

---

# System Component Requirements & Design

## Trouble Seekers

Rev 1

**Prepared by**

**Group Name:** Buttered Waffle

Andrew Raudys  
Nathan Mainville  
Matt Leduc  
Sean McLellan

0973268  
0960351  
0963077  
0960351

**Instructor:** Dr. Carette

**Course:** SFWR ENG 4GP6

**Date:** Feb. 24, 2013

# Table of Contents

## [1 Theme/Quest](#)

### [1.1 Abstract](#)

### [1.2 Concrete](#)

## [2 World](#)

### [2.1 Abstract](#)

### [2.2 Concrete](#)

## [3 Story](#)

### [3.1 Abstract](#)

### [3.2 Concrete](#)

## [4 Gameplay](#)

### [4.1 Abstract](#)

### [4.2 Concrete](#)

## [5 Gameplay Modes](#)

### [5.1 Abstract](#)

### [5.2 Concrete](#)

## [6 Core Mechanics](#)

### [6.1 Abstract](#)

### [6.2 Concrete](#)

## [7 Characters](#)

### [7.1 Player](#)

#### [7.1.1 Abstract](#)

#### [7.1.2 Concrete](#)

### [7.2 NPCs](#)

#### [7.2.1 Abstract](#)

#### [7.2.2 Concrete](#)

## [8 Presentation](#)

### [8.1 Abstract](#)

### [8.2 Concrete](#)

## [9 Levels](#)

### [9.1 Abstract](#)

### [9.2 Concrete](#)

## [10 Concrete Components](#)

### [10.2 Textures](#)

### [10.3 Sounds](#)

### [10.4 Particle Emitters](#)

### [10.5 Animation Sets](#)

### [10.6 Static Meshes](#)

### [10.7 Skeletal Meshes](#)

### [10.9 Kismet Sequences](#)

### [10.10 Matinee Sequences](#)

### [10.11 Map Geometry](#)

### [10.12 Scripts](#)

# 1. Theme/Quest

## 1.1 Abstract

The theme of the game is that the user is an orphan searching for his best friend after she goes missing. The player finds out that other members of villages nearby have also coincidentally gone missing, and tries to find out what is actually happening regarding the disappearances. There is a mysterious door that requires artifacts to unlock the door (like keys), so the main quest for the player is to find all these artifacts scattered around the world, in order to find out what happened to his best friend and the villagers.

### Design:

The main Quest of the game will be implemented using a series of Kismet triggers which will check to see if the player has gathered all the required artifacts in order to open the final door, by using this trigger we allow the player to complete the dungeons in the order they desire.

## 1.2 Concrete

The goal of the game will be explained using various matinee sequences (10.10.a-c) which tie into the story. Villagers around town can also give insight into the missing people (10.12.j).

# 2. World

## 2.1 Abstract

The world is inspired by medieval-era design, specifically from the European cultural influences. The game also contains elements of fantasy which draw upon other fantasy related writings/art (i.e. Tolkien, Zelda). The world is designed to have a cartoon-like quality, with proportions that are exaggerated and colour schemes that are bright. The world is in 3-Dimensions, but the user can only navigate on one plane because the game navigation is similar to dungeon crawlers like Diablo or Legend of Zelda. The world contains a mountain off the to the west of the main village, the mysterious door is located on the side of the mountain. The sound effects contained in the world fit the overall theme of the game and are cartoon-like as well as vivid.

### Design:

The map layouts of our game will be created in the UDK, the items and props which populate the world will be mainly static meshes which will be created in Blender. Kismet will be

used to create triggers within the world such as spawning enemies and interacting with other NPCs.

## **2.2 Concrete**

The world will be made up of several UDK maps (10.9, 10.11). There will be pathways that allow the player to travel between them, and each map will contain a number of static meshes (10.6.k-l) to compliment their layout.

# **3. Story**

## **3.1 Abstract**

The game story is separated into two categorization: the missing people story-arc and world lore. The player is told about the main story related set of dungeons/temples (at the beginning of each episode by a friendly NPC) that may contain an artifact(required to open the mysterious door). To progress through the story, the player must investigate each of these dungeons and find the artifact, when all the artifacts are found the user can open the door (but since our content is episodic, the player will not be able to open the door in this game). The dungeons can be done in any order; however each time the user completes a certain number of dungeons, more villagers will be missing and the quests to explore specific lore expansion dungeons will become available from certain NPCs (player-initiated conversation and completely optional). The game's side-quests involve the player going into other dungeons/temples but they do not contain story-required artifacts, they are mainly for skill advancement, as well as expanding the lore of the world that may or may not be related to why people are missing. The main way lore of the world can be expanded is through item descriptions, excerpts from in-game books, and scripted NPC dialog (mostly occurring in villages).

