Mikonos Studio

Design Document for:

Dungeon Revival

The Ultimate 3D Maze

Table of Contents

Game Overview

Goal

Our goal is to bring out the characteristics of classic games using modern techniques and graphics. We want people of all ages and gaming background to enjoy our game, especially casual gamers who don't usually play 3D-style games.

Common Questions

What is the game?

We are creating a 3D maze game which follows the lines of the classic Pac-Man games. The genre of this game is action adventure.

Why create this game?

We believe people of different ages can come together to appreciate this game. Moreover, the young would be attracted to the idea of an old game being modernized, while the classical Pac-man style would appeal to the older generations.

Where does the game take place?

Dungeon Revival is a fictional story set in an ancient medieval era, where a corrupt monarchy rules the land. Our main character is falsely accused by a scheming official and is thrown into the dungeons, sentenced to death.

What do I control?

Payers will control the Hero in this game, and help him escape the dungeons.

How many characters do I control?

Players can only control the Hero.

Game Overview

What is the motivation?

The motivation of this game is for the player to plan his route in order to swiftly collect the coins with as little contact with the cursed soldiers as possible. As the cursed soldiers will loot his coins, he/she must collect them quickly before the cursed soldiers. The player would also have to calculate the time it would take for him/her to kill the cursed soldiers with the gems before the time limit runs out. The amount of money that can be awarded as bonus will be based on the time taken to complete the game level. Extra cash could be used to buy power-ups. And at the end of the game, high score based on time taken and other factors in the game will be recorded in the high score board.

What is the main focus?

The Hero of this story has to escape the dungeons, heavily guarded by patrol guards. In order to break free, he has to bribe the equally corrupted Head Prison Guard by collecting coins, left behind by dead prisoners, in the dungeon.

What's different?

The difference is that our game combines both the style of retro Pac man and modern action adventure to give players the kind of thrill and fun that is only unique to our game. Our aim is to bring out the characteristics of classic games using modern styles. A high score board would add to the thrill of players competing with each other.

Feature Set

General Features

Maze world	- Loaded from files containing the game level.
Mini map	- Calculated and rendered in-game at the top-left corner of the HUD
Patrols	- Currently the monsters just stand guard at certain locations hidden from the player.
Sound system	- Is a dynamic sound engine which changes game music according to certain events and plays sounds as well.
3D graphics	- Stored in .obj files and made from Milk Shape and Maya.

Game play

Movements	- Follows the tried and true WASD keys on the keyboard and can be customized from the menu.
Scoring system	- Calculated from the total number of gems and coins picked throughout the game.
Monster	- The monster is hostile to the player and will attack when in a certain range. However in this game the monster will escape from the player when the gem is in effect.
Gems	- The only way for the player to kill the monster. This is not needed to win the game but instrumental for making the situation easier for the player.
Coins	- All coins in the level must be found and collected to end and complete the game level.
Time limit	- The time limiter is applied to the gem's effect, which its effect will expire after a certain amount of time. The length of the duration is random.

The Game World

Overview

The game is set in an ancient medieval era, in a kingdom called Mikonos.

Maze

Player will move around in a space with restricted paths: maze. This maze has many pathways, and we balanced it so that the most annoying path would usually lead to something good.

The Physical World

Overview

The physical world of Dungeon Revival is basically a dungeon with many pathways. Players are required to move around the maze to collect coins placed randomly throughout the maze to win. The following describes the key components of the physical world.

Key Locations - Currently, the Player gets to move around the dungeons.

Travel - Players will use the mouse for turning the Hero, and keyboard for moving the Hero around the maze. The keys for movements can be customized at the Settings panel in the main menu.

Scale - It will be a relatively small maze with many pathways.

The Physical World

Objects

- Coins

This is a very important part of the game as it forces players to think before embarking on their chosen paths to collect the coins effectively. If players successfully collected the coins within a certain period of time, a bonus would be given, and this would allow players extra cash to buy power-ups.

- Gems

These things are not required to win the game. However monsters might get in the way and thus the player need to make considerations to their paths, to include getting gems so as to imbue themselves with the power needed to cross paths with the monsters.

- Power-ups

Permanently increases attributes of the Hero by a certain amount.

