2020

TECHNICAL DESIGNER TEST

TECHNICAL PART DOCUMENTATION

IGNACIO LÓPEZ RIVAS

Technical Design Test

Regarding this part of the test I took the liberty of thinking of it as a game instead of an exercise. With that in mind I designed a little game in which the player controls a character behind the defense line and its duty is to replenish the resource needs of the defense line which is holding the position against the enemy. There are 4 allied camps and they will ask for random resources, the player has to generate them and use the character to carry them to fulfill the camp needs before the timer goes off or the line will not hold against the enemy. Each camp has two lives, represented by two characters, any time a requirement is not fulfilled, one of them dies. If both die, game is over. The goal of the game is to stay alive as long as possible.

For the purpose of the test I am not going to overdesign or go deeper in ideas to make the game better. This design includes a little level, a movable character and the ability to carry, manage and drop items that are generated by the player.

Requirements and notes on them:

- **Implement a player character controller and simple environment to navigate (the complexity of the player character and environment are not important for this task)**.
    - Character Controller: The only issue I had was that I had to deactivate the Y axis mouse to yaw the character because it was very difficult to handle.
    - Environment: I set up four blocking walls to limit the walkable area. I decorated with some blocks simulating rocks. Made new actors for the defense camps and the supply camp as they were going to need some logic. Made a top down fix camera that felt good to play.
- **Give the ability for the user to spawn a few different items of different types that the player character can then collect**.
    - I chose to spice up a little the game by forcing the player to lift his hand either from the mouse or the left side of the keyboard by placing the item generation keys at the middle of the keyboard. I chose UIOP as the four keys to generate four types of elements. I twisted it a little more just for fun, each time you generate an item, the generators will randomly replace themselves so that the player have to think every time he needs something which key has to press and therefore can not automatize the hand movement.

- **Provide a system that allows the player to view and manage the items they have collected.**
    - This request seemed to me too broad, I simplified it, hopefully not too much. I chose to show the items collected through an UI and give the player the ability to swap the elements from one slot to another. To achieve this and not over-communicate classes I chose just to display the colors on the UI instead of really holding references to the character content. So, when you display the UI you are seeing just the info that describes the object. When you interact with the elements the UI ask the character to do the swapping. Every time there is a change in the character cubes a notification event is thrown and the colors changed. I did make use of network functionality as a shortcut but only send messages to the owner.
- **Design and implement some form of restriction that prevents the player from collecting too many items, or provides consequences if they do so**.
    - By design it is only possible for the character to hold three items. It is controlled by checking right after the input if the character is holding something in that slot, if it is, he will drop it before allowing to collect anything else.
- **Allow the player to drop their collected items**.
    - The input keys perform the collect and drop action. At first, I tried to attach the actor to the character, however after trying several ways I came to conclusion that a Skeletal mesh is needed as I couldn't find a way to do it. There might be some junk files from trying. I did a workaround by destroying the element when you grab it, showing an ever-existing component in the character simulating the same object and hiding it and spawning a similar object when dropping. It may not be the most efficient way, but it is very reliable.

Controls

Character movement

- W, S – Forward, backward
- A, D – Strafe left and right
- X mouse axis – Rotate character around Z axis (yaw)

Character actions

- Left mouse button – left hand pick up / drop action
- Middle mouse button – back pick up / drop action
- Right mouse button – right hand pick up / drop action

Player actions

- U, I, O, P – Generate supply boxes
- Tab – Show score/inventory