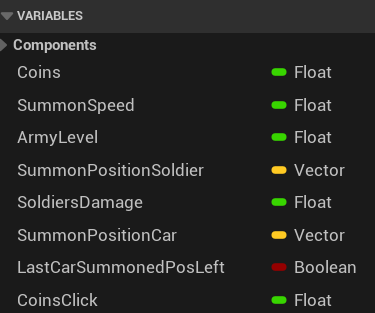
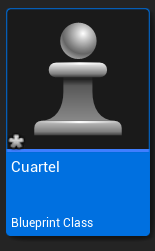
**Little Soldiers**

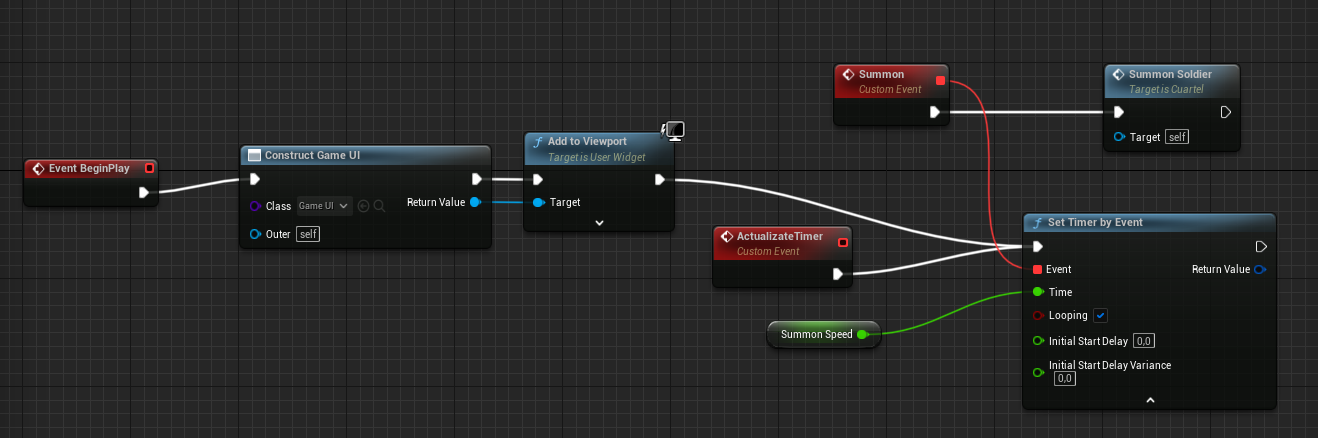
**Manuel Fernando Verutti Narvaez**

En el siguiente documento se explicará el funcionamiento de los blueprints que se implementaron en el desarrollo del juego.

