CODI I DISSENY

# 1. JUGADOR

Texto

El contenido generado por IA puede ser incorrecto.

Aquesta corutina és la que el *Player* fa servir per a poder interactuar amb el món. Si té un objecte interactuable al davant ens guardem la referència a aquell objecte i si és *remarkable* modifiquem els seus materials per tal d’afegir i treure l’*outliner*.

Texto

El contenido generado por IA puede ser incorrecto.

Si no trobem res interactuable al davant i l’últim objecte amb el qual hem interactuat és *remarkable* li traiem el material i posem l’objecte interactuable a null.

Texto

El contenido generado por IA puede ser incorrecto.

Quan el jugador està interactuant amb alguna cosa i pressiona la E, aquesta funció s’executa. Bàsicament el que es fa es cridar a la funció Interact de l’objecte i ell ja s’encarrega de la lògica.

# 2. PUZLES

## 2.1. PUZLE MORSE

Texto

El contenido generado por IA puede ser incorrecto.

Aquest és el diccionari que la llum fa servir per a fer la seva funció. Només vam posar en aquest cas les paraules que ens interessen, tot i que es podrien posar totes i no canviaria res.

Texto

El contenido generado por IA puede ser incorrecto.

Aquesta és la funció que fa que la llum parpellegi en Morse segons la paraula donada. Com es pot veure és una corutina i va recorrent caràcter a caràcter de la paraula, agafa el caràcter traduït a Morse gràcies al diccionari i recorre cada símbol per fer encendre’s i apagar-se segons les pauses pertinents.

Imagen de la pantalla de un computador

El contenido generado por IA puede ser incorrecto.

Per al panell on s’ha d’introduir la combinació també tenim un diccionari de Morse, però al revés i amb totes les lletres.

Texto

El contenido generado por IA puede ser incorrecto.

Aquesta funció s’executa quan el jugador pressiona un dels botons del panell. Fa una cosa diferent dependent del *input* que li arriba sempre i quan la frase actual sigui més curta que 14 caràcters.

## 2.2. PUZLE AMB IA

Texto

El contenido generado por IA puede ser incorrecto.

Aquesta funció agafa la *RenderTexture*, la qual captura el dibuix que el jugador ha fet amb la pissarra, i creem una textura 2D a partir d’aquesta. Vam tenir un problema amb la IA, ja que a la RenderTexture li afectava la llum i la manca d’aquesta. Per aquest motiu hem de recórrer la textura i els píxels que no siguin negres els pintem com a blanc. Per fer això vam haver de fer un *threshold*, ja que si no detectava píxels que no volíem com a negre. D’aquesta manera només és blanc el fons i el dibuix és veu perfectament.

Texto

El contenido generado por IA puede ser incorrecto.

Aquesta funció és la que fa que la nostra IA s’executi i reconegui la imatge passada per paràmetre.

# 3. WAVE EFFECT

Texto

El contenido generado por IA puede ser incorrecto.

Aquesta funció és la que crea l’ona de soroll gràcies a un so de *Perlin.*

Texto

El contenido generado por IA puede ser incorrecto.

A més del *Perlin* afegim octaves per tal que l’efecte de l’onada es noti més quan hi ha més so.

Texto

El contenido generado por IA puede ser incorrecto.

Aquesta funció es crida quan es fa un cop de so (fer un tret, llençar un objecte...). El que fa és augmentar els valors de l’ona durant 0.5 segons per a, seguidament, tornar a l’estat anterior.

# 4. Enemics

## 4.1. Enemic cec

La funció de patrulla de l’enemic cec s’utilitza tant per patrullar pels *waypoints* especificats en el mapa com per voltar al voltant del punt de so si el monstre ha escoltat un so. Els *waypoints* estan dividits en dues parts, la primera part és la part del mapa on apareix i on està localitzat el puzle dels jeroglífics. Quan el jugador resol aquest puzle la porta s’obre i permet al monstre anar a la nova zona amb els nous *waypoints*.



