CIS267 – Beginning Game Programming

Final Project Rubric

**Game Fundamental Requirements:** The game must be an rpg. The player needs to be able to affect their own health (enemies, eating food, etc.). The game must use a joystick controller at a minimum. There needs to be a minimum of 3 levels with clear objectives to get to the next level.

**Objective:** Get creative. Tell a story. In your paper you must give some kind of background information about your world (Where are we? When is this taking place?), player (Who are we? Why are we here?), a goal (What are we try to accomplish? Why are we doing what we are doing?), and enemies (Who are they? Why are they after us?). I want the game to have an overall theme but I do not want you to only focus on the theme. The major part of this project is the feel of the game and not the story because of your time constraints. I need you to describe to me how the levels will work in your game, how your character will interact with the world, how the character will move in this world. The game must be winnable. If this means that you have cheat codes to make this possible that is fine, but if I cannot get through your levels I cannot successfully grade your team.

**Movement:** How are you planning to have the player move besides using a joystick controller? (Mouse and keyboard, controller, point and click, etc.)

**Camera:** Top down view, platformer, etc.

**Enemies**: What different AI functionality will you have? How many different AI systems will you have?

**Combat**: How will the player interact with enemies? What weapons can the player use?

**Environment**: Is the environment editable by the player?

**Assets**: What kind of art style are you going to use? What kind of sounds will you have? What assets are your creating yourself and what assets will you be using from other resources?

**Timeline:** How do you justify being able to complete this must work in 4 weeks? Make a timeline of how you see the work being completed.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Topic | Topic not present or greatly below minimum requirements– 0 Points | Slightly below topic minimum requirements – 2.5 Points | Met topic minimum requirements – 5 Points | Slightly above topic minimum requirements – 7.5 Points | Above and beyond topic minimum requirements – 10 Points |
| Character |  |  |  |  |  |
| Storyline |  |  |  |  |  |
| Plot |  |  |  |  |  |
| Character Progression |  |  |  |  |  |
| Flow of Progression |  |  |  |  |  |
| Goal |  |  |  |  |  |
| User skills to complete game |  |  |  |  |  |
| Game mechanics |  |  |  |  |  |
| Interacting with Environment |  |  |  |  |  |
| Winning or Losing |  |  |  |  |  |
| User Interface Design |  |  |  |  |  |
| Opponent AI |  |  |  |  |  |
| Code Organization |  |  |  |  |  |
| Knowledge of Unity |  |  |  |  |  |
| Art Style |  |  |  |  |  |
| Music and Sounds |  |  |  |  |  |
| Group Reviews |  |  |  |  |  |

**Minimum Requirements:**

Character:

* Character can move throughout the world.
* Camera shows character when it needs to.
* Character design is thought out and fits with the overall theme of the game.

Storyline:

* There is an overall theme to the game.
* There is a story that the main character follows.
* There is a plot to the story.
* The storyline is presented to the player.
* The storyline is linear and progresses as the player progresses.

Plot:

* The character has a goal.
* The character has a way to win or lose the game.
* The plot is consistent throughout the game.
* The plot is presented to the player.

Progression:

* The player can progress through the game.
* The group determines progression. Gaining different abilities, gaining different gun, increasing score, different levels.
* Progression should come with both positive and negative effects.

Flow of Progression:

* Progression links with both the storyline and the plot of the game.
* Progression should result in gameplay increasing in difficulty.

Goal:

* Storyline and plot should have an end goal. Why is the player doing what he/she is doing?
* Goal must be accomplishable with sub goals throughout the game (progression).

Player Skill to Complete Game:

* Does your game have playability?
* Would someone play your game more than once?
* You game needs to have parts that require the player to have skill to accomplish the games goal.

Game Mechanics:

* Objects move as desired.
* Collisions happen as desired.
* Player cannot get “stuck” at different spots in the game.
* Movement feels consistent throughout the game across all objects.
* Physics are consistent throughout the game across all objects. Unless specified by the team for a desired game/object feel.

Interacting with Environment:

* Player can interact with different objects in the game.
* The player can interact with any object that requires physics.

Winning or Losing:

* The player has a way to win or lose the game.
* There is a winning or losing screen in your game.
* There is some type of “scoring” system to determine if a player has won or lost the game. The team determines the “scoring” system.

User Interface Design:

* GUI is well thought out and flows with the game.
* GUI matches the games art style.
* Basic fonts are not used to display GUI.
* Player has ability to play game via a GUI start window.
* Player has ability to edit game via a GUI settings window.
* Player has ability to replay game via a GUI end game window.
* Player has ability to exit game via a GUI settings window or end game window.

Opponent AI:

* The player must have enemies they must defeat to progress in the game.
* There must be three different types of AI opponents.
* Different AI opponents will have different types of movements and different types of attacks.

Code Organization:

* Your code must be well organized
* If I were to join your team, would I be able to follow what your code is doing?
* Scripts should solve one specific problem. A script should not solve multiple problems.
* You need to document your code.
* You need to explore different code functionality that we have not went over in class.

Knowledge of Unity Engine:

* You need to show an understanding of basic functionality inside of Unity.
* Show the ability to parent objects correctly.
* Show the ability to create and use prefabs correctly.
* Show the ability to use Unity specific libraries and functions.
* Show basic understanding of how the Unity editor works.

Art Style:

* Your art needs to follow a central theme.
* All of your art needs to feel like it belongs in the “World” that you have created.
* Your team must create three pieces of art.

Music and Sounds:

* You must incorporate one piece of music in your game.
* You must create three sound effects in your game.
* Demonstrate the different ways that we can trigger sounds in Unity.