### **Design:**

The first part of our story, the "missing people story-arc" will be implemented using Kismet scripts which will allow the player to communicate with NPCs to solve the mystery of the missing people. These scripts simplify the implementation of dialog trees which means we can create dialog trees with more layers or paths. The second part of our story, the world lore, will consist of Kismet triggers to spawn certain enemies in specific locations, allowing the player to explore the map and discover what is hidden inside our world. Some of the triggers will allow the player to discover some information which will help find the reason for the missing people.

## **3.2 Concrete**

Matinee sequences will be used to convey the story to the player (10.10.a-c).

## 4. Gameplay

### 4.1 Abstract

There will be one plot-centric dungeon in the first episode of the game. It will follow a short tutorial dungeon designed to get the player used to the game's mechanics. Each dungeon is themed (i.e. an elemental theme like fire). The dungeons are designed to be maze like (multiple ways to get to the same point, but also filled with dead-ends), with many traps designed to either kill or injure the player. The dungeons contain various enemies, also designed to fit the theme of the dungeon. At the end of each dungeon, the player finds an object of interest (either an artifact for main story dungeons, or a powerful tablet to advance their skills and abilities).

#### Design:

The gameplay is centered around solving puzzles as well as avoiding traps or destroying enemies. The levels will be designed to try and surprise the player with a different challenge, be it different types of enemies or a new type of puzzle, to make the player consider many possible strategies. As explained in the Core Mechanics section, the player can choose to assign tablets to different stats, meaning he can change from a lot of Health to a lot of Damage with the click of a button, which is why we will have puzzles and traps which will push the player to use the ability to change attributes quickly.

### 4.2 Concrete

All the UDK maps (10.11) designed will be used for gameplay. Each of the maps will be populated by NPCs (10.2.d-aa, 10.7.a-d, 10.12.d-f) and treasure (10.7.e, 10.12.h), which use various textures (10.2.d-aa), animations (10.5.a-n) and sounds (10.3.l-m,t,u,d). The dungeons themselves will use various textures (10.2.bb,cc,ff-ii) and sounds (10.3.o,r).

## 5. Gameplay Modes

### 5.1 Abstract

There are 2 modes of gameplay: dungeon play and exploration. In dungeon play, the dungeons can be explored by the player (multiple paths with dead-ends as previously mentioned) with a possibility to find lore related items but at the same time the paths of the dungeon are filled with enemies and traps. The dungeon play mode is completed when the player gets the main object of interest in that specific dungeon (the player is returned to outside the dungeon entrance). In exploration mode, the player is able to initiate conversation with NPCs around the different villages, as well as interact with certain objects that offer lore descriptions. The player can also explore the world without looking for quests and attack enemies which will be located in the wild.

When conversation is initiated, a dialog box will appear at the bottom of the screen, which takes up approximately a third of the screen. The dialog of the person speaking (indicated in the top left corner of the box) will appear in the center of the box, and can be advanced by pressing a specific key. When speech is complete, any player options to advance the conversation will appear in the box. The choice the player makes for each specific conversation, will branch the conversation in a specific way following a conversation tree for each NPC capable of conversation.

The inventory system is different than what is found in most RPGs. The player is given a limited number of inventory slots where they can store items they acquire after killing enemies. Certain enemies will drop a tablets which will affect the players stats once it is equipped in the tablet slots.