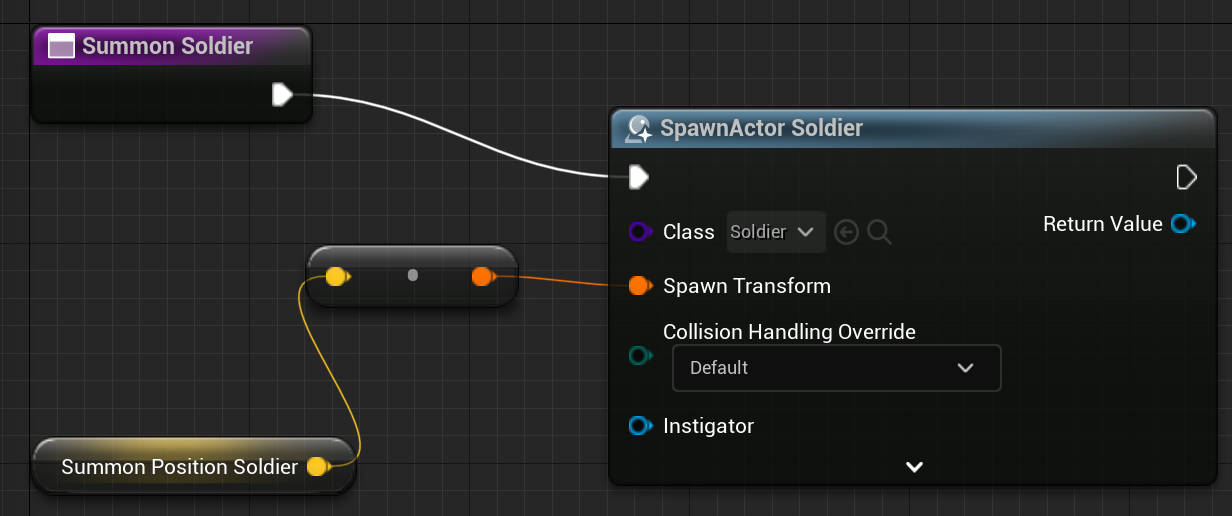
**CLASE CUARTEL**



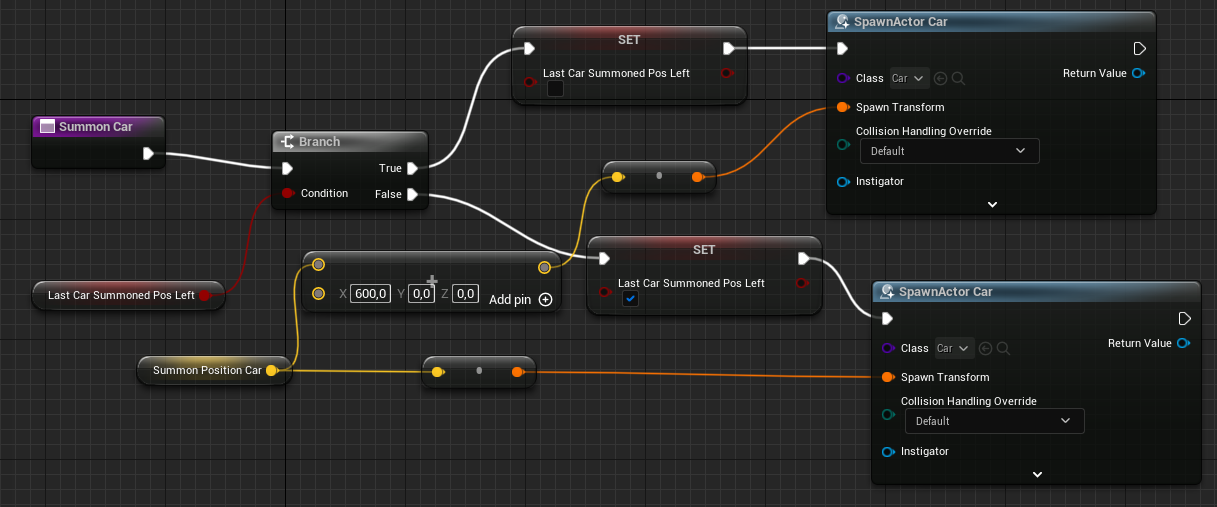
Este elemento es el Pawn que el jugador utilizará para interactuar con el juego.



Cuando se inicia este componente en el juego agrega al viewport la interfaz gráfica que contiene los botones, además se establece un timer que será el encargado de ejecutar cada cierto tiempo (Controlado por la variable summon speed) la creación de nuevos soldados.

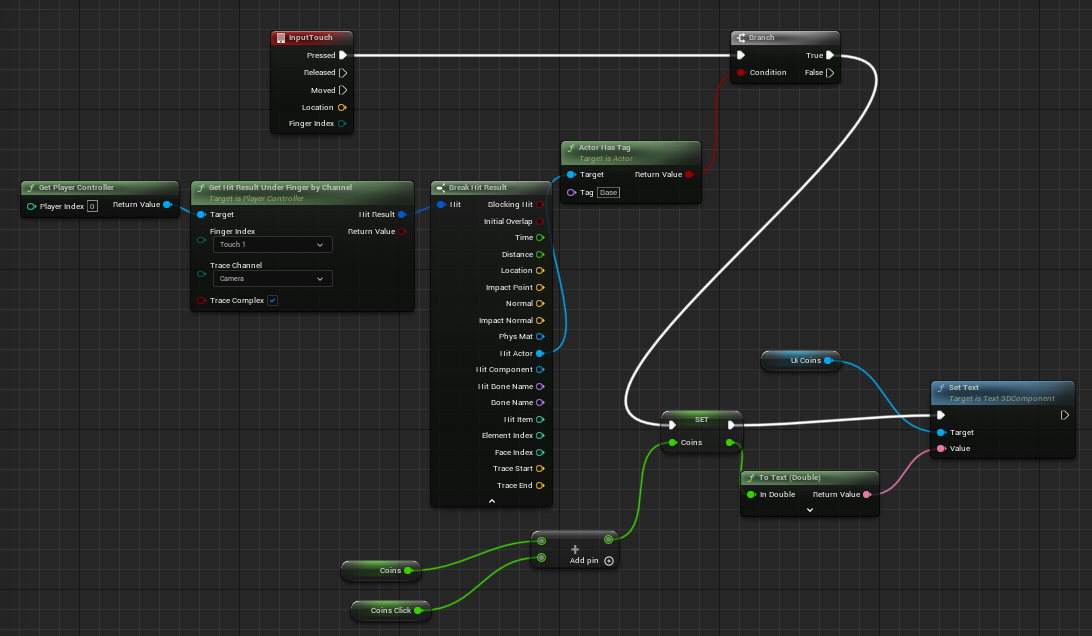


La función SummonSoldier instancia un actor de tipo soldier en un posición que se almacena en la variable SummonPositionSoldier.