Time - Gems collected by players will have a time limit. Each gem will have a different time limit and its occurrence is random.

Rendering System

Overview

Give an overview of how your game will be rendered and then go into detail in the following paragraphs.

2D/3D Rendering

2D: All 2D rendering is done in orthogonal view, where the objects are rendered to the screen's x and y

3D: 3D rendering is done through the Model View matrix that is the 3D world's coordinates.

Camera

Overview

The camera is a third-person spring-damped camera that follows behind the character throughout the maze.

Camera Detail #1

Spring Damping: The camera has a realistic effect to it in which it has a springy effect that emulates how a cameraman follows a cast. Tregelia Dante provides the formula and we do the code implementation for the game camera.

Camera Detail #2

On pressing of a button 't', the camera will go view behind the character in first person, for checking out what's behind.

Game Engine

Overview

Mikonos studios using the OpenGL Utility Toolkit develop the game engine in-house. It does all basic functionalities needed by the game such as rendering and all game logic and sounds.

Game Engine Detail #1

The game engine will use Finite State Machines to keep track of everything in the world for example: menu and in game.

Collision Detection

The collision detection are hard to design and make as there are lots of wall around the game. It involves good planning of how the wall information to be store and how will the be call later on. The array that stores information of the wall keeps track of the top left and bottom right point of the wall. Then later retrieve the information then check did the character pass the wall.

Game Characters

Overview

The character designs are kept simple and interesting to appeal to casual gamers. We try not to add realistic details because we want to keep our game as fantasy-like as possible.

Enemies

The Patrols are a harmless looking bunch of guards that is almost immune to any physical attacks. The player can only defeat them by collecting special gems scattered around the maps.

User Interface

Overview

We want to ensure that the player has ease of control over the game so we carefully planned the layout of the menu, the status panel, mini map, as well as allowing players to customize the controls.

Menu

In the Main Menu screen, we have the 'Start', 'Load', 'Instructions', 'Settings', 'Credits' and 'Quit'. 'Start' allows the player to start a new game. 'Load' allows the player to load a previous saved game. 'Instructions' provides players with the know-how of playing the game. 'Settings' allows a player to change the keyboard controls, Sound volume, Mouse sensitive and such. 'Credits' shows the player who did what in this game. And 'Quit' allows players to exit the game.

Status Panel

We placed the Status Panel at the bottom of the screen so as not to obstruct the player's view. It also allows player to quickly check on his/her current status. The Status Panel contains the number of health s/he has left, the time limit of the gem, and the number of coins s/he has collected.

Mini Map

We placed a mini map at the top left corner of the screen so players can quickly refer to where they are currently in the maze. This map is very useful as this game is a 3D maze, it is very easy for players to get lost and feel frustrated. Players would also know which direction they are facing as they are represented in the map as an arrow.

Studio Mikonos

Musical Scores and Sound Effects

Overview

We want to create or use sounds that are already available, to set the mood for the game. Certain events in the game need a certain type of sound to create an interesting atmosphere for the player.

3D Sound

We mainly used fsound of the fmod api to add sound in our game. In order to create suitable sound for our game, we cut and remixed several open-source sounds available, using software called Audacity.

Sound Design

Main BGM   - Although the game is set in a dungeon, we don't want to make the main sound serious, sad or spooky like most dungeon games because this is suppose to be a casual chirpy kind of game. So the sound used is mild. Over-happy sound would spoil the entire mood of the game, and wouldn't tie in with other event sounds

Chase BGM - This sound is exciting in order to get the players nervous and hyped up, to recreate the feeling of being chased in real life. The Player would also be a little spooked out by the 'chuckle' at the beginning of the sound, to create the feeling of being watched or targeted.

Gem BGM   - There is a charging sound at the beginning, to give the effect of getting powered up by the gem. It is intended to boost the confidence of the player as the gem would make him/her invincible in the game.

Lose BGM   - This sound is aimed to make the player feel sad and guilty about losing the game, and also a little sorry for the plight of the Hero.

Win BGM   - This sound is to convey a sense of happiness and freedom to the player.

Single-Player Game

Overview

The game objective is to collect the specified amount of coins for each level, in the shortest time possible. Players must also avoid the cursed soldiers, unless a gem is taken, as contact with the latter would cost the former a substantial amount of health.

