

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería



Platyfa: Caso de estudio del desarrollo de un Videojuego en un Entorno Universitario Colaborativo.

Elaboración de animaciones en modelos 3D para un videojuego basado en la historia de Gaus y la “Revolución Ornitoreense”.

Trabajo de graduación presentado por Mariana David Sosa para optar al grado académico de Licenciada en Ingeniería en Ciencia de la Computación y Tecnologías de la información

Guatemala

2024

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería



***Platyfa:* Caso de estudio del desarrollo de un Videojuego en un Entorno Universitario Colaborativo.**

Elaboración de animaciones en modelos 3D para un videojuego basado en la historia de Gaus y la “Revolución Ornitoreense”.

Trabajo de graduación presentado por Mariana David Sosa para optar al grado académico de Licenciada en Ingeniería en Ciencia de la Computación y Tecnologías de la información

Guatemala

2024

Vo.Bo.:

(f) _____

Ing. Anibal Arnaldo Gramajo Lucas

Tribunal Examinador:

(f) _____

(f) _____

(f) _____

Fecha de aprobación: Guatemala, _____ de _____ de 2024

Índice

Lista de figuras	VII
Lista de cuadros	X
Resumen	XII
I. Introducción	1
II. Justificación	4
III. Objetivos	7
1. General	7
2. Específicos	7
IV. Marco Teórico	7
1. Animación 3D	7
1.1 Definición de animación 3D	7
1.2 Tipos y técnicas de animaciones	8
1.3 Principios de animación	11
1.4 Rigging	17
2. Herramientas de animación y programación	19
2.1 Blender	19
2.2 Mixamo	21
2.3 Unity	21
3. Caracterización de personajes en videojuegos	26
3.1 Definición de caracterización de personajes	26
3.2 ¿Cómo la animación ayuda a la caracterización de personajes?	27
3.3 Ejemplos de personajes de videojuegos con buena caracterización	28
4. Animación y jugabilidad	30
4.1 ¿Cómo las animaciones afectan la jugabilidad?	30
4.2 Variables que pueden influir o afectar la jugabilidad.	36
V. Metodología	39
1. Recopilación de Recursos	39

1.2 Investigación y Experimentación con Herramientas de Animación.....	41
1.3 Desarrollo de Animaciones	42
1.4 Evaluación de Calidad y Retroalimentación	57
VI. Resultados.....	59
VII. Análisis de resultados	74
VIII. Conclusiones.....	78
IX. Recomendaciones	81
X. Anexos.....	84
XI. Glosario	89
XII. Bibliografía.....	92

Lista de figuras

Figura 1. Principio “Estirar y encoger”	12
Figura 2. Principio “Anticipación”	12
Figura 3. Principio “Puesta en escena”	13
Figura 4. Principio “Animación directa y pose a pose”	13
Figura 5. Principio “Acción continuada y superpuesta”	14
Figura 6. Principio “Entradas lentas y salidas lentas”	14
Figura 7. Principio “Arcos”	15
Figura 8. Principio “Acción secundaria”	15
Figura 9. Principio “Temporización”	16
Figura 10. Principio “Exageración”	16
Figura 11. Principio “Dibujo sólido”	17
Figura 12. Principio “Apelación”	17
Figura 13. Rigging de personaje con Auto-Rig Pro	21
Figura 14. <i>Animation State Machine</i> en la ventana de Animator	24
Figura 15. Integración del Componente <i>Animator</i>	25
Figura 16. Interfaz de Mixamo	41
Figura 17. Modelo Gaus en posición “T”.	42
Figura 18. Opción de agregación de marcadores para el modelo	43
Figura 19. Colocación de marcadores corporales en el modelo	43
Figura 20. Detección automática y asignación de huesos	44
Figura 21. Pesado de vértices	44
Figura 22. Corrección del esqueleto tras vinculación automática.	45
Figura 23. Opción de agregación de cola	46
Figura 24. Agregación de cola y asignar jerarquía de huesos	46
Figura 25. Búsqueda de animaciones base en Mixamo	47
Figura 26. Construcción de huesos al unificar armadura	48
Figura 27. Animación de disparo descargada de MIXAMO.	49
Figura 28. Animación de disparo mano izquierda.	49
Figura 29. Animación de disparo mano izquierda.	50
Figura 30. <i>Action Editor</i> para colocación de keyframes	50
Figura 31. Establecimiento de rango en el que se ejecuta la animación	51
Figura 32. Resultado de animación en Blender	52
Figura 33. Pantalla del componente Animator con el Game Object como controller	53
Figura 34. Resultados del árbol de estados de animación	54
Figura 35. Configuración de flecha de transición	54
Figura 36. Parámetros colocados	55

Figura 37. Índice para identificar <i>Bools</i> y <i>Triggers</i>	55
Figura 38. Script principal – Ejemplo: <i>GausAnimations</i>	56
Figura 39. Script controlador del personaje Gaus – Ejemplo: Animación de salto	57
Figura 40. Resultados del <i>rigging</i> del cuerpo de Gaus.	60
Figura 41. Resultados del <i>rigging</i> de las manos de Gaus.....	60
Figura 42. Resultados del <i>rigging</i> de las manos de Gaus.....	61
Figura 43. Resultados del <i>rigging</i> del cuerpo de ambos Dingos.....	61
Figura 44. Resultados del <i>rigging</i> de las orejas de ambos Dingos.....	62
Figura 45. Resultados del <i>rigging</i> de las manos de ambos Dingos.....	62
Figura 46. Resultados del <i>rigging</i> de la cola del Buho.	63
Figura 47. Resultados del <i>rigging</i> del Buho.....	63
Figura 48. Resultados del hueso extra para el arma del buho.	64
Figura 49. Resultados de animación KO (morir) de Gaus.	64
Figura 50. Resultados de animación caminar del Dingo normal.	65
Figura 51. Resultados de animación de lanzar arma del Dingo jefe.	65
Figura 52. Resultados de animación extra de volar del Buho.	65
Figura 53. Arbol de transiciones de animaciones de Gaus.....	66
Figura 54. Arbol de transiciones de animaciones de Dingo y Dingo jefe.....	67
Figura 55. Arbol de transiciones de animación del Buho.	67
Figura 56. Script de animaciones de Gaus.	68
Figura 57. Llamada a los métodos correspondientes de animaciones de Gaus.....	69
Figura 58. Llamada a los métodos correspondientes de animaciones de Gaus.....	69
Figura 59. Llamada a los métodos correspondientes de villanos con IA.....	69
Figura 60. Resultado del juego.....	70
Figura 61. Resultados de experiencia visual gracias a las animaciones.....	84
Figura 62. Resultados de la narrativa alcanzada gracias a las animaciones.....	84
Figura 63. Resultados de conexión hacia los personajes gracias a las animaciones.....	85
Figura 64. Resultados de fluides de las animaciones	85
Figura 65. Resultados de identificación de animaciones	85
Figura 67. Resultados de aplicación de principios de animación	86
Figura 68. Resultados de atracción visual gracias a las animaciones	86
Figura 69. Resultados de comprensión de la trama gracias a las animaciones	87
Figura 70. Resultados de credibilidad de contexto gracias a las animaciones	87

Lista de cuadros

Tabla 1. Lista preliminar de las animaciones requeridas por personaje	40
Tabla 2. Resultados de formulario del <i>GameTesting</i>	71

Resumen

El presente trabajo de graduación se enmarca dentro de un megaproyecto orientado a la innovación en la industria de videojuegos en Guatemala, con el objetivo principal de desarrollar un videojuego de ciencia ficción titulado "Gaus y la Revolución Ornitoarenses". Este proyecto se centra en la creación de animaciones 3D caricaturescas que no solo enriquezcan la narrativa del juego, sino que también ofrezcan una experiencia visualmente atractiva para los jugadores.

Para lograr este propósito, se emplearon herramientas líderes en la industria, entre ellas: Blender, Mixamo y Unity. Como software de animación 3D gratuito y de código abierto, se utilizó Blender, para el *rigging* y animación de personajes, mientras que Mixamo facilitó la creación de animaciones a partir de una biblioteca extensa de personajes y movimientos. Unity, por su parte, se encargó de la integración de estas animaciones en el entorno del videojuego, permitiendo la creación de árboles de animación que controlan las transiciones entre diferentes estados de los personajes.

A lo largo del proceso de desarrollo, se enfrentaron diversos desafíos, especialmente en el área de *rigging*, donde se optó por utilizar la herramienta Auto-Rig Pro de Blender para obtener una mayor flexibilidad y precisión en la creación de personajes animados. Las pruebas de usabilidad y *playtesting* revelaron que las animaciones cumplían con los estándares de calidad establecidos, logrando transmitir efectivamente la historia del juego y mejorar la inmersión del jugador.

Las conclusiones de este trabajo destacan la importancia de la capacitación continua en herramientas de animación y la documentación del proceso, así como la necesidad de enriquecer el videojuego. Se presentan recomendaciones para optimizar el uso de recursos en Unity y fomentar la colaboración en el desarrollo de futuros proyectos y futuras implementaciones.

Este trabajo no solo busca explorar la creación de animaciones 3D en un contexto narrativo, sino que también aspira a contribuir al fortalecimiento de habilidades en la industria de videojuegos en Guatemala, abriendo nuevas oportunidades para el desarrollo de productos de alta calidad.

I. Introducción

En el ámbito del desarrollo de videojuegos, la creación de animaciones 3D ha cobrado una relevancia significativa en la narrativa y la experiencia del usuario (Lapo, 2023). Este trabajo de graduación se enmarca dentro de un megaproyecto destinado a la innovación en la industria de videojuegos en Guatemala, cuyo objetivo principal es el desarrollo de un videojuego de ciencia ficción en tercera persona, centrado en la historia de Gaus y la "Revolución Ornitorense". Este megaproyecto, conformado por un equipo multidisciplinario, aspira a ser una experiencia de aprendizaje valiosa para todos los involucrados, así como a generar un producto de alto valor para el portafolio de cada participante, con miras a abrir nuevas oportunidades en la industria.

La creación de animaciones 3D caricaturescas desempeña un papel fundamental en la transmisión de la historia del juego. Para lograr este objetivo, se emplearán herramientas líderes en la industria como Unity, Mixamo y Blender, integrándose en el proceso de desarrollo del videojuego.

Blender, un software de animación 3D gratuito y de código abierto, proporciona una amplia gama de herramientas para modelado, rigging, animación y renderizado. Su comunidad activa de usuarios y desarrolladores lo convierte en una opción versátil y poderosa para artistas y desarrolladores de videojuegos. (Arévalo, 2022)

Mixamo, por otro lado, ofrece una plataforma en línea que facilita el acceso a una extensa biblioteca de personajes y animaciones listas para usar. Gracias a su tecnología de captura de movimiento, Mixamo agiliza la creación de animaciones 3D, permitiendo a los desarrolladores acceder a un conjunto diverso de movimientos y expresiones para sus personajes. (Melo, 2021)

Unity, como motor de juego ampliamente utilizado en la industria, proporciona integración nativa con herramientas de animación como Blender y Mixamo. Con su sistema de animación, Unity simplifica la importación y manipulación de animaciones, así como la creación de árboles de animación para controlar el flujo de estados y transiciones dentro del juego. (Unity, 2018)

Para concluir, este proyecto no solo busca explorar la creación de animaciones 3D, sino que también forma parte de un esfuerzo más amplio por innovar en la industria de los videojuegos en Guatemala, contribuyendo al desarrollo de un producto de calidad y al fortalecimiento de habilidades y capacidades en todos los involucrados.

II. Justificación

El proyecto *Platyfa*: Caso de estudio del desarrollo de un Videojuego en un Entorno Universitario Colaborativo, busca ser un espacio de aprendizaje integral donde los participantes puedan fortalecer sus habilidades y talentos, promoviendo el crecimiento personal y profesional sin importar los roles que desempeñen. Al crear un entorno realista de desarrollo de videojuegos, se ofrecerá a los estudiantes una experiencia cercana a la industria, facilitando la preparación para futuros proyectos. Además, de buscar establecer una base sólida que permita dar visibilidad no solo a este videojuego, sino también a otros en desarrollo o futuros, contribuyendo a la consolidación de un posible estudio de videojuegos a partir de esta experiencia.

Por otro lado, el módulo correspondiente, se justifica por la necesidad de proporcionar una experiencia inmersiva y visualmente atractiva para los jugadores, utilizando animaciones en un modelo 3D para transmitir la narrativa y la lucha por la libertad de los ornitorrincos en un mundo gobernado por villanos.

1. Impacto de las Animaciones en la Experiencia del Usuario

La investigación y análisis de Daniel Burriel han demostrado que las animaciones en los videojuegos juegan un papel crucial en la inmersión del jugador y en la transmisión efectiva de la historia. Animaciones bien diseñadas pueden aumentar la empatía del jugador hacia los personajes, mejorar la comprensión de la trama y aumentar el nivel de disfrute del juego. (Burriel, 2023)

2. Tendencias en la Industria de Videojuegos

La demanda de juegos con narrativas ricas y visuales impresionantes está en constante aumento. Según datos recientes de la asociación *Entertainment software Association (ESA)* “*Essential facts: About the U.S Video Game Industry*”, el 75% de los jugadores consideran que los gráficos son un factor importante al decidir qué juegos jugar. (ESA, 2023) Por lo tanto, la calidad de las animaciones juega un papel crucial en la comercialización y el éxito de un videojuego. (Unity, 2023)

3. Innovación en Herramientas de Animación

Las herramientas de animación como Blender, Mixamo y Unity continúan evolucionando con características avanzadas y mejoras de rendimiento. Mantenerse al día con estas innovaciones permite a los desarrolladores aprovechar al máximo las capacidades de animación y ofrecer experiencias de juego más convincentes y atractivas. (Tuduri, 2024)

III. Objetivos

1. General

Animar modelos 3D de manera caricaturesca para contar la historia de Gaus y la "Revolución Ornitorense" en un videojuego de ciencia ficción, y ofrecer una experiencia visualmente atractiva.

2. Específicos

- Crear animaciones en Mixamo, Blender o herramientas similares.
- Exportar las animaciones a Unity para su integración en el juego.
- Crear un árbol de animación dentro de Unity.
- Realizar las transiciones de las animaciones con el uso de scripts.
- Probar las animaciones en el juego en ejecución y crear un ejecutable para evaluar la calidad de las animaciones.

IV. Marco Teórico

1. Animación 3D

La animación 3D se presenta como una forma de arte y creatividad que emplea herramientas digitales para infundir vida a personajes, objetos y universos fantásticos en una variedad de plataformas, incluyendo películas, series, videojuegos y otros formatos visuales. Su versatilidad permite narrar historias de manera visualmente impactante, además de ofrecer una amplia variedad de estilos artísticos. (Azzan, 2024)

1.1 Definición de animación 3D

El término "animación" proviene del latín *anima*, que significa "alma". En este sentido, el concepto de animar se refiere a dotar de vida o alma a un personaje u objeto, haciéndolo parecer que piensa, actúa y se mueve de manera autónoma. En el caso de la animación 3D, este proceso no solo se enfoca en dar movimiento, sino que también implica la capacidad de girar y manipular los objetos dentro de un espacio tridimensional. (Cortez, 2024)

Por ende, la animación 3D involucra un proceso que permite otorgar movimiento y vida a objetos inanimados dentro de un espacio tridimensional. A diferencia de la animación tradicional en dos dimensiones (2D), la animación 3D se caracteriza por su capacidad de manipular objetos en tres ejes: X, Y y Z, lo que proporciona una sensación de profundidad y realismo. Este tipo de animación se realiza mediante el uso de softwares especializados que permiten modelar y esculpir digitalmente personajes y objetos. A través de estos programas, los diseñadores pueden dotar de personalidad y expresividad a los elementos animados, logrando transmitir emociones y narrar historias. (Torres, 2024)

1.1.1 Aplicaciones de la Animación 3D

La animación 3D se ha convertido en una herramienta fundamental en diversas industrias, transformando la manera en que se crean y visualizan contenidos. En el cine y la televisión, permite la creación de mundos y personajes con un realismo inigualable, como lo demuestra la película "*Toy Story*", pionera en el uso de la animación por ordenador. En los videojuegos, esta técnica ha sido clave para el desarrollo de entornos inmersivos y personajes detallados, como se puede ver en "*The Legend of Zelda: Breath of the Wild*". También ha revolucionado campos como la publicidad, la arquitectura y la educación, ofreciendo soluciones visuales creativas y precisas que mejoran la presentación de productos, la visualización de proyectos arquitectónicos y la comprensión de conceptos complejos. Estas aplicaciones se reflejan en múltiples sectores, mostrando la versatilidad de la animación 3D no solo como herramienta técnica, sino también como un medio para potenciar la creatividad y el impacto visual en diferentes contextos. (F.M, 2023)

Como es notorio, la animación 3D tiene una amplia gama de aplicaciones, ya que su propósito principal es dar vida a objetos inertes. Esta técnica es extremadamente flexible y puede adaptarse a las necesidades específicas de cada sector, utilizando diferentes

metodologías según los objetivos y el contexto de aplicación (Torres, 2024). Algunos de sus usos más destacados incluyen la creación de películas, series animadas, videojuegos, publicidad, efectos visuales (VFX), realidad virtual y aumentada, video mapping, aplicaciones y simulaciones científicas. (Cortez, 2024)

1.1.2 Estilos de Animación 3D

Existen innumerables estilos dentro de la animación 3D, ya que cada artista y cada estudio pueden interpretar y aplicar las técnicas según sus necesidades artísticas o presupuestarias. A pesar de esta variedad, se identifican tres grandes categorías de animación 3D: realista, *cartoon* y *snappy*. (Cortez, 2024).

- **Animación realista:** Este estilo busca emular los movimientos naturales de seres vivos, como humanos o animales. Aunque se puede crear de manera manual, la complejidad del movimiento hace que sea común el uso de sistemas de captura de movimiento (*motion capture*), especialmente en cine y videojuegos (Cortez, 2024).

- **Animación estilo *cartoon*:** Este enfoque simplifica la realidad de manera estilizada y atractiva, utilizando principios básicos de la animación para dar credibilidad a los personajes y escenarios. Ejemplos notables de este estilo se pueden encontrar en películas como *Angry Birds* y *Ugly Dolls*. (Cortez, 2024)

- **Animación estilo *snappy*:** Este tipo de animación es una versión exagerada de la animación *cartoon*, caracterizada por movimientos muy marcados y dinámicos. El uso acentuado de los principios de estiramiento, aplastamiento y exageración es clave en este estilo. Ejemplos incluyen producciones como *The Food Thief* y trabajos de animadores como Jorge Vigara. (Cortez, 2024)

1.2 Tipos y técnicas de animaciones

La animación 3D abarca una variedad de técnicas que se adaptan a distintos objetivos creativos y necesidades de producción. A continuación, se describen algunas de las técnicas más representativas:

- **Motion Design:** El *Motion Design* es una técnica de animación que combina diseño gráfico con movimiento, creando piezas audiovisuales visualmente impactantes. Utiliza elementos como formas, colores, música y efectos sonoros para generar una experiencia sensorial integrada. (Floriano, 2022)

- **Stop Motion:** El *Stop Motion* es una técnica de animación que utiliza objetos físicos, como muñecos o figuras, que son fotografiados fotograma a fotograma mientras se realizan ligeros cambios en su posición. Al juntar las imágenes, se crea

la ilusión de movimiento continuo. Esta técnica, aunque laboriosa, ha sido utilizada durante décadas en cine y televisión, destacando por su encanto artesanal y su capacidad para crear una animación con un estilo único. (F.M, 2023)

- **Pixilación:** Esta técnica puede ser un poco similar a la de *Stop Motion*, aunque se diferencia en que la pixilación no simula el movimiento de objetos o entidades. La pixilación es una técnica que emplea a personas reales como personajes animados. A través de la captura fotográfica de diferentes posiciones de las personas, se puede crear una secuencia en la que parecen moverse de manera inusual o antinatural. Este estilo de animación es útil para lograr efectos visuales que serían difíciles de recrear con animación tradicional o digital, ofreciendo una estética única y creativa. (F.M, 2023) La técnica se dedica a capturar imágenes estáticas para que continuamente, estas sean juntadas a una velocidad de 24 *frames* por segundo, lo que simula el movimiento. (Floriano, 2022)
- **Animación Hiperrealista:** La animación 3D hiperrealista tiene como objetivo recrear imágenes que sean casi indistinguibles de la realidad. Se utiliza para generar personajes, entornos y objetos con un nivel de detalle tan preciso que puede confundirse con imágenes reales. Esta técnica es común en publicidad, cine y videojuegos, especialmente cuando se requiere la recreación de entornos y personajes realistas que interactúan en mundos imaginarios o escenas de acción. (F.M, 2023)
- **Animación de Caricatura:** La técnica de animación 3D de *caricatura* recrea el estilo exagerado de los dibujos animados en un entorno tridimensional. Los personajes suelen tener rasgos distorsionados, proporciones fuera de lo común y movimientos exagerados para producir un efecto humorístico y visualmente llamativo. Esta técnica es utilizada en producciones de entretenimiento, donde la creatividad y el estilo cómico son esenciales. (F.M, 2023) El objetivo de este radica en hacerlos sencillos y divertidos. (Floriano, 2022)

1.2.1 Keyframing

La animación conocida como *keyframe* se refiere a una técnica en la que se introduce “keys” (claves) en *frames* (fotogramas) a lo largo de una línea de tiempo. Esta técnica se distingue de otras formas de animación, como el *stop motion* y el *motion capture*, en su enfoque y metodología.

La animación *keyframe* se utiliza en software de animación 3D y 2D, donde el animador debe tomar decisiones sobre qué claves aplicar y en qué fotogramas hacerlo. En el contexto de la animación de personajes, esto implica seleccionar poses específicas que serán registradas en momentos determinados de la secuencia a animar. (Giménez, 2024)

Los *keyframes* actúan como puntos de referencia dentro del software 3D, definiendo la posición, rotación y escala del objeto animado. A medida que el software interpola entre estas claves, se genera el movimiento del objeto, logrando así el efecto de animación deseado. Para establecer un *keyframe*, lo primero que se debe hacer es ubicarlo en la línea de tiempo del software de animación. En este punto se selecciona el instante en el que el efecto inicializará su transición y se fija el fotograma clave. Hasta esa marcación, la animación avanzará sin modificaciones; los cambios se activarán después de este punto. (Oliveira, 2021)

A continuación, se debe identificar el momento en que el efecto concluirá, lo que se logra mediante la creación de un segundo fotograma clave. Este segundo *keyframe* delimitará el intervalo del movimiento o cambio. Después de establecer ambos puntos, se procederá a modificar los valores deseados que se aplicarán en el intervalo seleccionado, permitiendo así que la animación se desarrolle de manera fluida y coherente. (Oliveira, 2021)

La animación *keyframe* se desarrolla a través de varias fases:

- **Blocking:** En esta etapa, el animador establece únicamente las *keys* más significativas de la secuencia, con el objetivo de contar la historia utilizando la menor cantidad de claves posible. Esta fase es fundamental para sentar las bases de la animación. (Giménez, 2024)
- **Blocking Plus:** En esta fase, el animador añade *keys* adicionales que ayudan a definir con mayor precisión el movimiento de los objetos. Esto permite una mayor claridad en la acción y en la narrativa visual. (Giménez, 2024)
- **Refine:** Durante la etapa de refinamiento, el animador revisa y ajusta las *keys*, añadiendo, eliminando o modificando las claves existentes para lograr un movimiento más pulido y creíble. Esta fase es crucial para asegurar que la animación sea fluida y convincente. (Giménez, 2024)

1.2.2 Motion Capture

Este proceso consiste en registrar el movimiento de seres humanos, animales o elementos físicos y trasladar esos datos a un modelo en dos o tres dimensiones. La captura de movimiento o mejor conocido como *Motion Capture* busca animar estos modelos utilizando los movimientos reales capturados.

Las aplicaciones de esta técnica son amplias y variadas. Uno de los usos más conocidos se encuentra en la industria del entretenimiento, donde se emplean para grabar las acciones de actores y replicarlas en modelos 3D generados digitalmente. Esta práctica permite que tanto las animaciones como los videojuegos alcancen niveles de realismo impresionantes, enriqueciendo así la experiencia del espectador. (De Guzmán, 2023)

Como Andreú menciona en su blog investigativo “*Motion Capture en la industria de la Animación.*” Esta técnica implica que una persona use un traje especial que está equipado

con sensores extremadamente sensibles. Estos sensores están estratégicamente colocados en distintas partes del cuerpo para registrar incluso los movimientos más sutiles. La finalidad de estos sensores es proporcionar a los artistas digitales, como animadores 3D y productores de efectos visuales, datos precisos y realistas sobre los movimientos del actor que lleva el traje. La información recopilada sirve como una base para los movimientos animados, ya que incluye detalles sobre el tiempo de cada acción, lo que facilita el proceso de postproducción.

El proceso de motion capture comienza con la creación de un modelo 3D por parte de los artistas digitales. Una vez que el modelo está diseñado, pasa al departamento de rigging, donde se le otorgan las estructuras necesarias para que pueda ser deformado y animado. Con el modelo listo, se lleva a cabo la captura de movimiento, donde los movimientos del actor se registran en tiempo real y se vinculan al modelo 3D. Esto permite que el modelo replique los movimientos del actor de manera instantánea, creando una animación más realista. (Andreu, 2022)

Después de realizar la captura de movimiento, los datos obtenidos se envían al departamento de animación, donde los animadores trabajan en perfeccionar las secuencias. En esta etapa, se crean *keyframes* y se ajustan los tiempos de los movimientos para asegurar una fluidez adecuada en las animaciones. Este meticuloso proceso, que combina la captura de movimiento con la animación y la expresión emocional, es fundamental para dar vida a las escenas que vemos en el cine hoy en día.