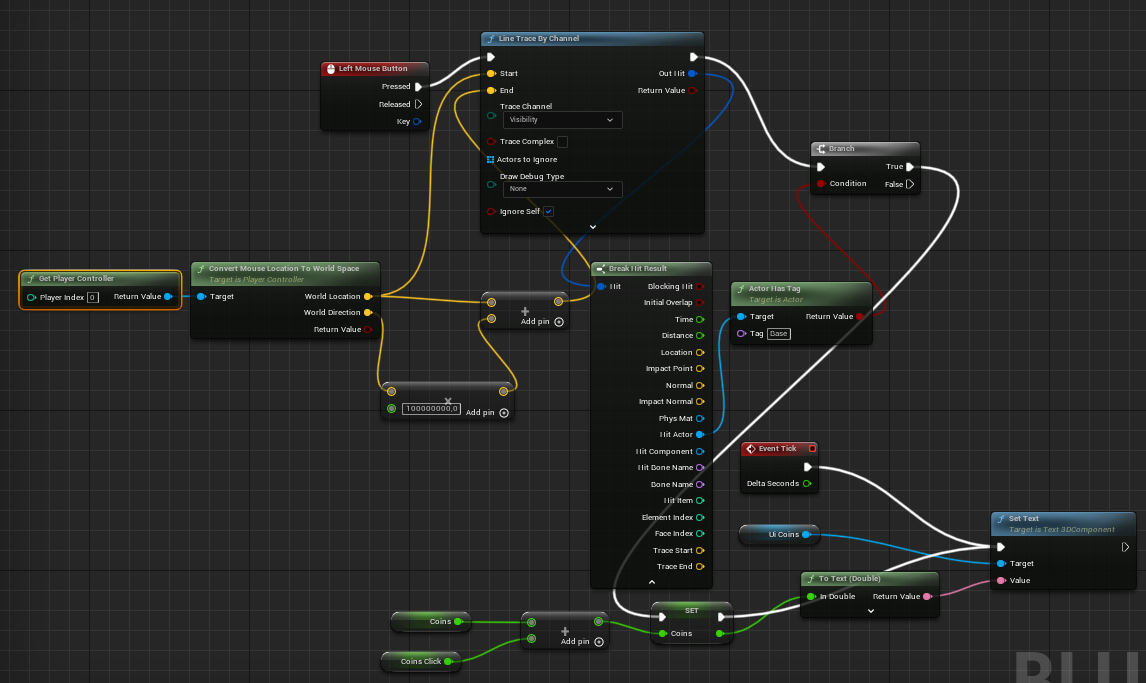


SummonCar es la función encargada de instanciar el tanque que se obtiene gastando 8mil puntos, en este caso se pregunta si la ultima posición de aparición del tanque fue a la izquierda, en caso de ser negativo se pondrá positivo y se instanciará el actor en la posición izquierda de la pantalla, sucederá en caso contrario si la variable es positiva.

Esto se hace con el objetivo de que los objetos puedan salir de los dos lados de la pantalla.

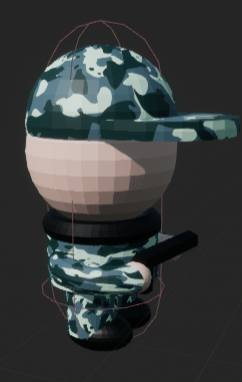


Con el evento Input Touch detectamos que el jugador ha presionado la pantalla, una vez se ejecuta llamamos al player controller y obtenemos el Actor que nos arroje la función Hit Result Under Finger, de esta forma se obtiene el objeto que presione el jugador, se pregunta si ese actor es de tipo base y en tal caso de serlo se aumenta la cantidad de puntos de acuerdo a la variable Coins Click, inicialmente esta variable tiene un valor de 1. Por ultimo actualizamos el texto que nos muestra la cantidad de puntos.

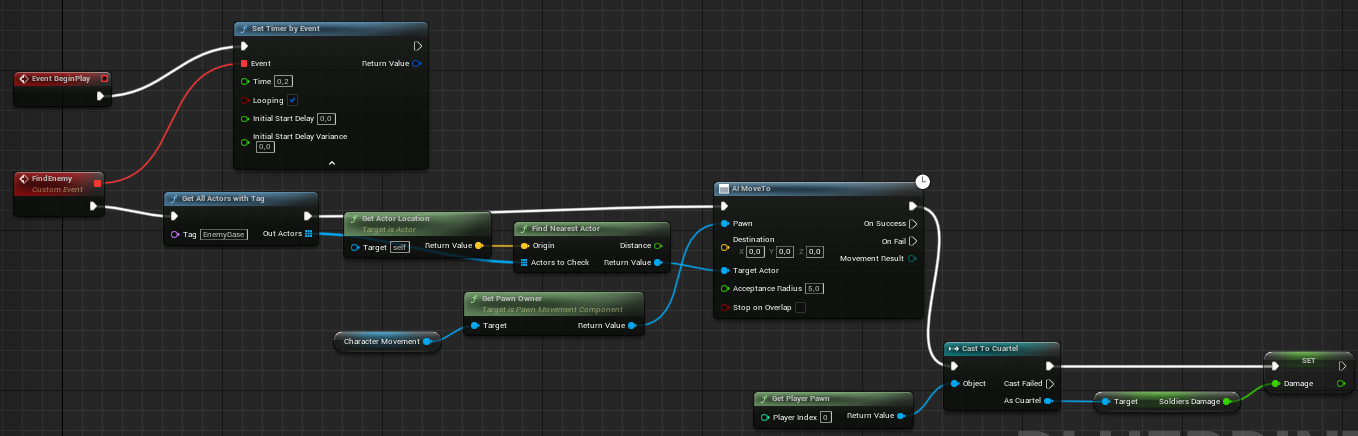


En caso que el jugador interactúe con el juego a través de un mouse, se obtendrá la posición del mouse en el espacio 3d del juego y se trazará un rayo que tendrá la dirección en que el jugador presionó, de esta forma obtendremos un Hit Result que en caso de tener el Tag Base aumentará los puntos del usuario tal y cómo se explicó en la interacción por pantalla táctil.

**Clase Soldado**



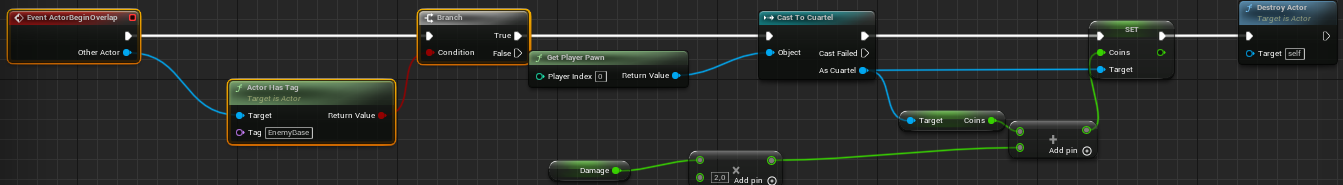
La clase soldado es aquella que contiene las funciones que le permiten el movimiento y la obtención de puntos a lo soldaditos instanciados.



