Simulador de Gestión de Recursos en Fábrica Automatizada



Ignacio Tapia Marfil

Diseño y desarrollo de videojuegos y entornos virtuales.

13/05/2025

# Índice.-

[Índice.- 1](#_i9bi5ubqlhdd)

[Descripción General.- 2](#_a6kerecx1z88)

[Descripción Técnica.- 3](#_e4xogg41l6mz)

[Líneas: 3](#_z5mnx9ibil7w)

[Central: 6](#_3epnn95sniyj)

[Uw\_Central: 10](#_y3sm6otuea55)

[Diario de Desarrollo.- 14](#_ly3ac4pdr916)

[Bibliografía.- 15](#_ti9v1hvyvg9x)

# 

# Descripción General.-

Este proyecto ha sido realizado para la asignatura de Tecnologías de Desarrollo de Videojuegos y Entornos Virtuales III. Trata de realizar la simulación de una fábrica que maneja tres líneas de recursos. Esta fábrica debe contar con un sistema de emergencia en el que si una de las líneas entra en estado crítico haya un pequeño minijuego para arreglarla. Para esto vamos a utilizar el C + + de Unreal Engine en el que usaremos los delegados para comunicarnos entre objetos y las AsyncTask para lanzar los procesos en distintos hilos.

PUNTOS PRINCIPALES:

* **Líneas**: Las líneas son UObjects que cuentan con 2 variables, recursos y eficiencia. Estas dos variables que van de 0 a 100, fluctúan en cada tick del Update entre -10 y +5. También cuenta con un estado que fluctúa en función de estas variables, los estados son: Operativo, Emergencia y Crítico.
* **Central**: Se encarga de gestionar todo. Recibe los datos de cada una de las líneas para actualizar la Uw y lanzar los eventos necesarios.
* **Uw**: Recibe los datos de la central y actualiza todos los datos y eventos para que sean visibles en el Uw.

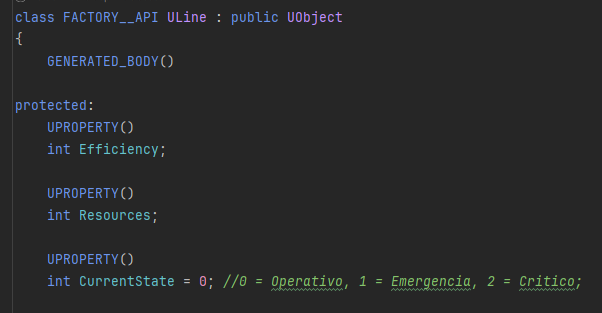
# 

# Descripción Técnica.-

## **Líneas:**

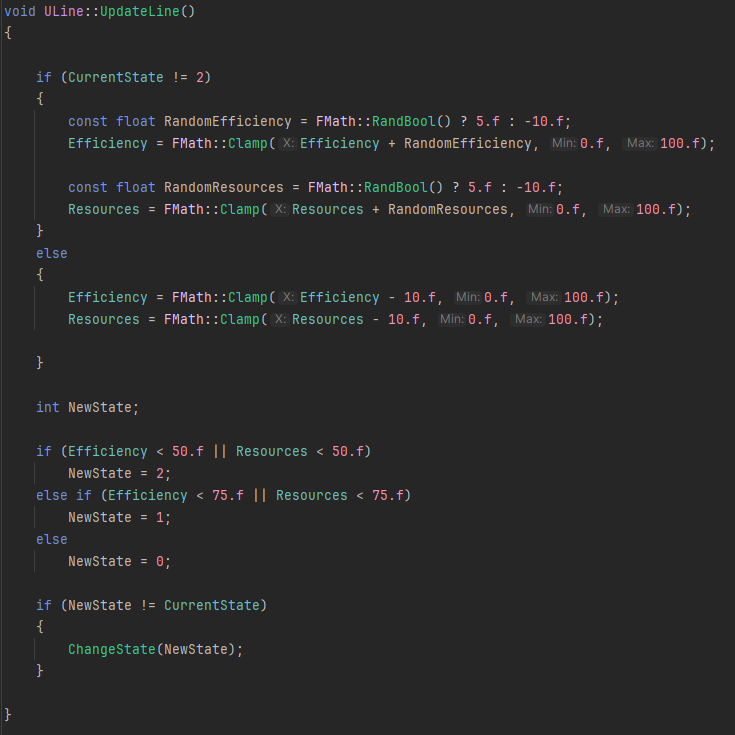
## 

Las líneas son un UObject que contiene tres variables: Resources y Efficiency, que son ints que van a fluctuar entre 0 y 100 y CurrentState que es un int que va de 0 a 2 e indica el estado actual en el que se encuentra la línea según el número de recursos o eficiencia.



