3. GamePlay Elements

2.3) Map and Zones

One of the major parts covered on "LOST" is map structure that indicates the position of user. User can observe his/her position by opening the map section on menu. In main menu, all arrow keys specifying the possible movements of user will be arranged according to the position on the map. In this case, there will be restricted zones which mostly refer to the sea and if the user arrives the seaside, arrow keys will relatively end up. For example, left arrow will disappear if the user located in the left part of map. The map includes several zones defined by the distribution of geographical formations, plant cover and climate conditions.

2.3.1) Target-spesific observation:

Just like in most of strategy games, "Lost" will provide target-specific observation. It means the area where the user located will be lightened up in a particular range. Other parts of map will be darkened and whenever the user changes his/her position, these parts start to be lightened up. The concept of game aims to thrill and bother the user during all the game. At the beginning, user does not know where he/she is and just walk away from the accident that drags him into mysterious area.

2.3.2) Various Zones/ Districts:

In the map, there will be domain-specific items, characters and events bounded by areas. Also all zones will contain a certain amount of items that are identical for only these areas. So if the user picks up and uses these items, the amount will be decreased and there will be a repair/renewal process that requires a specific time.

Forests: Forest is the location where the player borns at the beginning. That is why the difficulty level of the forests is low. They contain items and have various meek animals instead of dangerous ones.

Jungle: This part is more challenging than forests that the user located at the beginning. Productive lands with some wilderness.

Dark Forests: Dark forest is most hazardous part of forests. Includes wildest animals, various items and unexpected events that can be encountered.

Mountains: Mountains are one of the minimal occupancy areas in Lost. They include specific items like rock, stones, bushes, etc. Also wild animals like bear can be rarely encountered inside this area.

Abandoned Village: There is lots of traces that points human existence in long time before. Tents, used items can be found in this area.

Beach: Includes specific items which cannot be found in other areas which covered on beach items section.

Volcano Zone: This are fulfilled with death bodies. The view totally points the wilderness in this area! Player should not rove around this area if he/she does not trust on his/her power and items.

Swamp: Geological formation of this area poses a thread for player to sink under the land. But there will be several worthy elements that direct user to take the plunge.

Iceland: This land is leanest area in the map. It includes rare items and some wildest animals.

Coasts: All island is surrounded with sea, so in coast areas there will be a chance to catch a fish and nourishment. Also they are the boundaries of map.
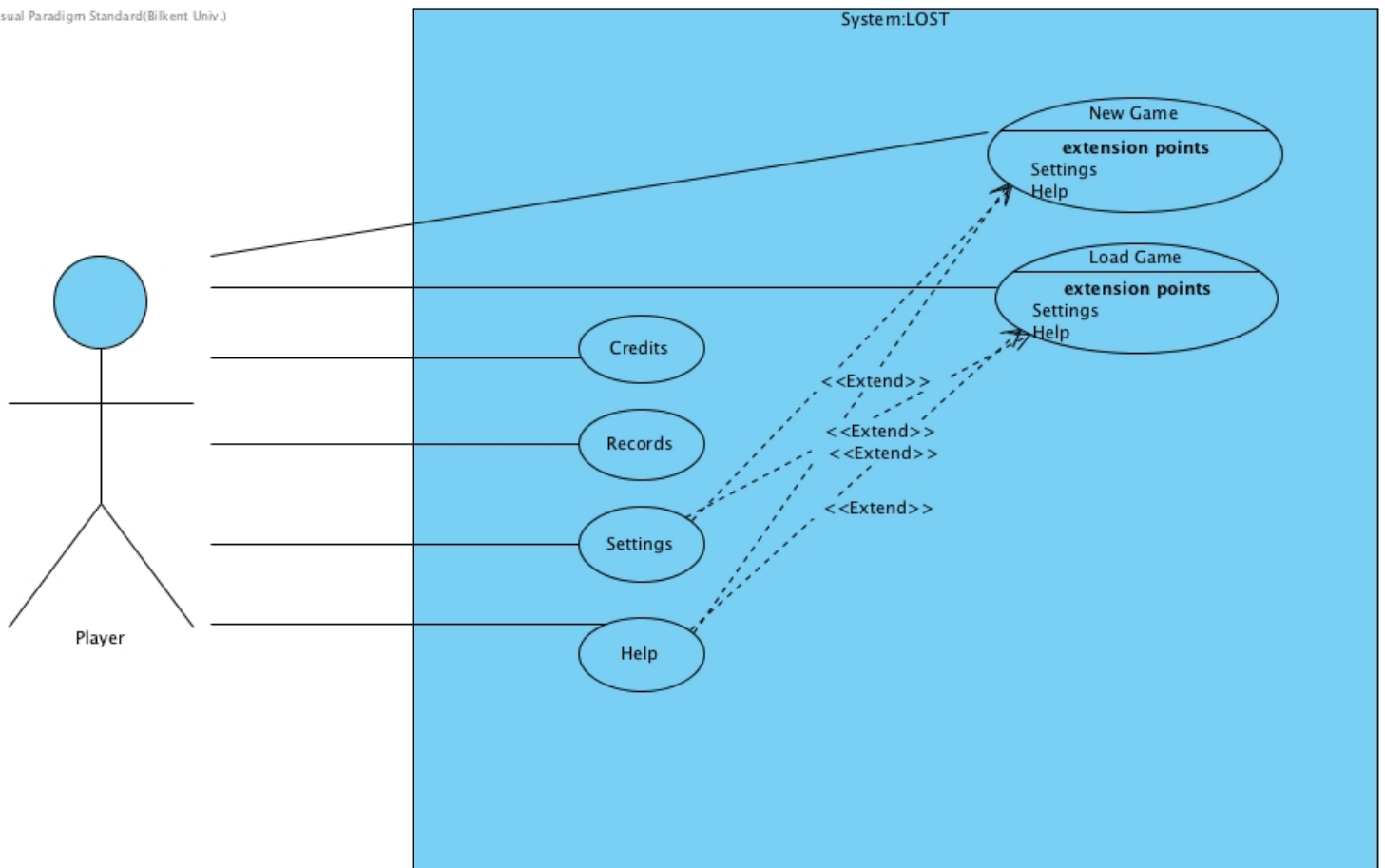
4) System Models