Una vez que el modelo 3D está listo, los artistas de efectos vinculan los movimientos registrados por el traje de motion capture a dicho modelo. Esto significa que, a medida que el actor se mueve, el modelo 3D responde en tiempo real, imitando esos movimientos de manera precisa. Posteriormente, la información es enviada al departamento de animación, donde comienza el proceso de refinamiento de las animaciones obtenidas. Durante esta fase, se generan *keyframes* y se desglosan las animaciones, refinando cada movimiento y ajustando los tiempos correspondientes para lograr una representación natural y fluida. (Andreu, 2022)

Las escenas que observamos en el cine son el resultado de un meticuloso proceso de captura de movimiento, animación y expresión emocional. Un ejemplo reconocido a nivel mundial de la captura de movimiento es "Avatar", una película dirigida por James Cameron, que se estrenó en 2009. Esta película es célebre por su innovador uso de tecnología de captura de movimiento y efectos visuales, lo que le permitió crear un mundo completamente nuevo y realista. (Chisaba,2021)

En la actualidad, el uso del motion capture se ha vuelto cada vez más común en diversas áreas del entretenimiento. A medida que la tecnología avanza e inclusive haciendo uso de la inteligencia artificial, también se está aplicando en campos como la ciencia y la biomecánica, entre otros.

1.3 Principios de animación

Como bien se conoce la animación es el arte de dar vida a objetos inanimados, creando movimiento a través de diversas técnicas como el dibujo, la pixilación, el stop motion y la

animación digital en 2D y 3D. Estas técnicas permiten contar historias y transmitir incluso emociones. (Lasseter, 1987)

El primer texto que establece una base conceptual sólida sobre los principios de la animación se publicó en 1981 por Ollie Johnston y Frank Thomas, y se considera fundamental para cualquier artista en este campo. Sin embargo, existen variedad de autores como John Lasseter y otros, que han ampliado y adaptado estos conceptos a la animación digital. (Manuel, 2015).

Los 12 principios de la animación, inicialmente diseñados para aumentar el realismo, siguen siendo relevantes en la actualidad. Estos principios, que se describen en la tesis doctoral de Cuesta Martínez (2015), incluyen:

- Estirar y encoger (*squash and stretch*): Deformar los objetos para añadir dramatismo y transmitir su peso y flexibilidad, observable en la Figura 1.

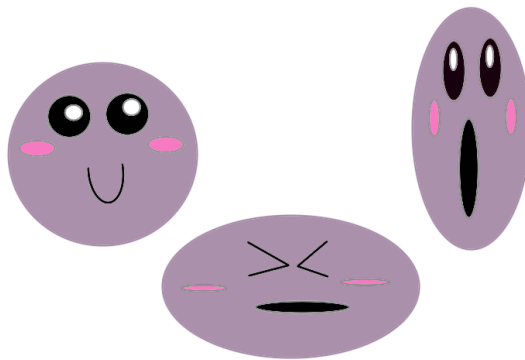


Figura 1. Principio “Estirar y encoger”

- Anticipación (*anticipation*): Preparar al espectador para una acción, guiando su atención hacia el lugar donde ocurrirá, observable en la Figura 2.

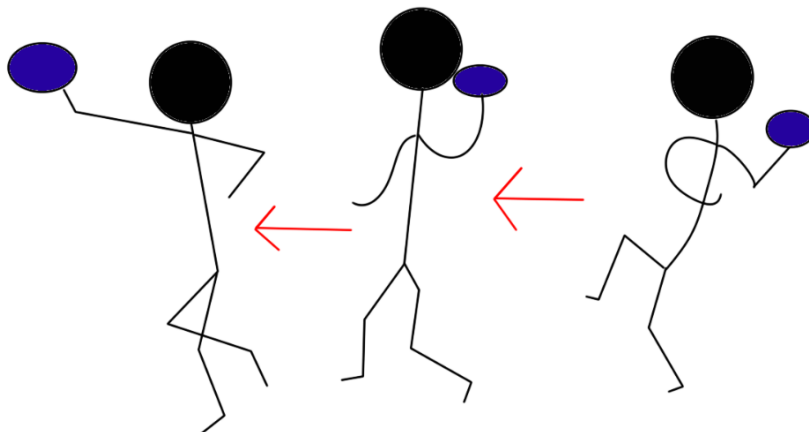


Figura 2. Principio “Anticipación”

- Puesta en escena (*staging*): Disposición de objetos y personajes para que la acción sea fácilmente comprensible, observable en la Figura 3.

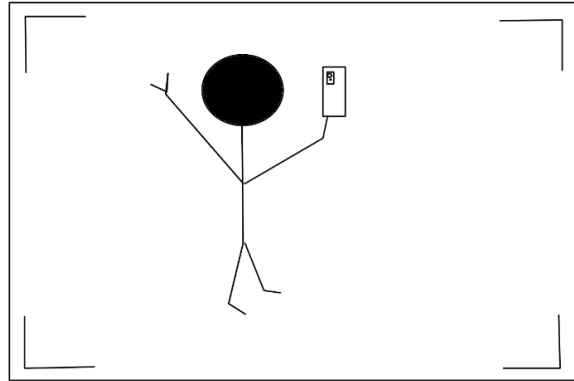


Figura 3. Principio “Puesta en escena”

- Animación directa y pose a pose (*pose to pose*): La animación directa se crea fotograma a fotograma de manera fluida, mientras que pose a pose implica definir primero los fotogramas clave y luego crear los movimientos intermedios, observable en la Figura 4.

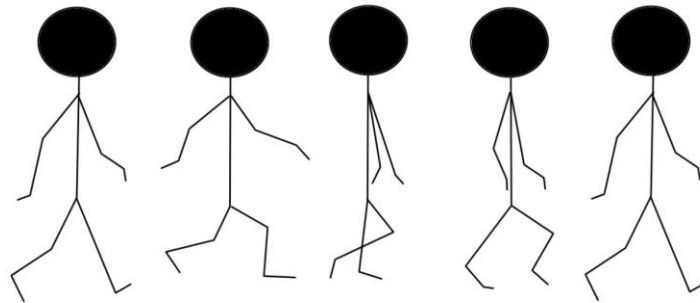


Figura 4. Principio “Animación directa y pose a pose”

- Acción continuada y superpuesta (*follow through and overlapping action*): Este principio asegura que las acciones del personaje se mantengan coherentes y realistas, mostrando que los movimientos siguen las leyes de la física, observable en la Figura 5.

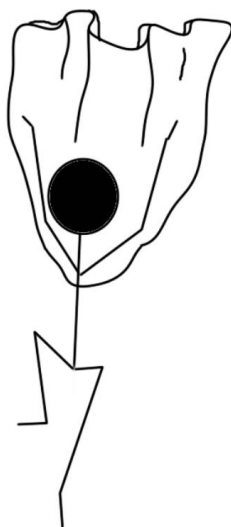


Figura 5. Principio “Acción continuada y superpuesta”

- Entradas lentas y salidas lentas (*slot in and slow out*): Los personajes u objetos necesitan tiempo para acelerar al inicio y desacelerar al final de una acción, ver en la Figura 6.

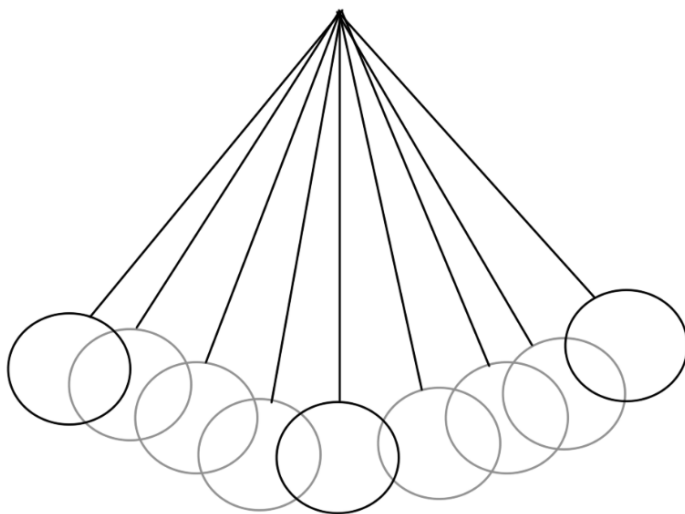


Figura 6. Principio “Entradas lentas y salidas lentas”

- Arcos (*arc*): Las acciones naturales se describen mediante trayectorias en arco, como el movimiento de los brazos o el salto, observable en la Figura 7.

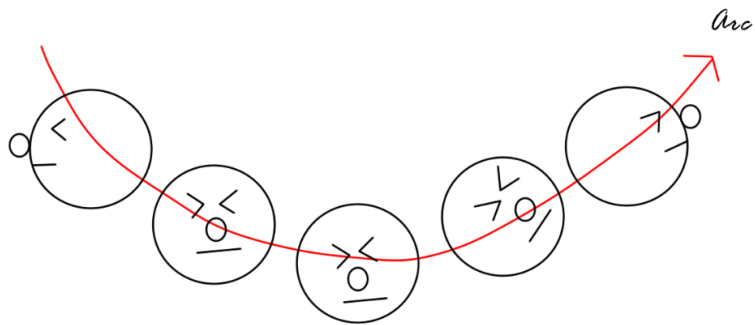


Figura 7. Principio “Arcos”

- Acción secundaria (*secondary action*): Agregar movimientos sutiles que complementen la acción principal, haciendo que esta sea más comprensible y natural, ver en la Figura 8.

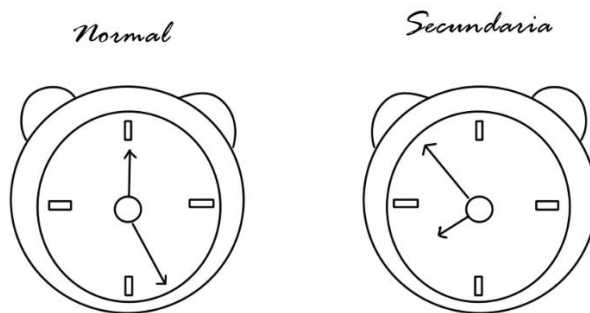


Figura 8. Principio “Acción secundaria”

- Temporización (*timing*): Se refiere a la cantidad de fotogramas utilizados en una acción, influyendo en la percepción de peso, velocidad y expresión, observable en la Figura 9.

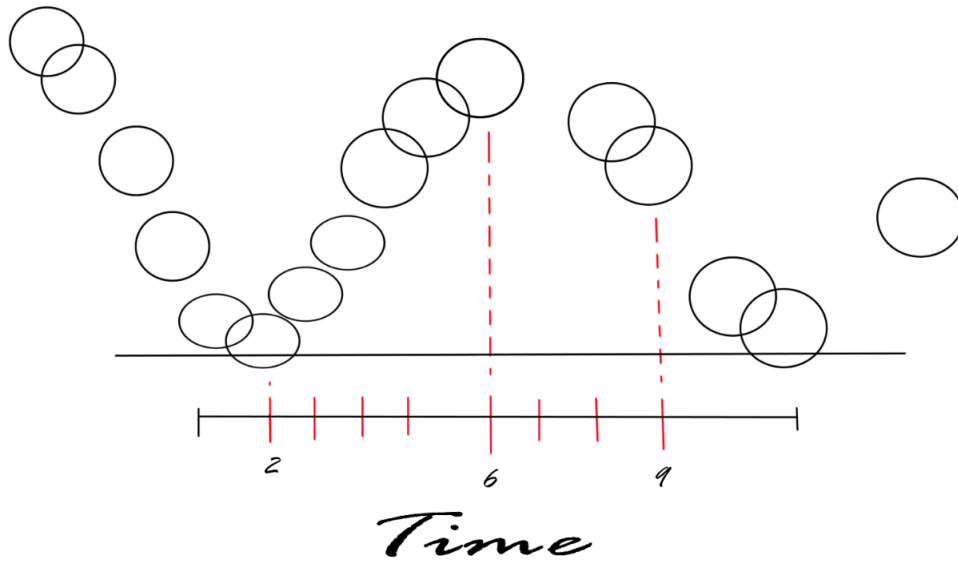


Figura 9. Principio “Temporización”

- Exageración (*exaggeration*): Ampliar los movimientos de los personajes para lograr un efecto dramático, observable en la Figura 10.

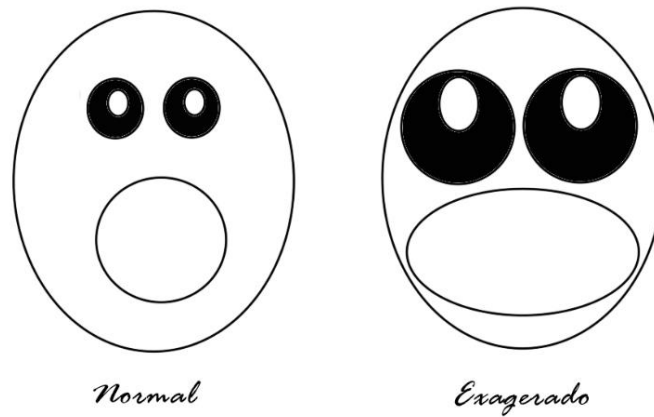


Figura 10. Principio “Exageración”

- Dibujo sólido (*solid drawing*): Presentar a los personajes de manera coherente, manteniendo profundidad y perspectiva durante todas sus acciones, ver en la Figura 11.

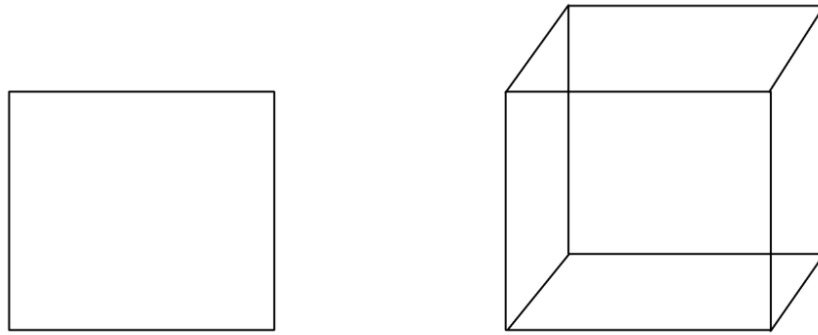


Figura 11. Principio “Dibujo sólido”

- Apelación (*appeal*): Los personajes deben tener una personalidad que conecte con el espectador, logrando una distinción basada en su apariencia, vestimenta y movimientos, observable en la Figura 12.

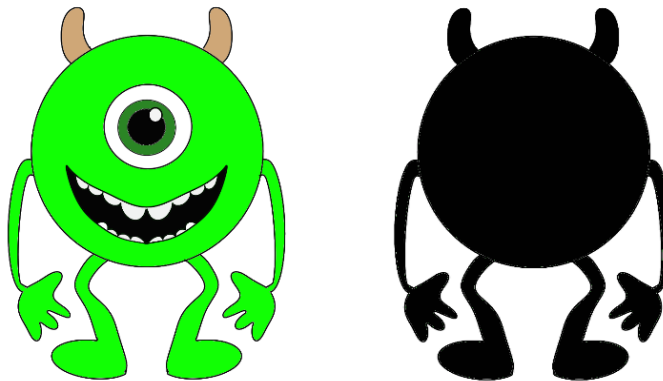


Figura 12. Principio “Apelación”

Estos principios, utilizados por los animadores de Disney desde finales de los años 20 y 30, son fundamentales para dar vida a los personajes y escenas animadas (Lowe & Schnotz, 2014; Pertíñez López, 2014).

1.4 Rigging

Este proceso es importante en la animación ya que facilita la creación de un esqueleto digital o "*rig*" que permite controlar los movimientos de un modelo 3D, facilitando la animación de personajes u objetos complejos.

1.4.1 ¿Qué es el rigging?

El rigging de personajes transforma un modelo 3D o 2D en una marioneta digital manipulable al agregarle una estructura esquelética de huesos o articulaciones, a la que

luego se une el modelo. Este esqueleto permite controlar los movimientos, y cuanto más complejo sea el *rig*, mayor sutileza se logrará en la animación. El artista de rigging también diseña un sistema de controles que facilita al animador manipular el personaje, ajustando la deformación del modelo para que se vea natural al moverse. Así, se logra un personaje visualmente atractivo y funcional para la animación. (Unity, 2024)

Esta es una técnica esencial para animar personajes, criaturas u objetos en cine, televisión y videojuegos, permitiendo que cualquier elemento pueda ser articulado para moverse. Con herramientas de software 3D, es posible automatizar movimientos complejos. Para aplicar el rigging, los animadores comienzan con un modelo 2D o 3D y diseñan una estructura jerárquica (esqueleto) y un conjunto de controles, decidiendo previamente cómo se moverá y en qué puntos se colocarán las articulaciones. (Unity, 2024)

1.4.2 ¿Importancia de hacer un *rigging*?

Para crear un buen *rig*, es necesario tener un boceto o modelo 3D del personaje u objeto a animar, que servirá de base para desarrollar una estructura esquelética adaptada a su forma. A este modelo se le añade una serie de huesos y articulaciones que le permitirán moverse. Programas como Blender, Maya y Autodesk 3ds Max son comunes para el *rigging* 3D, y Spine se usa para 2D. Un buen *rig* destaca por su precisión anatómica, rapidez de respuesta y facilidad de uso, facilitando el trabajo del animador y evitando problemas de movilidad o movimientos poco naturales. (Campus, C, 2024a)

El *rigging* es esencial en la animación 3D por varias razones clave:

1. **Movimiento natural:** Permite que los personajes se muevan de forma realista, imitando expresiones y gestos humanos, lo cual es vital para conectar emocionalmente con el público.
2. **Eficiencia en la animación:** Facilita el proceso al permitir que los animadores manipulen el modelo a través de esqueletos y controladores, en lugar de mover cada vértice manualmente, ahorrando tiempo y esfuerzo.
3. **Consistencia visual:** Un buen *rig* mantiene las proporciones y la apariencia del personaje, evitando deformaciones no deseadas y asegurando coherencia en la animación.
4. **Flexibilidad y ajustes:** Permite realizar correcciones en los movimientos sin cambiar el modelo completo, gracias a los controladores que ajustan el esqueleto.
5. **Interacción con el entorno:** Permite que los personajes interactúen con objetos y el ambiente, haciendo posibles movimientos complejos como sujetar objetos o caminar en superficies irregulares. (Superior, E. F. P, 2023)

2. Herramientas de animación y programación

En la actualidad, existe una gran variedad de programas tanto de modelado y animación en 3D que permiten crear objetos tridimensionales y generar imágenes fotorrealistas. Sin embargo, son pocos los que combinan tres características esenciales que resultan muy atractivas para los usuarios: ocupan poco espacio en el disco duro al ser instalados, requieren escasos recursos del ordenador y son de código abierto, lo que significa que su código fuente está disponible para el público; permitiendo a cualquier persona ver, modificar y distribuir el software.

2.1 Blender

Blender es un software gratuito y de código abierto que se utiliza para generar contenido en 3D. Esta herramienta todo en uno combina funciones de modelado, animación, simulación, edición de vídeo, creación de videojuegos y más. Su capacidad para integrarse con otras aplicaciones lo hace muy atractivo para diversas industrias, incluyendo la animación, el cine, el diseño de videojuegos y la visualización arquitectónica. (Campus.C, 2024)

Blender cuenta con diversas aplicaciones ya que es un programa versátil que encuentra uso en múltiples campos creativos y técnicos. A continuación, se presentan algunas de sus principales aplicaciones:

- **Modelado 3D:** Permite la creación de modelos tridimensionales de objetos, personajes y entornos, abarcando tanto el modelado poligonal (basado en vértices y aristas) como el modelado basado en curvas y superficies.
- **Animación:** Facilita la animación de personajes, la pixilación y la animación por fotogramas clave, así como el rigging para la configuración de esqueletos y animaciones físicas. Su sistema de armaduras y controladores ayuda a lograr movimientos realistas y dinámicos.
- **Simulación:** Ofrece diversos motores de simulación para crear efectos físicos realistas, como fluidos, humo, fuego, telas, partículas y cuerpos rígidos.
- **Creación de videojuegos:** Integra un motor de videojuegos (Blender Game Engine - BGE) que permite desarrollar juegos interactivos directamente en el software.
- **Realidad virtual y aumentada:** Gracias a su capacidad de exportar en formatos compatibles con dispositivos de VR/AR, Blender se ha consolidado como una herramienta esencial en este campo emergente. (Campus.C, 2024)

Además, Blender es compatible con todos los sistemas operativos, lo que significa que puede ser usado sin inconvenientes en Mac, Windows o Linux.

Características destacadas:

- Herramientas potentes para renderizado, modelado y escultura.
- Funciones avanzadas para animación y *rigging*.
- Capacidad para diseñar y editar en un entorno 3D.
- Edición de vídeo y efectos visuales (VFX), que incluyen seguimiento de movimiento, enmascaramiento y composición.
- Herramientas de simulación que ofrecen efectos realistas.
- Amplia API de Python para personalizar y desarrollar scripts.
- Interfaz de usuario y diseño de ventanas personalizables, así como accesos directos ajustables. (Sheikh, M., & Sheikh, M., 2024)

2.1.1 Auto Rig Pro

Auto-Rig Pro es una herramienta que sirve como vía para extender las funcionalidades de Blender; conocidos como *Add-ons*. Este es altamente reconocido para el *rigging* de personajes en Blender, diseñado para optimizar el proceso de creación de esqueletos y controladores. Este complemento se distingue por su algoritmo inteligente, que permite a los artistas 3D configurar *rigs* (esqueleto del personaje) con unos pocos clics, sin comprometer la precisión y el control necesarios para proyectos de animación de alta calidad. Su compatibilidad con plataformas de animación, como Mixamo, lo convierte en una herramienta versátil y eficiente. (Fernando, & Fernando, 2024)

Los beneficios que se obtienen al utilizar Auto-Rig Pro son significativos. En primer lugar, su interfaz rápida y fácil de usar permite realizar un *rigging* a cualquier personaje en cuestión de minutos, lo que ahorra tiempo valioso en el flujo de trabajo del artista. Además, a pesar de ser un proceso automatizado, el *rig* resultante es de calidad profesional, listo para la animación y con una gran precisión en los movimientos. La herramienta también facilita la integración de animaciones externas, lo que permite incorporar recursos adicionales sin complicaciones. (Fernando, & Fernando, 2024)

Con Auto-Rig Pro, los artistas no solo optimizan su flujo de trabajo, sino que también mejoran la calidad de sus proyectos, haciendo de esta herramienta una opción indispensable para aquellos que buscan agregar movimiento a sus personajes 3D de manera efectiva y eficiente. Al ser accesible de forma gratuita, permite que los usuarios comiencen a trabajar de inmediato, maximizando así su productividad y creatividad en el proceso de animación. (Fernando, & Fernando, 2024)

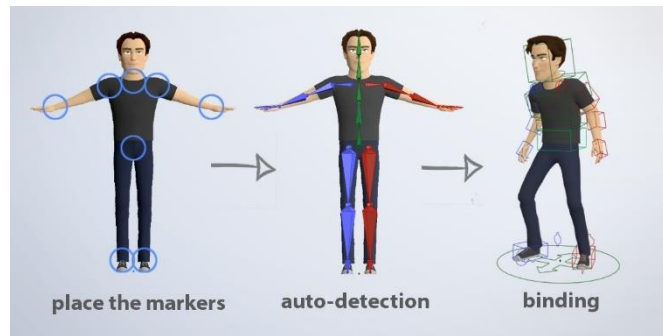


Figura 13. Rigging de personaje con Auto-Rig Pro

2.2 Mixamo

Mixamo es una herramienta creada por Mixamo Inc., una compañía fundada en 2008. Adobe Mixamo es una plataforma innovadora para la animación en 3D que permite a los usuarios crear y diseñar personajes tridimensionales para diversas aplicaciones, como películas, videojuegos y experiencias interactivas. Ofrece una amplia biblioteca de personajes 3D de alta calidad, completamente texturizados, y animaciones de cuerpo completo obtenidas a través de la captura de movimiento de actores profesionales. (Moody, A., 2009)

Una de las características más destacadas de Mixamo es su capacidad para cargar cualquier modelo 3D, al que se le aplica automáticamente un esqueleto humano, ajustándose según sus características específicas. Este proceso simplifica el *rigging*, permitiendo a los artistas centrarse en la animación y el diseño sin complicaciones técnicas.

Además, Mixamo facilita la creación y animación rápida de personajes, incluso para aquellos que no cuentan con un conocimiento profundo de animación. La plataforma ofrece un método ágil y personalizable, donde los usuarios pueden mezclar diferentes clips de movimiento de su biblioteca y redirigirlos a otros personajes en aplicaciones como Cinema 4D.

Finalmente, Mixamo permite exportar personajes y animaciones en múltiples formatos compatibles con plataformas como Unity, Blender, entre otros, lo que lo convierte en una herramienta valiosa para desarrolladores y animadores que buscan integrar personajes animados de manera eficiente en sus proyectos. (Maxon, 2024)

2.3 Unity

Este software cuenta con una serie de rutinas de programación que facilitan el desarrollo de aplicaciones interactivas que aborda múltiples herramientas para poder trabajar modelación y animación.

2.3.1 Definición de Unity

Unity, anteriormente conocido como Unity 3D, es un software integral que reúne todas las herramientas necesarias para el desarrollo de videojuegos, permitiendo a los usuarios crear juegos para una variedad de dispositivos, incluyendo PC, consolas y dispositivos móviles. Esta plataforma ofrece un editor visual y la posibilidad de programar a través de scripting, lo que posibilita lograr resultados altamente profesionales. Ejemplos notables de juegos desarrollados con Unity incluyen títulos populares como “*Monument Valley*” y “*Cuphead*”, y también es muy utilizada en el desarrollo de videojuegos móviles. Cabe mencionar que Unity es un motor de desarrollo, también conocido como motor de juegos, que permite diseñar, crear y ejecutar entornos interactivos, es decir, videojuegos (Martínez, 2023).