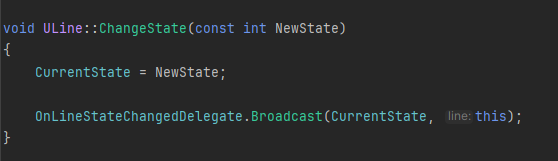
Estos objetos también tienen dos funciones principales, un Update y un ChangeState.

La función Update se va ha llamar desde el bucle principal en la central, desde un hilo individual para cada línea. El ChangeState se llama solo cuando hay un cambio de estado en la línea.



En el Update las variables van a fluctuar entre 0 y 100 en -10 y +5 salvo que la línea se encuentre en estado crítico que ahí solo baja.

Después si uno de estos valores pasa uno de los umbrales de estado se llama a la función ChangeState.



Aquí se cambia el estado de la línea y se lanza el delegado asignado para el cambio de estado.



Este es un delegado Multicast que envía dos datos, el estado al que se ha cambiado y la línea que cambia.

## 

## 

## 

## 

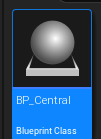
## 

## 

## 

## 

## **Central:**

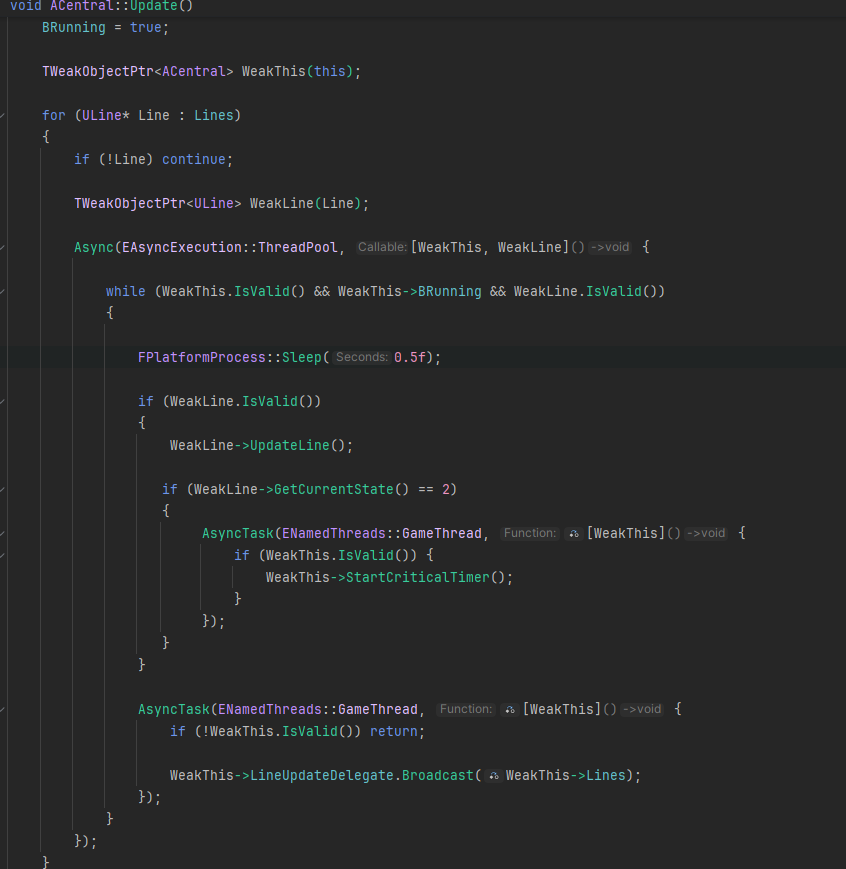


La central es un UActor en el que vamos a crear estas tres líneas y el Widget y va ha funcionar como nexo entre los dos.

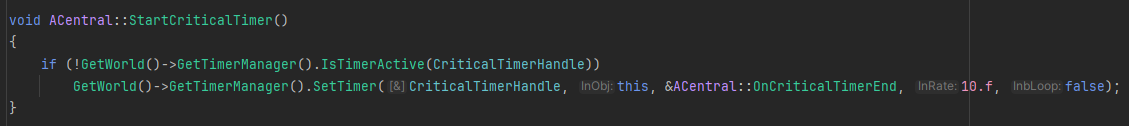


En el Begin Play de este Actor se inician las tres líneas y el Uw además de suscribirse a los respectivos delegados de cada línea y el Widget.