4.1) Use Case Model

Figure 4.1 - Shows the use case model of LOST

4.1.1) Use Case Descriptions

Use Case Name: New Game

Participating Actor: Player

Stakeholders and Interests: Player may want to create a new survival game.

Pre-condition: User must sign in and create a new account.

Post-condition: -

Entry condition: Player opens game and starts game by clicking relevant section.

Exit condition:
  • Player clicks on back button in Game screen,OR
  • Player dies, OR
  • Player succeeds to survive and escapes from island.

Main flows of events:
  1. Player enters a nickname and starts the game.
  2. System loads default settings and starts the game.
  3. Some guidelines end up on screen.
  4. Player interacts with objects around him/her and uses them correctly.
  5. Player completes the one of story events, achieve to survive.
  6. The obtained score is demonstrated on screen at the end of the game.

Alternative flows of events:

  4.1. Player does not use items correctly.

      a. Thirst, starvation becomes high level and the healthiness of user starts to drop out.

      b. Interactions become slower.

      c. Player dies because of mis-use, deficiency of items.

  4.2. Player uses items correctly.

      a. Player handles with starvation, finding good supplies and health condition is good enough.

      b. Player has troubles on interaction between objects, he/she always makes wrong choices in matches between characters.

      c. Player dies because of enemy attack.

4.3. Player gets bored and may want to close the game.

        a. The game instructions are loaded and whenever the user loads the game, player continues from the position where he/she left.

Use Case Name: Load Game

Participating Actor: Player

Stakeholders and Interests: Player wishes to continue previous uncompleted game.

Pre-condition: User must have a record that he/she plays this game before and not completed yet.

Post-condition: -

Entry condition: Player should click on load game button in main screen.

Exit condition:
- Player clicks on back button in Game screen,OR
- Player dies, OR
- Player succeeds to survive and escapes from island.

Main flows of events:
    1. Player starts the game in recorded location.
    2. Follows same steps defined new game use case.
    3. Player successfully or unsuccessfully complete the game.

Alternative flows of events:
    3. Player does not complete the game.
        a. Game instructions are loaded again and waiting for next connection of player.

Use Case Name: Credits

Participating Actor: Player

Stakeholders and Interests: Player can indicate the credits

Pre-condition: Player has to be in main menu.

Post-condition: -

Entry condition: User clicks on "Credits" in main menu.

Exit condition: User clicks on back button on game screen.

Interests: Player may want to review developers, contents and sources of game.

Main flows of events:
      1. System shows the text-based content for credits section.

Alternative flows of events:
      1. Player wants to return back to the main menu.
            a. Player clicks on back button on screen.
            b. System renders main menu screen.

Use Case Name: Records

Participating Actor: Player

Stakeholders and Interests: Player can review the scoreboard and indicate competition between users.

Pre-condition: Player has to be in main menu.

Post-condition: -

Entry condition: User clicks on "Records" in main menu.

Exit condition: User clicks on back button in Game screen.

Main flows of events:
      1. Player observes the scores of users and the leaderboard of game.

Alternative flows of events:
      1. Player may want to return to main menu.

            a. Player clicks on back button on screen.
            b. System renders main menu screen.

Use Case Name: Settings

Participating Actor: Player

Stakeholders and Interests:
      • Player wishes to close the sound effects.
      • Player wishes to change the game panel size.

Pre-condition: Player has to be in main menu or the game must currently be started.

Post-condition: -

Exit condition: User clicks on back button after reviewing the game settings.

Entry Condition: Press Settings section in main menu.

Main flows of events:

    1. Player picks availability of sounds and game panel size.

    2. Press apply button.

Alternative flows of events:

    2. Player may give up changing the game settings.

        a. Player directly returns main menu without applying the changes.

    2. Play may forget to apply the changes.

        a. Previous system requirements become active in this case.

Use Case Model: Help

Participating Actor: Player

Stakeholders and Interests: User may want to be informed about features of game, understand the game mechanics and take sort of guidelines, hints about "Lost".

Pre-condition: Player must be in main menu.

Post-condition: -

Entry Condition: Player selects About section in main menu.

Main flows of events:

    1. User opens the game and in order to take guidelines of game, opens Help section.

    2. Understand game mechanics and concepts.

    3. Turns back main menu.

    4. Starts game.

5. Player manages the character in the direction of given guidelines.

Alternative flows of events:

    1. User starts game without connecting about section in main menu.

    2. Does not understand game mechanics.

    3. Turns back main menu.

    4. Opens about section and tries to understand the guidelines of game.

    5. Continue game where he/she left.