



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

*Simulador 3D con tecnología i4.0 de Realidad Virtual para
adiestramiento inmersivo de tiro con arma de fuego*

Grado en Ingeniería Mecánica

ALUMNO: Rodrigo Bolívar Rodríguez

DIRECTORES: Xavier Núñez Nieto

CURSO ACADÉMICO: 2020-2021

Universida_{de}Vigo



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

*Simulador 3D con tecnología i4.0 de Realidad Virtual para
adiestramiento inmersivo de tiro con arma de fuego*

Grado en Ingeniería Mecánica
Intensificación en Tecnología Naval
Infantería de Marina

Universida_deVigo

RESUMEN

El auge actual de la Industria de Cuarta Generación (i4.0) ha supuesto un incremento significativo en la incursión y consolidación del mundo digital en el seno de las Fuerzas Armadas. En este marco operativo, la presencia de técnicas y metodologías orientadas a la simulación virtual para el adiestramiento militar, se ve reflejada en múltiples proyectos de gamificación que engloban a los tres ejércitos: Armada, Tierra y Aire. Este trabajo consiste en el empleo de dicha tecnología para el modelado 3D realista de un laboratorio de combate, que sirva como plataforma de entrenamiento virtual esférico (360°) para la práctica militar de tiro con arma de fuego. Mediante diversas técnicas de diseño avanzado se ha creado el entorno gráfico característico y se ha llevado a cabo la simulación fotorrealista del mismo. Además, se ha complementado el modelado asistido por ordenador con tecnología de Realidad Virtual en 360° (RV360), para conseguir una sensación totalmente inmersiva por parte del usuario. Todo el proceso se ha llevado a cabo mediante la combinación de diferentes técnicas y metodologías de cuarta generación.

PALABRAS CLAVE

Simulación de combate, Tecnología i4.0, Realidad Virtual, Gamificación, Inteligencia Artificial.

CONTENIDO

Contenido	1
Índice de Figuras	5
1 Introducción y objetivos	9
1.1 Descripción del apartado	9
1.2 Antecedentes	9
1.3 Justificación	10
1.4 Objetivos	10
1.5 Estructura del trabajo	11
2 Estado del arte	13
2.1 Descripción del apartado	13
2.2 Simulación en el ámbito de la Defensa	13
2.2.1 Simuladores en las Fuerzas Armadas	15
2.2.1.1 Ejército de Tierra	15
2.2.1.2 Ejército del Aire	17
2.2.1.3 Simuladores en la Armada	18
2.2.2 Simulación del tiro	19
2.2.2.1 VBS3	19
2.2.2.2 VICTRIX	20
2.3 Tecnologías i4.0	21
2.3.1 Inteligencia Artificial	21
2.3.2 Realidad Virtual, Aumentada y Mixta	21
2.4 Software y equipo	23
2.4.1 Software	23
2.4.1.1 Epic Games Store	23
2.4.1.2 Unreal Engine 4	24
2.4.1.3 Mixamo	24
2.4.2 Equipo y Material empleados	25
2.4.2.1 Lenovo Explorer	26
2.4.2.2 Tarjeta Gráfica NVIDIA RTX 2080 SUPER	27
3 Desarrollo	29
3.1 Descripción del apartado	29
3.2 Escenario y aplicación	29
3.2.1 Aplicación del simulador	29
3.2.2 Selección del entorno	31

3.2.2.1 Ejercicios de tiro.....	32
3.3 Diseño y modelado.....	34
3.3.1 Interfaz Unreal Engine 4.....	35
3.3.2 Estructura.....	38
3.3.2.1 E0: Preparatorio.....	38
3.3.2.2 Galería de Tiro.....	38
3.3.2.2.1 E1: Blanco único	40
3.3.2.2.2 E2: Blanco doble lateral	40
3.3.2.2.3 E3: Blanco doble en profundidad	41
3.3.2.2.4 E4: Blanco móvil.....	41
3.3.2.2.5 E5: Modo nocturno.....	41
3.3.2.3 E6: Tiro de precisión	42
3.3.2.4 E7: Poblado Afgano.	42
3.3.3 Third Person Player	43
3.3.3.1 Modelado del personaje.....	43
3.3.3.2 Modelado del HK-G36E	45
3.3.4 Blancos	47
3.4 Programación y código	48
3.4.1 Programación Blueprint.....	48
3.4.1.1 Tipos de Blueprint	49
3.4.1.1.1 Level Blueprint.....	50
3.4.1.1.2 Blueprint Class	50
3.4.1.1.3 Animation Blueprint.....	51
3.4.1.1.4 Widget Blueprint	52
3.4.1.2 Tipos de nodo	52
3.4.1.3 Variables.....	53
3.4.1.4 Materiales y Texturas	54
3.4.2 Programación del personaje.....	55
3.4.2.1 Componentes del personaje.....	55
3.4.2.2 Movimiento	56
3.4.2.3 Animaciones de movimiento.....	58
3.4.2.4 Cambiar de arma	59
3.4.2.5 Apuntar	60
3.4.2.6 Disparar	60
3.4.2.7 Recargar.....	61
3.4.2.8 Retroceso	62
3.4.3 Sistema de Armas	62

3.4.3.1	Disparar	63
3.4.3.2	Recargar.....	63
3.4.4	Proyectil.....	64
3.4.4.1	Componentes del proyectil.....	64
3.4.4.2	Generar agujeros de bala	65
3.4.5	Blancos	65
3.4.5.1	Sistema de puntuación.....	65
3.4.5.2	Aparición y caída de silueta	67
3.4.5.3	Movimiento	67
3.4.6	HUD.....	68
3.4.6.1	Menú Principal	68
3.4.6.2	FPS HUD.....	70
3.4.6.3	Pantalla de resultados	71
3.5	Realidad Virtual	72
3.5.1	Creación de BOLTRIX RV	73
3.6	Blog.....	73
4	Resultados	75
4.1	Descripción del apartado.....	75
4.1.1	Descripción del apartado	75
4.1.2	Escenario y aplicación	75
4.1.3	Diseño y modelado	75
4.1.4	Programación y código	78
4.1.5	Realidad Virtual.....	80
4.1.6	Blog.....	80
5	Conclusiones y líneas futuras	81
5.1	Descripción del apartado.....	81
5.2	Conclusiones	81
5.2.1	Escenario y aplicación	81
5.2.2	Diseño y modelado	81
5.2.3	Programación y código	82
5.2.4	Realidad Virtual.....	82
5.2.5	Conclusión final.....	82
5.3	Líneas futuras	82
6	Bibliografía.....	84
Anexo I:	Ejercicios de tiro de fusil de la brigada de infantería de marina	87

Anexo II: Bloques de código del personaje.....	95
Anexo III: Bloques de código de animación del personaje.....	113
Anexo IV: Bloques de código del arma	121
Anexo V: Bloques de código del proyectil.....	125
Anexo VI: Bloques de código del blanco silueta	127
Anexo VII: Bloques de código del blanco móvil.....	131
Anexo VIII: Bloques de código del blanco de precisión.....	137
Anexo IX: Bloques de código del blanco insurgente	141
Anexo X: Bloques de código del fps hud.....	147
Anexo XI: Bloques de código del menú principal	157
Anexo XII: Bloques de código de la pantalla de resultados.....	161
Anexo XIII: Manual de usuario de lenovo explorer (user guide)	165

ÍNDICE DE FIGURAS

Figura 1-1 La Industria 4.0. [2]	9
Figura 2-1 Portada de la ETID 2020 [5].....	14
Figura 2-2 Simulador en vivo DUELO-Individual. [6].....	16
Figura 2-3 Pirámide de empleo de simuladores en el ET. [6].....	17
Figura 2-4 Simulador de vuelo del F-18 en la base de Torrejón. [7]	17
Figura 2-5 Funcionalidades del Simulador Virtual Battlespace 3. [9].....	19
Figura 2-6 Componentes del simulador VICTRIX. [11]	20
Figura 2-7 Experiencia VR Zero Latency. [14].....	22
Figura 2-8 Aplicación de la AR en Pokémon-Go. [15].....	22
Figura 2-9 Programas de Software	23
Figura 2-10 Epic Games.....	23
Figura 2-11 Creación de un proyecto en Unreal Engine 4	24
Figura 2-12 Web de Mixamo.	25
Figura 2-13 Laboratorio de medidas	25
Figura 2-14 Dispositivo Lenovo Explorer.....	26
Figura 2-15 Tarjeta gráfica instalada.....	27
Figura 3-1 Etapas del trabajo.....	29
Figura 3-2 Clasificación de las Armas de Fuego	30
Figura 3-3 Galería de tiro de Figueirido.....	31
Figura 3-4 Situación de la Galería de Tiro de la Base Militar General Morillo.....	32
Figura 3-5 Portada del programa de Tiro de Fusil de la Brigada de Infantería de Marina.	33
Figura 3-6 Ejemplo de modalidad de tiro con sus correspondientes ejercicios.	34
Figura 3-7 Interfaz de Unreal Engine 4.....	35
Figura 3-8 UE4 - Buscador de contenidos.	36
Figura 3-9 UE4 - World Outliner.	36
Figura 3-10 UE4 - Gestor de Detalles.	36
Figura 3-11 UE4 - Buscador de Componentes.....	37
Figura 3-12 UE4 - Widgets de edición.....	37
Figura 3-13 UE4 - Barra de Herramientas.	37
Figura 3-14 Representación 3D de la galería de tiro.....	38
Figura 3-15 Detalle de las líneas de tiro.....	39
Figura 3-16 Detalle de las líneas de blancos.	39
Figura 3-17 Detalle del escenario E1.	40

Figura 3-18 Detalle del escenario E2.	40
Figura 3-19 Detalle del escenario E3	41
Figura 3-20 Detalle del escenario E5	41
Figura 3-21 Detalle del Escenario E6.....	42
Figura 3-22 Detalle del Escenario E7.....	43
Figura 3-23 Swat Guy	44
Figura 3-24 Unión entre personaje y arma	45
Figura 3-25 Modelo 3D del HK G36E	46
Figura 3-26 Edición de la malla esqueleto del arma	47
Figura 3-27 Blancos	48
Figura 3-28 ThirdPersonBP.....	49
Figura 3-29 Tipos de Blueprint principales.....	50
Figura 3-30 Blueprint de animación.....	51
Figura 3-31 Zona de Combinación.....	51
Figura 3-32 Nodo evento.....	52
Figura 3-33 Nodos función.....	52
Figura 3-34 Nodo de proyección.....	53
Figura 3-35 Nodo Create Widget	53
Figura 3-36 Tipos de variables	54
Figura 3-37 Plano de Material.....	54
Figura 3-38 Componentes del personaje	55
Figura 3-39 Generar HK G36.....	56
Figura 3-40 Entradas del proyecto	57
Figura 3-41 Función agacharse	57
Figura 3-42 Variables de animación	58
Figura 3-43 Zonas de combinación.....	58
Figura 3-44 Función cambiar arma	59
Figura 3-45 Transición entre arma primaria y secundaria.....	59
Figura 3-46 Función desactivar ADS	60
Figura 3-47 Función Disparar (1).....	61
Figura 3-48 Función Disparar (2).....	61
Figura 3-49 Widget de munición.....	61
Figura 3-50 Función Recargar.....	62
Figura 3-51 Función Retroceso	62
Figura 3-52 Función Fire.....	63
Figura 3-53 Fragmento de la función Reload.....	63

Figura 3-54 Componentes del proyectil	64
Figura 3-55 Función Generar agujeros de bala	65
Figura 3-56 Blanco silueta.....	66
Figura 3-57 Función Sistema de puntuación	66
Figura 3-58 Blanco de precisión.....	67
Figura 3-59 Función de movimiento de silueta.....	68
Figura 3-60 Menú Principal	69
Figura 3-61 Botón Seleccionar Escenario	69
Figura 3-62 Ajustar munición	70
Figura 3-63 Cargar Escenario.....	70
Figura 3-64 Obtener valor de la munición	71
Figura 3-65 Imagen en tiempo real del blanco	71
Figura 3-66 Pantalla de resultados	72
Figura 3-67 Generar y guardar pantalla de resultados.....	72
Figura 3-68 Edición del blog.....	74
Figura 3-69 Código QR del Blog	74
Figura 4-1 Detalles del escenario principal	76
Figura 4-2 Sistema de puntuación	76
Figura 4-3 Zoom del visor óptico.....	77
Figura 4-4 Efecto de la gravedad y el viento.....	77
Figura 4-5 Detalles del poblado afgano.....	78
Figura 4-6 Sistema de puntuación de precisión.....	79
Figura 4-7 Resultados de la simulación	79
Figura 4-8 Resultados almacenados	80
Figura 4-9 Blog BOLTRIX RV	80

1 INTRODUCCIÓN Y OBJETIVOS

1.1 Descripción del apartado

En este apartado inicial se expone el porqué de este proyecto, atendiendo a una serie de aspectos que, analizados desde un punto de vista general para posteriormente profundizar en el ámbito de las Fuerzas Armadas, y más concreto de la Armada, justifiquen el posterior desarrollo del proyecto. Para ello se definirán unos antecedentes, seguidos de la justificación del proyecto y culminados con el establecimiento de los objetivos a alcanzar en el presente trabajo. Por último, se detallará la estructura empleada en la presente memoria.

1.2 Antecedentes

A lo largo de la historia, el ingenio humano ha logrado desarrollar tecnologías que han dado pie a las sucesivas revoluciones industriales. La máquina de vapor, la optimización de la fabricación y la producción en masa, los avances en electrónica e informática... son tan solo algunos ejemplos. Hoy nos encontramos frente a la Cuarta Revolución Industrial gracias al desarrollo de sofisticados sistemas de software, el empleo de sensores y procesadores de alta capacidad, las nuevas tecnologías de telecomunicaciones... La conjunción de todas estas nuevas tecnologías es lo que hoy conocemos como la Industria 4.0 o i4.0. Esta se sustenta, entre otras, en tecnologías tales como la robótica colaborativa, la Realidad Virtual y Realidad Aumentada, el Big Data, o Internet de las cosas. [1]

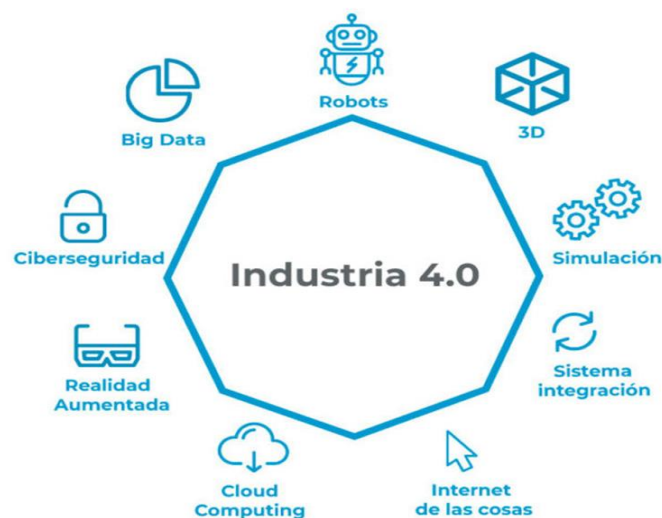


Figura 1-1 La Industria 4.0. [2]

En el presente proyecto nos centraremos en algunas de las tecnologías de la industria 4.0 antes mencionadas y mostradas en la Figura 1-1 ya que serán la base sobre la que sustentará el mismo. Se trata de los conceptos de simulación, realidad aumentada y 3D.

La Armada, al igual que muchas otras organizaciones, trata de mantenerse actualizada y ser puntera en este nuevo entorno 4.0 lleno de cambios y constantes evoluciones. Como tal, la Armada ha estado presente en todas las revoluciones industriales y esta no puede ser diferente. En este sentido, la experiencia acumulada y heredada permiten a la Armada adentrarse en este nuevo océano tecnológico con un sentido y rumbo propios, sin perder de vista su presente y su misión. [3]

Dentro del programa de la Semana Naval de Madrid, se celebró en su octava edición en 2018 una mesa redonda titulada: “Universo 4.0. Impacto en las organizaciones.” Donde se debatió el impacto de las nuevas tecnologías en la cultura y en las organizaciones y se presentó, por primera vez, la Visión de la Armada 4.0 [4]. La participación de la Armada de este nuevo mundo 4.0 se hace patente en diversos aspectos como el del Sosténimiento 4.0 y el Apoyo Logístico 4.0, el desarrollo de los astilleros 4.0 para las Fragatas F-110 o la implementación de simuladores de combate en diferentes ambientes.

Como no podía ser de otro modo, la Enseñanza Naval también se está viendo influenciada por este nuevo entorno 4.0 y las nuevas tecnologías. En la preparación de cara al futuro la tecnología será un factor diferencial y la Armada debe procurar la innovación y la adaptación a estos nuevos sistemas para aprovechar al máximo sus posibilidades. Entre otras, las aportaciones de estos nuevos sistemas podrían incluir la incorporación de contenidos inteligentes y el apoyo virtual de la enseñanza por medio de la gamificación y la visualización 3D. [3]

Estos últimos junto con las técnicas de Realidad Virtual o Aumentada abren una infinidad de posibilidades que abarcan desde la visualización de conceptos imposibles de reproducir de otra manera hasta el aprendizaje inmersivo mediante la simulación. En este concepto es el que nos concierne de cara al desarrollo del presente proyecto. [3]

Tanto es así que las Fuerzas Armadas están haciendo enormes esfuerzos por modernizarse en este campo, dando gran importancia a la simulación de entornos lo más parecidos a los que el combatiente puede encontrarse en situaciones de combate reales. Aunque existen en la actualidad infinidad de simuladores para todo tipo de ambientes, encontrar o desarrollar los adecuados para un adiestramiento lo más aproximado a la realidad es todo un reto. Dentro del Portal de Tecnología e Innovación del Ministerio de Defensa se ha creado dentro de la Estrategia de Tecnología e Innovación una línea de investigación tecnológica dedicada exclusivamente a la simulación para el adiestramiento de los combatientes. [5]

1.3 Justificación

Como hemos podido comprobar, el desarrollo de simuladores para el adiestramiento eficaz de los combatientes es un factor fundamental para la adaptación de la Armada al mundo 4.0, dotándole de los medios modernos para mantenerse actualizada en un mundo en constante desarrollo, sin perder nunca de vista su misión principal.

Por lo tanto, queda justificado el objetivo principal de este Trabajo de Fin de Grado en el cual se pretende diseñar un simulador 3D con tecnología propia de la industria 4.0 como la Realidad Virtual para adiestramiento inmersivo de los combatientes con armas de fuego. Esto permitirá dotar a la Armada de un medio capaz de complementar y, si fuese necesario, sustituir el adiestramiento con armas de fuego en escenarios reales.

1.4 Objetivos

El propio título del proyecto establece el objetivo principal del mismo, siendo este el diseño de un simulador 3D con tecnología i4.0 de Realidad Virtual para el adiestramiento inmersivo de tiro con arma

de fuego. Este objetivo se puede desglosar a su vez en los siguientes objetivos secundarios implícitos al mismo:

- Diseñar un simulador de tiro 3D empleando un motor de juego que aplica la Gamificación y que nos permita crear un entorno inmersivo de Realidad Virtual.
- Lograr un modelado realista del escenario en el que se desarrollará la simulación y que favorezca la inmersión del usuario en el simulador.
- Diseñar un simulador capaz de replicar ejercicios reales de adiestramiento con armas de fuego de la manera lo más realista posible.

De manera adicional y complementaria a los objetivos anteriormente expuestos se han alcanzado una serie de objetivos a raíz del desarrollo del proyecto:

- Lograr una programación adecuada para que todos los componentes de los escenarios cumplan con las funcionalidades requeridas dentro del mismo para que el simulador cumpla con el objetivo principal.
- Desarrollar un escenario de mayor complejidad en la que el usuario puede adiestrarse en el uso de armas de fuego dentro de situaciones similares a las que puede encontrarse en la realidad.
- Generar un blog explicativo del presente proyecto y que pueda servir de referencia a futuros proyectos dentro del mismo ámbito.

1.5 Estructura del trabajo

La estructura que seguirá la presente memoria será, de acuerdo a los criterios establecidos por el Centro Universitario de la Defensa, la siguiente:

1. Introducción y objetivos: Se describe la motivación del presente proyecto y sus antecedentes y se establecen los objetivos que se busca cumplir en su desarrollo.
2. Estado del arte: Se hace una detallada descripción de la situación actual de los ámbitos con los que se relaciona el presente proyecto, además de presentar el software y equipo empleados en su ejecución.
3. Desarrollo: Se expone de manera más detallada el transcurso de la ejecución del proyecto, englobando las diferentes partes en las que se divide el mismo.
4. Resultados: Se describen los frutos de la ejecución y desarrollo del proyecto, cuyo producto final es el simulador 3d con tecnología i4.0 de Realidad Virtual para adiestramiento inmersivo de tiro con arma de fuego.
5. Conclusiones y líneas futuras: Se hace un análisis del desarrollo y resultados del proyecto, añadiendo posibles evoluciones y aplicaciones del mismo.

2 ESTADO DEL ARTE

2.1 Descripción del apartado

En este apartado se trata, en primer lugar, de aclarar los conceptos relativos a los ejercicios de tiro realizados en el ambiente de las Fuerzas Armadas en la actualidad. Además, se desarrolla de forma genérica el estado actual de los diferentes aspectos relevantes del proyecto; en primer lugar, los ejercicios de tiro, a continuación, el desarrollo y adquisición de simuladores dentro de las Fuerzas Armadas, y dentro de estos, los específicos de tiro, seguido de las tecnologías i4.0 y, por último, el material empleado para el desarrollo del proyecto.

2.2 Simulación en el ámbito de la Defensa

Las Fuerzas Armadas están inmersas en una constante lucha por la mejora y el perfeccionamiento del adiestramiento de todos sus miembros. Dado que las condiciones para la realización de un adiestramiento convencional no son siempre óptimas, se está procediendo al desarrollo de sistemas innovadores que mitiguen los efectos de estos condicionantes. Al mismo tiempo, estos sistemas tratan de alcanzar los requisitos necesarios para lograr los niveles de adiestramiento requeridos en los diferentes currículos de la formación militar.

La tendencia, cada vez mayor, a economizar al máximo los recursos, y con ello el consumo de munición, la limitación de espacios correctamente habilitados para el tiro, el tiempo disponible y las condiciones meteorológicas impredecibles hacen que, en la actualidad, la realización de todos los ejercicios de los diferentes planes de tiro no siempre sea posible.

La ETID (Estrategia de Tecnología e Innovación) del Ministerio de Defensa trata de dar respuesta a esta situación. El desarrollo de la ETID ha permitido definir cuáles son las Metas Tecnológicas que guían y definen las actividades de I+D a realizar en los próximos años. Estas metas tecnológicas se agrupan en 6 Áreas y Líneas de Actuación Funcionas, las cuales se subdividen en diferentes líneas tecnológicas. [5]

Dentro del área de los sistemas C4I (Mando y Control, Comunicaciones, Computación e Inteligencia) se encuentra la familia tecnológica encargada de los aspectos tecnológicos que favorecen el empleo de simuladores en el ámbito de la Defensa y la Seguridad. Esta es la Simulación para adiestramiento y CD&E (Experimentación y Desarrollo de Conceptos). A esta familia pertenecen las siguientes 3 metas tecnológicas: [5]

- MT 6.4.1.: Modelado del campo de batalla y de su entorno. [5]
- MT 6.4.2.: Infraestructura de simulación y reutilización de componentes. [5]
- MT 6.4.3.: Apoyo al desarrollo de conceptos y soluciones técnicas mediante simulación. [5]



Figura 2-1 Portada de la ETID 2020 [5]

La primera de ellas engloba dentro del modelado del campo de batalla y su entorno a todos los simuladores para el combate, independientemente de su ámbito de aplicación. De este modo las Fuerzas Armadas tratan de dar solución al problema en cuestión, desarrollando tecnologías con esta capacidad a nivel nacional o contratándolas a nivel internacional.

La simulación en el ámbito de la defensa permite y favorece el adiestramiento en situaciones más complejas y adversas que las que se dan durante el entrenamiento convencional. Las nuevas amenazas, la guerra asimétrica y la complejidad de los escenarios en los que se desarrollan la mayoría de los combates hoy en día requieren que el combatiente se adapte a una serie de situaciones a las que no está acostumbrado. La simulación es la solución a esto, permitiendo el desarrollo de estos entornos según las necesidades de la unidad. [6]

La manera de anticiparse a las amenazas y la incertidumbre que predominan en el entorno actual consiste en el empleo de sistemas tecnológicamente avanzados y superiores a los del enemigo. Pero, al mismo tiempo, es necesario que los combatientes que operen estos nuevos sistemas tengan el adiestramiento adecuado en su uso para poder llevar a cabo su misión. Ante esto, la simulación permite reflejar con un alto grado de exactitud la realidad a la que se va a enfrentar el combatiente, permitiendo el adiestramiento del mismo a un coste muy reducido. [6]

De entre las tecnologías empleadas para la simulación, la Realidad Virtual permite eliminar las barreras espaciotemporales mediante la reproducción de una nueva realidad que captan los sentidos. Esta tecnología puede ser de dos tipos:

- Inmersiva: Un ordenador crea un ambiente tridimensional que puede ser manipulado a través de sistemas periféricos que son capaces de detectar y capturar la posición y rotación del cuerpo humano. [6]
- No inmersiva: Ofrece una realidad diferente a través de una pantalla y permite al usuario interactuar con diferentes personas en espacios inexistentes en tiempo real sin necesidad de medios adicionales al propio ordenador. [6]

La Realidad Virtual se puede presentar mediante diversos métodos: A través de simuladores que hacen que el usuario perciba que se encuentra dentro de un ambiente real, con avatares prediseñados o desarrollados, mediante la proyección de imágenes reales para generar modelos realistas en 3D, con un ordenador que muestra un mundo interactivo en tres dimensiones aunque carezca de visión periférica, y por último, mediante la inmersión en entornos virtuales que permite experimentar la realidad virtual a través de una conexión a una interfaz cerebro-máquina. [6]

Las aplicaciones que la Realidad Virtual tiene dentro del marco de la Defensa y la Seguridad son muchas, de entre las que destacan las siguientes:

- Adiestramiento y entrenamiento militar: Permite la inmersión e interacción con el entorno mediante dispositivos que permitan al usuario sentirse dentro de una situación real. Se emplean dispositivos que reproduzcan de una manera lo más verosímil posible todas las sensaciones que el combatiente puede encontrarse en multitud de situaciones de combate. [6]
- Condiciones NRBQ: Permite simular las situaciones propias de un ambiente provocado por un ataque NRBQ con el fin mejorar el adiestramiento de los combatientes en este tipo de guerra. [6]
- Psicología: Permite una interacción íntima del usuario con el entorno, analizando cómo reacciona el mismo a determinadas situaciones no desde un punto de vista táctico si no psicológico. [6]
- Medicina: Facilita la simulación de tratamientos traumatológicos, cirugías y procedimientos de evaluación médica que permiten lograr una mayor eficacia y eficiencia en futuras situaciones reales. [6]

Dado que en la simulación siempre existe una diferencia con la realidad, las Fuerzas Armadas tratan de investigar y desarrollar estas tecnologías para minimizar este diferencial al mínimo. De este modo estas tecnologías pueden aportar beneficios realmente diferenciadores con respecto a los medios convencionales de adiestramiento.

2.2.1 Simuladores en las Fuerzas Armadas

El 24 de abril de 2018 se celebró la Jornada Tecnológica de Simulación en el ámbito de la Defensa con el fin, entre otros, de presentar cuál es la situación actual de los sistemas de simulación y adiestramiento de los Ejércitos y la Armada, así como para definir los retos tecnológicos a los que se desean alcanzar. [6]

A continuación, se exponen de manera breve y concisa los medios actuales de los que se dispone.

2.2.1.1 Ejército de Tierra

En el Ejército de Tierra, los simuladores empleados en la actualidad se clasifican, según su naturaleza en:

- Simuladores reales o simulación en vivo: son sistemas para el adiestramiento de unidades en campos de maniobras, empleando equipos que simulan las acciones propias del combate. Actualmente se emplean los siguientes: [6]
 - Blancos: SAAB y Sistema LOMAH.
 - Duelo: Individual, C/C Leopardo, VCI. Pizarro, P. Árbitro, E. Análisis...
 - Tiro: NOPTEL, VICTRIX y Tirador de precisión STS y NOPTEL.
 - NBQ: CAMSIM y AN/PDR-77.
 - IED (Improvise Explosive Device): Kit IED y C-IED.

- Simuladores virtuales: Sistemas que representan los medios, el terreno y determinadas situaciones, permitiendo la interacción del usuario. Sirven en especial para el adiestramiento en el uso de sistemas de armas como carros de combate, misiles o plataformas aéreas. Actualmente se emplean los siguientes: [6]
 - Helicópteros: FBMS y FMS.
 - Carros y vehículos de combate: AULA EAO y CC LEOPARD 2A4 y 2E.
 - Escuela de conductores: Conducción y GRUA / RETROPALA
 - Artillería: SIMACA y Entrenador Bía. SIMACA
 - Misiles: Aula COAAAS-L/M, SAARA, EUMP MILAN DX-143, TOW PGTS...
 - UAV (Unmanned Air Vehicle): UAV Pasi
- Serious Games: Se trata de juegos comerciales adaptados a las necesidades del Ejército. Crean un entorno virtual en 3D de dimensiones variables. Los usuarios interactúan en el empleando ordenadores asumiendo un rol dentro de una unidad, representados por un soldado en 3D. Los empleados en la actualidad son el Virtual Battlespace 3 y el Steel Beast Profesional. Ambos enfrentan al jefe de una pequeña unidad a situaciones de combate en las que debe tomar decisiones en tiempo real y comprobar los resultados de las mismas. [6]
- Simuladores constructivos: Permiten el adiestramiento de unidades proporcionando a los enfrentamientos derivados de las decisiones tomadas en los distintos escalones de mando en la ejecución de la maniobra. Muestran a unidades simuladas sobre teatros de operaciones simulados. [6]



Figura 2-2 Simulador en vivo DUELO-Individual. [6]

Es interesante ver cómo el Ejército de Tierra establece una jerarquía en el modo en que emplea la simulación, relacionando el tipo de unidad con el simulador a emplear en cada caso. Puede ocurrir que, para una situación específica, más de un simulador sea adecuado con lo que el límite entre de uso entre ellos no es estricto. Como norma general, para unidades pequeñas se emplean simuladores de apoyo a la instrucción, mientras que para unidades de mayor entidad se emplean simuladores de apoyo al adiestramiento y evaluación de dichas unidades. [6]



Figura 2-3 Pirámide de empleo de simuladores en el ET. [6]

2.2.1.2 Ejército del Aire

La situación del Ejército de Aire es diferente ya que dispone de una lista muy variada de simuladores adquiridos en diferentes momentos, asociados cada uno al sistema de armas que pretenden simular. En la actualidad tiene en servicio, entre otros, los siguientes: simuladores de F18, C-101 y C-130, simuladores para diferentes tipos de helicópteros, simuladores de defensa aérea, guerra EW y apoyo aéreo e incluso un simulador de paracaidismo. [6]

A corto plazo, el mayor problema al que se enfrentan los simuladores del Ejército del Aire es la interoperabilidad. Se pretende poder conectar simuladores iguales para poder realizar el adiestramiento de misiones en un nivel táctico a través de una red WAN. [6]



Figura 2-4 Simulador de vuelo del F-18 en la base de Torrejón. [7]

2.2.1.3 Simuladores en la Armada

En la Armada la simulación ha cobrado especial relevancia en el ámbito de la instrucción y el adiestramiento debido al elevado coste del uso de material real para este fin. Mediante la simulación se complementa, perfecciona y, en ciertas ocasiones, posibilita el adiestramiento de las unidades. De acuerdo a su capacidad, los simuladores en la Armada se clasifican del siguiente modo: [6]

- Navegación y táctica:
 - SIMNAV: Adiestramiento de navegación y maniobra de buques, así como el empleo de los equipos de navegación presentes en los puentes de gobierno. Permite variar entre buques y escenarios muy diversos.
 - GALEÓN: Simulador táctico de operaciones navales para cualquier tipo de plataforma.
 - SCOMBA-T: Adiestramiento en los sistemas de combate a través de un PC.
 - SCE: Adiestramiento en el control de los desembarcos anfibios, así como el planeamiento de la maniobra y el control de las comunicaciones.
 - SACAZ: Simulador del sistema de combate para los Cazaminas de clase Segura
 - SATS (S-70): Adiestramiento específico para el uso de los sensores y sistemas de combate de submarinos.
 - TALOS: Adiestramiento en el mando y control de las unidades de Infantería de Marina. Permite integrar la coordinación y ejecución del apoyo de fuegos con la maniobra.
- Control de Plataforma:
 - SISMA (S-70): Adiestramiento de seguridad en la inmersión y simulación de la cámara central del submarino.
 - SIMPRA (S-70): Simulación de propulsión para submarinos.
 - SIFAS (S-70): Adiestramiento para seguridad en la inmersión y adiestramiento en fugas y vías de agua.
 - Simulador de control de plataforma fragatas clase Santa María.
 - Simulador de control de plataforma SICP Patiño y LPD clase Galicia.
 - Simulador SICP fragatas clase Álvaro de Bazán.
 - Simulador SICP COMPLEX.
 - Simulador de vuelo SOTV-EAV8B Plus.
 - Simulador de vuelo SOTV-ESH60B LAMPS.
- Seguridad Interior:
 - CISI: Adiestramiento contraincendios sin contaminación e inundaciones.
 - CASI: Adiestramiento contraincendios, de derrames, de aeronaves.
- Adiestramiento del combatiente:
 - Virtual Battlespace 3: Simulador comercial con gran capacidad de modelado de escenarios y situaciones tácticas que permite la interconexión entre usuarios para compartir la simulación de un mismo entorno virtual.
 - VICTRIX: Simulador de tiro de armas portátiles.
 - SCATT: Simulador de tiro de armas portátiles.
- Comunicaciones:
 - DIANA: Simula el sistema de distribución de datos de navegación en las F-100, LHD, BAC y BAM.
 - GMDSS: Permite simular las comunicaciones del GMDSS.
 - Simulador Sonar AN/SQS-56: Permite el adiestramiento de los futuros operadores y mantenedores del sistema real.
- Simuladores de inmersión:
 - Simulador de inmersión a gran profundidad con aire hasta 50 metros.
 - Simulador de inmersión a gran profundidad con mezclas respirables hasta 90 metros.

- Apoyo a la decisión y experimentación:
 - SIMDIS: Reproduce el lanzamiento de misiles para analizar la reacción de los buques y el comportamiento del misil.
 - ALSA: Analiza el lanzamiento de señuelos, empleado para el estudio del posicionamiento de los morteros lanzadores en los futuros buques.
 - EXODUS: Realiza estudios de rutas de evacuación en buques de nueva construcción.

2.2.2 Simulación del tiro

Como hemos podido comprobar en el anterior subapartado, tanto el Ejército de Tierra, el Ejército del Aire y la Armada, están apostando por un uso intensivo de los simuladores para cubrir todos los aspectos de la instrucción y adiestramiento de sus militares.

En lo que respecta al adiestramiento del combatiente, estos deben cubrir desde la instrucción más básica del combatiente, como es el empleo de las armas individuales y el tiro, para lo cual se ha optado por el simulador de tiro para armas portátiles VICTRIX, hasta el adiestramiento simulado de grandes unidades en diferentes escenarios táctico, para el cual ha dado resultados muy positivos el simulador Virtual Battlespace 3. [8]

2.2.2.1 VBS3

El VBS3 (Virtual Battlespace 3) es un producto de Bohemia Interactive Simulations empleado por más de 50 países a nivel OTAN, y más de 500.000 combatientes cada año. El US Army lo ha acreditado para el entrenamiento y la simulación de más de 100 tipos de situaciones tácticas para lo cual emplea un repositorio con miles de modelos 3D, soporta terrenos y escenarios masivos con una calidad gráfica extraordinaria, permite la conectividad entre múltiples jugadores y es altamente inmersivo. [9]



Figura 2-5 Funcionalidades del Simulador Virtual Battlespace 3. [9]

Este simulador permite editar misiones, crear terrenos según una cartografía o inteligencia específicas, crear unidades y vehículos... además, permite la integración con otros programas y simuladores en uso en la actualidad como el TALOS TÉCNICO, gracias al lenguaje HLA. Además, el simulador puede contener unidades enemigas y neutrales controladas por la IA (Inteligencia Artificial), lo que permite adiestrar a todo tipo de unidades con unas características y unas condiciones muy difíciles de establecer en la realidad. [10]

En la actualidad el Ejército de Tierra lo ha implantado en la mayoría de sus Brigadas mediante la contratación de licencias y la creación de aulas de simulación. Además, el Ejército de Tierra trata de integrar el VBS3 con otros simuladores como el STEEL BEASTS y el SAETA. [10] Infantería de Marina, a su vez, cuenta con simuladores VBS3 en la EIMGAF, TERNOR, USCAN y AGRUMAD. [6]

2.2.2.2 VICTRIX

De entre los proyectos desarrollados por Indra, el llamado VICTRIX es un simulador diseñado para mejorar y apoyar el entrenamiento de tiro de los combatientes mediante la reproducción de un campo de tiro virtual que permite la configuración de diversos parámetros. Este simulador permite generar ejercicios de tiro particulares, así como realizar cualquiera de las prácticas habituales que se realizan en las Fuerzas Armadas. Además, VICTRIX permite realizar entrenamientos a nivel táctico de manera individual y grupal (hasta nivel Pelotón) dentro de entornos 3D abiertos. [11]

Las armas empleadas en este simulador son modificadas de manera que el sistema funcione de manera totalmente independiente, empleando aire comprimido para simular el retroceso real de las armas. El abanico de armas que soporta el sistema es muy amplio y abarca desde la Glock 19, posible sustituto de la actual FN P9 empleada por la Infantería de Marina, hasta el fusil de asalto HK G36E o múltiples ametralladoras ligeras. [11]

El simulador está preparado para facilitar la labor del instructor, pudiendo variar los parámetros y condiciones dentro de los ejercicios de galería de tiro habituales, además de permitirle adaptar los ejercicios tácticos a las necesidades específicas de la unidad con el generador de ejercicios. Finalmente, VICTRIX permite realizar un seguimiento detallado de los ejercicios y proporciona resultados para su posterior análisis a la finalización de los mismos. [11]

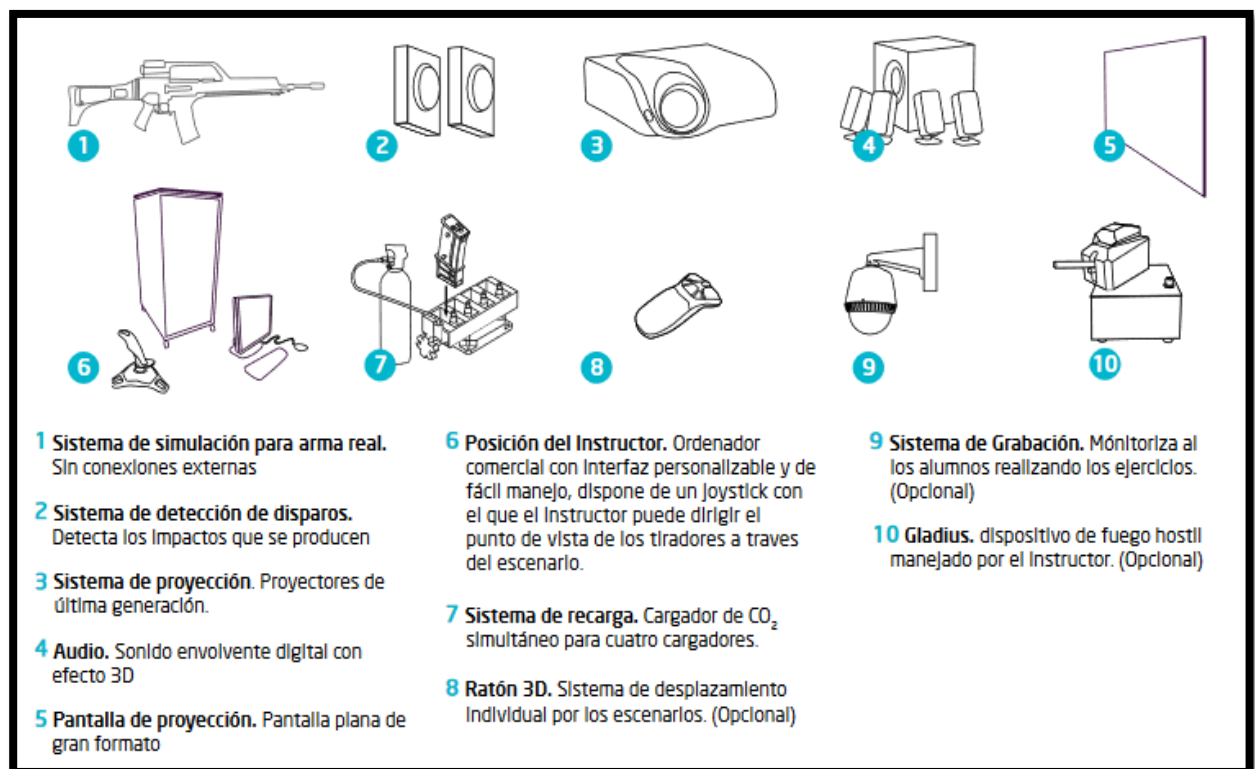


Figura 2-6 Componentes del simulador VICTRIX. [11]

Entre otras, las principales ventajas del empleo de este simulador son la seguridad, evitando el uso de munición real; la reducción de costes, de mantenimiento, munición y logísticos; la evaluación detallada y objetiva de los ejercicios; y la facilidad de adaptación y modificación del simulador sin necesidad de poseer grandes conocimientos informáticos. [11]

2.3 Tecnologías i4.0

Como ya se expuso en la introducción de la presente memoria, nos encontramos ante la Cuarta Revolución Industrial con la cual ha surgido el concepto de Industria 4.0 la cual se sustenta en la combinación de técnicas avanzadas de producción y el uso de tecnologías inteligentes.

El empleo de las nuevas tecnologías permite el desarrollo de productos y servicios completamente nuevos e innovadores. Este es el caso de la Simulación. En este apartado se explicarán los conceptos y las tecnologías relacionadas con la simulación y que, por tanto, serán de utilidad en el desarrollo de este proyecto. Nos centraremos en las siguientes: Inteligencia Artificial y, Realidad Virtual, Aumentada y Mixta de acuerdo a sus similitudes y diferencias.

2.3.1 Inteligencia Artificial

De entre todas las nuevas tecnologías que permiten el desarrollo de la industria 4.0, posiblemente sea la Inteligencia Artificial la más innovadora e importante de todas ellas. Puede ser muy complejo definir de manera exacta en qué consiste la Inteligencia Artificial, pudiendo diferir estas definiciones en función del ámbito o área en el que se aplique esta tecnología. De manera simplificada, la Inteligencia Artificial es la tecnología que permite a las máquinas aprender de la experiencia, ajustando su modo de actuar en base a la misma, actuando por lo tanto como lo hacen los humanos. [12]

La Inteligencia Artificial se hace posible combinando las grandes cantidades de datos almacenadas con sistemas de procesamiento rápido e iterativo y algoritmos “inteligentes”. Esto permite a las máquinas identificar patrones y características en los datos de manera automática y de este modo “aprender”. [12]

En la simulación, la Inteligencia Artificial permite generar personajes con comportamientos muy similares a los que tendría un ser humano en la realidad. De cara al desarrollo de simuladores para el adiestramiento del combatiente, esto permite crear escenarios realistas, permitiendo a los actores de una simulación, sean amigos o enemigos, controlados por la Inteligencia Artificial actuar y tomar decisiones propias de manera realista e independiente.

2.3.2 Realidad Virtual, Aumentada y Mixta

¿Qué es la realidad? Es un concepto definido de infinitud de maneras pero que, en la mayoría de las ocasiones se reduce a lo siguiente: la realidad es aquello que percibimos a través de los sentidos. Pero, ¿qué ocurre si modificamos aquello que percibimos a nuestro antojo y decidimos qué es aquello que queremos ver, oír, sentir...? De este modo nace el concepto de realidad virtual, en el que a través de unos sistemas y equipos se nos presenta una realidad diferente a la que realmente tenemos a nuestro alrededor.

De este concepto de Realidad Virtual surgen otros dos; el concepto de Realidad Aumentada y el de Realidad Mixta. A continuación, vamos a aclarar estos conceptos para entender qué son y cuáles son sus diferencias.

- Realidad Virtual: Esta tecnología es capaz de sustituir el entorno a nuestro alrededor por otro generado de manera digital. A través de un equipo compuesto normalmente por unas gafas y auriculares, el usuario se ve “transportado” a un lugar y un tiempo diferentes, en un mundo completamente virtual. En algunos casos, el equipo permite al usuario interactuar con este mundo virtual empleando controladores o dispositivos adicionales, logrando una experiencia más inmersiva y realista. [13]



Figura 2-7 Experiencia VR Zero Latency. [14]

En la actualidad están teniendo mucho éxito, en el ámbito del ocio, los videojuegos en realidad virtual. Como ejemplo de Realidad Virtual Interactiva, se muestra en la Figura 2-7 el videojuego multijugador en VR “Zero Latency”. En él, varios jugadores se ven inmersos en un entorno virtual en el que pueden interactuar con el propio entorno y entre ellos mismos.

- **Realidad Aumentada:** La realidad aumentada es una tecnología que permite introducir contenido virtual en el mundo físico que nos rodea. La diferencia esencial con la Realidad Virtual es que no cambia por completo lo que percibimos a través de los sentidos si no que se añade información, superponiendo aquello que se desea mostrar encima del entorno real. Además, la Realidad Virtual se limita al escenario o entorno generado mientras que la realidad aumentada permite al usuario emplearla allá donde desee. [13]



Figura 2-8 Aplicación de la AR en Pokémon-Go. [15]

En la Figura 2-8 se muestra el exitoso videojuego “Pokémon-Go”, en el que a través de un dispositivo móvil y tecnologías de Realidad Aumentada se situaban criaturas ficticias en el mundo real, permitiendo al usuario interactuar con ellas.