Apenas el soldado se inicializa, utiliza la función timer para ejecutar el evento find enemy cada 0.2 segundos, se hace a través de un timer para no hacer uso del tick event, ya que este podria ocasionar problemas de optimización en dispositivos moviles.

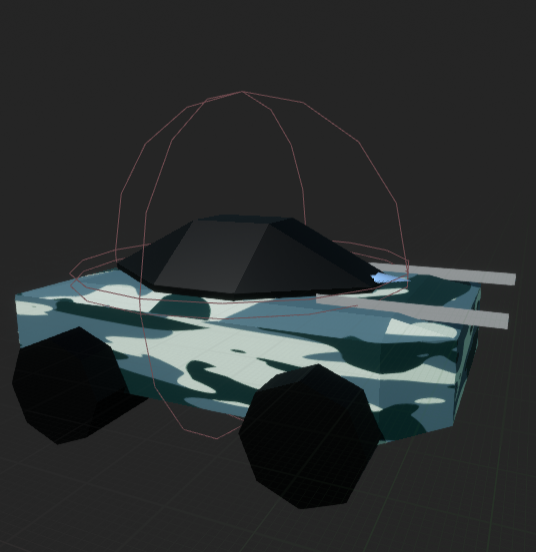
El evento find enemy busca todos los actores con el tag EnemyBase y obtiene su posición, posteriormente con el character Movement y la función Ai MoveTo hacemos que el personaje se dirija hacia la posición indicada.

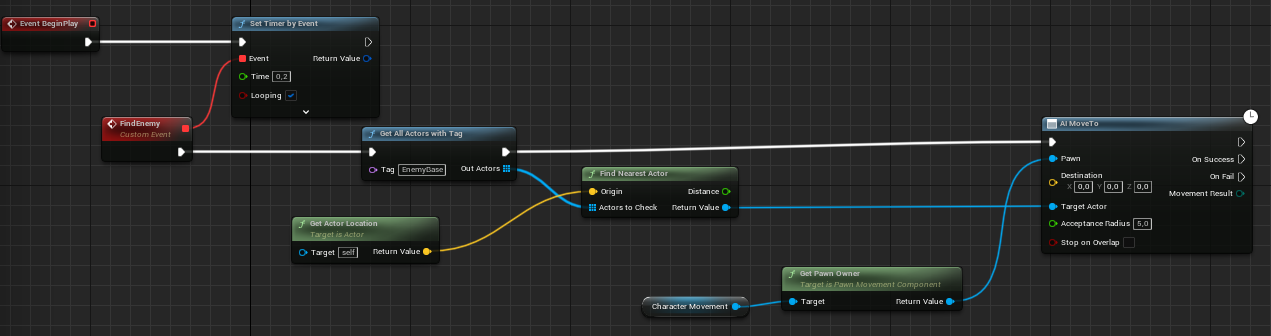
Por ultimo se establece el “daño” que el soldado hará (Cuantos puntos genera) de acuerdo a lo que la variable soldiers damage del Pawn tenga establecido.



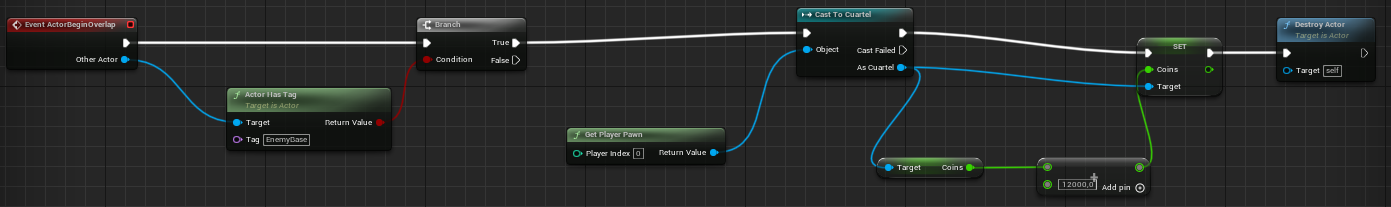
El evento Actor Begin Overlap se ejecutará cada que el mesh colisione con un objeto, en caso que el objeto tenga el tag EnemyBase, el soldado sumará a la variable coins del pawn el doble de la cantidad de daño que el soldado posea, finalmente de ejecuta la función destroy para eliminar el personaje.

**Clase Car**

****

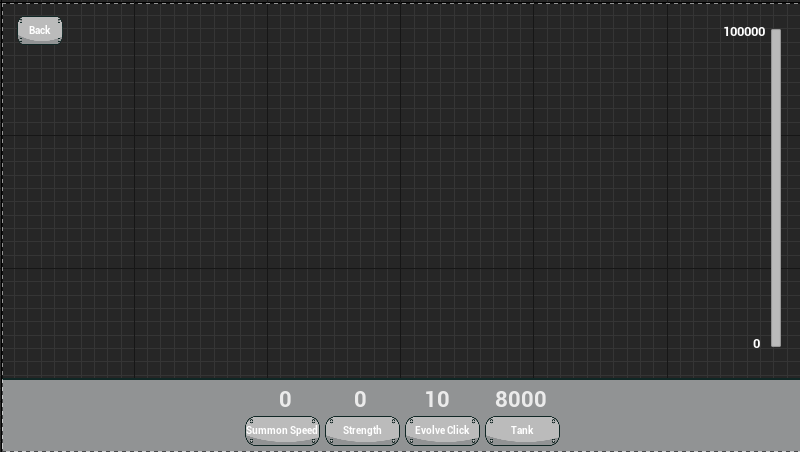
****

Al igual que el soldado, el carro utiliza el mismo sistema de movimiento para identificar la base enemiga y trasladarse.

