

Game Software Design Specifiation Document for
The Scattered Lands

Tinus
©2006 LazyBumWare

August 14, 2006

Contents

I	Overview	4
1	Introduction	5
1.1	Goal	5
1.2	Philosophy	5
1.2.1	Fun	5
1.2.2	Experience	5
1.3	Target	5
1.3.1	Audience	5
1.3.2	Platform & OS	5
2	Feature Set	6
2.1	General Features	6
2.2	Setting	6
2.3	Gameplay	6
II	Setting	7
3	General Setting	8
4	World Layout	9
4.1	Key Locations	9
5	Envirionment	10
5.1	Animals	10
5.1.1	Wolves	10
5.2	Plants & Fungi	10
5.2.1	Oak	10
6	Magic	11
7	History	12
8	Characters	13
8.1	PC	13
8.2	NPC's	13
9	Items	14
9.1	Weapons	14

10 Travel	15
11 Organizations	16
12 Gods	17
13 Weather	18
14 Time	19
14.1 Day and Night	19
 III User Interface	 20
15 GUI	21
16 Rendering	22
17 Sound	23
17.1 Music	23
17.2 Sound Effects	23
 IV Mechanics	 24
18 Fighting System	25
18.1 The Sword-Equiped PC vs. The Vicious Rat	25
19 Ethical & Political System	26
20 Dynamic Terrain	27
21 Saving & Loading System	28
 V Resources	 29
22 C++	30
22.1 Coding Standards	30
23 External Libairies	31
23.1 Ogre 3D	31
23.2 Ode	31
23.3 OpenAL	31
 VI Miscellaneous	 32
24 Design History	33
25 Licence	34

26 About Us	35
26.1 LazyBumWare	35
26.1.1 Damlaj	35
26.1.2 Jeff	35
26.1.3 John Crambs	35
26.1.4 Justin	35
26.1.5 Pabst	35
26.1.6 Scott	35
26.1.7 Tinus	35
27 Thanks To	36

Part I

Overview

Chapter 1

Introduction

1.1 Goal

The goal is to make a realistic (coherent), yet playable fantasy RPG with an interesting and non-linear story.

1.2 Philosophy

1.2.1 Fun

1.2.2 Experience

1.3 Target

1.3.1 Audience

The specific group that TSL might be intended for is a small community of dedicated players.

1.3.2 Platform & OS

TSL will run on both Microsoft Windows and Linux.

We will consider other operating systems, if they are supported by our 3D engine, and if there is someone from the development team who has access to a computer with that OS.

Chapter 2

Feature Set

2.1 General Features

- Single-player game
- 32-bit color
- 3D (tile-less) world
- Skill based (no levels) character advancement
- Simultaneous turn-based combat
- Advanced ethics & politics system
- Dynamic terrain
- Intruiging storyline

2.2 Setting

- Medieval fantasy
- Little magic
- No industrialisation

2.3 Gameplay

The player will controll one medieval character that is sent on a mission through the ennemie's territory. During this mission the character unravels the plot. (See the story) Advancement is made by fighting, thievery and conversations.

Part II

Setting

Chapter 3

General Setting

TLS takes part in a medieval fantasy world, where magic is costly and industrialization is nonexistent.

Chapter 4

World Layout

4.1 Key Locations

Chapter 5

Environment

5.1 Animals

5.1.1 Wolves

5.2 Plants & Fungi

5.2.1 Oak

Chapter 6

Magic

Chapter 7

History

Chapter 8

Characters

8.1 PC

8.2 NPC's

Chapter 9

Items

9.1 Weapons

Chapter 10

Travel

Chapter 11

Organizations

Chapter 12

Gods

Chapter 13

Weather

Different colour fog is used to set the mood for the part of the map that the player is currently in.

Rain that comes down but doesn't, for instance land under a tree. Note: Is this technically possible?

Chapter 14

Time

14.1 Day and Night

Part III

User Interface

Chapter 15

GUI

TLS will have a World Of Warcraft-like camera handling. There won't be a detailed map, as it is unrealistic. (Is that ok for everyone?)

Chapter 16

Rendering

We will use the Ogre 3D engine for the rendering.

Chapter 17

Sound

17.1 Music

17.2 Sound Effects

Part IV

Mechanics

Chapter 18

Fighting System

As said before, the battle is turn-based and concurrent. The system we'll use is quite complicated, so I'll give an example first.

18.1 The Sword-Equiped PC vs. The Vicious Rat

At the moment the PC notices (sees, feels, ...) the rat, the game switches to the turn-based mode.

Round 1: At this point both the rat and the player decide what they will do. As they both decide to attack each other, they both need to come closer first, which they do.

Round 2: They are still too far away from each other, so they both move closer.

Round 3: Now our PC can almost hit the rat with his sword, so he decides to wait for the rat to come closer, which it does.

Round 4: The PC decides to start to swing his sword to hit the rat. The rat needs to come even closer.

Round 5: The PC notices that the rat moved so his sword won't hit it, so he (automaticly) re-targets the swing. The rat jumps to the PC's ankles to bite them.

Round 6: The PC misses the rat because it moved again. The rat bites in our PC's ankles.

Round 7: The rat continues to bite, but the PC will stab the rat now in its back.

Round 8: The rat is deadly hit in the back, and the PC moves back to make sure he doesn't get bitten by the dying rat.

As the rat doesn't impose a threat anymore, the game returns to normal mode.

Note: an exact description will follow later.

Chapter 19

Ethical & Political System

I was thinking about an two axes system:

Libertarian (think 'Chaotic') - Authoritarian (think 'Lawfull')

Left (economical left, think 'Good') - Right (economical right, think 'Evil')

Chapter 20

Dynamic Terrain

It should be possible to:

- Dig holes in the ground.
- Cut down trees.
- ...

Chapter 21

Saving & Loading System

Part V

Resources

Chapter 22

C++

22.1 Coding Standards

Chapter 23

External Libraries

23.1 Ogre 3D

23.2 Ode

23.3 OpenAL

Part VI

Miscellaneous

Chapter 24

Design History

Chapter 25

Licence

Chapter 26

About Us

26.1 LazyBumWare

26.1.1 Damlaj

26.1.2 Jeff

26.1.3 John Crambs

26.1.4 Justin

26.1.5 Pabst

26.1.6 Scott

26.1.7 Tinus

Chapter 27

Thanks To

- SourceForge.net, for their free hosting
- Ctaylor, for his design template