- Realidad Mixta: Esta tecnología consiste en una combinación “inteligente” de las dos tecnologías expuestas anteriormente. En ella se fusionan el mundo real y el mundo virtual, donde los elementos virtuales emplazados en el mundo real son “conscientes de su entorno y se adaptan a él, de tal forma que se vean afectados igual que los elementos reales. [13]

2.4 Software y equipo

En este subapartado se presentarán tanto el software como el equipo empleado en el desarrollo del proyecto. En primer lugar, se definirán los diferentes programas de software con una breve reseña de cada uno, y se finalizará explicando cuáles fueron y el porqué de los materiales empleados.

2.4.1 Software



Figura 2-9 Programas de Software

Para el desarrollo de los escenarios y las funcionalidades del simulador de tiro se han empleado los siguientes programas de software: Epic Games y Unreal Engine 4.25 para la creación y modelado de los escenarios y Mixamo para obtener el modelo para el personaje principal.

Los diferentes logos de los principales programas de software que se han empleado para el desarrollo de los escenarios del simulador se muestran en la Figura 2-8.

2.4.1.1 Epic Games Store

Epic Games es un programa desarrollado por una empresa estadounidense de desarrollo de videojuegos. En él se presenta y ofrece a jugadores y desarrolladores de videojuegos una amplia gama de aplicaciones, servicios y bibliotecas con actores que se pueden implementar posteriormente en motores de juego (como es Unreal Engine 4) que ofrece la misma plataforma.

En la Figura 2-10 se puede observar la interfaz principal de este programa.

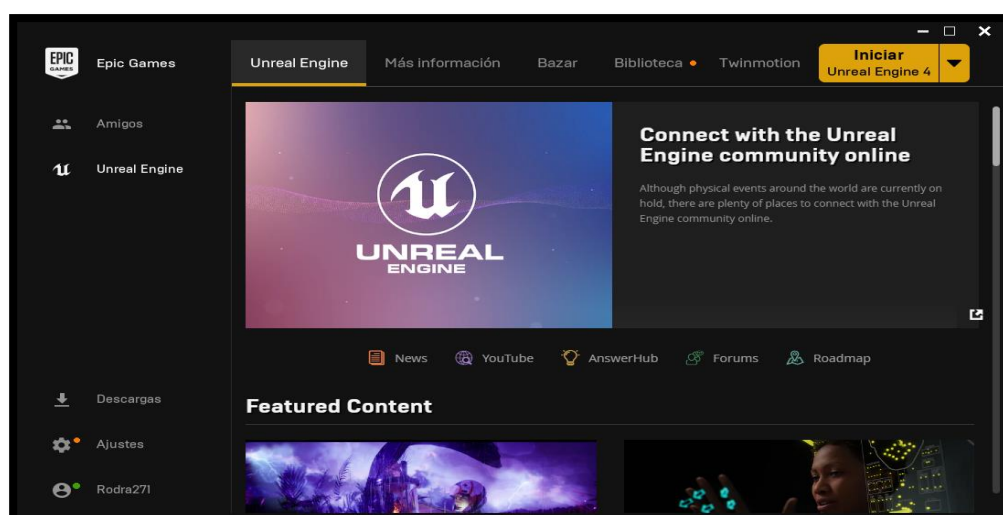


Figura 2-10 Epic Games

2.4.1.2 Unreal Engine 4

Como se ha mencionado con anterioridad, Unreal Engine es un programa de software creado por la empresa Epic Games y accesible a través de la Epic Games Store. Se trata de un motor de juego que contiene las herramientas necesarias para el desarrollo de juegos de alta calidad y visualización de diseño, como son el editor de video, estudio de sonido, el código o la renderización de animaciones.

Para el desarrollo de videojuegos, Unreal Engine 4 permite la creación de proyectos en diversas modalidades, incluyendo escenarios en 2D, en primera y tercera persona y hasta escenarios e Realidad Virtual.

Para ello, Unreal Engine permite al desarrollador escoger entre dos tipos de programación: El clásico lenguaje de programación multiparadigma C++ o el código Blueprint que se trata de un lenguaje de programación multinodal.

Para el presente proyecto se ha empleado la versión de Unreal Engine 4.25.4. Es la última actualización lanzada por Epic Games y se ha obtenido a través de la Epic Games Store.

En la Figura 2-11 se muestra el proceso de creación de un proyecto mediante Unreal Engine 4.

2.4.1.3 Mixamo

Mixamo es un programa de software desarrollado inicialmente por la Universidad de Stanford y adquirido posteriormente por Adobe Systems. La finalidad de este software es el desarrollo y animación de personajes 3D a partir de modelos de los mismos.

La web de Mixamo dispone de una biblioteca que permite la descarga gratuita de estos modelos junto con diversas secuencias de animaciones que permiten dotar a los personajes de un mayor realismo a la hora de implementarlos en un videojuego. Para el diseño del personaje que representará al usuario dentro del simulador de tiro se empleará uno de estos modelos 3D junto con sus animaciones.

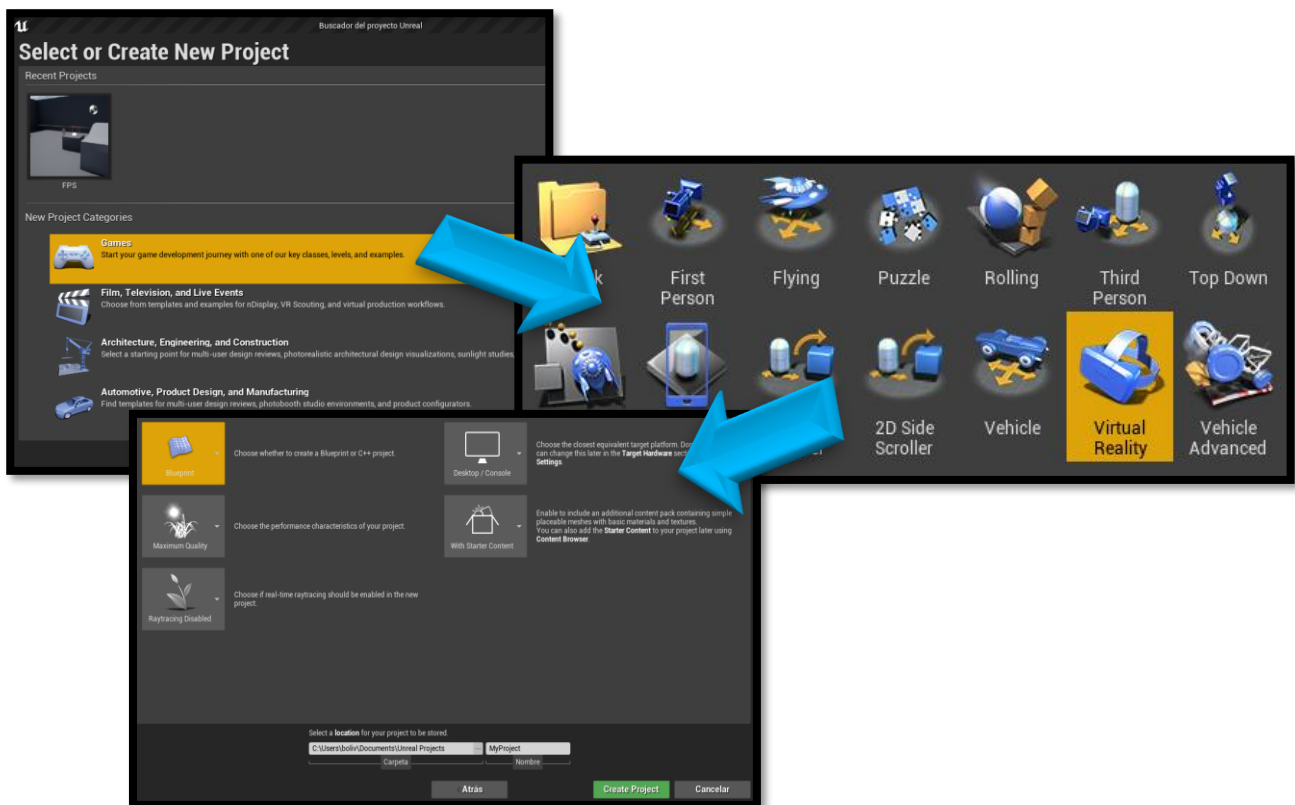


Figura 2-11 Creación de un proyecto en Unreal Engine 4

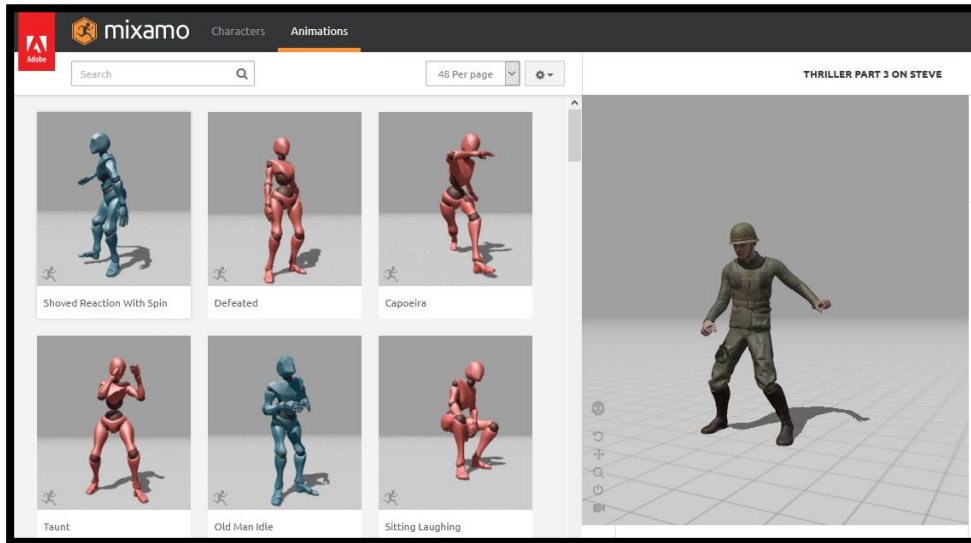


Figura 2-12 Web de Mixamo.

En la Figura 2-12 se muestra la web de Mixamo en la que se muestra la selección de uno de los personajes de la biblioteca online junto con una serie de animaciones que se pueden implementar en dicho personaje. Desde aquí se pueden descargar tanto las animaciones como el modelo 3D del personaje, para posteriormente importarlos en el motor de juego y determinar el comportamiento del personaje en el proyecto creado.

2.4.2 Equipo y Material empleados

Para el desarrollo del proyecto se ha empleado equipo y material con las características y requerimientos necesarios para el diseño, modelado y programación del simulador de tiro.

La mayor parte del trabajo tuvo lugar en el Laboratorio de Medidas dentro de los Laboratorios de Investigación del CUD, en las instalaciones de la Escuela Naval Militar. En él se contó con una CPU con las prestaciones necesarias para el trabajo con Unreal Engine 4, un monitor, ratón y teclado. Cabe destacar que, una vez finalizado, el simulador de tiro con todos sus escenarios y funcionalidades ocupa 39,9GB, incluyendo 6255 archivos distribuidos en 751 carpetas.



Figura 2-13 Laboratorio de medidas

Con el fin de crear y visualizar el entorno en Realidad Virtual para el simulador de tiro, se emplearon un dispositivo MR Lenovo Explorer de Realidad Virtual, una tarjeta gráfica RV Ready instalada en la CPU y un dispositivo Bluetooth para permitir la conexión de los Motion Controllers del MR Lenovo Explorer.

2.4.2.1 *Lenovo Explorer*

El dispositivo empleado para la visualización de entornos virtuales es el Lenovo Explorer, el primero de los dispositivos de Realidad Virtual de la marca Lenovo. Este dispositivo emplea la tecnología puntera de Windows Mixed Reality y consta de un casco ajustable con unas gafas MR ancladas a su parte frontal. Para su correcto funcionamiento el dispositivo cuenta con dos cámaras de seguimiento de movimiento, un sensor de proximidad, un giroscopio, un acelerómetro y un magnetómetro.

Una vez dentro de un escenario virtual, las gafas presentan al usuario la misma en dos pantallas de visualización de 2,89 pulgadas, con un FOV o ángulo de visión de 110 grados. Además, el dispositivo permite la conexión de dos controladores de movimiento o “Motion Controllers” capaces de comunicarse con las gafas con el fin de interactuar con el entorno de realidad virtual o realidad mixta.

El motor de juego empleado para el desarrollo del proyecto permite, gracias a sus últimas actualizaciones, el empleo de los dispositivos de Realidad Virtual de Windows, facilitando al desarrollador la posibilidad de crear un entorno compatible con el dispositivo. Para su conexión es necesaria una conexión de video (Unreal Engine – Lenovo Explorer) además de una conexión mediante un USB3.0 (Lenovo Explorer – PC) por cable. Esto último es una desventaja ya que limita el radio de acción de las gafas a la longitud de dicho cable.



Figura 2-14 Dispositivo Lenovo Explorer

En la Figura 2-14 se pueden observar los diferentes componentes del dispositivo Lenovo Explorer.

2.4.2.2 Tarjeta Gráfica NVIDIA RTX 2080 SUPER

Debido a la alta demanda de recursos del motor de juego y a implementación de la tecnología de Realidad Virtual, se ha adquirido e instalado en el equipo una tarjeta gráfica VR Ready, en concreto la NVIDIA RTX 2080 SUPER, que cuenta con 8GB GDDR6 (Graphics Double Data Rate 6) de capacidad de memoria de video, además de tres puertos Display y un puerto HDMI 2.0b.

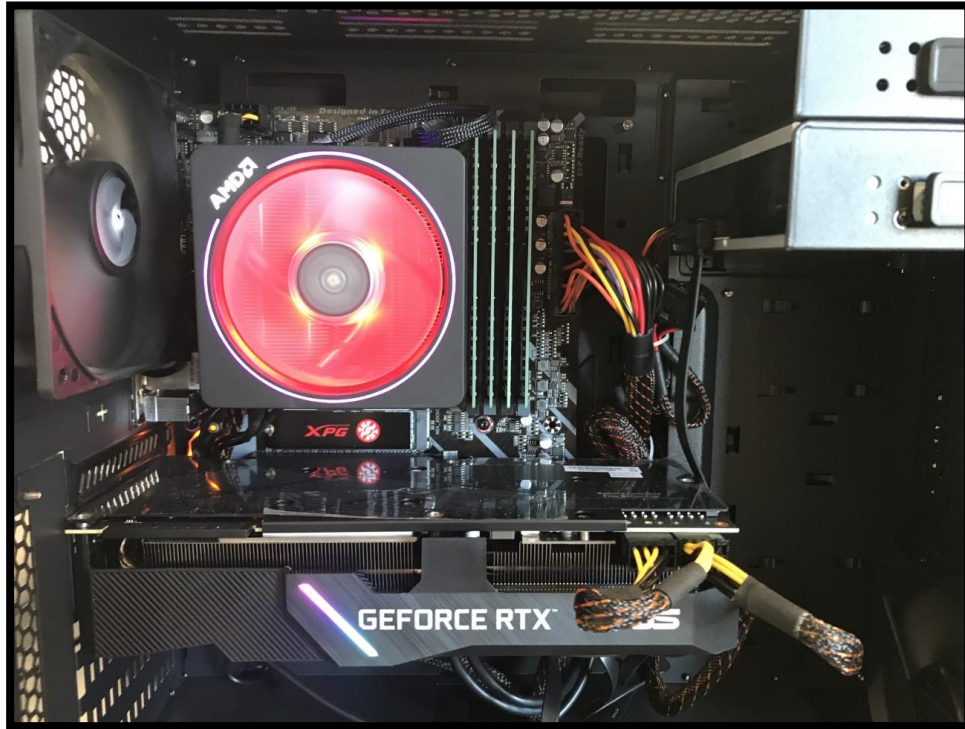


Figura 2-15 Tarjeta gráfica instalada

En la Figura 2-15 se muestra la tarjeta gráfica instalada en el interior del equipo y lista para su uso con Unreal Engine 4 y el dispositivo Lenovo Explorer.

3 DESARROLLO

3.1 Descripción del apartado

En este apartado se procede a explicar el desarrollo del presente proyecto en el que se ha creado un escenario 3D virtual apto para adiestramiento inmersivo de tiro con arma de fuego. Para ello se ha aplicado tecnología i4.0 a este escenario, creando un entorno de Realidad Virtual que facilite al usuario una experiencia lo más realista posible.

Para lograr esto último, el punto de partida consistirá en la elección de una localización real que servirá de referencia para nuestro escenario. Una vez seleccionada, se procederá al desarrollo y modelado del escenario y sus componentes, dotándolo de una apariencia realista y reconocible para el usuario y, a continuación, se programará el código necesario para que el escenario pueda cumplir con nuestras necesidades. Finalmente se creará el entorno de Realidad Virtual que permita la inmersión e interacción con dicho escenario.



Figura 3-1 Etapas del trabajo

3.2 Escenario y aplicación

3.2.1 Aplicación del simulador

Antes de seleccionar el escenario sobre el que se diseñará el simulador de tiro, es conveniente establecer cuál será la aplicación del simulador fruto del presente proyecto. Como vimos anteriormente, existe en la actualidad una gran variedad de simuladores para multitud de ámbitos, generales o

específicos, dentro de los Ejércitos y la Armada. Este simulador, como indica el título del proyecto, servirá para el adiestramiento de tiro con arma de fuego.

Dado que el concepto de arma de fuego puede abarcar muchos tipos de armamento diferentes, acotaremos en primer lugar las armas de fuego para las que estará enfocado el simulador. Un arma de fuego, o arma pirobalística, es toda aquella que impulsa un proyectil sobre un objetivo, con mayor o



Figura 3-2 Clasificación de las Armas de Fuego

menor precisión, aprovechando la deflagración de la pólvora. Estas se pueden dividir en pesadas, caracterizadas principalmente por la artillería, y portátiles. A su vez, las armas portátiles se clasifican en armas de uso colectivo y armas individuales.

El presente proyecto estará enfocado al adiestramiento del combatiente para la instrucción en el tiro sistemas de armas individuales, y dentro de estas, nos centraremos en las armas largas. Tanto el Ejército de Tierra como la Infantería de Marina dentro de la Armada dan gran importancia, en todas sus escalas, al adiestramiento individual de todo combatiente en el tiro. Tanto es así, que está reflejado en los diferentes currículos, y planes de estudios de los diferentes cuerpos y escalas como se muestra a continuación:

- En la Guía Docente para el presente curso del Centro de Formación de Tropa N°1 para Enseñanza de Formación para el acceso a la escala de tropa del Cuerpo General del Ejército de Tierra, en su módulo de Instrucción y Adiestramiento I, establece: [16]
 - Competencia General 5: Dominar las habilidades y destrezas de la instrucción individual del Ejército de Tierra, así como el manejo de las armas, municiones, equipos y demás medios materiales de dotación individual reglamentaria que se determinen. [16]
 - Unidad de Competencia 5.6: Adquirir los conocimientos básicos de la teoría elemental del tiro y las habilidades prácticas necesarias para efectuar eficazmente el tiro con las armas y municiones de dotación individual. [16]
 - Resultado de Aprendizaje 5.6.4: Practicar correctamente las acciones que componen la secuencia del tiro. [16]
 - Resultado de Aprendizaje 5.6.6: Superar los ejercicios de tiro de FUSA que se determinan para la Fase, de acuerdo a lo establecido en el Manual de Instrucción de Tiro de Armas individuales en vigor. [16]
 - Resultado de Aprendizaje 5.6.7: Aplicar el empleo táctico del FUSA, adquiriendo objetivos y batiéndolos oportuna y eficazmente. [16]
- En el Currículo de la enseñanza de formación para el acceso a las escalas de marinería y de tropa del Cuerpo General y del Cuerpo de Infantería de Marina de la Armada, aprobado en el BOE N.º 188, de 7 de agosto de 2019, se establece: [17]

- Competencia General 5: Alcanzar una adecuada formación naval-militar y dominar las habilidades y destrezas de la instrucción básica de combate de la Armada, así como el manejo de armas, municiones, equipos y demás medios de dotación individual reglamentaria que se determinen. [17]
- Competencias específicas de la Especialidad fundamental de Infantería de Marina:
 - Competencia específica 1: Dominar el manejo de las armas, municiones, equipos de comunicaciones, de optrónica y resto de equipo individual reglamentario asignado, responsabilizándose de su correcto funcionamiento y mantenimiento y ejecutando con seguridad y precisión acciones de tiro con fuego real. [17]
- En el Currículo de la enseñanza de formación de oficiales para la integración en la escala de oficiales del Cuerpo General de la Armada y del Cuerpo de Infantería de Marina mediante las formas de ingreso sin titulación previa en la Escuela Naval Militar, se establece: [18]
 - Competencia General 15: Manejar con destreza el equipo de la dotación individual reglamentaria en la Armada, en especial el arma reglamentaria, así como adquirir la habilidad y destreza necesaria en la instrucción individual y de combate. [18]
 - Competencias específicas de la Especialidad fundamental de Infantería de Marina:
 - Competencia Específica 4: Conocer y saber emplear los sistemas de armas individuales, colectivos medios y de Apoyo de Fuego de Infantería de Marina para ejercer sus funciones como mando de las pequeñas unidades de la Fuerza de Infantería de Marina. [18]

Debido a que el adiestramiento para el tiro está limitado por una serie de factores como el tiempo, la infraestructura y el coste material y logístico entre otros, este simulador tratará de servir como complemento o incluso alternativa para permitir que se alcancen las competencias anteriormente expuestas, necesarias para finalización de los diferentes cursos de formación de manera satisfactoria.

3.2.2 Selección del entorno

Con el objetivo de recrear un entorno lo más realista posible y el cual el usuario pueda reconocer, se seleccionará un emplazamiento real, del cual se hará un modelado 3D y servirá como base para la creación de los diferentes niveles del simulador de tiro.



Figura 3-3 Galería de tiro de Figueirido

De este modo, se representará de manera aproximada la galería de tiro del Cuartel General Morillo de la Brigada Galicia VII situada en Figueirido, en la provincia de Pontevedra. Con el fin de determinar las dimensiones de dicha edificación, se realizarán medidas sobre la misma en la Sede Electrónica del Catastro dadas las actuales restricciones de movilidad y acceso. La reconstrucción del interior de la galería será lo más fidedigna posible siempre que no perjudique el rendimiento y funcionamiento del simulador.

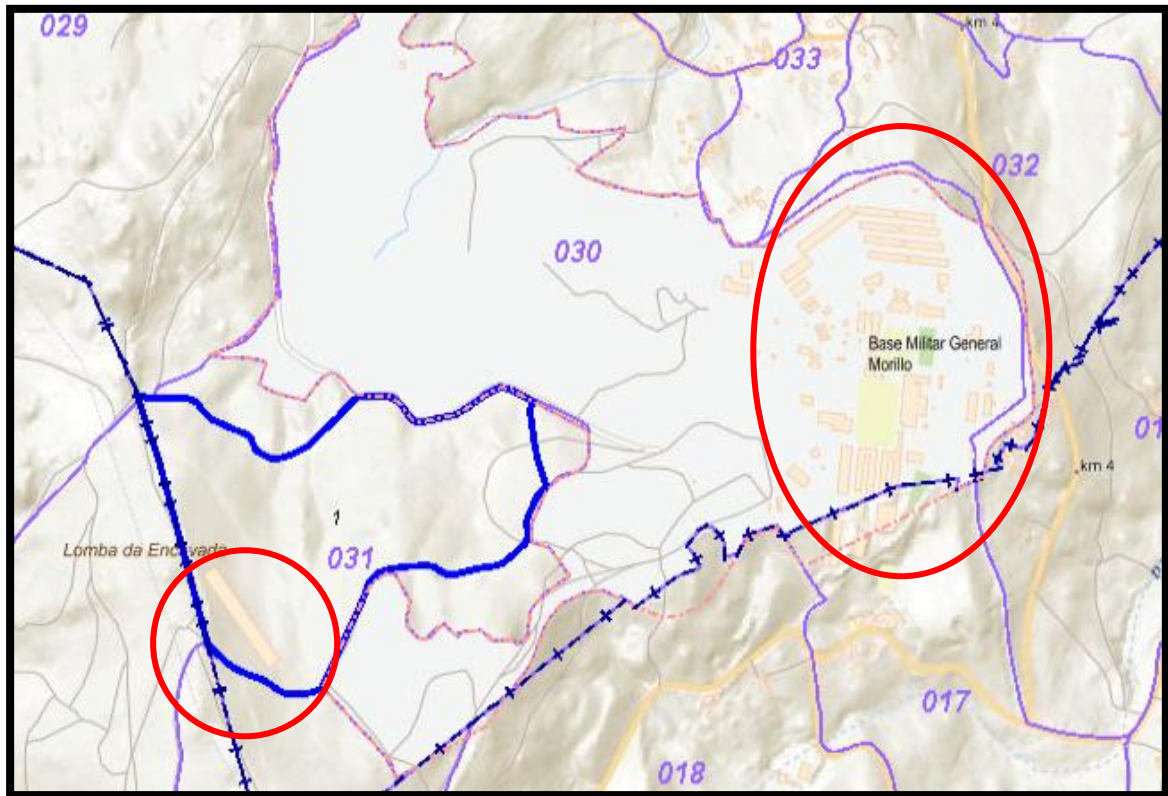


Figura 3-4 Situación de la Galería de Tiro de la Base Militar General Morillo.

3.2.2.1 Ejercicios de tiro

En la actualidad existen diferentes programas de ejercicios enfocados al adiestramiento individual del combatiente en el tiro. Entre ellos se dan diferentes modalidades de tiro, enfocadas a las diferentes situaciones que el combatiente se encontrará en el combate real, como son el tiro de fusil básico, el tiro de precisión, el tiro de combate o el tiro nocturno.

Con el fin de desarrollar un simulador de tiro 3D en Realidad Virtual, capaz de reproducir los ejercicios necesarios para alcanzar el nivel de adiestramiento requerido en estos aspectos, se seleccionará un programa de tiro vigente y en uso en la actualidad. La viabilidad del simulador de tiro desarrollado en este proyecto y la posibilidad de emplearlo como complemento o alternativa se basará en la capacidad de reproducir con la mayor eficiencia posible todos los ejercicios reflejados en dicho programa de tiro.

En junio de 2019 se implantó, para todos los miembros de la Brigada de Infantería de Marina, un programa de tiro de fusil que, una vez finalizado con éxito, cualifica a sus miembros en el empleo de este sistema de armas individual.



Figura 3-5 Portada del programa de Tiro de Fusil de la Brigada de Infantería de Marina.

Este programa de tiro se divide en las siguientes modalidades:

- Tiro de fusil básico (BP).
- Tiro de fusil básico de combate (BC).
- Tiro de fusil avanzado de combate diurno (ACF).
- Tiro de fusil avanzado de combate nocturno (ACFN).

Cada modalidad a su vez se divide en un conjunto de ejercicios de los cuales se especifica el desarrollo, el tipo, la distancia y la disposición del blanco o blancos, la munición a emplear, la posición del tirador y los requisitos para superar de manera satisfactoria dicho ejercicio. De cara al diseño y modelado de los escenarios, nos fijaremos en los tipos de blanco empleados y su disposición, ya que en un mismo escenario se podría llevar a cabo más de un tipo de ejercicio, lo que simplifica y disminuye la cantidad de escenarios necesarios para realizar el programa de tiro en su totalidad. La secuencia, la munición disponible y la distancia para realizar el ejercicio serán variables que se podrán modificar dentro de los propios escenarios, así como los requisitos a alcanzar para superar el mismo.

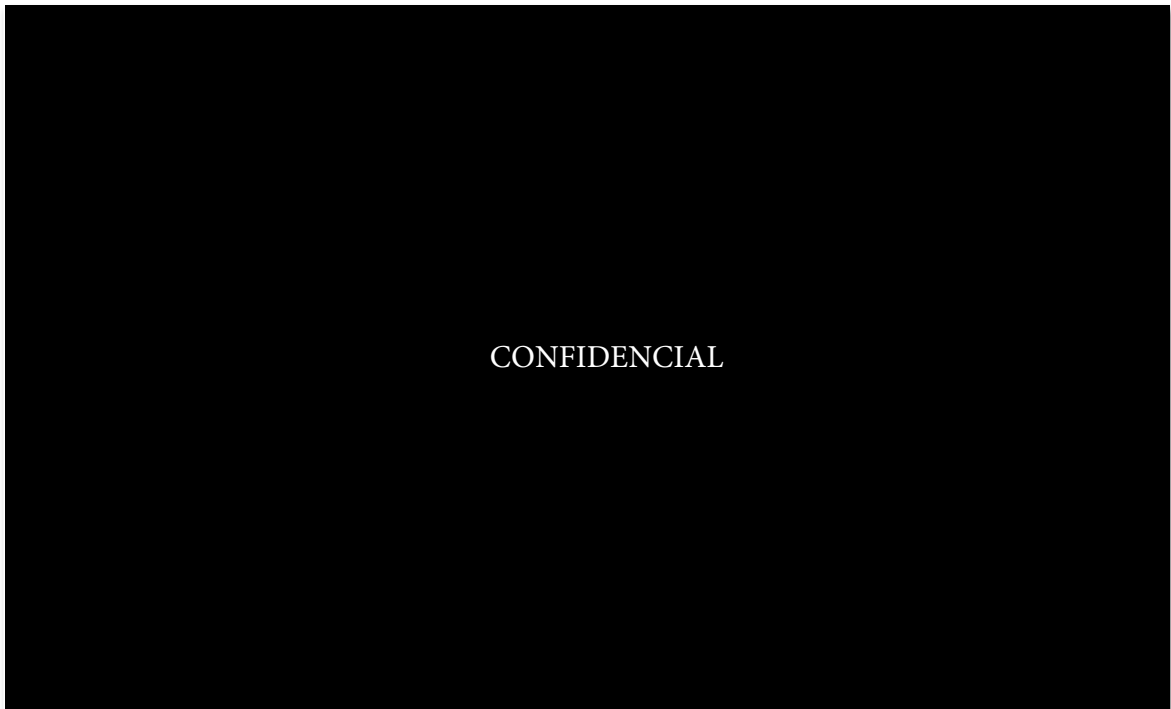


Figura 3-6 Ejemplo de modalidad de tiro con sus correspondientes ejercicios.

De este modo se crearán los siguientes escenarios, los cuales se asignarán a una serie de ejercicios concretos del programa de tiro (se muestran los códigos de cada ejercicio):

- Escenario 0: Nivel base
- Escenario 1: Blanco único
 - Tiro de Fusil Básico: BP1-BP7.
 - Tiro de Fusil Básico de Combate: BC1-BC7.
 - Tiro de Fusil Avanzado de Combate Diurno: ACF3, ACF5 y ACF6.
- Escenario 2: Blanco doble lateral
 - Tiro de Fusil Avanzado de Combate Diurno: ACF1 y ACF4.
- Escenario 3: Blanco doble en profundidad
 - Tiro de Fusil Avanzado de Combate Diurno: ACF2, ACF7 y ACF 8.
- Escenario 4: Blanco móvil
 - Tiro de Fusil Avanzado de Combate Diurno: ACF9
- Escenario 4: Modo nocturno
 - Tiro de Fusil Avanzado de Combate Nocturno: ACFN1-ACFN8

En el Anexo I: Ejercicios de tiro de fusil de la brigada de infantería de marina se muestran todos los ejercicios contenidos dentro del programa de tiro de fusil de la Brigada de Infantería de Marina.

El escenario “Modo nocturno” incluirá todos los tipos de blancos ya que todos ellos se emplean en esta modalidad. Además, de manera adicional, se creará otro escenario basado en los anteriores, pero de dimensiones mayores, para el adiestramiento en el tiro de precisión. En este último, el tiro no se verá influido únicamente por las acciones del propio usuario, sino que también intervendrán factores externos como las precipitaciones o la velocidad del viento sobre el proyectil.

3.3 Diseño y modelado

Con el fin de que el simulador de tiro funcione correctamente es necesario un diseño del escenario adecuado y detallado, atendiendo a todos los componentes que conforman el mismo. Solo de este modo se logrará una experiencia similar a la que el usuario se puede encontrar en la realidad. La programación

(ver Programación y código) será la que permita que todas el escenario diseñado y modelado en este apartado conjuguen con las funcionalidades necesarias para el correcto funcionamiento del simulador.

En este apartado se tratará de desarrollar el diseño y modelado del simulador, tras una introducción a las nociones básicas necesarias para el empleo del motor de juego empleado, desglosando el mismo en los diferentes escenarios, niveles y componentes que forman parte esencial del simulador.

3.3.1 Interfaz Unreal Engine 4

Al abrir el motor de juego para el desarrollo de cualquier proyecto, Unreal Engine 4 nos muestra una interfaz con una serie de herramientas, servicios y funcionalidades que permiten el diseño y modelado de los escenarios.



Figura 3-7 Interfaz de Unreal Engine 4.

La Figura 3-7 muestra la interfaz del motor de juego en la que aparece en la ventana central el escenario en desarrollo. A su alrededor se observan las diferentes herramientas que ofrece el motor, disponibles para su uso e implementación directa sobre la propia escena de manera cómoda y sencilla. Además, la disposición de dichas herramientas puede ser modificada, pudiendo situar las mismas en el lugar más más adecuado y cómodo para cada usuario. Esto hace de Unreal Engine un motor de trabajo muy visual y amigable para el desarrollador. A continuación, se explicarán las diferentes herramientas y funcionalidades, señaladas mediante un código de colores en la imagen anterior.

En primer lugar, y como podemos observar en la Figura 3-8, se muestra en la parte inferior de la interfaz el Buscador de Contenido. Este permite al usuario crear, importar, organizar, visualizar y modificar todos los contenidos incluidos en el editor. Además, permite su distribución y organización en carpetas, además de modificar los activos cambiando su nombre y ubicación, copiarlos o duplicarlos y visualizar sus referencias. Por último, en su parte superior dispone de un buscador que permite la localización de cualquier activo dentro del contenido del editor de manera rápida y sencilla.



Figura 3-8 UE4 - Buscador de contenidos.

A continuación, nos encontramos con otro importante panel en la parte superior derecha, el World Outliner.

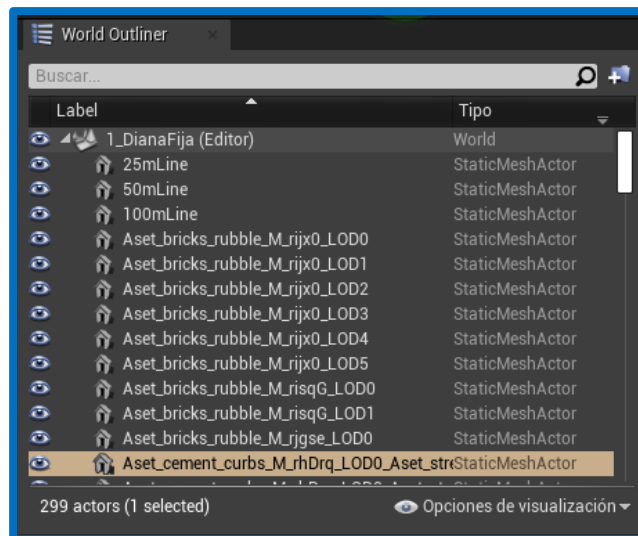


Figura 3-9 UE4 - World Outliner.

En dicho panel, como se puede observar en la Figura 3-9, se muestran todos los activos que se hayan incluido en el escenario de manera jerárquica. Estos actores pueden ser seleccionados y modificados directamente desde el panel.

El tercer panel que podemos observar es el Gestor de Detalles. En la Figura 3-10 se muestra dicho panel, el cual permite modificar y editar las propiedades de los elementos seleccionados dentro del escenario, desde trasladarlos, rotarlos o escalarlos, hasta seleccionar los materiales que lo componen, así como cualquier otra funcionalidad que posea el tipo de activo seleccionado.

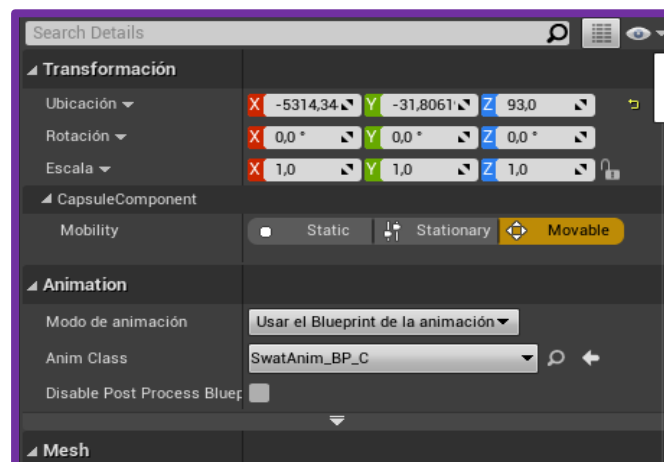


Figura 3-10 UE4 - Gestor de Detalles.

Encima del Gestor de Detalles, y relacionado con él, se encuentra el Buscador de Componentes que, como podemos apreciar en la Figura 3-11, nos permite visualizar de manera jerárquica todos los componentes que forman parte del activo seleccionado. Esto nos permite editar en el Gestor de Detalles uno de los componentes en concreto, además de añadir otros nuevos o abrir la ventana del Blueprint que le corresponde.

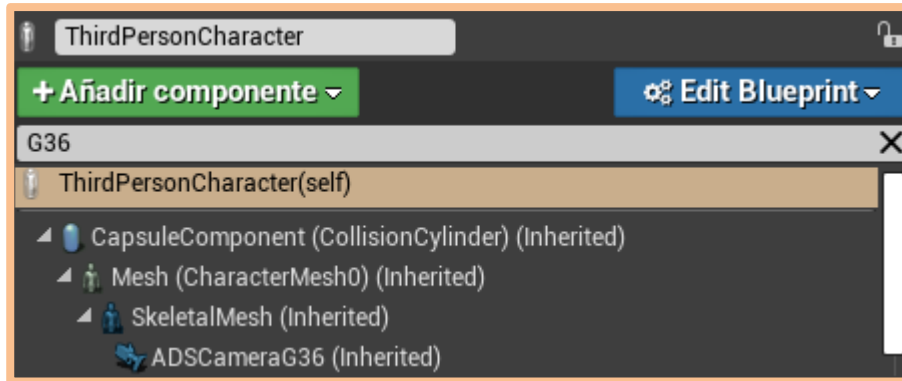


Figura 3-11 UE4 - Buscador de Componentes.

Dentro de la ventana central que muestra el escenario, aparecen en la parte superior una serie de Widgets que le dan al motor una serie de funcionalidades que permiten al usuario editar, mover, rotar o escalar los objetos de manera manual, rápida y sencillamente, así como ajustar la velocidad de la cámara del editor en la escena. Todas estas funcionalidades son las que se muestran en la Figura 3-12.



Figura 3-12 UE4 - Widgets de edición.

Por último, en la parte superior se encuentra la Barra de Herramientas que se muestran en la Figura 3-13 UE4 - Barra de Herramientas. Cada una de ellas tiene una funcionalidad diferente, de entre las que destacan las siguientes: Save Current (Guardar) permite guardar todos los cambios realizados sobre el proyecto hasta el momento; Blueprints permite abrir la ventana de edición del Blueprint propio del escenario, buscar y seleccionar uno existente o crear uno nuevo; Jugar nos da la posibilidad de probar en tiempo real la simulación, pudiendo así comprobar que todo funciona correctamente; y Modes, la cual nos permite seleccionar el modo de trabajo sobre el escenario.

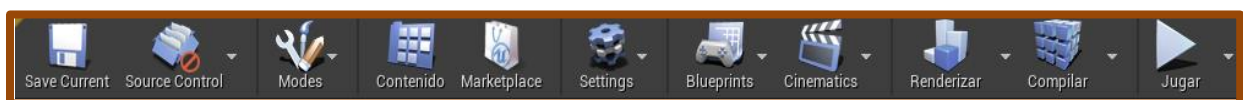


Figura 3-13 UE4 - Barra de Herramientas.

De entre los diferentes modos, el que usaremos con mayor frecuencia será el modo Colocar ya que es el que nos permite seleccionar e incluir nuevos componentes y objetos a nuestro escenario. Estos pueden ser formas básicas, luces, efectos visuales, volúmenes o modelos estáticos. El modo Terreno se emplea para crear nuevos paisajes, incluyendo diferentes componentes (manipular), modificando su forma con diversas herramientas (esculpir) o dándole la apariencia deseada con diferentes texturas (pintar). Además, este modo permite importar mapas de altura creados con fuentes externas. Por último, el modo Follaje sirve para introducir dentro del escenario grandes cantidades de modelos estáticos de manera rápida, dando a nuestro escenario mayor realismo.

3.3.2 Estructura

El simulador de tiro se estructurará en un conjunto de escenarios que permitirán al usuario realizar todos los ejercicios necesarios para superar de manera satisfactoria el programa de tiro seleccionado como base para el desarrollo de este proyecto (Ver Ejercicios de tiro). Las dimensiones del modelo se han obtenido empleando una escala de conversión de unidades de medida reales a unidades ficticias de Unreal Engine. Cada unidad de Unreal Engine se corresponde con 1 centímetro en la realidad. De cara a tener una representación visual de las distancias, se creará un Blueprint de medición que nos facilitará el realizar diversas mediciones dentro del escenario.

La diferencia esencial entre estos escenarios radica en los tipos de blancos que se emplean, su disposición dentro de la galería de tiro. Además, en cada escenario se dispondrá de un tiempo y una munición determinada para llevar a cabo los ejercicios, aparte de incluir instrucciones específicas que el usuario debe llevar a cabo en el transcurso del ejercicio.

Además, se incluirá un escenario preliminar en el que el usuario podrá adaptarse al empleo del simulador. Este tendrá una apariencia irreal, pero en el aparecerán todos los elementos con los que el usuario podrá interactuar en el resto de escenarios, siéndole así familiares una vez comience la simulación real.

3.3.2.1 E0: Preparatorio

Como ya se ha expuesto, este nivel servirá para que el usuario pueda prepararse para hacer un uso correcto del simulador y que, mediante una experiencia preparatoria pueda manejar con soltura los diferentes elementos que se encontrará en la simulación real.

El escenario, libre de detalle, incluye componentes de todos los demás escenarios y en él se dispondrá de un tiempo y munición ilimitados de manera que el usuario pueda simplemente interactuar con el escenario, sin ser este evaluable.

3.3.2.2 Galería de Tiro

Con el fin de simplificar el diseño y modelado de los escenarios, se creará un escenario “tipo” sobre el que se realizarán modificaciones para el desarrollo de los escenarios funcionales que posteriormente se implementarán en el simulador.



Figura 3-14 Representación 3D de la galería de tiro.

Como se indicó anteriormente, se tomará como referencia la galería de tiro del Cuartel General Morillo para la reproducción del escenario. En él se podrán observar las diferentes líneas de tiro en las que se podrán realizar diferentes ejercicios, o para la futura implementación de múltiples usuarios.



Figura 3-15 Detalle de las líneas de tiro.

Además, se añadirán diferentes líneas en las que se podrán colocar los blancos a diferentes distancias de la línea de tiro con el fin de poder realizar todos los ejercicios. Estas estarán situadas a 25, 50 y 100 metros de la posición desde donde se abrirá fuego. Las distancias desde la línea de tiro hasta dichas líneas de blancos se han calculado empleando el Blueprint de medición.

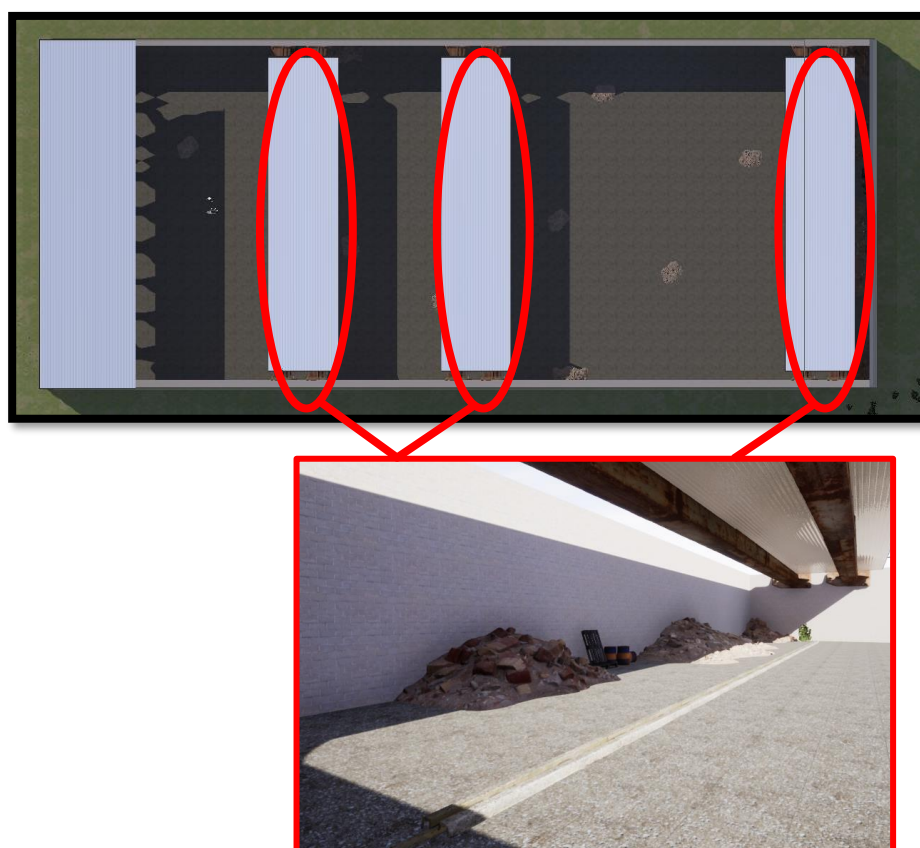


Figura 3-16 Detalle de las líneas de blancos.

Para el diseño y modelado del escenario se emplearán objetos 3D, materiales y texturas tanto de creación propia como obtenidas en las librerías de contenido de Epic Games. Además, algunos activos serán importados y posteriormente editados o modificados de acuerdo a las dimensiones y características del escenario en desarrollo.

