Use Cases

Group: Group 9

Use Case Name: Move Hero

Use Case ID: UC-1

Scope: ROKUE LIKE-Phase 1

Level: User Goal

Primary Actor: Player

Stakeholders and Interests:
-Player: Wants to be able to move the hero in the chosen direction on the grid without delay, as even a millisecond of latency can result in monster damage.
-Developer: Wants a bug-free game that offers a decent user experience.

Preconditions:
- The game is in play mode.

Postconditions:
-The hero was moved east/west/north/south in response to user keyboard commands.

Main Success Scenario:
1. The player presses the arrow keys that allow the Hero to move in the desired direction within a hall.
2. The hero goes east/west/north/south without delay based on the key tapped by the user.
3. Hero stays at its new location.
4. Hero can still move inside the hall's established boundaries.

Extensions:
*a. At any time, the System fails.
    1. The player restarts the Game.
2a. The player reaches the visible boundaries of the hall.
    1. Hero stops at its current location and does not move out of hall.

Technology and Data Variations List:
1a. The player uses the keyboard to move the Hero.

Frequency of Occurrence:
-Almost every time interval. The player will almost always move the hero to find the rune, collect new enchantments or escape from monsters except when the game is paused.

Use Case Name: Gather Enchantment

Use Case ID: UC-2

Scope: ROKUE LIKE-Phase 1

Level: User-goal

Primary Actor: Player

Stakeholders and Interests:
-Player: Wants to gather enchantments that occur across the map. Specifically interested in gathering extra time, hints about the location of rune, luring gems, cloak of protection, and extra lives.
-Developer: Wants the enchantments to spawn across random and available points in the map.

Preconditions:
1. The game is in play mode.

Postconditions:
1. The enchantments have been manipulated for players' decisions (used or sent to inventory).
2. Hero's inventory has been updated based on gathered enhancements.

Main Success Scenario:
1. The player left-clicks on enchantments that show up on the map.
2. The player decides between using the enchantment immediately or storing it in his/her inventory.
3. The number of the enchantment in the hero's inventory is increased by 1 if it is decided to be stored.

Extensions:
*a. At any time, the System fails.
1. The player restarts the game.
2a. In case the player inventory is full, the system returns the message "inventory is full".

Special Requirements:
The player uses certain mouse interactions (left click) to gather up enchantments that are randomly located on the grid.
Extra time and Extra Life enchantments cannot be stored in the inventory.
Some methods and data structures to account items in the hero's inventory.

Technology and Data Variations List:
The player uses the mouse to execute it.

Frequency of Occurrence: A random type of Enhancement appears every 12 seconds.

Use Case Name: Gather Rune

Use Case ID: UC-3

Scope: ROKUE LIKE-Phase 1

Level: User-goal

Primary Actor: Player

Stakeholders and Interests:
-Player: Wants to find the object that hides the rune.
-Developer: Wants the rune to spawn inside chest objects and manipulate them according to the game rules.

Preconditions:
1. The game is in play mode.
2. Hero is within the range object, i.e. the Hero is located next to the object.

Postconditions:
1. The items inside the chest have been manipulated for players' decisions.
2. The chest has been closed.

Main Success Scenario:
1. The player triggers the objects.
2. Rune shows up.
3. Rune is obtained by the player.

Extensions:
*a. At any time, the System fails.
1. The player restarts the game.

1a. The hero is not within the range of any objects
1. Nothing happens after the player tries to interact with a chest.

1b. The hero is equidistant to two chests
1. The chest closest to the bottom left (with being in the bottom primed) is selected as the chest to be interacted with.

Special Requirements:
The player uses a certain keyboard action to move towards objects. Also uses mouse interactions to gather runes.
Some methods and data structures to account across all inventories.

Technology and Data Variations List:
The player uses the mouse and keyboard to execute it.

Frequency of Occurrence: Frequently occurring throughout the game as it is the main game mechanic until the game ends and finding the rune is the main purpose of the game.

Use Case Name: Pause/Resume Game

Use Case ID: UC-4

Scope: ROKUE LIKE-Phase 1

Level: User-Goal

Primary Actor: Player

Stakeholders and Interests:
Player: Wants to pause the game and then successfully continue from the paused moment

Preconditions:
    1. The game is in playing mode.

Postconditions:
    1. The game was paused/resumed.

Main Success Scenario:
    1. The player starts the game and it's in running mode
    2. The player presses a button to pause the game
    3. Game is paused.
    4. Pause button is replaced with the resume button.
    5. The player presses a button to resume the game.

