

## Use cases for *Ghost Town*

---

**Use case:** *InitiateGame*  
**Primary Actor:** Player  
**Goal in context:** To launch the application  
**Preconditions:** The player has the application installed on their system  
**Trigger:** Player wants to play the game

**Scenario:**

1. The player launches the game.
2. The game displays the main menu which contains a set of function buttons: "Play", "Restart" and "Exit".

**Exceptions:**

1. Some form of memory corruption: displays "game error".
2. Incompatible systems: displays "game error".

**Priority:** High priority  
**When available:** First increment.  
**Frequency of use:** Infrequent.

---

**Use case:** *PlayGame*  
**Primary Actor:** Player  
**Goal in context:** To put the player into an initial game state  
**Preconditions:** The game is already launched  
**Trigger:** The player wants to play

**Scenario:**

1. The player is in the main menu.
2. The player selects "Play".
3. The game displays the map and relevant game objects.
4. The player moves around the map using keyboard controls.

**Exceptions:**

1. The player selects "Restart" from the main menu: Displays error message.
2. The player selects "Exit" from the main menu: See use-case "ExitGame".

**Priority:** High priority  
**When available:** First increment  
**Frequency of use:** infrequent

---

**Use case:** *MoveCharacter*  
**Primary Actor:** Player  
**Goal in context:** To move the character on the map in the direction specified by the player  
**Preconditions:** A game is being played  
**Trigger:** The player wants to move the character

**Scenario:**

1. The player is in a game.
2. The player presses "W".
3. The character moves up the map a step.

**Exceptions:**

1. The player presses "A", "S" or "D": The character moves left/down/right the map a step respectively.
2. The player presses any other key: Displays an error message showing the controls.
3. The player walks into a wall: Displays a small visual effect informing the user that the character is walking into a wall. Movable enemies still move.
4. The player ends his movement on a reward space: see use-case "ClaimReward".
5. The player ends his movement on a space with punishment: see use-case "TakePunishment".

**Priority:** High priority  
**When available:** Second increment  
**Frequency of use:** Very frequent

---

**Use case:** *ClaimReward*  
**Primary Actor:** Player  
**Goal in context:** To display and update the score of the player  
**Preconditions:** A game is being played  
**Trigger:** The player wants to claim rewards

**Scenario:**

1. The player is in a game.
2. The player walks onto a reward space using keyboard controls.
3. The map updates to remove the reward.
4. The scoreboard updates to show the current score.

**Exceptions:**

1. The player walks onto a normal space.
2. The player walks onto a space with punishment: see use-case "TakePunishment".

**Priority:** Medium priority  
**When available:** Third increment  
**Frequency of use:** Frequent

---

**Use case:** *TakePunishment*  
**Primary Actor:** Player  
**Goal in context:** To display and update the score of the player  
**Preconditions:** A game is being played  
**Trigger:** The player wants get past a punishment

**Scenario:**

1. The player is in a game.
2. The player ends his movement on a space with punishment.
3. The map updates to remove the punishment.
4. The scoreboard updates to show the current score.

**Exceptions:**

1. The player walks onto a normal space.
2. The player walks onto a reward space: see use-case "ClaimReward".

**Priority:** Medium priority  
**When available:** Third increment  
**Frequency of use:** Frequent

---

**Use case:** *RestartCurrentGame*  
**Primary Actor:** Player  
**Goal in context:** To reset the game into an initial state  
**Preconditions:** The game is already launched and being played  
**Trigger:** The player wants to restart the current game

**Scenario1:**

1. The player is playing a game.
2. The player selects "Restart".
3. The game resets the map and relevant game objects.

**Scenario 2:**

1. The current game ended because the player won/lost.
2. The player selects "Restart".
3. The game resets the map and relevant game objects.

**Exceptions:**

1. The player selects "Play" from the main menu: Displays error message.
2. The player selects "Exit" from the main menu: See use-case "ExitGame".