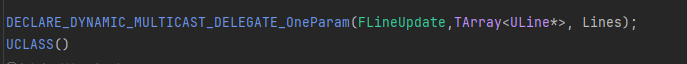


Además de esto se llama por primera vez al Update que va ha empezar el bucle principal de la fábrica.  
  


Aquí primero crea un puntero débil al propio objeto, esto para que cuando este objeto sea destruido, hacer que los hilos que lanzamos se corten.  
  
Después por cada una de las líneas se lanzan las acciones en hilos distintos al hilo principal del juego, dentro de estos hilos está el bucle principal desde el que llama a los Updates de la línea. Después, si la línea se encuentra en estado crítico lanza, desde el hilo principal del juego, la función que llama al timer de 10 segundos de estado crítico. Esta vuelta al hilo principal es porque en esa función se inicia un timer con el FTimerHandle y esto solo se puede hacer desde el hilo principal del juego.



Posteriormente y también desde el hilo principal del juego, va ha llamar a un delegado creado para enviarle la información de las líneas a él Uw.



Función que llama al recibir el delegado de cambio de estado de las líneas.



Después tenemos otras dos funciones las cuales, una es a la que llama cuando el temporizador crítico se acaba y para el bucle de ejecución de la central.  
La otra la llama cuando recibe el delegado del widget, este delegado se manda al arreglar una línea. Esta función para el temporizador crítico si no hay ninguna otra línea en ese estado.



## 

## 

## 

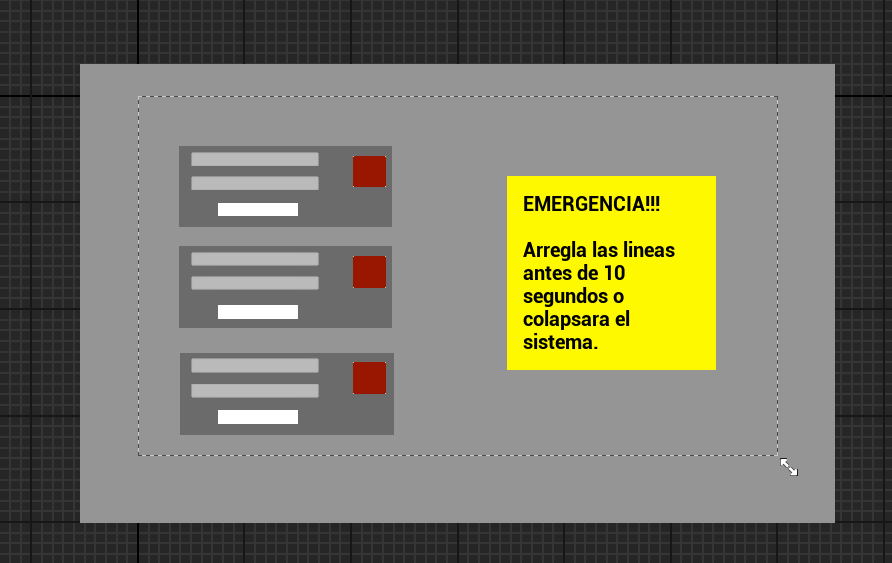
## 

## 

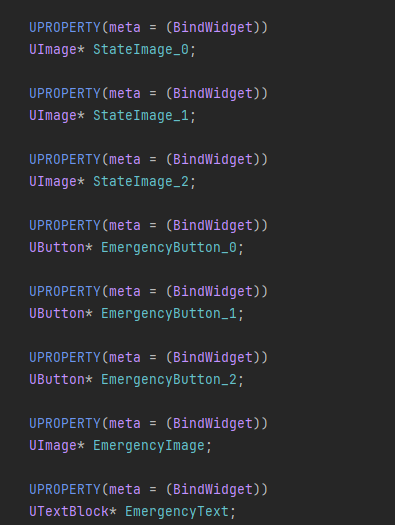
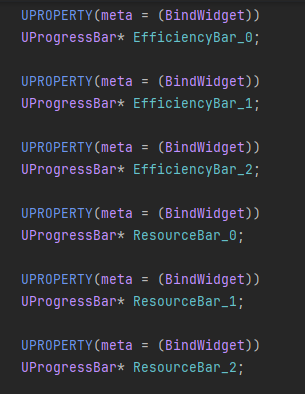
## 