Uno de los principales atributos de Unity es su amplia comunidad de usuarios, lo que brinda acceso a una gran cantidad de documentación, foros y grupos de discusión donde se pueden plantear y resolver dudas. Esta comunidad también facilita el aprendizaje de nuevas técnicas y métodos. Además, Unity se considera una excelente puerta de entrada para quienes desean comenzar en la industria del desarrollo de videojuegos. (MasterD, 2019)

Entre las funcionalidades más comunes que ofrece un motor de videojuegos como Unity se encuentran:

- Un motor gráfico que permite renderizar gráficos en 2D y 3D.
- Un motor físico que simula las leyes de la física en el entorno del juego.
- Herramientas para la creación de animaciones.
- Capacidades para la gestión y reproducción de sonidos.
- Integración de inteligencia artificial para el comportamiento de los personajes y elementos del juego.
- Opciones de programación o scripting para personalizar la lógica del juego y las interacciones. (MasterD, 2019)

2.3.1 Mecanim

Unity cuenta con un sistema de animación avanzado conocido como *Mecanim*, que ofrece una serie de características y funcionalidades para facilitar la creación y gestión de animaciones. Entre las principales características se incluyen:

- Flujo de trabajo simplificado: Proporciona un proceso de animación sencillo para todos los elementos en Unity, incluyendo objetos, personajes y propiedades.
- Compatibilidad con clips de animación: Soporta tanto clips de animación importados como aquellos creados directamente en Unity.
- *Retargeting* de animaciones humanoides: Permite aplicar animaciones de un modelo de personaje a otro, lo que simplifica la reutilización de animaciones.

Generalmente se maneja un flujo de trabajo de la animación. El sistema de animación de Unity se basa en el concepto de *Animation Clips*, los cuales contienen información sobre cómo deben cambiar la posición, rotación y otras propiedades de ciertos objetos a lo largo del tiempo. Cada clip puede considerarse como una grabación lineal. Los clips de animación pueden ser creados por artistas o animadores utilizando herramientas externas como 3ds Max o Maya, o pueden provenir de estudios de captura de movimiento y otras fuentes. (Technologies. U ,2016)

Entre las funcionalidades adicionales que ofrece el sistema de animación de Unity se incluyen:

- Previsualización conveniente: Permite a los animadores previsualizar clips de animación, transiciones e interacciones, facilitando el trabajo independiente de los programadores y la creación de prototipos antes de integrar el código del juego.
- Herramientas visuales para interacciones complejas: Ofrece una herramienta de programación visual para gestionar interacciones complicadas entre animaciones.
- Animación segmentada: Permite animar diferentes partes del cuerpo con lógicas diferentes.
- Capas y enmascaramiento: Incluye características de capas (*layering*) y enmascaramiento (*masking*) para una mayor flexibilidad en la animación. (Technologies. U ,2016)

2.3.2 Animator y Animations en Untity

El componente *Animator* se utiliza para asignar animaciones a un *GameObject* (Objeto vacío) dentro de una escena en Unity. Este componente necesita una referencia a un *Animator Controller*, el cual define qué clips de animación se utilizarán y controla el momento y la forma en que se mezclan y realizan transiciones entre ellos.

Los Animation Clips se organizan en un sistema similar a un diagrama de flujo llamado *Animator Controller* o *Animation State Machine*. Este controlador actúa como una "máquina de estados" que rastrea qué clip debe estar reproduciéndose en cada momento y cuándo deben cambiar o mezclarse las animaciones.

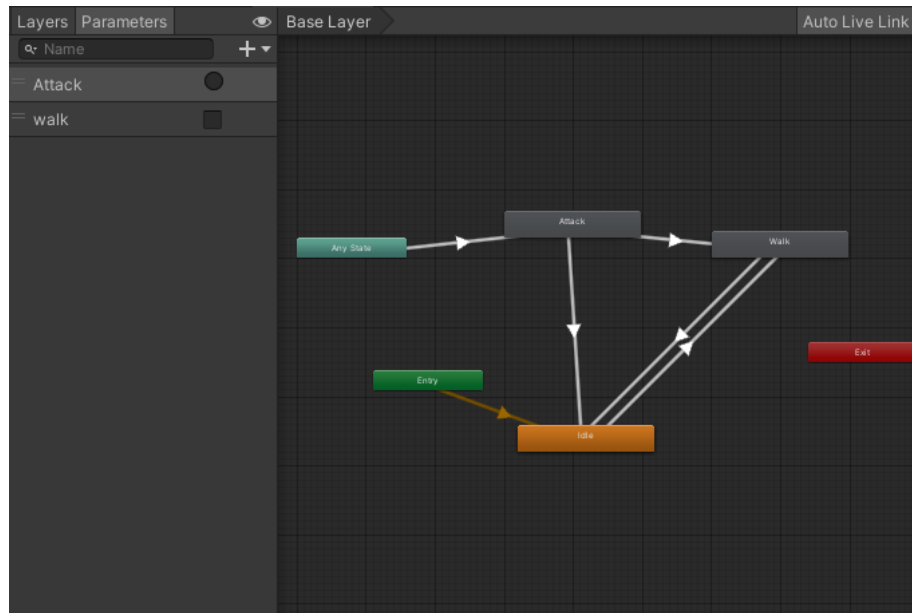


Figura 14. *Animation State Machine* en la ventana de Animator

Un Animator Controller simple podría contener solo uno o dos clips, por ejemplo, para controlar un efecto de giro o una puerta que se abre y cierra. Sin embargo, un Animator Controller más complejo puede manejar múltiples animaciones humanoides, permitiendo la mezcla de varios clips al mismo tiempo para lograr movimientos fluidos mientras el jugador se desplaza por la escena. (Unity Technologies, 2016)

El sistema de animación de Unity también incorpora características especiales para manejar personajes humanoides, permitiendo la adaptación a modelos de personajes propios, además de ajustar las definiciones musculares. Estas características son gestionadas por el sistema de Avatar de Unity, que mapea los personajes humanoides a un formato interno común.

Cada una de estas piezas, como el *Animator Controller* se integran en un *GameObject* a través del *Animator Component*. Este componente referencia un *Animator Controller* y, si es necesario, el Avatar correspondiente para el modelo. A su vez, el *Animator Controller* contiene referencias a los *Animation Clips* que utiliza. (Technologies. U ,2016)

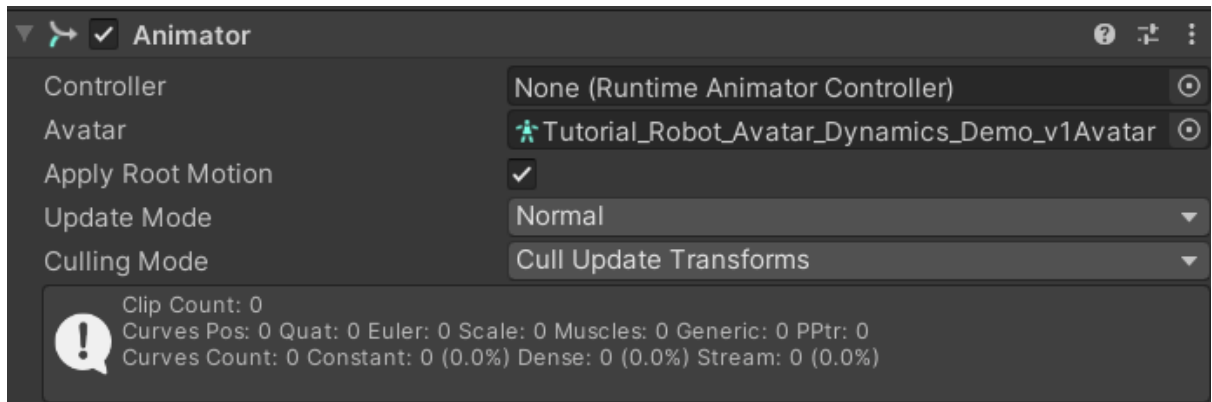


Figura 15. Integración del Componente *Animator*

El componente *Animator* cuenta con varias propiedades, ver Figura 15, clave que permiten un control preciso sobre la animación de un *GameObject*. A continuación, se detallan estas propiedades y sus funciones:

- **Controller:** Esta propiedad se refiere al *Animator Controller* que está adjunto al personaje. El *Animator Controller* define cómo se gestionan y se organizan los clips de animación.
- **Avatar:** Esta propiedad se utiliza para asignar el Avatar al personaje, especialmente si el *Animator* se está utilizando para animar un personaje humanoide. Este permite que las animaciones se adapten correctamente a las características del modelo 3D.
- **Apply Root Motion:** Esta propiedad determina si se debe controlar la posición y rotación del personaje desde la animación misma o desde un *script*; brindando mayor flexibilidad al decidir de qué manera interactuar con el movimiento del modelo.
- **Update Mode:** Esta propiedad permite seleccionar cuándo se actualiza el *Animator* y qué escala de tiempo debe utilizar.

Estas propiedades permiten personalizar y optimizar la animación de personajes en Unity, mejorando la experiencia general del desarrollo de videojuegos. (Technologies. U ,2015)

2.3.3 Integración de animaciones en Unity

El sistema de animación de Unity permite que las transiciones entre estados de control se animen completamente, utilizando el potente sistema de animación disponible en la plataforma. Este enfoque es uno de los métodos más flexibles y avanzados, ya que permite animar múltiples propiedades simultáneamente durante las transiciones. Para aplicar este

tipo de transiciones, es necesario adjuntar un componente *Animator* al objeto controlado. Esto puede realizarse de manera automática.

A diferencia de otros controladores de animación, este en particular almacena también las animaciones relacionadas con las transiciones del controlador, permitiendo que estas sean personalizadas si es necesario. El *Animator* se encarga de manejar la animación de la transición entre estos estados.

Cualquier cantidad de propiedades del objeto puede configurarse en este keyframe, permitiendo gran flexibilidad en el control de las animaciones. (Technologies. U , 2016)

3. Caracterización de personajes en videojuegos

Los personajes son elementos fundamentales en muchos videojuegos, ya que contribuyen a dar vida a la narrativa y permiten que los jugadores se identifiquen o interactúen con el mundo del juego. La forma en que se diseñan y desarrollan los personajes impacta directamente en la experiencia del jugador y en la inmersión dentro de la historia. A continuación, se aborda una definición y un análisis más detallado del proceso de caracterización de personajes.

3.1 Definición de caracterización de personajes

La caracterización de personajes es un proceso fundamental en la creación y presentación de personajes dentro de una obra, ya sea teatral, cinematográfica o de cualquier otra forma narrativa. Este proceso es clave para que el público logre comprender y apreciar la profundidad de los personajes, ya que involucra la construcción de figuras complejas y realistas que establecen una conexión emocional con la audiencia. Para lograrlo, actores y directores colaboran en desarrollar personajes con gran detalle y autenticidad. (EIOE, 2024)

La caracterización se construye mediante la combinación de elementos físicos, psicológicos y emocionales. Los actores investigan y exploran las características internas de los personajes, como su personalidad, motivaciones, historias personales y las relaciones que mantienen con otros personajes. Esta investigación es esencial para que los actores puedan interpretar de manera creíble y profunda a sus personajes en escena.

Los aspectos físicos también juegan un papel crucial en el proceso de caracterización. El vestuario, el maquillaje y los peinados son herramientas importantes que ayudan a dar vida a los personajes y a transmitir su identidad de manera visual. Además, trabajar en detalles como gestos, posturas y movimientos corporales, que no solo reflejan la apariencia del personaje, sino también sus emociones y estado mental.

Otro aspecto relevante es la creación de la voz y el acento del personaje. La forma en que un personaje habla, incluyendo su tono, ritmo y lenguaje, es crucial para hacerlo más creíble y auténtico. (EIOE, 2024)

En el caso de los videojuegos, las animaciones, es decir los movimientos de los modelos también pueden brindar caracterización a los personajes. Desde la manera en que caminan, pueden generar visualmente una proyección de emociones de felicidad, ánimo etc. Los movimientos corporales también indican que sucede con el personaje. (EIOE, 2024)

3.2 ¿Cómo la animación ayuda a la caracterización de personajes?

La animación 3D ha revolucionado la forma en que se cuentan historias, especialmente en el ámbito de los videojuegos. Esta técnica permite que las experiencias narrativas sean más inmersivas, así como también la caracterización de este, haciendo que el jugador se sienta parte activa de la historia. En los videojuegos creados con animación 3D, el usuario no solo observa, sino que interactúa con la trama, convirtiéndose en el protagonista de una película interactiva. (Tai, 2022)

En efecto, la animación juega un papel crucial en la caracterización de personajes en los videojuegos. A través de la animación, los personajes cobran vida de una manera que complementa su diseño visual, personalidad y rol en la historia. Algunas características de cómo estas pueden ayudar son:

- **Expresión de emociones y personalidad:** Las animaciones faciales y corporales permiten que los personajes expresen una amplia gama de emociones, como felicidad, tristeza, miedo o ira. Esto ayuda a que los jugadores se conecten emocionalmente con los personajes.

Los movimientos corporales también refuerzan la personalidad de un personaje. Por ejemplo, un héroe confiado podría caminar con pasos firmes y postura erguida, mientras que un villano furtivo podría moverse de manera sinuosa o cautelosa. (Dumitru, 2021)

- **Estilo de movimiento único:** La forma en que un personaje se mueve, cómo corre, salta, ataca o interactúa con el entorno puede contar mucho sobre quién es. Un personaje torpe o cómico tendrá animaciones más exageradas y descoordinadas, mientras que un personaje ágil tendrá movimientos fluidos y precisos.

- **Refuerzo de habilidades y características:** Las animaciones de combate o de uso de habilidades ayudan a reflejar las capacidades del personaje. Por ejemplo, un mago que utiliza hechizos podría tener animaciones elegantes y místicas, mientras que un guerrero con fuerza bruta podría tener movimientos más contundentes y agresivos.

- **Contexto narrativo:** Las escenas animadas o cinemáticas permiten que los personajes interactúen entre sí, revelando aspectos clave de su historia, relaciones y motivaciones. Las animaciones en estas escenas ayudan a contar la historia de manera más vívida y a desarrollar los personajes.

- **Comportamiento no verbal:** A veces, lo que un personaje no dice es tan importante como lo que dice. Las pequeñas animaciones o gestos, como una mirada

desconfiada, un suspiro o una postura de defensa, pueden añadir capas a la personalidad del personaje y transmitir información que el diálogo no cubre.

- **Inmersión y realismo:** La animación contribuye a la sensación de realismo e inmersión, haciendo que los personajes se sientan vivos y reactivos al mundo que los rodea. Esto aumenta la conexión del jugador con los personajes y el mundo del juego. (Statham. N ,2023)

Por ello, la animación es un componente fundamental en la caracterización de personajes, ya que refuerza visualmente su personalidad, emociones, habilidades y su rol en la historia, haciendo que se sientan más auténticos y creíbles dentro del juego.

3.3 Ejemplos de personajes de videojuegos con buena caracterización

Las animaciones son fundamentales para la caracterización de personajes en los videojuegos, ya que permiten transmitir emociones, personalidades y características distintivas a través de acciones físicas. Los gestos, posturas y maneras de moverse pueden ofrecer a los jugadores una comprensión intuitiva de un personaje, complementando otros aspectos como el diseño visual y el diálogo. Un buen uso de la animación no solo hace que los personajes se sientan más realistas, sino que también añade capas de profundidad a su caracterización.

Por otro lado, existen las expresiones faciales y corporales. Las animaciones que capturan las emociones y expresiones de los personajes ayudan a los jugadores a conectarse emocionalmente con ellos. Esto puede incluir gestos, movimientos de la cabeza o cambios en la postura que reflejan la personalidad o el estado emocional del personaje. (Bisht, V, 2024)

Esto es particularmente evidente en personajes que, a través de su forma de moverse o interactuar con el entorno, revelan rasgos específicos como vulnerabilidad, fuerza o agilidad. A continuación, se presentan algunos ejemplos de personajes de videojuegos cuya caracterización ha sido reforzada por la calidad y la coherencia de sus animaciones:

1. Spider-Man - *Marvel's Spider-Man* (2018)

- **Animaciones de movimiento:** Las animaciones de balanceo en la ciudad son fundamentales para caracterizar a Spider-Man como ágil y fluido. El movimiento es rápido, pero también tiene una ligereza que refleja su habilidad como héroe. Además, sus animaciones de combate muestran un estilo acrobático y flexible, haciendo que el jugador sienta que está controlando a alguien extremadamente hábil, pero con un toque juvenil y relajado.
- **Caracterización:** Las animaciones no solo muestran sus poderes, sino que también destacan su confianza y experiencia como superhéroe. (Kyle, J, 2019).

2. Link - *The Legend of Zelda: Breath of the Wild*

- Animaciones de movimiento: En este juego, las animaciones de Link, como trepar montañas, esquivar ataques o correr, transmiten una sensación de fragilidad y humanidad. Su animación al escalar es lenta y cansada, mostrando esfuerzo, lo cual refuerza la idea de que Link, aunque valiente, es un héroe con limitaciones físicas.
- Caracterización: Las animaciones sutiles como la respiración pesada después de correr largas distancias o el temblor al colgarse de los acantilados hacen que el jugador sienta que Link está constantemente desafiando sus propios límites. (IGN, 2024)

3. Lara Croft - *Tomb Raider (2013)*

- Animaciones de movimiento: Las animaciones de movimiento de Lara Croft, especialmente las de sus interacciones con el entorno, son clave para mostrar su vulnerabilidad y evolución. Las animaciones de caída, saltos difíciles y sus movimientos cuando está herida transmiten a la perfección su lucha por sobrevivir.
- Caracterización: Sus animaciones de movimiento están diseñadas para mostrar su transformación de una arqueóloga inexperta a una sobreviviente decidida. A través de su movimiento, el jugador puede sentir su crecimiento tanto física como emocionalmente.

4. Kratos - *God of War (2018)*

- Animaciones de movimiento: Las animaciones de Kratos al luchar son pesadas y llenas de fuerza, lo que refuerza su carácter violento y poderoso. Sus ataques tienen un peso significativo, y su forma de moverse, lenta pero precisa, muestra su experiencia y fuerza física. (IGN, 2018)
- Caracterización: Sus movimientos reflejan su carácter: calculado, controlado, y lleno de ira contenida. Las animaciones transmiten mucho sobre su estado emocional, especialmente cuando interactúa con su hijo, Atreus. (Cooper, 2019)

5. Geralt de Rivia - *The Witcher 3: Wild Hunt*

- Animaciones de movimiento: Las animaciones de combate de Geralt, como el uso de espadas y magia, son fluidas y detalladas, mostrando su destreza como cazador de monstruos. Además, sus gestos sutiles en momentos tranquilos, como encender una hoguera o preparar pociones, le dan vida al personaje más allá de las batallas.

- Caracterización: Geralt se mueve de manera eficiente y calculada, lo que coincide con su naturaleza de cazador veterano y experto en su oficio. Las animaciones sutiles también muestran su actitud pragmática y a menudo fría, pero también su lado reflexivo. (IGN, 2018)

Estos personajes están magistralmente caracterizados a través de sus movimientos, que no solo aportan realismo, sino que también transmiten sus emociones, experiencia y personalidad. (Anderson, 2021)

4. Animación y jugabilidad

Las animaciones en los videojuegos juegan un papel crucial en la jugabilidad, ya que no solo mejoran la experiencia visual del jugador, sino que también influyen en la mecánica del juego, la inmersión y la percepción del personaje.

4.1 ¿Cómo las animaciones afectan la jugabilidad?

Las animaciones en los videojuegos son un componente esencial que influye directamente en la jugabilidad de diversas maneras. A continuación, se presentan los aspectos más relevantes sobre cómo las animaciones afectan la experiencia del jugador:

Inmersión y Realismo

La inmersión y el realismo en los videojuegos se pueden potenciar significativamente a través de animaciones bien diseñadas. Algunas maneras en que las animaciones contribuyen a estos aspectos, es en el movimiento natural. Las animaciones fluidas y naturales hacen que los personajes y objetos se sientan más reales. Esto incluye no solo los movimientos de caminar o correr, sino también cómo interactúan con el entorno, como saltar, agacharse o reaccionar a impactos.

La interacción dinámica en los videojuegos es un factor clave para aumentar la inmersión del jugador. Las animaciones que responden a sus acciones o a eventos dentro del juego generan una experiencia más auténtica. Un ejemplo común es cuando el jugador dispara un arma; una animación que represente de manera convincente el retroceso del arma y el movimiento del personaje contribuye a una sensación de realismo, mejorando la experiencia de juego. (Bisht, V, 2024)

Por otro lado, un entorno vivo también juega un papel importante en la inmersión. Animar elementos del entorno, como árboles que se balancean con el viento o luces que parpadean, ayuda a crear una atmósfera más dinámica y vibrante. Este tipo de detalles hacen

que el jugador perciba el mundo del juego como un espacio real y activo, generando una mayor conexión con el entorno virtual.

Las cinemáticas y la narrativa son otra herramienta eficaz para contar historias dentro de los videojuegos. Las animaciones en escenas cinemáticas no solo ayudan a avanzar la trama, sino que también permiten que el jugador se sienta parte de la historia. De este modo, el uso de animaciones cuidadosamente elaboradas en secuencias narrativas puede profundizar la inmersión y el compromiso emocional del jugador con el juego. (Bisht, V, 2024)

El *feedback* visual es fundamental para la comprensión del estado del juego. Animaciones que brindan retroalimentación, como un efecto visual al recoger un objeto o al completar una tarea, permiten al jugador recibir información clara sobre su progreso. Este tipo de animaciones, aunque sutiles, aumentan el sentido de logro y contribuyen a una mayor interacción y satisfacción del jugador. (Bisht, V, 2024)

Feedback Visual

El *feedback* visual en los videojuegos es fundamental, ya que permite a los jugadores comprender las consecuencias de sus acciones y fortalecer su conexión con el entorno del juego. Existen diversas formas en que las animaciones contribuyen a este *feedback*, proporcionando una respuesta visual clara e inmediata.

Una de estas formas es a través de las reacciones de los personajes. Cuando un jugador realiza una acción, como golpear a un enemigo o recoger un objeto, una animación de reacción, como un enemigo tambaleándose o el personaje celebrando, ofrece una respuesta inmediata. Este tipo de animaciones refuerzan la idea de que la acción del jugador ha tenido un impacto real en el mundo del juego, mejorando la sensación de interacción. (Wirtz, B. 2023)

Los efectos visuales también juegan un papel importante en el *feedback* visual. Animaciones que acompañan acciones específicas, como explosiones, destellos de luz o partículas en movimiento, añaden un nivel adicional de información visual. Por ejemplo, al disparar un arma, la combinación de una animación de retroceso con efectos tomar el arma, respiración, entre otros, hace que la acción se sienta más satisfactoria y realista, aumentando la inmersión. (Wirtz, B. 2023)

Las animaciones también son esenciales en la interacción con objetos. Acciones como abrir una puerta, presionar un botón o recoger un ítem se ven reforzadas por animaciones que confirman el éxito de la interacción. Estos movimientos suaves, acompañados de efectos visuales y sonoros, ofrecen al jugador un *feedback* inmediato y satisfactorio sobre su acción.

El estado de salud y condiciones del personaje es otro elemento que puede comunicarse mediante animaciones. Por ejemplo, el parpadeo de la pantalla al recibir daño o una

animación de curación proporcionan información clara al jugador sobre su estado actual, permitiéndole tomar decisiones rápidas y estratégicas. (Wirtz, B. 2023)

El cierre de acciones también es relevante en el *feedback* visual. Animaciones que indican la finalización de una tarea, como un gesto de victoria o un efecto de desenfoque al completar un nivel, ayudan a los jugadores a entender que han alcanzado con éxito sus objetivos, reforzando el sentido de logro.

Por último, la consistencia y claridad en el uso de animaciones es clave para crear un lenguaje visual coherente dentro del juego. Si las animaciones son predecibles y los jugadores pueden anticipar cómo reaccionan los personajes u objetos, se sentirán más en control y seguros en su experiencia de juego. (Wirtz, B. 2023)

Por ello, la implementación de un *feedback* visual efectivo a través de animaciones no solo mejora la jugabilidad, sino que también aumenta la satisfacción del jugador al proporcionar una experiencia más rica y envolvente.

Construcción de Personajes

La construcción de personajes en los videojuegos se ve profundamente enriquecida por el uso de animaciones. Estas no solo dan vida a los personajes, sino que también ayudan a contar su historia, definir su personalidad y mejorar la inmersión del jugador en el mundo del juego. A continuación, se destacan algunos aspectos clave sobre cómo las animaciones contribuyen a este proceso.

Otro aspecto relevante es cómo las animaciones reflejan la personalidad y el estilo de un personaje. Un guerrero fuerte y robusto puede tener movimientos pesados y contundentes, mientras que un personaje más ágil y astuto podría presentar animaciones rápidas y fluidas. Este contraste ayuda a los jugadores a identificar rápidamente la naturaleza de los personajes, facilitando su comprensión del papel que juegan en la historia.

En cuanto a las interacciones sociales, las animaciones también juegan un papel importante. Las reacciones de los personajes durante conversaciones o al interactuar con otros refuerzan las relaciones y la dinámica entre ellos. Esto aporta credibilidad a las escenas y mejora la inmersión, haciendo que las interacciones entre personajes sean más naturales y envolventes. El desarrollo de arcos narrativos es otro elemento que se ve potenciado por las animaciones. A medida que los personajes evolucionan a lo largo de la historia, sus animaciones pueden reflejar estos cambios. Por ejemplo, un personaje que comienza siendo tímido y luego se vuelve valiente puede mostrar una evolución en su postura y en la forma en que se mueve, evidenciando su transformación a través de la narrativa. (Bisht, V, 2024)

La identidad visual de un personaje se ve fortalecida por sus animaciones, ya que movimientos únicos y patrones de comportamiento característicos permiten que un personaje sea fácilmente reconocible y memorable, incluso en juegos con numerosos personajes. Esta distinción contribuye a la creación de personajes icónicos que dejan una impresión duradera en los jugadores. Además, la conexión con el jugador se intensifica

cuando los personajes se mueven y reaccionan de manera convincente, un aspecto crucial en juegos de rol o narrativos donde los personajes son el núcleo de la experiencia. Animaciones efectivas permiten a los jugadores sentirse más involucrados con los personajes, enriqueciendo así la experiencia general del juego.