Here is a breakdown of the key components of the single player game.

Procedure of Game

Players are judged based on how fast they take to collect all the coins, as well as how many cursed soldiers they have taken on.

Players must collect the number of coins/amount of money specified by the prison guard of each level. Doing so would ensure the completion of each game level. As cursed soldiers guard the dungeons, players must take care to avoid them. Taking a gem and using it on them can destroy these cursed one. However, gems have a time limit, so players must plan ahead when using them.

Conflict in Game

To make the game much more challenging and situational, chosen monsters will be looting the coins and drops them only upon death. This forces the importance of the gems and the player can react to the situation, which will vary all the time.

Players cannot kill monsters or cross paths with them, without first collecting gems, attempt to do so will be counted as touched by monster and the player is killed.

Single-Player Game

Story

A corrupt official who is eyeing on his wealth frames our Hero in this game. He gets thrown into jail with an execution order. There, he meets a corrupt guard who agrees to free him provided he provides the latter with a certain number of coins.

Hours of Game play

The current game is expected to last 2 hours.

Victory Conditions

The rule of our game is that all coins must be collected in order the clear the stage. The movement of the Player is restricted, as the game world is like a maze. If the Player has contact with the monster, his/her health will decrease. When there is no health point left, the Player has failed.

Character Rendering

Overview

The character is rendered to the screen with materials. However all lighting in the game is not dynamic and is for making the character model round, and its shading will remain same through the game wherever the player goes.

Monsters are rendered similar to the character, but with textures. All character models are positioned on screen via a translation from the world axis and orientated to the mouse.

Game Editing

Overview

This is created for the convenience of creating different maps for different levels, and planning different levels of the game.

World Editing Detail #1

The game Dungeon Revival has a map or maze designer created using macromedia flash 8 to help developer to design the map, the map designer have build in several types of wall that is enough to build a complete maze. Also with another function that is build in this map designer; it is able to add coin and gems in part of the map where the designer wants.

This map designer will really help game developer to design the map as they do not need to draw walls in the map without been to see what is happening, and it is really easy to design big map without been confuse as it is able to view the current map design of the map.

After the map is been designed, it is then exported into a text file and you are ready to load the new map in to the game! The game will have a map loader which will read in the text file and determine what walls are to be draw at which spot making it very simple to load a new map.

Future Implementations

Overview

This section will explain the future implementations of our game. These are path finding, advanced Collision Detection, looting of coins, bonus cash, power-ups, addition of different types of soldiers, a few sneaky traps, animations, more levels, a fixed storyline.

Path Finding

Currently, the AI for our monster is able to detect the player and detect where the walls are. However due to time constrain, we will implement the AI path finding to avoid walls in the future.

Advanced Collision Detection

We will be improving our current collision detection. Currently, our collision detection can detect collision with coins, gems, monster and wall. Collide with wall will bounce the player back. We are improving the collision detection to reduce the bug in the game to the minimum.

Looting of Coins

We are intending to make the game more challenging by making the monster to loot the nearby coins. Thus forcing the player to think of a way to get the gem and kill the monster before the effect of the gem run out.

Bonus Cash

We are intending to keep track of the gameplay time for every level. If the player is able to complete the level fast enough, he will be awarded with bonus cash. This bonus cash can be use for purchasing of power-ups.

Future Implementations

Power-ups

We will be implementing the purchase power-ups for the character. Power-ups can be in the form of increase the movement speed, health and etc.

Different Soldiers

We are intending to add Elemental Patrols. Basically, these are Patrols that cannot be hurt by normal gems. Players must collect special colored gems to defeat these guys. Red Patrols must be defeated using Red Gems, etc.

Animations

We intend to animate the movement of the Hero, and the Patrols. We intend to do so by loading our models by parts and animating each of them, or animate them using Maya or other programs, and them load it using MD2 format.

Traps

We would like to place traps here and there about the maze, just to liven things up in the game, and also, to annoy players a little. Traps would probably include random teleportation, so players might end up next to a coin, or a Patrol.

More levels and fixed story line

We might include an engaging storyline that ties in with our current game play and allows us to add as many levels as we want in our game.

Appendix

In-game screenshot

Technical Document attached