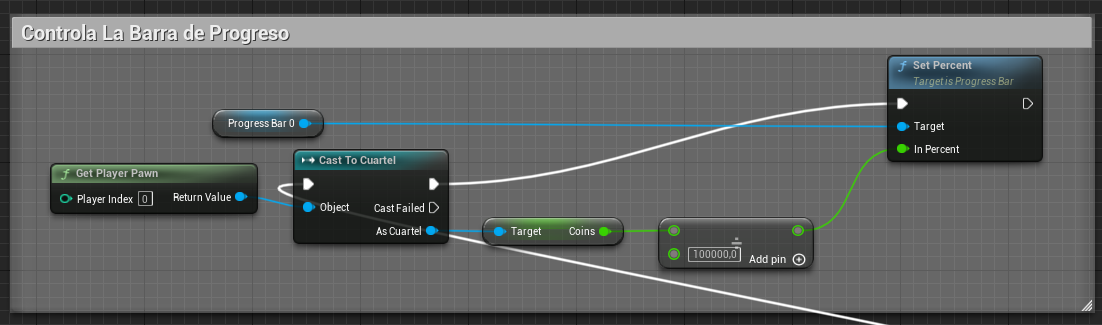
****

En caso que el tanque detecte la colisión con la base enemiga, sumará 12000 a la variable coins y finalmente ejecutará la función Destroy para eliminar su instancia.