3.3.2.2.1 E1: Blanco único

Este escenario, basado en la representación 3D de la galería de tiro, incluye todos los componentes de la misma, aparte de dos blancos a una distancia de 25 metros. Uno de ellos, el cual denominaremos blanco silueta, se empleará para los ejercicios que únicamente requieran que los impactos de los disparos se encuentren dentro de la silueta del blanco, sin importar el punto exacto del impacto. El otro, denominado blanco de incapacitación, será utilizado en los ejercicios en los que los impactos contabilizarán exclusivamente si se han realizado dentro de la denominada zona de incapacitación. En las Figura 3-18 y 3-15 se puede observar la disposición de los mismos.



Figura 3-17 Detalle del escenario E1.

3.3.2.2.2 E2: Blanco doble lateral

Del mismo modo que en el E1, este escenario se basa en el modelo 3D de la galería de tiro. En esta ocasión se han situado dos blancos silueta en la línea de 25 metros, con una separación lateral de 2,5 metros según indican las instrucciones del programa de tiro. En una línea de tiro contigua, y para la realización de otros ejercicios, se han situado otros dos blancos de incapacitación, también con una separación lateral de 2,5 metros. Las distancias entre los blancos se han calculado empleando el Blueprint de medición. A continuación, en la Figura 3-18, se muestra la disposición de este escenario.



Figura 3-18 Detalle del escenario E2.

3.3.2.2.3 E3: Blanco doble en profundidad

Este escenario es muy similar al anterior. Como se aprecia en la Figura 3-19, la diferencia consiste en que, en la línea de 25 metros se han incluido un blanco silueta, y en la línea de tiro contigua un blanco de incapacitación. Además, en la línea de 50 metros se ha añadido otro blanco de cada tipo, dejando un margen de separación lateral de 1,5 metros.



Figura 3-19 Detalle del escenario E3

3.3.2.2.4 E4: Blanco móvil

Este escenario, empleado únicamente para los ejercicios que requieren un blanco en movimiento, es en esencia igual que el E1, con la diferencia de que al blanco silueta se le ha añadido un Blueprint de movimiento que permite al blanco desplazarse lateralmente por la línea de 25 metros, recorriendo el ancho de la línea de tiro en la que se realizará el ejercicio. La velocidad de movimiento se ha establecido como la velocidad media a la que camina una persona, siendo esta de 5 kilómetros por hora.

3.3.2.2.5 E5: Modo nocturno

Este es el último escenario basado en el modelo 3D de la galería de tiro. En él se han incluido todos los tipos de blanco, con sus diferentes disposiciones, para poder realizar todos los ejercicios en modo nocturno. En este caso la iluminación del escenario ha sido reducida y, a petición del instructor, esta disminuirá casi al completo durante un tiempo preestablecido para cada ejercicio. En este periodo de tiempo el usuario podrá abrir fuego sobre los blancos para la realización de los ejercicios. En la Figura 3-20 se puede observar el cambio de iluminación que se produce en este escenario.

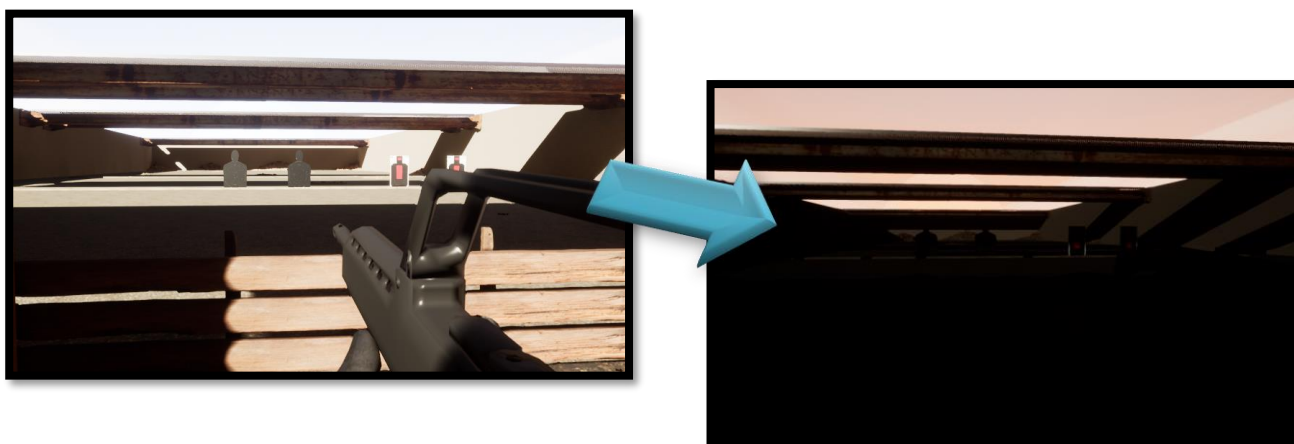


Figura 3-20 Detalle del escenario E5

3.3.2.3 E6: Tiro de precisión

Este último escenario entraña una dificultad mayor al resto debido a su extensión. En el usuario podrá adiestrarse en el tiro de precisión, en el cual varios factores afectan a la trayectoria que el proyectil realiza desde que se abre fuego hasta que impacta en el blanco. Este concepto es conocido como balística externa.

Aunque en la realidad son muchos los factores que influyen en la balística externa, de manera didáctica solo se incluirán en el simulador aquellos con un impacto mayor sobre la misma, como pueden ser la fuerza del viento sobre el proyectil o el efecto de las precipitaciones, en el supuesto de que se diesen. Estos factores se introducirán más adelante como variables con un valor aleatorio que el usuario conocerá para poder realizar los cálculos necesarios para llevar a cabo el tiro.

El modelado del escenario incluye una línea de tiro similar a los escenarios anteriores solo que en este caso la galería de tiro no está cerrada, sino que se abre hacia un terreno abierto en el cual se han situado diversos blancos, que denominaremos blancos de precisión, a 750, 1000, 1250 y 1500 metros de distancia. El terreno entre la línea de tiro y dichos blancos está modelado con los modos Terreno y Follaje (ver Interfaz Unreal Engine 4) para dotar de mayor realismo y detalle al escenario.

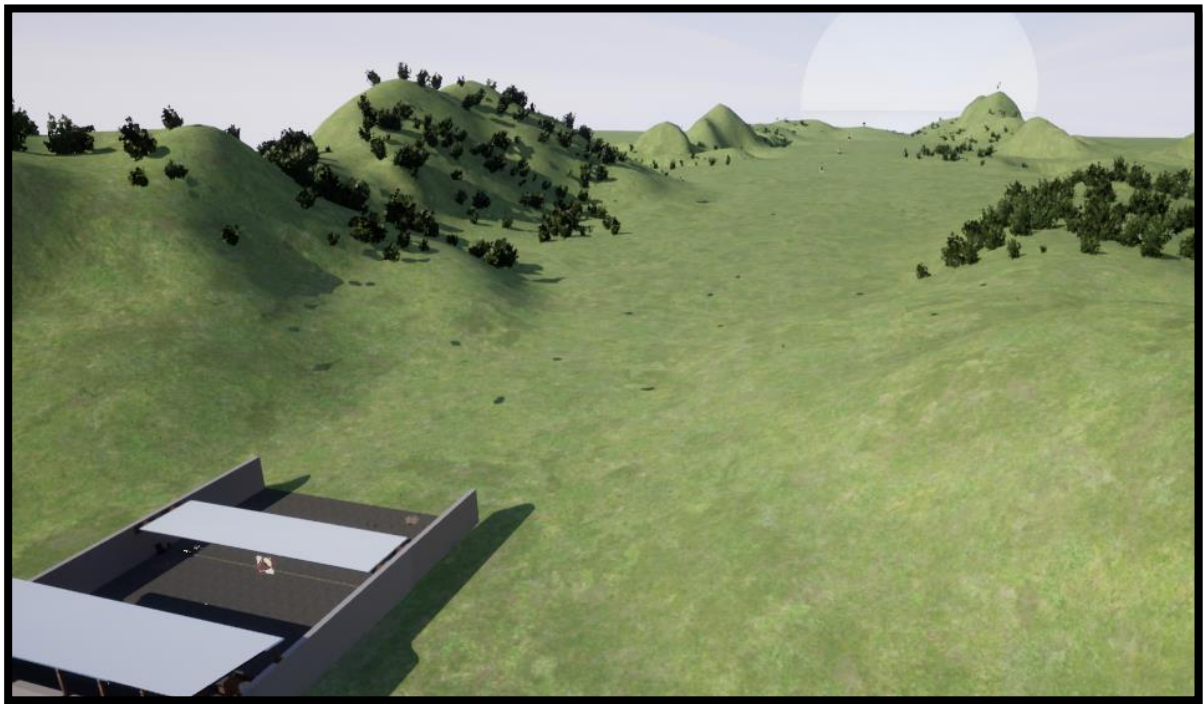


Figura 3-21 Detalle del Escenario E6

3.3.2.4 E7: Poblado Afgano.

De manera adicional y ajena a realización del programa de tiro seleccionado para el simulador, se ha diseñado un escenario con una representación 3D de un poblado afgano con el fin de adiestrar al combatiente en situaciones lo más realistas posibles.

A pesar de que el diseño del escenario es totalmente ficticio, se ha tomado como referencia el poblado afgano que se encuentra en inmediaciones de la Base Militar General Morillo. El escenario cuenta con varias edificaciones a las que el usuario puede acceder, progresando por diversas estancias en las que aparecerán diversos blancos conforme vaya progresando.

Este escenario permite evaluar la soltura del usuario en escenarios más parecidos a los que se podrá encontrar en la realidad, así como el tiempo de reacción ante la aparición de diferentes amenazas.



Figura 3-22 Detalle del Escenario E7

3.3.3 Third Person Player

El Third Person Player, o FPS, es la representación física del usuario en la simulación y que permite a este sentirse realmente parte de la misma. Aunque el desarrollo de un personaje concreto no sea estrictamente necesario para el correcto funcionamiento del simulador de tiro, esto lo dotará de un mayor realismo, permitiéndonos además añadir elementos adicionales al personaje como son diferentes sistemas de armas individuales para la realización de diferentes ejercicios de tiro.

3.3.3.1 Modelado del personaje

Con el fin de que el usuario pueda verse inmerso en el entorno virtual del simulador de tiro es de especial importancia crear, además de un entorno realista y funcional, un personaje que represente al usuario del simulador dentro del entorno. Además, será este personaje el que pueda realizar las acciones que permitan llevar a cabo los ejercicios de tiro propios de cada uno de los escenarios.

Para poder cumplir con los objetivos del simulador de tiro el personaje deberá poder realizar lo siguiente:

- Movimiento libre y realista por el escenario.
- Adoptar la posición de rodilla en tierra y moverse en esta posición para adoptar posiciones de tiro alternativas.
- Saltar por encima de objetos.
- Portar sistemas de armas individuales.
- Abrir fuego con los sistemas de armas y ser capaz de recargar las mismas.
- Apuntar por medio de un visor y hacer zoom con el mismo para ampliar el rango de visión del usuario.
- Impactar e influir en los diferentes blancos.
- Presentar en pantalla la munición restante y el tiempo disponible para la simulación.

- Presentar en pantalla una imagen en tiempo real de los blancos sobre los que se abre fuego.
- Recibir daño para posibles escenarios alternativos.
- Navegar por el menú principal para seleccionar las opciones de simulación y el escenario deseado.
- Presentar en pantalla los resultados de la simulación una vez finalizada la misma y guardarlos en los archivos locales del ordenador.

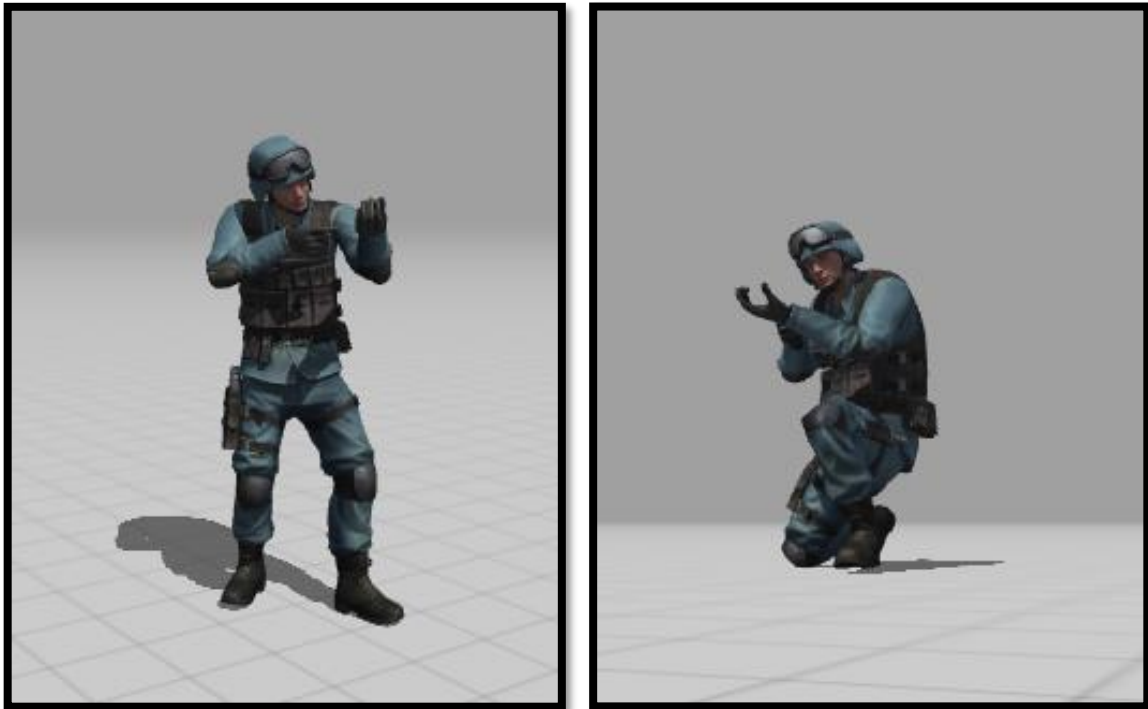


Figura 3-23 Swat Guy

El modelo estático que forma la base del personaje es el que se puede observar en la Figura 3-23. Este ha sido obtenido a través de las bibliotecas de contenido de la plataforma Mixamo (ver Mixamo). En este caso, el modelo seleccionado para el personaje que representará al usuario en el entorno virtual es el de “Swat Guy”. Esto permitirá al usuario mirar a su alrededor y percibir una imagen virtual de su cuerpo basada en este modelo.

Para que el modelo estático y las animaciones que determinan sus movimientos operen con el entorno virtual se debe generar un Blueprint en el que se determinen todas las funcionalidades y al cual se le asignará la malla de esqueleto correspondiente a modelo estático obtenido anteriormente. Las animaciones a su vez se definen en otro Blueprint de animación que permite al usuario combinar diferentes movimientos como andar y correr en diversas direcciones, saltar, apuntar o recargar de una manera fluida y realista

Para que el personaje pueda portar el sistema de armas que se empleará en el simulador, es necesario incluir esta dentro del Blueprint del personaje para que puedan ser generadas una vez comience la simulación. El diseño y modelado del arma se explicará en el siguiente apartado, pero en el modelo del propio personaje se determina cómo se relacionan este y el arma.

Como podemos observar en la Figura 3-24, para que el personaje sostenga el arma de una manera realista hay que determinar, dentro de la malla de esqueleto, cuáles serán los huesos a los que se adherirá el arma. Para ello se selecciona el hueso en cuestión (en este caso el de la mano derecha) y se le añade un “Socket” (clic derecho sobre el hueso y selección de la opción “Add Socket”) al que llamaremos “Weapon attach”. Una vez creado, se añade un activo de vista preliminar que nos permitirá visualizar cómo quedará el arma una vez sea generada en la simulación. Esta se deberá escalar y situar de manera que parezca que el personaje la porta de manera natural.

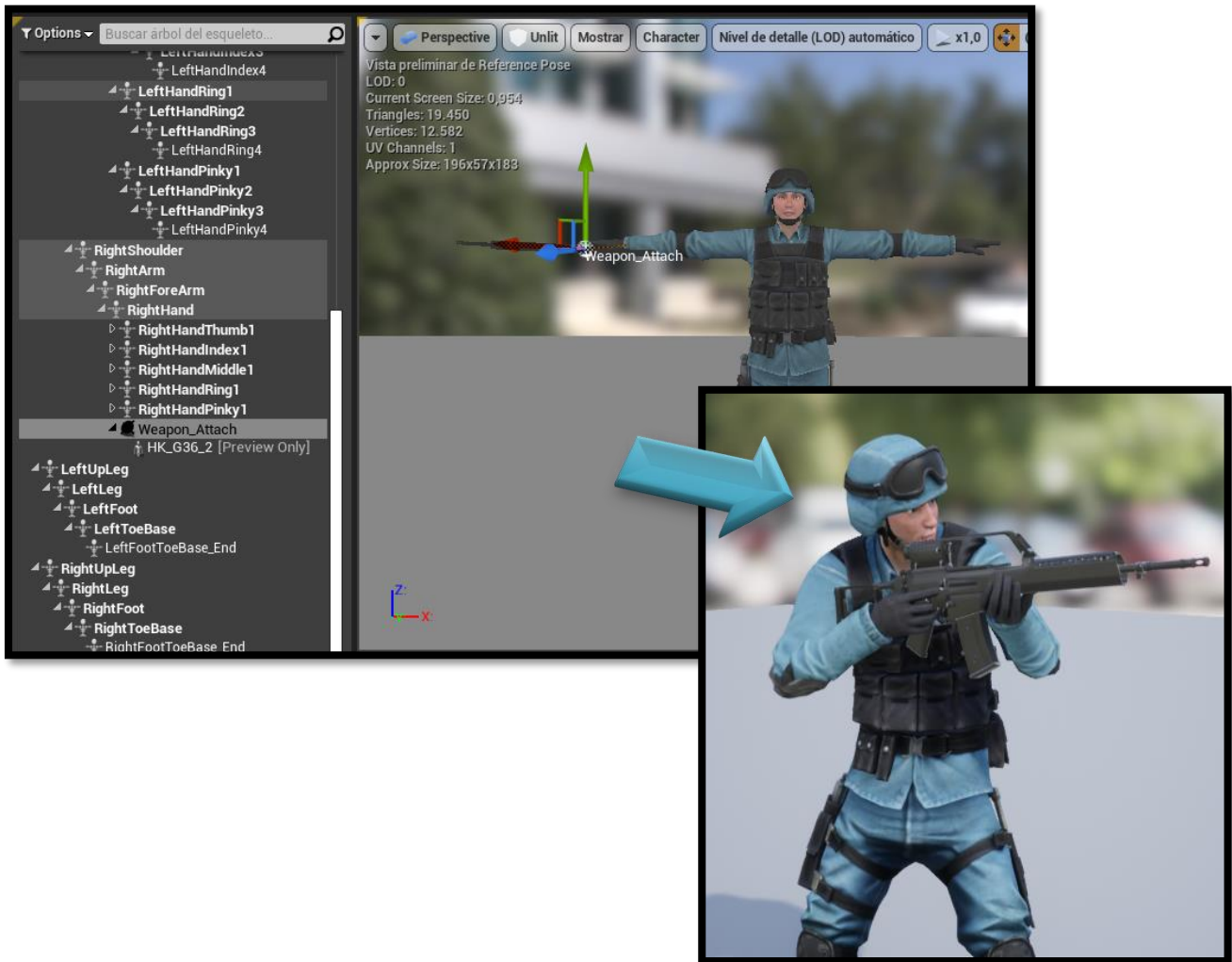


Figura 3-24 Unión entre personaje y arma

Una vez modelado y diseñado el personaje, solo queda definir los límites por los que podrá moverse durante la simulación. Para ello, dentro de los propios escenarios, se añadirá un nuevo componente llamado “Volumen de límites de malla de navegación” y se ajustará al espacio por el que se podrá mover el personaje.

3.3.3.2 Modelado del HK-G36E

El sistema de armas seleccionado para el simulador de tiro se trata del fusil de asalto empleado, en gran mayoría por las Fuerzas Armadas de España en la actualidad. Este es el fusil de la marca Heckler & Koch G36E. Para ello se importará un modelo 3D de dicha arma en el motor de juego y, del mismo modo que con el personaje, se ha de crear un Blueprint propio del arma que permita al usuario emplearla en el simulador de la manera más parecida a la realidad.

EL modelo 3D del HK G36E empleado en el motor de juego se obtuvo de manera gratuita en una de las diversas bibliotecas de modelos 3D online (Sketchfab). Este modelo se introdujo en Unreal Engine en el formato aceptado por el motor de juego (.fbx) y se le añadieron diversas texturas de creación propia para dotar al arma de realismo.



Figura 3-25 Modelo 3D del HK G36E

En la Figura 3-25 se puede observar la malla estática que conforma el HK G36E dentro de Unreal Engine, una vez aplicadas las texturas para la misma.

De cara a que el personaje pueda realizar todas las acciones necesarias (ver Modelado del personaje) y que el resultado sea lo más preciso y parecido a la realidad posible, el arma debe realizar las siguientes funciones:

- Disparar en modo tiro a tiro o en modo automático cuando lo requiera el usuario.
- Disponer de una cantidad determinada de munición.
- Administrar la munición máxima y la munición en el cargador una vez el usuario recargue.
- Generar un proyectil en la boca del cañón una vez se abre fuego para que este pueda interactuar con los blancos.
- Generar un sistema de partículas y efectos que simulen el fognazo de da disparo.
- Generar un sonido de disparo.
- Simular el efecto del retroceso en cada disparo.

Del mismo modo que se hizo con el modelo 3D del personaje para añadirle el arma, hay que seleccionar un punto determinado de la malla de esqueleto del arma donde se generarán los proyectiles y el efecto del fogonazo. Para ello se añade un “Socket” al que llamaremos “Muzzle” (en español, boca del arma). A este se le añadirá el activo de vista preliminar que corresponde con el efecto del fogonazo para que una vez se genere este durante la simulación, se haga con la dirección y el tamaño deseados.

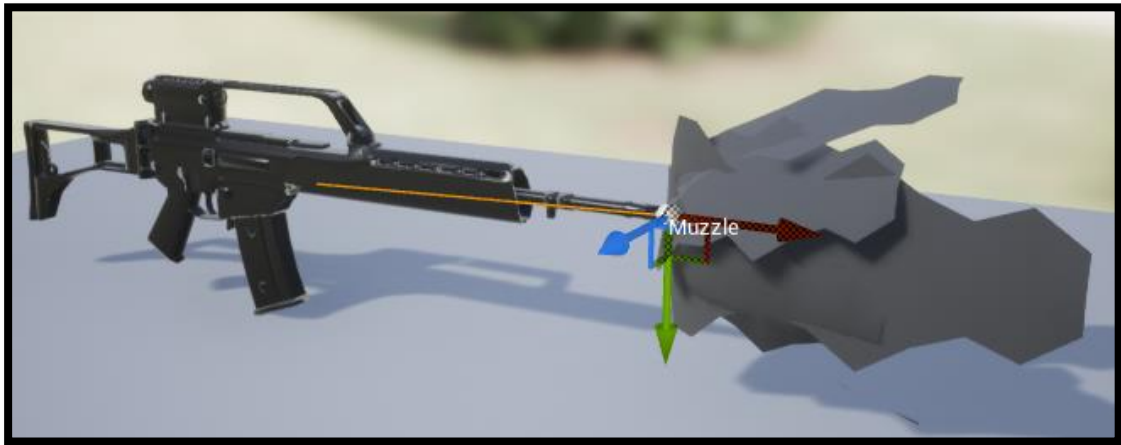


Figura 3-26 Edición de la malla esqueleto del arma

En la Figura 3-26 se puede ver el activo preliminar antes mencionado. A pesar de ello, este, al igual que el proyectil, serán generados durante la simulación mediante la implementación del Blueprint del arma, desarrollado posteriormente.

3.3.4 Blancos

Dado que en el programa de tiro seleccionado para su implementación en este proyecto se emplean diferentes tipos de blanco para los diversos ejercicios de tiro que componen el programa, del mismo modo se deberán diseñar, modelar y programar diferentes blancos que cumplan con una finalidad específica en cada escenario.

Los blancos se diferenciarán en su diseño, las zonas sensibles a los impactos de los proyectiles, su respuesta ante el impacto del proyectil y su movilidad. Atendiendo a estos factores, se emplearán los siguientes blancos:

- Blanco silueta
- Blanco de incapacitación
- Blanco móvil
- Blanco de precisión
- Blanco insurgente.

Para cada uno de los blancos se generará un Blueprint específico para que este genere los resultados deseados que se explicará más adelante. Más allá de esto, el diseño de los blancos difiere únicamente en la forma que se les ha dado y la textura empleada para el mismo.

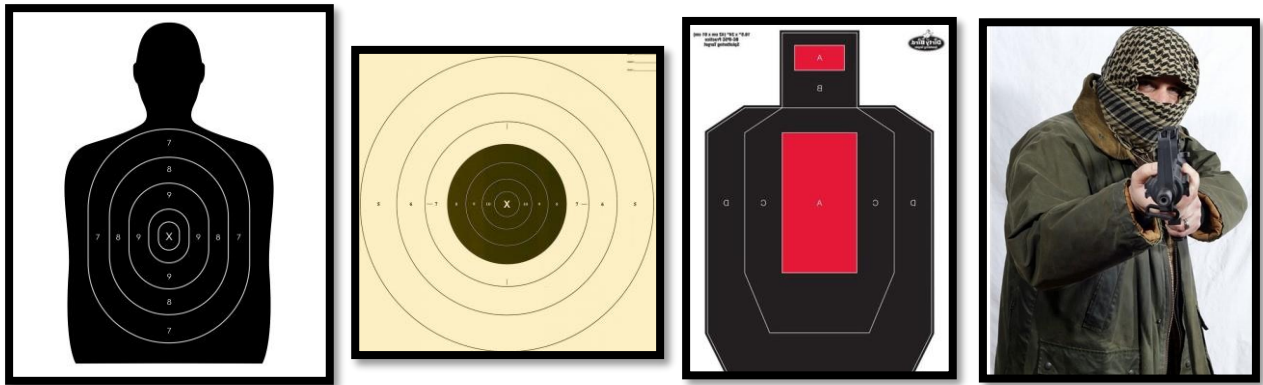


Figura 3-27 Blancos

En la Figura 3-27 se pueden ver algunas de las diferentes texturas empleadas en los diferentes blancos.

Las imágenes empleadas para generar las texturas de los blancos se han obtenido de una web dedicada a la venta de material y equipamiento para personal militar o de seguridad, entre el que se incluyen blancos para determinadas agencias o a particulares. [19]

3.4 Programación y código

Para que el diseño de los escenarios y los actores del simulador de tiro funcionen de manera correcta y puedan realizar las funciones necesarias para cumplir con los requisitos del programa de tiro dentro del simulador, es necesario programar en detalle todos los componentes del mismo.

En este subapartado se explicará pues a explicar el proceso de desarrollo del código empleado en el simulador tras una introducción a la programación mediante Blueprints, o programación multinodal. Después de esto se continuará con la programación del personaje que representa al usuario, los bloques de código para el sistema de armas, los blancos empleados, los HUD's (Heads-up display) que muestran en pantalla durante la simulación la información relevante y necesaria para la realización de los ejercicios, el menú principal del simulador y, por último, la pantalla de resultados.

3.4.1 Programación Blueprint

Como ya mencionamos anteriormente, Unreal Engine 4 ofrece la posibilidad de programar un proyecto tanto en C++ como en Blueprint. Para el presente proyecto se optará por la segunda opción debido a que presenta determinadas ventajas de visualización frente a la programación en C++.

El sistema de Blueprints permite al desarrollador crear, mediante un sistema de nodos, funciones de elevada complejidad e interacciones entre elementos sin la necesidad de los conocimientos necesarios para el desarrollo de códigos basados en líneas de lenguaje alfanumérico. EL motor de juego permite acceder a múltiples herramientas que, de otra manera, solo serían al alcance de expertos de programación, mediante un sistema intuitivo y visual.

Dicho sistema gráfico permite una visión global y, en consecuencia, un mayor control del código desarrollado. Crear secuencias de nodos y prever las necesidades para el correcto funcionamiento del código se convierte en algo rápido y eficaz, sin perjudicar por ello las infinitas posibilidades y el nivel del código.

En el presente proyecto todos los actores han sido programados mediante Blueprints, facilitando así la relación entre ellos y, por tanto, la comprensión del código en su totalidad. A continuación se explicará de manera detallada los diferentes tipos de Blueprint, nodos, variables y elementos empleados, para alcanzar una mayor comprensión del código que se desarrollará para cada uno de los actores.

La Figura 3-28 muestra un ejemplo de la visualización general del código de uno de los actores del proyecto, pudiendo observar en ella los distintos elementos que componen este tipo de programación y cómo, de manera visual, se puede ver de manera rápida y sencilla la relación entre ellos.

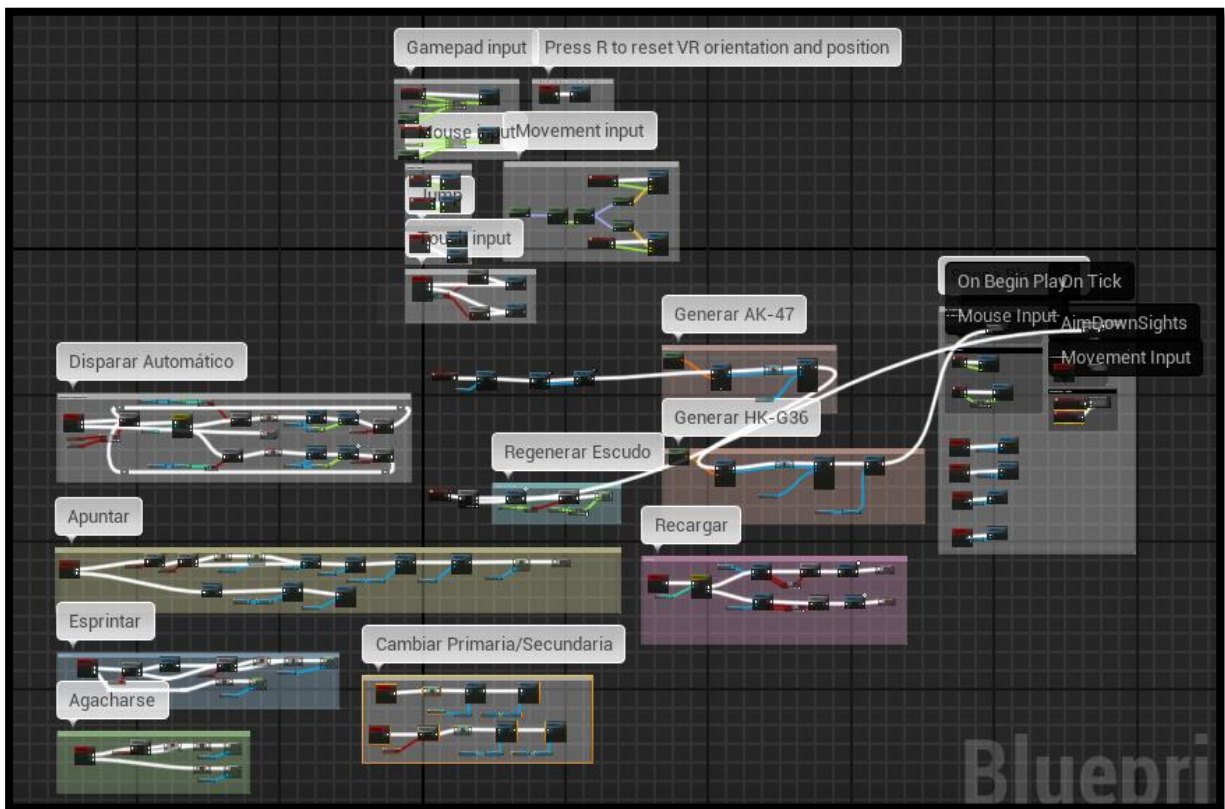


Figura 3-28 ThirdPersonBP

3.4.1.1 Tipos de Blueprint

Los Blueprints constituyen la principal herramienta de trabajo dentro del motor de juego Unreal Engine 4. Estos son activos que proporcionan al desarrollador una interfaz intuitiva sobre la que se generará el código visual, basándose en un nodo que permite crear nuevos tipos de actores y eventos. Estos a su vez podrán interactuar con el entorno creado, con otros actores y con el propio usuario.

Para el desarrollo de los Blueprints se dispone de tres ventanas de trabajo principales, además de un panel de detalles para el elemento seleccionado, una ventana con las funciones y variables que componen el Blueprint y una última ventana con una lista jerarquizada de los elementos y componentes del Blueprint. Las ventanas de trabajo empleadas son las siguientes:

- Ventana gráfica: Ofrece una imagen física del Blueprint en desarrollo, permitiendo la adición o modificación de componentes dentro del mismo. Estos componentes pueden asignarse a variables, dando acceso a los mismos desde el mismo u otro Blueprint.
- Construction script: Contiene un gráfico que permite al Blueprint la inicialización de operaciones dentro de la simulación, o al desarrollador generar funciones que no están predefinidas en Unreal Engine 4.
- Gráfico del evento: Contiene otro gráfico en el que se emplean eventos y funciones que determinan la relación y respuesta del Blueprint en cuestión a los sucesos durante la simulación.

Existen diferentes tipos de Blueprint y cada uno se emplea para un tipo de actor o recurso a emplear dentro de la simulación. En la Figura 3-29 se muestran los tipos de Blueprint más comunes y a

continuación, en los subtítulos del actual subapartado, se explican de manera más detallada la finalidad y utilidad de cada una de las clases ya que, para el desarrollo del simulador, se ha tenido que emplear en una o varias ocasiones cada uno de los diferentes tipos de Blueprint.



De izquierda a derecha, los tipos de Blueprint empleados son los siguientes:

Figura 3-29 Tipos de Blueprint principales

- Level Blueprint
- Blueprint Class
- Animation Blueprint
- Widget Blueprint

3.4.1.1.1 Level Blueprint

Es un tipo concreto de Blueprint que, dentro de un gráfico de eventos, permite al desarrollador manejar la programación dentro de un nivel concreto, donde se encuentra uno de los escenarios específicos del simulador de tiro.

Cada nivel posee su Level Blueprint específico y no se permite la creación de uno nuevo a través de la interfaz del editor. En él se incluyen los eventos que afectarán al nivel como un todo, incluyendo todos sus componentes, activando secuencias de acciones mediante “Llamadas de Función” u “Operaciones de Control de Flujo”.

El acceso a este Blueprint se encuentra a través de la barra de herramientas, abriendo el desplegable “Blueprint”, y haciendo clic en la opción “Open Level Blueprint”.

3.4.1.1.2 Blueprint Class

Se trata de un tipo de Blueprint que ofrece la posibilidad de añadir contenido y funcionalidades específicas a los diferentes tipos de actores dentro de la simulación. Estos pueden ser creados de manera visual, sin necesidad de aplicar líneas de código, y guardarse como activos dentro de la carpeta de contenidos del proyecto. De este modo, un mismo Blueprint puede ser empleado tantas veces y en los niveles que se desee sin tener por ello que duplicar el código.

Cuando se crea un nuevo Blueprint Class, Unreal Engine ofrece una serie de clases “Padre” predeterminadas de las que nuestro Blueprint heredará las propiedades principales que podremos modificar. Estas clases predeterminadas son las siguientes:

- Actor: Se trata de un recurso u objeto que puede colocarse en la escena que no dispone de movilidad propia.
- Peón: Es un actor que puede ser controlado o recibir inputs de un controlador, como puede ser un controlador de inteligencia artificial (AI Controller).
- Personaje: Es un tipo de peón que tiene la capacidad de moverse con libertad por la escena y al que se pueden aplicar animaciones.
- Controlador del reproductor: Es el recurso empleado para controlar al personaje que empleará el usuario.
- Modo de Juego: Este recurso define el tipo de simulación, sus reglas y sistemas de puntuación y cualquier otra faceta relevante al tipo de simulación.

- Componente de actor: Es un componente reutilizable que puede ser añadido a cualquier actor dentro de la ventana de componentes.
- Componente de escena: Se trata de un componente que contiene una transformación del escenario o nivel.

3.4.1.1.3 Animation Blueprint

Con el fin de generar una secuencia de animaciones para una determinada malla de esqueleto, es necesario programar un Blueprint de este tipo para determinar los inputs necesarios para que la malla realice las animaciones deseadas y los detalles y transiciones entre estas.

Las animaciones se unen a otras mediante “Zonas de Combinación” de animaciones, lo que controla los huesos de la malla de esqueleto para que las transiciones de una animación a otra transcurran de una manera fluida.

En la Figura 3-30 se muestra uno de los nodos del Blueprint de animación desarrollado, en el que se determina cuándo y cómo se pasará de una animación a otra.

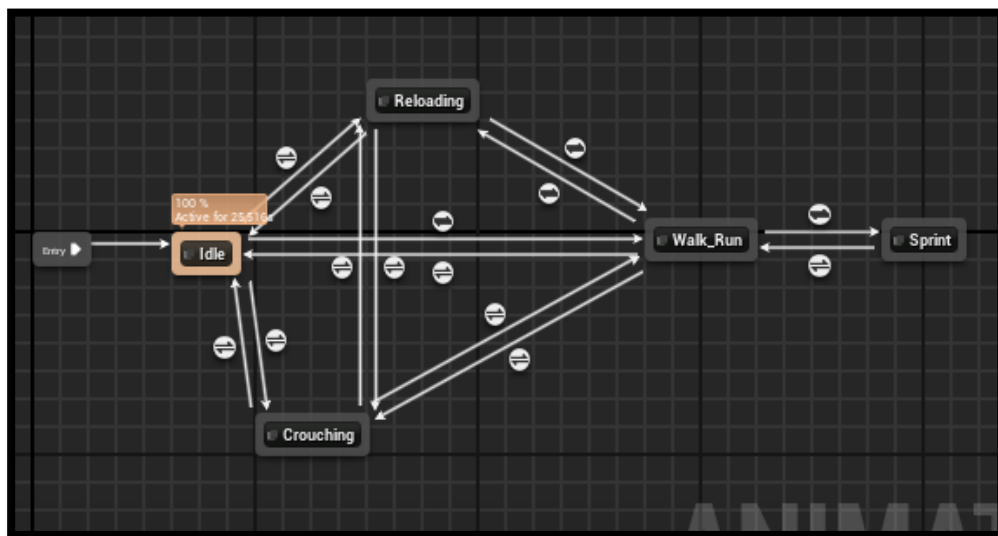


Figura 3-30 Blueprint de animación

Además, como se aprecia en la Figura 3-31, las “Zonas de Combinación” unen diferentes sets de animaciones para que transcurran de manera natural cuando el actor cambia de dirección o velocidad.

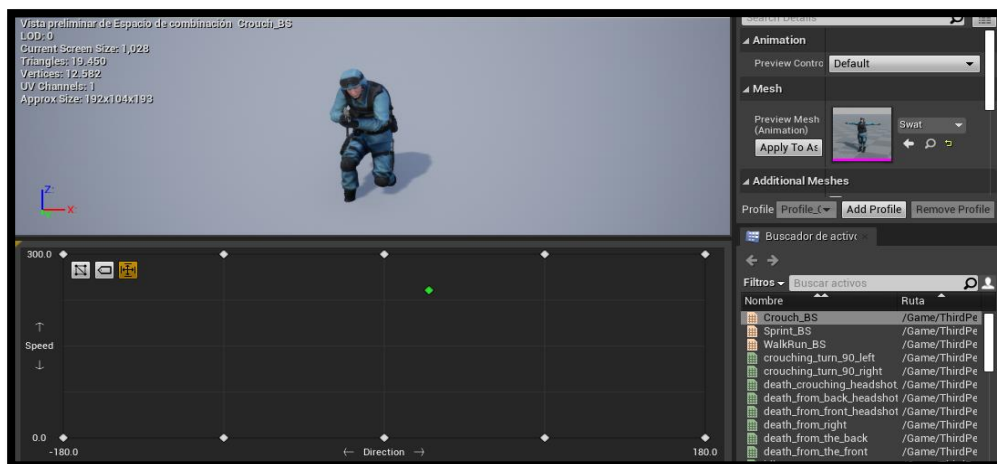


Figura 3-31 Zona de Combinación

3.4.1.1.4 Widget Blueprint

Esta clase de Blueprint permite la creación de menús o HUD que se verán en pantalla en los momentos de la simulación que el desarrollador crea conveniente. Además, permite añadir funcionalidades a los elementos que se muestran en pantalla, como es el caso de botones o imágenes de la escena en tiempo real durante la simulación.

3.4.1.2 Tipos de nodo

Una vez explicado y definido el concepto de Blueprint, junto con los diferentes tipos que nos encontraremos en este proyecto, procederemos a presentar los diferentes tipos de nodo que componen este lenguaje y sus funcionalidades.

En primer lugar, los elementos que suelen comenzar las secuencias de programación multinodal son los eventos. Estos son activados cuando se da una determinada situación u ocurre un suceso durante la simulación. Existen eventos predeterminados propios del nivel que se ejecuta, como es el ejemplo de “BeginPlay”, que se activa una vez comienza la simulación, o eventos relacionados con alguno de los actores presentes en la escena.



Figura 3-32 Nodo evento

En la Figura 3-32 se muestra a la izquierda un ejemplo de los posibles eventos que se ofrecen en el panel de detalles de uno de los componentes seleccionado dentro de un Blueprint, y a la derecha, uno de estos eventos implementados en el Gráfico del Evento del Blueprint.

A continuación, uno de los tipos de nodo más importantes a la hora de programar son las funciones. Unreal Engine ofrece un amplio abanico de funciones predefinidas que se pueden aplicar a cualquier proyecto, además de permitir al desarrollador crear funciones propias en el Construction Script. Una vez creadas, estas funciones pueden ser llamadas dentro del código general del Blueprint.



Figura 3-33 Nodos función

En la Figura 3-33 se puede ver, a la izquierda, un ejemplo de función predeterminada y, a la derecha, una llamada a una función propia creada dentro del propio Blueprint.

Para referirse a un actor o un Blueprint determinado desde otro, se emplean los nodos de proyección. Esto permite al desarrollador acceder a los componentes y variables de un Blueprint desde otro, ya sea para obtener sus valores o datos como para modificarlos. En el simulador de tiro se deberá proyectar en numerosas ocasiones al Blueprint del personaje, llamado “ThirdPersonCharacter”, ya que es este el que controlará el usuario e interactuará con el resto de elementos de la simulación. En la Figura 3-34 se muestra este tipo de nodo.

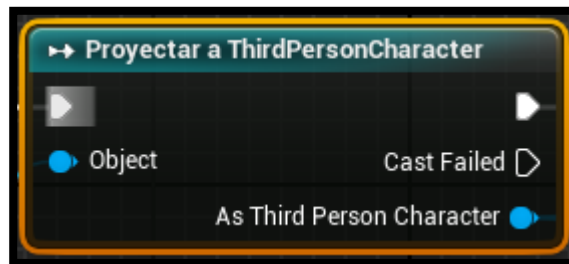


Figura 3-34 Nodo de proyección

Con el fin de mostrar en pantalla los elementos creados en los Widget Blueprints, se debe emplear el nodo “Create Widget”, lo que hará que aparezca en la pantalla del usuario el diseño del widget creado una vez se dé el evento que precede a la creación del mismo. El único widget que no requiere de activación es el HUD seleccionado por defecto en el modo de juego de cada nivel dentro del panel de ajustes del mundo. La Figura 3-35 muestra este tipo de nodo.



Figura 3-35 Nodo Create Widget

Por último, existen los nodos de ramificación que permiten crear condiciones bajo las cuales la simulación implementará determinadas secuencias de código. Los más comunes son los nodos Bifurcar, Secuencia y Cambiar a número entero.

El primero de ellos determina si una condición es verdadera o falsa, tomando un camino u otro en base a esta. El segundo ejecuta una serie de secuencias de código en un orden preestablecido. El último determina qué secuencia de código debe ejecutar en base a un identificador (integer).

3.4.1.3 Variables

Dentro de las secuencias de código, los Blueprint emplean variables de diferentes tipos. Estas tienen un valor inicial predefinido establecido por el desarrollador, pero pueden ser modificados dentro del propio código en el transcurso de la simulación.

Para modificar los valores de las variables dentro del código se emplean nodos operadores y nodos de ajuste de variables, en los que se introduce el valor resultante del operador o se le da un nuevo valor establecido también por el desarrollador, una vez se da la condición previa a este nodo.

Los principales tipos de variable existentes y que se pueden emplear en Unreal Engine 4 son las que aparecen en la Figura 3-36.

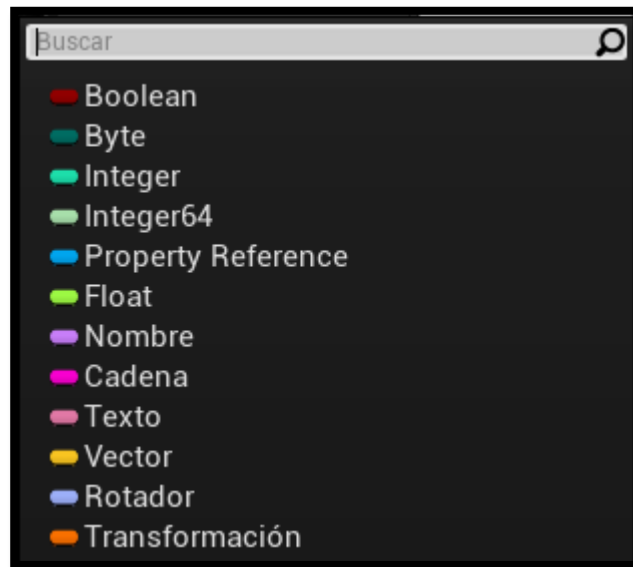


Figura 3-36 Tipos de variables

3.4.1.4 Materiales y Texturas

Para dar realismo a los escenarios y los actores Unreal Engine emplea materiales y texturas que se pueden aplicar y combinar. De manera simplificada, los materiales podrían considerarse la “pintura” de los objetos situados en el escenario, pero van más allá ya que son los que definen el tipo de superficie del que en apariencia está hecho un objeto.

