

**1/26/2006**

**Team #7: Pez**

**Project: Empty Clip**

**Members: Alan Witkowski, Steve Huff, Thos Swallow, Travis Cooper**

**Document: Project Plan**

## Project Plan

### 1.0 Introduction

#### 1.1 Project Overview

To create an Action RPG computer game.

#### 1.2 Definitions, Acronyms, and Abbreviations

Action RPG – Role-playing Game with action elements.

A.I. - Artificial Intelligence.

GUI – Graphical User Interface

LAN – Local Area Network

MB – Megabytes

Mbps – Megabits per second

MHZ – Megahertz

RAM – Random access memory

#### 1.3 References

Davis, T., Jackie, N., Shreiner, D., and Woo, M. *OpenGL Programming Guide*, Pearson Education, Boston, MA, 2004

NeHe Productions. "NeHe Productions: OpenGL Lessons", 8 December 2001. Online. Internet [25 January 2006]. Available WWW: <http://nehe.gamedev.net/>

Norvig, Peter and Stuart J. Russell. *Artificial Intelligence*, Pearson Education, Inc., Upper Saddle River, NJ, 2003

#### 1.4 Overview of Document

In section 2, we will describe how the project is organized, schedule, roles, deadlines.

Section 3 describes meeting times, skills, and team name.

Section 4 talks about user requirements and the game synopsis.

## 2.0 Project Organization

### 2.1 Process Model

We will be using the waterfall model. Its structure will allow us to easily go back in steps. We will be testing early and often to ensure each component is working before moving on. We will make sure our structure and design are sound before implementing them, ensuring that later components will integrate easily.

Week 1	Bids, form teams, project web site.
Week 2	Read book.
Week 3	Project plan
Week 4	SRS, High level review, project plan
Week 5	SDS v1, Map system, basic graphics, collision detection
Week 6	SDS v2, Menu system, basic weapons.
Week 7	VVP, Inventory system, monsters
Week 8	Items, File I/O for maps and characters
Week 9	VVR v1, Stage Release 1, Playable character, basic monsters, menu system, inventory system.
Week 10	
Week 11	Monster AI, start networking
Week 12	VVR v2, Stage Release 2, Developing levels, monster AI, start networking.
Week 13	
Week 14	VVR v3, Stage Release 3, Networking, audio
Week 15	Product Release, Polishing, extra details, final graphics.
Week 16	Legacy Turn-In, Final Report, Demo

## 2.2 Organizational Structure and Responsibilities

Alan Witkowski – Team leader, graphics, collision detection  
 Thos Swallow – Networking, audio  
 Steve Huff – A.I., art  
 Travis Cooper – Menu system, GUI

## 2.3 Organizational Boundaries and Interfaces

Tom Henderson – Instructor, University of Utah, Salt Lake City – Utah. Responsible for giving feedback, setting deadlines, and helping with management.  
 Eric Jensen – Teach Assistant, University of Utah, Salt Lake City – Utah. Will meet with the team, give feedback, and help with planning.  
 Amber Swallow – Graphic Artist. Consultant for Graphics. Denver, Colorado.

## 2.4 Reviews, walkthroughs, inspections, and audits

### 2.4.1. Procedures for reviews, walkthroughs, inspections, and audits

Every week we will review the previous week's work according to our schedule and goals. We will conduct a walkthrough before each release to see what needs to be fixed. The team leader will conduct inspections to make sure integration stays

intact between team members.

#### 2.4.2. Schedule of reviews, walkthroughs, inspections, and audits

We will do a weekly review every monday at 1:00 PM. We will conduct the walkthrough the wednesday before each release date. We will perform inspections and audits specifically for each milestone.

### 3.0 Team-specific aspects

#### 3.1 Management Objectives and Priorities

The team leader will make sure all team members are completing their work, integrating correctly, and getting all the necessary information that they need. Individual team members are responsible for prioritizing their work, getting help when needed, updating their logs, and helping other members when needed.

Our goals for the project are to have fun, produce a functional and interesting game. We aspire to add character movement, monsters, levels, weapons, sound, multiplayer, puzzles, and good graphics.

#### 3.2 Team name

Pez. The name PEZ was derived from the German word for peppermint, Pfefferminz.

#### 3.3 Possible Meeting Times

Wednesday, Friday from 1:00 PM.  
Monday – 1:00 PM until 4:00 PM  
Saturday – Plan as needed

#### 3.4 Team's Range of Skills and Experience

1 = very experienced, 3 = some experience, 5 = no experience

<i>Skill</i>	<i>Alan</i>	<i>Travis</i>	<i>Steve</i>	<i>Thos</i>
C++	1	1	2	1
OpenGL	1	5	5	3
A.I.	2	4	2	2
Networking	3	3	5	2
Audio	4	5	5	4
GUI	4	1	4	3
Game Playing	1	1	1	1
Leadership	2	3	1	2
Design	2	3	2	2

### 4.0 Team-specific aspects

#### 4.1 Overview of Functional Requirements

The game will allow you to build up a character, find better weapons, solve puzzles, kill monsters, complete missions(alone or cooperatively), and battle each other over the LAN.

#### **4.2 Overview of Data Requirements**

Input comes from the user manipulating the keyboard and mouse. Output is from the monitor. The product will use data that is stored on the harddrive.

#### **4.3 General Constraints, Assumptions, Dependencies, Guidelines**

Windows XP Operating System  
500 MHZ Pentium or equivalent  
GeForce 2 equivalent video card with 32 MB of memory.  
20 MB Hard Drive space  
128 MB RAM  
10 Mbps Ethernet Card (optional)

The following is a list of components that will interact with each other.

- Game Environment: Playing area for character. Includes monsters, objects, textures, boundaries, etc.
- User interface: Information about the current state of the character and game, including inventory, health, and skills
- Graphics: OpenGL system used to control and maintain what the user sees on the screen.
- Character: An object in the game that is controlled by the user, using keyboard and mouse.
- Monsters: animated objects on the screen that interact with the character.
- Objects: unanimated textures(objects) that the character can interact with.
- Textures: the smallest unit of the game playing area. The playing area is set up in blocks.
- Physics/Collision Detection: The system used to ensure that the character interacts with the environment in a logical manner (e.g. Not walking through walls).
- Menu: The initial setup of the character as specified by the user.
- Map Editor: The tool used to create environments in the game.
- File Structure: The system used to keep track of object, characters, and monsters, as well as the map editor.

#### **4.4 User View of Product Use**

Some specific features that makes our game unique

- Testing the limits of the number of objects on the screen at once.
- Collaboration in solving puzzles.
- Upgradable weapons and characters.
- Drag-and-drop user interface.
- Realistic implementation of guns and recoil.

## **Game Synopsis**

You and your friends find yourselves deserted in a haunted mansion, with only your pistol and sharp wit. As you explore your surroundings, you realize this is no ordinary mansion. You become swarmed by endless amounts of zombies, werewolves, aliens, spiders, skeletons, mercenaries, and demons. Lucky for you, these ghoulish monsters happen to drop ammunition and health packs when killed. To overcome the thousands of threats before you, your team must gain experience and advance in levels to use better equipment. To complete each level, you must solve puzzles and defeat a boss enemy.