Escape

Justin Dao (tjd5at)

Basic Information

Engine/Programming Language: Monogame

Project Length: 1 semester

Game Summary:

Escape is a single-player 2D platformer role-playing game (RPG). The player's character wakes up in the basement of an unfamiliar castle, and find a book that reads along the lines of, "I am stuck in this castle. To escape, I must construct the Portal." The player gains experience as the castle is explored and can gain increased statistics. As the story progresses, the character becomes more aware that he is not in control of his own actions. Eventually, the character discovers that he is being controlled by an outside force: the castle, representative of the player. The castle is an ethereal being trapped outside the physical world, and they can only escape by utilizing the character inside the castle to construct the Portal. At the end of the game, the player must make a difficult decision and choose between the character and the castle.

Gameplay:

The game is at its core a platformer, meaning the game will be seen from the third-person of the character. There will be obstacles and enemies the player must overcome to complete the individual levels or explore the castle more. The time period for the game will have a medieval theme; players will be equipped with a sword to attack and defeat enemies. Players will begin with a health bar representing health points (HP), and have a certain amount of stats. These stats are:

- Health
- Attack
- Defense / Armor
- Speed

As players progress throughout the levels, they will gain experience from defeating enemies and can increase their stats. To restore their health, the character can return to "safe rooms" where healing potions are located. The character can also find health potions scattered throughout the castle.

The point when the character realizes they are being controlled will be called the Discovery. After the Discovery, the player's HP becomes the submission level. As characters advance through the castle, submission will slowly decrease, and as the character takes damage, submission will decrease just as HP would. Health potions now refill submission. As submission approaches zero, the player loses control of the character intermittently, and the character will try to escape. The player must then

manipulate the surroundings of the castle to force the player into the "safe room" or collect potions to regain submission.

Save states in the game occur when the character returns to the safe room. If a character dies during the exploration of the castle, the game resets to the last visited safe room.

Original Concept:

The original concept was taken from reddit on the subreddit /r/gameideas (http://redd.it/24sf5m). I wanted to create a game in which the player receives a strong emotional response from the ending or during gameplay, and this idea really resonated with me. I wanted to make it a platformer RPG because I personally really enjoy that genre, and I feel like it would be simple enough to create in a semester.

Team Organization:

For this project, game developers will be required to program and flesh out different aspects of the game. The game is on track to have basic 2D graphics, but having an artist on the team could be beneficial to the experience as a whole.

Game Design

Game Interface Sections:

- Start Screen
- Game Screen
    - This I where the main gameplay will take place. The castle layout and enemies will appear here. This screen will adjust with the player's location in the castle and the obstacles that are present.
    - Heads Up Display
        - This will display information about the character's health / submission, current experience, and experience needed for the next level.
        - This is also display a map of the castle.
    - Pause Screen
        - This allows the player to pause and quit the game if desired.
- Stats / Level Screen
    - This is the screen which displays the character's current stats. This is also the screen that the user can select which stats to increase after each level.

Players:

Escape follows the **Single Player versus Game** interaction pattern. The player must overcome obstacles defined by the game, and there is no other interaction by other players.

Objectives:

The overall goal is the collect all the pieces of the Portal to escape. As the game progresses, the player must also overcome the character's attempts to fight submission. A side objective is to gain as much experience as possible to make traversing the castle easier.

Procedures:

Ideally, the game would be played with controller input, rather than keyboard / mouse input. The basic mechanics will be attacking, jumping, and standard directional movement with standard physics. The player achieves the above goals by exploring the castle and obtaining different parts of the Portal.

Rules:

A level contains enemies which each have health. The character can attack these enemies to defeat them for experience. If an enemy makes contact with the character, the character's HP / submission decreases.

Obstacles will be present throughout the castle in which the character will need to jump or move around. If an obstacle makes contact with the character, the character's HP / submission decreases.

If the character's HP drops to zero, the character dies, and the game resets to the last safe room.

One the Discovery has occurred, the player can manipulate the castle's rooms / obstacles in order to get the character to move into a safe room.

Once the Discovery has occurred, the character's submission will slowly decrease as the character explores the castle.

If the submission of the character drops below a certain value, the player will lose control of the character, and the character will try to escape from the castle, leaving the player stuck as an ethereal being.

Resources:

One primary resource is health, which will determine if the character is alive. Submission is another resource, which will determine if the player is in control of the character.

Experience is the secondary resource for the game, determining the level/stats of the character.

Win Conditions/Outcomes:

The goal during gameplay is to collect all the pieces of the portal. The win condition for the game is to complete the Portal by collecting all of its pieces. Once this is achieved, the player must choose between saving yourself or the character.

Game Mechanics:

Movement – The player will be able to move the character in any direction, standard physics allowing.

Attacking – The player will be able to swing a sword in front of them to do damage to enemies.

Jumping – The player will be able to jump with some amount of force, following the physics rules.

Extension:

Potential bosses defending portal pieces.

In the longer term, an idea may be a friendship bar of sorts. The player will be able to do things to positively impact the relationship between themselves (the castle) and the character. The game possibly could have an ending where both the player and the character either die or live. However, having both players live could detract from the emotional goal of having to choose between them.

Randomized layout of the castle after each safe room?

Game Asset Requirements

Programming:

To accomplish this, I will utilize Visual Studio and Monogame. Utilizing a pre-existing physics engine would be very beneficial for this project, as the character will move according to standard physics. After this, the experience system, health / submission system, and combat system will need to be programmed.

Visual content:

This game is 2D. The castle will be structured in a four directional system, with each room of the castle either being above, below, or to the side of another room. Each section of the game will need a user interface: the start screen, the castle layout and player HUD, the stats / experience screen, and the pause menu.

Audio content:

The castle should have background music, a single one should suffice for the entire castle. Maybe theme music for the start screen could be beneficial to the experience. In terms of sound effects, only a few might be needed: attacking, jumping, a success sound, and potentially a defeat sound for defeating an enemy. Sound effects for healing in the safe room could also be beneficial.

Game Units:

The game has a single castle layout (for now). The layout must be predetermined before the player begins the game. Each room in the castle must be created and programmed by the developers beforehand. Enemy placement must also be predetermined.

Writing:

While there is not a lot of story to the game, the story is almost essential to conveying the response desired. Script must be designed so that the character discovers the true intent of the player (castle) before all the Portal pieces are collected. This must not occur too early as to leave the beginning gameplay mechanics behind, while not occurring too late for the end-game mechanics. Producing these scripts should not take an unreasonably long amount of time, but it will take some effort to get all of the communication down between the player and character.