Finalmente, las acciones y habilidades de los personajes son otro punto clave. Animaciones que muestran habilidades especiales o ataques pueden hacer que las acciones se sientan más satisfactorias. Por ejemplo, una animación espectacular durante un ataque puede hacer que el jugador se sienta poderoso y efectivo, mejorando la experiencia de juego. (Bisht, V, 2024)

Por ello, la integración efectiva de las animaciones en la construcción de personajes puede llevar la experiencia del jugador a un nivel superior, haciendo que los personajes sean más reales, memorables y significativos.

Fluidez y Dinámica del Juego

La fluidez y la dinámica del juego son esenciales para mantener a los jugadores comprometidos y garantizar una experiencia satisfactoria. Las animaciones juegan un papel crucial en estos aspectos,

El control responsivo también es un aspecto importante. Animaciones que responden de manera inmediata a las entradas del jugador, como los saltos, ataques o movimientos, crean una sensación de fluidez y precisión en el control del personaje. (Wirtz, B ,2023)

La cadencia y el ritmo de las animaciones pueden influir directamente en la dinámica del juego. Por ejemplo, en un juego de acción, animaciones rápidas generan una sensación de urgencia e intensidad, mientras que, en juegos más pausados, animaciones lentas pueden crear una atmósfera más reflexiva y calmada. Este uso del ritmo en las animaciones ayuda a establecer el tono del juego y la forma en que los jugadores lo experimentan.

Los efectos de impacto son fundamentales para la dinámica del juego, ya que las animaciones que acompañan ataques y colisiones ofrecen un *feedback* visual claro sobre las acciones de los jugadores, lo que no solo ayuda a entender lo que ocurre, sino que también hace que las interacciones sean más satisfactorias. Además, las animaciones que reflejan la vida en el entorno, como *NPCs* caminando y elementos naturales reaccionando al viento, añaden dinamismo y autenticidad al mundo del juego, creando una experiencia más envolvente y rica. (Wirtz, B ,2023)

Otro aspecto importante son las acciones encadenadas. Animaciones que permiten a los jugadores encadenar acciones de manera fluida, como pasar de un ataque a un combo o de un salto a un giro, fomentan un estilo de juego más dinámico. Esto les permite a los jugadores experimentar con diferentes estrategias y tácticas, lo que enriquece la jugabilidad y aporta profundidad a la experiencia del juego.

Las interacciones contextuales también se ven mejoradas por las animaciones. Cuando un personaje reacciona de manera distinta según el contexto, como agacharse para esquivar

un disparo o interactuar de manera diferente con el entorno, mejora la sensación de realismo e inmersión. Esto hace que los jugadores se sientan más conectados con el juego, ya que las reacciones del personaje se alinean con las situaciones que enfrentan. (Wirtz, B ,2023)

Por último, el *feedback* visual inmediato que proporcionan las animaciones es crucial para mejorar la fluidez de la experiencia de juego. Animaciones que ofrecen retroalimentación visual clara y rápida, como un destello al recibir daño o un efecto de sonido al completar una acción, permiten a los jugadores entender las consecuencias de sus acciones de manera instantánea, manteniendo el ritmo del juego sin pausas innecesarias. Por eso, la implementación efectiva de animaciones mejora tanto la fluidez como la dinámica del juego, haciendo que la experiencia sea más entretenida y envolvente para los jugadores. (Wirtz, B ,2023)

Mecánicas de Juego

Las animaciones son fundamentales para implementar y mejorar las mecánicas de juego en los videojuegos, ya que aportan vida a las acciones de los personajes y hacen que las interacciones sean más fluidas y comprensibles. (Bisht, V, 2024c)

Las animaciones son fundamentales para representar acciones en los videojuegos, ofreciendo una representación visual clara de lo que los personajes pueden hacer, como saltar, atacar o ejecutar habilidades especiales. Esto facilita a los jugadores la comprensión de las mecánicas del juego y proporciona una referencia visual inmediata para interactuar con el mundo del juego. Además, las animaciones ofrecen *feedback* inmediato; por ejemplo, tras un ataque, la animación de impacto refuerza la sensación de éxito y efectividad de la acción del jugador, mejorando así la experiencia de juego al informar instantáneamente sobre la efectividad de sus acciones.

Algunas mecánicas de juego dependen de condiciones de activación vinculadas a animaciones. Por ejemplo, una habilidad especial que solo puede ejecutarse después de una animación de carga introduce un elemento estratégico, ya que los jugadores deben prestar atención al momento y lugar adecuado para ejecutar ciertas habilidades. (Bisht, V, 2024c)

Las animaciones también mejoran las interacciones contextuales. En situaciones donde un personaje entra en un área específica o interactúa con un objeto, las animaciones activadas en función del contexto permiten a los jugadores experimentar diferentes mecánicas de manera más inmersiva, como al abrir puertas o recoger objetos.

El encadenamiento de acciones es otra área en la que las animaciones son esenciales. Permiten que los jugadores combinen varias acciones sin interrupciones, como en un juego de combate donde un jugador puede enlazar un ataque, esquivar y luego realizar una habilidad especial. Esto crea combos fluidos y dinámicos, lo que hace que la jugabilidad sea más emocionante y rápida. La consistencia en el juego es clave para una experiencia clara y predecible. Si las animaciones para ciertas mecánicas son coherentes a lo largo del juego, los jugadores pueden anticipar cómo funcionarán sus acciones en diferentes contextos. Esto les permite dominar más fácilmente las mecánicas y optimizar sus estrategias.

Las animaciones permiten la creación de mecánicas de juego innovadoras. Por ejemplo, un juego que manipule el tiempo a través de animaciones puede permitir que los jugadores ralenticen o aceleren el tiempo, afectando de manera creativa la jugabilidad y brindando nuevas formas de interactuar con el mundo del juego. Las reacciones de personajes a las acciones del jugador, como cuando sufren daño o interactúan con el entorno, también son facilitadas por animaciones. Estas reacciones mejoran la inmersión y ayudan a los jugadores a entender mejor las consecuencias de sus decisiones dentro del juego.

Por último, la variabilidad en el juego es posible gracias a animaciones alternativas. Tener diferentes animaciones para la misma acción en distintos contextos ofrece más opciones y estrategias a los jugadores, lo que enriquece la jugabilidad y añade variedad a la forma en que se abordan los desafíos. Por ello, las animaciones son un componente esencial en el diseño de mecánicas de juego, mejorando la jugabilidad, la inmersión y la satisfacción de los jugadores. (Bisht, V, 2024c)

Identidad y Estilo Visual

La identidad y el estilo visual de un videojuego son elementos esenciales que definen la experiencia del jugador y contribuyen a que un juego sea memorable y atractivo. Las animaciones juegan un papel clave en la creación y el mantenimiento de esta identidad visual, aportando coherencia y profundidad al estilo del juego. A continuación, se destacan varias formas en que las animaciones contribuyen a este aspecto.

En primer lugar, las animaciones aseguran la consistencia estilística a lo largo del juego. Si los personajes, objetos y entornos comparten una misma estética animada, el juego resulta visualmente más armónico y atractivo. Esto es crucial para mantener una experiencia inmersiva, ya que una inconsistencia en las animaciones podría romper la cohesión visual. (Artadmin ,2023b)

Además, que el diseño de personajes también se ve altamente influenciados movimientos. Un personaje cómico, por ejemplo, podría tener animaciones más exageradas y dinámicas, mientras que un personaje más serio mostraría movimientos más controlados y sutiles, ayudando a comunicar su esencia sin necesidad de palabras.

Las animaciones refuerzan el estilo de arte general del juego. Ya sea que se trate de gráficos 2D, 3D, *pixel art* o incluso *stop-motion*, la manera en que los personajes y objetos son animados debe complementar y realzar el estilo artístico elegido. Esto crea una experiencia visual más coherente y envolvente, que facilita la inmersión del jugador.

La cultura y temática del juego también se ven enriquecidas por las animaciones. Un juego ambientado en una cultura específica puede incorporar movimientos, gestos y posturas que reflejen los valores o tradiciones de esa cultura, añadiendo autenticidad y profundidad al juego.

A nivel narrativo, las animaciones contribuyen a la narrativa visual. Los movimientos y las interacciones de los personajes con su entorno cuentan una historia en sí mismos,

comunicando emociones, intenciones y conflictos sin necesidad de diálogos. Esto permite que los jugadores interpreten la historia de manera más intuitiva y emocional.

La animación puede incluso contribuir a la identidad de marca. Un estilo de animación distintivo hace que un juego sea fácilmente reconocible y recordado por los jugadores, lo que puede aumentar su popularidad y establecer una conexión más fuerte entre el juego y su audiencia. (Aminian. P, 2024)

Por último, las animaciones que transmiten emociones y sentimientos profundos permiten a los jugadores conectar con los personajes y la historia en un nivel más personal. Esto crea una experiencia más significativa y duradera, generando una conexión emocional que deja una impresión positiva en los jugadores. (Artadmin, 2023)

Por ende, las animaciones son fundamentales para la jugabilidad en los videojuegos, afectando la inmersión, el feedback, la narrativa, la fluidez y las mecánicas del juego. La calidad de las animaciones puede determinar en gran medida cómo los jugadores experimentan y disfrutan de un juego. A medida que la tecnología de animación continúa evolucionando, su importancia en el diseño de juegos sólo aumentará.

4.2 Variables que pueden influir o afectar la jugabilidad.

Las animaciones y transiciones en videojuegos tienen un impacto significativo en la jugabilidad, afectando tanto el tiempo de respuesta del jugador como la fluidez de las interacciones dentro del juego. Una de las variables clave es la duración de las animaciones. La velocidad de estas es crucial; si son demasiado largas o lentas, pueden ralentizar la jugabilidad y frustrar a los jugadores. Esto se hace evidente en juegos de acción, donde las animaciones de ataque o recuperación que tardan demasiado pueden interrumpir el ritmo del juego. Además, la sincronización de las acciones es esencial. Animaciones bien cronometradas permiten una mejor percepción de control, mientras que tiempos prolongados o imprecisos pueden generar una sensación de control reducido. Por ejemplo, en juegos de ritmo rápido, las animaciones de salto o esquiva deben ejecutarse rápidamente para mantener la dinámica del juego. (Murphy. C, 2013).

Otro aspecto importante son las transiciones entre animaciones. La fluidez en la transición entre diferentes estados de animación, como pasar de caminar a correr o de correr a saltar, es fundamental. Si las transiciones son abruptas o rígidas, pueden crear una sensación antinatural que afecta la inmersión del jugador. Implementar transiciones suaves ayuda a mantener la coherencia de los movimientos y la experiencia general. Además, en algunos juegos, es vital que estas transiciones se realicen mediante interpolaciones suaves, lo que afecta la percepción de control y realismo. Si son torpes o inexistentes, el jugador podría sentir que el personaje es difícil de manejar.

La inclusión de *frames* de invulnerabilidad también es relevante. En juegos de combate o plataformas, estas animaciones permiten que el jugador no reciba daño durante momentos específicos, como al esquivar o saltar. La gestión de estos tiempos y su coherencia con las

animaciones impactan directamente en la jugabilidad estratégica, ya que los jugadores deben ser conscientes de cuándo pueden actuar sin riesgo. (Kajala. N, 2022).

Finalmente, la carga cognitiva y los tiempos de respuesta son factores a tener en cuenta. Las animaciones demasiado rápidas o complejas pueden sobrecargar la percepción visual del jugador, mientras que las que son demasiado lentas pueden provocar aburrimiento o desconexión. Encontrar un equilibrio adecuado es esencial para mantener la atención del jugador sin causar fatiga.

Como es notorio, tanto la duración de las animaciones como las transiciones tienen un efecto profundo en el flujo, la estrategia y la percepción del jugador dentro de los videojuegos. Por lo tanto, es fundamental que estas se equilibren cuidadosamente para no afectar negativamente la jugabilidad y garantizar una experiencia más fluida y satisfactoria.

V. Metodología

La creación de animaciones 3D caricaturescas para representar la historia de Gaus y la "Revolución Ornitoreense" en un videojuego de ciencia ficción fue un proceso complejo que demanda un enfoque metodológico estructurado y profesional. Este documento detalla las etapas críticas del desarrollo, desde la conceptualización hasta la entrega del producto final, asegurando que cada fase contribuya a un resultado cohesivo y de alta calidad.

1. Recopilación de Recursos

La primera etapa fue crucial para poder establecer la base creativa y técnica del proyecto. Para ello, se llevó a cabo una recopilación exhaustiva de recursos visuales, conceptuales y narrativos que informarán la dirección artística y el desarrollo de las animaciones.

1.1. Arte Conceptual

Para esta parte se realizó una búsqueda de imágenes de referencia que inspiraron el diseño visual de los personajes, escenarios y objetos dentro del mundo de Gaus. Esto se llevó a cabo con la finalidad de poder crear una caracterización más pertinente para cada uno de los modelos que se animaron.

Como se estudió con anterioridad el tener acceso al contexto del personaje, brindó una mejor información para poder caracterizar el mismo e ingeniar las animaciones que se solicitaron en cada modelo.

1.1.1 Guiones y narrativas

Durante este punto se realizó una revisión de los guiones del juego. Esto, porque se pretendió indicar los momentos claves que debían ser reforzados por las animaciones. La sinopsis del videojuego es el siguiente: En un mundo donde los ornitorrincos vivían en paz y habían dominado el arte de la cocina, sus depredadores deciden invadirlos y obligarlos a cocinar para ellos. Uno de estos ornitorrincos, Gaus, ama tanto a la cocina como a sus ingeniosas invenciones. Un día, una de estas invenciones no sale como lo planea, y termina envuelto en un plan para llevar a cabo la revolución Ornitoreense. La narrativa fue de importancia ya que para ello se desarrolló una lista preliminar de las animaciones requeridas, ver Tabla 1.

	Animaciones	IDLE	KO	Caminar	Correr	Saltar	Disparar	Extra
Personajes	Gaus	X	X	X	X	X	X	X
	Dingo	X	X	X	X		X	
	Dingo Boss	X	X	X	X		X	
	Buho							X

Tabla 1. Lista preliminar de las animaciones requeridas por personaje

Nota: La columna de EXTRA se refiere a una animación fuera de lo normal. Ejemplo: Buho – tiene una animación de volar. Mas adelante del documento se describe con exactitud la animación de estas.

1.1.2 Referencias históricas y estilísticas

Para este apartado se hizo una investigación analítica de estilos de animaciones caricaturescas que se fueran aptos con la visión del proyecto. Además, se consideró cómo los estilos influyen en la percepción del jugador y la narrativa general.

Como parte de dicha investigación se definió la animación como caricaturesca y tridimensional. Tratándose de que estas mismas cumplieran con algunos de los principios básicos de la animación. Estos fueron los siguientes:

- Estirar y encoger
- Anticipación
- Puesta en escena
- Animación directa y pose a pose
- Acción continuada y superpuesta
- Acelerar y desacelerar
- Arcos
- Acción secundaria
- Exageración
- Atractivo

Estos procesos no solo facilitaron la obtención de una visión clara y compartida entre los miembros del equipo, sino que también se establecieron las bases para decisiones informadas en las siguientes etapas del desarrollo.

1.2 Investigación y Experimentación con Herramientas de Animación

Con los recursos recopilados, se procedió a la investigación de las herramientas para llevar a cabo el proyecto.

Las herramientas que se investigaron al inicializar el proceso fueron las siguientes: Blender, Mixamo y Unity. Como primera instancia se investigaron los fundamentos de la animación con Blender. Aquí surgieron conceptos claves como: *Keyframes* y *Rigging*.

Por otra parte, las herramientas de mixamo fueron de ayuda, ya que brindó una solución de animación en 3D. Permitiendo ajustar animaciones de personaje de cuerpo completo capturadas por actores de movimientos profesionales.

1.2.1 Mixamo: Flujo de trabajo

Mixamo, mantiene un flujo mucho más simple y fácil de usar, que cualquiera puede hacer uso de ella sin mucha capacitación o conocimiento. Además, cuenta con una interfaz de usuario más optimizada e intuitiva, ver Figura 16.

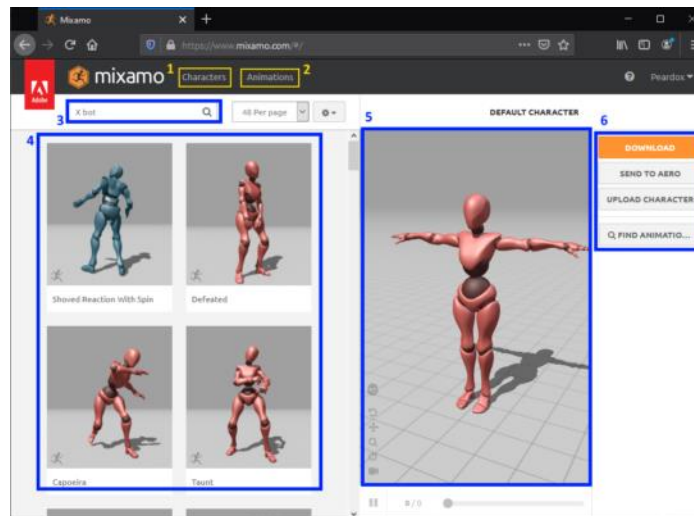


Figura 16. Interfaz de Mixamo

Como consiguiente se experimentó con Unity, uno de los motores de creación de juego más usados en la actualidad (MasterD, 2019b). Esta es una herramienta que ayudó con la integración de las animaciones al videojuego como tal. En este se investigaron sobre las herramientas que usa de animación, las cuales son: Componente *Animator* y máquina de estados.

Aparte de esta investigación se procedió a la experimentación con las herramientas de animación, la cual trajo como consiguiente el uso de una nueva herramienta que no estuvo definido en la inicialización de esta. Y este es una herramienta de Blender conocida como *Add-ons* llamada AutoRig Pro el cual ayudó para el procedimiento del *Rigging* que se presenta más adelante.

1.3 Desarrollo de Animaciones

En esta sección, se detalla el proceso de desarrollo de las animaciones del personaje principal. Cabe mencionar que los modelos utilizados para el desarrollo de las animaciones fueron brindados por otra compañera como parte del megaproyecto. Posteriormente, se aplicaron los mismos procedimientos a los demás modelos, teniendo en cuenta sus características individuales para lograr animaciones proporcionales y coherentes en cada uno de ellos.

1.3.1 Creación del Rigging

Para este primer proceso se optó por asegurar que el modelo utilizado viniese en posición de “T”. Esta posición se utiliza comúnmente en las animaciones y modelado por ser una postura neutral antes de aplicar cualquier tipo de animación. Además, que este facilita el establecimiento de un punto de partida, a verificar la simetría y poder hacer de manera generar una evaluación visual de forma y proporciones para garantizar un *Rigging* exitoso, ver Figura 17.

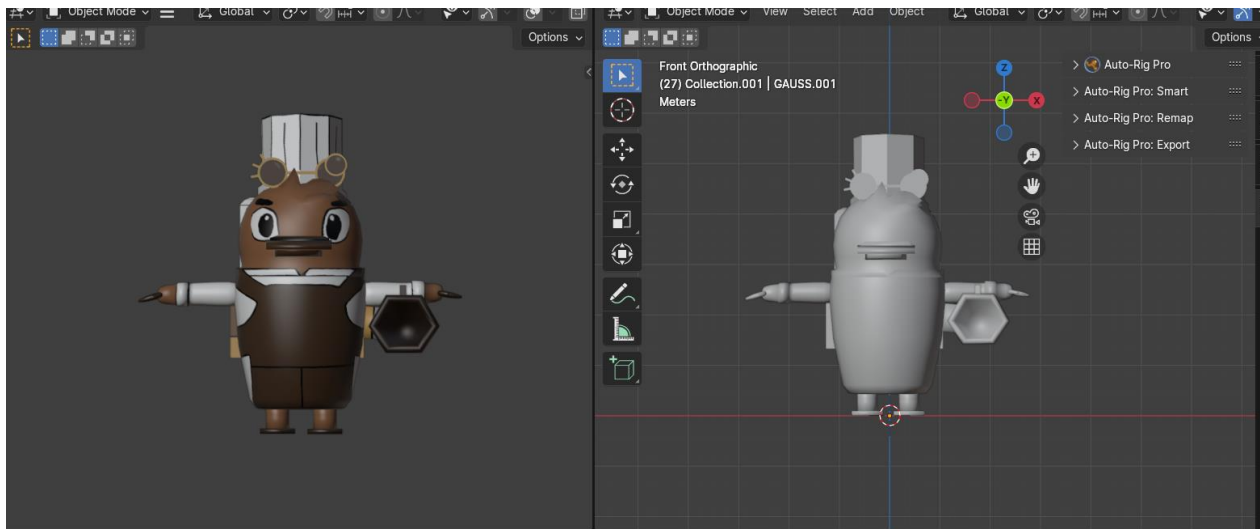


Figura 17. Modelo Gauss en posición “T”.

1.3.1.1 Colocación de marcadores

Después de hacer una breve evaluación visual y colocación del posicionamiento correcto se procedió a la creación del Rigging con el uso de la herramienta Auto-Rig Pro de Blender. Para ello se realizó la colocación de marcadores en el cuerpo completo del modelo, ver Figura 18.

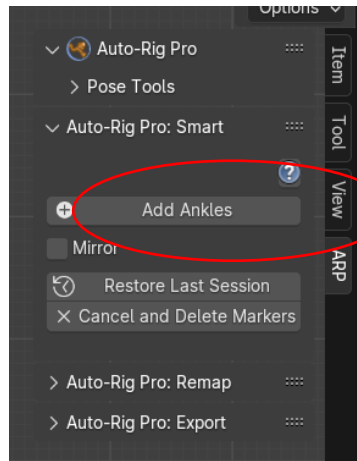


Figura 18. Opción de agregación de marcadores para el modelo

Se procedió a colocar todos los marcadores que por defecto ofrece la herramienta de Auto-Rig Pro hasta terminar, ver Figura 19.

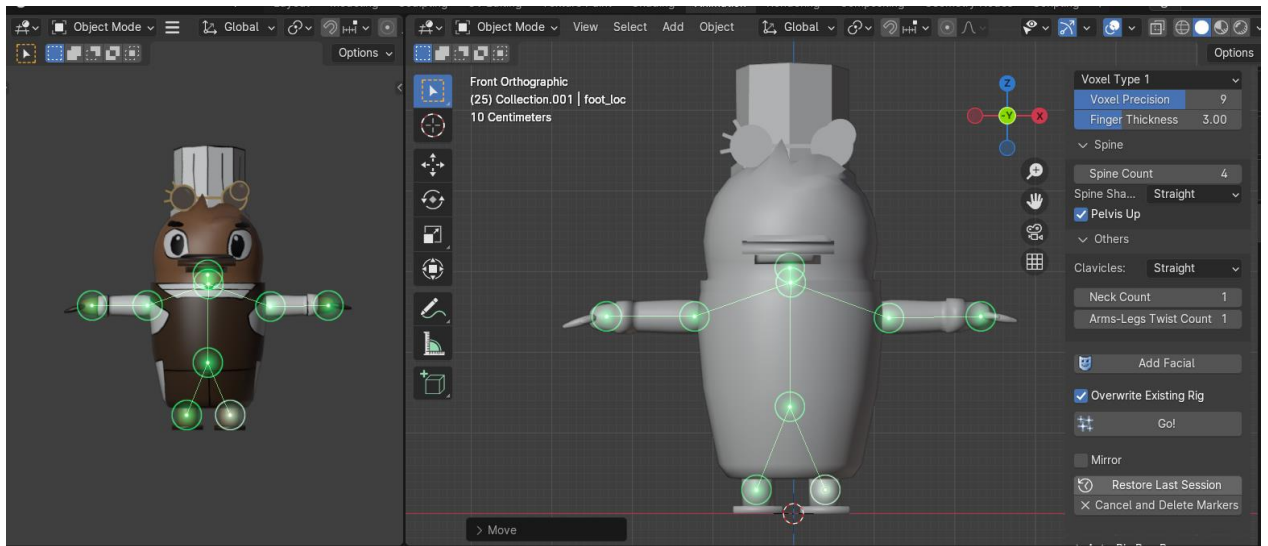


Figura 19. Colocación de marcadores corporales en el modelo

1.3.1.2 Detección automática

La característica de detección automática se realizó por parte del Auto-Rig Pro generando los huesos en el modelo, facilitando la configuración inicial del *rig* (esqueleto) al detectar la estructura básica del personaje y posicionar los huesos principales de la manera más precisa posible.

Durante esta etapa sucedieron varios procedimientos generales. El primero, consiste en que el plugin analiza la forma general del modelo para identificar las partes principales del cuerpo. Continuamente el sistema realizó una detección de las proporciones y alineamientos del modelo para ajustar la posición y así redujo la necesidad del ajuste manual, ver Figura 20.

