.

Use cases: **Moving onto exit tile**

Primary actor: Player.

Goal in context: For player to end the game.

Preconditions: Player has collected all regular rewards (chickens).

Trigger: Player wants to end the game.

Scenario:

1. Player observes game board.
2. Player press keyboard keycaps to move character.
3. Character moves into exit tile.
4. Player observes end screen with a final score.

Exceptions:
1. Player has not collected all regular rewards: Character cannot walk onto exit tile, resulting in a wasted move.
2. Player has collected all regular rewards: end screen will appear with a final score.

Priority: Essential, must be implemented.
When available: first implement.
Frequency of use: once per game.
Channel to actor: Via keyboard.
Secondary actors: Exit tile.
Channels to secondary actors: Game board.
Open issues: n/a.

Use cases: **Restarting the game**
Primary actor: Player.
Goal in context: To allow player to restart the game.
Preconditions: System must have a menu.
Trigger: Player wants to start over again.
Scenario:
1. Player starts a new game.
2. Player presses the 'esc' key.
3. Player observes the menu.
4. Player selects the 'restart' button within the menu.
5. Player presses 'esc' key again.
6. Player observes the initial state of the game board.
7. Player starts another new game.

Exceptions:
1. n/a.

Priority: Optional
When available: second implement.
Frequency of use: from zero to many times per game.
Channel to actor: Via keyboard.
Secondary actors: n/a.
Channels to secondary actors: n/a.
Open issues: n/a.

Use cases: **Changing the game volume**

Primary actor: Player.

Goal in context: For player to change the volume of the game.

Preconditions: System must have a menu.

Trigger: Player wants to adjust the volume of the game.

Scenario:

1. Player starts a new game.
2. Player presses 'esc' key.
3. Player selects the 'volume' button within the menu.
4. Player selects volume within the 'volume' section
5. Player presses 'esc' key twice.
6. Player listens to change in volume.

Exceptions:

1. Player selects 'high' within the 'volume' section: Volume will be at the maximum setting.
2. Player selects 'medium' within the 'volume' section: Volume will be at the regular and recommended setting.
3. Player selects 'low' within the 'volume' section: Volume will be at the lowest setting.

Priority: Optional

When available: Third implement.

Frequency of use: From zero to many times per game

Channel to actor: Via keyboard

Secondary actors: n/a.

Channels to secondary actors: n/a.

Open issues: n/a.