Extensions:
*a. At any time, the System fails.
    1. The player restarts the game

2a. After pressing the resume button the game is still in pause mode
    1. The player restarts the game

2b. After pressing the pause button the game is still running
    1. The player restarts the game

3a. The player presses the desired button while the game is paused.

Technology and Data Variations List:
1a. The player uses the mouse to pause/resume the game.

Frequency of Occurrence:

Depends on the player; as long as the system is operational, the player can halt and continue.

Start Game: The actor is the player, who starts the game. The game begins with Building Mode.

Build game: Where the player is the actor. The user can adjust the game difficulty, and the fall speed of objects changes accordingly. The player can define the amount of game objects, unit L, and the shape of certain molecules.

Run Game: The actor is the player, and the player starts the game in Running Mode. It ends the Building mode. Running mode continues till the finish of the game.

Use Case Name: Teleport Hero

Use Case ID: UC-5

Scope: ROKUE LIKE-Phase 1

Level: User-Goal

Primary Actor: System

Stakeholders and Interests:
The system wants the hero to move on to the next hall.

Preconditions:
1. The game is in running mode.
2. The player successfully opened the door of the hall and was sufficiently close to the door.

Postconditions:
1. Hall is updated accordingly to the next stage.

Main Success Scenario:
1. The hero comes next to the door of the hall.
2. The hero opens the door.
3. The hero is teleported to a random location on the map of the next hall.
4. The map and hall are updated accordingly to the next stage.

Extensions:
*a. At any time, the System fails.
1. The player restarts the game

1a. The player has not opened the door with the rune yet
1. The system returns a warning, indicating the door is not opened yet if the player is close to a door that is not open.

Frequency of Occurrence:
Happens once per every hall after the player acquires the rune, opens the door, and reaches for it. If a game is successfully completed, then the hero is teleported three times.

Use Case Name: Reveal Rune

Use Case ID: UC-6

Scope: ROKUE LIKE-Phase 1

Level: User Goal

Primary Actor: Player

Stakeholders and Interests:
-Player: Wants to cast reveal enchantment.

Preconditions:
-The game is in running mode.
-The hero has at least one reveal enchantment in his/her inventory.

Postconditions:
-The number of reveal enchantment has decreased in the hero's inventory.

Main Success Scenario:
1. The player presses the **R** button on the keyboard.
2. A 4x4 square containing the rune is displayed as a hint to the player.
3. The system decreases the number of "reveal" enchantments in the hero's inventory by 1.

Extensions:
*a. At any time, the System fails.
1. The player restarts the game

2a. There isn't any "reveal" left in the player inventory.
1. The system returns a warning, indicating there is no "reveal" enchantment remaining in the hero's inventory.

Technology and Data Variations List:
1a. The player uses the keyboard to use enhancement **Reveal**.

Frequency of Occurence:
The action can be done as long as there is at least one "reveal" item left in inventory.

Use Case Name: Protect Hero

Use Case ID: UC-7

Scope: ROKUE LIKE-Phase 1

Level: User Goal

Primary Actor: Player

Stakeholders and Interests:
-Player: Wants to equip the hero with the cloak of protection to hide the hero from archer monster

Preconditions:
-The game is in running mode.
-The player has at least one cloak of protection in his/her inventory.

Postconditions:
-The number of the cloak of protection enchantment has decreased in the player's inventory.

Main Success Scenario:
1. The player presses the **P** button on the keyboard.
2. The player equips the hero with the cloak of protection to make the hero invisible for archer monster until the determined time ends for the cloak of protection
3. The system decreases the number of cloak of protection enchantments in the player's inventory by 1.

Extensions:
*a. At any time, the System fails.
   2. The player restarts the game

2a. There isn't any cloak of protection left in player inventory.
   2. The system returns a warning, indicating there is no cloak of protection remaining in the player's inventory.

3a. The player is already equipped with one cloak of protection.
   1. The remaining time of the cloak of protection resets to the initial time determined for the cloak of protection.

Technology and Data Variations List:
1a. The player uses the keyboard to use the enchantment **Cloak of Protection**.

Frequency of Occurence:
The action can be done as long as there is at least one cloak of protection left in inventory.

Use Case Name: Throw Luring Gem

Use Case ID: UC-8

Scope: ROKUE LIKE-Phase 1

Level: User Goal

Primary Actor: Player

Stakeholders and Interests:
-Player: Wants to throw a luring gem to fool the fighter monster

Preconditions:
-The game is in running mode.
-The player has at least one luring gem in his/her inventory.

