

## User Stories for TwitterNethack

Assignment in the course PA1415 Programvarudesign <2017-04-11>

Author: Morgan Lexander, Henrik Nilsson, Magnus Nyqvist och Victor Olsson.

Author Name	Social Security Number	Thinking	Writing
Morgan L.	971107-xxxx	25%	25%
Henrik N.	970416-xxxx	25%	25%
Magnus N.	940429-xxxx	25%	25%
Victor O.	970612-xxxx	25%	25%

## System description

The system is going to be based around the classic game NetHack but in 2D graphics. NetHack is a sort of adventure and roleplaying game where you create a character and goes through random generated dungeons. In the dungeons you can meet various different monsters and collect items. The final goal is to find the amulet of Yendor and escape with it alive.

Nethack is a very advanced game and because of this and the limited time we are given, our version of NetHack is not going to include all of the features that the original have. The functionalities we are aiming to implement:

- An opening screen where you can start a new game or join an existing one.
- A screen where you can setup the character.
- Monsters and creatures that freely move around the dungeons.
- Different items to pick up and drop.
- Multiplayer support.
- Generate dungeons with help of twitter.

The dungeons in our NetHack version is going to be generated by the Twitter API. Its job is to read certain keywords in Twitter posts and use these to create specific types of monsters and items for every room in the game. When you enter a new cave, your system shall briefly display information about the “owner” of the cave.

## High-level Epics:

The priority of these epics and user stories are based around the minimal business value and the risk that some offer. Since the game is inspired by Nethack, which have a genre called roguelike we focus somewhat on the core elements of that genre. A typical roguelike game offer procedurally generated maps, in our case we use twitter to generate maps. Another key element is the interaction with other characters. One thing that is huge in the genre is the control the player has over their character. That is why we prioritise the customization. Next step is items. With each epic we get closer to the full roguelike experience but if we would run out of time we could still call it roguelike.

### 1. Epic (gameplay)

As a player I want dynamic gameplay so that everytime I play I get an unique experience.

### 2. Epic (customize)

As a player I want to customize my character so that I can identify and feel a deeper connection to my avatar.

### 3. Epic (item)

As a player I want items to collect so that I can progress and become stronger.

## All Epics and User stories:

### 1. Epic (gameplay)

As a player I want dynamic gameplay so that everytime I play I get an unique experience.

#### 1.1. Epic (gameplay.map)

As a player I want a dynamic map so that I get to explore the world and the gameplay becomes progressive.

##### 1.1.1. User story (gameplay.map.generation)

As a player I want the map to be generated by the Twitter API so that every cave is unique and intriguing.

##### **Condition of satisfaction:**

The caves are generated by the Twitter API.

##### 1.1.2. User story (gameplay.map.interact)

As a player I want to be able to interact with the map so that I feel that I can actually affect my gameplay.

##### **Condition of satisfaction:**

The player are able to make changes to the rooms.

##### 1.1.3. User story (gameplay.map.random)

As a player I want unpredictable elements in the game so that my progression don't become linear.

##### **Condition of satisfaction:**

Interactions with the game have randomised effects.

## 1.2. **Epic (gameplay.characters)**

As a player I want to meet different characters and creatures so that my experience gets more lifelike and interesting.

### 1.2.1. **User story (gameplay.character.enemies)**

As a player I want different types of enemies which requires different strategies to slay, so that the combat becomes dynamic.

#### **Condition of satisfaction:**

There are multiple types of enemies that interact in various ways.

### 1.2.2. **User story (gameplay.character.friendly)**

As a player I want friendly NPC's so that the experience becomes more lifelike and my decisions have a deeper impact on the game world.

#### **Condition of satisfaction:**

The player will be able to interact with friendly NPC's.

## 2. **Epic (customize)**

As a player I want to customize my character so that I can identify and feel a deeper connection to my avatar.

### 2.1. **User story (customize.gender)**

As a player I want to choose gender for my character so that I can get more immersed with my avatar.

#### **Condition of satisfaction:**

The player will be able to play as a man or as a woman.

### 2.2. **User story (customize.raceclass)**

As a player I want to choose race and class for my character so that I can get a more customized gameplay.

#### **Condition of satisfaction:**

The player will be able to choose race and class.

### 2.3. **User story (customize.attributes)**

As a player I want to have different stats/attributes on my character so that the gameplay is impactful and interesting.

#### **Condition of satisfaction:**

The player will have attributes which affects the player.