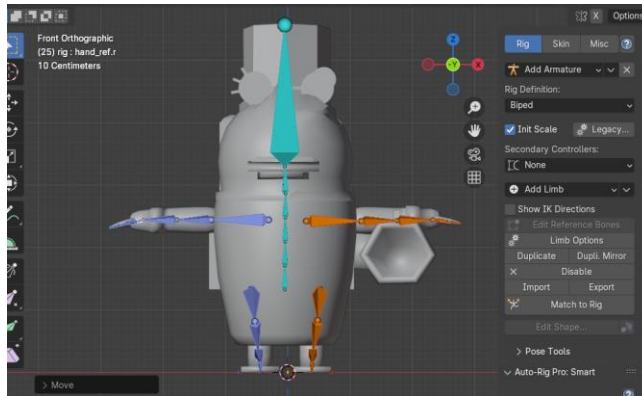


Figura 20. Detección automática y asignación de huesos

1.3.1.3 Vinculación *armature* vs *mesh*

El proceso de *Binding* es donde el esqueleto (*rig*) se conecta con el modelo 3D o bien el *mesh* (malla). Esto permitió que los huesos controlen la deformación de la malla. En otros términos, este procedimiento es el paso que asoció los vértices del modelo con los huesos del *rig*, para que, al realizar un movimiento a un hueso la malla también se deforme.

Durante este proceso suceden acontecimientos importantes. El primero es la asignación de piel. El proceso de vinculación implica aplicar una piel virtual al modelo 3D. Esta piel está compuesta por vértices de la malla, como se mencionó con anterioridad, que se asignan a diferentes huesos del esqueleto. Como segundo punto se generó el pesado de vértices. Aquí cada vértice del modelo se asocia con uno o más huesos mediante “pesos” que determina cuanta influencia tiene cada hueso sobre ese vértice. Por ejemplo, un vértice en el codo puede estar influenciado en un 70% por el hueso del brazo superior y en un 30% por el hueso del antebrazo. Esto permite una transición suave y natural entre las diferentes partes del cuerpo, ver Figura 21.

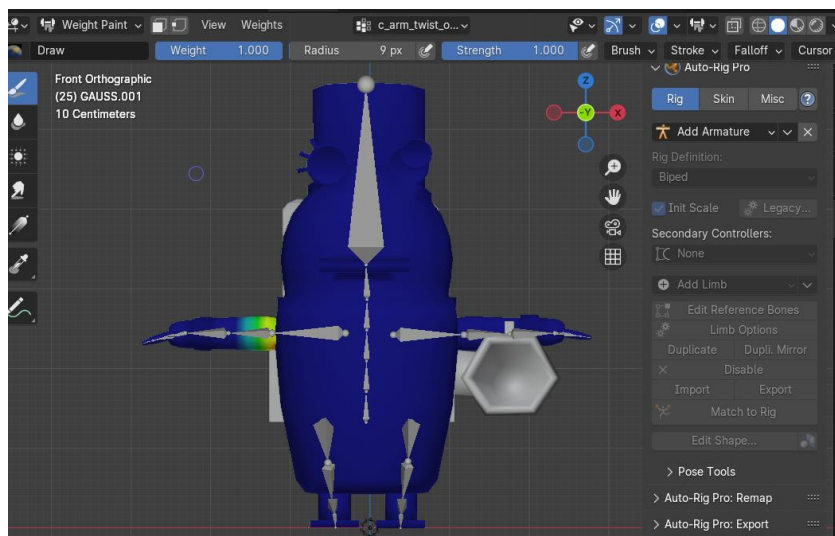


Figura 21. Pesado de vértices

Como se muestra en la Figura 21, el color rojo indica el mayor nivel de influencia, con un valor de 1.0. Esto significa que los vértices pintados de rojo se moverán completamente con el hueso correspondiente. Conforme el color cambia a naranja o amarillo, la influencia del hueso disminuye, pero sigue siendo significativa, afectando de manera parcial los vértices. Los vértices pintados de verde tienen un peso moderado, lo que indica que se moverán menos que los amarillos, pero aún responderán al movimiento del hueso.

A medida que el color se aproxima al azul claro, el peso es bajo, lo que significa que los vértices apenas se ven afectados por el hueso. Finalmente, el azul oscuro indica que el peso es cero, y los vértices que están pintados de este color no serán influenciados por el hueso en absoluto, permaneciendo fijos durante los movimientos del *rig*.

Este sistema de colores es clave para asegurar que las deformaciones en la malla sean controladas de manera precisa y que los movimientos resultantes sean fluidos y coherentes con la animación. Este sistema de pesos lo ejecutó de manera optimizada el procedimiento de vinculación.

1.3.1.4 Ajustes finales/extras

Después del *binding*, el *rig* está completamente funcional y listo para ser animado. Los movimientos que se apliquen a los huesos del esqueleto afectarán directamente la malla, deformándola de manera que se vea natural en la mayoría de los casos. Sin embargo, se ajustó en todos los modelos el esqueleto que este proyecto. En ocasiones la herramienta no nos arroja resultados correctos, por lo cual se realizó una corrección manual. A continuación, en la Figura 22. Se muestra del lado izquierdo la generación del esqueleto incorrecto y del lado derecho su debida corrección, ver Figura 22.

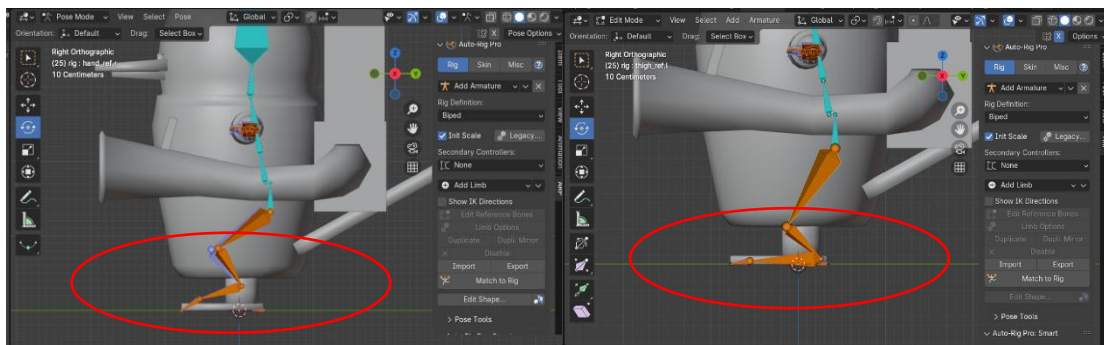


Figura 22. Corrección del esqueleto tras vinculación automática.

Adicionalmente se agregó un hueso extra como parte del esqueleto base generado. Esto se debe a que Auto-Rig Pro genera el *rig* como si el modelo fuera un humanoide. Sin embargo, el modelo con el que se cuenta es antropomorfo, es decir es aquel que combinan características humanas y animales. Aunque el ornitorrinco es un animal y podría ser considerado como un modelo cuadrúpedo, el hecho de que esté en dos patas y use sus patas

delanteras para disparar como un humano le otorga características humanoides, ver Figura 23.

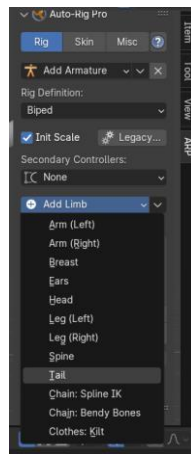


Figura 23. Opción de agregación de cola

Además de la colocación de la cola extra al *rig*, también se hizo una asignación de jerarquía de huesos. A este procedimiento se le conoce también como *Parent* o “aparentar huesos”. Esto se refiere al proceso de establecer relaciones jerárquicas entre los huesos de un esqueleto (o *rig*) en un modelo 3D, de modo que los huesos secundarios (hijos) sigan el movimiento de los huesos primarios (padres). En este caso se aparentó en la cola al hueso de la cadera, ver Figura 24.

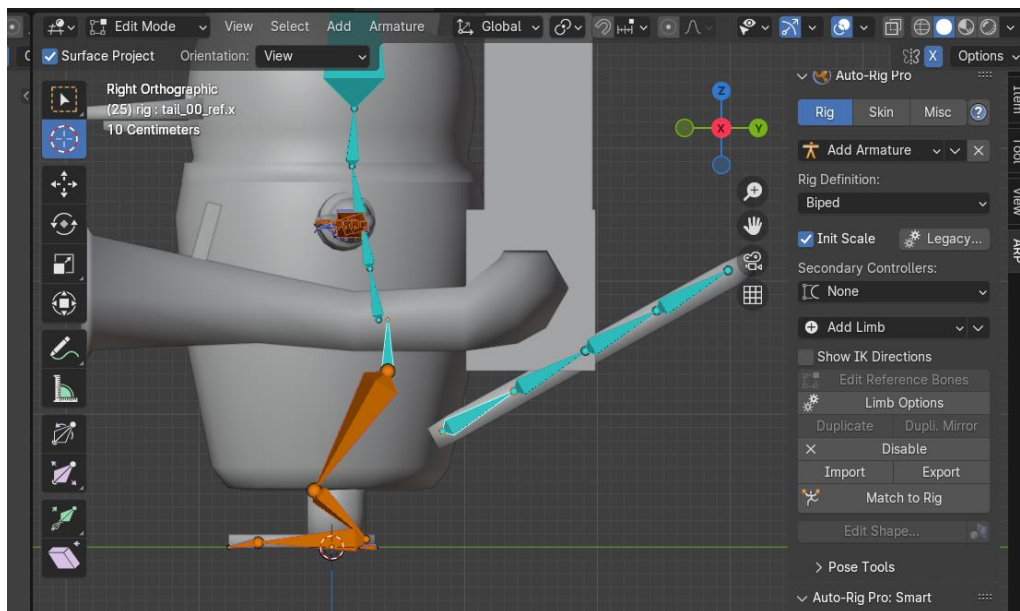


Figura 24. Agregación de cola y asignar jerarquía de huesos

Luego de la ejecución de correcciones se seleccionó la opción de *Match to rig* para unificar el objeto con el *rig* y se obtiene el modelo preparado para poder ser animado.

1.3.2 Búsqueda de animaciones base Mixamo

Se utilizó Mixamo como una herramienta clave para la búsqueda y descarga de animaciones base. Mixamo ofreció para el proyecto animaciones predefinidas que sirvieron como punto de partida para la creación de las animaciones específicas requeridas en el videojuego, ver Figura 25.

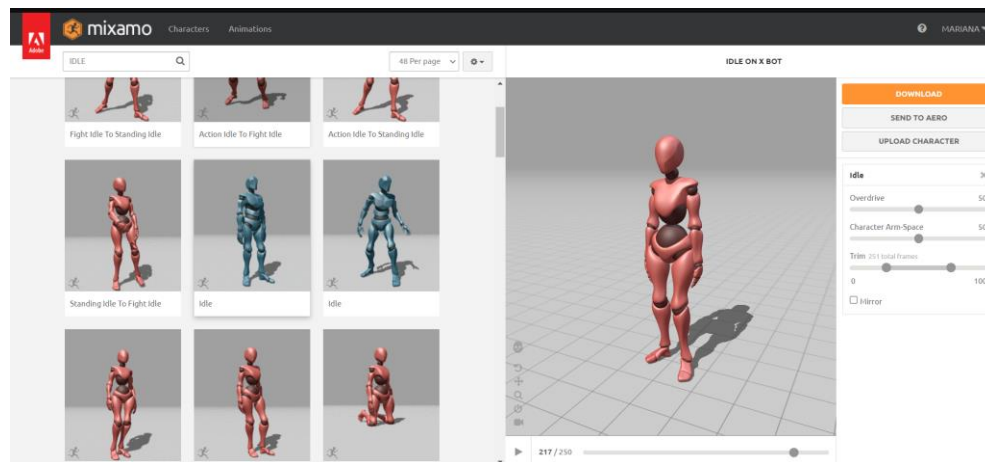


Figura 25. Búsqueda de animaciones base en Mixamo

Estas animaciones se adaptaron a los modelos del juego, conservando ciertos aspectos fundamentales, como el posicionamiento y movimiento corporal, mientras que otros componentes se ajustaron o personalizaron manualmente para cumplir con las exigencias particulares de la jugabilidad, como el uso de armas y otras interacciones especializadas. Esto permitió agilizar el proceso de animación sin comprometer la precisión ni la coherencia con las mecánicas del juego.

1.3.2.1 Descarga y exportación animación mixamo

En la etapa de descarga y exportación de animaciones desde Mixamo a Blender, se seleccionaron animaciones base específicas de la biblioteca de Mixamo que se alinearan con las necesidades del videojuego (IDLE, KO, JUMP, RUN, WALKING, SHOTING, THROWING). Una vez elegidas, las animaciones fueron descargadas en formato FBX, garantizando la compatibilidad con Blender.

Este flujo de trabajo facilitó la integración eficiente de animaciones de alta calidad en el entorno de Blender, permitiendo una edición precisa para personalizar las acciones y movimientos requeridos en el proyecto.

1.3.2.2 Unificación de armaduras

En la etapa de unificación de armaduras, se utilizó el complemento Auto-Rig Pro en Blender para combinar el *rig* de Mixamo, previamente descargado, con el *rig* personalizado del modelo del videojuego. Para llevar a cabo esta integración, primero se accedió a la pestaña *Skin* del complemento Auto-Rig Pro. Allí, se seleccionaron ambas armaduras, tanto

la de Mixamo como la del modelo propio. Luego, se generó la lista de huesos mediante la opción *Build Bones List*. Ver figura 26.

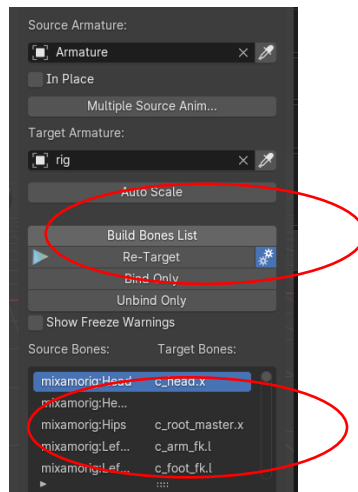


Figura 26. Construcción de huesos al unificar armadura

Después de generar la lista de huesos, se seleccionó el hueso base principal (*root bone*) del *rig*, designando el *root_master* como *Set as Root* para asegurarse de que ambas estructuras estuvieran correctamente alineadas. Una vez realizados estos ajustes, se ejecutó la opción *Re-target*, lo que permitió adaptar la animación de Mixamo al *rig* personalizado.

Finalmente, para vincular la animación al modelo, se seleccionaron tanto la armadura como el modelo y se utilizó la opción *Bind*, logrando que la animación importada se ajustara de manera óptima al modelo final, manteniendo coherencia y funcionalidad dentro del flujo de trabajo de animación del proyecto.

1.3.3 Edición de animación

En esta área se abordó el proceso de edición de animaciones dentro de Blender, donde se realizaron ajustes y personalizaciones a las animaciones previamente importadas y unificadas como se mostrará más adelante. Este proceso es fundamental para la adaptación de las animaciones a las especificaciones del videojuego, garantizando que los movimientos de los modelos se alineen con las mecánicas y dinámicas del juego.

1.3.3.1 Eliminación/Agregación *KeyFrames*

Se llevaron a cabo modificaciones en los *keyframes*, permitiendo el ajuste preciso de las trayectorias y posiciones de los personajes durante las interacciones, como el uso de armas y otras acciones específicas.

Algunos de estos ajustes adicionales involucraron el eliminar por completo los *keyframes* de animación que traía el esqueleto de Mixamo por partes del personaje y realizar las animaciones propias. Un ejemplo de este es para la animación del disparo de Gaus. A continuación, se dará a conocer una de ellas.

Como se observa en la Figura 27, se descargó una animación básica de disparo de mixamo.

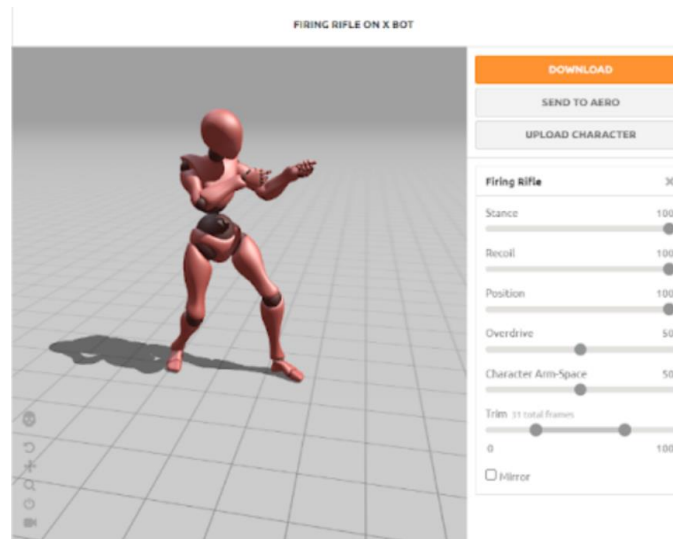


Figura 27. Animación de disparo descargada de MIXAMO.

Sin embargo, el modelo de Gaus maneja un arma diferente a la de un rifle como se muestra en la Figura 26. Su arma es una mochila que cuenta con un expulsor en forma de tubo en la parte inferior o baja del lado izquierdo por lo cual todos los *keyframes* con respecto a ambos brazos eran inservibles. Para este caso específico se eliminaron toda la animación que correspondía a los brazos para que de manera propia se pudieran animar con el arma específica de Gaus, ver Figura 28 y 29.

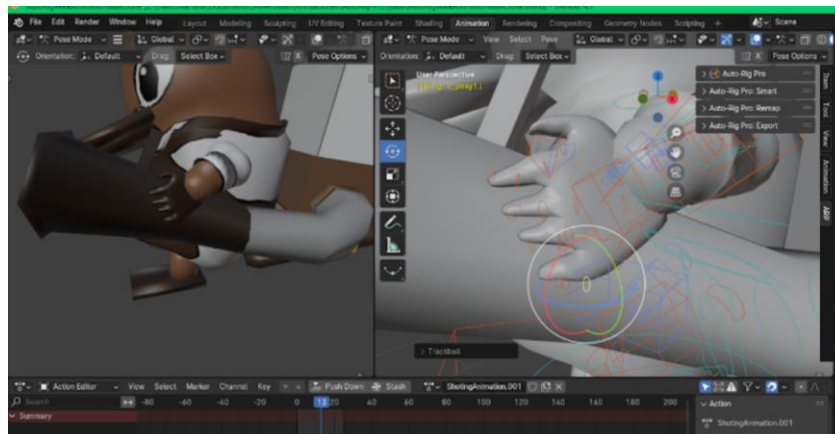


Figura 28. Animación de disparo mano izquierda.

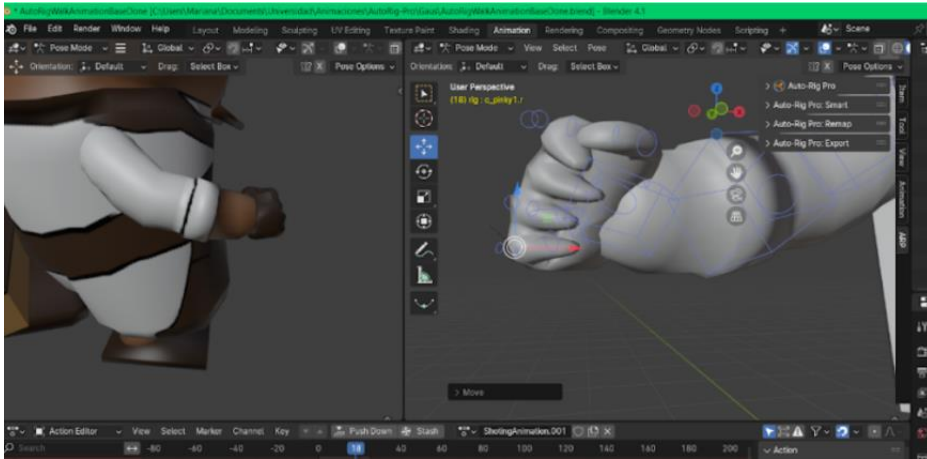


Figura 29. Animación de disparo mano izquierda.

Además, se abordó el uso del *Action Editor* para la eliminación y agregación de *keyframes*, un paso esencial para personalizar las animaciones base importadas desde Mixamo. Este editor permitió una manipulación precisa de los *keyframes*, facilitando la eliminación de aquellos que no se alineaban con la visión deseada para la animación. Además, se agregaron nuevos *keyframes* para introducir ajustes específicos que enriquecieran la acción y la expresividad del modelo, ver Figura 30.

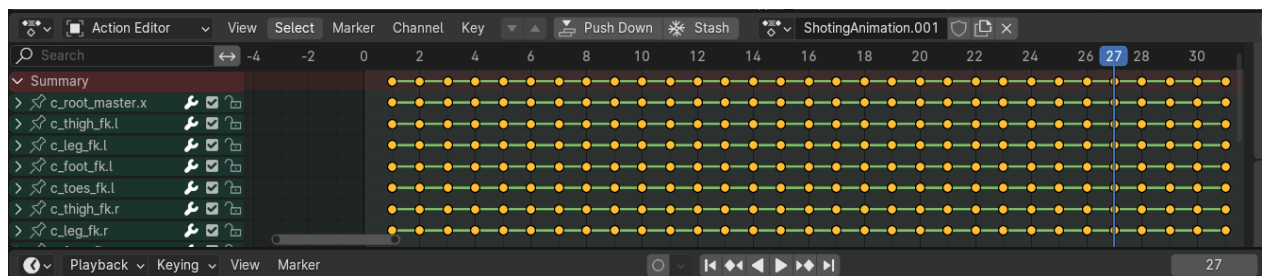


Figura 30. Action Editor para colocación de keyframes

Para definir claramente el inicio y el final de la animación, se utilizó la herramienta *Manual Frame Range*, que permitió establecer el rango de fotogramas en el que se ejecutaría la animación. Este enfoque no solo mejoró la fluidez y la coherencia de los movimientos, sino que también garantizó que las animaciones personalizadas se integraran de manera efectiva en la narrativa y mecánicas del videojuego, ofreciendo así una experiencia más envolvente para el jugador, ver figura 31.

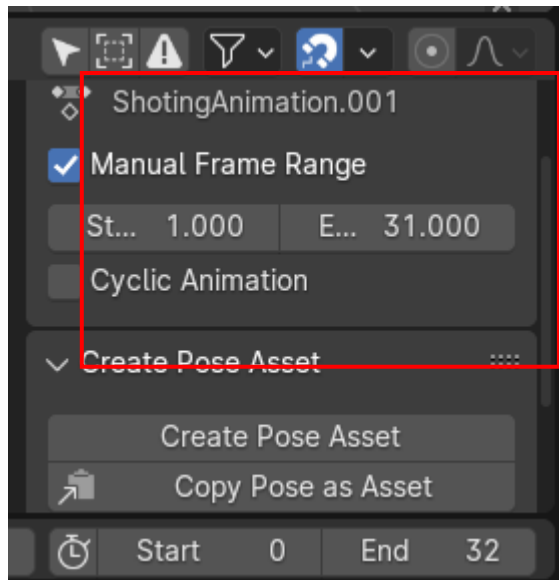


Figura 31. Establecimiento de rango en el que se ejecuta la animación

1.3.3.2 Pruebas

Para finalizar la etapa del uso de Blender, se realizaron pruebas de las animaciones ajustadas. Este paso es crucial porque ayudó a verificar que todas las acciones y movimientos se ejecuten de manera adecuada y natural dentro del contexto del videojuego. Durante las pruebas, se prestó especial atención a elementos como el movimiento de la cola, que debía reflejar la agilidad y el carácter del modelo, así como la correcta articulación de los brazos y otras extremidades.

Cada animación se revisó minuciosamente para asegurar que todos los aspectos, desde la sincronización hasta la fluidez de los movimientos, estuvieran alineados con las expectativas de diseño. Este proceso no solo permitió identificar y corregir posibles errores en la animación, sino que también garantizó que el resultado final contribuyera a una experiencia de juego inmersiva y visualmente atractiva para el jugador, ver Figura 32. Lo cual nuevamente influye en darle una caracterización al personaje por medio de los movimientos realizados.

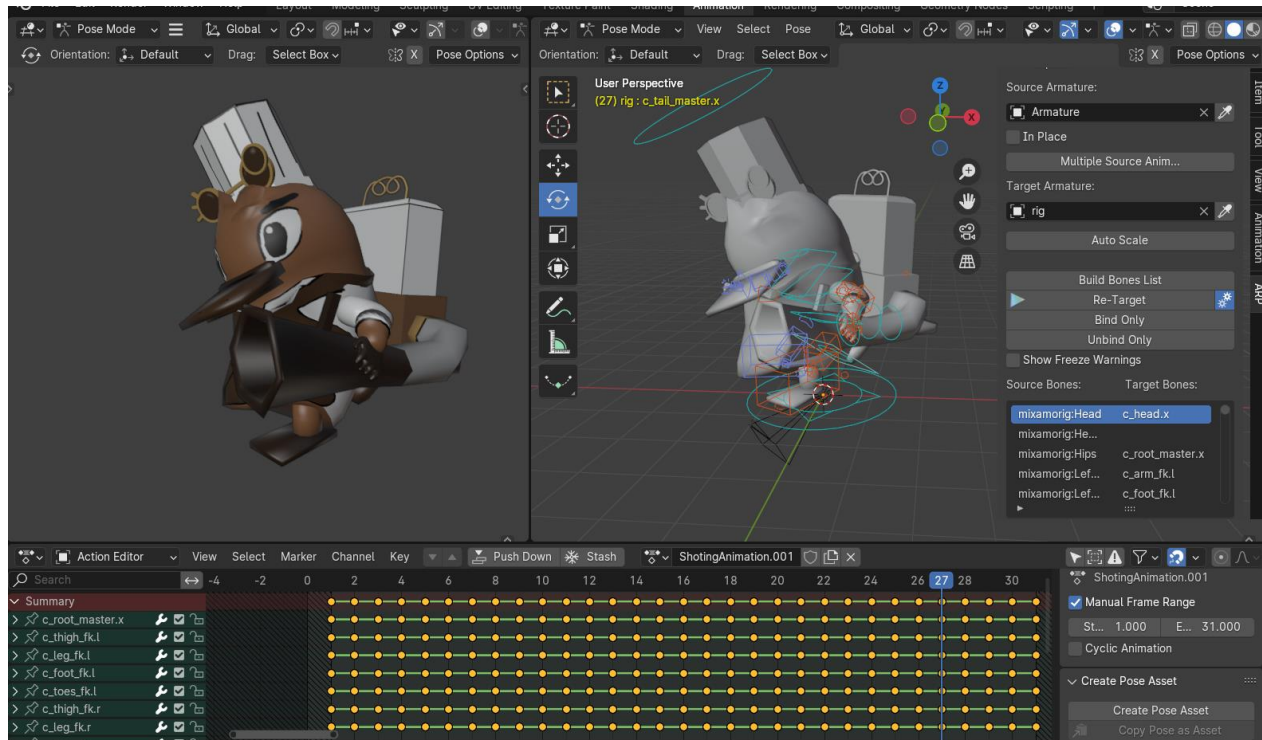


Figura 32. Resultado de animación en Blender

1.3.4 Integración a Unity

Se abordó el proceso de integración de las animaciones finalizadas en Blender al motor de desarrollo Unity. Esta etapa es esencial para garantizar que las animaciones se implementen correctamente dentro del entorno del videojuego, permitiendo su uso en diversas interacciones y mecánicas del juego. La integración requiere la exportación de las animaciones en un formato compatible, asegurando que todas las características y ajustes realizados en Blender se mantengan intactos durante el proceso de importación.