Postconditions:
The number of luring gem enchantment has decreased in the player's inventory.

Main Success Scenario:
1. The player presses the **B** button on the keyboard and then one of the **W A S D** buttons to select the direction to throw.
2. The player throws the luring gem in the selected direction.
3. The system decreases the number of luring gem enchantments in the player's inventory by 1.

Extensions:
*a. At any time, the System fails.
   1. The player restarts the game

2a. There isn't any luring gem left in the player inventory.
   3. The system returns a warning, indicating there are no luring gems left in the player's inventory.

3a. The player is moving while throwing the luring gem.
   2. The luring gem has been thrown as if the player was not moving at that location (the momentum of the player does not add up to the gem momentum)

Technology and Data Variations List:
1a. The player uses the keyboard to use enhancement **Luring Gem**.

Frequency of Occurence:
The action can be done as long as there is at least one luring gem left in inventory.

Use Case Name: Select Game Mode

Use Case ID: UC-9

Scope: ROKUE LIKE-Phase 1

Level: User-Goal

Primary Actor: Player

Stakeholders and Interests:
-Player: Wants to start the game by selecting either Build Mode or Play Mode. The player can use saved maps in previously designed build mode.

Preconditions:
-The game must be launched, and the player must be at the Main Menu screen.

Postconditions:
-The game mode is successfully initialized
-The player starts in either Build or Play Mode.

Main Success Scenario:
1. Game is initialized.
2. Player clicks a button to switch to the **Build Mode** screen.
3. The system transitions to **Build Mode**, where the player designs the inside of the halls.
4. The system transitions to **Play Mode**, where the player uploads the design of the worlds.
5. The system transitions to **Help**, and the player is informed about game mechanisms.

Extensions:
*a. At any time, the System fails.
1. The player restarts the Game.

1a. The player does not meet the minimum criteria for placing objects in the halls during Build Mode.

1. The system gives a warning indicating that the hall does not meet the minimum requirements.
   a. There must be at least 6 objects in the earth hall.
   b. There must be at least 9 objects in the air hall.
   c. There must be at least 13 objects in the water hall.
   d. There must be at least 17 objects in the fire hall.

2a. There are no saved Halls for play mode.
1. The player goes back to the Main Menu and select Build Mode.

Technology and Data Variations List:
The player uses a mouse click to select mode.

Frequency of Occurrence:
Selecting a game mode occurs at the start of every new game session.
If the player restarts the game after exiting or losing, they will need to select the game mode again.

Use Case Name: Build Game

Use Case ID: UC-10

Scope: ROKUE LIKE-Phase 1

Level: User-Goal

Primary Actor: Player

Stakeholders and Interests:

-Player: Wants to build and design the inside of each hall in the dungeon to create a challenging and interesting play area.

Preconditions:
1. The player has clicked **Build Mode** from the Main Menu.
2. The game transitioned to **Build Mode**.

Postconditions:
1. Number of objects populated is based on player's object placement choice along with the minimum number of object criteria..
2. The player proceeds to **Play Mode**.

Main Success Scenario:
1. The system presents the Hall of Earth as the first level to design.
2. The player selects and places various objects in the hall.
3. The player must meet the minimum requirement of objects for each hall:
4. At least 6 objects in the Hall of Earth.
5. The player completes the Hall of Earth and proceeds to the next halls:
6. Hall of Air (requires at least 9 objects),
7. Hall of Water (requires at least 13 objects),
8. Hall of Fire (requires at least 17 objects).
9. Once all halls are built, the player clicks Finish Build, and the system transitions to Play Mode.

Extensions:

*a. At any time, the System fails.
1. The player restarts the game

3-5a. The player does not place the required minimum number of objects in a hall.
1. System gives warning
2. The player continues placing objects until the minimum requirement is met.

3-5b. The player does not place the required minimum number of objects in a hall.
1. The player selects the object and clicks remove.

2. The system updates the hall accordingly.

3-5c. The player decides to exit Build Mode before completing all halls.
1. The player clicks the **Exit** button.
2. System prompts with a confirmation message: "Are you sure you want to exit? Unsaved progress will be lost."

Technology and Data Variations List:

1) The player uses a mouse to select and drag objects to their desired position within the hall.
2) The game displays the halls in a grid format to facilitate object placement.

Frequency of Occurrence: If the player restarts the game, they will need to design the halls again from scratch.