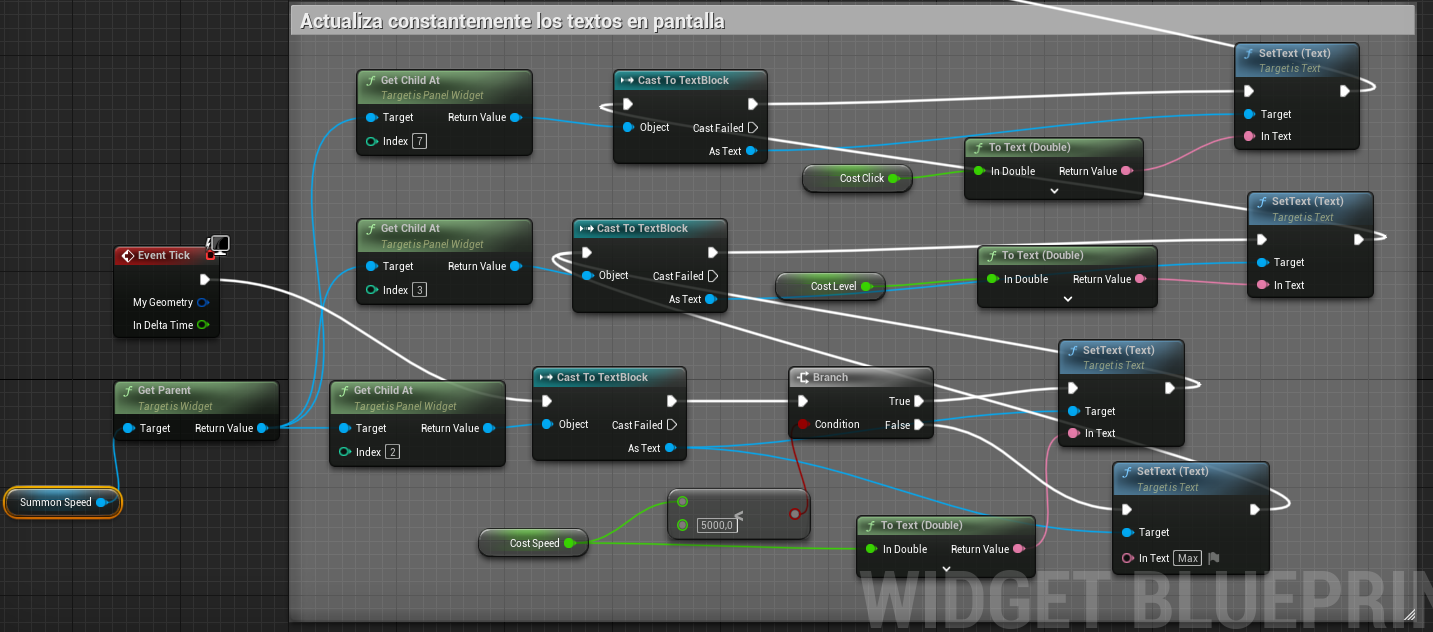
**Interfaz de Juego**

****

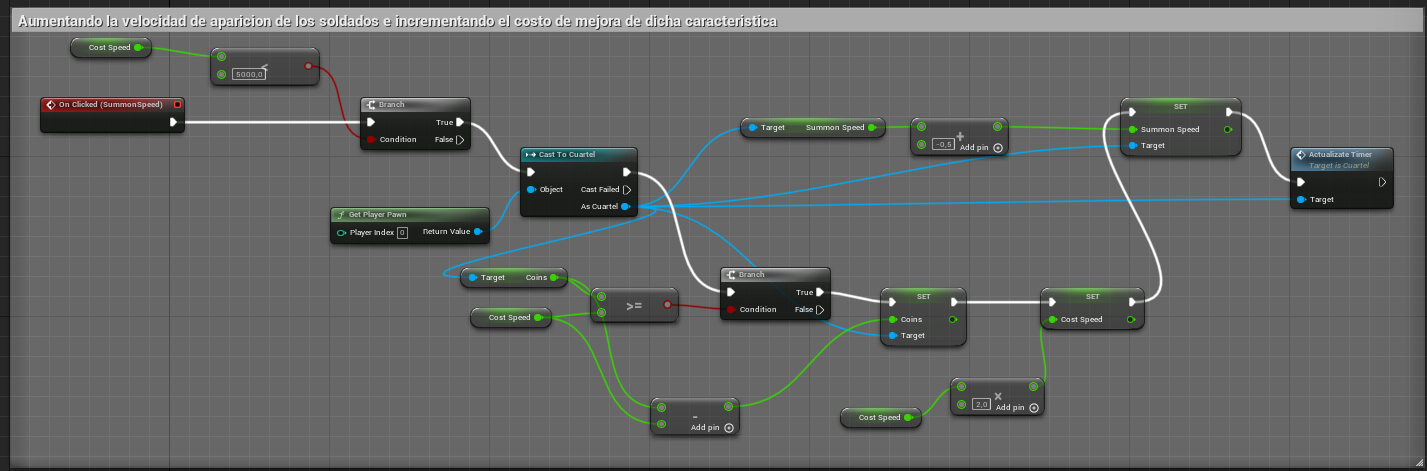
Esta es la interfaz gráfica que se muestra cuando el jugador se encuentra ya en juego.

****

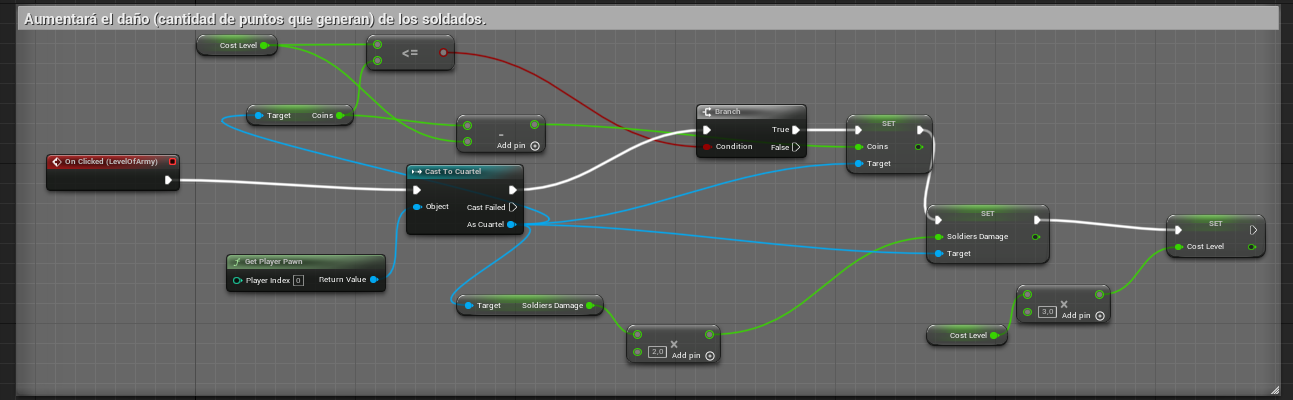
A partir de la variable Coins, se obtiene el porcentaje de 0 a 100.000 y se le envía a la Progress Bar para que se actualice.

****

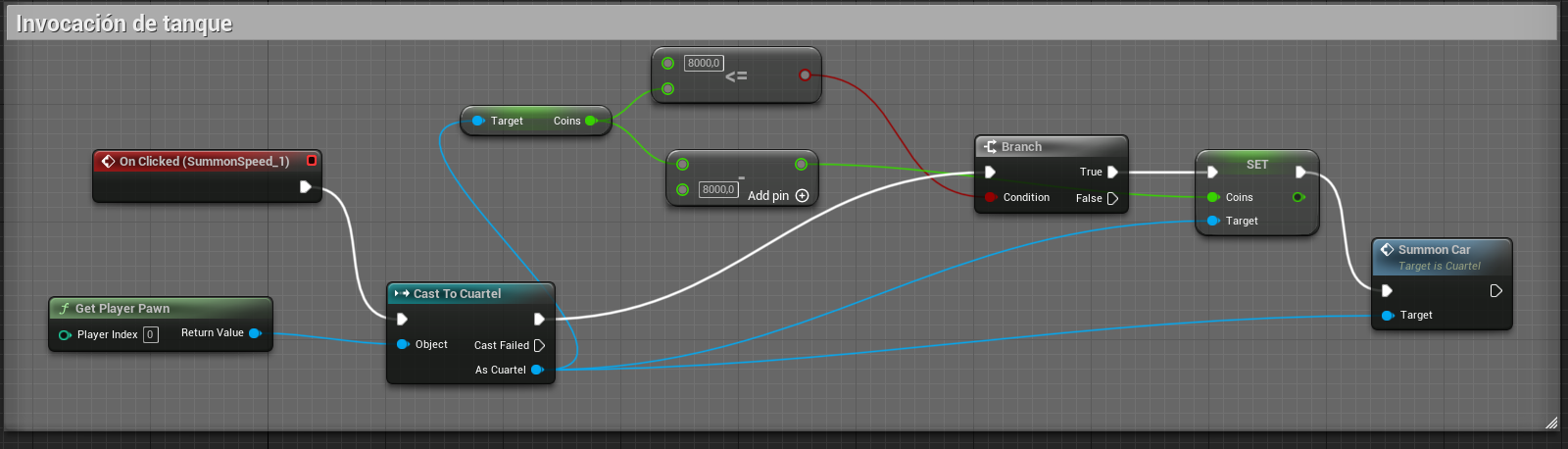
A partir del nodo padre se obtiene con el indice los textos en pantalla que se desean actualizar en tiempo real, esto se hace para los valores de costo de las mejoras y se ejecuta a través del Event Tick.