Los materiales tienen planos específicos donde se el desarrollador puede editarlos para darles el color, brillo, opacidad... deseados, definiendo, a fin de cuentas, cómo interacciona la luz que impacta con un objeto con su superficie. Esto se consigue empleando referencias a las texturas que se encuentran dentro de los contenidos del proyecto, operadores matemáticos y las diversas propiedades inherentes a los materiales dentro del motor de juego.

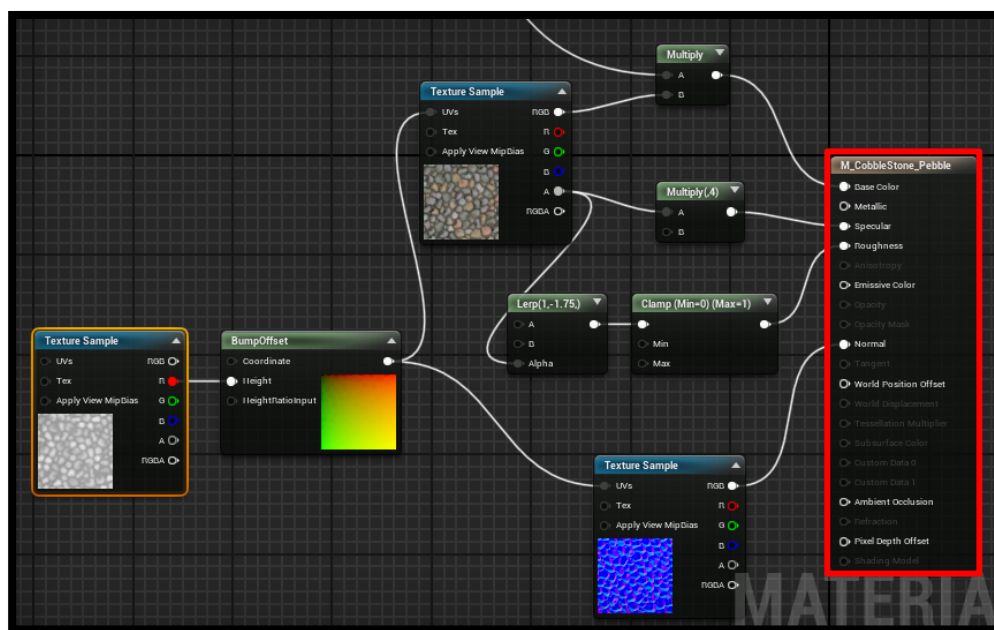


Figura 3-37 Plano de Material

En la Figura 3-37 se muestra un ejemplo de uno de los materiales empleados en el presente proyecto, donde se puede ver cómo se emplean diferentes texturas junto con diversos operadores, conectando los resultados a los diferentes valores del nodo central (señalado en rojo) que nos dará el material resultante que podremos aplicar dentro del escenario.

La gran mayoría de los materiales empleados en el presente proyecto han sido obtenidos del contenido inicial que ofrece Unreal Engine cuando se crea un nuevo proyecto y de las bibliotecas de contenido de Epic Games Store.

3.4.2 Programación del personaje

En este subapartado se procede a explicar la programación del personaje que constituye la representación física del usuario en la simulación y le permite realizar las funciones necesarias para completar el programa de tiro dentro del simulador, es decir. Estas funciones fueron establecidas en el diseño del personaje (ver subapartado 3.3.3.1 Modelado del personaje).

3.4.2.1 Componentes del personaje

En primer lugar, es necesario explicar cuáles son los componentes que constituyen el ThirdPersonCharacter que más adelante se emplearán dentro del Gráfico del Evento. En la Figura 3-38 se muestra la Ventana Gráfica del Blueprint junto con la lista jerarquizada de sus componentes.

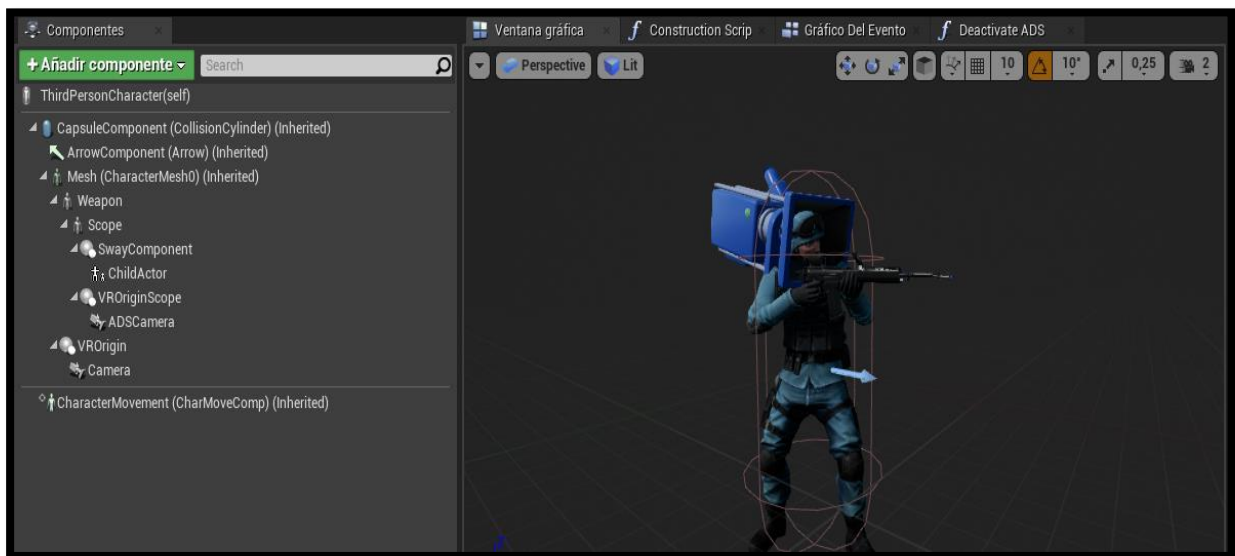


Figura 3-38 Componentes del personaje

Englobando al personaje encontramos en primer lugar la cápsula de colisión (Capsule Component) que nos permitirá generar eventos de colisión con nuestro propio personaje, representando el espacio que el motor asume que ocupa una vez el personaje es situado dentro del escenario. El “Arrow Component” indica la dirección principal hacia la que se mueve el personaje, estableciendo al frente el origen de la dirección.

Dentro de la cápsula de colisión se encuentra la malla de esqueleto del personaje en la posición que se ha determinado como estándar dentro del Blueprint de animación que más adelante se expondrá. A esta malla se subordina el arma que adherimos al personaje en el diseño de mismo y es generada en la simulación una vez comienza esta. La Figura 3-39 muestra el código empleado para generar dicha arma.

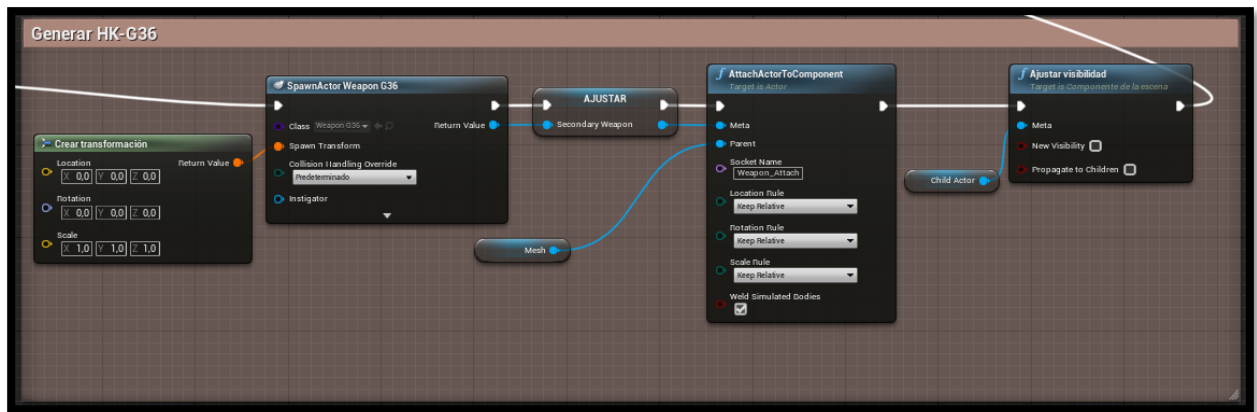


Figura 3-39 Generar HK G36

Para generar el arma se hace referencia a un Blueprint particular de la misma denominado “Weapon-G36”. En el código podemos observar que en los dos últimos nodos se establece la visibilidad de un componente como nula. Este componente es el visor del arma (Scope), desde el cual se hace referencia al Blueprint del que implementará el visor en la simulación y que a su vez se divide en dos componentes.

El primero de estos dos componentes es el “Child Actor”, el cual nos permitirá posicionar el visor dentro de la ventana gráfica de tal manera que, cuando se apunte con el arma, las trazas de los proyectiles estén alineadas con la cruz filar del visor. El otro componente es la “ADSCamera”, empleada para ver a través del visor cuando se realiza la acción de apuntar.

Subordinado a la malla estática del personaje se encuentra también el componente “Camera” que constituye la cámara principal de nuestro personaje y la que será la vista principal mientras dure la simulación. Esta ha sido acoplada al hueso de la cabeza para que se mueva de manera solidaria con esta cuando el personaje realiza cualquier movimiento que conlleve una animación.

Por último, se encuentra el componente “CharacterMovement” que nos permitirá establecer los parámetros básicos de movimiento del personaje tales como la velocidad máxima al correr, la fuerza con la que salta el personaje, el control de la rotación y otras muchas posibilidades.

Dado que el proyecto debe poder implementarse en Realidad Virtual, existen dos componentes denominados VROrigin a los que se han acoplado ambas cámaras. Estos permiten que, una vez iniciada la simulación en Realidad Virtual, el usuario pueda meterse en la piel del personaje y visualizar el entorno en primera persona, sin que la cámara se desacople del personaje y de lugar a fallos de simulación.

3.4.2.2 Movimiento

Con el fin de que el usuario pueda moverse con el personaje por el escenario de manera libre, es necesario introducir una serie de entradas que el motor pueda reconocer y que hagan que el personaje realice las acciones que el usuario le indica.

Para ello hay que abrir antes de nada los ajustes del proyecto, desplegando la ventana “Editar” y haciendo clic en “Ajustes del Proyecto”. La interfaz abrirá una nueva pestaña en la cual, dentro de los ajustes del Motor se seleccionará la opción de Entradas. Este proceso se realizará cada vez que se quiera añadir un input nuevo a la simulación, como puede ser el pulsado de diferentes teclas o botones de los Motion Controllers.

En la Figura 3-40 se muestran, señaladas en rojo, las diferentes entradas creadas para el movimiento del personaje. A estas se hará referencia desde el Gráfico del Evento del Blueprint del personaje con nodos de tipo evento que implementarán la secuencia de código a la que preceden.

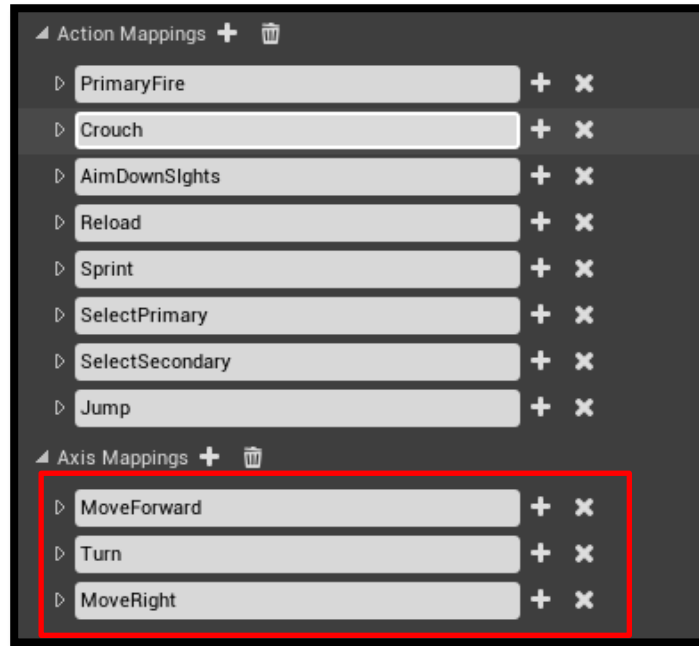


Figura 3-40 Entradas del proyecto

Estas acciones nos permiten desplazarnos frontalmente, hacia ambos lados, y marcha atrás. Además, permiten al personaje pivotar sobre sí mismo sin desplazarse.

Aparte de los movimientos básicos, al personaje se le dará la posibilidad de esprintar, agacharse para adoptar la posición de rodilla en tierra y saltar, para poder sobrepasar obstáculos. Para evitar que las animaciones se mezclen y produzcan fallos en la simulación, y para dar realismo a la misma, estas acciones tienen ciertas limitaciones. El personaje no podrá esprintar mientras se encuentra rodilla en tierra y viceversa. Además, tampoco podrá apuntar mientras esprinta.

Para lograr esto se introducen en el personaje una serie de variables de tipo “Boolean” con valores de verdadero o falso, el cual cambiará si el personaje está realizando una u otra acción. En la Figura 3-41 se muestra la secuencia de código para que el personaje se agache en la cual se puede observar cómo se realiza la comprobación de si el personaje estaba previamente esprintando, y de ser así, desactivar la variable “Sprint True” para posteriormente activar la variable “Crouch True”. Finalmente se ajusta la velocidad máxima del personaje ya que, estando agachado, esta debe ser menor que la velocidad normal a la que puede andar.

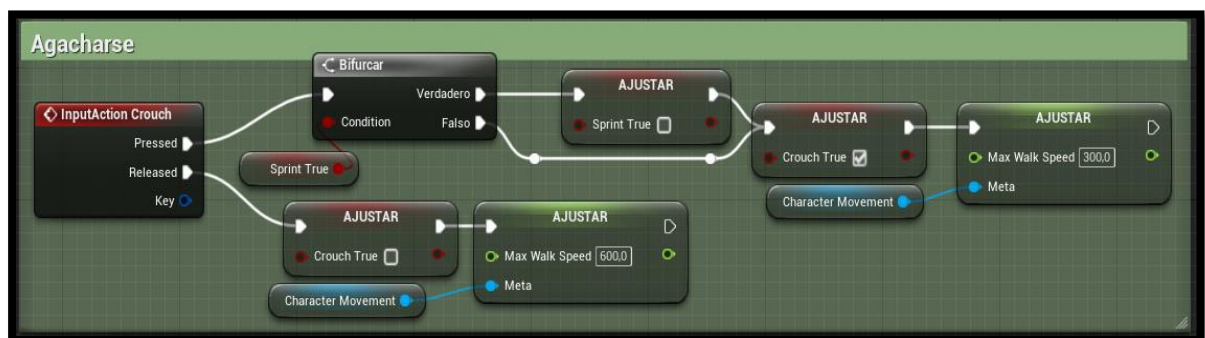


Figura 3-41 Función agacharse

Una vez se vuelve a poner en pie el personaje, la velocidad máxima se reestablece a su valor original.

3.4.2.3 Animaciones de movimiento

Ningún movimiento de los anteriormente expuestos se podría realizar dentro de la simulación sin la existencia de un Blueprint de animación al que se haga referencia desde el ThirdPersonCharacter. Para ello se ha creado el Blueprint “SwatAnim_BP” en el cual se establece cómo se comportará el personaje según las demandas del usuario. El código completo de este Blueprint puede verse en el Anexo III: Bloques de código de animación del personaje. En este subapartado se explicarán los conceptos básicos y la combinación de las animaciones empleadas.

Dentro del Gráfico de animación del Blueprint se ha creado una máquina de estado que determina, según los inputs que recibe el motor de juego del jugador durante la simulación, la posición final del personaje. Esta máquina de estado se ha denominado “Walk_Run”. Por otro lado, en el gráfico del evento se han creado dos variables tipo “Float” para la velocidad y dirección del personaje que nos permitirán establecer, dentro de las Zonas de combinación, que velocidad y dirección tomará el personaje para cada animación.

Además, se han creado tres variables tipo “Boolean” para determinar si el personaje esta agachado, esprintando o recargando. Estas reciben su valor desde el Blueprint del personaje e indican al Blueprint de animación que realicen una u otra acción. En la Figura 3-42 se aprecia cómo se establecen todas estas variables.

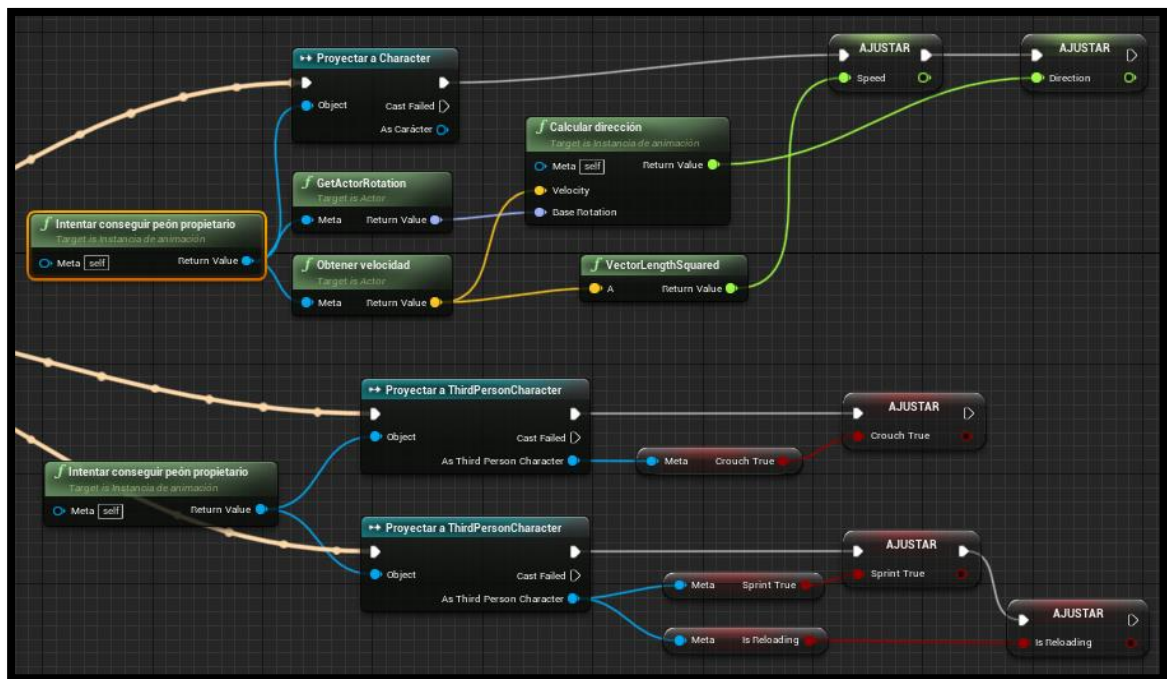


Figura 3-42 Variables de animación

Las tres Zonas de combinación creadas son, de izquierda a derecha en la Figura 3-43, “WalkRun_BS”, “Crouch_BS” y “Sprint_BS”. Para implementarlos se hace una llamada a los mismos dentro de los nodos de la máquina de estado “Walk_Run”.

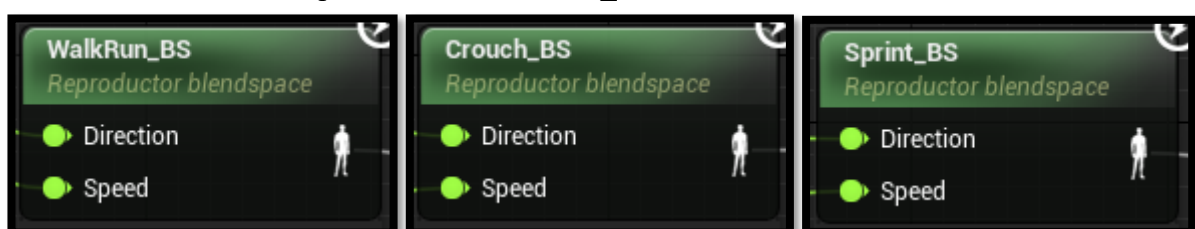


Figura 3-43 Zonas de combinación

Si las variables mencionadas anteriormente toman el valor de verdadero, el personaje realizará la acción correspondiente. En caso contrario, el personaje adoptará la postura indicada como predeterminada.

3.4.2.4 Cambiar de arma

Aunque el simulador de tiro está enfocado principalmente al adiestramiento con el sistema de armas seleccionado, es decir, el fusil de asalto HK G36E, se ha implementado la posibilidad de realizar un cambio de arma que permitiría realizar otros ejercicios de tiro que impliquen transiciones entre el arma primaria y la secundaria de un combatiente.

Para ello, una vez comienza la simulación, se generan ambas armas, pero para que no aparezcan las dos en pantalla, se ajusta su visibilidad, de tal modo que el arma principal tenga la visibilidad activada y la secundaria desactivada. Una vez el usuario realice la acción que active la entrada de cambio de arma, la visibilidad de ambas se alternará, dando lugar a la transición entre primaria y secundaria. En la Figura 3-44 se muestra el código empleado para la transición entre armas.

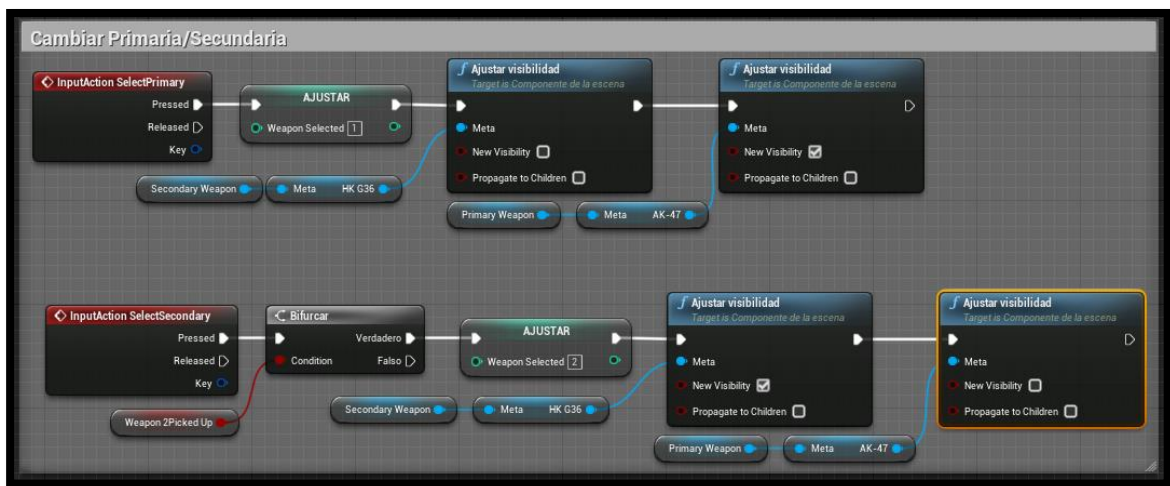


Figura 3-44 Función cambiar arma

En el presente proyecto se ha importado como ejemplo la malla estática de un fusil de asalto AK-47 a modo de ejemplo. Este podría ser sustituido por un modelo 3D de cualquier otro sistema de armas. Además, todas las funcionalidades del arma principal han sido duplicadas para que funcionen de manera independiente con cada arma, como la generación de proyectiles, el apuntado, la recarga y la gestión de la munición. Esto se ha hecho duplicando el Blueprint del arma principal, dándole el nombre del arma secundaria y modificando las referencias a la misma dentro del Blueprint de nuestro personaje. En la Figura 3-45 se puede apreciar el resultado el cambio de arma.

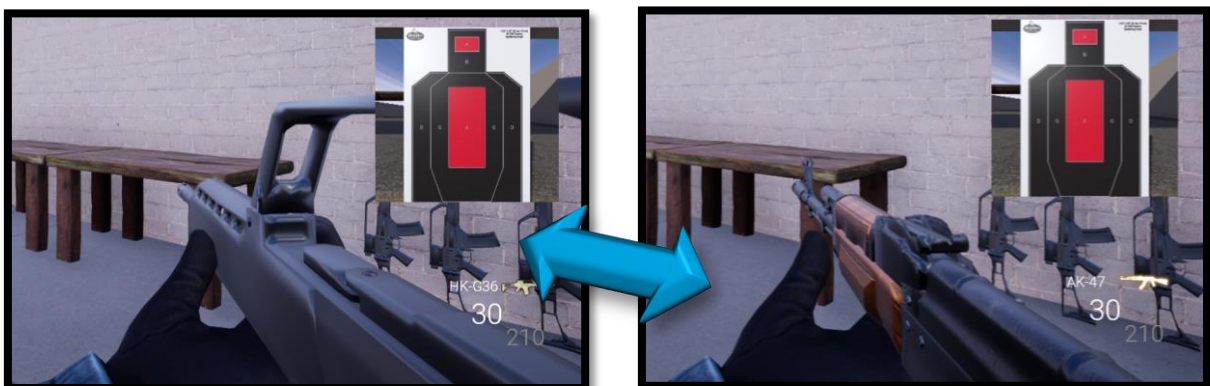


Figura 3-45 Transición entre arma primaria y secundaria

3.4.2.5 Apuntar

En este subapartado se explicará cómo se realiza el proceso de apuntado del arma mediante Blueprint. Dado que lo que el usuario ve durante la simulación es lo que muestra la cámara que esté activada en el momento, para apuntar un arma existen dos opciones. La primera consiste en trasladar la cámara principal del personaje desde su posición original en la cabeza del mismo hasta el lugar donde se encuentra el visor. La segunda opción, y la implementada en este proyecto, consiste en emplear dos cámaras diferentes que se activarán o desactivarán en función de si el usuario apunta o no con el arma.

La cámara principal, que se activará cuando comienza la simulación, está situada a la altura de los ojos del personaje, dando al usuario la sensación de ver a través de los ojos del mismo. Además, la cámara principal está acoplada a la cabeza del personaje para que su movimiento sea solidario. La segunda cámara se sitúa alineada con el visor, en la posición donde habría que situar los ojos en la realidad para apuntar con el mismo.

En el momento en que el usuario realiza la acción asignada a la entrada que corresponde con la acción de apuntar, la cámara principal se desactiva y permite la activación de la cámara secundaria (ADSCamera). Además, mientras el personaje apunta, se limita su velocidad de movimiento para aumentar la precisión en el tiro. Una vez el usuario deja de apuntar, las cámaras vuelven a su estado original, volviendo a ver desde los ojos del personaje y reestableciendo la velocidad máxima de movimiento del personaje.

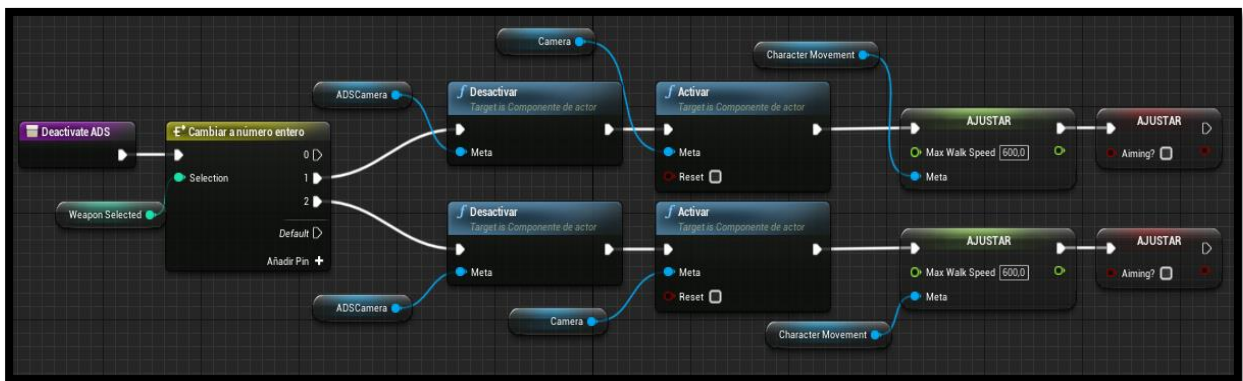


Figura 3-46 Función desactivar ADS

Para entender mejor este proceso se muestra en la Figura 3-46 la función empleada para desactivar la “ADSCamera” una vez se deja de apuntar.

3.4.2.6 Disparar

La finalidad principal del presente proyecto es poder disparar con un sistema de armas sobre un blanco y obtener los resultados que nos permitan determinar si el ejercicio en cuestión se ha realizado de manera satisfactoria.

Para ello se ha creado una función que permite al usuario abrir fuego realizando una acción que active la entrada de disparo tras la cual se ejecute una secuencia de código que genere un proyectil en la boca del cañón del arma, disminuya la munición del arma y produzca un retroceso similar al que se daría en la realidad debido a la deflagración de la pólvora.

La primera parte del código, como se muestra en la Figura 3-47, comprueba que el personaje no esté corriendo o recargando, en cuyo caso no se podrá hacer fuego, además de ver si queda munición en el cargador. De no ser así, activará la variable “IsFiring” de tipo “Boolean” que determinará que el personaje está haciendo fuego en ese momento. A la vez, determina si se está disparando con el arma primaria o secundaria.

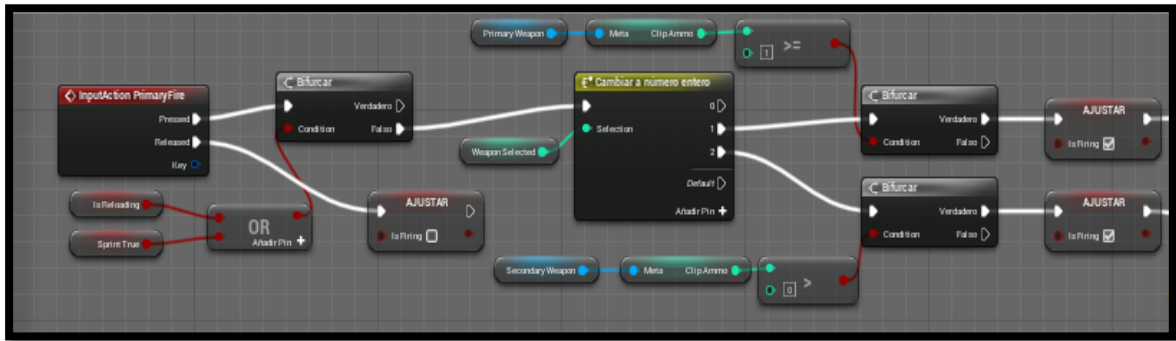


Figura 3-47 Función Disparar (1)

La segunda parte de la secuencia de código que puede verse en la Figura 3-48 activa la función “Fire” dentro del Blueprint del arma y ejecuta posteriormente una función de espera en base a la cadencia establecida para el arma que se esté disparando. A continuación, hace una llamada a la función que produce el retroceso que más adelante veremos. El último nodo de bifurcación puede unirse al comienzo de la secuencia para activar el disparo en modo automático hasta que la munición del cargador se termine y el usuario recargue el arma.

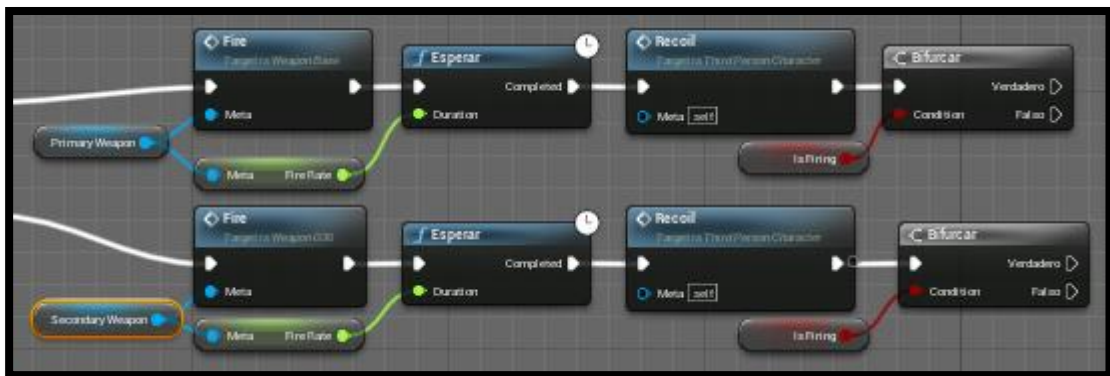


Figura 3-48 Función Disparar (2)

3.4.2.7 Recargar

Dado que en la realidad no se dispone de munición infinita, además, esta va distribuida en varios cargadores; se ha establecido un sistema de munición que dota al simulador de un mayor realismo.

Se ha establecido como valor predeterminado para la munición total la cantidad que suele llevar un fusilero en condiciones normales. Esto equivale a un total de 210 cartuchos distribuidos en cargadores de 30. Una vez se consumen estos 30 cartuchos, el usuario debe recargar para poder seguir haciendo fuego. Además, si el usuario recarga antes de terminar la munición de un cargador, está no se guardará si no que se desechará junto con su cargador. De este modo el usuario deberá evaluar la situación antes de decidir si recargar o no antes de consumir la munición de un cargador.

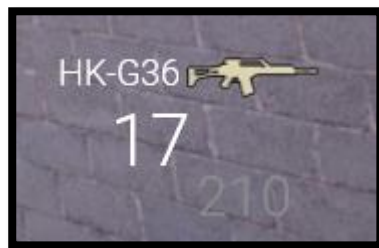


Figura 3-49 Widget de munición

En la Figura 3-49 se aprecia cómo se presenta en pantalla la munición total de la que dispone el personaje (en gris en la esquina inferior derecha) y la munición del cargador actual (en blanco). Además, se indica en la parte superior el sistema de armas que se está empleando en dicho momento.

En el proceso de recarga se ha introducido una animación que hace que el personaje realice el movimiento propio de la recarga. En la Figura 3-50 se muestra el código de la función Recargar en la que se introduce un nodo de espera de 3,3 segundos, tiempo en el cual el personaje realiza la animación de la recarga y durante el cual no se podrá hacer fuego ni apuntar con el arma.

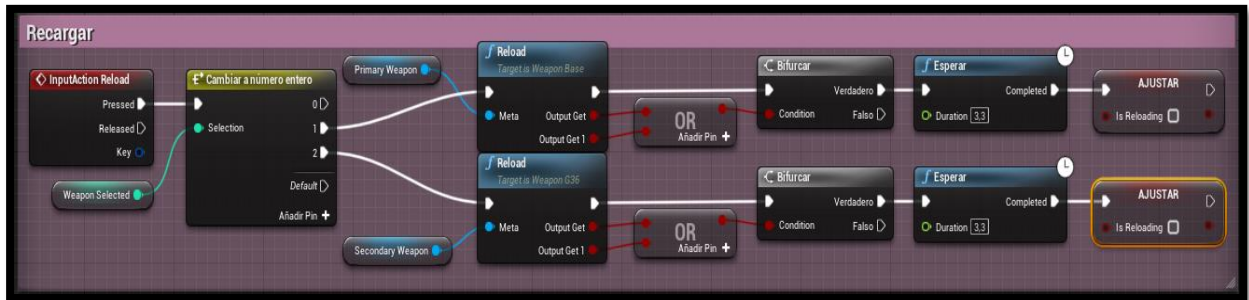


Figura 3-50 Función Recargar

3.4.2.8 Retroceso

Cuando se dispara con un arma de fuego, la deflagración de la pólvora permite impulsar el proyectil y que este se desplace hasta el blanco. A su vez esto provoca una fuerza contraria al movimiento del proyectil y que se conoce comúnmente como retroceso. Las armas están diseñadas para minimizar su efecto y, de esta manera, facilitar su empleo. De todos modos, es físicamente imposible eliminar por completo el retroceso de un arma.

El sistema de armas empleado en el simulador tiene también un determinado retroceso. De cara a que este pueda reproducirse de manera realista en el simulador, cuando el usuario abre fuego con el personaje, este último sufre un movimiento moderado de la boca del arma hacia arriba y un ligero movimiento lateral que simula la inestabilidad del tirador al abrir fuego.



Figura 3-51 Función Retroceso

En la Figura 3-51 se muestra la función Retroceso a la cual se hace referencia desde la función disparar. Cuando se ejecuta esta función, el ángulo de elevación del personaje se ve modificado de manera aleatoria dentro de un rango establecido por el desarrollador. Del mismo modo, el ángulo de guiñada sufre una variación aleatoria hacia uno u otro lado.

3.4.3 Sistema de Armas

En este subapartado se expone el código que rige el comportamiento del sistema de armas empleado por el personaje durante la simulación. Aunque este se compone de un Blueprint propio que ahora

veremos, la gran mayoría de las acciones relativas al empleo del arma se ejecutan desde el Blueprint ThirdPersonCharacter.

Dado que el Blueprint del arma, denominado “Weapon_G36”, está formado por un único componente de malla estática, no es necesario entrar en detalle en su composición. A esta malla estática se le ha asignado la malla de esqueleto del HK-G36E importada en el motor de juego.

3.4.3.1 Disparar

Como vimos en la función Disparar del ThirdPersonCharacter, en ella se hacía una llamada a la función “Fire” ejecutada dentro del Blueprint del arma como se puede ver en la Figura 3-52.

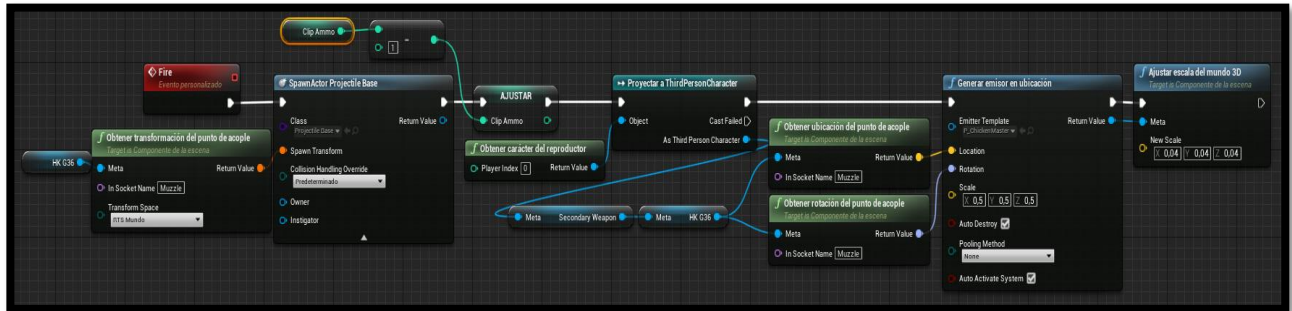


Figura 3-52 Función Fire

En esta función se realizan tres acciones principales. En primer lugar, se genera un proyectil en la boca del cañón del arma mediante el Socket creado previamente con el nombre de Muzzle. A continuación, se resta un cartucho a la munición del cargador del arma. Por último, se genera un emisor también en la boca del arma que simula un pequeño fognazo una vez se hace fuego.

Una vez disparado, el proyectil seguirá una trayectoria determinada por las características asignadas al movimiento del proyectil dentro del Blueprint del mismo que más adelante veremos.

3.4.3.2 Recargar

Del mismo modo que con la función “Fire”, desde el Blueprint del personaje se hace una llamada a la función “Reload” propia del Blueprint del arma. Dentro de esta función se gestiona la munición total del jugador.

Para hacer esta operación se emplean múltiples operadores matemáticos que comprueban en primer lugar si el cargador está lleno y si la munición disponible es mayor a la del cargador. Una vez realizada la acción de recargar, se resta a la munición total la cantidad de munición propia del cargador y en el caso de que la munición disponible sea menor a la que cabe en un cargador, se asigna al cargador la munición restante.

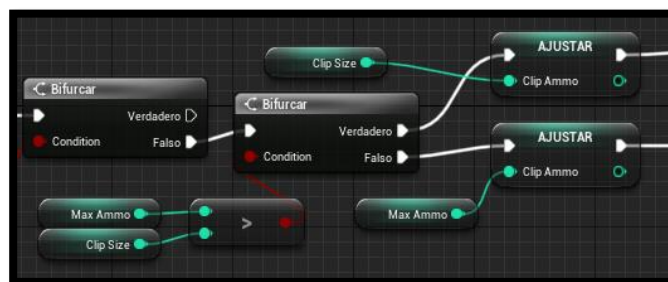


Figura 3-53 Fragmento de la función Reload

En la Figura 3-53 se muestra un fragmento de dicha función con algunos de los operadores matemáticos empleados en la gestión de la munición.

3.4.4 *Proyectil*

Igual que el resto de actores, el proyectil posee un Blueprint propio al que llamaremos “Projectile_Base”, en el que se diferenciará entre varios componentes y funcionalidades propias que tiene el proyectil para que la simulación funcione correctamente. En este subapartado se explicarán estos componentes y cómo se emplean para obtener los resultados deseados durante la simulación.

3.4.4.1 *Componentes del proyectil*

Aunque algo más simple que el personaje, el proyectil empleado en la simulación también se divide en varios componentes que tienen una determinada función dentro de la simulación. En la Figura 3-54 se puede ver la Ventana Gráfica del Blueprint del proyectil junto con la lista jerarquizada de sus componentes.

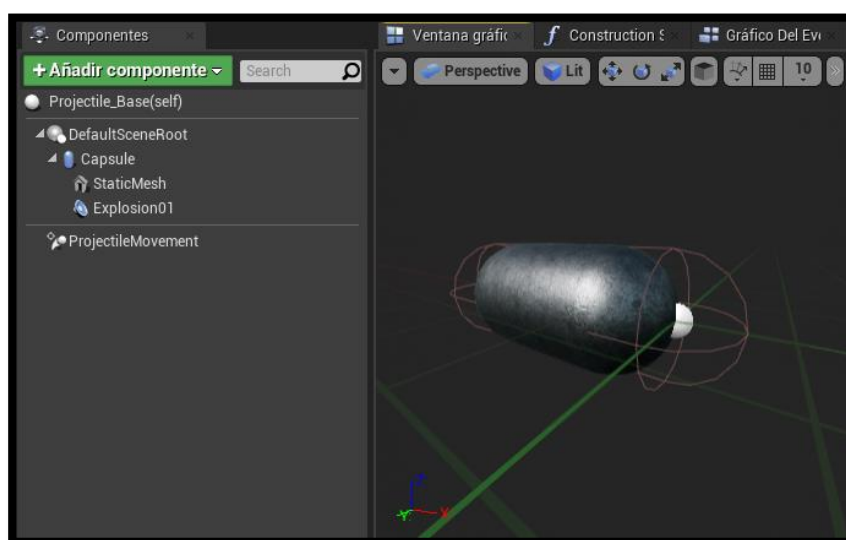


Figura 3-54 Componentes del proyectil

En primer lugar y englobando a los demás está la cápsula de colisión del proyectil, ajustada a sus dimensiones, pero con un pequeño aumento de su dimensión longitudinal. Esto se ha hecho ya que durante el movimiento de los objetos en Unreal Engine se realiza una interpolación que va actualizando su posición. Debido a su tamaño es posible que en esta interpolación queden determinados espacios sin cubrir y por los que el motor entiende que el proyectil no ha pasado. La prolongación de esta cápsula de colisión disminuye la posibilidad de que se de este error durante la simulación.

La cápsula de colisión permitirá obtener los resultados del impacto del proyectil con los blancos mediante el evento de comienzo de superposición como veremos en la programación de los diferentes tipos de blancos.

Subordinados a la cápsula de colisión se encuentran la malla estática del proyectil a la cual se le ha asignado la forma de una cápsula incluida en la biblioteca de contenido inicial de Unreal Engine y un material de aspecto para darle una apariencia más realista, y un componente de audio que simula una explosión. Este componente se reproducirá en la simulación en la localización en la que se genera el proyectil.

Por último, se ha añadido al proyectil un componente de movimiento de proyectil denominado “ProjectileMovement” dentro del cual Unreal Engine ofrece diversas posibilidades con las que modificar el comportamiento del proyectil una vez es generado. Entre otras se han incluido la velocidad de salida y máxima reales del proyectil del HK-G36E (920 m/s), un componente de gravedad que afecta al proyectil a lo largo de su trayectoria. Además, se ha disminuido el intervalo de tiempos entre los que el motor de juego actualiza la localización del proyectil.

3.4.4.2 Generar agujeros de bala

Cuando un proyectil impacta con cualquier superficie deja una marca o atraviesa la misma. En la simulación se han implementado varias texturas que simulan agujeros de bala y que permanecen en la superficie de cualquier objeto del escenario una vez el proyectil impacta con las mismas.

Para ello se ha desarrollado una función dentro del Blueprint del proyectil que calcula la trayectoria que tendrá este una vez es generado y donde esta trayectoria intersece con algún objeto, se generará un agujero de bala con una de las texturas importadas al proyecto.

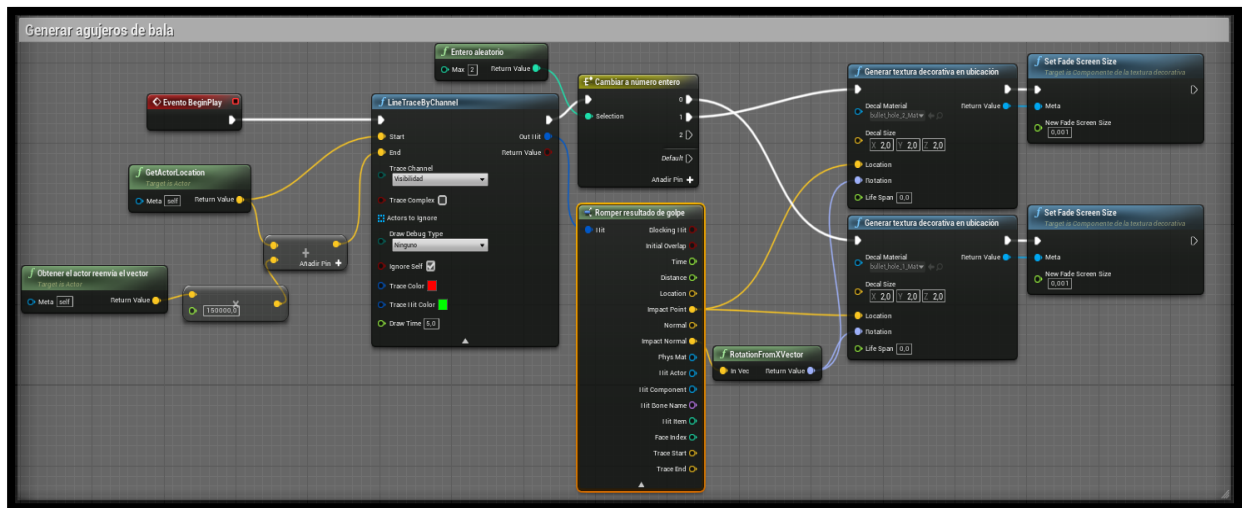


Figura 3-55 Función Generar agujeros de bala

En la función que aparece en la Figura 3-55 se emplea la función predeterminada “Line Trace by Channel” que calcula la trayectoria del proyectil y obtiene la localización del impacto de la misma. En esta localización

3.4.5 Blancos

Dentro de los escenarios, los elementos diferenciales y que permitirán la realización de los diferentes ejercicios de tiro del programa de tiro seleccionado son los blancos. Para ello se ha creado un Blueprint para cada tipo de blanco con el fin de obtener los resultados del ejercicio de tiro realizado por el usuario y así determinar si un ejercicio se ha superado de manera satisfactoria o no.

