Group: Project 20

Use Case: Control the Main Character

Iteration: 1, First Edition
Primary actor: Player
Goal in context: To control the player and guide the movement as directed
Preconditions: The game is started, and the main character is placed at the starting location on the game board.
Trigger: A key on the keyboard is pressed by the player to indicate the main character will move up, down, left, or right.
Scenario:

1. The game is in progress, and the main character is on the game board.
2. The player presses a key (W/A/S/D or arrow keys) to move the main character in a desired direction.
3. The system checks if the move is valid (no walls, barriers, or invalid board positions in the direction of the move).
4. If the move is valid, the main character moves one cell in the specified direction.

Exceptions:

1. If the player attempts to move the main character into a wall or barrier, the move is ignored, and the character remains in the current position.
2. If the player's move would result in an encounter with a moving enemy, trigger the "Game Over" sequence. See use case **Caught by Enemy**.
3. If the player presses a key that is not one of the assigned move keys the player will stay still while the enemies will advance.

Priority: High, as character movement is a core mechanic of the game.
When available: Only while the game is active, not during pause or menu screens.
Frequency of use: Very frequent, character movement is near constant while the game is active.
Channel to actor: Direct game interaction using a keyboard
Secondary actors: N/A
Channels to secondary actors: N/A
Open issues: N/A

Use Case: Enemy Movement

Iteration: 1, First Edition
Primary Actor: Game System
Goal in Context: To control enemy movements towards the player or along a set path, providing challenges and obstacles.
Preconditions: The game is started, enemies are spawned on the game board.
Trigger: Each game tick triggers enemy movement.
Scenario:

1. A game tick occurs.

2. The system calculates the next move for each enemy based on their AI, either towards the player's last known position or along a predetermined path.
3. Enemies move one cell in the chosen direction if it's not blocked by walls or barriers.

Exceptions:

1. If an enemy's move would place it on the same cell as the player, trigger an encounter. See use case **Caught by Enemy**.
2. If the path is blocked, select an alternative route or stay in place.

Priority: High, as enemy interactions are a main challenge of the game.
When Available: Only while the game is active, not during pause or menu screens.
Frequency of Use: Constant during gameplay, with movements occurring every game tick.
Channel to Actor: Game system logic controls enemy movement.
Secondary Actors: Main Character (enemy following it)
Channels to Secondary Actors: Game mechanics.
Open Issues:

1. N/A

Use Case: Collect Rewards

Iteration: 1, First Edition
Primary Actor: Player
Goal in Context: To increase the player's score or abilities by collecting rewards scattered across the game board.
Preconditions: The game is started, and rewards are placed on the game board.
Trigger: The main character moves to a cell containing a reward.
Scenario:

1. The player moves the main character to a cell with a reward.
2. The system detects the collection and updates the game state—increasing the player's score or providing a temporary power-up.
3. The reward disappears from the board.

Exceptions:

1. If a special condition is required to collect the reward (e.g., a key or a certain score), and it's not met, the reward is not collected.
2. Rewards and special reward cells will provide different rewards.

Priority: High, as collecting rewards are part of the win condition.
When Available: Anytime during the game when rewards are present on the board.
Frequency of Use: Medium, but the frequency will vary based on player strategy and amount of rewards present.
Channel to Actor: Direct game interaction through keyboard controlled character movement.
Secondary Actors: The rewards on a given location.
Channels to Secondary Actors: Game logic and the board.
Open Issues:

1. If rewards should have time limits, or an optional setting for it based on difficulty.

Use Case: Reach the Exit to Win the Game

Iteration: 1, First Edition

Primary Actor: Player

Goal in Context: To conclude the game by reaching the "exit" cell after collecting all rewards necessary for winning.

Preconditions:

1. All regular rewards collected.
2. The main character is in control of the player.
3. The "exit" cell is defined, reachable, and the path to it is clear.

Trigger: The player guides the main character to the "exit" cell after having collected all regular rewards.