## **Design:**

The modes of gameplay (dungeon play and exploration) are handled in slightly different ways. In dungeon play, the player is able to attack most NPCs in the various dungeons built with the UDK level designer. The dungeons will have no more than 3 possible branching paths per path to reduce design complexity for both the player (reduce navigation confusion) and for the designers (making some changes to the level will not require too much overhead). Furthermore, the number of traps per level (medium sized) will be within the realm of 10-15, enough that the levels are challenging but manageable for the player to remember upon playthroughs. The traps themselves will be handled using kismet event triggers. The combat actions are initiated using the mouse controls (left click to attack). However, in exploration mode, conversations are initiated with friendly NPCs by being within a specific radial distance and using an interact action ( bound to a specific keyboard stroke). The player will also be unable to attack the villagers while in towns, preventing players from possibly halting progress in the game. Additionally, the player will be able to interact with lore objects by being within a certain radial distance from an object and using the same interact action. These interactions will be determined by specific .uc files holding scripted dialogue and AI behaviour for each individual friendly NPC and a general AI behaviour for each grouping of enemy NPC. When engaging with friendly NPCs, the dialog boxes that appear will be rendered using scaleform UI tools, allowing the player to choose decisions through the use of Actionsript 2 events and relaying the decisions to interact with specific .uc files. The player will generally not have more than 3-4 choices when conversing with an NPC, this is to simplify the overall complexity of managing conversation trees for both the player (too many choices is a bad thing) and the designers (too many branches makes one NPC conversation very hard to change). Regarding the inventory system, the interface will be designed using scaleform (flash), Actionsript 2 and a relay to a .uc file about inventory management, similar to that of the NPC dialog. The inventory system will initially have a limit on the number of tablets you can hold at once, but the number of tablets you can equip is going to be fixed around 1-3 for the player. The purpose of limiting the capacity of tablets that a player can hold is to add a level of depth to the item management system/skill system, if a player can hold everything at all times, they make no sacrifices. This prevents the game from becoming too easy with no consequences for gameplay choices. The tablets that an enemy drops will be controlled in the .uc file of the specific type of enemy, based on some type of RNG, and a specific percent chance for each type of tablet. This

will give the game an element of replayability and add an optional amount of grinding for specific types of tablets, prolonging the time spent in game, and providing the player with another motive to kill enemies.

## **5.2 Concrete**

Gameplay logic will be handled through Kismet (10.9) as well as a custom gamemode built with UnrealScript.

# **6. Core Mechanics**

## **6.1 Abstract**

The main character in the game has many different attributes: Health, Attack Damage, Attack Speed, and Defense, to name a few. Each of these attributes can be changed with our inventory system by adding more tablets to the tablet slots. The player can also attach different weapons to the weapon slot and armour to the armour slots. Experience can be earned but is attributed to tablets as opposed to the player, which means that some tablets are able to level up while the player cannot. Certain tablets can also be obtained which allow the player to use specific abilities, these tablets are actives and differ from the passive tablets. Items which can be equipped as weapons increase in strength as the game progresses, which means as the player defeats more powerful enemies, the items become stronger.

### **Design:**

The core mechanics of the game are those of a 3rd person top down isometric hack and slash. The user controls the player character using the keyboard for movement (WASD or arrow keys) and using the mouse buttons for attacking. The main attributes of the character are health, attack damage, attack speed, and defense. These attributes can be set in the .uc file of the main character. In addition, enemy units will also have these attributes set in their respective .uc files. The modification of these attributes through picking up tablets will be done by the game modifying the variables pertaining to the attributes using a separate .uc file which inherits from the original attributes file. The levelling up of certain items will be done by using an item class which all individual items inherit from, simply, there will be a variable that denotes what level the current item is at, and a multiplier of that level with certain attributes of the item. These Unreal Script files are all implemented using UDK gametypes.

## **6.2 Concrete**

Core mechanics are defined within UDK gamemodes, which make use of UnrealScript (10.12.c and 10.12.h). The sounds and particle effects that make up the abilities the player has include (10.3.a-c, 10.3.v, 10.4.a-e).

# **7. Characters**

## **7.1 Player**

### **7.1.1 Abstract**

The character is originally designed as a generic male character, the player then has the choice to customize the character given certain presets. The main character does not have a voice and will only produce generic sounds when attacking or performing certain actions (10.3). Since this is an RPG where we allow player customization, the character must be very generic in order to allow as much of the player base as possible to connect with the character.

### **7.1.2 Concrete**

The player will use a generic NPC model (10.7.a) but have a custom texture (10.2.z). The player will be able to further customize this model by using equipment and weapons, which are static meshes (10.6.a-c,e-j) combined with custom textures (10.2.a-t). The player is a human and all humans will make use of their specific animation set (10.5.a-f)

## **7.2 NPCs**

### **7.2.1 Abstract**

We will be implementing 3 types of NPCs: The quest givers, the vendor, and the generic NPCs. The quest givers are any NPCs which affect the main storyline of the game or the side quests which includes the main villain of the game. These characters will be different than generic NPCs to allow the player to more easily recognize these characters. They will also have a voice as they will need to speak with the main character, each of the voices will be different in order to comply with the look of each character. The vendor will have a unique look and will be located in a shop within the village, the player will easily be able to locate him throughout the game. He will also have voice lines which try to hint at the player that the vendor is greedy. The generic NPCs will be wandering around the village as well as the world and will not communicate with the player, if they do it will be a single line with generic purpose. These NPCs serve as filler to populate the world we created, they will also be the victim of the kidnappings throughout the game.



