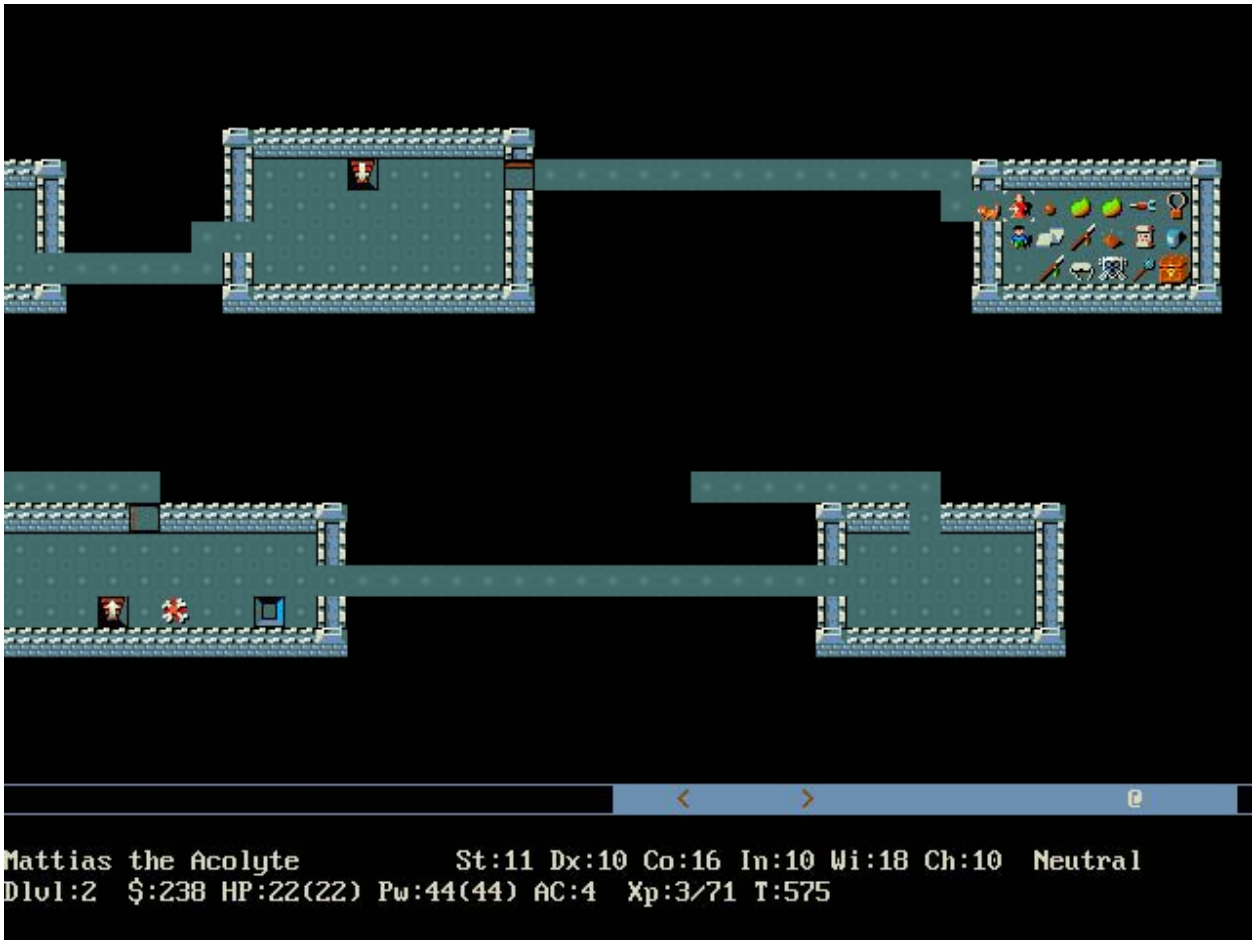


3 Week Roll Call Assignment

Assignment in the course PA1435 Objektorienterad Design

05/04/2017

Henry Bergström	19920924-3113	All	All
Daniel Fredriksson	19941206-3415	All	All
Oliver Glandberger	19940731-2299	All	All

Blekinge Tekniska Högskola

System Description

The goal of the system is to allow the player(s) to create characters to explore and survive randomly generated labyrinth-like dungeons. Within the dungeon players can interact with NPC's, items, the environment, and even other players within multiplayer.

The algorithm that controls all random generation uses information taken from a specific twitter feed. This twitter feed will be chosen by the player before beginning the game.

Use Cases

1. Character Creation

The player creates a character. The character's 'head', 'torso' and 'legs' are chosen from a limited selection. Following this, the player chooses a name and the character creation is finished. The system prepares and stores the character for the game.

- **Preconditions:**
  Player has launched the game, no other character has been created.