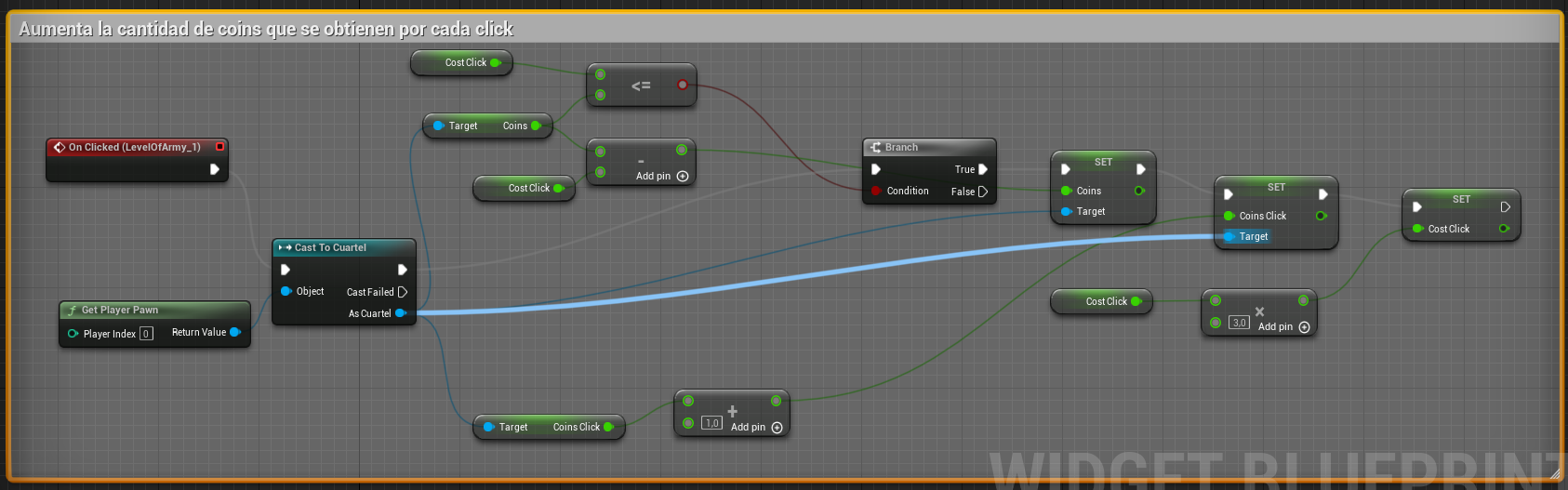
En caso que el jugador haga click sobre el botón Summon Speed, se preguntará si el costo de coins es inferior a 5000, esto ya que a este precio se encuentra la el maximo nivel de mejora para esta característica. Posteriormente se pregunta si el costo es menor o igual a la cantidad de Coins del jugador en caso de ser positivo de duplicará el costo de la mejora y se reducirá en 0.5 segundos la aparición de soldados.



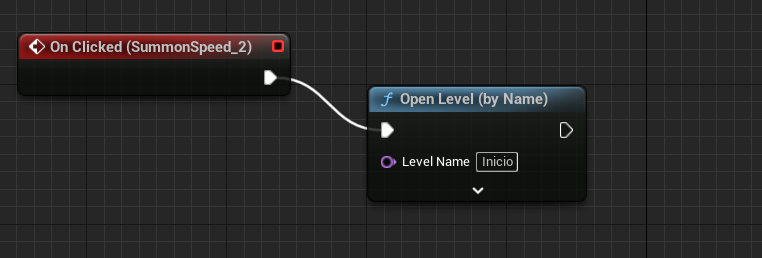
En caso que el jugador presione el botón, se confirma que posea la cantidad de coins necesarios y en caso de ser positivo se duplicará el daño de los soldados y se triplicará el costo de la mejora.



En caso de presionar el botón Tank se confirma que el jugador tenga 8000 coins, en caso de ser así se restará la cantidad y se ejecutará la función del Pawn Summon Car.

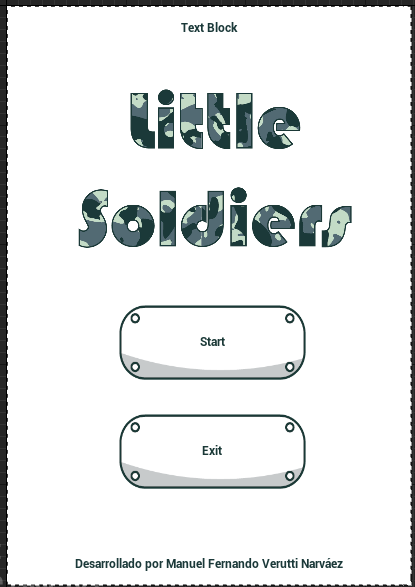


La ultima mejora permite al jugador aumentar la cantidad de coins obtenidos en 1 por cada click que haga sobre el Cuartel, como consecuencia el coste de esta mejora se triplicará.