3.4.5.1 Sistema de puntuación

Conforme se realizan los disparos durante la simulación, el usuario podrá ver en pantalla su puntuación, en la que se indicará el número de impactos acertados en cada tipo de ejercicio.

Para los blancos silueta y los blancos de incapacitación se ha generado un código a partir de sus componentes que guarda el resultado dentro de una variable de tipo “Integer” a la que hemos llamado “Points”, dentro del Blueprint del personaje. Para saber si el proyectil ha impactado en el blanco o no, en sus Ventanas Gráficas se han añadido cajas de colisión, como se puede observar en la Figura 3-56, a partir de las cuales se ha generado un evento de superposición para que, cuando el proyectil entre en dichas cajas, contabilice el disparo como acertado.



Figura 3-56 Blanco silueta

El código generado tras el evento de superposición es el que se muestra en la Figura 3-57. En él se especifica que únicamente la superposición del proyectil con la silueta sea lo que contabiliza un punto en el sistema de puntuación, empleando operadores matemáticos que incrementen la puntuación de uno en uno.

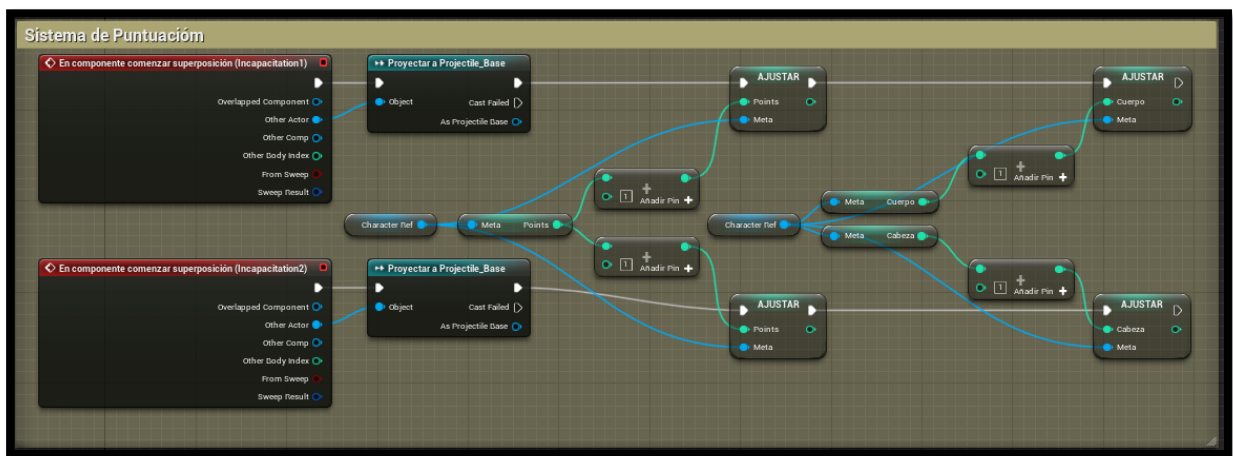


Figura 3-57 Función Sistema de puntuación

En el caso de los blancos de precisión, se han generado múltiples esferas de colisión para poder diferenciar si un impacto ha sido en una u otra zona, sumando los puntos de cada zona respectivamente cuando el proyectil impacta sobre esta. En la Figura 3-58 se puede observar la disposición de estas esferas de colisión.

Además de incrementar la puntuación general, se ha creado una variable para cada zona del blanco de precisión, cuyo resultado aumenta de uno en uno de tal forma que se sepa al final de la simulación no solo la puntuación total, sino también el número de impactos en cada zona.

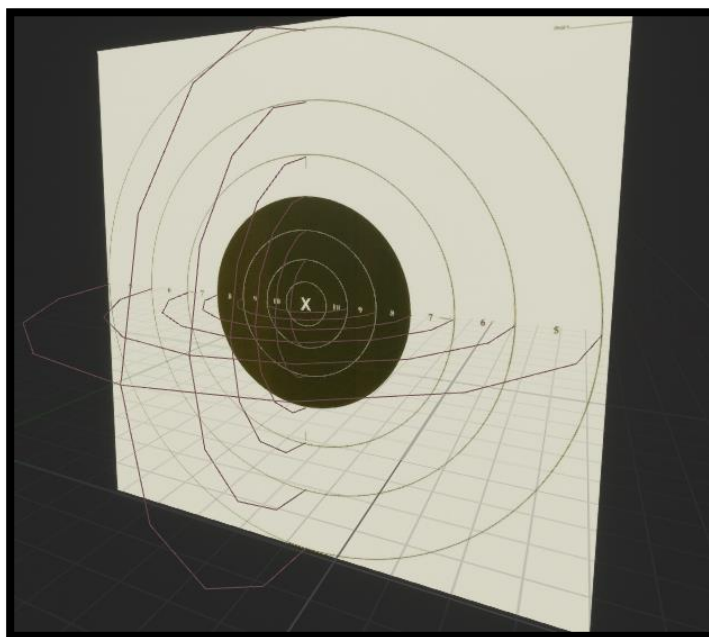


Figura 3-58 Blanco de precisión.

3.4.5.2 Aparición y caída de silueta

En el escenario del poblado afgano las siluetas se encuentran inicialmente escondidas y una vez el usuario se aproxima a su localización, estas se levantan. Una vez el usuario abre fuego e impacta sobre la silueta del blanco, este cae y se destruye.

Para determinar el momento en que el blanco ha de levantarse se ha generado un volumen de control en cuyo Blueprint se ha introducido la modificación de la posición del blanco una vez el personaje entra dentro de dicho volumen. Todos los blancos que se encuentren dentro de un mismo volumen se levantarán al mismo tiempo mientras que dos blancos dentro de dos volúmenes diferentes se levantarán cuando el personaje entre dentro de su correspondiente volumen. De este modo se puede secuenciar la aparición de los blancos conforme el usuario progresa por las diferentes edificaciones y estancias del poblado afgano.

3.4.5.3 Movimiento

En determinados ejercicios de tiro se emplean blancos silueta con la capacidad de desplazarse lateralmente. Para ello, dentro del Blueprint de este tipo de blanco se ha introducido, aparte del mismo sistema de puntuación expuesto en el anterior subapartado, una secuencia de código que permite a la silueta trasladarse lateralmente desde un extremo al otro de cada línea de tiro o, si el instructor lo considerase necesario, desde un lado a otro de la galería.

Para lograr este movimiento se han creado dos variables de tipo “Vector” para las posiciones inicial y final de la silueta. La posición inicial se obtiene mediante la función “Get Actor Location”, la cual nos devuelve como resultado un vector con las coordenadas de posición de la misma dentro del escenario. Las coordenadas en altura y profundidad se mantienen constantes para la posición final de la silueta.

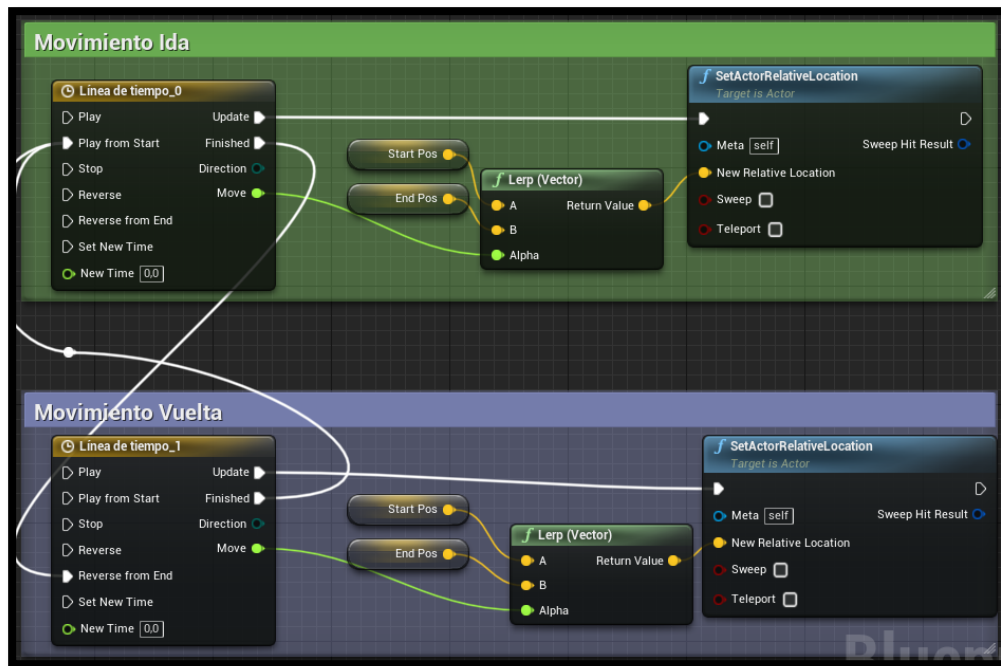


Figura 3-59 Función de movimiento de silueta

En las funciones de ida y vuelta de la silueta mostradas en la Figura 3-59 se ve cómo, mediante una línea de tiempo se establece el movimiento constante que, una vez finalizado en un sentido, comienza en el sentido contrario. Dentro de estas líneas de tiempo se establece la velocidad de movimiento de las siluetas, siendo esta la velocidad media a la que anda una persona, es decir, 5 km/h.

3.4.6 HUD

Los HUD permiten a la simulación proporcionar al usuario información útil en pantalla mientras se desarrolla la misma. Además, son la manera de crear una interfaz navegable a modo de menú principal del simulador, así como pantalla de resultados una vez finalice la simulación.

Las HUD se implementan haciendo una llamada al Widget Blueprint que se quiera emplear dentro del mismo. De este modo se han desarrollado tres Blueprints para las diferentes situaciones en las que puede encontrarse el usuario durante la simulación: Navegación en el Menú Principal (“MainMenu”), simulación dentro de un Escenario (“FPSHUD”) o pantalla de resultados (“EndGame”).

3.4.6.1 Menú Principal

Desde el menú principal el usuario podrá seleccionar el escenario al que quiere acceder para comenzar la simulación, acceder a un apartado de opciones que se implementarán dentro del simulador y, por último, cerrar el simulador haciendo clic en los diferentes botones que se muestran en pantalla.



Figura 3-60 Menú Principal

La Figura 3-60 muestra este menú principal con los diferentes botones que permiten la navegación dentro del mismo. Una vez se hace clic en uno de los botones, se ajusta la visibilidad de los mismos mediante dentro del Gráfico del Evento del propio Blueprint como se puede observar en la Figura 3-61 en el caso del botón “Seleccionar Escenario”.

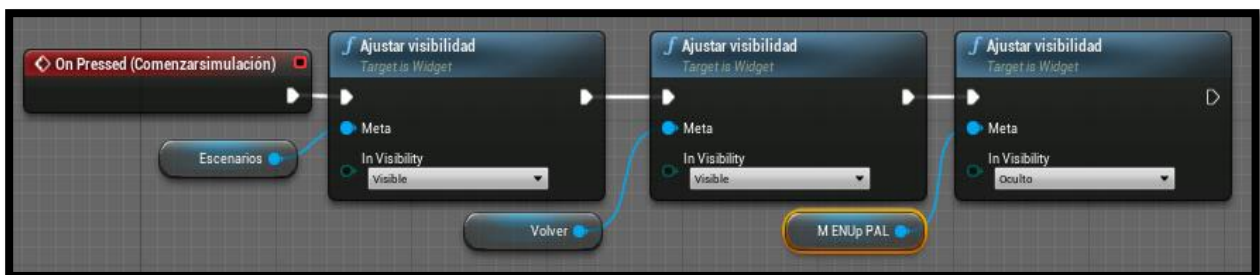


Figura 3-61 Botón Seleccionar Escenario

Desde los botones del menú principal se pueden ajustar parámetros dentro del juego como la munición disponible del personaje o el tiempo de simulación, además de cargar cualquiera de los niveles cuando se hace clic sobre el botón de uno de ellos. En las Figura 3-62 y 3-61 podemos ver un ejemplo de cada caso.

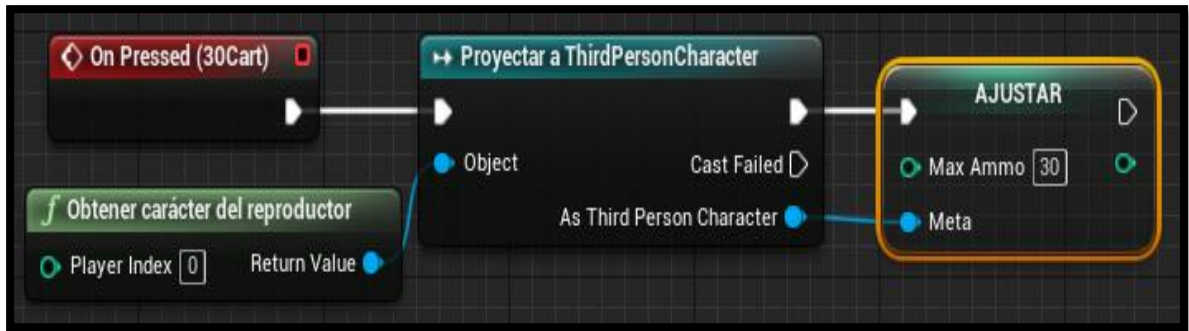


Figura 3-62 Ajustar munición

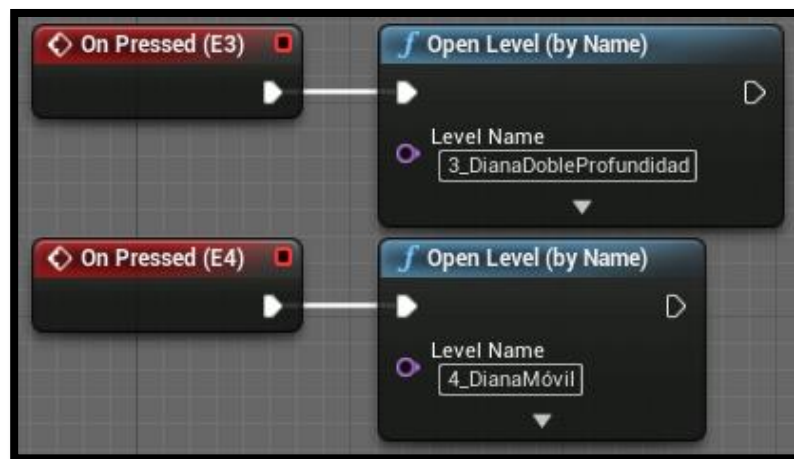


Figura 3-63 Cargar Escenario

3.4.6.2 FPS HUD

Durante la simulación, el usuario dispondrá en pantalla de la siguiente información:

- Objetivo dentro del escenario
- Puntuación actual
- Munición restante en el cargador
- Munición total
- Sistema de armas seleccionado
- Imagen en tiempo real del blanco sobre el que se hace fuego.

De manera adicional se ha añadido una barra de vida funcional que podría emplearse en futuras simulaciones en las que el personaje pueda recibir daño de disparos durante situaciones de combate real.

Para obtener todos los datos que se mostrarán en pantalla, se hace referencia al Blueprint del personaje desde el Gráfico del Evento del Widget Blueprint y se obtienen los valores relativos a la puntuación, salud, munición...

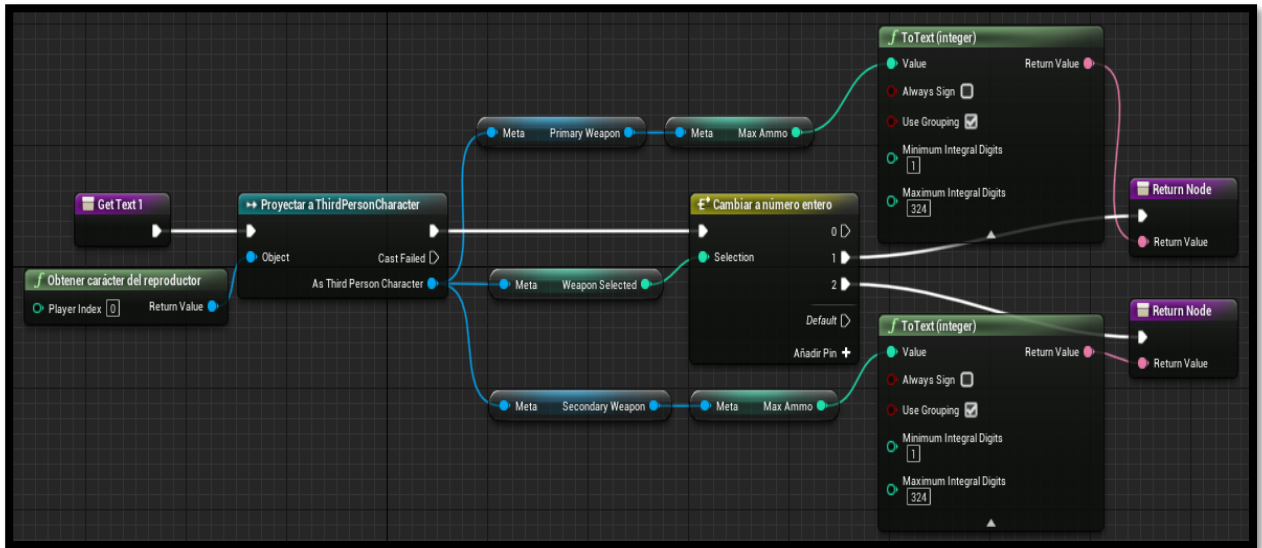


Figura 3-64 Obtener valor de la munición

En la Figura 3-64 se muestra la secuencia de código empleada para obtener el valor de la munición total almacenada dentro de una variable de tipo “integer” en el “ThirdPersonCharacter”. Este valor se transforma a un formato de texto que pueda mostrarse en la pantalla.

Para obtener la imagen en tiempo real de los blancos sobre los que se está haciendo fuego se ha situado una cámara en frente de cada uno de ellos dentro de la ventana gráfica de los escenarios. La imagen que se ve a través de estas cámaras es renderizada y convertida en una textura a la que posteriormente se puede hacer referencia desde el Widget Blueprint. Esta textura se irá actualizando conforme se desarrolla la simulación, permitiendo al usuario ver los impactos realizados sobre el blanco.



Figura 3-65 Imagen en tiempo real del blanco

En la Figura 3-65 se observa la disposición de la cámara respecto al blanco sobre el que se hace fuego y que nos dará la imagen del mismo en tiempo real.

3.4.6.3 Pantalla de resultados

Una vez finaliza la simulación dentro de uno de los escenarios, se mostrará en pantalla los resultados obtenidos por el usuario durante el ejercicio realizado.

Estos incluirán la situación final de los blancos con todos los impactos sobre los mismos, lo que permitirá evaluar la agrupación y precisión del usuario, así como la cantidad de disparos acertados, que a su vez estarán divididos en impactos en el cuerpo del blanco o en la cabeza. En la Figura 3-66 se muestra una pantalla de resultados tras la realización de un ejercicio.



Figura 3-66 Pantalla de resultados

Desde esta pantalla se puede cargar de nuevo el mismo nivel o cargar el menú principal del mismo modo que se hacía desde el Menú Principal. Los resultados se obtienen a su vez de la variable del sistema de puntuación del “ThirdPersonCharacter”

De manera adicional, una vez se genera esta pantalla, los resultados se guardan de forma permanente en la memoria interna del ordenador mediante la secuencia de código que se muestra en la Figura 3-67. En esta secuencia se emplea un nodo que ejecuta el comando “bugsscreenshot” que realiza una captura de la pantalla del simulador, siendo esta la de resultados en el momento de la captura. De este modo el evaluador puede tener acceso a los resultados de todos los ejercicios realizados dentro del simulador cuando lo requiera.

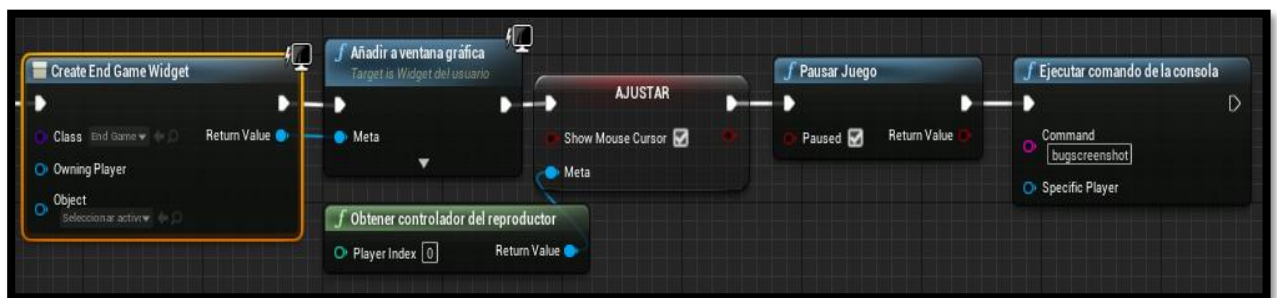


Figura 3-67 Generar y guardar pantalla de resultados

3.5 Realidad Virtual

Como se expone en el título del presente proyecto, la finalidad del simulador es el adiestramiento con arma de tiro en un entorno de Realidad Virtual. Para ello se requiere un simulador que permita la inmersión del usuario dentro de un escenario en 3D con el que pueda interactuar y por el que pueda desplazarse de una manera lo más parecida a la realidad posible

Hasta el momento, todo el diseño y programación desarrollados por medio de Unreal Engine se han realizado sobre la base de un proyecto de tipo “Third Person”. Para poder implementarlo en realidad virtual, Unreal Engine facilita los medios para generar un proyecto de tipo “Virtual Reality” en el que se podrán implementar los controles necesarios para la inmersión en este tipo de entorno.

3.5.1 Creación de BOLTRIX RV

El simulador 3D con tecnología i4.0 de Realidad Virtual para adiestramiento de tiro con arma de fuego desarrollado en este proyecto tendrá el nombre de “BOLTRIX RV”. La creación el mismo se ha llevado a cabo a partir del proyecto en tercera persona sobre el que se ha trabajado hasta el momento.

En primer lugar, se ha generado una copia del mismo, dándole el nombre establecido. A continuación, se ha añadido el contenido propio de los proyectos de realidad virtual de Unreal Engine añadiendo al proyecto los ficheros que permitirán la ejecución del proyecto en Realidad Virtual. Como la configuración del proyecto sigue siendo la de uno en Tercera Persona, se ha creado un proyecto nuevo en Realidad Virtual. Los archivos de configuración que Unreal Engine genera de manera automática en la carpeta Config del nuevo proyecto se han copiado dentro de las opciones de configuración de BOLTRIX RV para, de esta forma, habilitar la ejecución del mismo en Realidad Virtual.

Una vez realizado todo lo anterior, el proyecto ya está listo y el entorno virtual generado. Lo único que queda variar con respecto al proyecto en tercera persona son las entradas para que el personaje realice las diferentes acciones que puede llevar a cabo ya que originalmente estaban diseñadas para realizarse por medio de un teclado y un ratón. El dispositivo Lenovo Explorer incluye dos Motion Controllers con diversos botones que se han programado para realizar las diversas acciones del personaje, sustituyendo de este modo al teclado y al ratón.

Por último, podemos observar que en la Barra de Herramientas no aparece la posibilidad de arrancar el simulador en modo RV. Para ello habrá que añadir, en la ventana Editar, el plugin de Realidad Virtual que permiten el empleo del equipo de RV y su ejecución desde el motor de juego. Una vez instalado el plugin, basta reiniciar el proyecto y éste quedará listo para su uso.

3.6 Blog

La naturaleza de este tipo de proyecto hace que sea extremadamente difícil al lector de esta memoria hacerse una idea correcta y adecuada del funcionamiento del producto final obtenido tras el desarrollo del proyecto. Tratar de plasmar sobre el papel el funcionamiento de un simulador en un entorno virtual es prácticamente imposible por lo que se ha decidido crear un blog a través de la plataforma web “Blogger” con el objetivo de mostrar, de una manera más precisa y detallada el funcionamiento del simulador.

De este modo se ha creado el blog con el nombre del simulador, BOLTRIX RV, donde se incluirán videos explicativos de los diferentes escenarios y su funcionamiento de tal forma que se muestren las diferentes aplicaciones que puede tener el simulador, además de servir como referencia para futuros proyectos dentro del mismo ámbito.

En la Figura 3-68 se muestra la interfaz principal de “Blogger” en la que se ha creado ya el Blog y se han introducido las entradas. Esta interfaz nos permite modificar el diseño del Blog, así como la disposición de las entradas y muchas opciones de configuración para lograr un blog único y totalmente personalizado.

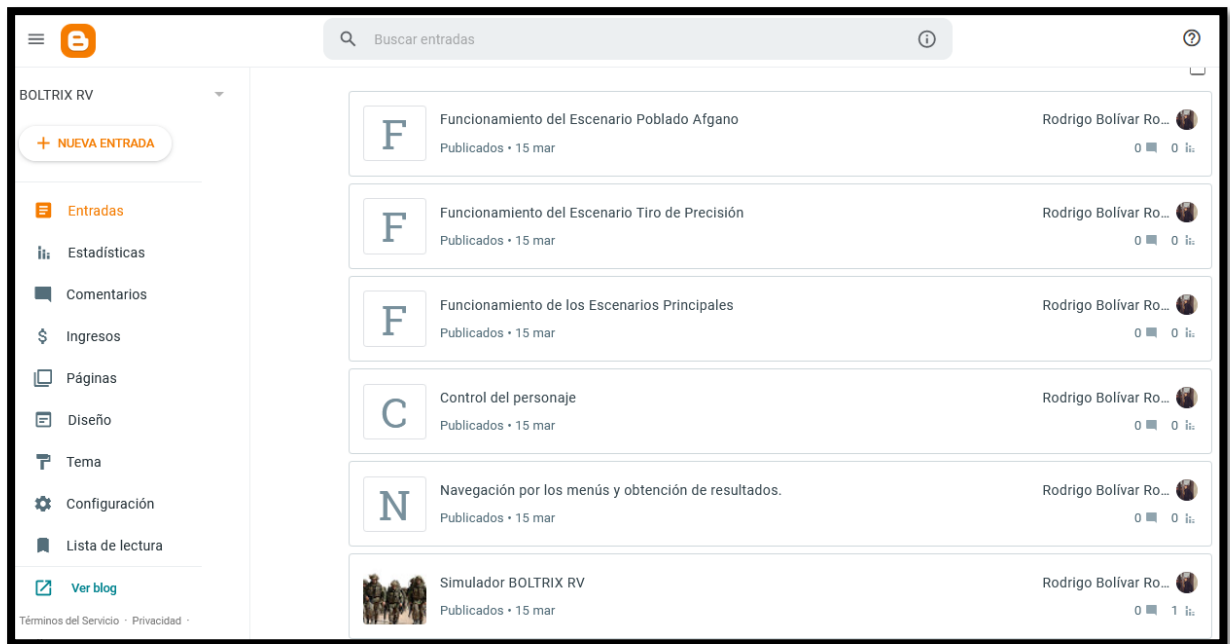


Figura 3-68 Edición del blog

Para acceder al blog “[BOLTRIX RV](https://boltrixrv.blogspot.com/)” basta con hacer Ctrl+Clic sobre el título del blog subrayado anteriormente, o introducir en el motor de búsqueda de internet el siguiente enlace: <https://boltrixrv.blogspot.com/>. Por último, se ha generado un código QR que permite el acceso directo al blog mediante un escaneado del mismo con la cámara de cualquier dispositivo móvil.



Figura 3-69 Código QR del Blog

4 RESULTADOS

4.1 Descripción del apartado

4.1.1 Descripción del apartado

Tras el desarrollo del presente proyecto, se presentarán en este apartado los frutos de todo lo anteriormente expuesto. Para ello se exponen los resultados siguiendo un esquema similar al del apartado anterior, siguiendo los diferentes pasos de desarrollo del proyecto.

4.1.2 Escenario y aplicación

En cuanto a la selección del escenario que se implementa en BOLTRIX RV se ha decidido generar una representación 3D de una localización real que permita al usuario del simulador reconocerla para favorecer de este modo la inmersión dentro del entorno virtual. Además, la posibilidad de realizar los mismos ejercicios de tiro que se realizan dentro del programa de tiro de fusil de la Brigada de Infantería de Marina, obteniendo resultados útiles para el evaluador, hacen de BOLTRIX RV un buen complemento y/o alternativa para el adiestramiento del combatiente con arma de tiro en escenarios reales.

Además, se lleva a cabo el diseño de dos escenarios ficticios alternativos. Uno en el que se representa un poblado afgano y otro en el que se permite la realización de ejercicios de tiro de precisión. Estos amplían las posibles aplicaciones de BOLTRIX RV, permitiendo al usuario adiestrarse en diferentes tipos de tiro o en diferentes situaciones que pueden darse en la vida real.

4.1.3 Diseño y modelado

En el presente proyecto el diseño y modelado de los escenarios juega un papel de gran relevancia para la inmersión realista del usuario dentro del simulador. Como resultados dentro de esta parte del proyecto se han obtenido los diferentes escenarios con sus diversas posibilidades.

El primero de ellos, y sobre el que se han generado los escenarios E1 a E5, consiste en una representación 3D de la galería de tiro situada en la Base Militar General Morillo. En su modelado se han permitido algunas licencias como la reducción de la longitud para adecuarla a los ejercicios que se pretende realizar con BOLTRIX RV, o la eliminación de la antesala de la galería, cuya presencia no reportaría ningún beneficio al simulador, sino que por el contrario únicamente ralentizaría su funcionamiento.

Este escenario permite llevar a cabo la totalidad de los ejercicios propuestos en el programa de tiro de fusil de la Brigada de Infantería de Marina publicado en 2019.



Figura 4-1 Detalles del escenario principal

Dentro de este escenario se han implementado los diferentes blancos que se han generado, y cuyas funcionalidades son esenciales para cumplir con el objetivo del simulador. Estos son el blanco de tipo silueta, el blanco de incapacitación y el blanco móvil. Además, todos están concetados con el sistema de puntuación que puede observarse durante la simulación y, una vez finalizada esta, en la pantalla de resultados.

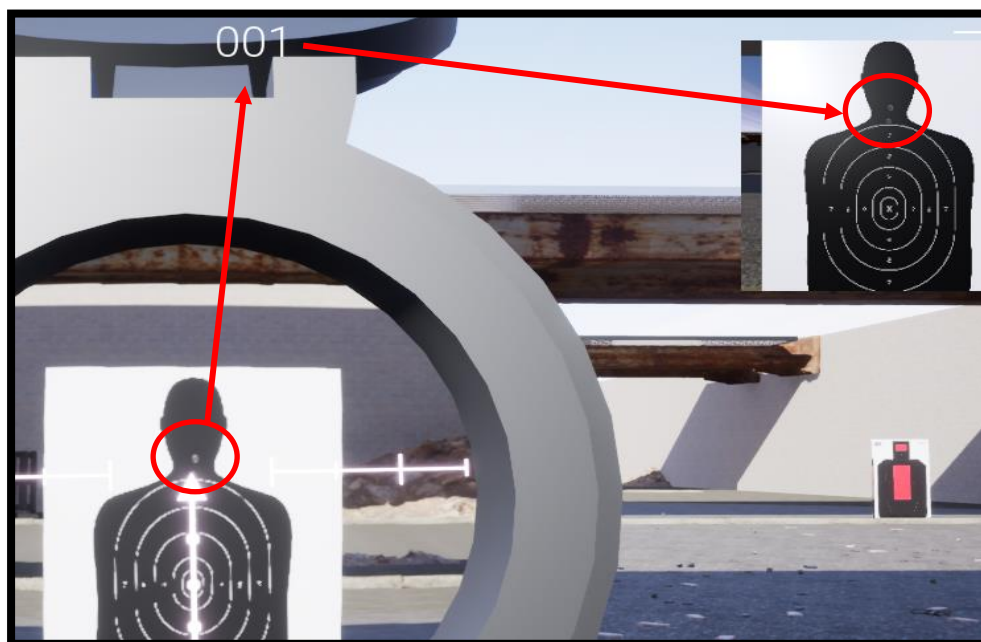


Figura 4-2 Sistema de puntuación

En segundo lugar, se ha creado un escenario para el entrenamiento en el tiro de precisión, tomando como base la galería de los escenarios anteriores. A esta se le ha eliminado el lado opuesto a la línea de tiro, dando esta última a un campo abierto en el que se han situado diversos blancos de precisión a diferentes distancias. Estos blancos además determinan con mayor exactitud el resultado de los impactos de los proyectiles sobre el mismo, diferenciando entre las diferentes zonas del blanco con un sistema de puntuación más sofisticado que los blancos empleados en el escenario anterior.

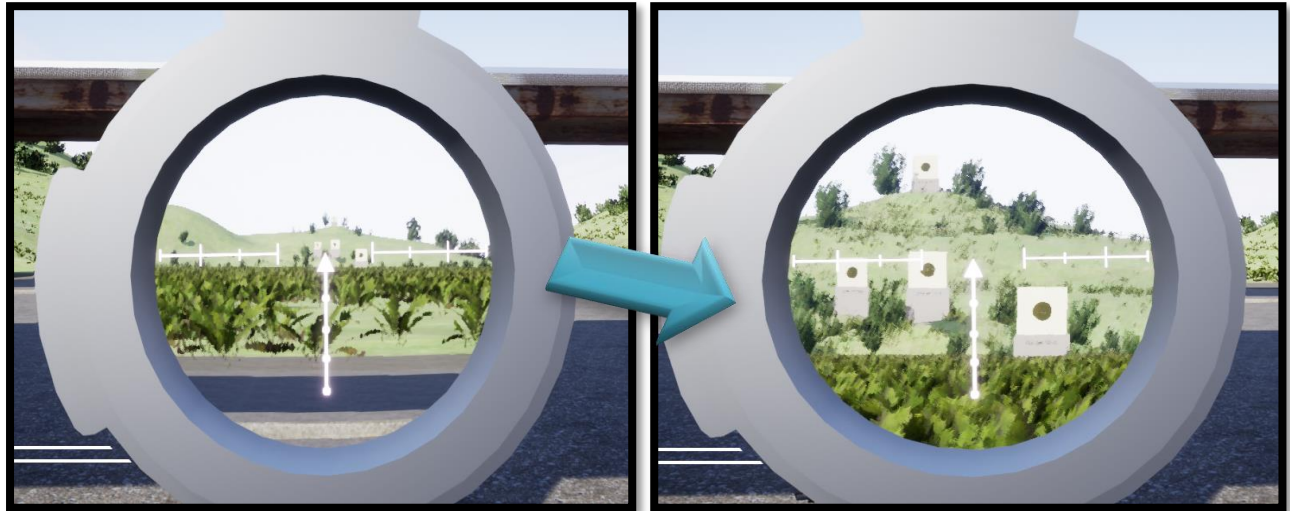


Figura 4-3 Zoom del visor óptico

Dentro de este escenario el usuario puede apuntar y hacer zoom con el visor óptico del fusil, con el fin de adiestrarse en el tiro a largas distancias. Además, en este caso influyen de manera sustancial sobre el tiro factores externos como el viento o la gravedad.



Figura 4-4 Efecto de la gravedad y el viento

Por último, y de manera adicional, se ha generado un escenario totalmente ficticio en el que se ha representado un poblado afgano. En él, el usuario puede desplazarse por diferentes edificaciones y estancias dentro de las mismas y, mientras lo hace, irán surgiendo diferentes blancos que permiten al evaluador medir el tiempo de reacción del usuario del simulador, así como la respuesta ante posibles situaciones de combate reales.



Figura 4-5 Detalles del poblado afgano

Los blancos empleados en este escenario no son visibles para el usuario desde el comienzo de la simulación y, una vez un proyectil impacta sobre ellos, estos caen al suelo y se destruyen, permitiendo al usuario continuar avanzando por el escenario.

4.1.4 Programación y código

Los resultados de la detallada programación desarrollada para el correcto funcionamiento del simulador BOLTRIX RV se hacen tangibles en las diferentes funcionalidades y posibilidades que ofrece el simulador.

A través de su representación virtual dentro del simulador, el usuario puede realizar las siguientes acciones:

- Movimiento libre y realista por el escenario.
- Adoptar la posición de rodilla en tierra y moverse en esta posición para adoptar posiciones de tiro alternativas.
- Saltar por encima de objetos.
- Portar sistemas de armas individuales.

- Abrir fuego con los sistemas de armas y ser capaz de recargar las mismas.
- Apuntar por medio de un visor y hacer zoom con el mismo para ampliar el rango de visión del usuario.
- Impactar e influir en los diferentes blancos.
- Presentar en pantalla la munición restante y el tiempo disponible para la simulación.
- Presentar en pantalla una imagen en tiempo real de los blancos sobre los que se abre fuego.
- Recibir daño para posibles escenarios alternativos.

Además, se ha creado un sistema que conecta el proyectil, los blancos y la puntuación de la simulación. De este modo se puede determinar si el usuario ha acertado sobre el blanco, dónde ha impactado el proyectil y, finalmente, si el ejercicio en cuestión ha sido superado de manera satisfactoria.



Figura 4-6 Sistema de puntuación de precisión

Un aspecto fundamental logrado es la posibilidad de almacenar los resultados de las simulaciones en la memoria del ordenador. Esto permite al evaluador acceder a los resultados cuando lo requiera, así como realizar una evaluación progresiva del usuario en la que se pueda observar la mejoría de este a lo largo de los diferentes ejercicios de tiro.



Figura 4-7 Resultados de la simulación

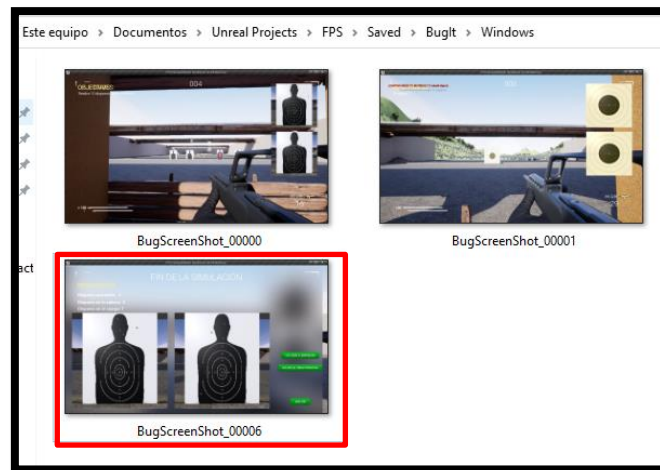


Figura 4-8 Resultados almacenados

4.1.5 Realidad Virtual

Se ha logrado, además de la creación de un entorno virtual para el simulador de tiro BOLTRIX RV, su correcta implementación y su funcionamiento con el dispositivo Lenovo Explorer.

En el simulador se puede emplear el dispositivo de Realidad Virtual para visualizar el escenario, controlar al personaje que representa al usuario en cuanto a desplazamientos y rotación, realizar todas las acciones implementadas y, por tanto, realizar los ejercicios de tiro de manera satisfactoria.

Cabe reseñar que, dada la limitación del cable de conexión del dispositivo de Realidad Virtual, el movimiento del personaje se realiza mediante los joysticks de los Motion Controllers, reduciendo por tanto el realismo de la simulación puesto que el usuario está realmente quieto cuando el personaje se desplaza.

4.1.6 Blog

Se ha generado a través de “Blogger” un blog explicativo del funcionamiento de BOLTRIX RV en aras de un mejor entendimiento de la finalidad que se persigue con el desarrollo del presente proyecto y las posibles aplicaciones del mismo.

Para acceder al blog “[BOLTRIX RV](https://boltrixrv.blogspot.com/)” basta con hacer Ctrl+Clic sobre el título del blog subrayado anteriormente, o introducir en el motor de búsqueda de internet el siguiente enlace: <https://boltrixrv.blogspot.com/>. Además, se ha generado en la plataforma web “GOQR” un código QR a través del cual se puede acceder también al blog. Este nos redirigirá al blog propiamente dicho, el cual tiene una apariencia como la que se puede observar en la Figura 4-9.

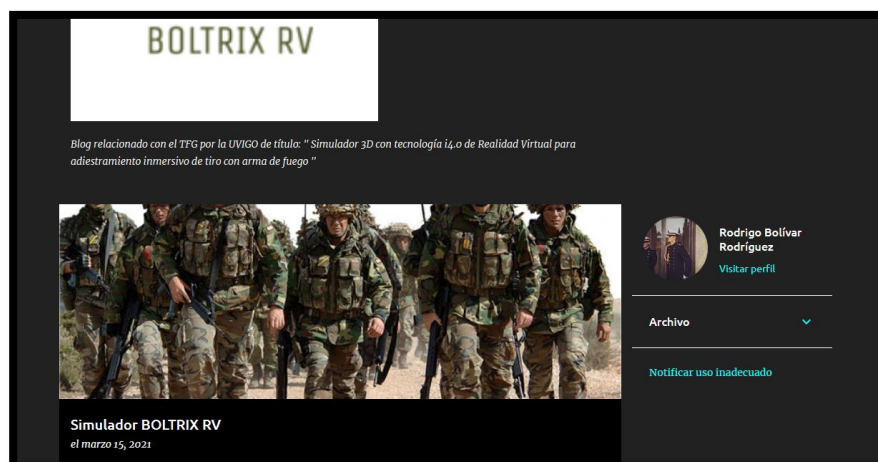


Figura 4-9 Blog BOLTRIX RV

5 CONCLUSIONES Y LÍNEAS FUTURAS

5.1 Descripción del apartado

Este apartado expone las conclusiones obtenidas de las diferentes etapas del desarrollo del presente proyecto. A continuación, se procede a hacer una recapitulación con una conclusión final del proyecto en su totalidad, en la que se determina si se han cumplido de manera satisfactoria los objetivos establecidos al inicio del trabajo. Finalmente se hace una breve reseña de las posibles aplicaciones futuras del proyecto, así como algunas posibilidades de desarrollo que permitirían ampliar el alcance y capacidades del simulador desarrollado.

5.2 Conclusiones

5.2.1 *Escenario y aplicación*

La selección del escenario y el programa de tiro a realizar dentro del simulador surgieron de la necesidad de mostrar la aplicabilidad del simulador a un ejemplo real como puede ser el escogido. Una vez implementado el simulador BOLTRIX RV para la realización de ejercicios de tiro con arma de fuego dentro del programa de tiro de fusil de la Brigada de Infantería de Marina, en una representación 3D de la galería de tiro de Figueirido, puede afirmarse la utilidad del proyecto, pudiendo emplearse para diferentes programas si fuese necesario.

La selección del escenario y el ámbito de aplicación del simulador constituyeron la guía para el resto de fases del proyecto, las cuales tratan de llevar la idea a la realidad, pasando desde el diseño y modelado, por la programación, hasta su implementación final en Realidad Virtual.

5.2.2 *Diseño y modelado*

La fase de diseño y modelado se realizó siguiendo una estructura dividida en escenarios derivados de un escenario principal. Esta decisión constituyó un gran acierto dada la complejidad e ineficiencia que habría supuesto la elaboración de escenarios totalmente independientes, además de permitir la escalabilidad del proyecto, mediante la modificación del escenario principal para obtener otros alternativos empleando diferentes componentes en cada uno.

Además, los elementos del escenario principal fueron empleados en su mayor medida para el diseño de los escenarios adicionales, reduciendo así el uso de recursos del simulador y, por tanto, mejorando la experiencia de uso del mismo.

Finalmente, el diseño de un menú que permita la interconexión entre escenarios y la obtención de resultados tras la simulación logran un uso intuitivo y sencillo del simulador, con los mejores resultados posibles, tanto para el usuario como para el evaluador.

5.2.3 Programación y código

Con el fin de que lo diseñado en la fase previa pudiese cumplir con los objetivos del proyecto se empleó la programación multinodal. Esta probó su eficacia gracias a la posibilidad de programar de una manera visual e intuitiva, en la que se pudo obtener una visión global del proyecto mucho más detallada, permitiendo detectar errores o añadir mejoras a todos los componentes del proyecto en cualquier momento del desarrollo.

Sin los contenidos elaborados durante esta fase no habría sido posible lograr un diseño que cumpliera con los objetivos establecidos al inicio del proyecto.

5.2.4 Realidad Virtual

La capacidad de inmersión del proyecto gracias a la aplicación de la tecnología de Realidad Virtual mediante la creación de un entorno virtual supone un elemento diferenciador en la capacidad de este para lograr el adecuado adiestramiento en el empleo de armas de fuego en todos los posibles escenarios. El usuario puede sentirse “dentro” del simulador y, con ello, llevar a cabo ejercicios de la manera lo más realista posible.

5.2.5 Conclusión final

De manera general, y considerando todo lo mencionado anteriormente, se puede afirmar que los objetivos establecidos al comienzo del proyecto se han alcanzado de manera satisfactoria. Esto quiere decir que se ha logrado el diseño de una plataforma de entrenamiento virtual esférica para la práctica militar de tiro con arma de fuego, útil para el adiestramiento de los miembros de las Fuerzas Armadas de manera complementaria o alternativa a los métodos de adiestramiento de tiro actuales. Se ha realizado un modelado realista del laboratorio de combate que incluye los diversos escenarios y sus componentes que posteriormente se ha implementado en Realidad Virtual, logrando una visualización inmersiva del mismo y la posibilidad de interactuar con los elementos que lo componen.