### 7.2.2 Concrete

NPCs will share similar models (10.6.a) but have varying textures (10.2.w-y) and equipment to distinguish them. They are constructed in a fashion similar to the player character but are made to look generic. Each NPC contains its own behaviour through scripts (10.12.d-g) which is also affected by the state of the game (10.12.k), and when players interact with friendly NPCs dialog trees are open based on (10.12.i-j).

#### **Design:**

All characters in game will consist of a skeletal mesh and appropriate models, the skeletal meshes will be used by many characters and the materials will be changed in order to minimize the size of the game file. These characters will fit into one of two sets: humans and monsters. Humans (including the player character) will make use of variations of the same model and textures, while monster characters use models and textures that are easily distinguishable as separate from the humans. A member of any particular group cannot hurt other members of the same group, but are free to damage any members of the opposite group. That is, humans can't hurt each other but are able to hurt monsters, and vice versa. The purpose of human characters are to assist the player (who is also a human) by advancing the plot and augmenting the player's abilities, while the purpose of monster characters are to act as an obstacle to challenge the player as they progress through the game.

## 8. Presentation

### 8.1 Abstract

The player will see the world from a 3rd person perspective, about 3 meters above and behind the main character. When entering buildings the ceilings will disappear in order to allow the player to see what is happening.

When the player is walking around in the open world, the player sees a HUD where the player's health is displayed. The player can also see a minimap in the top right corner showing them their surrounding area. Along the bottom of the screen is the action bar, which displays all of the actions the player is able to use at the current time. At the bottom right of the screen is an inventory menu as well as an option menu which allow the player to access their inventory as well as quit the game. Once the inventory button is selected, the rest of the screen becomes darker and the player's inventory is displayed in the middle of screen with a somewhat low opacity setting. Selection the menu button will bring up an option menu, the rest of the screen will be considerably darker and the menu will not have lowered opacity.

#### **Design:**

The game's default camera angle will be defined by Kismet. It will always be looking at the player model and will be fixed to the player model's relative position, with a small offset to the X and Y coordinates and a large one to the Z coordinate. This will create a top-down camera

that will look down on the player with a pitch angle between 60° and 90°.

Level geometry will be designed to accommodate the camera's position above the player, such that unimportant details do not obstruct the player's camera and important objects are not hidden from view. A scaleform HUD will be applied on top of the screen, and a mouse cursor will be visible to show its position on the screen relative to the player.

## **8.2 Concrete**

The menus will use scripts (10.12.h, 10.12.c) to appropriately extract information and display HUD elements to the user for quick access of character stats, and the HUD display elements (10.2.nn-uu).

# **9. Levels**

## **9.1 Abstract**

Four key areas are currently planned for the first episode. They will consist of a tutorial dungeon, the first plot-centric dungeon, the overworld, and the NPC village. The NPC village will contain most of the NPC characters, while the other areas will have enemies to fight.

The overworld acts as a hub for the player to explore and perform game actions outside of the plot. It also is the medium through which the player must travel to get from one location to another.

The first level is the tutorial dungeon. It is a small area that is based off of the larger dungeons to come. The player's objective is to exit the dungeon. They accomplish this by having the dungeon teach them how to play the game. There will be no distinctive theme for the tutorial level as it is to serve as an introduction to how all future levels will play out. This is one of the very first elements encountered in the game's storyline.

The second level is a much larger version of the tutorial dungeon. It builds off of what the player has learned in the tutorial dungeon, but still follows the first dungeon's theme. Here, the player is expected to enter the dungeon of their own accord and retrieve an item that is required to further the storyline. This dungeon is literally an extension of the tutorial dungeon, as the player is compelled by a quest NPC to return to the tutorial dungeon to retrieve an item that they missed the first time they were there.