1.3.4.1 Exportación archivo

Las animaciones se exportaron en formato FBX, el cual es ampliamente reconocido y soportado por Unity. Este formato permite incluir tanto la geometría del modelo como las animaciones, asegurando que todos los ajustes realizados se conserven en la transición. Además, que también se configura para que mantuviera la textura.

Al exportar, se prestó atención a los parámetros de configuración, como la escala y la orientación, para evitar problemas durante la importación en Unity. Este proceso no solo facilitó la integración de las animaciones en el motor, sino que también sentó las bases para su implementación efectiva en la dinámica del videojuego.

1.3.4.2 Creación de Game Object

Se procedió a la creación de un *Game Object* en Unity, que es fundamental para organizar y gestionar los modelos y sus respectivas animaciones dentro del proyecto. Al

abrir el proyecto en Unity, se inició la creación de un nuevo *Game Object* que serviría como contenedor para el modelo animado.

Este *Game Object* permitió agrupar todas las componentes necesarias, facilitando la manipulación y control del modelo durante la ejecución del juego. Al estructurar el *Game Object* de esta manera, se establece un marco organizado en el que se integrarán las animaciones y se gestionarán las interacciones del modelo dentro del entorno del videojuego.

1.3.4.3 Componente animator

Se abordó la incorporación del componente *Animator* al *Game Object* previamente creado. Este componente ayudó para la gestión de las animaciones del modelo, ya que permitió la interacción con el *Animator Controller*, que alberga el árbol de estados de animación. Al agregar el componente *Animator*, se configuró para utilizar el controlador específico que se había creado previamente, lo que permite que todas las animaciones importadas sean accesibles y controlables dentro de Unity, ver Figura 33.

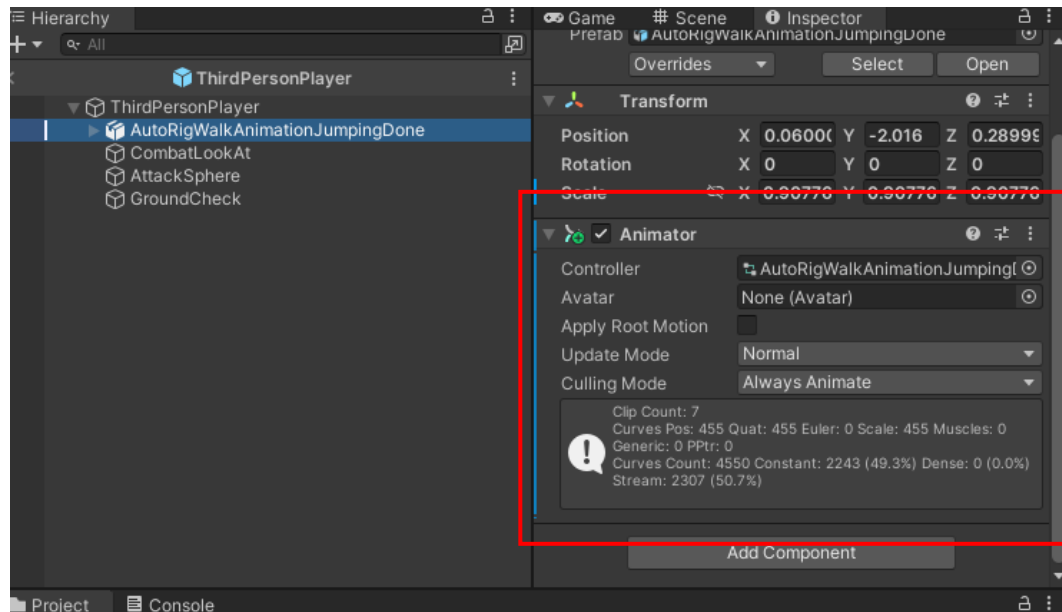


Figura 33. Pantalla del componente Animator con el Game Object como controller.

Esta integración asegura que las transiciones entre animaciones se realicen de manera fluida y que las interacciones del modelo con el entorno del juego respondan correctamente a las acciones del jugador.

1.3.5 Arbol de estados de animación

Se exploró el diseño del árbol de estados de animación, una herramienta clave para gestionar las diferentes animaciones que el modelo puede ejecutar. El árbol de estados ayudó a definir y organizar las transiciones entre las animaciones, lo que es vital para la coherencia del comportamiento del personaje dentro del juego.

1.3.5.1 Creación del árbol de estados

Continuamente se realizó la creación del árbol de estados de animación en Unity. Utilizando la interfaz de *Animator*, se comenzaron a añadir los diferentes estados que representan cada una de las animaciones del modelo, ver Figura 34.

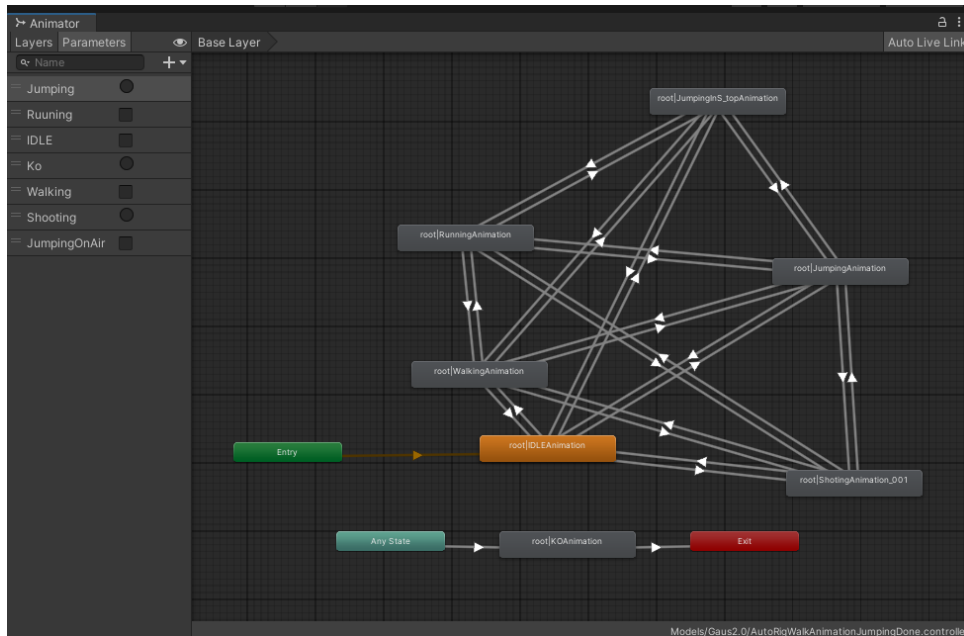


Figura 34. Resultados del arbol de estados de animación

Este árbol se construyó de manera que cada estado estuviese conectado a transiciones que facilitan el cambio de una animación a otra, dependiendo de las condiciones predefinidas, ver Figura 35.

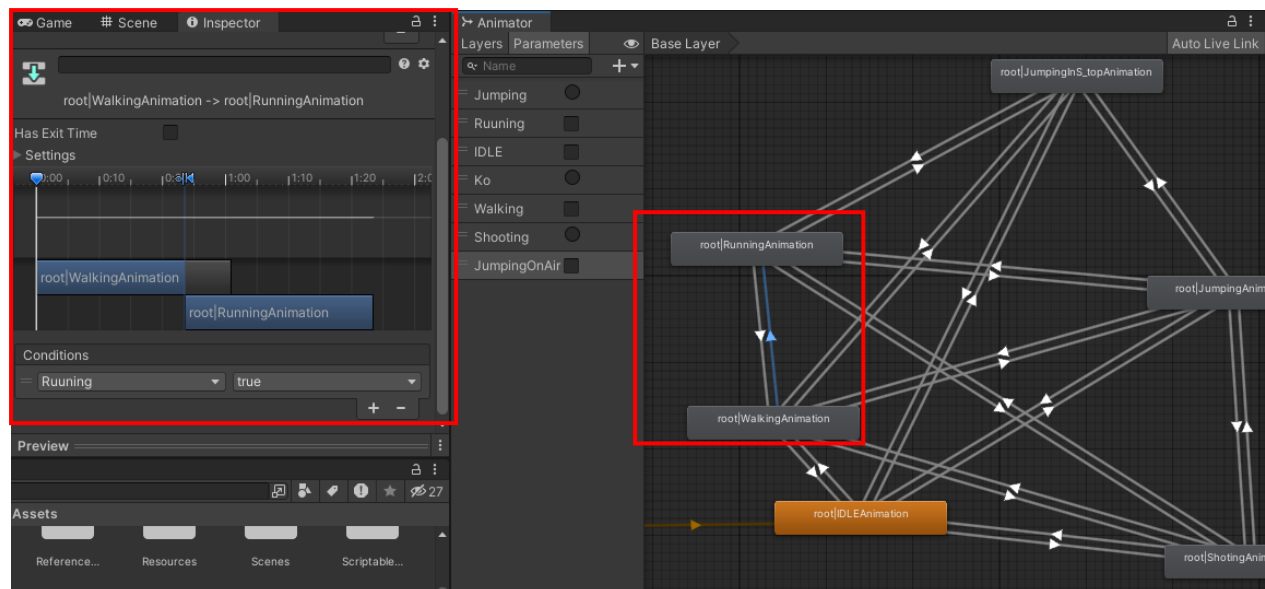


Figura 35. Configuración de flecha de transición

La correcta estructuración de este árbol es crucial para asegurar que las animaciones se ejecuten en el momento adecuado, mejorando la fluidez del movimiento y la respuesta del personaje a las acciones del jugador.

1.3.5.2 Definición de parámetros

Se colocaron parámetros que controlan las transiciones dentro del árbol de estados de animación. Se implementaron booleanos y *triggers* para gestionar las condiciones que activan las diferentes animaciones, ver Figura 36.



Figura 36. Parámetros colocados

Por ejemplo, un parámetro booleano puede utilizarse para determinar si el personaje está en movimiento o en reposo, mientras que un *trigger* podría activarse al ejecutar una acción específica, como un salto o un ataque. Esta configuración nos permitió una gestión dinámica y precisa de las animaciones, asegurando que el personaje responda adecuadamente a las acciones del jugador y mantenga una coherencia en su comportamiento, ver Figura 37.



Figura 37. Índice para identificar *Bools* y *Triggers*

1.3.6 Uso de scripts

Se centró en el uso de scripts para controlar las animaciones y la lógica del juego, permitiendo una interacción más rica y compleja entre el jugador y el entorno.

1.3.6.1 Creación de scripts principal

Se llevó a cabo la generación de varios scripts principales para cada personaje, que se designó con un nombre representativo, como "*GausAnimation*" y así por cada nombre de los correspondientes modelos. Estos scripts son el núcleo que gestiona todas las interacciones del personaje, incluyendo las llamadas a las diferentes funciones de animación definidas anteriormente. A través de estos scripts, se pueden controlar las animaciones en respuesta a las entradas del jugador, permitiendo que el personaje se mueva, ataque o realice otras acciones de acuerdo con la lógica del juego. Esta estructura de scripting es fundamental para garantizar que las animaciones se integren de manera efectiva con la jugabilidad, ver Figura 38.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class GausAnimations : MonoBehaviour
{
    [SerializeField] private Animator _animator;

    public void WalkingAnimation(bool walkingAnimation)
    {
        _animator.SetBool("Walking", walkingAnimation);
    }

    public void RuningAnimation(bool runningAnimation)
    {
        _animator.SetBool("Ruuning", runningAnimation);
    }

    public void Jumping()
    {
        _animator.SetTrigger("Jumping");
    }

    public void JumpingInStop(bool jumpingOnAirAnimation)
    {
        _animator.SetBool("JumpingOnAir", jumpingOnAirAnimation);
    }

    public void ShootAnimation()
    {
        _animator.SetTrigger("Shooting");
    }

    public void KO()
    {
        _animator.SetTrigger("Ko");
    }
}
```

Figura 38. Script principal – Ejemplo: *GausAnimations*

1.3.6.2 Llamada de métodos

Se procedió a la realización de las llamadas a los métodos dentro del script principal. Cada función de animación creada (Ejemplo: en "GausAnimation") se invoca en momentos específicos del juego, como cuando el jugador presiona un botón o al alcanzar ciertos eventos dentro de la lógica del juego, ver Figura 39.