Además, se han cumplido los objetivos implícitos al objetivo principal del proyecto, así como los objetivos adicionales que surgieron durante el desarrollo del mismo. De este modo se ha logrado la programación de los componentes de los escenarios para que el diseño y modelado pueda cumplir con los objetivos establecidos. Se ha logrado la implementación de un programa de tiro vigente en la actualidad y la realización de todos ejercicios en el presente.

Habiendo concluido el proyecto se puede concluir que el desarrollo de entornos virtuales para el adiestramiento del combatiente en diversos ámbitos es un bien necesario para la adaptación de las Fuerzas Armadas al mundo de la industria 4.0, logrando un adiestramiento de calidad y de bajo coste.

5.3 Líneas futuras

Por último, en este subapartado se exponen las posibles aplicaciones que puede tener el presente proyecto a corto y largo plazo

5.3.1 A corto plazo

En primer lugar, lo primero que se podría llevar a cabo sería la mejora y optimización del código desarrollado en el proyecto, logrando un funcionamiento más fluido y la implementación de nuevas funcionalidades. Además, cabría la posibilidad de diseñar diversos escenarios o ambientaciones para el adiestramiento inmersivo con armas de tiro en diferentes situaciones.

Además, se podría incluir, mediante la adquisición de nuevos equipos de Realidad Virtual, la opción de un modo multijugador en el que varios usuarios puedan realizar el mismo ejercicio dentro de la misma simulación, e incluso interactuar entre ellos en el transcurso de la misma.

Por otro lado, sería recomendable la reproducción de un dispositivo real de tiro al que se le implementasen los Motion Controllers, dándole la forma del fusil de asalto empleado en las Fuerzas

Armadas para, de este modo, aumentar la sensación de realismo y con ello la efectividad del adiestramiento del combatiente en el simulador.

5.3.2 A largo plazo

A largo plazo se debería de tratar de fomentar y expandir el empleo de las tecnologías i4.0 de Realidad Virtual, Aumentada y Mixta para e adiestramiento e instrucción de los componentes de las Fuerzas Armadas en diferentes ámbitos.

A raíz de lo anterior, se podría plantear el desarrollo de simuladores enfocados a las diferentes competencias que deben adquirir los miembros de las Fuerzas Armadas durante sus periodos de adiestramiento, como pueden ser simuladores de vuelo, de navegación, simuladores tácticos o de apoyo de fuegos... para lograr la plena integración de las Fuerzas Armadas en el mundo 4.0.

6 BIBLIOGRAFÍA

- [1] U. d. Deusto, *Industria 4.0: la 4ª revolución industrial*, 2015.
- [2] EVICITI, «¡Bienvenido a la Industria 4.0, la Cuarta Revolución Industrial!,» [En línea]. Available: <https://www.eviciti.com.mx/blog-post/bienvenido-a-la-industria-4-0-la-cuarta-revolucion-industrial/>. [Último acceso: 23 Enero 2021].
- [3] C. A. V. Sales, «Enseñanza Naval y entorno 4.0,» [En línea]. Available: <https://armada.defensa.gob.es/archivo/rgm/2018/08/rgm080918cap14.pdf>. [Último acceso: 23 Enero 2021].
- [4] A. Española, «“Universo 4.0. Impacto en las organizaciones”, mesa redonda en la Semana Naval de Madrid,» Armada Española, 27 Septiembre 2018. [En línea]. Available: https://armada.defensa.gob.es/ArmadaPortal/page/Portal/ArmadaEspañola/conocenosnoticias/prefLang-es/00noticias--2018--09--NT-107-UNIVERSO4-SEM-NAV-es?_selectedNodeID=3557085&_pageAction=selectItem. [Último acceso: 23 Enero 2021].
- [5] Ministerio de Defensa, «Portal de Tecnología e Innovación del Ministerio de Defensa,» [En línea]. Available: <https://www.tecnologiaeinnovacion.defensa.gob.es/es-es/Estrategia/HojasDeRuta/Paginas/C4I.aspx>. [Último acceso: 27 Enero 2021].
- [6] Ministerio de Defensa, «Monografías del SOPT - Jornada Tecnológica: Simulación en el ámbito de la Defensa,» Diciembre 2018. [En línea]. Available: https://publicaciones.defensa.gob.es/media/downloadable/files/links/m/o/monografia_sopt_17.pdf. [Último acceso: 17 Enero 2021].
- [7] «InfoDefensa,» 19 Octubre 2019. [En línea]. Available: <https://www.infodefensa.com/es/2019/10/19/noticia-indra-actualizara-simuladores-millones-euros.html>. [Último acceso: 26 Enero 2021].
- [8] Ministerio de Defensa, «Simuladores en las Fuerzas Armadas,» *Revista Española de Defensa*, nº 283, pp. 6-11, Abril 2012.
- [9] B. I. Simulations, «VBS3: Virtual Desktop Training & Simulation Host,» [En línea]. Available: <https://bisimulations.com/products/vbs3>. [Último acceso: 17 Enero 2021].
- [10] Ministerio de Defensa, «Presente y Futuro del Tercio de Armada,» *Boletín de Infantería de Marina*, nº 22, pp. 19-25, Abril 2014.

- [11] Indra, «Indra Company,» [En línea]. Available: https://www.indracompany.com/sites/default/files/indra-simulador_de_tiro_militar-victrix.pdf. [Último acceso: 18 Enero 2021].
- [12] S. A. S. & Solutions, «Inteligencia Artificial: Qué es y por qué es importante,» [En línea]. Available: https://www.sas.com/es_es/insights/analytics/what-is-artificial-intelligence.html#howitworks. [Último acceso: 30 Enero 2021].
- [13] Editeca, «Realidad virtual, aumentada y mixta. Qué son y diferencias.,» [En línea]. Available: <https://editeca.com/realidad-virtual-aumentada-y-mixta-que-son-y-en-que-se-diferencian/>. [Último acceso: 28 Enero 2021].
- [14] «Zero Latency,» [En línea]. Available: <https://www.zerolatencyvr.es/>. [Último acceso: 28 Enero 2021].
- [15] «Pokémon Go Live - AR Photo Contest,» [En línea]. Available: <https://pokemongolive.com/post/arphotocontest/?hl=es>. [Último acceso: 28 Enero 2021].
- [16] M. d. Defensa, «Guía Docente Enseñanza de Formación para el acceso a la escala de tropa del CGET,» [En línea]. Available: https://ejercito.defensa.gob.es/Galerias/Imagenes/unidades/Caceres/cefot1/Batallon/GUIA_DOCENTE_CURSO_DE_EFM_PARA_TROPA_DEL_CGET.pdf. [Último acceso: 11 Febrero 2021].
- [17] A. E. B. O. d. Estado, «Boletín Oficial del Estado,» Agosto7 2019. [En línea]. Available: <https://www.boe.es/boe/dias/2019/08/07/pdfs/BOE-A-2019-11570.pdf>. [Último acceso: 11 Febrero 2021].
- [18] M. d. Defensa, «Plan de Estudios Oficiales de la Armada sin titulación previa,» [En línea]. Available: <https://www.defensa.gob.es/Galerias/ministerio/organigramadocs/subdef/digerem/plan-estudios-Oficiales-Armada-Sin-titulacion-previa.pdf>. [Último acceso: 11 Febrero 2021].
- [19] «INV Protools,» [En línea]. Available: <https://invprotools.com/product-category/gun-range-targets-and-backers/paper-targets/>. [Último acceso: 24 03 2021].

ANEXO I: EJERCICIOS DE TIRO DE FUSIL DE LA BRIGADA DE INFANTERÍA DE MARINA

CONFIDENCIAL

CONFIDENCIAL

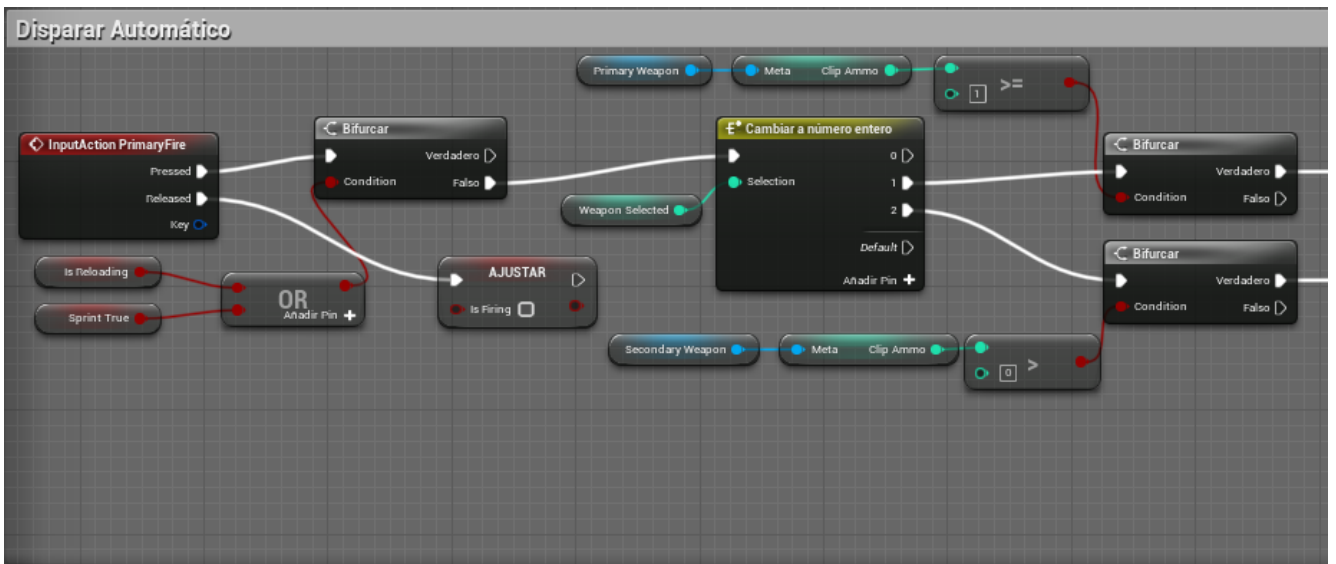
CONFIDENCIAL

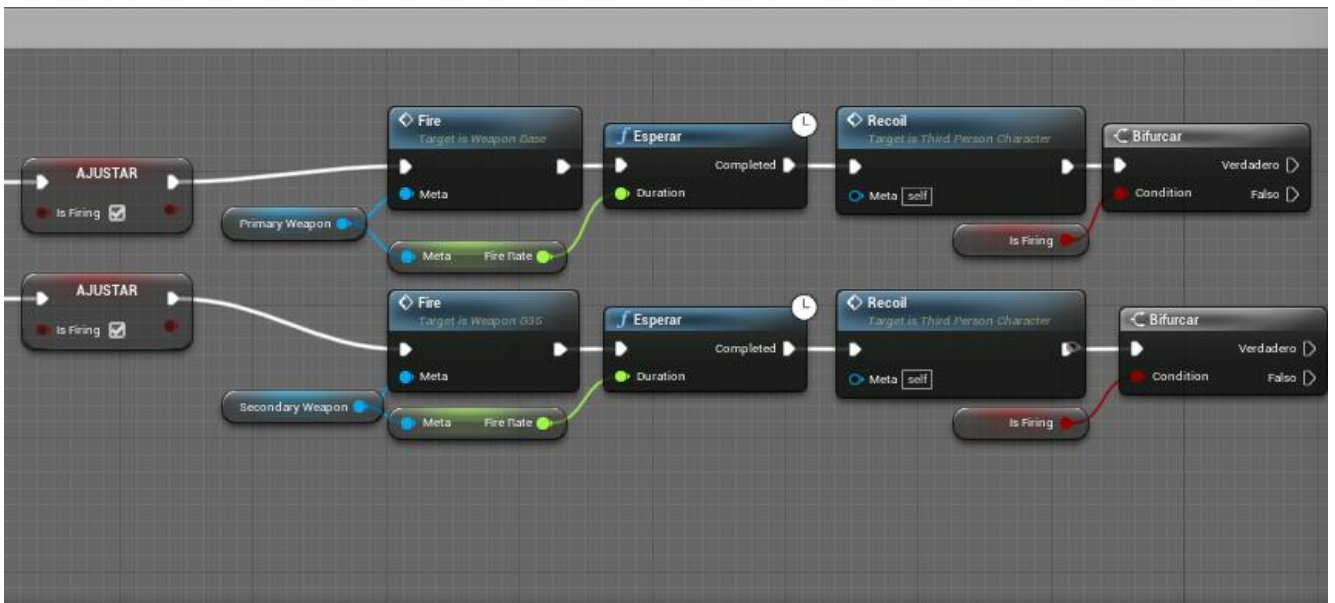
CONFIDENCIAL

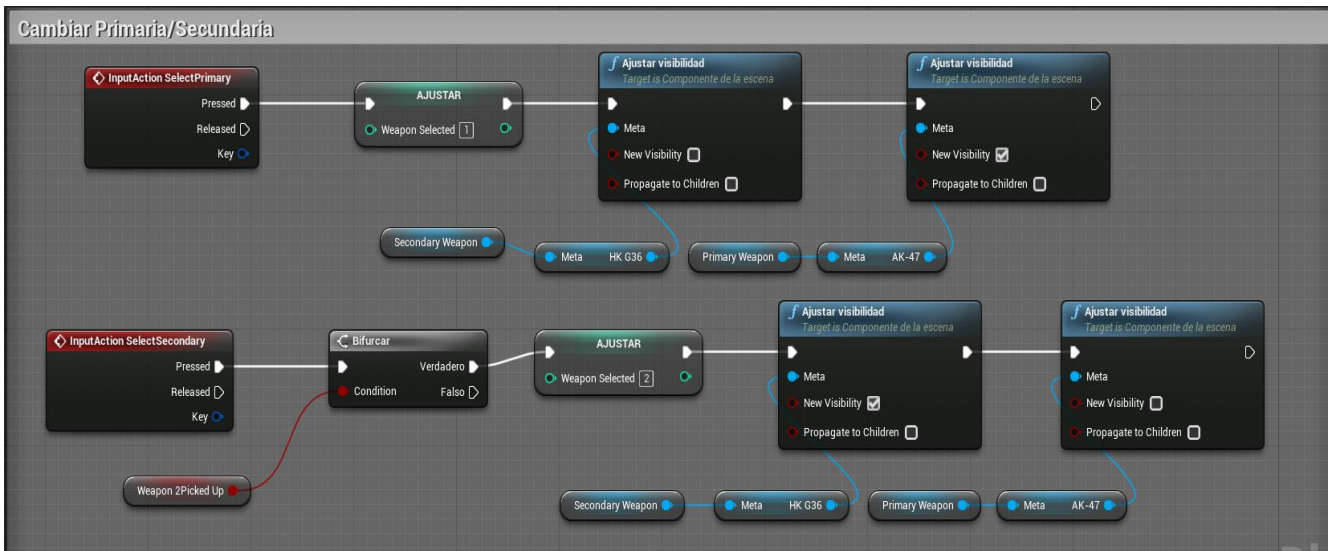
CONFIDENCIAL

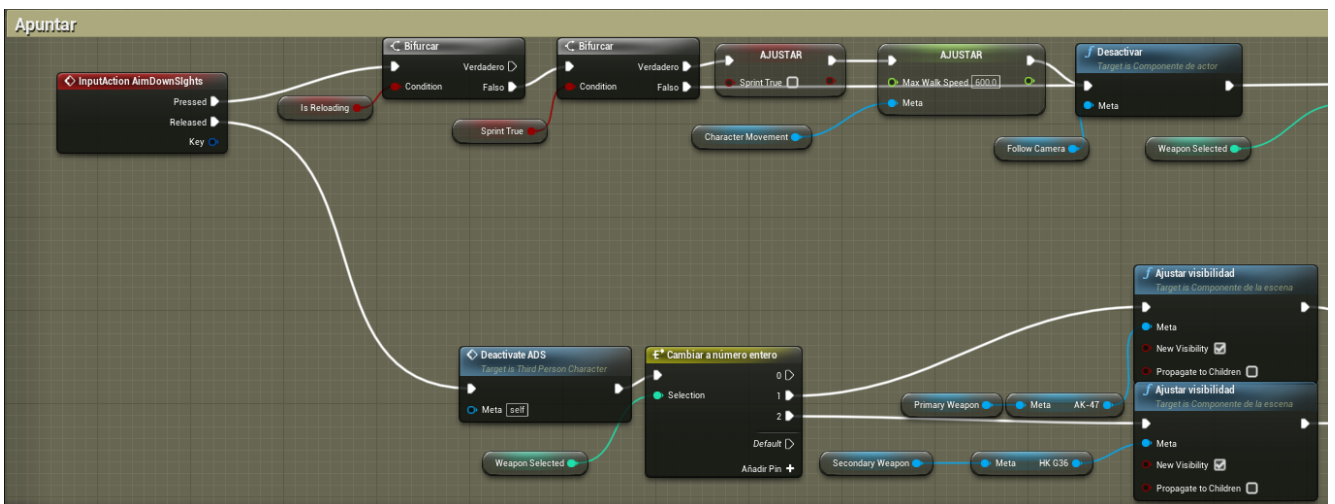
CONFIDENCIAL

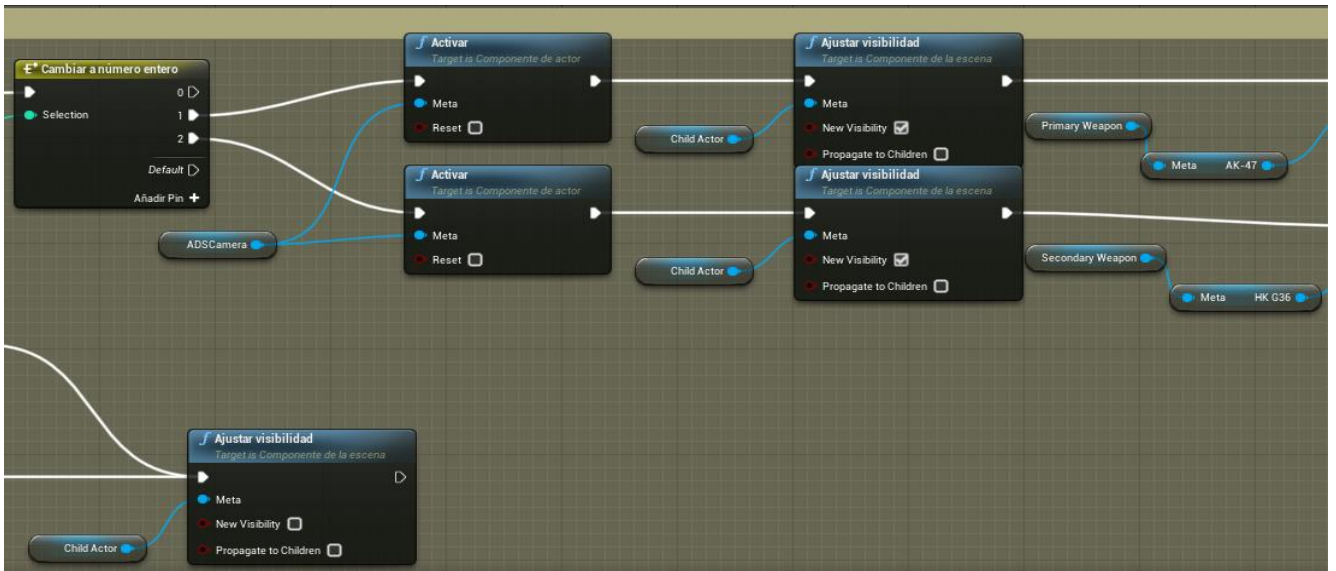
ANEXO II: BLOQUES DE CÓDIGO DEL PERSONAJE

