Abyss Unreal Game Engineer Test

Platformer

Create a PC Platformer game project using Unreal and Blueprints/C++.  Specifications are included below as a guide, but feel free to embellish or expand upon certain aspects of your game to best convey your strengths as an Unreal engineer.

**Game Specifications**

All these conditions should be met before submitting your test:

* This is a 3D game sample.
* UI should be minimal.
* You should not include any additional plugins/assets from the Marketplace, use Unreal default assets. You can however start from a predefined unreal game template.
* Game code can be C++ or Blueprints or both, in any combination.
* You must implement a game over screens and the player should be able to restart the game.
* The game should define at least the following aspects

*Content*: Basic Room (platformer style) where we can do the basics of activities (walk, jump, fall and rewind)

*Content:* Walls will not allow the player to go past them

*Content:* Platforms will allow the player/enemies to walk on them

*Time Mechanics*: Rewind

*Mechanics details*

- Rewinding reverts the time and state of the game

- Player can rewind at the push of a key (rewinding starts at key press and ends at ley release)

- Rewinds influences main player state (position/orientation/animation state)

- The world rewinds as well (The state you get when the rewind stops is a valid state from which you can continue to play the game). Remaining recorded time (after a rewind) will remain recorded and used if the player decides to rewind more

- You can rewind as much as you want (to the start of the game) in one go or you can rewind multiple times (one after the other)

Rewind mechanic reference: https://youtu.be/iBjBJRjRBn4?t=172

Player: Character is controlled by player similar to what we see in game example (we use keyboard for movement/interactions and character must animate based on movement state it’s in). The player can never die (once an enemy touches the player (on any other side but the bottom), the game pauses and you can push the rewind button to come back to a state when you weren’t dead)

*Enemy*

Movement: Patrol on the platform it is spawned. If it reaches end of platform or a wall it will turn and walk back.

Spawn: There are enemies spawn points placed in the world where enemies spawn when the game starts. We should be allowed to define as many spawn points as we like.

Death: Enemies die if main character falls on them

Rewind: Enemies are influenced by the rewind of the world. Rewinds influences enemy state (position/orientation/animation state)

The game ends when you have killed all enemies.

**Evaluation**

We will be evaluating your test based upon the following merits:

* Are all specifications met?
* Is the project well organized?
* Is the game playable?
* Is the code/blueprints readable and easily maintainable?
* Is the game optimized and stable?

**Deliverables**

In a zipped archive, provide the following:

* Executable
* Unreal Project with source code and assets
* Description of implementations and assumptions