## **Design:**

There will be five maps - One for the NPC village, one for the overworld, one for the tutorial dungeon and one for the first dungeon, in addition to a fifth which will save as the interiors of any buildings that the player wishes to enter. The player will start the game in the NPC village and find themselves forced into the tutorial dungeon. From then on, the restrictions on the player are lifted and they are freely able to travel between any of the four maps. Players

will move from area to area by walking through a doorway which triggers a kismet sequence that will load the corresponding map.

## 9.2 Concrete

Areas are built as maps within UDK (10.11). There will be kismet sequences (10.9.b,d,g-j) used for different dungeon events such as opening/closing doors and spawning enemies.

# 10 Concrete Components

## 10.2 Textures

- a Metal Sword
- b Wooden Sword
- c Metal Axe
- d Metal Chestplate
- e Ornate Chestplate
- f Obsidian Chestplate
- g Demonic Chestplate
- h Metal Greaves
- i Ornate Greaves
- j Obsidian Greaves
- k Demonic Greaves
- l Metal Gauntlets
- m Ornate Gauntlets
- n Obsidian Gauntlets
- o Demonic Gauntlets
- p Metal Helmet Covering Face
- q Ornate Helmet Covering Face
- r Obsidian Helmet Covering Face
- s Demonic Helmet Covering Face
- t Metal Skull Cap
- u Skeleton
- v Hulk
- w Human (Dark-Brown skinned)
- x Human (Pale skinned)
- y Human (Tan skinned)
- z Human (Main Character)
- aa Ratmen
- bb Tree

cc Rock  
dd Treasure Chest  
ee Ornate Treasure Chest  
ff Grass  
gg Stone  
hh Cracked-stone  
ii Dirt  
jj Horn  
kk Blood spatter  
ll Artifact  
mm Tablet  
nn Fire Spell Icon  
oo Frost Spell Icon  
pp Shield Spell Icon  
qq Sword Icon  
rr Icon for each armor piece  
ss Health Bar  
tt Energy bar  
uu Icon for each NPC

### **10.3 Sounds**

- a Kindling/Fire Burning
- b Wind
- c Twinkling
- d Monster Groaning
- e Walking on Concrete
- f Running on Concrete
- g Walking on Grass
- h Running on Grass
- i Bash in Air
- j Slice in Air
- k Hitting Metal Plate
- l Grunting Noise
- m Impact with ground
- n Ambient Village Noise
- o Ambient Dungeon Noise
- p Ambient Jungle Noise
- q Village Music
- r Dungeon Music
- s Jungle Music
- t Mice Squeaking
- u Bones Rustling
- v Liquid-in-a-Bottle Sound

### **10.4 Particle Emitters**

- a Fire
- b Smoke
- c Ice/Frost
- d Shield
- e Healing Particles
- f Blood
- g Poison

### **10.5 Animation Sets**

- a Human Walking
- b Human Running
- c Human Swinging Weapon

- d Human Casting Magic
- e Human Dying
- f Human Taking Damage
- g Hulk Walking
- h Hulk Smashing
- i Hulk Dying
- j Hulk Taking Damage
- k Ratmen Running
- l Ratmen Swinging Weapon
- m Ratmen Dying
- n Ratmen Taking Damage

## **10.6 Static Meshes**

- a Sword
- b Axe
- c Artifact
- d Tablet
- e Helmet Covering Face
- f Skull Cap
- g Horn
- h Metal Chestplate
- i Metal Greaves
- j Metal Gauntlets
- k Tree
- l Rock

## **10.7 Skeletal Meshes**

- a Human
- b Skeleton
- c Ratmen
- d Hulk
- e Treasure Chest

## **10.9 Kismet Sequences**

- a Entering House
- b Entering Dungeon

- c Player Death
- d Triggering Trap
- e Triggering Quest Availability
- f Completing Quest
- g Spawning Hulk
- h Spawning Skeletons
- i Spawning Ratmen
- j Completing Dungeon

## **10.10 Matinee Sequences**

- a Intro Story
- b Friend Kidnapped
- c End Story

## **10.11 Map Geometry**

- a Overworld
- b Skybox
- c Tutorial Dungeon
- d Cave
- e Forest
- f Volcano

## **10.12 Scripts**

- a Camera logic
- b Player Movement Control
- c Player Character Attributes/Abilities
- d Skeleton AI
- e Hulk AI
- f Ratmen AI
- g Villager AI
- h Item Manager
- i Dialog for Lore Expansion
- j Dialog for Main Quest
- k Quest Event Manager