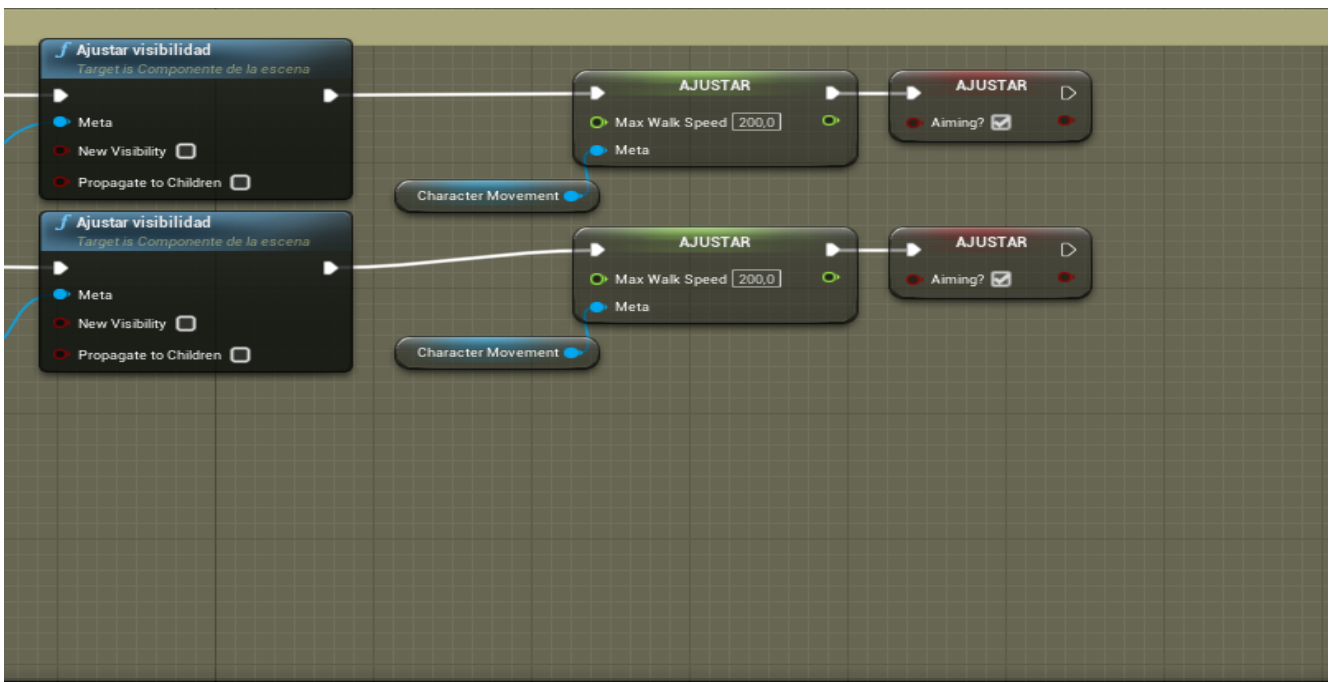


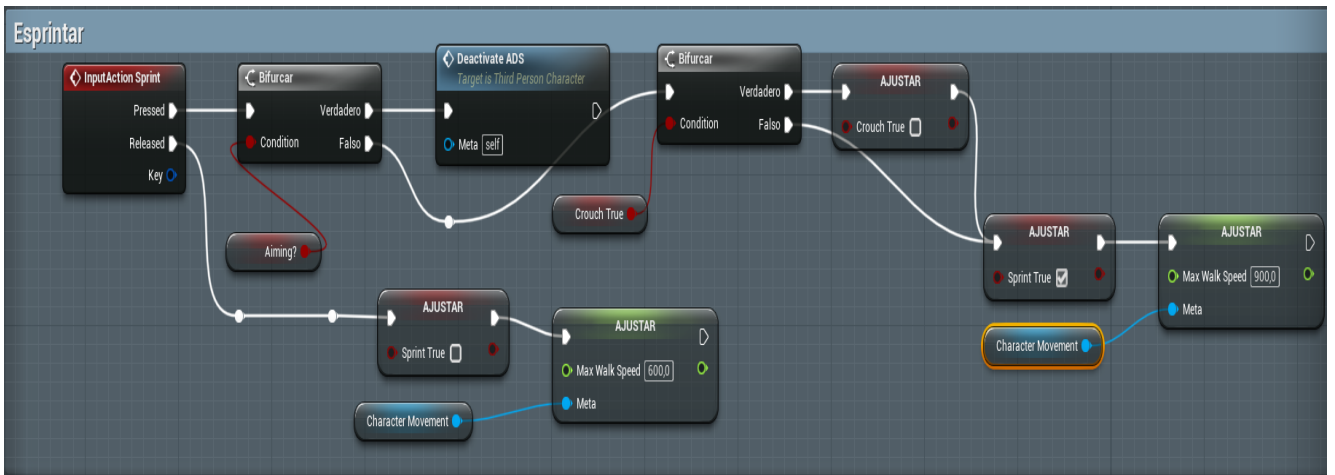


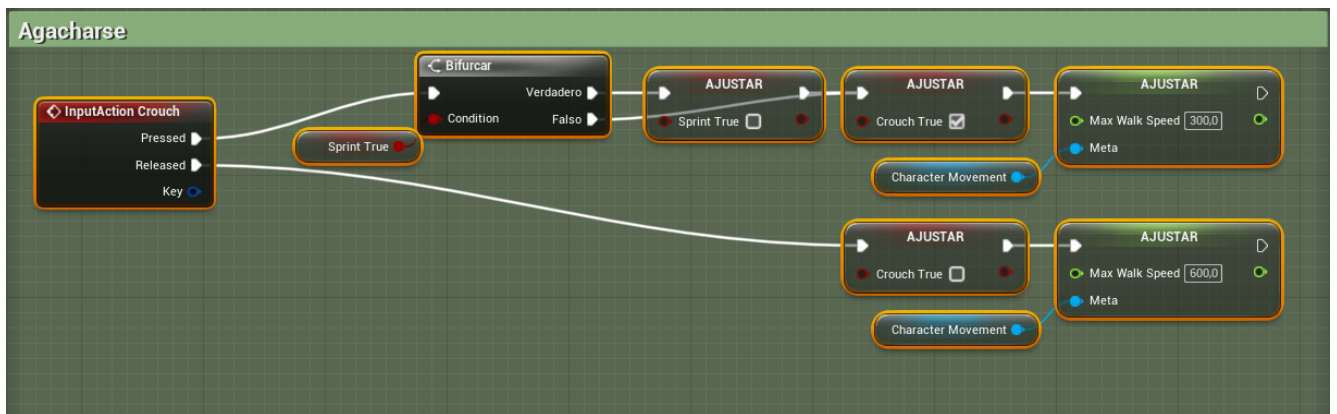


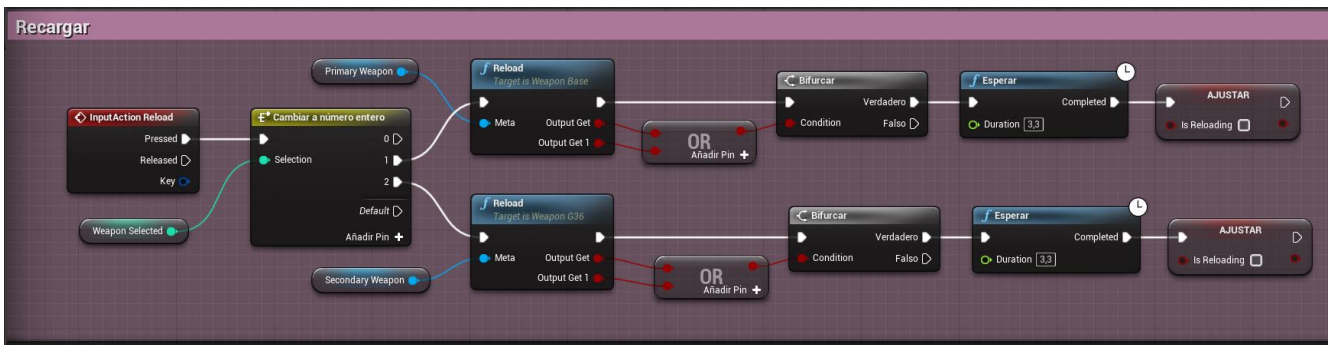


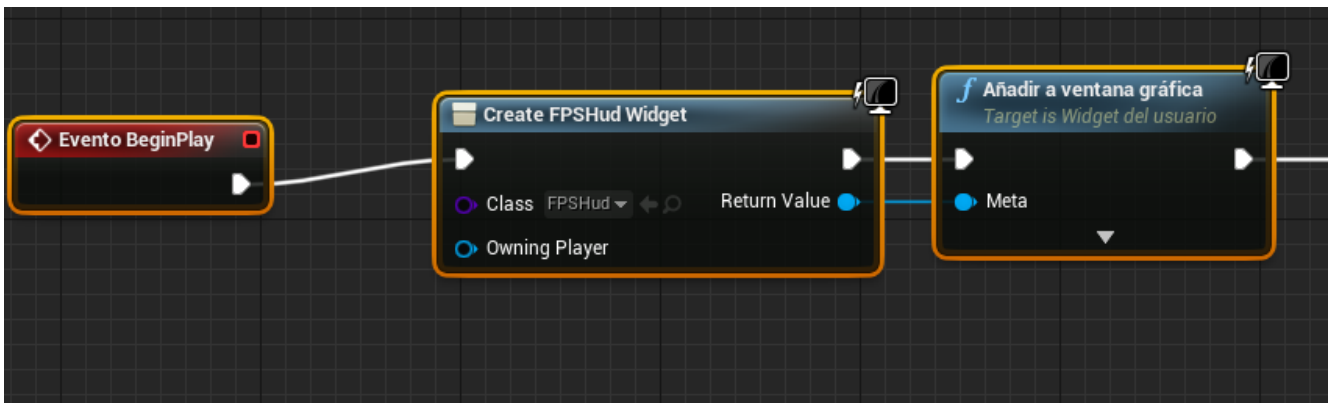


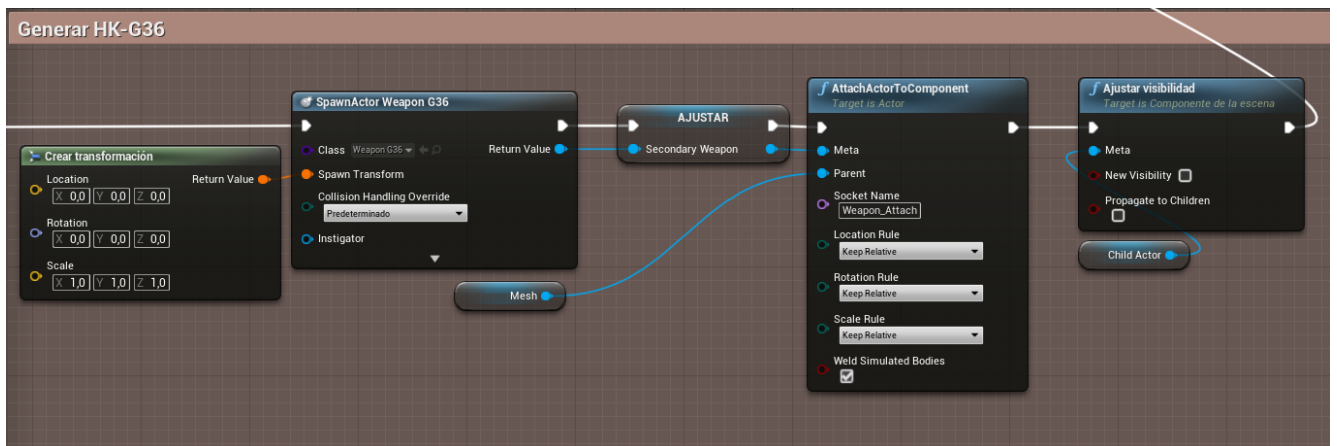


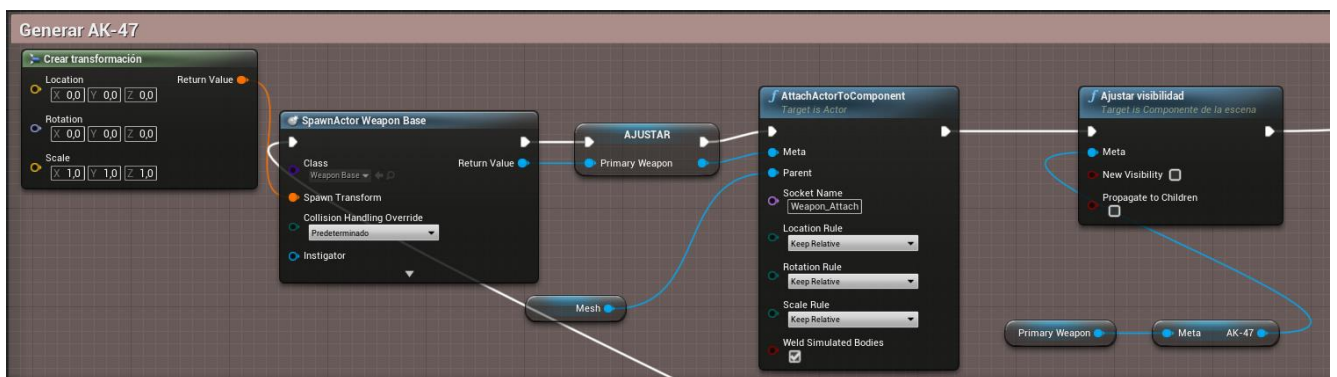


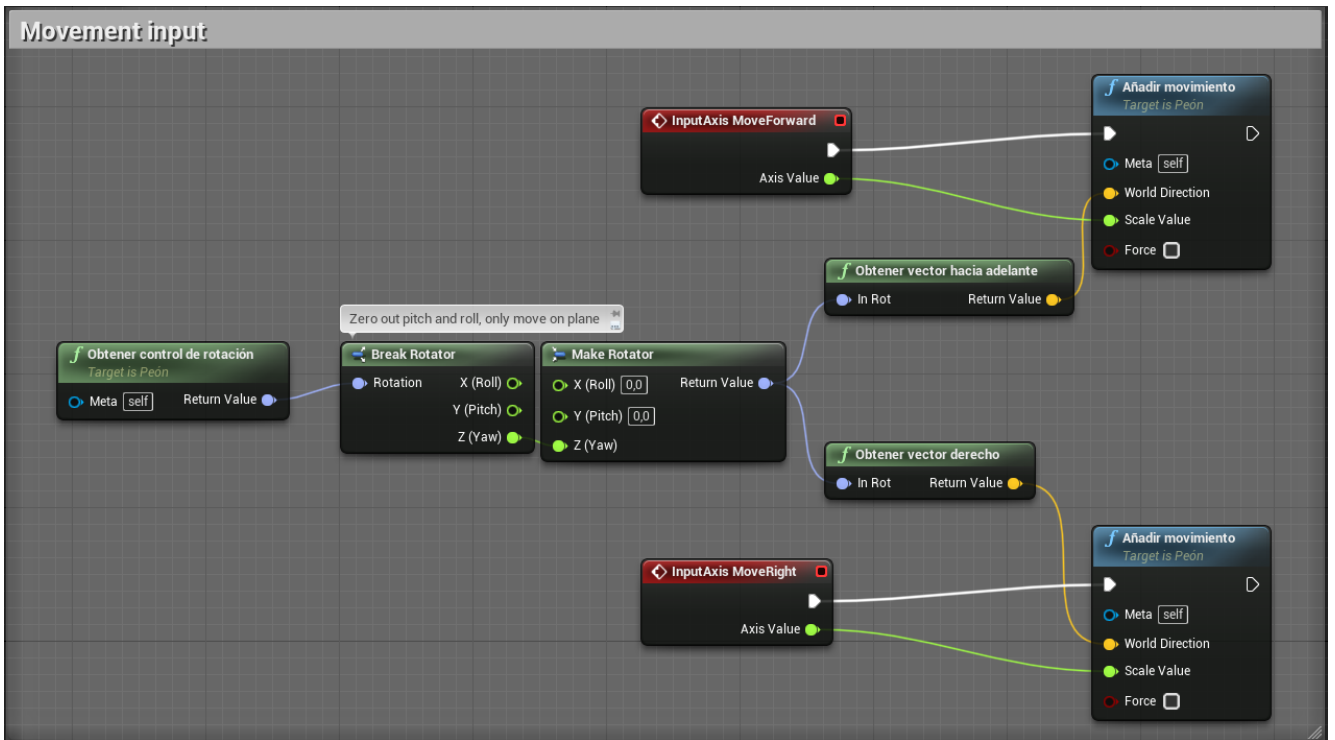


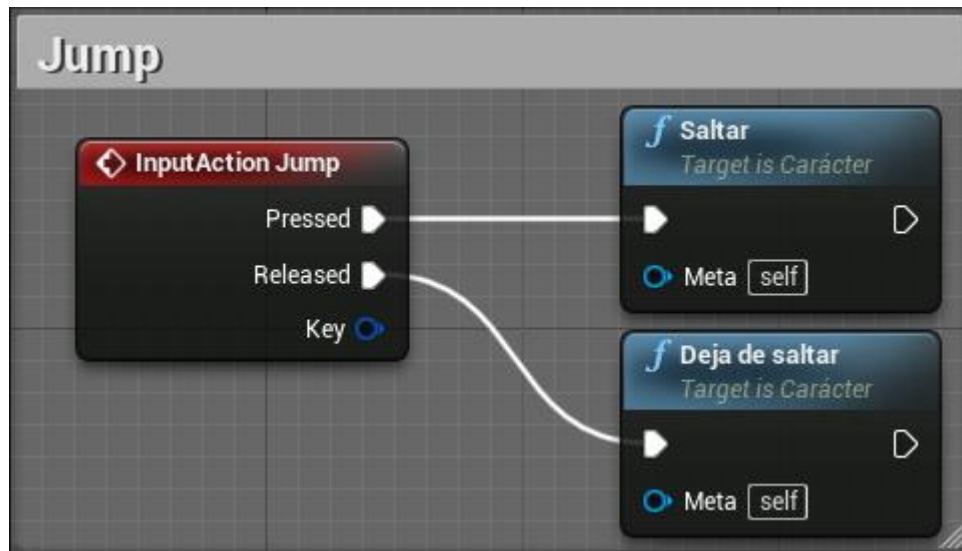




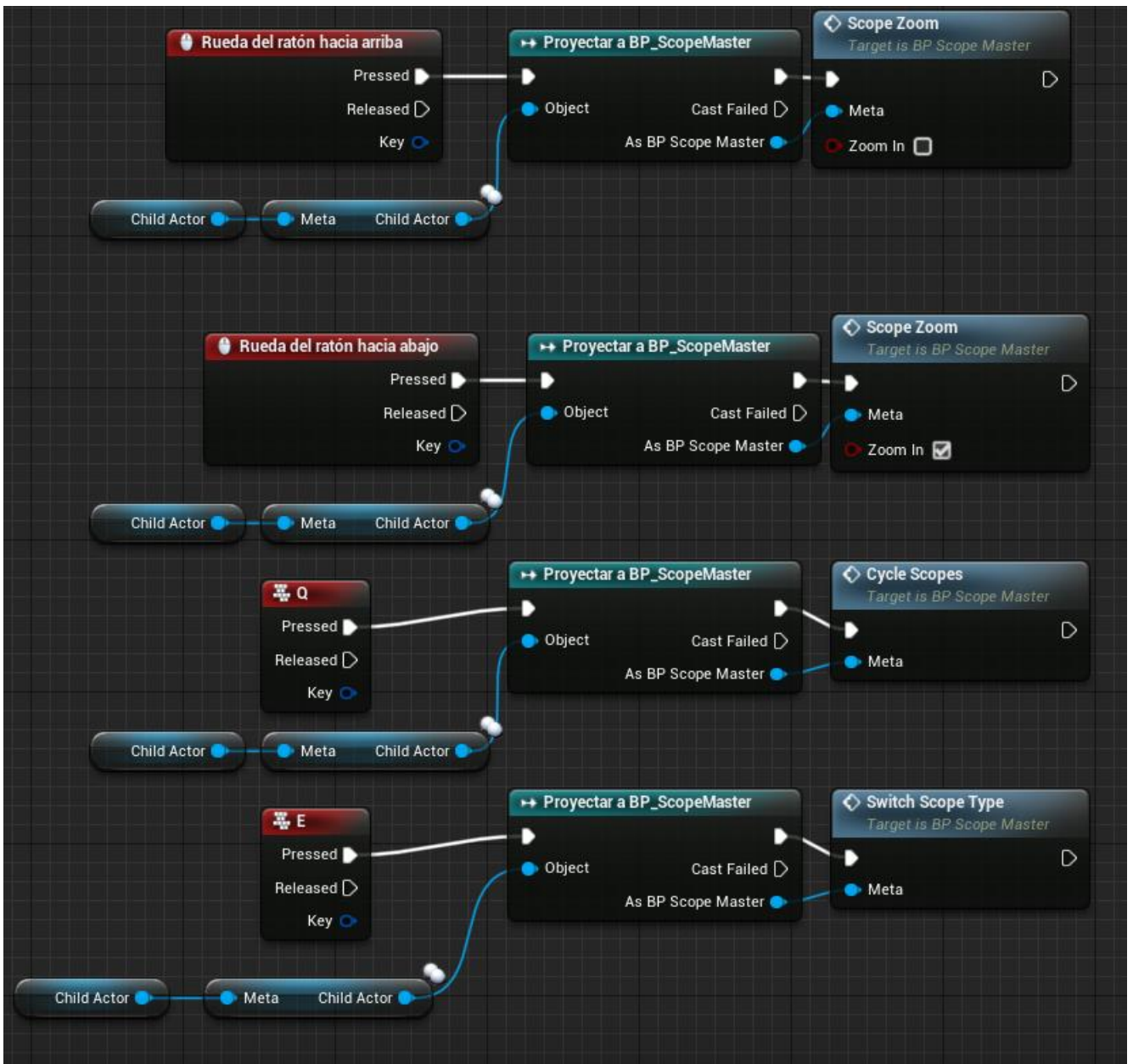


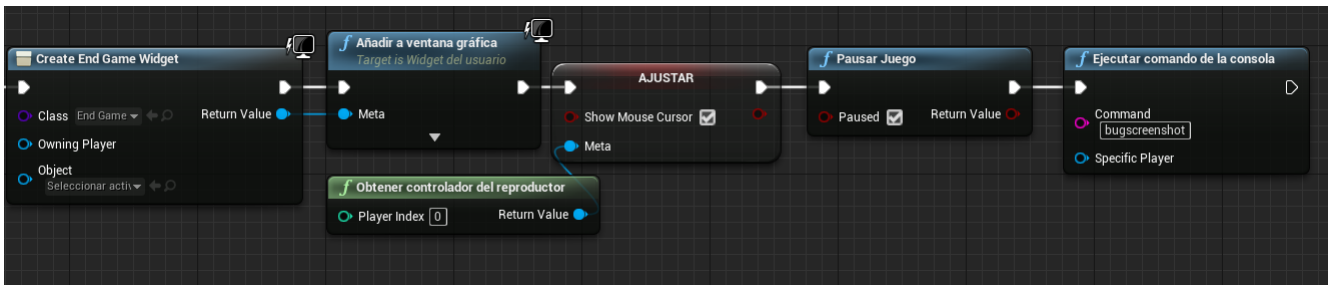




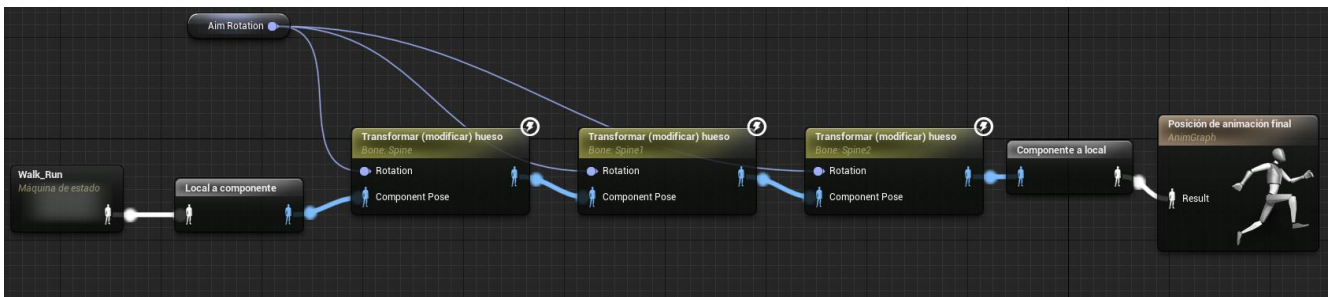


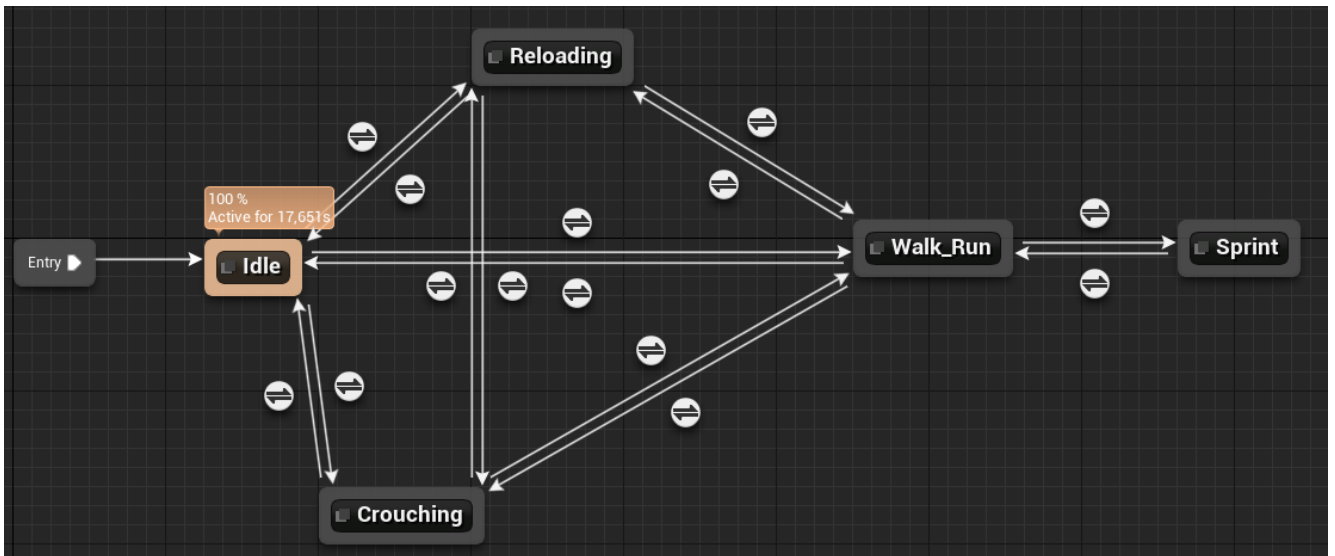


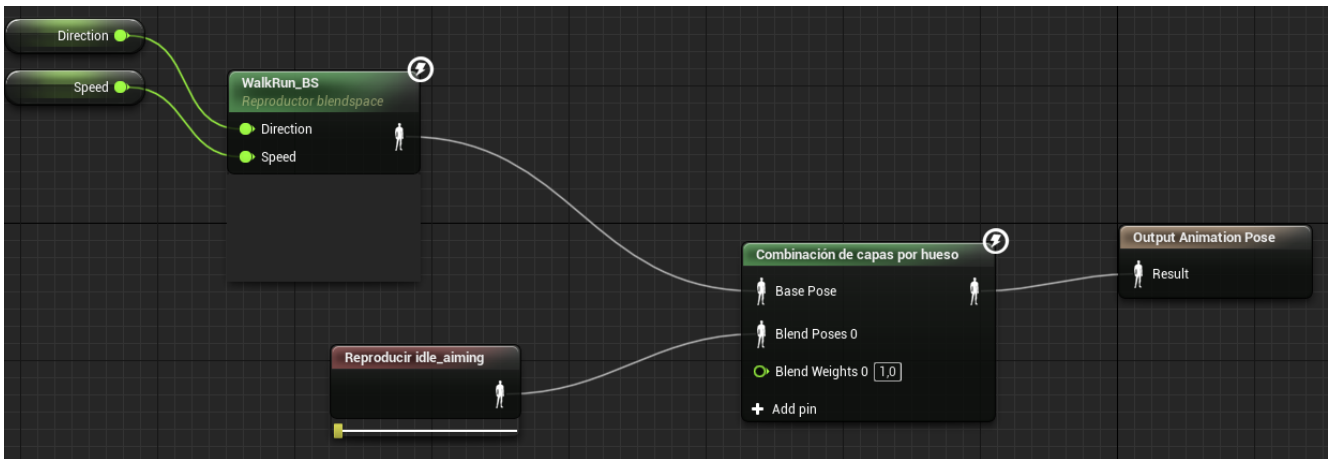


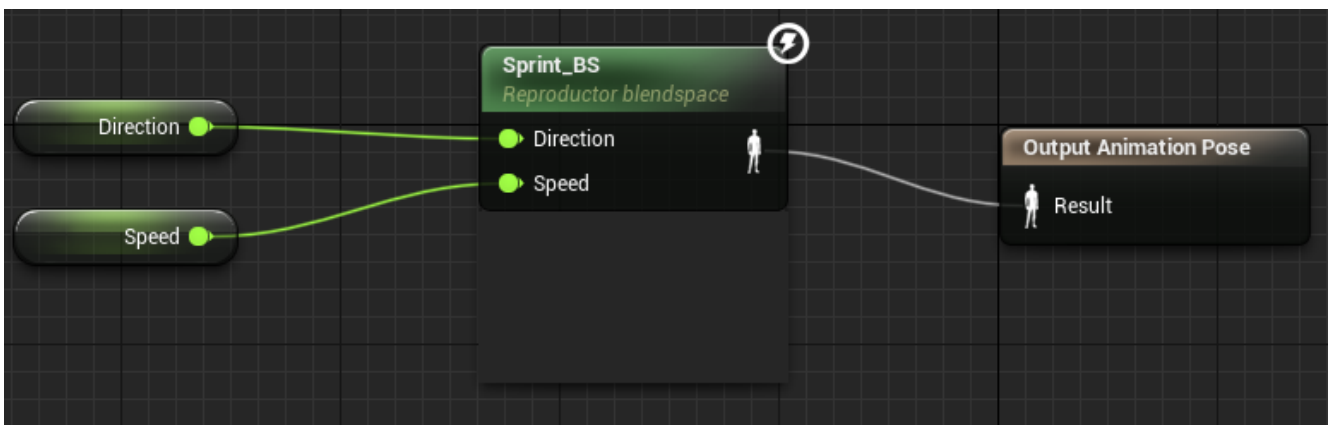


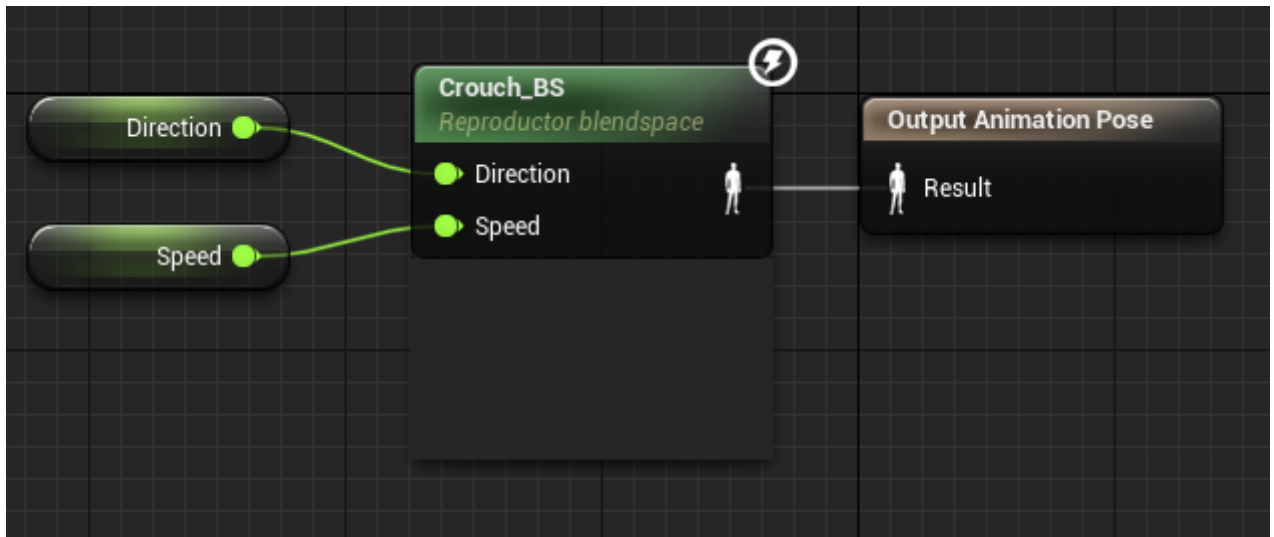
ANEXO III: BLOQUES DE CÓDIGO DE ANIMACIÓN DEL PERSONAJE

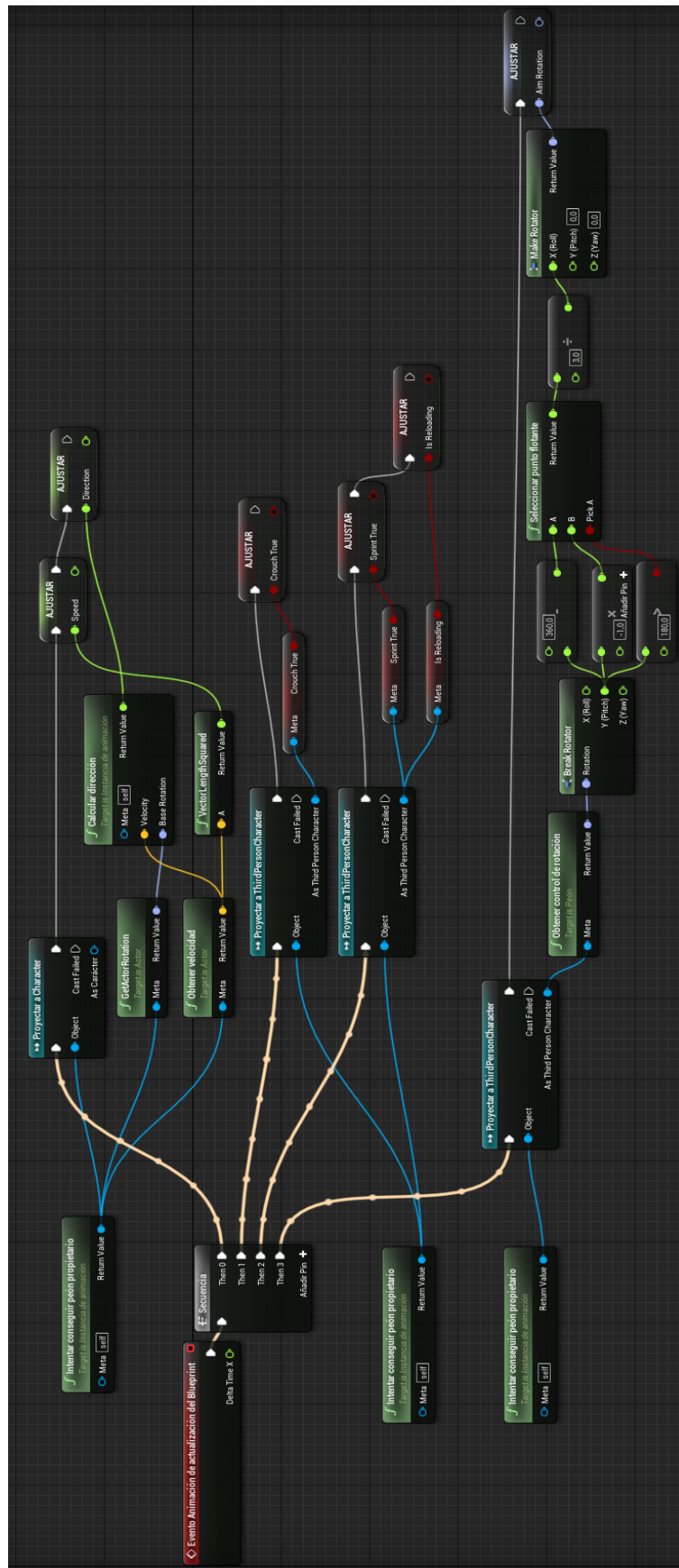




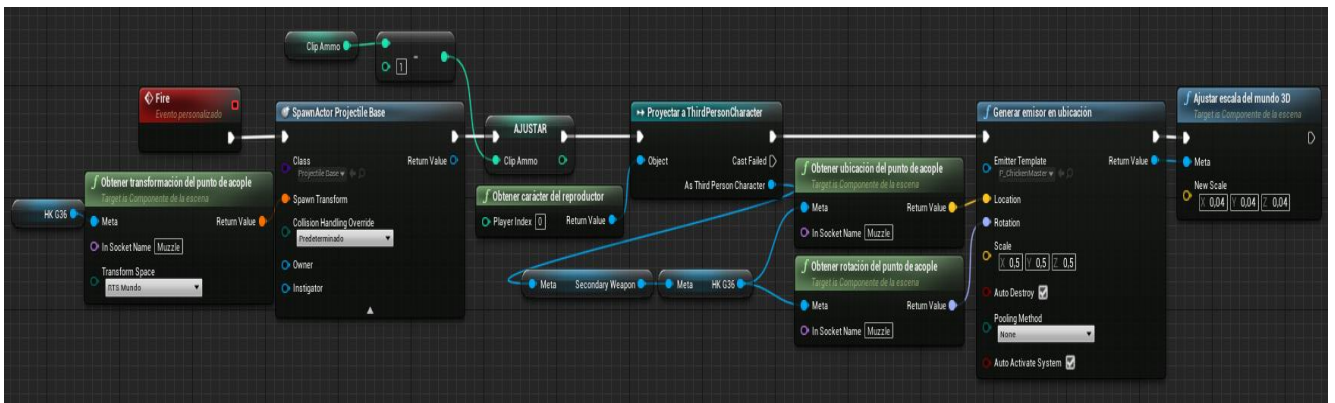


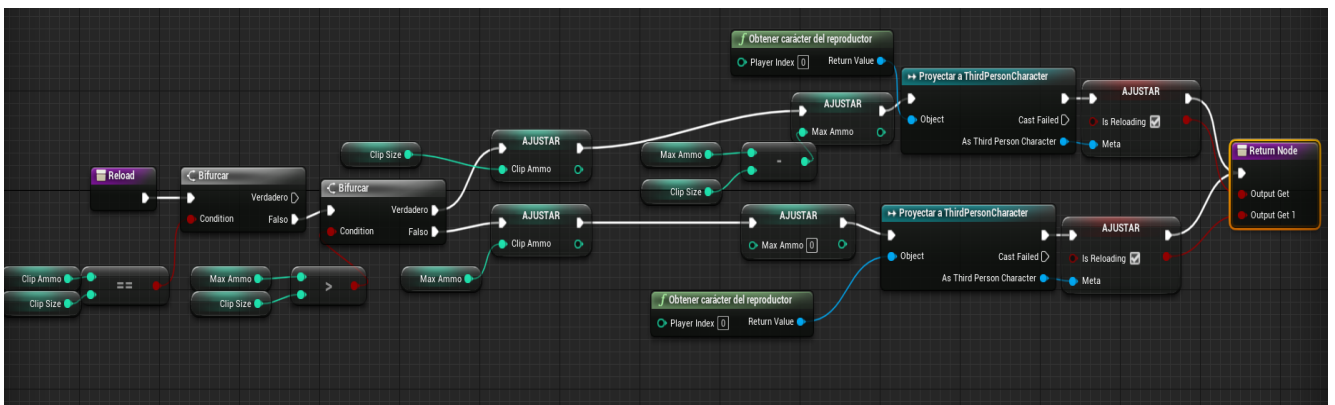




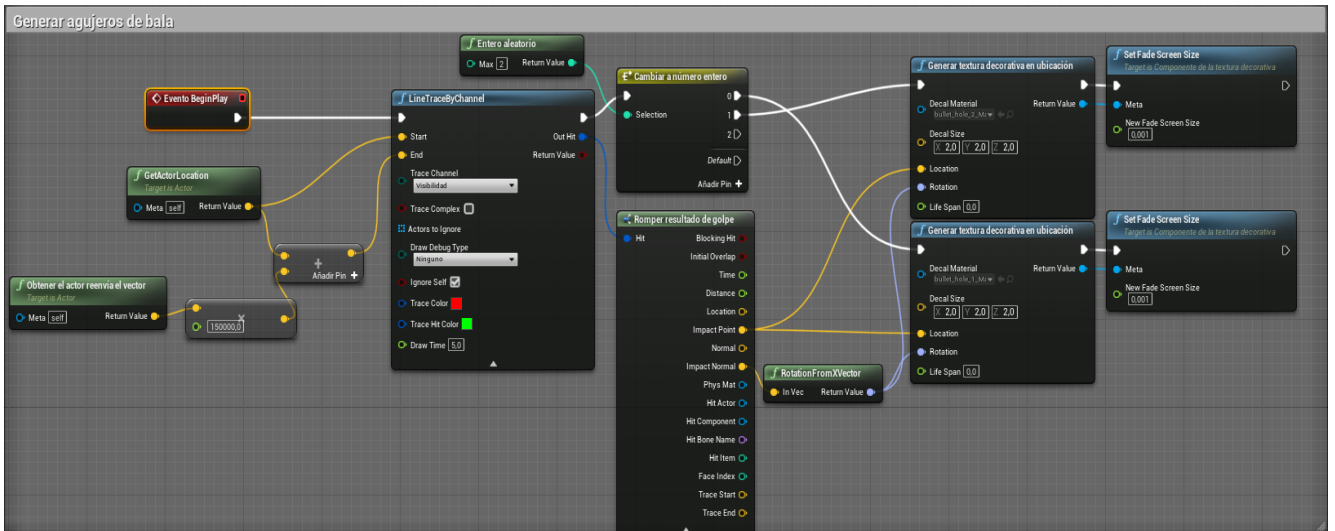


ANEXO IV: BLOQUES DE CÓDIGO DEL ARMA

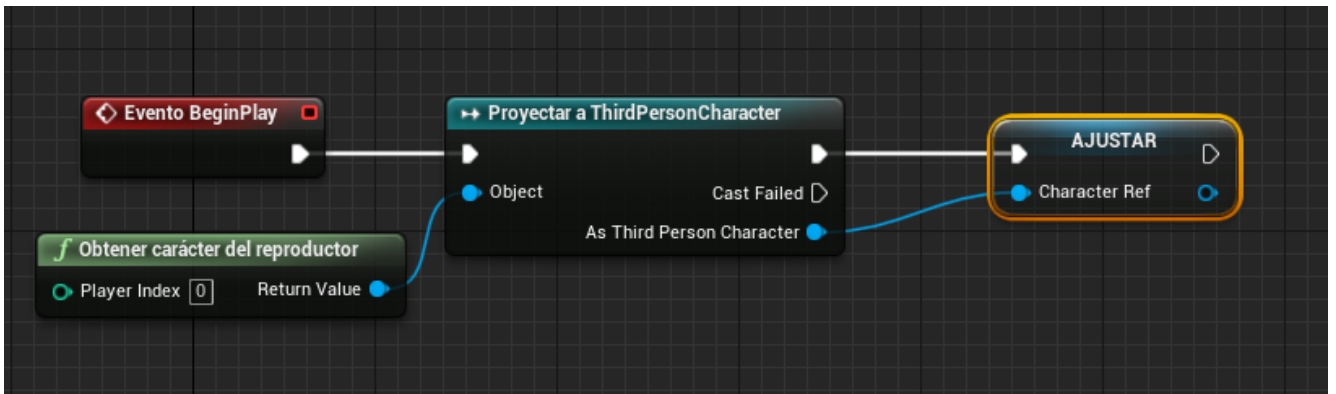


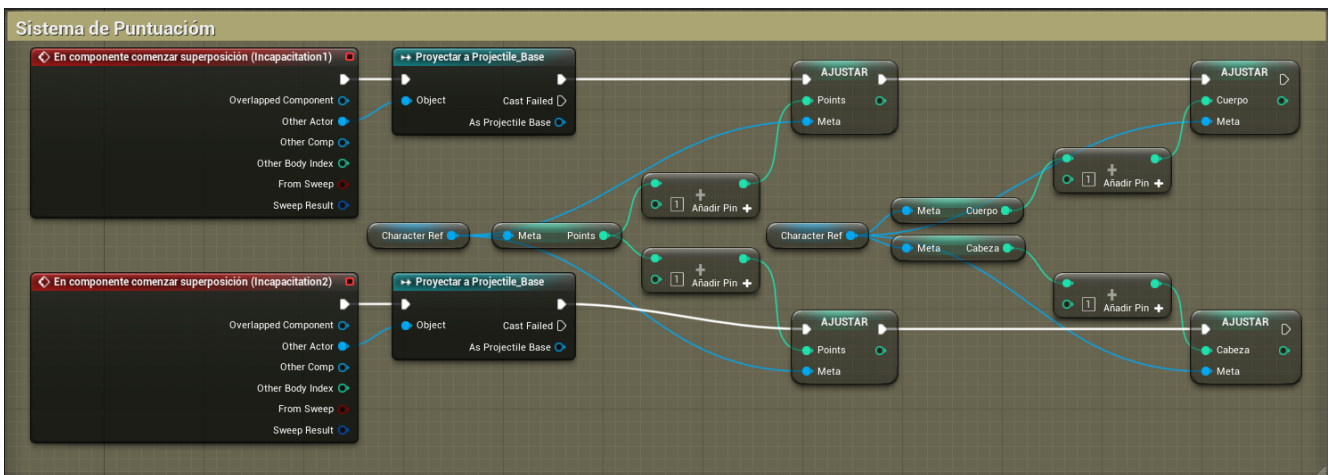


ANEXO V: BLOQUES DE CÓDIGO DEL PROYECTIL

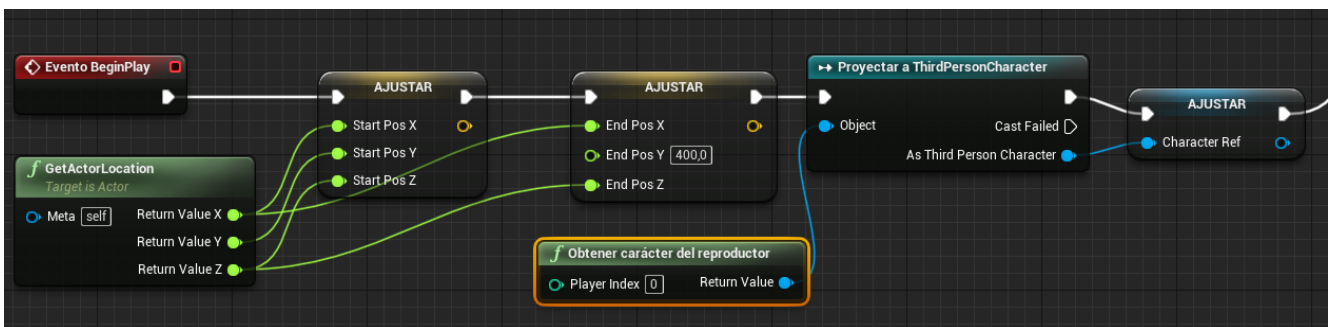


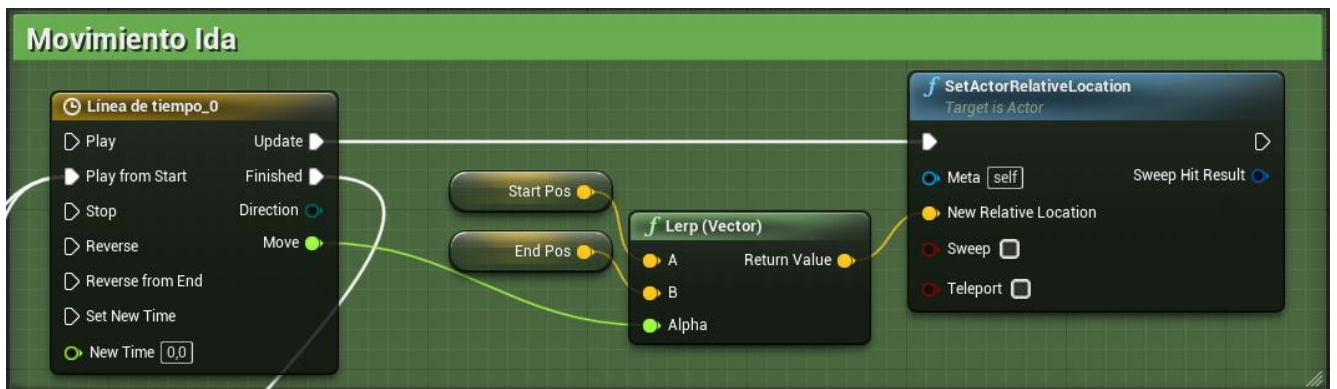
ANEXO VI: BLOQUES DE CÓDIGO DEL BLANCO SILUETA

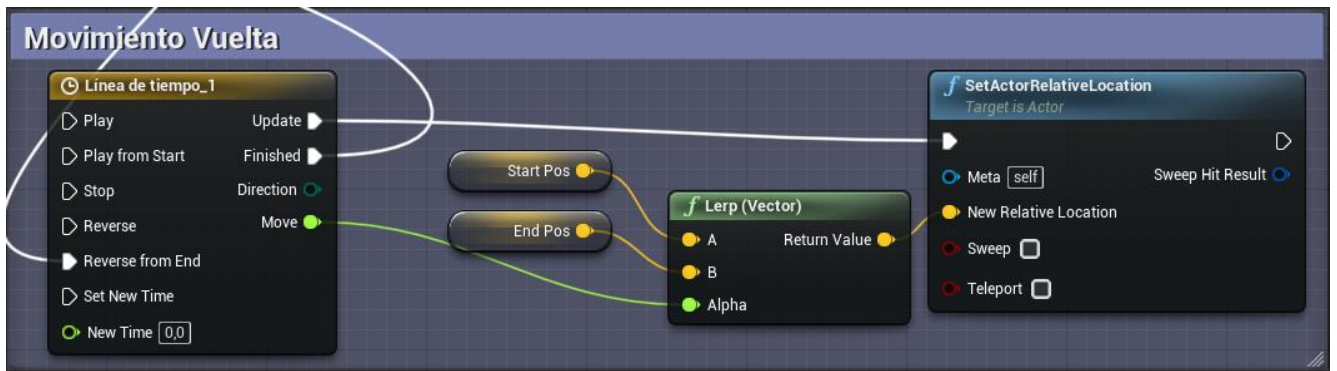


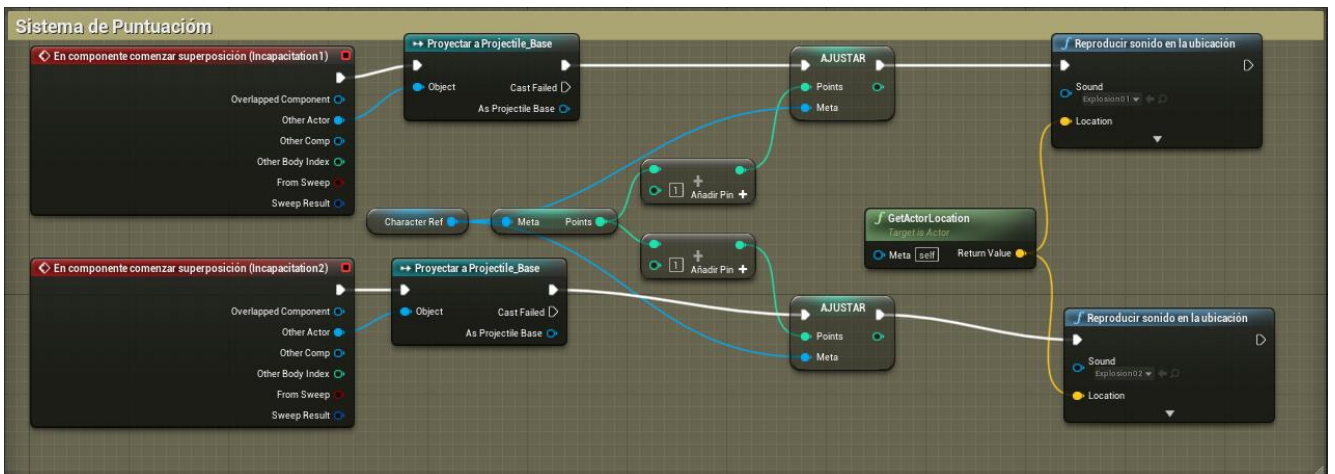


ANEXO VII: BLOQUES DE CÓDIGO DEL BLANCO MÓVIL

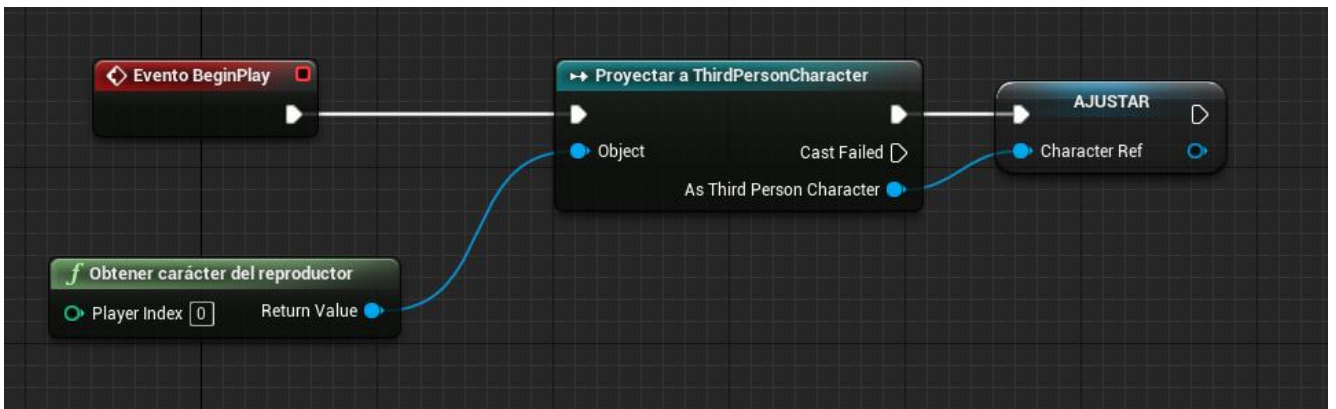


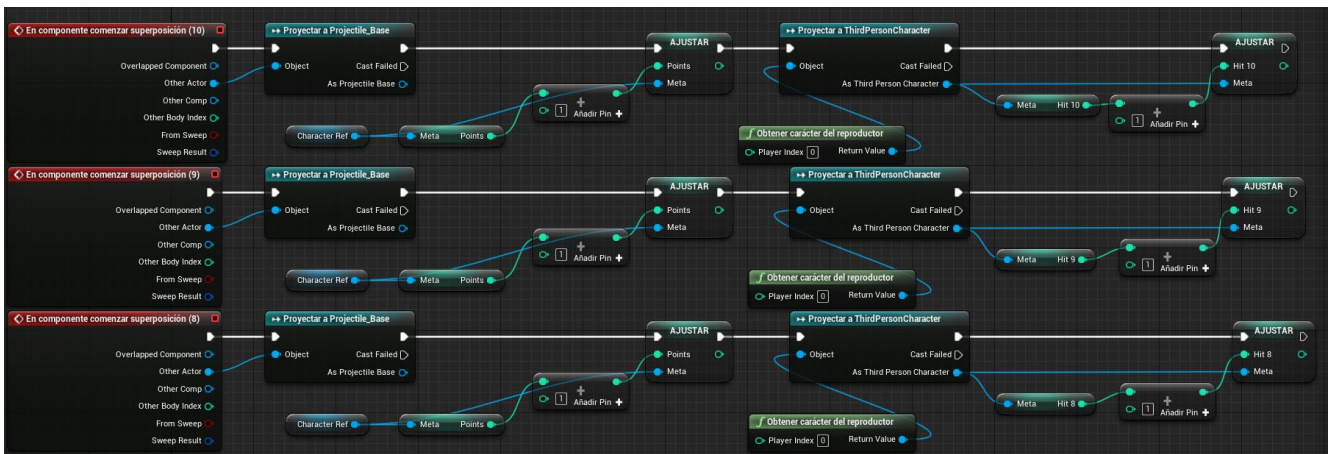


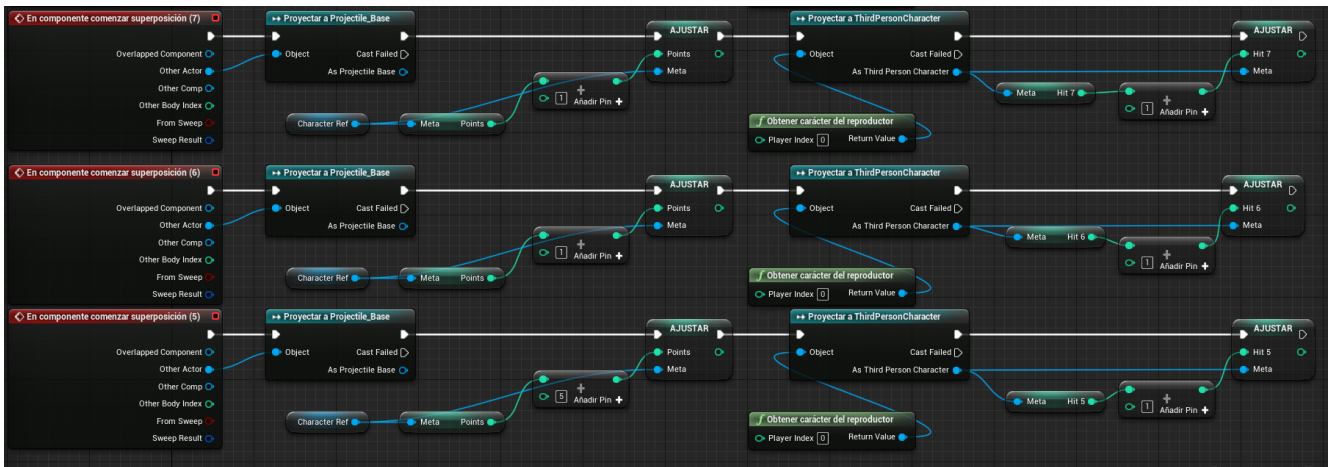




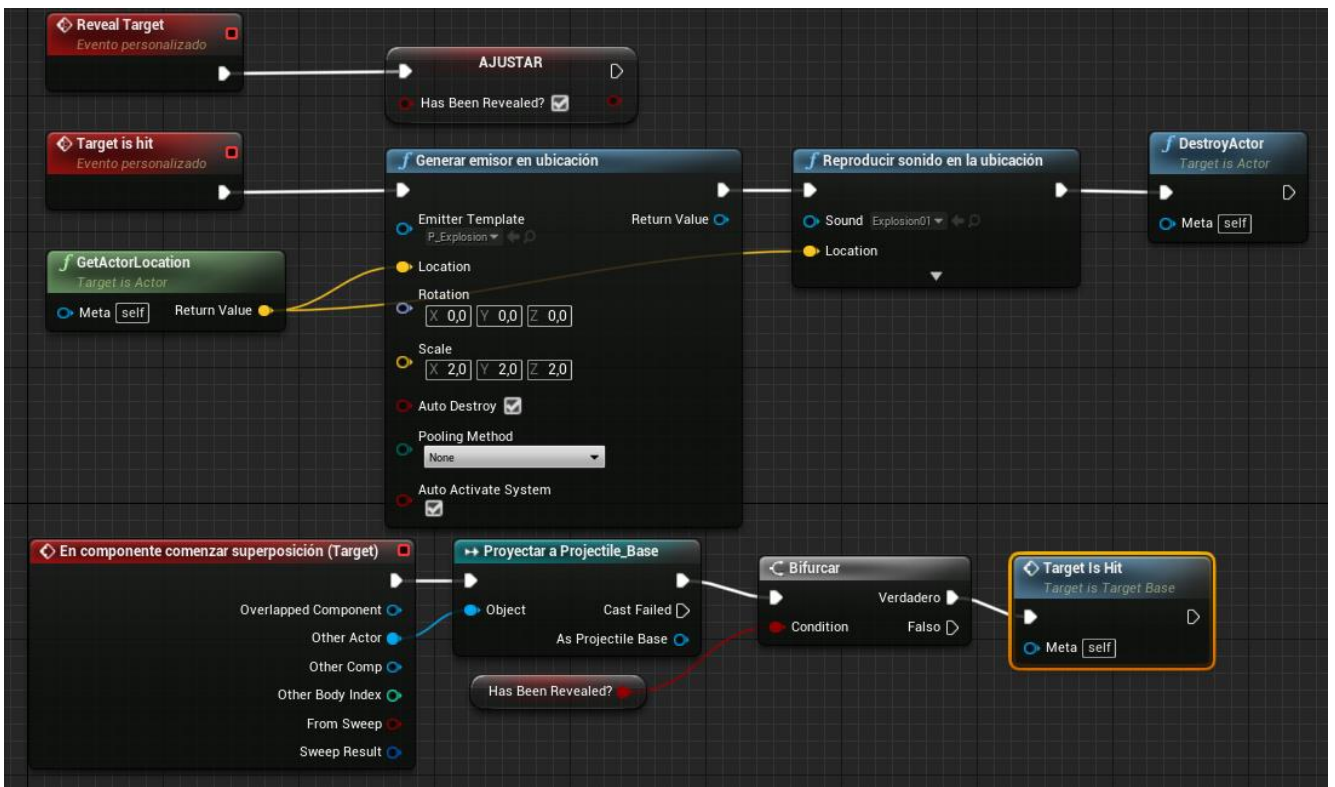
ANEXO VIII: BLOQUES DE CÓDIGO DEL BLANCO DE PRECISIÓN

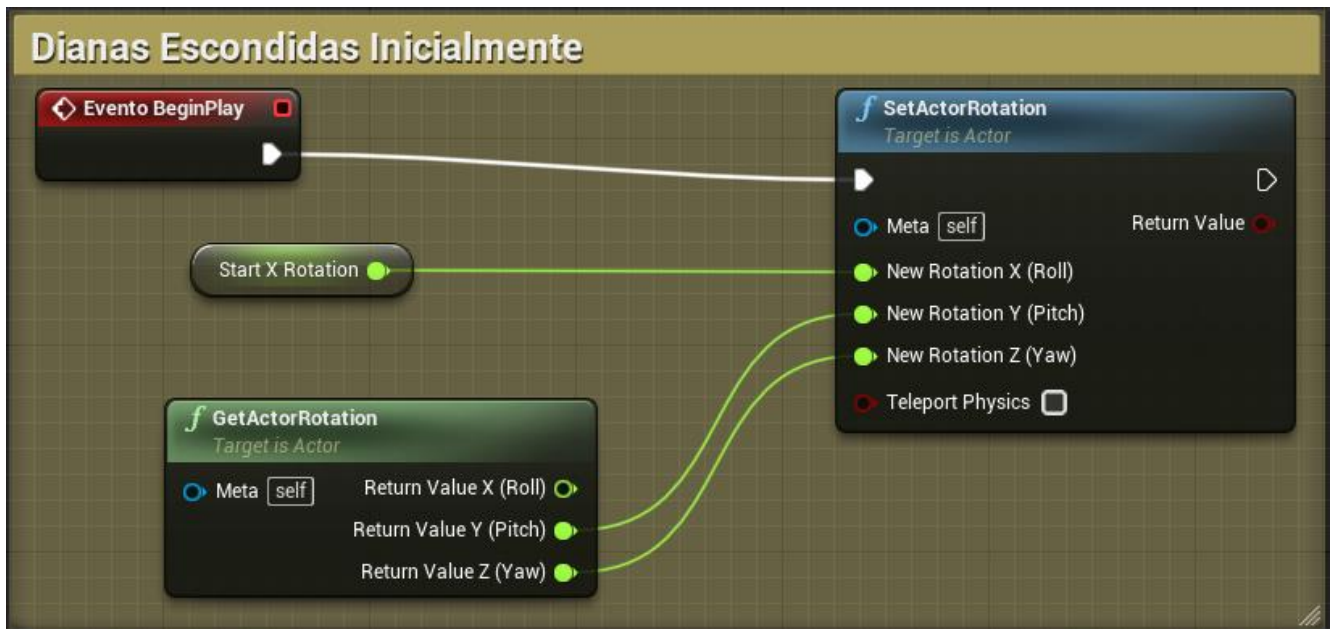


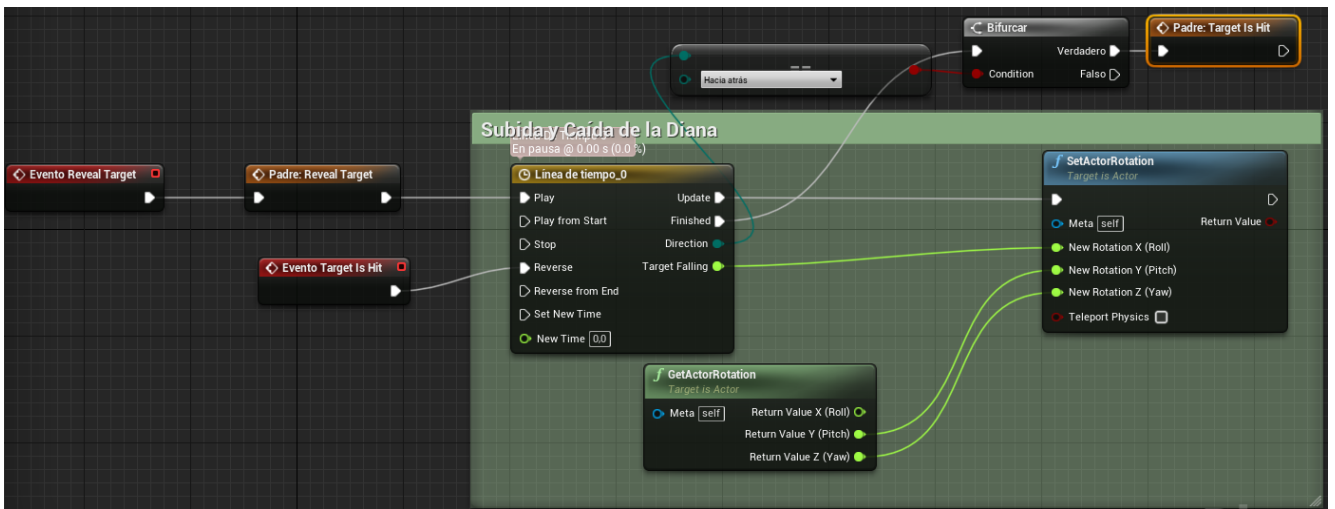


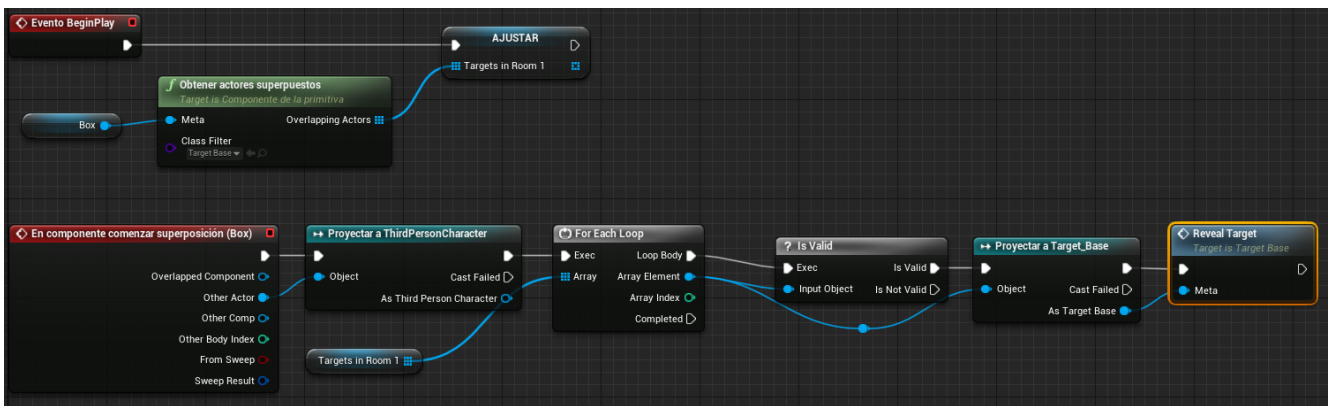


ANEXO IX: BLOQUES DE CÓDIGO DEL BLANCO INSURGENTE

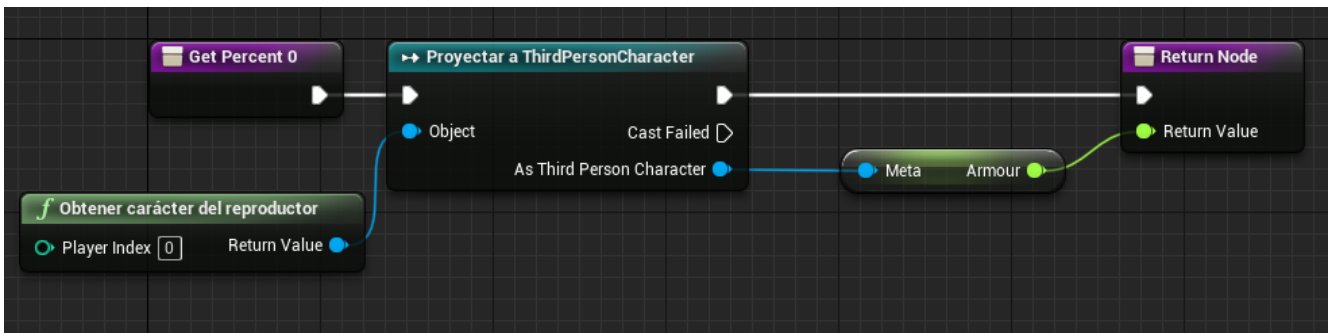


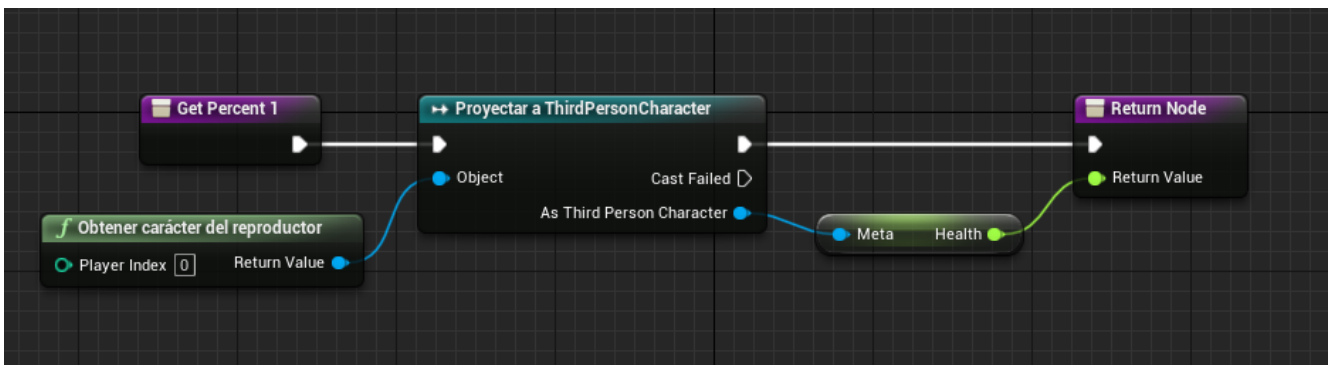


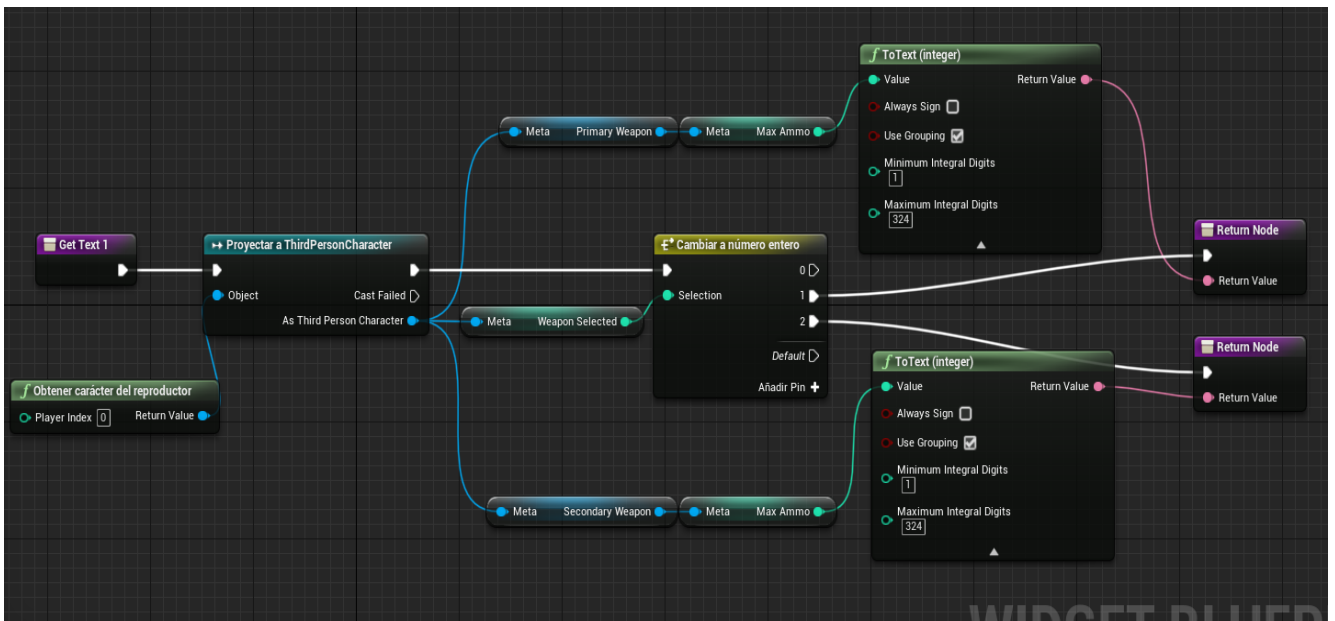


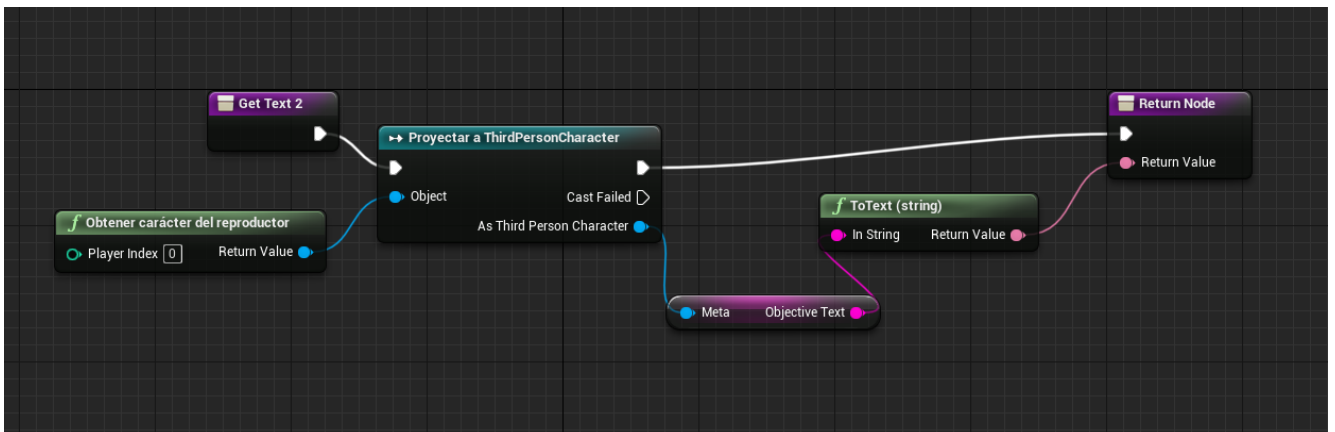


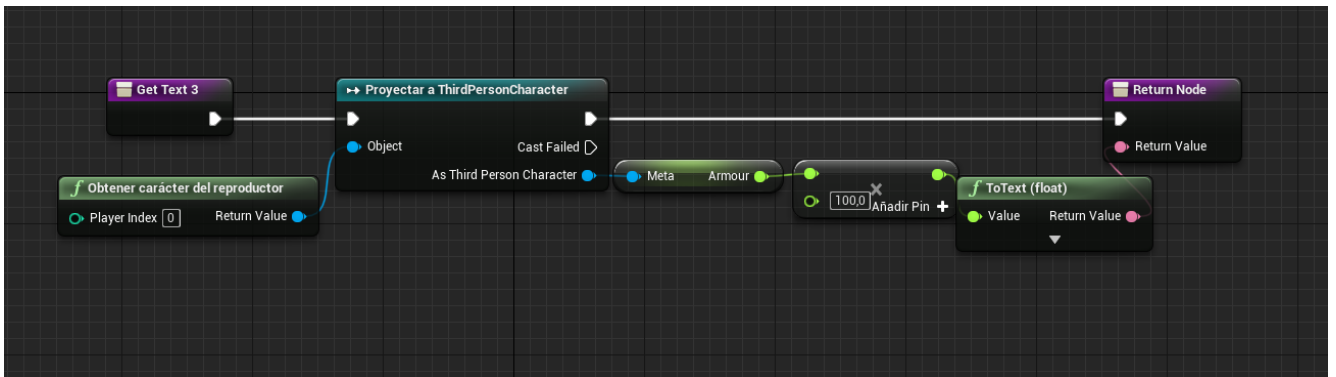
ANEXO X: BLOQUES DE CÓDIGO DEL FPS HUD

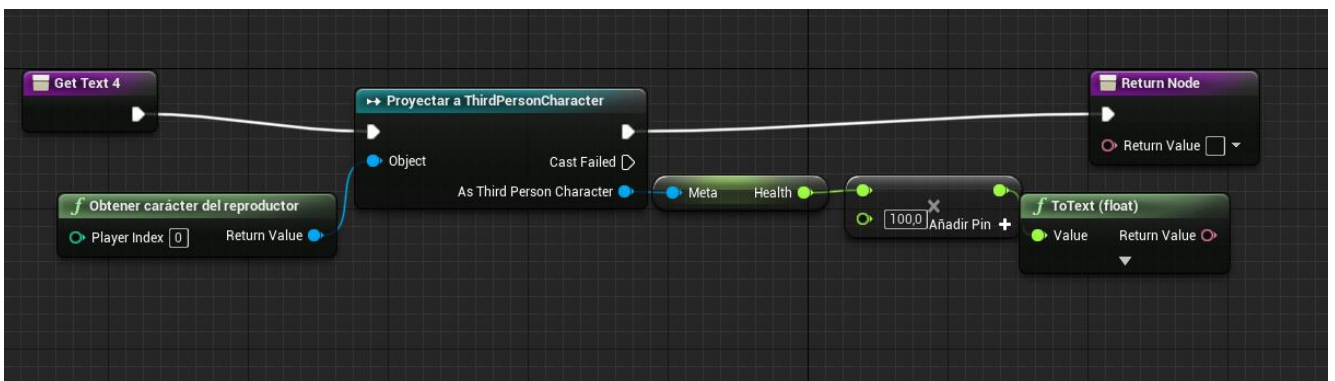


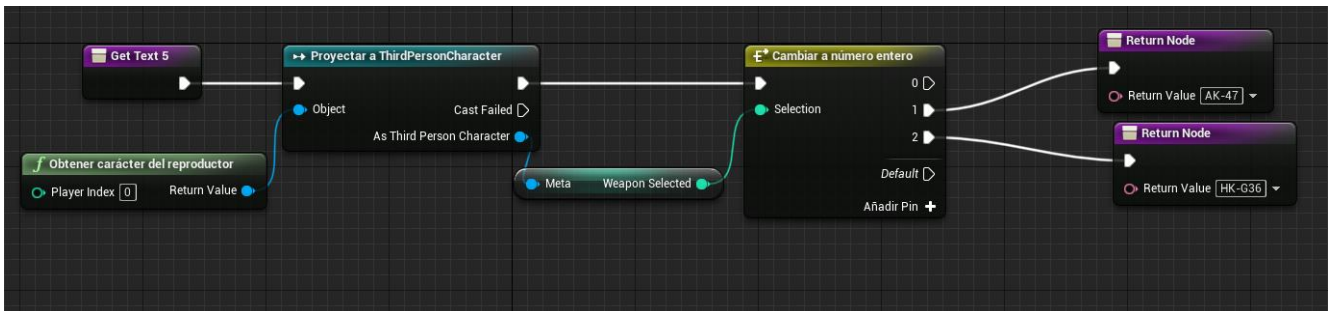


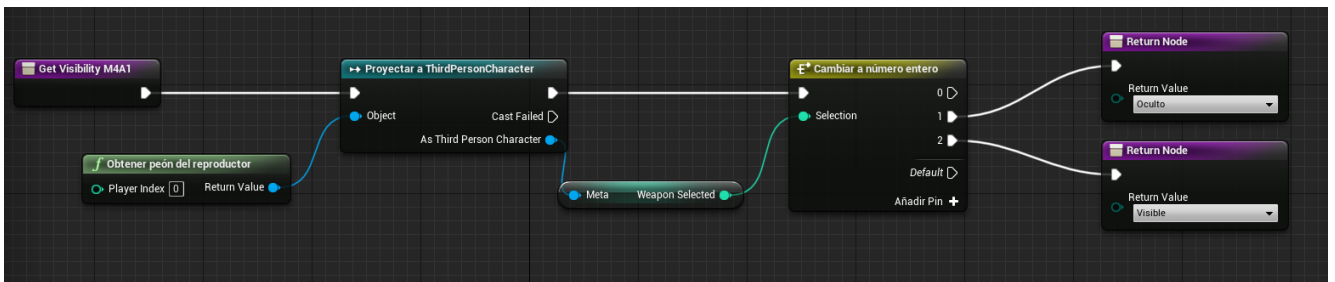


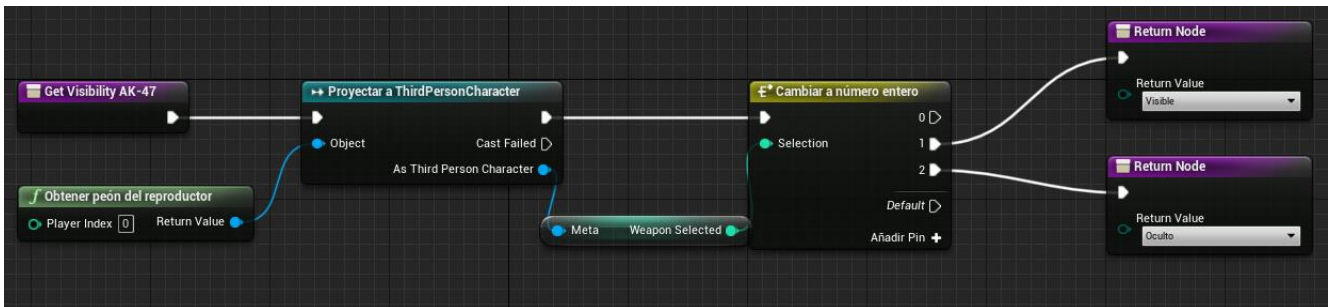






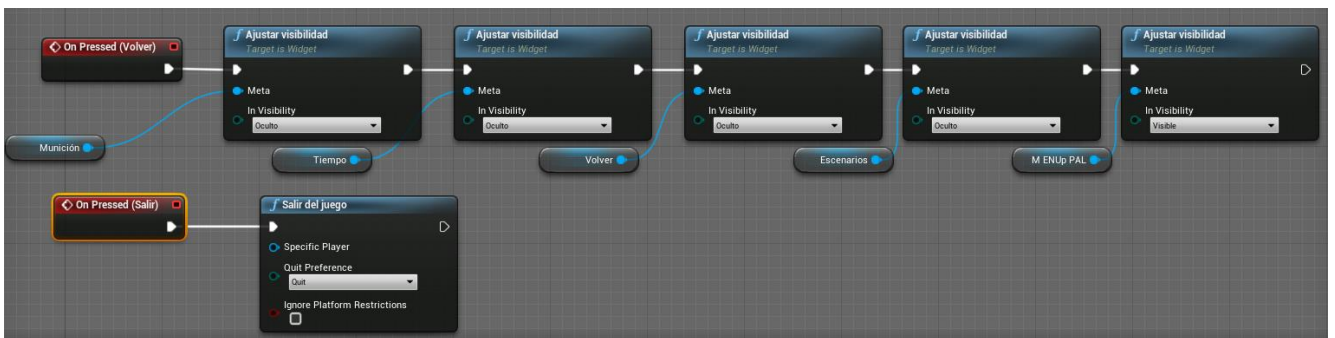


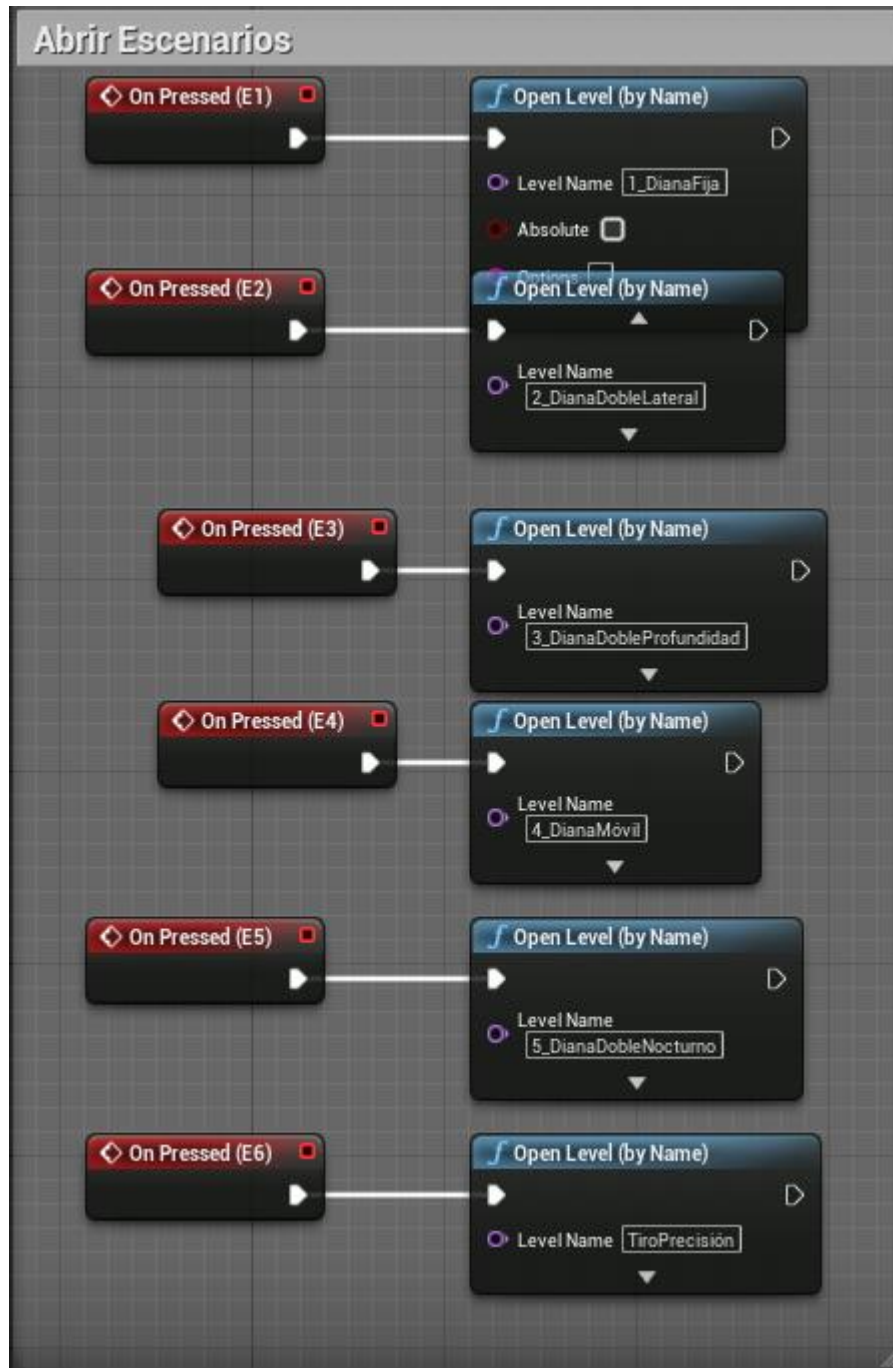




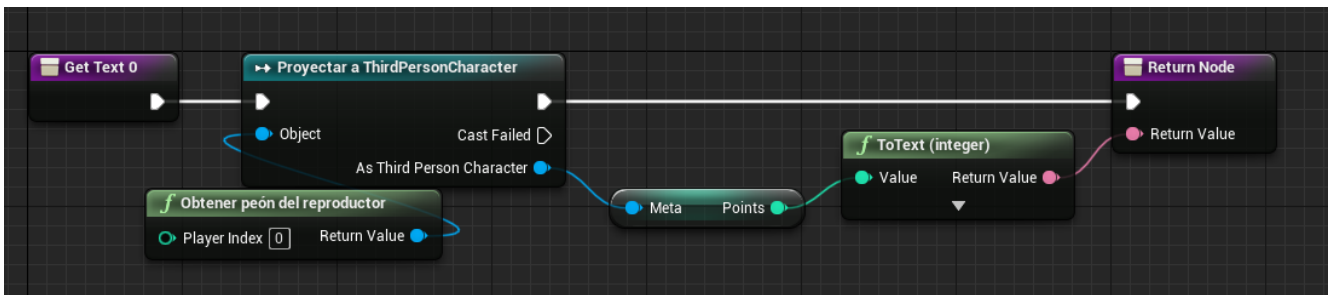
ANEXO XI: BLOQUES DE CÓDIGO DEL MENÚ PRINCIPAL

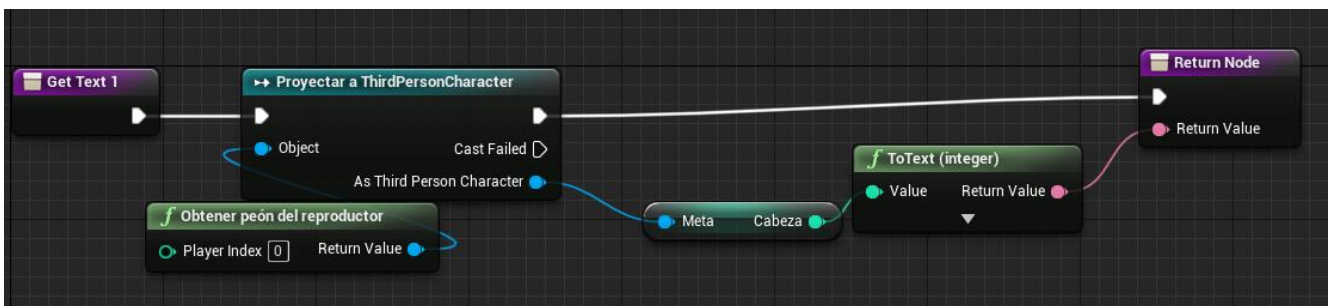


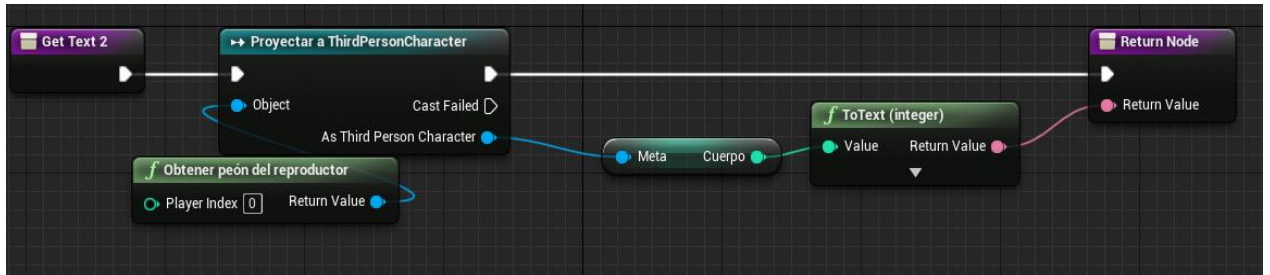




ANEXO XII: BLOQUES DE CÓDIGO DE LA PANTALLA DE RESULTADOS







ANEXO XIII: MANUAL DE USUARIO DE LENOVO EXPLORER (USER GUIDE)

Lenovo Explorer

Immersive headset for Windows Mixed Reality

Lenovo™

User Guide



Contents

Lenovo Explorer User Guide	1
Guide d'utilisation Lenovo Explorer	12
Lenovo Explorer – Benutzerhandbuch	23
Lenovo Explorer ユーザー・ガイド	34
Руководство пользователя Lenovo Explorer	45
Lenovo Explorer 用户指南	56
Lenovo Explorer 使用手冊	67

Important safety information

Before using the product, ensure that you read and follow the safety information in this user guide. By carefully following the safety information, you can help protect yourself from hazards and create a safer and more comfortable work or entertainment environment.

Important

Ensure that your health condition allows you to use the product.

Consult a doctor before using the product if:

- You have any medical condition or issue that could affect your ability to perform physical activities safely.
- You are or may be pregnant.
- You have heart, respiratory, back, joint, orthopedic, or other serious medical conditions.
- You have high blood pressure, tendinitis, or carpal tunnel.
- You have difficulty with or have been instructed to restrict physical activities.
- You are elderly.
- You have binocular vision abnormalities.
- You have psychiatric disorders.

Use in a safe environment.

The product produces an immersive virtual reality experience that can distract you from and completely block your view of the actual world. The image displayed in the product does not match with the real world. It is important that you use the product in a safe environment and observe the following instructions:

- Always be aware of your surroundings, including the cables running from the product.
- Before using the product, look in all directions to identify all potential obstacles and make sure that you have plenty of space to move around.
- While using the product, mind the cables to avoid tripping over, falling, or accidentally pulling on them. Move your feet slowly and carefully and maintain good footing to avoid running into walls or other obstacles. Tripping over the cables or contacting obstacles can result in injury or damage.
- Always remember that everything that you see in the virtual world is not real. Do not sit or stand on the objects you see in the virtual world or use them for support. Do not rely on the virtual boundary system alone for protections.

Allergen notice

Lenovo® avoids using known allergens in our products. For detailed testing results, go to http://www.lenovo.com/support/lenovo_explorer.

Caution

Do not use the product under a high or low environmental temperature.

The recommended environmental temperature is 0°C to 35°C (32°F to 95°F). Stop using the product if you feel that the product is overheated. Prolonged contact with an overheated product can cause a skin burn.

High volume usage notice



Exposure to loud noise from any source for extended periods of time might affect your hearing. The louder the sound, the less time is required before your hearing could be affected. To protect your hearing:

- Limit the amount of time you use headsets or headphones at high volume.
- Avoid turning up the volume to block out noisy surroundings.
- Turn down the volume if you cannot hear people speaking near you.

If you experience hearing discomfort, including the sensation of pressure or fullness in your ears, ringing in your ears, or muffled speech, you should stop listening to the device through your headset or headphones and have your hearing checked.

Danger

Do not use the product if you have implanted electronic medical devices.

The product can emit radio waves that could affect the operation of nearby electronic devices. Use of the product near implanted electronic medical devices, such as a cardiac pacemaker, may cause injury or other serious medical consequences.

Do not use the product if you have a history of photosensitive epilepsy.

The photosensitive epilepsy refers to the severe dizziness, seizures, eye or muscle twitching, or blackouts triggered by light flashes or patterns. About 1 in 4000 people may experience these symptoms while watching TV, playing video games, experiencing virtual reality, or engaging in other activities. These symptoms may occur even if you have no history of seizures or epilepsy. The photosensitive epilepsy is more common in children and young people under 20 years. If you experience any of these symptoms, stop immediately and see a doctor. If you have experienced any of these symptoms, see a doctor before using the product.

Do not use the product if you are sick, tired, uncomfortable, or under the influence of alcohol or drugs.

To have a comfortable virtual reality experience, you must have an unimpaired sense of motion and balance. Use of the product may make you more susceptible to adverse symptoms.

Stop and rest if you experience any discomfort.

Stop and rest until you feel better if you experience any of the following symptoms:

- Seizures, loss of awareness, involuntary movements, muscle twitching, dizziness, disorientation, impaired sense of motion and balance, or impaired hand-eye coordination
- Discomfort or pain in the head, excessive sweating or salivation, nausea, lightheadedness, drowsiness, fatigue, or symptoms similar to motion sickness
- Discomfort or pain in the eyes, eye strain, eye twitching, or other vision abnormalities (such as altered, blurred, or double vision)
- Tingling, burning, numbness, stiffness, or other repetitive stress injuries to your body (such as muscles, joints, or skin)
- Other adverse symptoms due to prolonged wear of the product

These symptoms could persist and become more apparent for hours after use. These post-use symptoms may decrease your ability to perform multiple tasks and increase the risk of injury when you engage in normal activities. So, until fully recovered from these symptoms, do not drive a car, operate a machine, or take part in any activity that may cause serious consequences. See a doctor if you have serious or persistent symptoms.

General notices

Observe the following notices:

- Keep your fingers away from the viewer hinge while pivoting the viewer.
- Keep the product away from children.
