FOV:

Texto

Descripción generada automáticamente con confianza media

Declaración dentro de un IF, típico para unreal

Pitch, yaw, roll ángulos tipo eulerangles, en angular.hpp para que no haya que meter código

Texto

Descripción generada automáticamente

Así me dan ya los ángulos

Print

Texto

Descripción generada automáticamente

Git Branch

Git checkout ja/day00\_entity\_basic

Git pull

Git fetch: me dice las ramas nuevas que hay para saber que están. (noa ctualiza)

Aimed

Reache

Texto

Descripción generada automáticamented

## IMPORTANTE: SABER SI ESTOY DELANTE O DETRAS

Imagen que contiene texto, persona, pizarrón, interior

Descripción generada automáticamente

Si el dot producto forward y delta es > 0 lo tengo delante, si es negativo lo tengo detrás (SIN NOTMALIZAR ni nada)

Y si lo hago con left, dot(left\*delta) > 0, entonces lo tengo a la izq.

En el transform esta el test de distancia y de angulo

Texto

Descripción generada automáticamente

## DEBERES: ver clase anterior para el YAW y eso

Q el punto no sea uno fijo dado por mi, si no que sea un punto de otra entidad.

Puede haber una entidad player, que tiene un Transform, CmpInput, etc. Lo que queremos es la transform.

Ahora vamos a hacer que pueda ir a una posición o a una entity.

Pero los enemigos deben tener un campo de visión (hacer el test de visión cada frame F).

## Primero buscar otras entidades

* **Por nombre**: getEntityByName() y me da un Handle (se puede comprobar si isValid), me da el primero que encuentra

Imagen de la pantalla de un celular con letras

Descripción generada automáticamente con confianza baja

## Ahora vamos a hacer que una entidad cree una entidad

Vamos a lanzar un evento de he llegado al punto o algo así (algo puntual, que no sea cada frame). 🡪 SpawnEntity, y lo vamos a meter en un json (my\_bomb.json), pq es mas fácil tocar el json a mano que tocar los componentes y añadirlos a mano.

* Nombre
* Transform (de momento lo pondremos en el 0)

El spawn va a requerir un template spawn(prefab, contexto). El contexto trae info (quien es el padre, transform del mundo para q si spawneo aquí el 0,0 es este).

Deberes: la bomba tiene un timer.

(common/comp\_auto\_Destroy) se autodestruye

Yo puedo decir: dime todos los tios que tienen componente Vida y puedo iterar sobre estos componentes. Tengo un manager que le puedo decir componentes de vida de la escena. Cada componente puede ir a su padre para coger el transform. La parte de los mensajes no los tenemos pero vamos a ver como acceder a todos los componentes.

Tareas:

* Crear un componente CompLife (float life = 10)
* Ponerselo al player
* Bomba tiene un timer, (usar como ejemplo el comp\_autodestroy)
* cuando explota buscar las componentes de tipo CompLife y … coger su Owner (una Entity y mandar un mensaje a la entity). El objetivo del mensaje es q yo no sé lo que vas a hacer con el mensaje pero yo te digo que te he atacado. Tu ya verás si te afecta o no. **Si el componente explosión hablara directamente con el compnoente Life MALO. Estamos haciendo que la interacción sea muy estrecha. Si luego yo quiero añadir partículas y la bomba solo habla con el CompLife luego la entidad no se entera de que tienen que salir partículas. Lo suyo es hablar con la entidad y que ella gestione si le hace daño o si tiene que poner pareticulas o que.**

Para hacer un PUSH

Texto

Descripción generada automáticamente