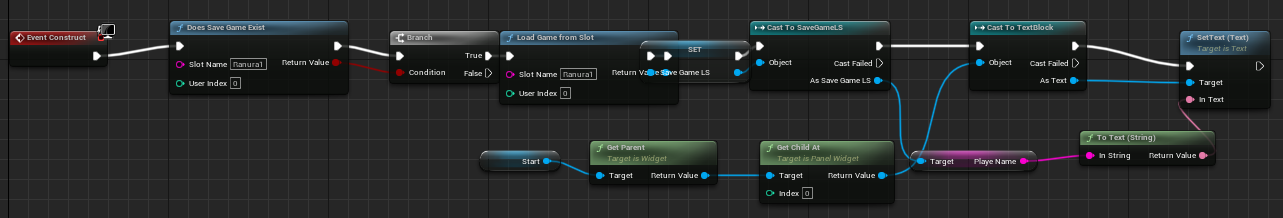


El botón BACK devolverá al jugador al menú principal.

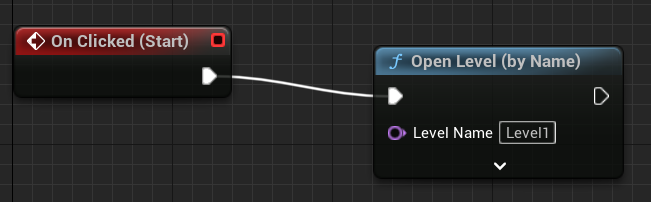
**Interfaz de Inicio**



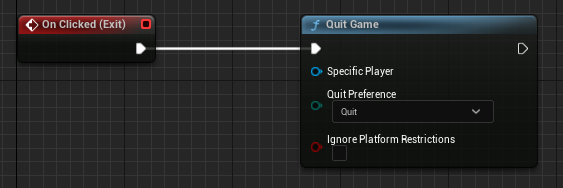
En el menú principal aparecen los creditos, el nombre del jugador que ingrese, el titulo y el botón para jugar,



Apenas se inicia la interfaz de inicio pregunta si existe un juego guardado en la ranura uno, si es así cargará los datos que existan en esa ranura y los casteará en un objeto de tipo SaveGameLS que es el blueprint que tiene las variables que se almacenan, se obtiene la variable player name y se imprimirá en el texto superior de la pantalla.



El botón Start abre el nivel 1 que será nuestro juego.

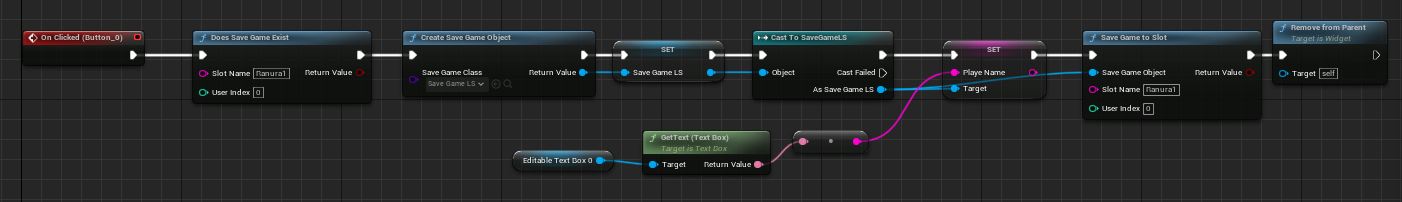


En caso de presionar Exit el juego será cerrado con la función Quit Game.

**Widget Ask Name.**



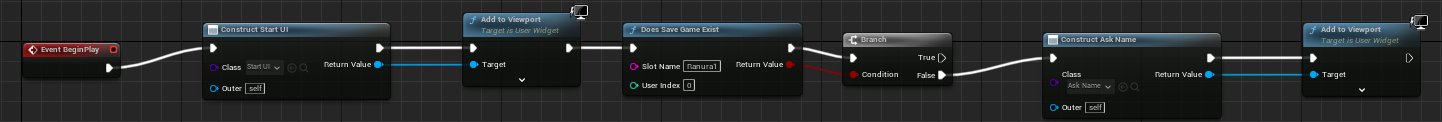
Este es el elemento de interfaz que preguntará el usuario la primera vez que ejecute el juego su nombre, estos datos los almacenará.



El botón hecho creara un objeto de tipo Save game, luego se hace el Set a una variable de la interfaz llamada Save Game LS que se castea en el blueprint diseñado para almacenar los datos, finalmente se llama a save game to slot donde se envia nuestro objeto creado save game ls para almacenar los datos en la Ranura1, por ultimo se elimina el widget de la interfaz.

**Nivel Inicio**

El nivel del menú principal cuenta con un blueprint que se ejecuta en su contrucción.



lo que hace es añadir a la pantalla la interfaz de inicio, posteriormente confirma si existe algún juego guardado en la ranura 1, en caso de no existir dibujará en pantalla el widget que preguntará y guardará el nombre del usuario.