

9,4

Videojuego – Programación I

Descripción:

- 1 – Movimiento del jugador
- 2 – Interacción con el entorno

Valoración negativa

Movimiento del jugador

- Se crea una condición para cuando el input sea 0, basta con multiplicarlo por la dirección
- MoveDirection no se normaliza ni se elimina su componente Y
- Se crean más funciones de las necesarias para los input, “grab” y “croach” son “ActionInputs”

Interacción con el entorno

- El mov. del objeto se ejecuta en 3 lugares, dividida en el player y otra en el objeto (no interviene)
- ¿Se debe poder agarrar a los objetos cuando salta?

Errores generales

- No se respeta la CamelCase. Se utiliza un “_” en algunos scripts
- Existe alguna variable que sobra, se debe replantear la lógica para simplificar
- Se incluyen librerías que no deberían incluirse.
- Algunos nombres son poco descriptivos o a veces se usa un idioma, a veces otro.

Valoración positiva

- Funciona correctamente y la carga de trabajo es buena

Mejoras / Soluciones

- Separar más el funcionamiento. El controller se debe centrar en hacer funcionar al player, el action o interaction de interactuar con otros objetos y pasar información o activar su funcionamiento. En el caso de la escalera, afecta al movimiento del player, en el caso del objeto afecta al objeto, por tanto este debe moverse. El player, aunque en la realidad sea lógico que mueva al objeto, no debe encargarse de mover a nadie más que a sí mismo. De esta manera delegamos procesos a otras clases y el player no ha de calcular dos movimientos, el suyo y el de la caja.

9

Videojuego - Programación I

Descripción:

1 - Inteligencia artificial del enemigo

Valoración negativa

Enemigo

Se ha hecho en clase y con ayuda

Falta completar el attack

General

Se han creado librerías innecesarias en los scripts

Valoración positiva

Funciona correctamente

Ha invertido tiempo en avanzar temario

Mejoras / Soluciones

- Se debe añadir el ataque al estar cerca del player
- Se debe terminar de configurar para que todo funcione correctamente

7,6

Videojuego - Programación I

Descripción:

1 - Mecánica de esconderse de los enemigos

Valoración negativa

Escondite

	Mecánica sencilla
	Es posible simplificar y optimizar. Ahora mismo, por cada escondite habrá un Update ejecutándose
	Todos los enemigos van a comprobar si hay "hide zone" y no pueden detectarla independientemente
	El enemigo no es capaz de detectar al player
	Uso del "OnStay" cuando puede ser un "OnEnter" (supone un mayor rendimiento)

General

Hay librerías que no deberían estar en el código.

Valoración positiva

Mecánica desarrollada independientemente

Mejoras / Soluciones

- ¿De qué depende que el player se esconda? De que entre en la zona
- ¿Cuándo falla el escondite? Cuando el enemigo ha visto al player escondiéndose
- Solución: "OnEnter" del player, HideZone comprueba si hay un enemigo. Si no lo hay, el player tendrá una bool "isHided" para que el enemigo pueda pasar a un estado de perseguir si el player que detecta tiene o no la bool "isHided" en true. Si hay enemigo, no puede ponerse a true y el enemigo puede seguir persiguiendo,

7,1

Videojuego - Programación I

Descripción:

1 - Objeto que alerta al enemigo

Valoración negativa

AlertSystem

Mecánica muy sencilla

Hubiera estado bien completar el enemigo con el debug de la posición del ruido "noisePosition". De esta manera se puede ver que llegas a ese punto (no lo he tenido muy en cuenta)

General

EnemyAlert no hace nada, de hecho, el tag "objeto" no existe

Valoración positiva

Funciona

Mejoras / Soluciones

– Se debe conectar con el enemigo pasándo ese "noisePosition" que se ha mencionado

9,5

Videojuego - Programación I

Descripción:

- 1 - Abrir puerta mediante llaves
- 2 - Mecánica de atraer objetos

Valoración negativa

Puerta

Optimización: "OpenBars" repite procesos y pueden simplificarse

Cámara

—

General

Mecánicas sencillas

Los scripts contienen librerías innecesarias que podrían llegar a crear algún conflicto

Valoración positiva

Funciona correctamente

Se han programado ciertas animaciones

Creación de inventario para el player

Mejoras / Soluciones

- "OpenBars" podría optimizarse e incluso que sea la propia puerta que contenga el mecanismo y las barras
- Si añadimos más cámaras, se deberá pensar en cómo organizar la prioridad. Recomendando que cada zona o "ChangeCamara" tenga públicas variables para jugar con la prioridad.
- El cambio de color, el cambio de target... Se comprueban constantemente en el Update. Lograr evitar eso y que se ejecuten en el momento más oportuno no solo optimizaría el código, también lo simplificaría.

9

Videojuego – Programación I

Descripción:

- 1 – Main Menu Interfaz
- 2 –Gameplay Interfaz

Valoración negativa

Interfaz

No hay reescalado (solo funciona en 1920x1080)

General

Variables innecesarias com “keyCounter”, mezcla de idiomas como “activationNiebla” y sin definir como “pressE”

Creación de librerías sin uso

Valoración positiva

Funciona correctamente (en 1920x1080)

Se ha desarrollado de forma independiente

Mejoras / Soluciones

- El “EnemyController” no debería tener esa función, de ser así todos los enemigos estarían comprobando si el player ha de “encender” ese efecto o no. Debería ser el player o, mejor, un script que se encargue de configurar la interfaz del player, que busque si hay algún enemigo para aplicar los efectos
- Me parece más apropiado que el “Press E” se situara sobre el objeto en vez de en el canvas principal.