**Priority:** High priority  
**When available:** Second increment  
**Frequency of use:** Frequent

---

**Use case:** *ExitGame*  
**Primary Actor:** Player  
**Goal in context:** To exit the game  
**Preconditions:** The game is already launched  
**Trigger:** The player wants close the game

**Scenario:**

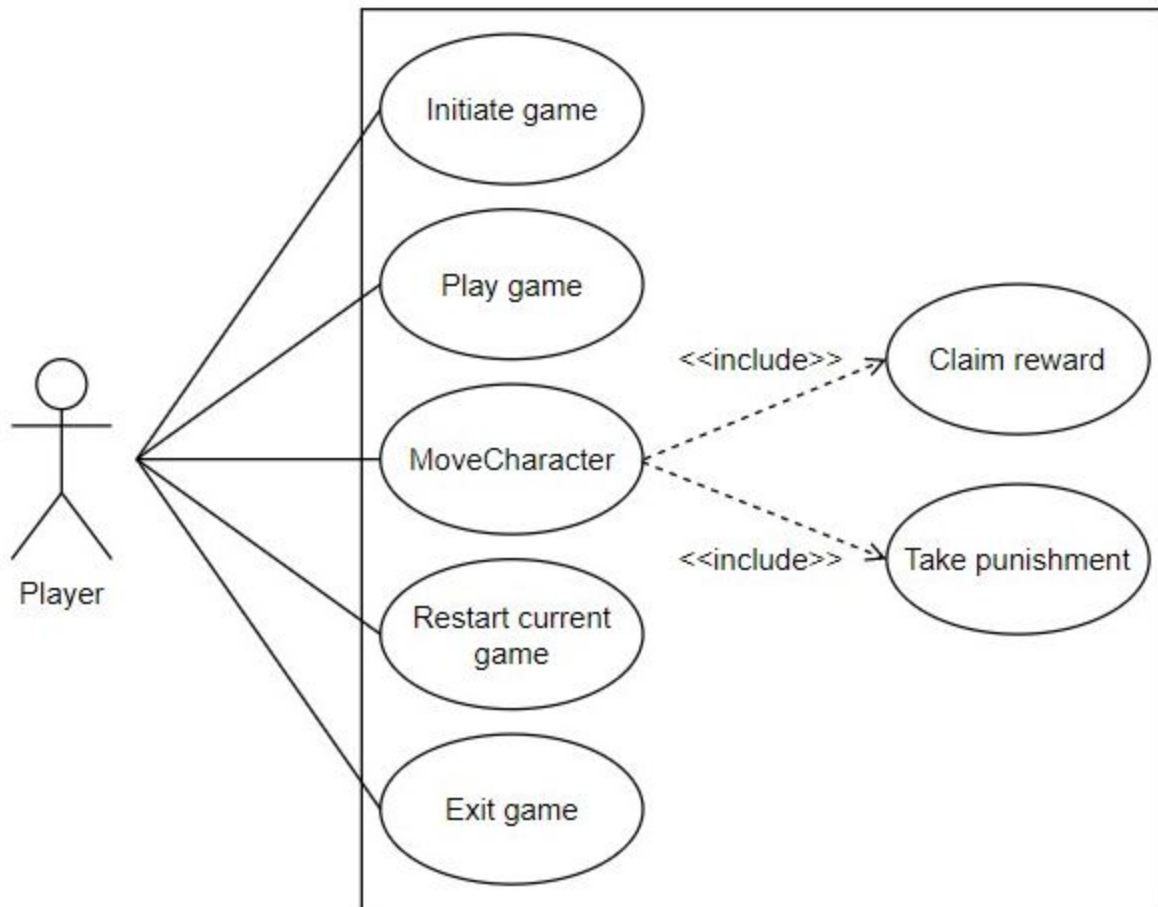
1. The player selects "Exit" from the set of function buttons.
2. The application terminates.

**Exceptions:**

1. The player selects "Play" from the main menu: Displays error message.
2. The player selects "Restart" from the main menu: See use-case "RestartCurrentGame".

**Priority:** High priority  
**When available:** First implement  
**Frequency of use:** Infrequent

---



UML use case diagram for *Ghost Town* application