Course: CMPT-276

***Use Case 1: collecting the reward***

Primary actors: the player

Goal in context: to increase the final score of the player for winning the game

Precondition: complete loading the map

Trigger: the character collides with the reward

Scenario:

1. The system loads the map and its features and reward:
   1. Regular Reward: appears throughout the entire round until the character collected
   2. Bonus Reward: only appear for a limited amount of time/until the character collected
2. Player controls the character moving around the map and avoiding enemy
3. Players control character to collect the reward
4. Reward will disappear after being collected and display the effect
5. The player’s final score will increase according to the types of reward
6. If player collect enough rewards, the gate for the next level will display on the viewing window
7. Players can either staying to collect more reward or enter the gate leads to the next level

Exception:

1. The system is incomplete or unable to load the map. The use case end
2. The player aborts the game, the use cased being aborted. The use case end
3. The player encounters an enemy – see the use case “interact with enemy”
4. The time is running out, but the player does not collect enough reward to open new level. The use case end

Priority: High priority, fundamental function of the game engine

When available: Initial of the game

Frequency of use: highly frequent

Channel to actor: PC-based application

Secondary actor: the character

Channel to secondary actors: Map

Open issues:

1. Potential problems related to maintain high rate of frame per second for stable motion of the game
2. Reward and punishment appear at the same coordinate
3. The system i update the user’s final score when the character “touch” the reward
4. The game crash during the run time

***Use Case 2: Interact with moving enemies***

Primary actors: the player

Goal in context: terminate the game if the character has contact with moving enemies

Precondition: complete loading the map

Trigger: player’s character collides with the moving enemy

Scenario:

1. The system displays the entire environment and moving enemies on viewing window:
   1. Two moving enemies will be randomly generated at beginning of the game
   2. Moving enemies can move 1 cell at the “tick”
   3. Moving enemies can only move within the boundary of the game.
2. The player starts to move the character around the map
3. The moving enemy will target on the character if she stays within their radar’ radius and approach her. Otherwise, the moving enemy will randomly wander around the map
4. There will be no effect if the enemy pass the reward and punishment
5. The player controls the character to avoid the enemy and obstacles
6. If the character collides with the enemy, the game end and the final score turn to 0
7. The system displays game-over message on the screen

Exception:

1. The system is not fully load or unable to load the map. The use case end
2. The player aborts the game, the use cased being aborted. The use case end
3. The player enters the gate to new level. The use case will be aborted (being loaded at the next level)
4. No interaction between character and moving enemies during run time. The use case continues
5. The character collides with the moving enemies, but shows no effect, game crash. The use case end

Priority: high priority – fundamental functions of the game

When available: Initial of the round

Frequency of use: highly frequent

Channel to actor: PC-based system

Secondary actors: character, moving enemies

Open issues:

1. Moving enemies are unable to move or get trapped inside the obstacles/walls
2. Moving enemies do not change direction while colliding with walls or obstacles
3. Moving enemies do not chase the character while character’s position stays inside enemy’s radar
4. There is no exit at the initial location of moving enemies
5. The game crash during the run time.

***Use Case 3: Interact with punishment***

Primary actors: the player

Goal in context: deduct the player’s final score or terminate the game

Precondition: complete loading map

Trigger: character “touch” the punishment

Scenario:

1. The system displays the entire environment and punishment (no motion) on viewing window
2. The player starts to move the character around the map
3. The character touches the punishment
4. The player’s final score will be subtracted by the amount of the punishment
   1. if the final soccer is 0 at the time of interaction, the user will lose the game, the system sends game-over message
5. The punishment will disappear after the interaction with the character

Exception:

1. No interaction between punishment and character during run time. The player’s final score remains unchanged. The use case continues
2. The user aborts the game (during run time), the use case also being aborted. The use case end
3. The character “touches” the punishment while the player’s final score is 0. The use case end
4. The user enters the next level gate. The use case end (will be loaded for the next level).

Priority: high priority – fundamental function of the game

When available: Initial of the game

Frequency of use: medium frequent

Channel to actor: PC-based application

Secondary actor: character, punishments

Open issues:

1. The interaction between character and punishment shows no effect
2. After the interaction with character, the punishment still appears on the map
3. The system fails to update the player’s final score after interaction of the character and punishment
4. The punishment appears at the same position with reward
5. The game crash during run time

***Use Case 4: access the game***

Primary actors: the player

Goal in context: display the map and related features on the screen

Precondition: the player logs-in the game

Trigger: the system initially starts running

Scenario:

1. The player logs on to the game
2. The system loads the map and its features
3. The system displays the boundary of the map, barriers, and features (rewards, moving enemies and punishments) on screen
4. The system displays the final score and time within viewing window
5. The system displays the ready message at each game, player presses any button to trigger
6. The player controls the character moving around the map by 4 keyboard buttons (left, right, up, and down)
7. The system displays the effect when the character interacts with features (rewards, moving enemies, and punishments)
8. The system displays the winning message if the user successfully enters the level gate
9. The system displays the game-over message if the user fails to complete the level (either “touch” moving enemies or “touch” punishment while the score equal to 0 or running out of time)

Exception:

1. The system is not fully load or unable to load the map. The use case end
2. The player aborts the game, the use cased being aborted. The use case end
3. The character being eliminated (“touch” moving enemy at any time or punishment while the final score is 0). The use case end
4. The time is running out, but the player does not collect enough score to open new level. The use case end

Priority: High priority, fundamental function of the game engine

When available: Initial of the game

Frequency of use: highly frequent

Channel to actor: PC-based application

Secondary actor: system/game

Open issues:

1. Power outage occurs, system suddenly shutdowns
2. The game crash during run time