Implementation Plan for System TwitterNethack

Assignment in the course PA1435 Object Oriented Design

April 11th, 2017

Course project on GitHub

Author Name	Social Security Number	Thinking	Writing
Simon Lindgren	960815-1131	20	25
Peter Meunier	951230-0113	10	5
Viktor Enfeldt	960101-9616	25	25
Tobias Fast	961011-6676	25	20
Samuel Asp	960125-1458	20	25

1   System Description

A multi-player dungeon crawler that uses information from Twitter to generate rooms and to populate them with items and characters. The players will be able to freely move around in the room they're in, pick up, drop, equip, craft, and use items, attack monsters and other players, move between rooms in the labyrinth, and chat with other players.

The types of items will range from potions and rocks, to sophisticated weapons like swords and crossbows (the latter of which will require ammunition). It will be possible to craft certain items (such as arrows) by combining two or more other items.

2   Prioritized List of Use Cases

2.1   Motivation for Priorities

Use cases that are either essential for the basic functionality of the game or the main gameplay mechanics are prioritized. This is done so that a basic prototype can be developed and tested as quickly as possible to see whether or not the concept behind the game is good/fun.

Use cases that aren't essential to the core gameplay or to other use cases are not prioritized since they aren't as important to the rest of the development as the previously mentioned use cases.

2.2   Use Cases

1. Generate room

2. Create character

3. Move character

4. Enter room

5. Attack enemy

6. Pick up item

7. Equip weapon

8. Drop item

9. Use item

10. Inspect item

11. Inspect character

12. Drop loot

13. Use potion

14. Craft item

15. Interact

16. Multi-player

3   Estimated Velocity Per Iteration

MAX: 7
MIN: 5
AVG: 6

4   Implementation Plan

4.1   Motivation for Implementation Plan

The entire system depends on the existence of rooms so the creation of those comes first. To move through the rooms a movable character is needed so that will also be implemented in the first iteration.

Interaction with other characters on a surface level (attack, inspect, etc) is a core part of the gameplay so that will be implemented early on in order to be able to test whether or not the game is actually fun.

To expand on the aforementioned basic gameplay mechanics more items and characters are needed so more and more of those will subsequently be added.

The multi-player aspects of the game are not essential to most of the development of the game and its mechanics so that will be implemented fairly late in the development.

4.2   Iterations and Use Cases

4.2.1   Iteration 1

Generate room (no items).
Generate character.
Move character.
Enter room.
Attack enemy.

4.2.2   Iteration 2

Generate room (with items).
Pick up item.
Drop item.
Drop loot.
Use item.
Equip weapon.

4.2.3   Iteration 3

Implement more items and non-playable characters.
Use potion.
Craft item.
Inspect character.
Inspect item.
Interact.
multi-player.