## 

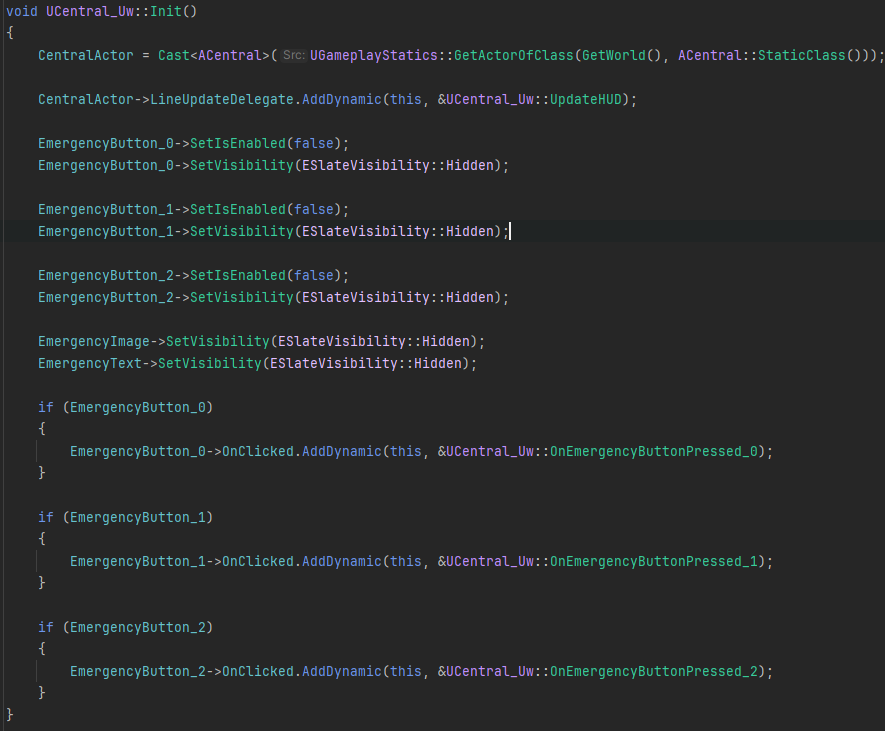
## **Uw\_Central:**



Para construir este UW lo primero que se hace es crear en C++ los componentes que se van a necesitar.



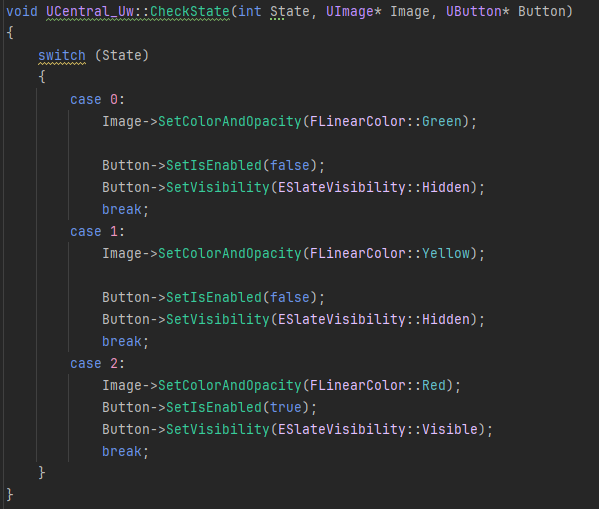
Se crea las barras de progreso para ver las variables de cada linea, una imagen par acada linea que cambiara de color dependiendo del estado de la misma. Los botones de cada línea y la imagen de emergencia junto a su texto, que se muestran cuando las líneas están en estado crítico.



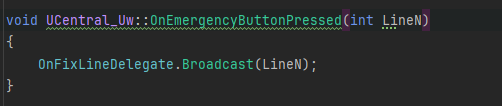
En el Init que se llama al crear el widget, oculta y desactiva todo lo que no se ve desde un inicio, se suscribe a el delegado de la central y establece las funciones del OnClick de cada boton.



Esta es la función que llama con el delegado del Central, esto se llama en cada tick y lo que hace es actualizar las barras de las variables, después busca si hay alguna línea en estado crítico para mostrar el mensaje de emergencia.  
  
También en cada línea llama al CheckState para ver si hay que mostrar el botón de emergencia para arreglar la línea y también para cambiar el color de la imagen de estado.



La función que llaman los botones, sirve para llamar al delegado que tiene la central.



# Diario de Desarrollo.-

Este proyecto realizado en UnrealEngine 5 ha supuesto un reto más grande de lo esperado, debido a que el tema de las AsyncTask no lo había tratado nunca y al principio no sabia muy bien por donde cogerlo .

Después de investigar un poco su funcionamiento y teniendo en cuenta que los delegados ya los había usado con anterioridad fue siendo más fácil.

# 

# 

# Bibliografía.-

[AsyncTask](https://dev.epicgames.com/documentation/en-us/unreal-engine/API/Runtime/Core/Async/AsyncTask)

[Chat Blackbox](https://www.blackbox.ai/)

[ChatGPT](https://chatgpt.com/)