- **Main Course of Events:**
  Player selects Head/Torso/Legs parts of the character and inputs a name.

- **Alternative Flow of Events:**
  Player chooses to randomize the appearance of the character and inputs a name.

- **Special Requirements:**
  None.

2. Create A Singleplayer Game

The player provides a twitter link. The player starts the game and the system begins generating content based on the twitter link. The game starts.

- **Preconditions:**
  Player has created a character.

- **Main Course of Events:**
  Player inputs a twitter link and chose to 'Start Game'. Content is generated and game starts.

- **Alternative Flow of Events:**
  Player inputs an invalid twitter link and chooses to 'Start Game'. An error message appears, displaying: 'Invalid twitter link!'.

- **Special Requirements:**
  A Twitter link.

3. Create A Multiplayer Game

The player provides a twitter link and a lobby name. All the players in the lobby share a chat. The player is presented with the option to 'kick' other players, 'disband the lobby', or 'start' the game. The player starts the game and the system begins generating content based on the twitter link. The game starts.

- **Preconditions:**
  Player has created a character.

- **Main Course of Events:**
  Player inputs twitter link and a lobby name. Lobby is created. Two other players join the game and the 'host' player starts game.

- **Alternative Flow of Events:**
  Player inputs twitter link and a lobby name. Lobby name is busy and player inputs another lobby name. Lobby is created. Another player joins game, the 'host' player kicks the newer player and disbands lobby. Player is returned to previous menu.

- **Special Requirements:**
  Twitter link, Lobby Name.

4. Join A Multiplayer Game

The player specifies a specific lobby name. The player joins the lobby and is presented with a lobby chat. The player chats with other lobby members. The host starts the game and the system begins generating content based on the twitter link. The game starts.

- **Preconditions:**
  Player have created a character.

- **Main Course of Events:**
  Player A inputs a lobby name called 'ourGame'. Player B chose to 'Join a Multiplayer game'. A list is presented with available lobbies. Player B marks the lobby named 'ourGame' and chose to 'Join Game'. Player A greets player B using chat and then chose to 'Start Game'.

- **Alternative Flow of Events:**
  Player B is trying to join a lobby. Max amount of players in met and player B is displayed with a prompt saying: 'Lobby is full'.

- **Special Requirements:**
  None.

5. Player Movement

The player walks north. The top-down camera follows the player north.

- **Preconditions:**
  Player is inside a game. The space the player is trying to move to is 'available'.

- **Main Course of Events:**
  Player tries to move the character in direction specified by player's input. Character moves.

- **Alternative Flow of Events:**
  Player tries to move character in the direction specified by player's input. Wall is in the way, character does not move.

- **Special Requirements:**
  The character is not encumbered.

6. Player-Environment Interaction

The generated labyrinth causes the player to walk south. The player finds a chest, opens it and receives some items. The player's inventory is updated. The player avoids falling rocks but triggers a spike trap. The player's health is updated negatively. The player finds the current level's ladder, and continues to the next level.

- **Preconditions:**
  Player is inside a game.

- **Main Course of Events:**
  Player encounters a ladder and chooses to use(climb) the ladder. The next level is loaded and started for the player.

- **Alternative Flow of Events:**
  Player walks into a concealed trap and the character loses life.

- **Special Requirements:**
  There is something to interact with.

7. Player-Hostile NPC Interaction

The player encounters a hostile monster. Wielding a weapon, the player maneuvers around the monster attacking it and dodging incoming attacks. The player deals damage to the monster. The monster's health is updated. The monster runs out of health and dies.

- **Preconditions**
  Player is inside a game.

- **Main Course of Events**
  Player encounters a hostile monster. Using attack button, player damages hostile monster. Hostile monster deals damage to player. Both health bar is updated. Hostile monster's health is reduced to zero and disappears.

- **Alternative Flow of Events**
  Player encounters a hostile monster. Using attack button, player damages hostile monster. Hostile monster deals damage to player. Both health bar is updated. Player's health is reduced to zero and screen shifts color with a prompt saying: 'Game over'.

- **Special Requirements**
  Player is not in a safe zone.

8. Player-Unknown NPC Interaction

The player encounters a neutral wizard. A dialogue is displayed between the wizard and the player. The player is presented with a list of responses. The player selects an option which pleases the wizard. The player receives an item. The inventory is updated. The player initiates another dialogue with the wizard. The player selects a new response. The wizard becomes triggered and turns into a hostile NPC.

- **Preconditions:**
  Player is inside a game.

- **Main Course of Events:**
  Player encounters an unknown NPC. Descriptive text of the unknown NPC is displayed. Contrasting dialogue responses are presented. The player chooses one of them which yields the player's character an item.