Scenario:

1. Following the collecting of all rewards the player moves towards the "exit" cell.
2. The player reaches the "exit" cell.
3. The game system verifies that all conditions for winning have been met.
4. A victory message is displayed, and the game transitions to a winning sequence.

Exceptions:

1. If the player has not collected all regular rewards, the system will prevent the win condition from being met, even if the player reaches the "exit" cell, and the game will be lost. See use case **Exit Without All Rewards**.

Priority: High, this action concludes the game and is part of the win condition.

When Available: After all regular rewards have been collected.

Frequency of Use: Once per game session when the win conditions are met.

Channel to Actor: Direct game interaction through character movement towards the "exit" cell.

Secondary Actors: The “exit” cell

Channels to Secondary Actors: The player movement and the board of the game.

Open Issues:

1. Should the “exit” cell spawn in a random location every time or in a predetermined spot.

Use Case: Caught by Enemy

Iteration: 1, First Edition

Primary Actor: Player

Goal in Context: To handle the scenario where the player’s character is caught by an enemy, resulting in a game over.

Preconditions: The game is in progress, and enemies are active on the game board.

Trigger: The player's character comes into contact with an enemy.

Scenario:

1. The player navigates the game environment.
2. The player's character is caught by an enemy.
3. The game system recognizes this loss condition.
4. The game ends, and a game over message is displayed.
5. The player is returned to the main menu.

Exceptions: N/A

Priority: High
When Available: During active gameplay.
Frequency of Use: As often as the player encounters enemies, it will be capped at the player's health based on the difficulty.
Channel to Actor: Direct game interaction through keyboard controlled character movement.
Secondary Actors: The enemies
Channels to Secondary Actors: Game logic for enemy encounters and the board.
Open Issues:

1. N/A

Use Case: Score Drops Below Zero

Iteration: 1, First Edition
Primary Actor: Player
Goal in Context: To conclude the game when the player's score drops below zero due to punishments.
Preconditions: The game is in progress, and the scoring system is active.
Trigger: The player's score drops below zero.
Scenario:

1. The player interacts with a punishment that reduces the score.
2. The score reaches a value below zero.
3. The game system recognizes this as a loss condition.
4. The game ends, and a game over message is displayed.

Exceptions: N/A
Priority: High, this is a major loss condition.
When Available: During active gameplay when score and punishments are present.
Frequency of Use: If it occurs in a game, it will only happen once.
Channel to Actor: Direct game interaction with punishment cell.
Secondary Actors: The punishment cells.
Channels to Secondary Actors: Game logic for enemy encounters and the board.
Open Issues:

1. N/A

Use Case: Exit Without All Rewards

Iteration: 1, First Edition
Primary Actor: Player
Goal in Context: To process the game over scenario when the player reaches the exit without collecting all regular rewards.
Preconditions: The game is in progress with rewards placed on the game board.
Trigger: The player reaches the exit cell without having collected all regular rewards.
Scenario:

1. The player navigates towards the exit cell.
2. The player reaches the exit without collecting all required rewards.

3. The system detects the missing rewards and triggers a game loss.
4. The game ends, displaying a game over message.

Exceptions:

1. N/A

Priority: High, users can still end the game without winning.
When Available: When the exit is accessible during gameplay.
Frequency of Use: Once per game session.
Channel to Actor: Direct game interaction through character movement towards the "exit" cell.
Secondary Actors: The “exit” cell
Channels to Secondary Actors: The player movement and the board of the game.
Open Issues:

1. N/A

Use Case: Start Game

Iteration: 1, First Edition
Primary Actor: Player
Goal in Context: To initiate a new game session, allowing the player to begin the gameplay experience.
Preconditions: The game application is launched, and the player is at the main menu.
Trigger: The player selects the "Start Game" option.
Scenario:

1. The player selects "Start Game" from the main menu.
2. The system initializes a new game session, setting the score to zero and placing the main character at the starting location.
3. The system populates the game board with rewards, enemies, and barriers according to the initial game level design.
4. The game begins, and player control is enabled.