## 3. **Epic (item)**

As a player I want items to collect so that I can progress and become stronger.

### 3.1. **Epic (item.weapon)**

As a player I want wieldable weapons so that I get more realistic approach to combat.

#### 3.1.1. **User story (item.weapon.melee)**

As a player I want melee weapons so that I can slay my foes from a close-up perspective which give me an intense experience.

#### **Condition of satisfaction:**

The player will be able to use melee weapons.

### **3.1.2. User story (item.weapon.ranged)**

As a player I want ranged weapons so that I can kill my foes from a safe distance.

#### **Condition of satisfaction:**

The player will be able to use ranged weapons.

### **3.2. User story (item.buffs)**

As a player I want to have the ability to give my character temporary bonuses so that I can make interesting game decisions.

#### **Conditions of satisfaction:**

The player will be able to use temporarily buffs.

### **3.3. User story (item.armor)**

As a player I want to have equipable armor so that I can protect my character from damage.

#### **Condition of satisfaction:**

The player gain attributes based on the equipped armor.

### **3.4. User story (item.resources)**

As a player I want resources to collect so that I can see keep track of my progress.

#### **Condition of satisfaction:**

The player will be able to collect gold.

## **4. Epic (client)**

As a player I want a Game Client so that I can start a new game or join an existing one.

### **4.1. User story (client.serverlist)**

As a player I want a server list over currently active game sessions so that I can pick a session that fit my preference.

#### **Condition of satisfaction:**

A server list with descriptions for each available server.

### **4.2. User story (client.singleplayer)**

As a player I want to start a singleplayer session so that I can play alone.

#### **Condition of satisfaction:**

The player will be able to start a singleplayer session and move around.

## **5. Epic (multiplayer)**

As a player I want to meet real players which I can interact with so that the game becomes competitive and social.

### **5.1. User story (multiplayer.kill)**

As a player I want to kill other players so that I can take their items and become stronger.

#### **Condition of satisfaction:**

Players are able to kill each other.

### **5.2. User story (multiplayer.group)**

As a player I want to team up with other players so that we can overcome the

challenges the game has to offer.

**Condition of satisfaction:**

Players are able to group up and collaborate.

**5.3. User story (multiplayer.trade)**

As a player I want to trade with other players so so that we can overcome the challenges the game has to offer.

**Condition of satisfaction:**

Players are able to trade with each others.

## User stories in prioritised order:

The first eight user stories come with an explanation about their reason for the placement.

The rest is placed accordingly to their business value. The user stories which is higher up have a greater purpose to the system and/or takes time to implement. The ones at the bottom is seen as a “bonus” to the system.

Order motivation:

**1. gameplay.map.generation:**

The main focus is to have the map generation with Twitter done right. This is something we as a group agreed is the most risk full story of them all. If we are not able to succeed with this step, there may be a decision whether to continue or not with the project.

**2. client.singleplayer:**

Being able to start a singleplayer session is essential to be implemented first. The case is being able to showcase the game early for both the customer and stakeholders.

**3. customize.attributes:**

Having a working statistics system is a core in the nature of this game. Every entity will have some kind of statistics therefore it is placed rather early in the implementation of this project.

**4. gameplay.character.enemies:**

With a working statistics system, we can continue to implement enemies. The user story do request to have different strategies with each enemy. This can be resourceful and is placed accordingly.

**5. item.weapon.melee**

With the implementation of weapons, there must be some enemies to fully test and balance. That is why this user story is prioritised after the implementation of enemies.

**6. gameplay.map.interact**

For the opportunity to fully implement an interaction system that works with the map, the map must first be implemented. Although we did take into account to implement the above first, because the game can be somewhat enjoyable without an interaction system. Another reason for the placement is that it can come with an risk from the conjunction with Twitter.

**7. item.resources**

Resource system is essential for a game like this. Furthermore, more systems will benefit by having this system implemented here.

**8. gameplay.character.friendly**

Friendly Non-Player character won't be hard to implement because we already have

implemented enemies. They are essentially the same but not aggressive. Some may be able to trade with the player, so the decision to implement a resource system before makes sense.

9. **gameplay.map.random**

10. **item.armor**

11. **item.buffs**

12. **item.weapon.ranged**

13. **customize.raceclass**

14. **client.serverlist**

15. **multiplayer.group**

16. **multiplayer.kill**

17. **multiplayer.trade**

18. **customize.gender**