- **Alternative Flow of Events:**
  Player encounters an unknown NPC. Descriptive text of the unknown NPC is displayed. Contrasting dialogue responses are presented. The player chooses one of them which turns the unknown NPC into a hostile NPC.

- **Special Requirements:**
  An event triggered the appearance of the unknown NPC.

9. Player-Player Interaction

Player 1 uses a health potion on a nearby player 2. Player 1's inventory is updated and player 2's health is updated.

- **Preconditions:**
  Player is inside a multiplayer-game.

- **Main Course of Events:**
  Player 1 uses a health potion on Player 2. Player 1's inventory is updated and Player 2's health is updated.

- **Alternative Flow of Events:**
  Player 1 attacks Player 2 using an equipped weapon. Player 2's health is reduced and updated.

- **Special Requirements:**
  Target player is within range of the action chosen.

10. Player Inventory

The player opens their inventory. The player destroys one item, the player equips a sword, and the player uses a health potion. After each interaction, the inventory is updated.

- **Preconditions:**
  Player is inside a game.

- **Main Course of Events:**
  Player opens the inventory, equips a sword, and uses a health potion. After each interaction, the inventory is updated.

- **Alternative Flow of Events:**
  Player opens the inventory, and exits out of it; it is empty.

- **Special Requirements:**
  Player is not currently in a dialogue interaction of any kind (Player-Player chat or Player-Unknown NPC dialogue).

11. In-game Chat

The player initiates the chat box, types a message, and confirms the message. The message is broadcasted to all players. Two other players respond. The chat box is updated for each message.

- **Preconditions:**
  Player is inside a game.

- **Main Course of Events:**
  Player A initiates the chat box and types a message. Player B reads the message and responds. All players inside the Multiplayer-game read the messages.

- **Alternative Flow of Events:**
  Player A initiates the chat box and types a message. Player A is inside a Singleplayer-game; there is no response.

- **Special Requirements:**
  None.

12. Trading

The player approaches another player and initiates a trade. A trade window is presented to both players. Both players add items to their trade windows. Both players 'accept' and the trade is finished. Both players' inventories are updated.

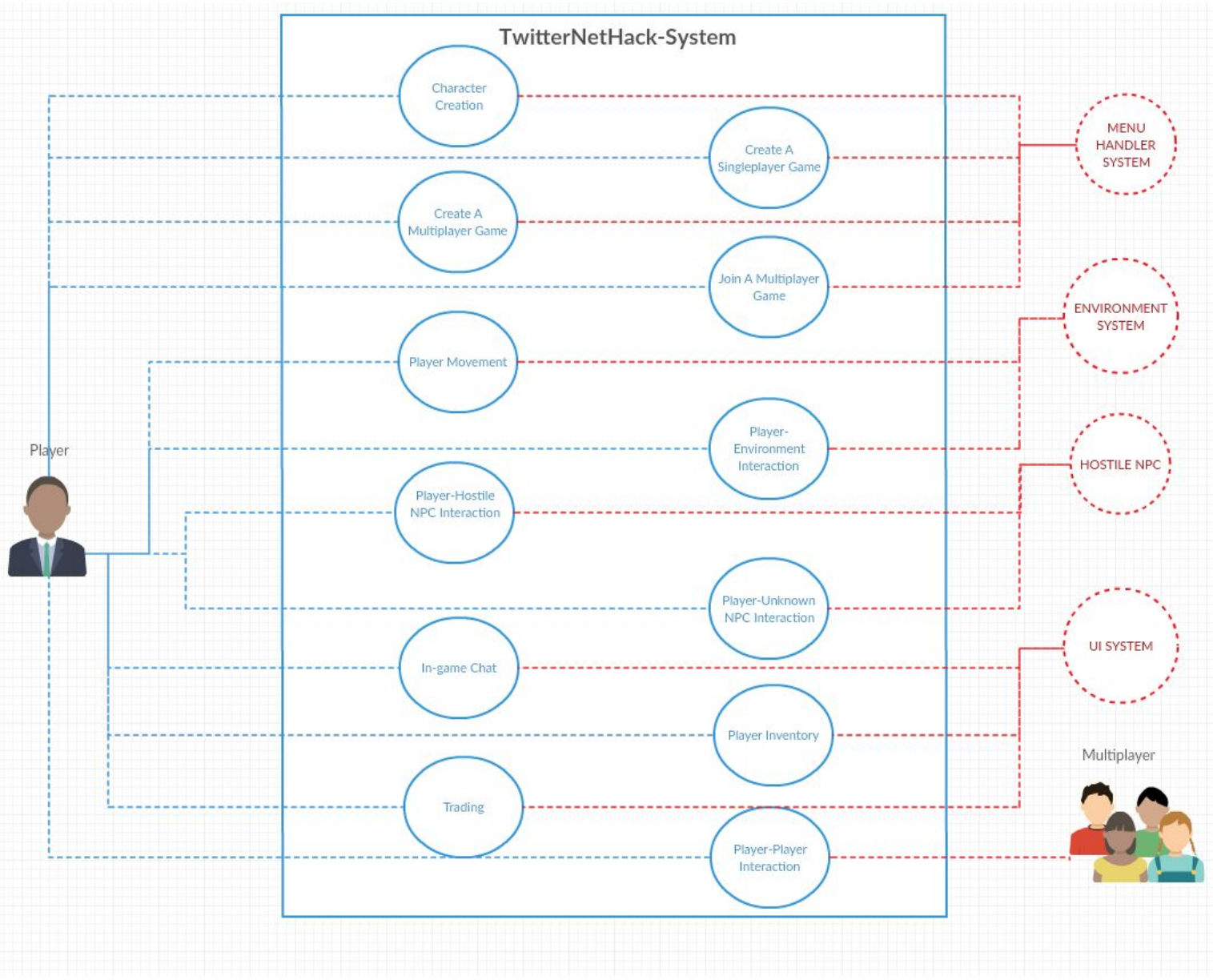
- **Preconditions:**
  More than two players are inside same game.