Exceptions:

1. If there is an error in loading game resources, the system displays an error message and may return the player to the main menu.

Priority: High, this is critical as it starts the game..
When Available: At the main menu after the game application is launched.
Frequency of Use: Once per game session.
Channel to Actor: User interface selection.
Secondary Actors: N/A
Channels to Secondary Actors: N/A
Open Issues:

1. Deciding to include an optional tutorial sequence.

Use Case: Menu End Game

Iteration: 1, First Edition
Primary Actor: Player

Goal in Context: To conclude a game session by voluntarily exiting.
Preconditions: The game is in progress and a game menu is available
Trigger: The player selects to exit the game from the menu
Scenario:

1. The player selects "Exit" from the pause menu.
2. The system processes the end of the game session.
3. The game session is closed, and the main menu pops up.

Exceptions:

1. The other method of ending the game without intention of finishing is to go straight to the exit tile without collecting the rest of the rewards, which would result in a game loss. See use case **Exit without all rewards**.

Priority: Medium, ending a game session by menu is optional when there is another method of exiting. Implemented for player convenience.
When Available: Anytime during an active game session from the pause menu.
Frequency of Use: Once per game session if needed by the player.
Channel to Actor: The settings menu.
Secondary Actors: Game System (handles the transition)
Channels to Secondary Actors: N/A
Open Issues:

1. Determining if a confirmation should be implemented.
2. Determining if a game loss should be displayed.

Use Case: Adjust Settings

Iteration: 1, First Edition
Primary Actor: Player
Goal in Context: To customize the game settings according to the player's preferences, including sound levels, difficulty, or controls.
Preconditions: The game is either in progress or at the main menu/pause menu.
Trigger: The player selects the “Settings” menu by hitting the key ‘esc’.
Scenario:

1. The player accesses the “Settings” menu from the main menu or during gameplay.
2. The player navigates through various settings categories, such as audio, video, or controls.
3. The player selects and adjusts individual settings, such as lowering the sound volume, or changing the difficulty level.
4. The system applies the changes immediately where applicable or prompts the player to confirm the changes.
5. Once adjustments are made, the player exits the settings menu, and the game resumes with the new settings in effect.

Exceptions:

1. If settings require a game restart or return to the main menu to take effect, the game ignores the call and possibly informs the player.

2. If an invalid configuration is attempted, the system reverts to the last valid setting set or default.

Priority: Low, not essential but adjusting settings enhances player satisfaction and accessibility.
When Available: Any time during the game for settings that can be changed in real-time, or only from the main/pause menu for settings that require a restart, such as difficulty.
Frequency of Use: Infrequent, typically adjusted when starting the game for the first time or when the player desires to change their gameplay experience.
Channel to Actor: User interface within the game.
Secondary Actors: N/A
Channels to Secondary Actors: N/A
Open Issues:

1. Determining if any settings should be locked or restricted based on game progression or other criteria, such as difficulty

---

Use Case: Calculate Score

Iteration: 1, First Edition
Primary Actor: Game System
Goal in Context: To accurately calculate and update the player's score based on in-game actions, such as collecting rewards or receiving punishments.
Preconditions: The game is in progress, and the scoring system has been initialized at the start of the game.
Trigger: The player collects a reward or encounters a punishment.
Scenario:

1. The player maneuvers the main character to collect a reward or accidentally encounters a punishment.
2. The game system detects the collection of a reward or the encounter with a punishment.
3. The system calculates the change in score which increments for rewards and decrements for punishments.
4. The updated score is immediately reflected in the game's user interface.

Exceptions:

1. If the score calculation results in a negative number, game over condition is triggered. See use case **Score drops below zero**.
2. If there is an error in score calculation.

Priority: High, the score calculation is essential for gameplay feedback and progression.
When Available: Constant during gameplay, as rewards and punishments are encountered.
Frequency of Use: Each time a reward or punishment is encountered.
Channel to Actor: Directly through game logic and mechanics.
Secondary Actors: N/A
Channels to Secondary Actors: N/A
Open Issues:

1. N/A