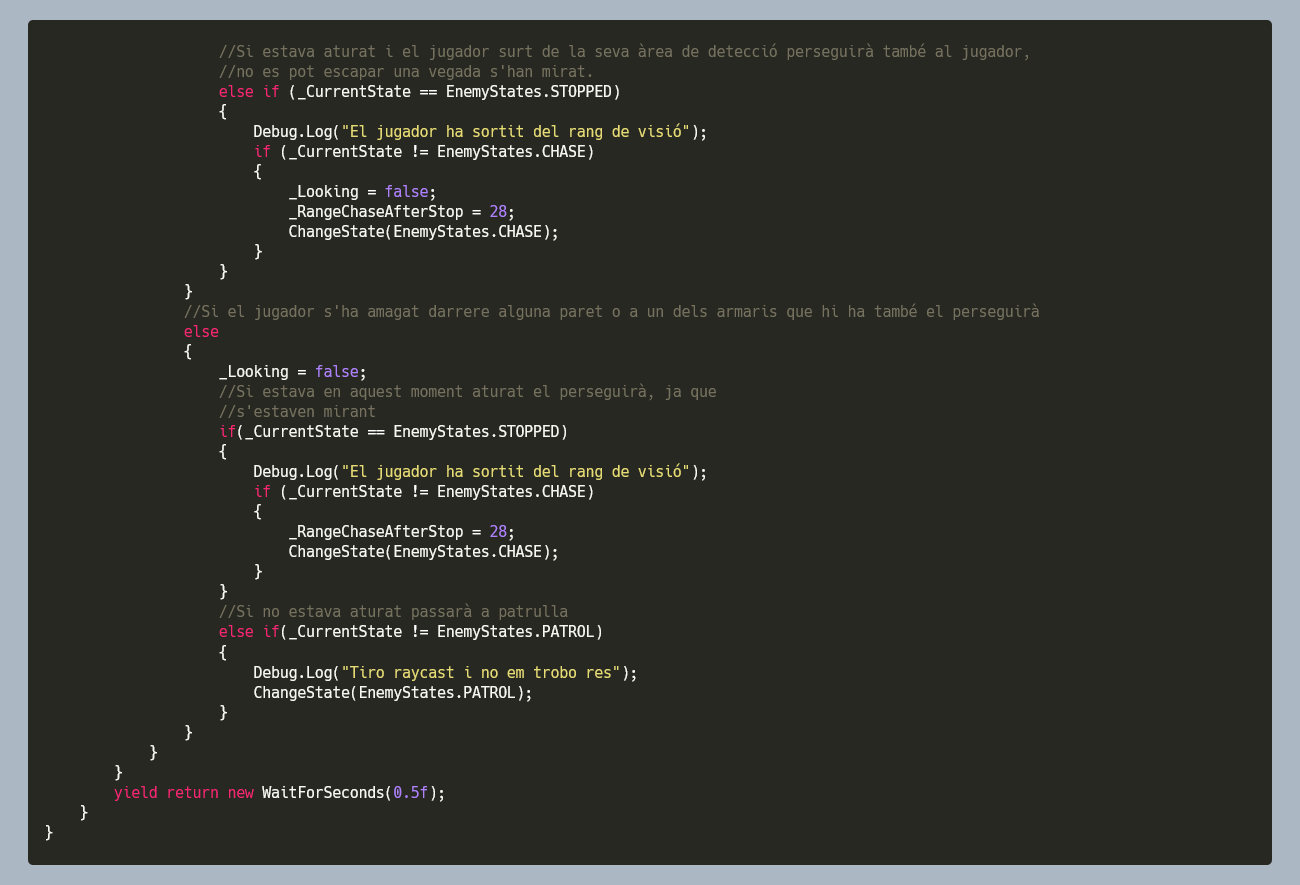
Aquesta funció és la que criden tant el jugador com els objectes llençables quan realitzen un so. L’enemic crea un *raycastall* des de la seva posició fins a la d’origen del so, agafant tots els objectes. Després els parseja mirant si cap d’aquests atenua el so, i si ho fan es redueix el so que es passa per paràmetre. Una vegada reduït el so calcula la distància entre ell i el punt de so i si aquest és menor a 10 i no té cap paret davant canvia el so calculat anteriorment per un valor que supera el llindar màxim del monstre, sinó posa un valor entremig.

Aquest valor després el comprova amb uns paràmetres per reaccionar d’acord amb aquests. Com és fort és més precís serà el monstre a buscar al que ha provocat aquest so. Si el volum és extremadament fort i la distància és idònia salta cap a la font de so, atacant el punt, tant si no hi ha ningú com si hi està el jugador.

## 4.2. Enemic gras

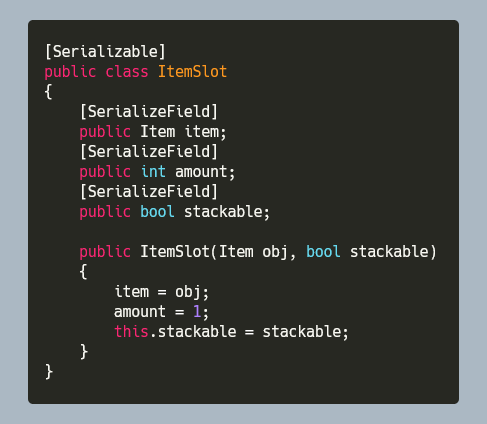
Quan el jugador entra dins de l’àrea de persecució del monstre gras aquest començarà a perseguir al jugador, i si tot seguit entra a l’àrea d’atac canviarà l’estat a atacar. Si el jugador en cap moment surt d’aquestes àrees canvia els estats menys quan està atacant, que s’espera a que acabi l’animació per canviar d’estat, comprovant on està el jugador per canviar a l’estat adient (si està molt a prop a *CHASE* si no a *PATROL*).

## 4.3. Enemic ràpid

Aquesta funció és la més important de l’enemic ràpid, on està tot el processament de les mirades entre l’enemic i el jugador. Primer mirem si el jugador està dins de l’àrea de detecció. Si aquest es troba dins calculem amb un Vector3.Dot si els *forwards* de les mirades estan l’una davant de l’altre per utilitzar-lo més endavant. Fem un *raycast* des de la posició de l’enemic fins a la del jugador i si l’objecte que troba és el jugador i utilitzant el .Dot anterior canviarà l’estat a aturat. Quan aquest .Dot canviï de valor canviarà o a *chase* o a *attack* depenent de la distància a al que estigui el jugador de l’enemic.

# 5. Guardat

Per a que el guardat no ens guardés posicions de memòria dels objectes del joc vam crear unes classes amb els mateixos valors que aquestes però amb variables bàsiques.

Aquestes llistes son les conversions de les classes amb variables complexes a unes que sí ho son.



Aquests son uns exemples dels canvis de classe, d’una que té una variable Item a una que només guarda la id d’aquesta.