- **Main Course of Events:**
  Player A approches player B and initiates a trade. A trade windows is presented to both players. Player A adds an item to his trade window from their inventory. Player B adds an item to their trade window from their inventory. Both players 'accept' and items have switched inventory.

- **Alternative Flow of Events:**
  Player A approches player B and initiates a trade. A trade windows is presented to both players. Player A adds x amount of item to their trade window from their inventory. Player B adds less than x amount item to their trade window from their inventory. Both players 'accept'. Player B's inventory is full and cannot carry all items. Trade is cancelled.

- **Special Requirements:**
  Both players are within range.

Use Case Overview



Created using Creately.com

Implementation Plan

The following implementation plan and prioritization has been built using the MoSCoW ranking technique.

Must Have

1. Player Movement **(Story Points: 1)**
2. Player-Environment Interaction **(Story Points: 3)**
3. Player-Hostile NPC Interaction **(Story Points: 2)**

Should Have

1. Character Creation **(Story Points: 2)**
2. Create A Singleplayer Game **(Story Points: 2)**
3. Player Inventory **(Story Points: 2)**

Could Have

1. Player-Unknown NPC Interaction **(Story Points: 5)**
2. Create A Multiplayer Game **(Story Points: 3)**
    a. Join A Multiplayer Game *[Ranking: Must Have, assuming 2]* **(Story Points: 1)**
    b. Player-Player Interaction *[Ranking: Must Have, assuming 2]* **(Story Points: 5)**
    c. In-game Chat *[Ranking: Should Have, assuming 2]* **(Story Points: 2)**
    d. Trading *[Ranking: Should Have, assuming 2]* **(Story Points: 2)**

Won't Have

None.

In order to create a minimum viable product, all of the 'Must Have' points will be implemented. These 3 points (player movement, the player's interaction with the environment, as well as some kind of challenging aspect in the form of hostile NPCs) are the key components for this product.

NOTE: *'Character Creation' and 'Create A Singleplayer Game' are currently in the 'Should Have' list. This implies that if such user-driven features aren't implemented, then the system will automatically create the character and the game upon launching the application.*

During our first iteration, the 'MIN' value regarding the amount of story points we will have time to complete will be 6 points. This will ensure that we get all 3 core components ready. Our 'MAX' value will initially be set to thirty story points; resulting in every single use case being implemented. With our 'MIN' and 'MAX' in mind, we decided that our 'AVERAGE' amount of story points per iteration should be 6. This means that we'd need a total of 2 iterations to complete our 'Must Have' and 'Should Have' points, or 5 iterations if we also would like to complete our 'Could Have' points.

Iteration 1

- Player Movement **(Story Points: 1)**
- Player-Environment Interaction **(Story Points: 3)**
- Player-Hostile NPC Interaction **(Story Points: 2)**

Iteration 2

- Character Creation **(Story Points: 2)**
- Create A Singleplayer Game **(Story Points: 2)**
- Player Inventory **(Story Points: 2)**

Iteration 3

- Player-Unknown NPC Interaction **(Story Points: 5)**

Iteration 4

- Create A Multiplayer Game **(Story Points: 3)**
- Join A Multiplayer Game *[Ranking: Must Have, assuming 2]* **(Story Points: 1)**
- In-game Chat *[Ranking: Should Have, assuming 2]* **(Story Points: 2)**

Iteration 5

- Player-Player Interaction *[Ranking: Must Have, assuming 2]* **(Story Points: 5)**
- Trading *[Ranking: Should Have, assuming 2]* **(Story Points: 2)**