

Use Cases

1. Use Case: Play Game

Primary Actor: Player

Description: This use case represents the initial step where the player starts the game.

Preconditions: The game application is running.

Flow of Events:

1. The player chooses to enable or disable cheat mode and specify Tokimons and Fokimons (if available).
2. The player launches the game.
3. The game initializes, displaying the game grid and relevant information.
4. The player inputs their initial location on the grid.

Postconditions: The game is set up, and the player's character is placed on the grid. Cheat mode settings are applied.

Alternative Flows:

- If the player decides not to enable cheat mode, they proceed to the next step.
- If the player enters an invalid initial location, they are prompted to re-enter a valid one.
- If the player enter invalid arguments, the program exits with the hint how to launch the game properly.

2. Use Case: Navigate within Grid

Actor: Player

Description: This use case describes how the player can navigate within the game grid.

Preconditions: The game is in progress.

Flow of Events:

1. The player can move their character up, down, left, or right within the grid using keys W, A, S, and D.
2. The player's location is updated on the grid.
3. The player can collect Tokimons and encounter Fokimons as they move.

Postconditions: The player's character is at a new grid location, and game elements may be revealed or collected.

Alternative Flows:

- If the player attempts to move to an invalid grid location (e.g., off the grid), the game prevents the movement and asks the user to input a valid character.

3. Use Case: Use a Spell

Actor: Player

Description: This use case outlines the player's ability to use special spells within the game.

Preconditions: The game is in progress, and the player has spells left.

Flow of Events:

1. The player selects a spell from the available options.
2. The chosen spell's effect is applied to the game grid, such as randomly moving the player to another location, revealing and collecting the location of a Tokimon, or eliminating a Fokimon.
3. The number of spells remaining is reduced.

Postconditions: The chosen spell's effect is applied, and the number of remaining spells is updated.

Alternative Flows:

- If the player doesn't have any spells left, they are informed that no spells are available for use.

4. Use Case: Exit the Game

Actor: Player

Description: This use case represents the player's choice to exit the game.

Preconditions: The game is in progress.

Flow of Events:

1. The player selects the option to exit the game.
2. The game session is terminated, and the game closes.

Postconditions: The game session is terminated, and the game application is closed.

Alternative Flows:

- If the player accidentally selects the exit option, there is no way of going back.

5. Use Case: Game Over

Actor: Game System

Description: This use case describes the game's response when the player encounters a Fokimon.

Preconditions: The player's character is on the same grid location as a Fokimon.

Flow of Events:

1. The game displays a message indicating the player has encountered a Fokimon.
2. The game reveals the locations of all Fokimons on the grid.
3. The game session ends.

Postconditions: The game session ends, Tokimon and Fokimon locations are revealed.

Alternative Flows:

- No Alternative Flows.

6. Use Case: Winning the Game

Actor: Game System

Description: This use case represents the game's response when the player collects all Tokimons and wins the game.

Preconditions: The player has collected all Tokimons on the grid.

Flow of Events:

1. The game displays a victory message, declaring the player as the Master of Tokimons.
2. The game reveals the locations of all Fokimons on the grid.
3. The game session ends.

Postconditions: The game session ends, and Fokimon locations are revealed.

Alternative Flows:

- No Alternative Flows.