```
private void HandleInputs()
{
    //Jump -space
    if(Input.GetButtonDown("Jump") && isGrounded)
    {
        velocity.y = Mathf.Sqrt(jumpHeight * -2f * gravity);
        GameManager.Instance.PlayerDataUpdate(PlayerDataEnum.jumps);
        AudioManager.Instance.PlayOneShot(FMODEvents.instance.gaussJump, transform.position);
        _animations.Jumping();
    }
    velocity.y += gravity * Time.deltaTime;
}
```

Figura 39. Script controlador del personaje Gaus – Ejemplo: Animación de salto

Esta invocación de métodos no solo permitió que las animaciones se ejecuten en el momento adecuado, sino que también asegura que el modelo reaccione de manera fluida y coherente a las acciones del jugador, mejorando así la experiencia de juego.

1.4 Evaluación de Calidad y Retroalimentación

Por último, se abordó la evaluación de calidad y retroalimentación, un paso crucial en el proceso de desarrollo para garantizar que las animaciones y mecánicas del juego cumplan con los estándares deseados. Durante esta fase, se realizaron pruebas exhaustivas de las animaciones en el contexto del juego, observando su rendimiento y la respuesta del jugador. La retroalimentación recogida de las pruebas ayuda a identificar áreas de mejora y ajustes necesarios, asegurando que el producto final sea de alta calidad y ofrezca una experiencia de usuario envolvente. Se decidió hacer un “*PlayTesting*” en la Universidad del Valle de Guatemala, en donde a cada participante se le brindó un formulario para que llenara antes y después de haber jugado el videojuego y que evaluara las diversas áreas de este; entre ellos el de las animaciones. Este proceso de evaluación y ajuste es fundamental para optimizar la jugabilidad y el disfrute del jugador, y para garantizar que las animaciones se integren sin problemas en la narrativa y mecánicas del videojuego.

VI. Resultados

Se completó el proceso de *rigging* para los cuatro modelos seleccionados, permitiendo que cada uno contara con una estructura esquelética funcional. Esta estructura proporcionó la base para la movilidad y la manipulación de los modelos en las animaciones posteriores.

Para realizar el *rigging* de los cuatro modelos seleccionados, se utilizó la herramienta Auto-Rig Pro de Blender. Primero, se colocaron los indicadores de posición en las áreas clave del modelo, permitiendo a la herramienta realizar una detección automática de los huesos principales en la estructura. Durante este proceso, algunos huesos no coincidieron de forma precisa con la anatomía de los modelos, por lo que fue necesario realizar ajustes manuales para optimizar la alineación y asegurar que el esqueleto se adaptara correctamente al cuerpo de cada modelo.

Asimismo, se añadieron huesos adicionales en áreas específicas que requerían movimientos complejos. Posteriormente, se completó el proceso de *binding* para vincular el *mesh* del modelo al *rig*, asegurando que los movimientos de la estructura esquelética se transmitieran fielmente al exterior del modelo y quedaran listos para las animaciones. A continuación, se presentarán los resultados del *rigging* de cada modelo.

El proceso de *rigging* del personaje principal, Gaus, un ornitorrinco, requirió una adaptación considerable debido a las particularidades anatómicas del modelo en comparación con un humanoide. Al iniciar con el primer modelo de Gaus, se observó que la estructura ósea debía ajustarse proporcionalmente, ya que el cuerpo de un ornitorrinco es más pequeño y sus huesos vertebrales son relativamente más largos en comparación con un modelo humano estándar, ver Figura 40.

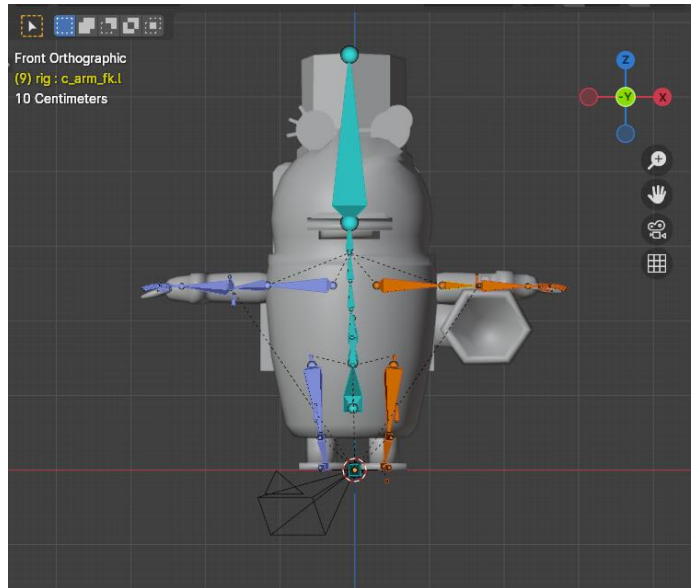


Figura 40. Resultados del *rigging* del cuerpo de Gaus.

Además, se modificaron las extremidades, dado que las patas de Gaus son más cortas y robustas que las de un humano. Así como también se ajustaron las manos, ver Figura 41.

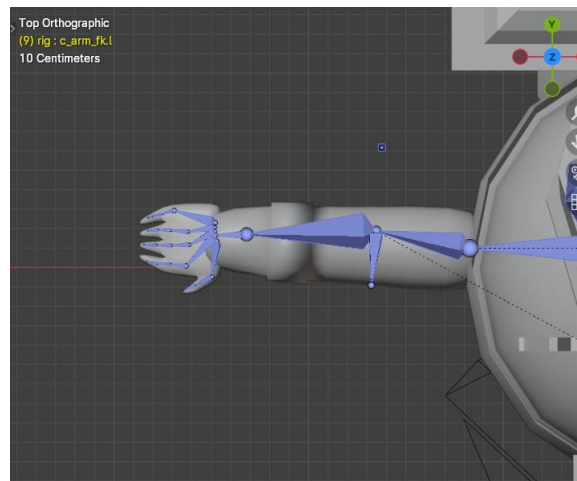


Figura 41. Resultados del *rigging* de las manos de Gaus.

Fue necesario agregar huesos específicos en la zona de la cola, ya que la herramienta Auto-Rig Pro, que se empleó para el rigging, no detecta correctamente las estructuras de ciertos animales salvo casos específicos, como aves, entre otros. Por lo tanto, el *rigging* de Gaus se realizó siguiendo un enfoque humanoide, al cual se le añadieron huesos adaptados al tamaño y proporciones del ornitorrinco, sin necesidad de eliminar huesos existentes, sino ajustándolos para mantener la coherencia con la anatomía del personaje. La Figura 42 muestra el rigging de la cola en vista de perfil.

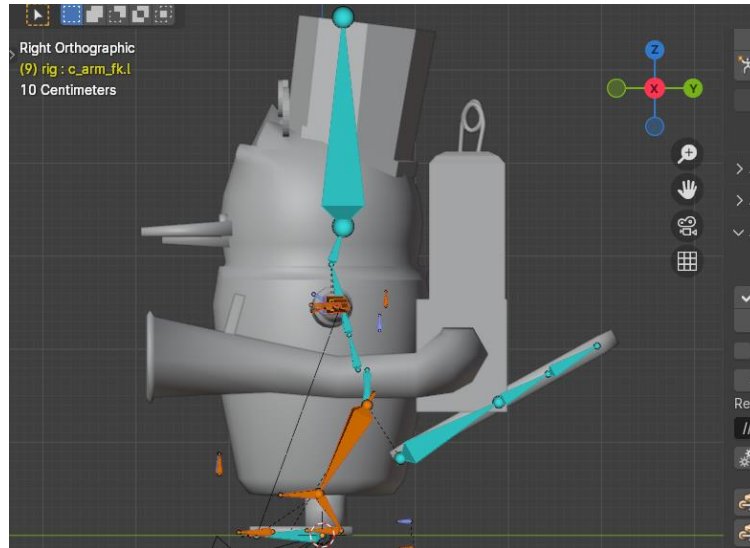


Figura 42. Resultados del *rigging* de las manos de Gaus

Por otro lado, en el caso del segundo y tercer modelo, correspondientes al Dingo normal y el Dingo jefe, se aplicaron ajustes similares en el *rig*, adaptándolo a sus tamaños respectivos y agregando huesos adicionales, ver figura 43.

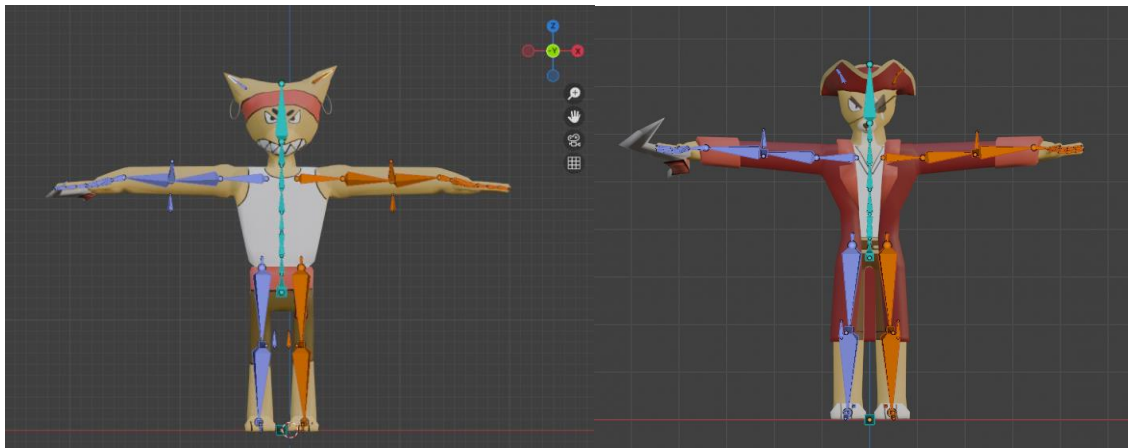


Figura 43. Resultados del *rigging* del cuerpo de ambos Dingos.

Sin embargo, para estos modelos fue necesario un mayor número de modificaciones, agregando más huesos y eliminando algunos. Además de incorporar huesos en la cola, se añadieron huesos en las orejas para permitir un movimiento sutil en esta área, ver figura 44.

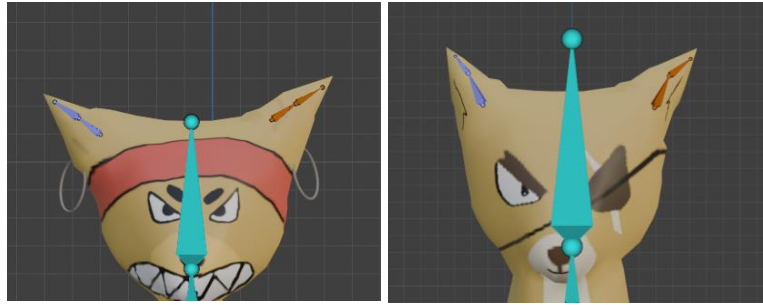


Figura 44. Resultados del *rigging* de las orejas de ambos Dingos.

En cuanto a la simplificación, se eliminó el hueso correspondiente al dedo medio de ambas manos, dado que tanto el Dingo normal como el Dingo jefe solo tienen cuatro dedos en sus manos, ver figura 45.

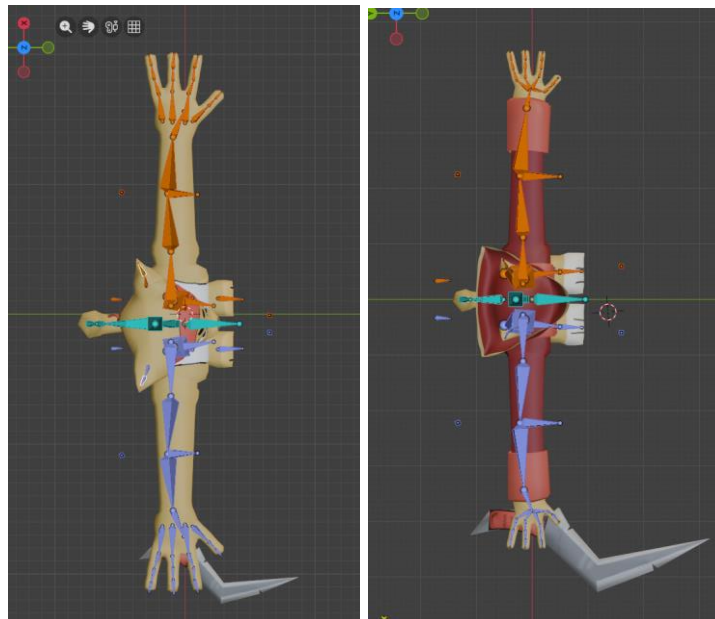


Figura 45. Resultados del *rigging* de las manos de ambos Dingos.

En el cuarto modelo, correspondiente al búho (otro personaje villano), el *rigging* fue completamente diferente debido a que la estructura del búho no tiene similitud con un humanoide. Auto-Rig Pro ofrece una herramienta específica para generar un *rigging* básico para aves, lo cual facilitó la creación de la estructura inicial. En este modelo también se ajustó el posicionamiento del cuerpo y se agregó un hueso en la cola ver figura 46.



Figura 46. Resultados del *rigging* de la cola del Buho.

Además, los huesos de las alas se ajustaron cuidadosamente para alinearse con las alas del modelo. Para las plumas, se añadieron los huesos necesarios para lograr una animación fluida, optimizando la cantidad de huesos, ya que agregar un hueso para cada pluma en un modelo 3D caricaturesco podría distorsionar la apariencia del modelo. Por ello, se eliminaron algunos huesos en las alas, en áreas de plumas donde no eran esenciales para la animación ver figura 47.

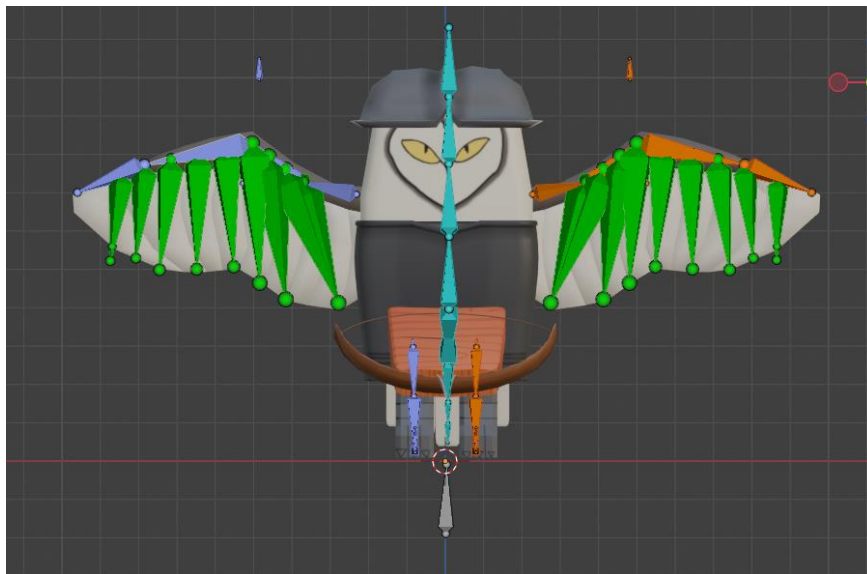


Figura 47. Resultados del *rigging* del Buho.

Finalmente, para este modelo fue necesario agregar un hueso adicional en la zona inferior, cerca de las patas, específicamente en el centro. Este hueso se utilizó para vincular el arma del personaje, una ballesta, permitiendo una integración y movimiento coherentes entre el arma y el personaje durante la animación, ver figura 48.

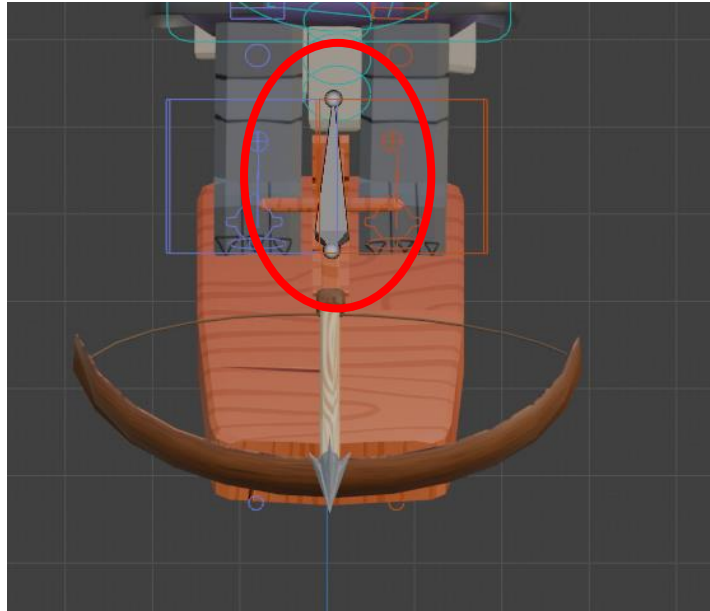


Figura 48. Resultados del hueso extra para el arma del buho.

En cuanto el resultado de las animaciones, para cada modelo se crearon animaciones específicas que complementan su personalidad y función dentro del juego. En el caso de Gaus, el ornitorrinco, se desarrollaron las siguientes animaciones: Idle (inactivo), KO (muerte), salto, correr, caminar, apuntar el arma y una animación extra, que consiste en un salto prolongado para simular una caída larga. Ver figura 49, que ejemplifica la animación de KO (morir).

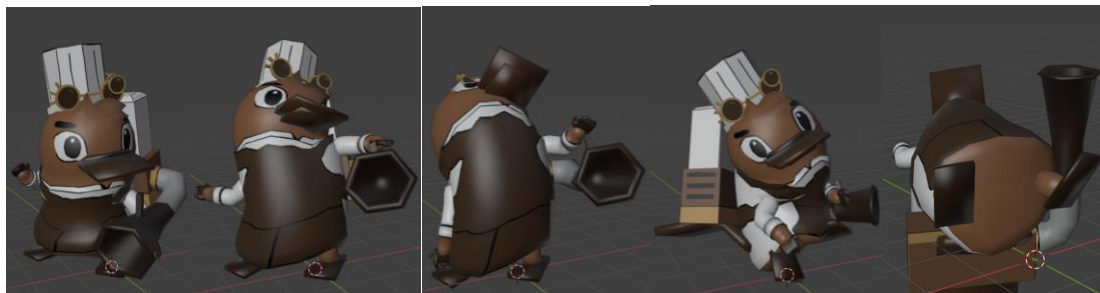


Figura 49. Resultados de animación KO (morir) de Gaus.

Para los modelos del Dingo normal y el Dingo jefe, se realizaron animaciones similares, adaptadas a su rol en el juego: Idle, KO, correr, caminar y lanzar el arma. Ver figuras 50 y 51, que ejemplifican una de las animaciones realizadas por cada uno.



Figura 50. Resultados de animación caminar del Dingo normal.



Figura 51. Resultados de animación de lanzar arma del Dingo jefe.

Por último, para el modelo del búho, se llevó a cabo únicamente la animación extra que consistió en volar mientras sostiene su arma, ver figura 52.



Figura 52. Resultados de animación extra de volar del Buho.

Por otro lado, los árboles de transiciones de animaciones en Unity, gestionados a través del componente *Animator*, permitieron controlar de manera efectiva el flujo de animaciones de cada modelo. A continuación, se muestran los árboles de transiciones correspondientes a cada uno de los personajes.

Para el modelo de Gaus, las transiciones entre los estados de Correr, *Idle* y Caminar y saltar en el aire utilizaron parámetros booleanos, lo que permitió activar o desactivar estos estados en función del movimiento del personaje. Además, las animaciones de Saltar, *KO* y Disparar se gestionaron mediante *triggers*, que se activaron en momentos específicos, como al recibir daño o al ejecutar un ataque y saltar rápidamente, ver figura 53.

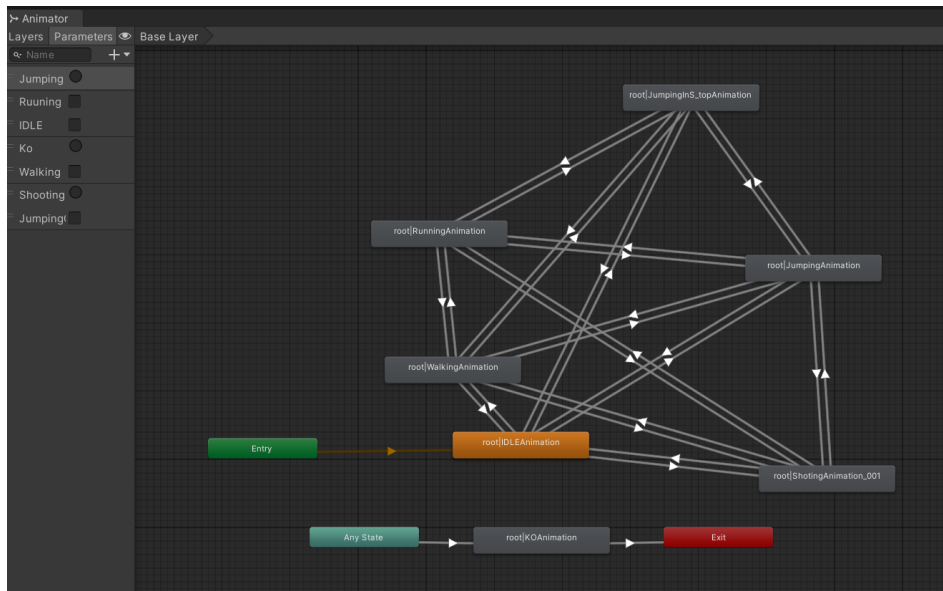


Figura 53. Arbol de transiciones de animaciones de Gaus.

En el caso de los modelos del Dingo, tanto el normal como el jefe, el árbol de transiciones abarcó los estados de *Idle*, Caminar, Correr, Lanzar arma y *KO*. Para estos modelos, se utilizaron parámetros con variables booleanas para las transiciones de *Idle*, Caminar y Correr, lo que permitió activar o desactivar estos estados según el movimiento del personaje. Además, las animaciones de *KO* y Lanzar arma se gestionaron mediante *triggers* que se activaron en momentos específicos, como cuando el personaje recibía daño o realizaba un ataque, ver figura 54.

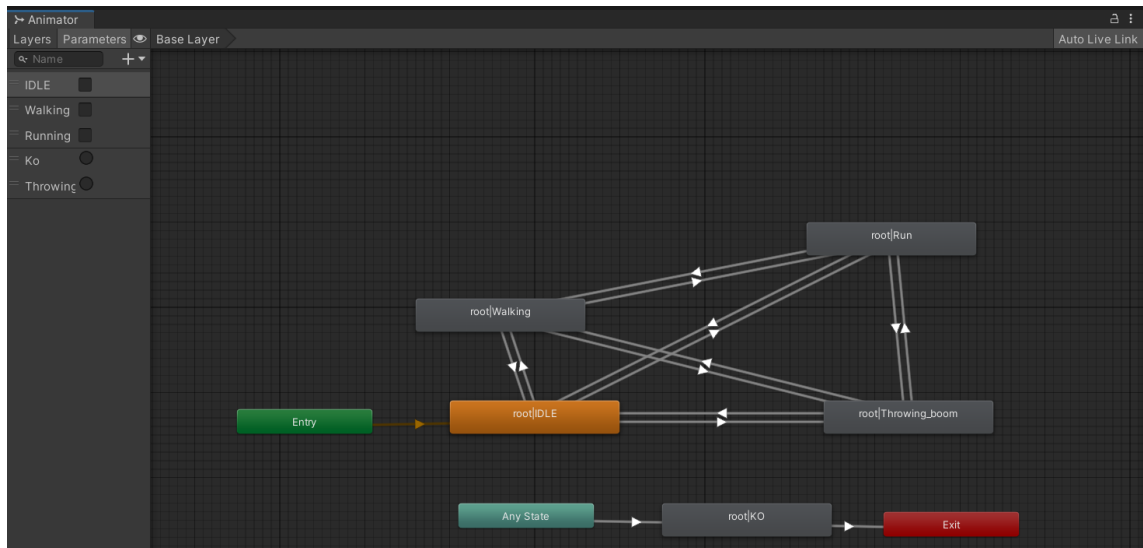


Figura 54. Arbol de transiciones de animaciones de Dingo y Dingo jefe.

Para el modelo del búho, se desarrolló un árbol de transiciones más simplificado, dado que la única animación ejecutada fue la animación extra que consistía en volar mientras sostenía su arma. Esta animación se activó bajo condiciones específicas, dependiendo de la interacción del jugador y la lógica del juego, ver figura 55.

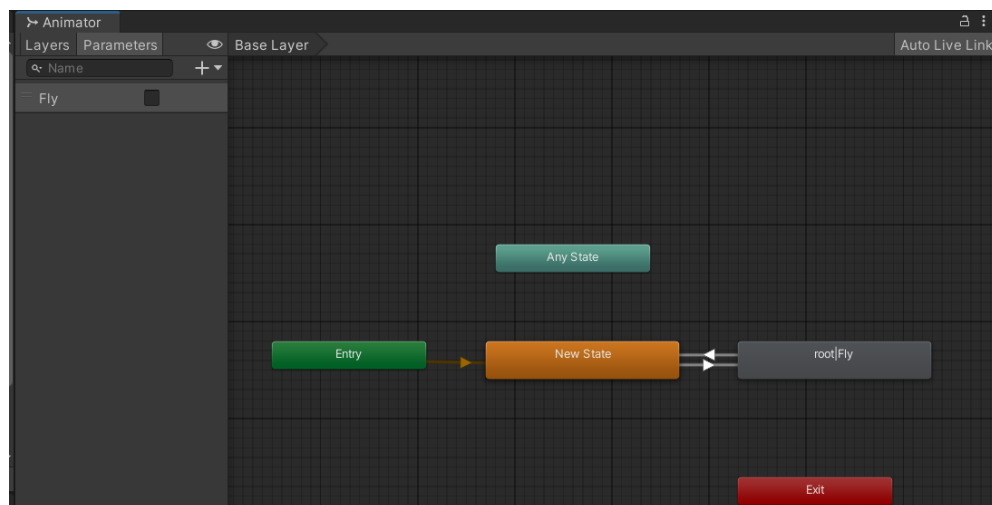
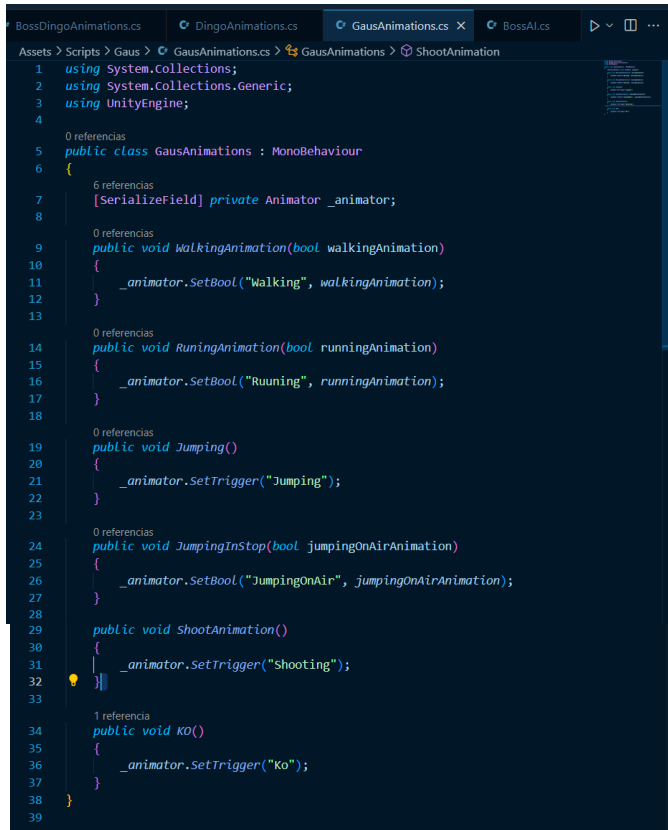


Figura 55. Arbol de transiciones de animación del Buho.

Se desarrollaron diversos *scripts* en Unity, para integrar las animaciones a los movimientos de los personajes, garantizando una interacción fluida y dinámica durante el juego. Cada script fue diseñado con una clase correspondiente al nombre del modelo, como *GausAnimations*, *DingoAnimations* y *BossDingoAnimations*, entre otros. Dentro de cada

clase, se definieron métodos específicos para cada animación, a excepción de la animación *Idle*, que se inicializa automáticamente al comenzar el juego.

Por ejemplo, en el *script* de *GausAnimations*, se implementaron métodos como *WalkingAnimation*, *RuningAnimation*, *Jumping*, *JumpingInStop*, *ShootAnimation* y *KO*. Dependiendo de la naturaleza de la animación, se utilizaron parámetros booleanos o triggers para activar las transiciones. Un ejemplo de esta implementación es el siguiente:



```
Assets > Scripts > Gaus > GausAnimations.cs > GausAnimations > ShootAnimation
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 0 referencias
6 public class GausAnimations : MonoBehaviour
7 {
8     6 referencias
9     [SerializeField] private Animator _animator;
10
11     0 referencias
12     public void WalkingAnimation(bool walkingAnimation)
13     {
14         _animator.SetBool("Walking", walkingAnimation);
15     }
16
17     0 referencias
18     public void RuningAnimation(bool runningAnimation)
19     {
20         _animator.SetBool("Runing", runningAnimation);
21     }
22
23     0 referencias
24     public void Jumping()
25     {
26         _animator.SetTrigger("Jumping");
27     }
28
29     0 referencias
30     public void JumpingInStop(bool jumpingOnAirAnimation)
31     {
32         _animator.SetBool("JumpingOnAir", jumpingOnAirAnimation);
33     }
34
35     public void ShootAnimation()
36     {
37         _animator.SetTrigger("Shooting");
38     }
39
40     1 referencia
41     public void KO()
42     {
43         _animator.SetTrigger("ko");
44     }
45 }
```

Figura 56. Script de animaciones de Gaus.

Posteriormente, en el script que maneja la mecánica del modelo, se inicializó la conexión con las animaciones utilizando la declaración: “private GausAnimations _animations;”. Esto permitió definir el componente responsable de llamar a las animaciones: “_animations = GetComponent<GausAnimations>();”,

A continuación, se llamaron los métodos correspondientes según las entradas del jugador. Por ejemplo, al recibir la entrada de salto, se invocó el método, ver figura 57:

```
//Jump -space
if(Input.GetButtonDown("Jump"))
{
    velocity.y = Mathf.Sqrt(jum
    GameManager.Instance.Player
    AudioManager.Instance.PlayC
    _animations.Jumping();
}
```

Figura 57. Llamada a los métodos correspondientes de animaciones de Gaus.

De manera similar, para las animaciones de correr y caminar, se utilizaron los métodos, ver figura 58.

```
_animations.RuningAnimation(_isRunning);
_animations.WalkingAnimation(_horizontal != 0 || _vertical != 0);
_animations.JumpingInStop(!isGrounded);
```

Figura 58. Llamada a los métodos correspondientes de animaciones de Gaus.

En el caso de los villanos, como el Dingo normal, el Dingo jefe y el búho, que operan con inteligencia artificial, la implementación se realizó de manera similar, pero con ajustes específicos para reflejar su comportamiento autónomo. Para estos modelos, la condición que activa la animación de caminar se estableció de la siguiente manera, ver figura 59.