- Keep the product away from fire or excessive heat.
- Keep the product away from water or excessive moisture.
- To avoid the risk of electrical shock, protect the product from liquids. Stop using the product if the cables are cracked, broken, or damaged.
- Do not attempt to open, disassemble, or service the product. Unauthorized changes, modifications, or service of the product could void the warranty of the product.

Chapter 1. Introduction

The Lenovo Explorer is an immersive headset. It works with your compatible computer to give you access to an immersive 360-degree virtual world.

The Lenovo Explorer headset is supported by the Mixed Reality Portal application on the Windows® 10 operating system (Windows 10 Fall Creators Update).

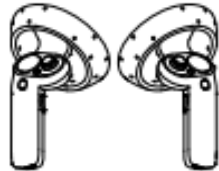
In the box



Lenovo Explorer headset



Documentation



Motion controllers*



AA batteries*

*Available only in the Lenovo Explorer headset and controller combo package

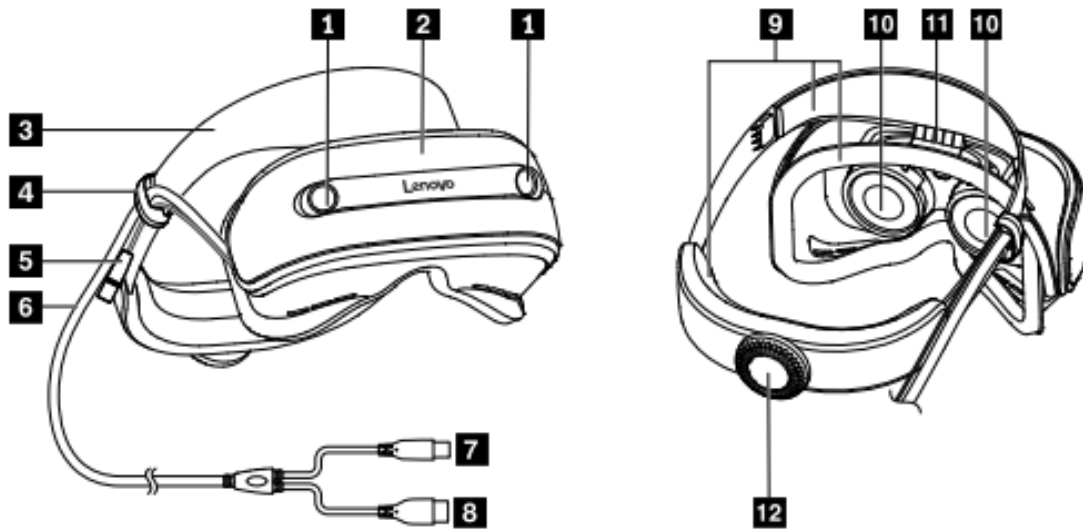
Note: Your products might look slightly different from the illustrations in this document.

Online support information

You can find the following information about the Lenovo Explorer headset at http://www.lenovo.com/support/lenovo_explorer:

- Compatible computer list
- Computer requirements
- Supported language lists for keyboard input, voice input, and voice dictation
- Frequently asked questions
- Documentation

Chapter 2. Parts and locations

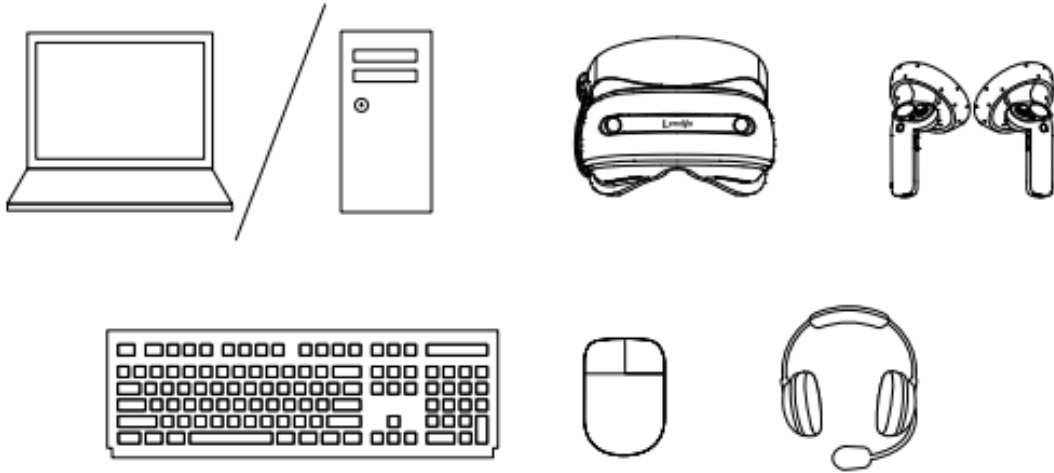


1 Sensors (2)	2 Viewer
3 Head strap	4 Cable clip
5 Headphone connector – 3.5 mm (0.14 inches)	6 Y cable – 4 m (13 feet)
7 Y-cable USB 3.0 connector	8 Y-cable video connector
9 Foam cushions (3)	10 Viewer lenses (2)
11 Viewer hinge	12 Head-strap adjustment knob

Note: If you purchased the Lenovo Explorer headset and controller combo package, refer to “Chapter 6. Enhancing the Lenovo Explorer” for the parts and locations associated with controllers.

Chapter 3. Preparation before initial setup

Prepare the following devices before the initial setup of the Lenovo Explorer headset:



- **Compatible computer**

Turn on the computer.

For detailed information about software and hardware requirements for your computer, go to http://www.lenovo.com/support/lenovo_explorer.

- **Lenovo Explorer headset**

Remove the protective film from each of the viewer lenses.

- **Controllers (optional)**

Connect either of the following controllers to your computer:

- Motion controllers (recommended) for motion tracking and menu control
For detailed information about the motion controllers, refer to "Chapter 6. Enhancing the Lenovo Explorer".
- Windows-compatible controller (such as XBOX ONE® or XBOX 360® controller) for menu control

- **Keyboard and mouse (optional)**

Connect a keyboard and a mouse to your computer to help input data.

- **3.5-mm (0.14-inch) earphones, headphones, or audio headset (optional)**

To listen to the sound from the computer, connect earphones, headphones, or an audio headset to the 3.5-mm (0.14-inch) headphone connector on the Lenovo Explorer headset.

To communicate with the Cortana® personal assistant, use a Cortana-certified audio headset.

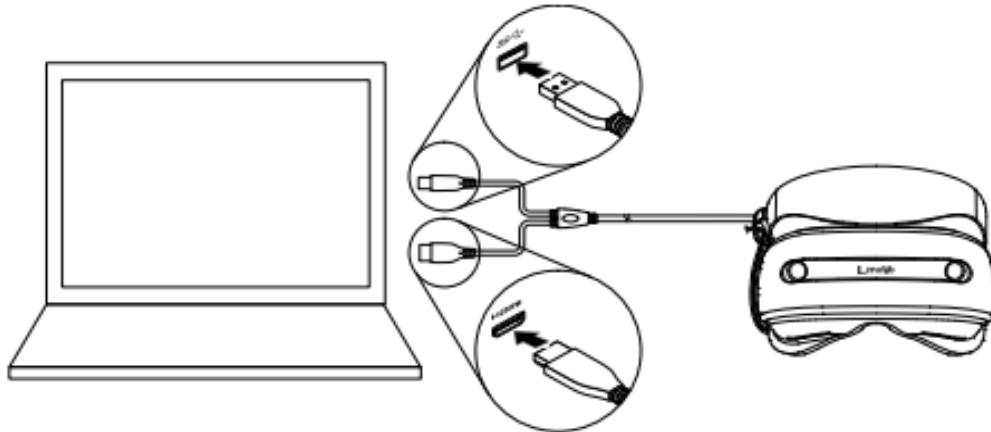
Chapter 4. Initial setup

To set up the Lenovo Explorer headset for the first time, do the following:

1. Connect the Y-cable Universal Serial Bus (USB) 3.0 connector to the USB 3.0 connector (SS) on your computer. Then connect the Y-cable video connector to the High-Definition Multimedia Interface (HDMI™) connector on your computer.

Notes:

- Do not connect the Y-cable USB 3.0 connector to the USB 2.0 connector (USB) on your computer. Otherwise, the Lenovo Explorer headset might not work correctly.
- The maximum allowable distance between the Y-cable connectors is about 38 cm (15 inches). Depending on the distance between the corresponding connectors on your computer, you might need an extension cable, such as a USB 3.0 extension cable, to complete the connection.

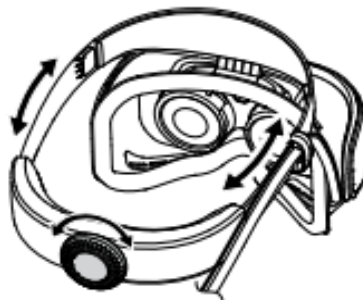


2. Wait several minutes until the Lenovo Explorer driver is installed automatically and the Mixed Reality Portal application launches automatically on your computer. If the application does not launch automatically, you can open the Windows Start menu and click **Mixed Reality Portal** to launch the application manually.

Then follow the instructions on the screen to pair the motion controllers with your computer and complete settings for the Lenovo Explorer headset. For example, the virtual boundary helps guide your movements when you are wearing the Lenovo Explorer headset.

For more information about the motion controllers, refer to "Chapter 6. Enhancing the Lenovo Explorer".

3. Put on the Lenovo Explorer headset.
 - (1) Rotate the head-strap adjustment knob clockwise to tighten or counterclockwise to loosen the head strap depending on your need.

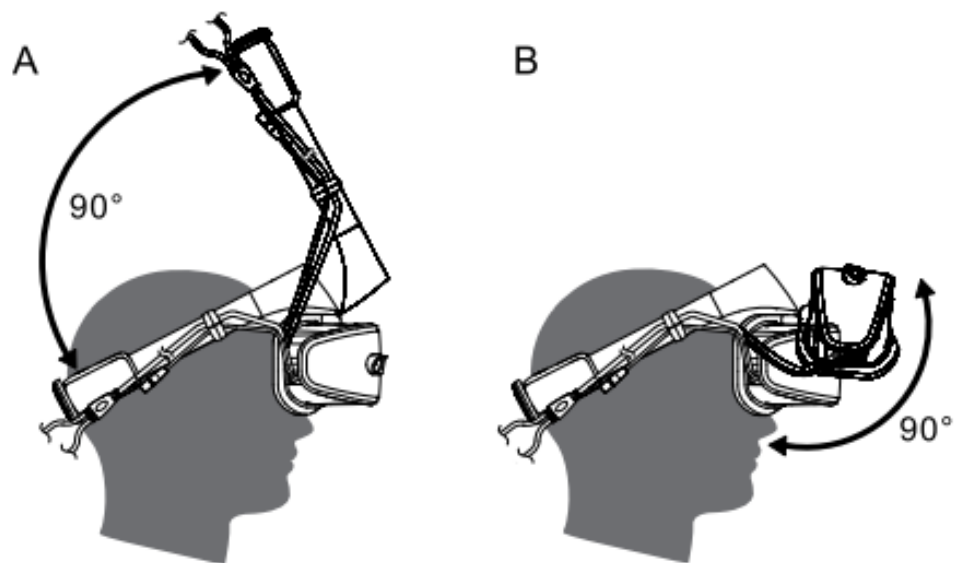


- (2) Use either method to wear the Lenovo Explorer headset:
- Pivot the head strap up, fit the viewer to your eyes, and then pivot the head strap down as shown in the illustration A.
 - Pivot the viewer up, put the head strap on your head, and then pivot the viewer down to cover your eyes as shown in the illustration B.

Attention: When pivoting the head strap or viewer, do not apply excessive force or attempt to adjust the viewer beyond the 90-degree limits.

Then you can slightly adjust the head-strap adjustment knob, head strap, and viewer for a comfortable fit.

To see your physical surroundings without removing the Lenovo Explorer headset, pivot the viewer up as shown in the illustration B.



CAUTION: To avoid pinching your fingers, keep your fingers away from the viewer hinge while pivoting the viewer.



Chapter 5. Using the Lenovo Explorer

After you set up the Lenovo Explorer headset, you will find yourself at the **Windows Mixed Reality** home. This home is a virtual place where you can see a cursor and a Start menu. The cursor is a visualized gaze and it follows the movement of your head. The Start menu provides various applications.

You can stay at the **Windows Mixed Reality** home or open a three-dimensional (3-D) application from the Start menu. Then, you can interact with the virtual world while wearing the Lenovo Explorer headset:

- Target an object by gazing.
- Navigate by turning your head, not your eyes.
- Walk around by moving through physical space.
- Search for, install, move, or launch an application from the Start menu using available devices, such as audio headset, controller, keyboard, and mouse.
- Communicate by voice input or voice dictation using an audio headset.

Note: Certain languages are supported by keyboard input, voice input, and voice dictation. For a list of supported languages, go to http://www.lenovo.com/support/lenovo_explorer. This list is subject to change based on future development and testing.

For more interactions with the virtual world, go to <http://windows.microsoft.com/>, and then search with the keyword windows mixed reality home.

Maintenance

With appropriate care and maintenance, the Lenovo Explorer headset will serve you reliably.

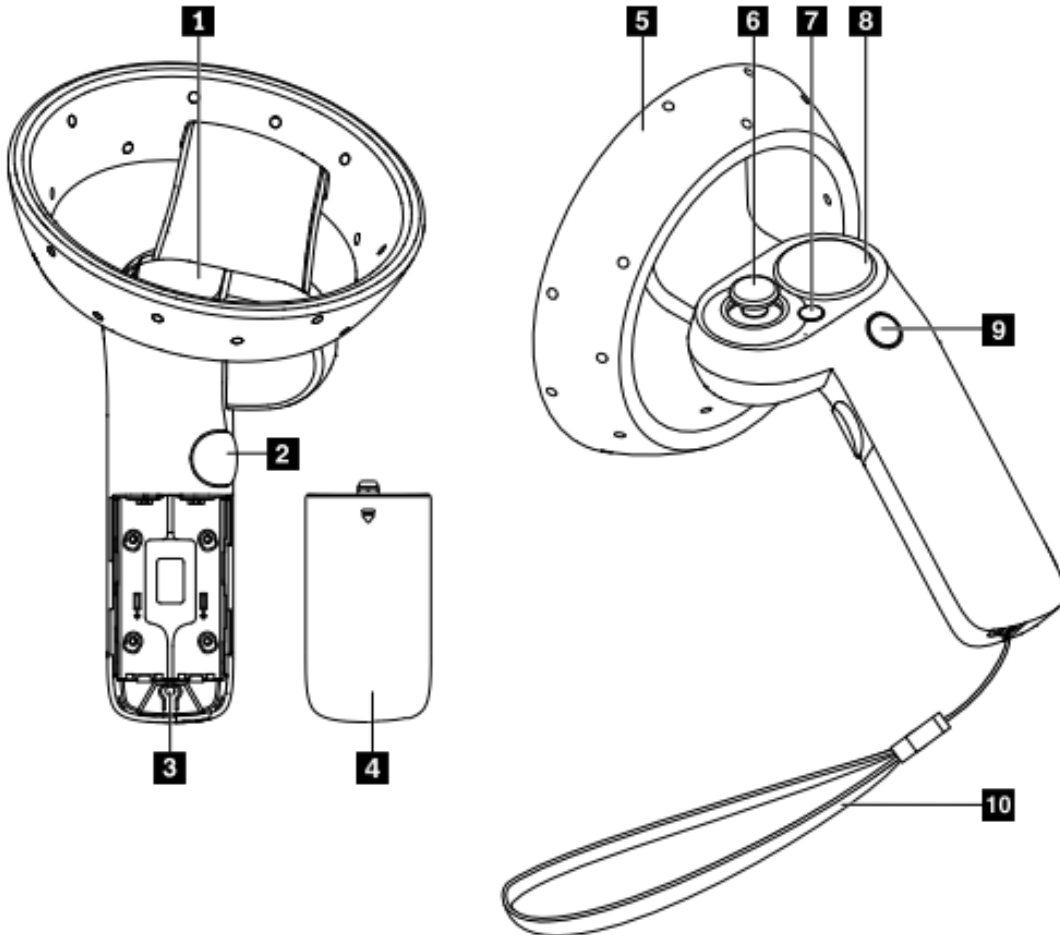
Here are some maintenance tips:

- Do not scratch the sensors and viewer lenses.
- Keep your face dry and clean when wearing the Lenovo Explorer headset.
- If the sensors, viewer lenses, or foam cushions on the Lenovo Explorer headset become dusty or dirty, gently wipe the area with a dry, soft, lint-free cloth, such as an eyeglass cloth. If necessary, moisten the cloth with water or eyeglass cleaner, and then wipe again. Do not apply liquids directly to the sensors, viewer lenses, or foam cushions.

Chapter 6. Enhancing the Lenovo Explorer

The motion controllers help to enhance your Lenovo Explorer headset experience.

Parts and locations

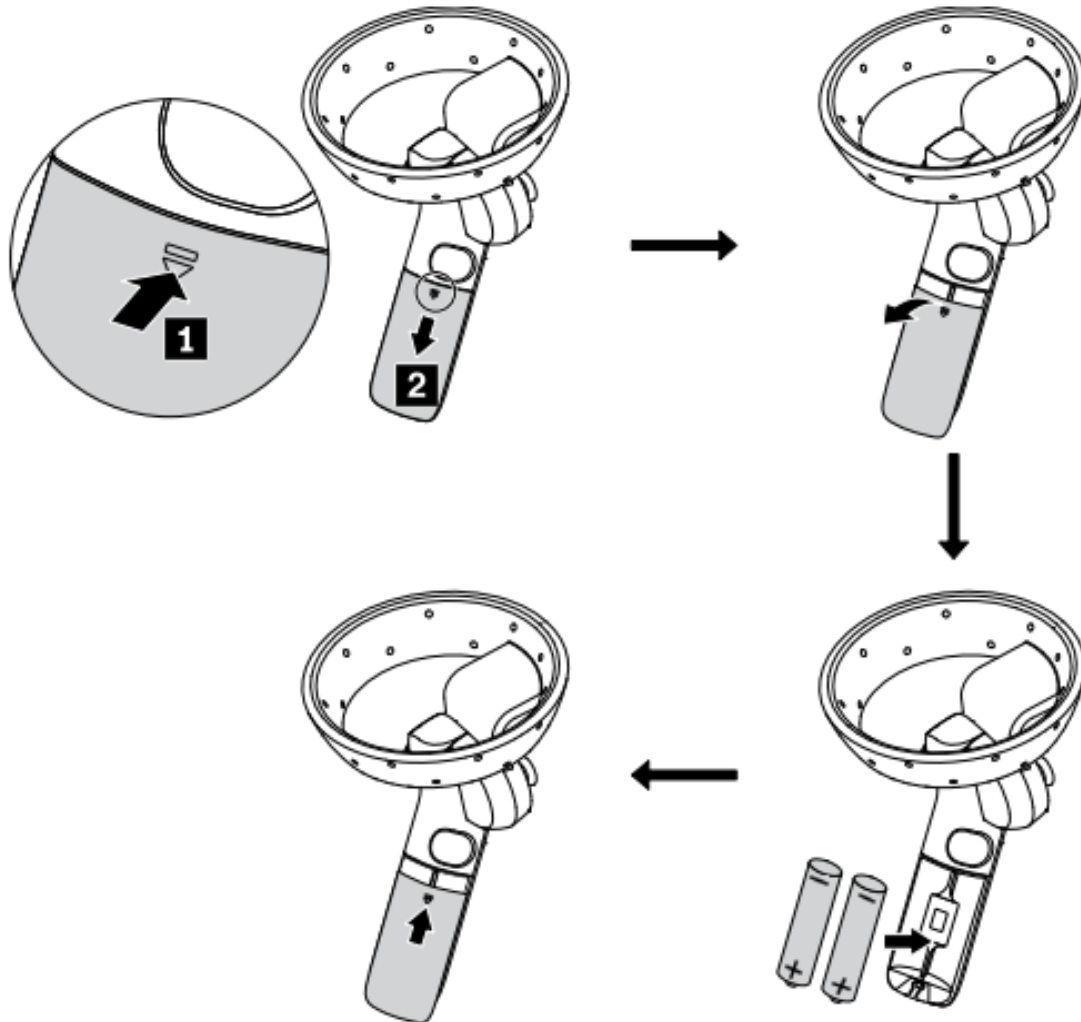


1 Trigger button	2 Grab button
3 Pairing button	4 Battery compartment cover
5 Constellation	6 Thumbstick
7 Menu button	8 Touchpad
9 Windows button	10 Controller strap

Notes:

- To turn on the motion controller, press and hold the Windows button until the constellation lights are on. To turn off the motion controller, press and hold the Windows button until the constellation lights are off.
- When using the motion controllers, it is recommended that you wear the straps on your wrists to avoid dropping the motion controllers by accident.

Installing the AA batteries



For more information about the motion controllers, go to:
http://www.lenovo.com/support/lenovo_explorer

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

*Lenovo (United States), Inc.
1009 Think Place - Building One
Morrisville, NC 27560
U.S.A.*

Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.