```
if (!isTiedUp)
{
    agent.SetDestination(player.position);
    // ! ANIMATION:
    _animations.RuningAnimation(agent.velocity.magnitude > 0.1f);
}
```

Figura 59. Llamada a los métodos correspondientes de villanos con IA.

Se accedió a la propiedad “*agent.velocity.magnitude*” para determinar la velocidad actual del personaje controlado por inteligencia artificial (IA) en el juego.

Una vez implementadas y verificadas las animaciones de los personajes en Unity, se procedió a realizar un *GameTesting* de juego para evaluar la experiencia de los jugadores ver figura 60.



Figura 60. Resultado del juego.

El juego fue puesto a disposición de un grupo aleatorio de participantes, quienes interactuaron con los diferentes modelos animados en diversas situaciones del entorno del juego.

Después de la sesión de juego, los participantes completaron un formulario diseñado para recopilar sus impresiones sobre las animaciones. Este cuestionario incluyó preguntas específicas sobre la percepción de las animaciones, etc. La retroalimentación obtenida, fue analizada para identificar áreas de mejora y confirmar la efectividad de las animaciones implementadas. Los resultados se muestran en la siguiente tabla. Ver tabla 2.

Pregunta	Tipo de Respuesta	Resultado Promedio	Porcentaje de Respuestas Afirmativas
1. ¿Qué tan inmersiva le pareció la experiencia visual del juego gracias a las animaciones de los personajes?	Escala 1-10	Promedio: 7.6	-
2. ¿Las animaciones lograron transmitir la narrativa de "Gaus y la Revolución Ornitoreense"?	Sí/No	-	80% Sí
3. ¿Se sintió más conectado o empático hacia los personajes debido a sus animaciones?	Sí/No	-	76% Sí

4. ¿Las animaciones fueron fluidas y se adaptaron al ritmo del juego?	Escala 1-5	Promedio: 4.0	-
5. ¿Identificó los distintos estados de animación (caminar, correr, saltar, etc.)?	Sí/No	-	80% Sí
6. ¿Qué tan satisfactorias le parecieron las transiciones entre animaciones?	Escala 1-10	Promedio: 8.0	-
7. ¿Notó los principios de animación como anticipación y encogerse en los movimientos de los personajes?	Sí/No	-	76% Sí
8. ¿Las animaciones contribuyeron al atractivo visual del juego?	Escala 1-10	Promedio: 7.3	-
9. ¿Las animaciones mejoraron su comprensión de la trama?	Escala 1-5	Promedio: 4.3	-
10. ¿Las animaciones fueron creíbles dentro del contexto caricaturesco y de ciencia ficción del juego?	Escala 1-5	Promedio: 4.2	-

Tabla 2. Resultados de formulario del *GameTesting*

Para cada pregunta en escala (del 1 al 5 o del 1 al 10), el promedio fue calculado multiplicando cada opción por la cantidad de votos recibidos, sumando todos los valores y luego dividiendo por el total de votos (25). Para las preguntas de "Sí/No", el porcentaje de respuestas afirmativas fue calculado en base a 25 respuestas totales.

Por último, los principios de animación exitosos en el transcurso de las dieciocho animaciones generadas fueron: estirar y encoger, anticipación, puesta en escena, pose a pose, acción continua y superpuesta, exageración, acelerar y desacelerar movimiento en arcos, acción secundaria y atractivo.

VII. Análisis de resultados

La implementación de los sistemas de rigging y de animación demostró ser una herramienta clave para alcanzar los objetivos de diseño planteados. El rigging de cada personaje se llevó a cabo con un enfoque detallado, creando un esqueleto que otorgara flexibilidad y precisión en el control de las extremidades, la cola y las articulaciones principales. Este enfoque permitió que con la ayuda de herramientas como AutoRig-Pro los movimientos de cada personaje fueran fluidos y naturales, facilitando la implementación de diversas animaciones sin que se produjeran distorsiones en la malla. Al garantizar un control preciso, el rigging resultó esencial para mantener la coherencia visual y la integridad de los personajes dentro del entorno del juego, especialmente en situaciones complejas como los movimientos de combate o en las transiciones de estado en el árbol de animaciones.

El sistema de animaciones fue configurado cuidadosamente para lograr transiciones suaves y naturales entre diferentes acciones, como caminar, correr, saltar y disparar. Se implementó un árbol de animaciones organizado en nodos jerárquicos que facilitaron la transición entre acciones simples y complejas, logrando evitar interrupciones bruscas entre ellas. Cada nodo en el árbol de transiciones permitía que las animaciones se sucedieran de manera orgánica, contribuyendo a una experiencia de juego fluida y sin interrupciones. La estructura jerárquica del árbol no solo optimizó la progresión de movimientos, sino que también permitió una transición eficiente entre las animaciones de combate y las de interacción con el entorno, como cuando el personaje está en reposo o en movimiento activo como se observan en la sección de resultados.

La integración del sistema de animación con los scripts de control resultó también eficaz. Se programaron scripts que coordinaban las animaciones de acuerdo con las acciones del jugador y las situaciones contextuales en el juego. Esta integración aseguró que el personaje respondiera correctamente a las entradas del usuario y se moviera en sincronía con los eventos en pantalla, mejorando tanto la jugabilidad como la inmersión del jugador. Además, el *rigging* permitía que estas animaciones se adaptaran a distintos estados sin problemas técnicos, manteniendo la estabilidad de las transiciones y la coherencia visual del personaje.

Paralelamente, gracias a los resultados obtenidos en el *GameTesting*, estos indicaron que las animaciones implementadas lograron un alto nivel de aceptación entre los participantes, cumpliendo de manera exitosa con los objetivos de diseño planteados. En

general, los puntajes promedio para las preguntas en escala se situaron consistentemente en el rango superior: en una escala del 1 al 10, la experiencia inmersiva recibió un promedio de 7.6, y la satisfacción con las transiciones de animación alcanzó un 8.0. Esto sugirió que las animaciones no solo proporcionaron un nivel de inmersión adecuado, sino que también cumplieron con las expectativas de continuidad y fluidez en los movimientos, un aspecto fundamental en la narrativa interactiva del juego.

Además, el 80% de los participantes consideró que las animaciones lograron transmitir eficazmente la narrativa de "Gaus y la Revolución Ornitorense", lo que indica que los elementos visuales contribuyeron significativamente a la comprensión de la historia y a la conexión emocional con los personajes. La percepción positiva de los principios de animación aplicados (76% los notó) y la valoración general del atractivo visual de las animaciones (promedio de 7.3) demuestran que las técnicas de anticipación y encogimiento, entre otras fueron efectivas, logrando una integración coherente con el entorno caricaturesco y de ciencia ficción.

Por último, la calificación promedio de 4.2 en cuanto a la credibilidad de las animaciones en el contexto del juego y el alto puntaje (4.3) en la mejora de comprensión de la trama resaltan el éxito de las animaciones en su función narrativa y visual. Estos resultados reflejan que las animaciones contribuyeron de manera significativa al atractivo visual y a la experiencia general del juego, superando las expectativas medias y validando la efectividad de la implementación para generar una experiencia inmersiva y visualmente coherente.

Por otra parte, un aspecto adicional del análisis de resultados se centra en el cumplimiento de los principios de animación fundamentales, los cuales fueron integrados en diversas áreas del sistema de animaciones. El principio de "estirar y encoger" fue implementado en las animaciones de salto, donde los personajes levantan los brazos y flexionan las rodillas al caer, logrando un cambio dinámico en la postura y añadiendo fluidez a la animación. La anticipación fue aplicada en la animación de combate, donde el personaje adopta una posición estable para disparar, reforzando la intención de la acción y creando una expectativa visual clara. El principio de "puesta en escena" se mantuvo en las animaciones, evitando detalles superfluos que pudieran desviar la atención del jugador y logrando así que el foco visual permaneciera en la acción principal. Así como cada animación se hizo con el principio de pose a pose para alcanzar las animaciones obtenidas como se puede observar en resultados.

Principios adicionales como la acción continuada y superpuesta fueron incorporados en la animación de los personajes secundarios, como en el caso del búho, cuyas patas cuelgan para simular el efecto de la gravedad. La exageración se utilizó especialmente en las animaciones de KO, donde el movimiento abrupto y ampliado en la caída añade un toque caricaturesco y refuerza la narrativa de ciencia ficción. Se incorporaron también los principios de acelerar y desacelerar, evidentes en la animación de salto extendido, donde el personaje cae a una velocidad más lenta en la parte final del salto. El movimiento en arcos se hizo visible en la animación de la cola del personaje principal, que sigue un patrón de arco al caminar, añadiendo naturalidad a la acción y manteniendo el atractivo visual. Como acción secundaria, se animaron las orejas y cola del Dingo, agregando vivacidad y profundidad a los personajes, mientras que el atractivo se consolidó en el estilo y ritmo de las caminatas, contribuyendo a una estética coherente y visualmente llamativa.

Por ende, los resultados obtenidos reflejan un alto nivel de efectividad en la implementación de los sistemas de rigging y de animación. Estos sistemas no solo cumplieron con los principios fundamentales de la animación, sino que también contribuyeron significativamente a la inmersión y la narrativa del juego, como lo reflejan los altos puntajes en satisfacción general y conexión narrativa. La calificación promedio de 4.2 en cuanto a la credibilidad de las animaciones y el puntaje de 4.3 en la mejora de la comprensión de la trama resaltan el éxito de las animaciones en su función narrativa y visual. En general, los elementos implementados no solo superaron las expectativas medias, sino que también validaron la efectividad del enfoque en crear una experiencia inmersiva y visualmente coherente que enriquece la interacción del jugador con el mundo del juego.

VIII. Conclusiones

El desarrollo de las animaciones y sistemas de rigging para el proyecto "Gaus y la Revolución Ornitoarensa" ha sido exitoso, cumpliendo con los objetivos generales y específicos planteados al inicio del proyecto. Los resultados obtenidos en las pruebas de *GameTesting* y el análisis de cada animación validan que los esfuerzos en rigging y diseño de animación no solo alcanzaron los estándares de calidad deseados, sino que también mejoraron la inmersión y la narrativa del juego, elementos fundamentales para una experiencia interactiva efectiva.

El objetivo general de crear un sistema de animación integral que pudiera integrarse armoniosamente con la narrativa y mejorar la experiencia visual se cumplió satisfactoriamente. Las animaciones generadas lograron una alta aceptación entre los usuarios, quienes valoraron la fluidez y naturalidad de los movimientos. En este sentido, la puntuación promedio de 8.0 en las transiciones de animación y 7.6 en la experiencia inmersiva indican que los esfuerzos en rigging, estructuración del árbol de animaciones y la integración de scripts se alinearon efectivamente con las expectativas de los usuarios.

Asimismo, se lograron cumplir los objetivos específicos de manera exitosa. En primer lugar, se diseñó un *rigging* efectivo y flexible que soportó las diversas animaciones de los personajes. Los resultados evidenciaron que el *rigging* permitió movimientos fluidos y realistas en cada personaje, con especial énfasis en elementos móviles como la cola y las extremidades. Esto fue fundamental para conseguir transiciones de animación coherentes, lo que se confirmó a través de la alta puntuación en el *GameTesting*.

En segundo lugar, se implementaron los principios de animación fundamentales para mejorar el realismo y la inmersión de los personajes. El análisis detallado de los principios de animación aplicados mostró que elementos como "estirar y encoger", anticipación, acción continuada y superpuesta, entre otros, fueron correctamente integrados, contribuyendo a la naturalidad y el atractivo visual de las animaciones. Esto se vio reflejado en el 76% de los jugadores que notaron estos elementos y valoraron su contribución a la narrativa del juego.

Adicionalmente, se optimizó el sistema de animación mediante un árbol de transiciones efectivo, que aseguró transiciones suaves y sin interrupciones entre los estados de movimiento. La estructura jerárquica del árbol permitió un flujo de animaciones sin interrupciones, garantizando que los personajes se comportaran de manera coherente al interactuar con el entorno. Este logro fue respaldado por el promedio de satisfacción de 8.0 en la prueba de *GameTesting*, destacando la efectividad de las transiciones.

Finalmente, se integraron de manera eficiente las animaciones con los scripts de control, lo que aseguró la sincronización entre las acciones del jugador y las respuestas de los personajes. Esta integración fue exitosa y se tradujo en una jugabilidad fluida y una

experiencia visual coherente, lo cual fue un factor importante para alcanzar un nivel de inmersión satisfactorio en el juego.

La utilización de herramientas como Mixamo, AutoRig- Pro, Blender y Unity fue decisiva para el éxito del proyecto. Cada herramienta aportó funcionalidades especializadas que facilitaron el cumplimiento de los objetivos específicos. Mixamo fue fundamental para agilizar el proceso de creación de animaciones complejas, mientras que Blender permitió refinar el *rigging* y ajustar los detalles visuales de los personajes. AutorRig-Pro facilitó la creación de esqueletos específicos que mejoraron la adaptabilidad y naturalidad de las animaciones, y Unity, como motor principal, proporcionó un entorno integrado para la gestión de animaciones, programación y ejecución del *GameTesting*. Estas herramientas, al complementar las capacidades de *rigging* y animación, permitieron una implementación técnica sólida que respaldó los principios de animación y los objetivos de diseño.

Por ello, el desarrollo de sistemas de *rigging* y animación cumplió con los objetivos planteados al principio del proyecto y fue respaldado por resultados positivos en *GameTesting*, lo que reafirma que los métodos y herramientas seleccionadas fueron adecuadas y efectivas. Esto demuestra que el uso estratégico de principios de animación y una estructura de *rigging* optimizada no solo contribuye a mejorar la estética de los personajes, sino también a reforzar la narrativa y la inmersión del juego.

IX. Recomendaciones

A lo largo del desarrollo de "Gaus y la Revolución Ornitorense", ha quedado clara la importancia de la adaptabilidad y la capacitación continua del equipo en el uso de herramientas de animación como Blender y Mixamo. Estas herramientas ofrecen características avanzadas que no solo pueden mejorar la eficiencia en el proceso de animación, sino también la calidad del producto final. Invertir tiempo en cursos y tutoriales permitirá a los nuevos miembros del equipo familiarizarse rápidamente con las funciones esenciales, mientras que los integrantes existentes podrán profundizar en técnicas que optimicen el flujo de trabajo y enriquezcan la narrativa visual del juego.

Desde el inicio del proyecto, se había contemplado el desarrollo de un solo nivel; sin embargo, a medida que avanzó el trabajo, se evidenció la necesidad de ampliar el alcance del juego. Este proceso evolutivo dio lugar a un segundo nivel y a la introducción de nuevos personajes y villanos. Esta capacidad de expansión sugiere que en el futuro el juego puede continuar desarrollándose, permitiendo la creación de más niveles y una variedad aún mayor de personajes animados. Esto no solo enriquecerá la experiencia del jugador, sino que también abrirá nuevas posibilidades narrativas y de jugabilidad.

Durante el desarrollo, se recomienda establecer ciclos de *feedback* más frecuentes tanto con los jugadores como con los miembros del equipo. Este enfoque permitirá realizar ajustes en las animaciones y en la jugabilidad antes de alcanzar la fase final del proyecto. Un *feedback* constante asegurará que el producto no solo cumpla con las expectativas del usuario, sino que también mantenga su atractivo en un mercado altamente competitivo.

Asimismo, es fundamental mantener una documentación exhaustiva del proceso de animación, que abarque desde el rigging hasta la integración en Unity. Esta documentación servirá como una valiosa referencia para futuros proyectos, permitiendo que el conocimiento se transfiera de manera eficiente dentro del equipo. Incluir guías sobre problemas comunes y sus soluciones facilitará la resolución de inconvenientes y optimizará el tiempo de desarrollo en el futuro.

Explorar nuevas técnicas de animación, como la captura de movimiento o la integración de software de animación 2D, también se sugiere como una forma de complementar las animaciones en 3D. Estos enfoques pueden enriquecer la narrativa del videojuego, ofreciendo una experiencia más dinámica al jugador, lo cual es clave para mantener su interés en la historia y en el universo de "Gaus y la Revolución Ornitorense".

Finalmente, es esencial realizar evaluaciones periódicas del desempeño del juego y recopilar *feedback* de los usuarios una vez que el videojuego esté en el mercado. Estas evaluaciones ayudarán a identificar áreas de mejora y a implementar actualizaciones que mantengan el interés y la satisfacción de los jugadores a largo plazo. Fomentar la

colaboración con otros desarrolladores y profesionales en la industria también puede abrir nuevas oportunidades para el aprendizaje y la innovación, beneficiando el desarrollo de futuros proyectos y asegurando que "Gaus y la Revolución Ornitoreense" se mantenga relevante y atractivo para su audiencia.

X. Anexos

Resultados de formulario para el *GameTesting*

En una escala del 1 al 10, ¿qué tan inmersiva le pareció la experiencia visual del juego gracias a las animaciones de los personajes?

25 respuestas

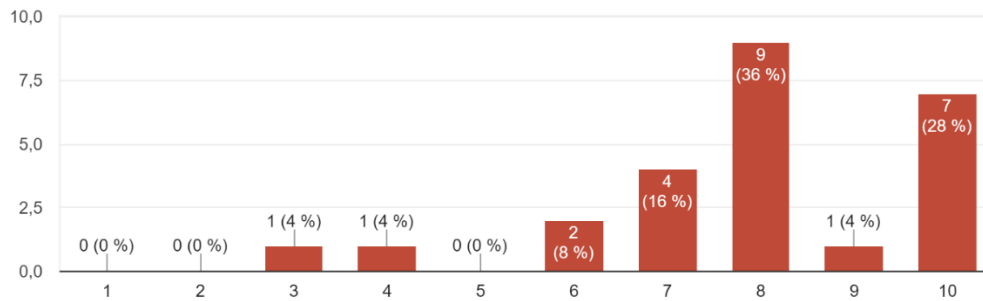


Figura 61. Resultados de experiencia visual gracias a las animaciones

¿Cree que las animaciones de los personajes lograron transmitir la narrativa de "Gaus y la revolución de los onritorrincos"?

25 respuestas

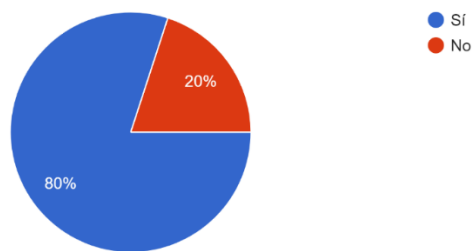


Figura 62. Resultados de la narrativa alcanzada gracias a las animaciones

¿Se sintió más conectado o empático hacia los personajes debido a sus animaciones?
25 respuestas

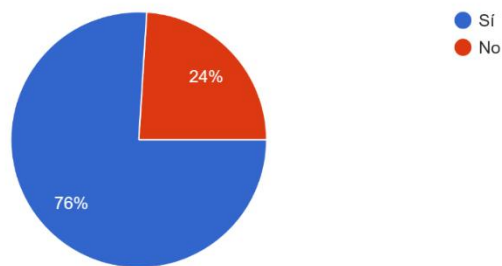


Figura 63. Resultados de conexión hacia los personajes gracias a las animaciones

En una escala del 1 al 10, ¿Considera que las animaciones fueron fluidas y se adaptaron adecuadamente al ritmo del juego?

25 respuestas

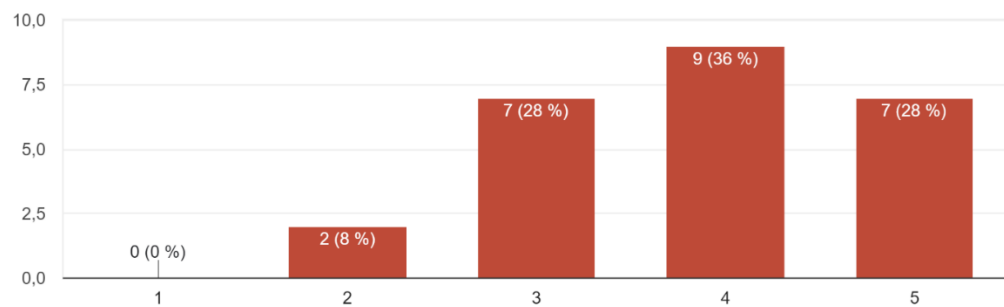


Figura 64. Resultados de fluides de las animaciones

¿Logró identificar los distintos estados de animación de los personajes, como correr, caminar, saltar, etc.?

25 respuestas

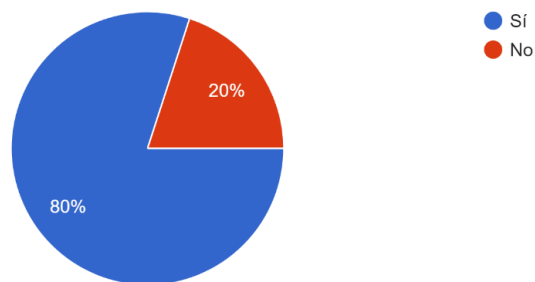


Figura 65. Resultados de identificación de animaciones

En una escala del 1 al 10, ¿qué tan satisfactorias le parecieron las transiciones entre las animaciones (por ejemplo, de caminar a correr)?

25 respuestas

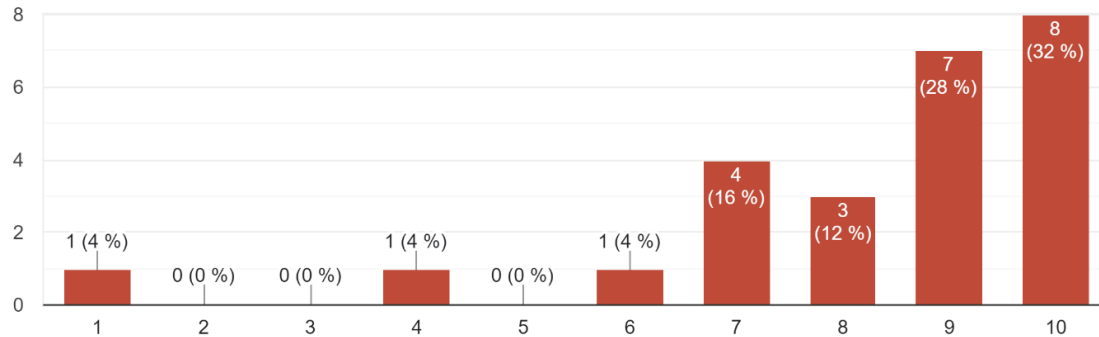


Figura 66. Resultados de satisfacción de transiciones de las animaciones

¿Notó la aplicación de los principios de animación como anticipación y encogerse, en los movimientos de los personajes?

25 respuestas

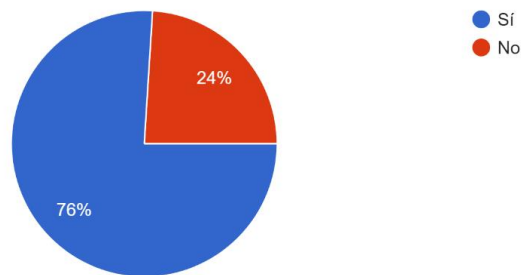


Figura 67. Resultados de aplicación de principios de animación

¿Percibió que las animaciones del juego contribuían al atractivo visual general?

25 respuestas

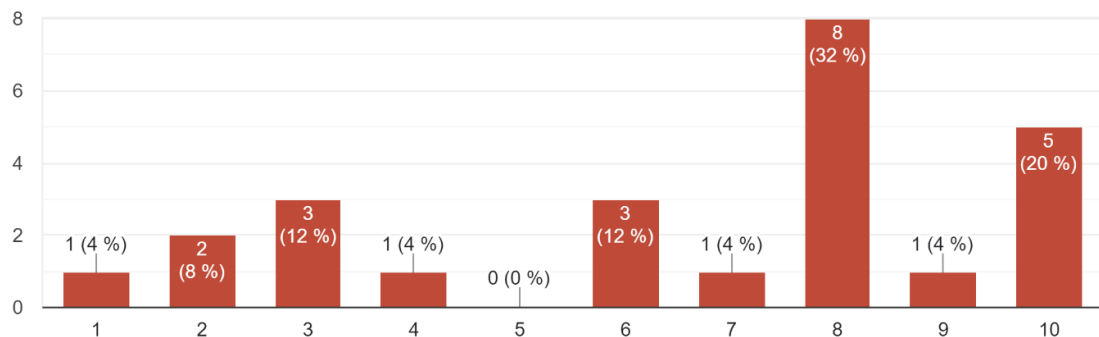


Figura 68. Resultados de atracción visual gracias a las animaciones

En una escala del 1 al 5, ¿considera que las animaciones mejoran su comprensión de la trama y la lucha entre los ornitorrincos y sus depredadores?

25 respuestas

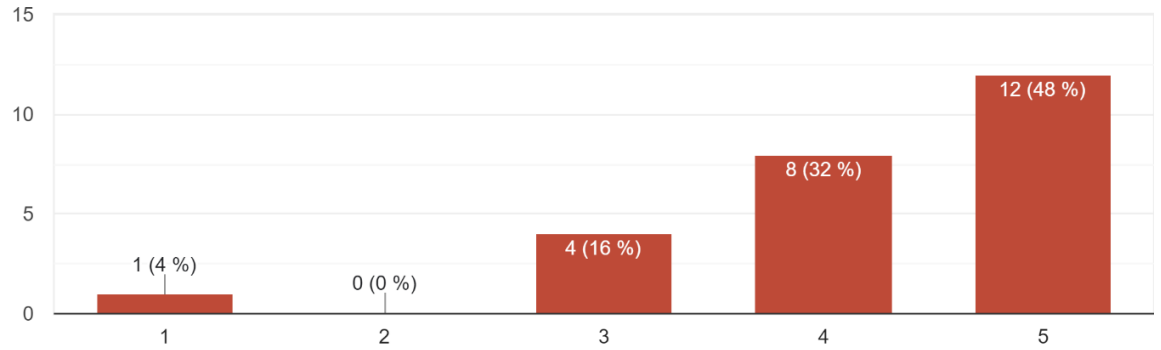


Figura 69. Resultados de comprensión de la trama gracias a las animaciones

En una escala del 1 al 5, ¿las animaciones le parecieron realistas o creíbles dentro del contexto caricaturesco y de ciencia ficción del juego?

25 respuestas

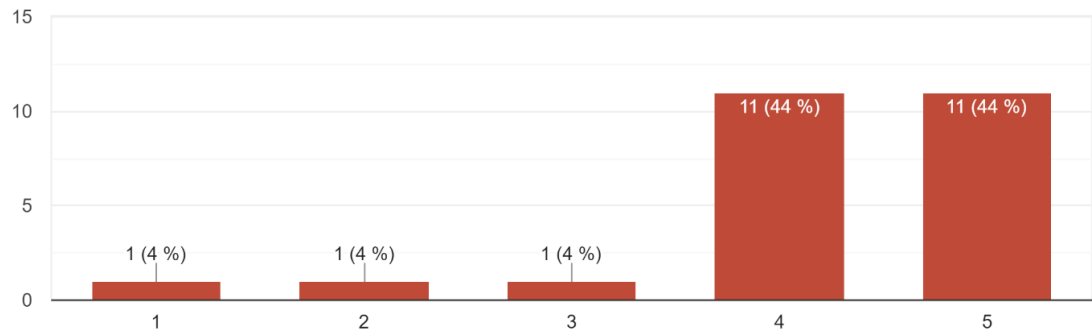


Figura 70. Resultados de credibilidad de contexto gracias a las animaciones

XI. Glosario

1. **Animation Clips:** Secuencias de animación predefinidas aplicadas a objetos o personajes en un videojuego o proyecto de animación.
2. **Animation State Machine:** Sistema dentro de un controlador de animación que permite manejar las diferentes transiciones y estados de las animaciones de un personaje u objeto.
3. **Animación:** Técnica de dar movimiento a imágenes o modelos, creando la ilusión de que están vivos o en acción.
4. **Animator Controller:** Componente en motores gráficos (como Unity) que controla las animaciones de un personaje u objeto, permitiendo transiciones entre diferentes animaciones.
5. **Bool (booleano):** Tipo de dato lógico que puede tener uno de dos valores: verdadero (true) o falso (false), utilizado para representar estados binarios.
6. **Fotograma:** Imagen individual dentro de una secuencia de imágenes en una animación o video. La secuencia de fotogramas crea el movimiento en una animación.
7. **Frames:** Imágenes consecutivas en una animación o video que, al proyectarse rápidamente, crean la ilusión de movimiento.
8. **Game Object:** Elemento básico en un entorno de desarrollo de videojuegos que representa cualquier objeto dentro del juego, ya sea visible (como un personaje) o invisible (como un punto de control).
9. **GameTesting:** Proceso de evaluación de un videojuego para identificar errores, fallos de funcionamiento y áreas de mejora antes de su lanzamiento.
10. **Get Component:** Función en programación de videojuegos que permite acceder a un componente específico dentro de un objeto del juego.
11. **IA (Inteligencia Artificial):** Simulación de procesos de inteligencia humana en sistemas informáticos, como aprendizaje, percepción y toma de decisiones.
12. **Idle:** Estado de inactividad de un personaje o elemento en un videojuego, comúnmente animado para simular espera.
13. **Keyframe:** Punto clave en una animación que define el inicio o fin de una acción o transición específica.
14. **KO (morir):** Estado de un personaje que indica que ha sido derrotado o incapacitado en un videojuego.
15. **Mecanim:** Sistema de animación en el motor Unity que permite controlar y gestionar las animaciones de personajes de manera más intuitiva y flexible.
16. **Mesh:** Red de polígonos que forma la estructura visual de un modelo 3D en gráficos por computadora.

17. **Motion Capture:** Técnica de captura de movimiento real, utilizada para animar personajes digitales con los movimientos de actores o personas reales.
18. **NPC (Non-Playable Character):** Personaje en un videojuego que no es controlado por el jugador, sino que actúa de manera programada.
19. **Rigging:** Proceso de creación de un esqueleto para un modelo 3D, lo que permite que el modelo se pueda mover y animar.
20. **Root Master:** Controlador o nodo principal de una estructura de animación o jerarquía de objetos en una escena.
21. **Script:** Código escrito para programar comportamientos y funcionalidades de los elementos en un videojuego o aplicación.
22. **Stop Motion:** Técnica de animación que crea movimiento al fotografiar objetos en posiciones ligeramente distintas en cada fotograma.
23. **Triggers:** Condiciones o eventos que desencadenan una acción o cambio de estado en un videojuego o animación.
24. **VFX (Visual Effects):** Efectos visuales generados digitalmente o en posproducción, que se utilizan para agregar elementos visuales que no son posibles o prácticos de capturar en una filmación real.
25. **Video Mapping:** Técnica de proyección en la que imágenes o animaciones se proyectan sobre superficies irregulares, como edificios, para crear efectos visuales.

XII. Bibliografía

Admin, S., & Admin, S. (2023, September 21). *¿Qué es la captura de movimiento y cuáles son sus aplicaciones?* STT Systems. <https://www.stt-systems.com/es/blog/que-es-la-captura-de-movimiento-y-cuales-son-sus-aplicaciones/>

Aminian, P. (2024, August 6). Stylized Visionaries: artistic interpretations of games. *Pixune*. <https://pixune.com/blog/stylized-art-style/>

Andreu, A. P. (2022, December 7). *Motion Capture en la industria de la Animación*. Blog CEDIM. <https://blog.cedim.edu.mx/animacion/motion-capture-en-la-industria-de-la-animacion/>

Artadmin. (2023a, November 15). *Video Game Art Styles: A Glimpse into Gaming Aesthetics*. Gamepack Studio. <https://gamepackstudio.com/video-game-art-styles-a-glimpse-into-gaming-aesthetics/>

Artadmin. (2023b, November 15). *Video Game Art Styles: A Glimpse into Gaming Aesthetics*. Gamepack Studio. <https://gamepackstudio.com/video-game-art-styles-a-glimpse-into-gaming-aesthetics/>

Auto-Rig Pro. (n.d.). Blender Market. <https://blendermarket.com/products/auto-rig-pro>

Azsan, F. (2024, August 26). Exploring CGI and 3D animation- Polydin. *Polydin*. <https://polydin.com/cgi-and-3d-animation/>

Bisht, V. (2024a, May 16). Animation in video games: creating immersive experiences. *aajjo.com*. <https://blog.aajjo.com/post/animation-in-video-games-creating-immersive-experiences>

Bisht, V. (2024b, May 16). Animation in video games: creating immersive experiences. *aajjo.com*. <https://blog.aajjo.com/post/animation-in-video-games-creating-immersive-experiences>

Bisht, V. (2024c, May 16). Animation in video games: creating immersive experiences. *aajjo.com*. <https://blog.aajjo.com/post/animation-in-video-games-creating-immersive-experiences>

Burriel, D. T. (2023, November 1). *El impacto del Motion Design en la experiencia de usuario*. Blog - UX Torresburriel Estudio. <https://torresburriel.com/weblog/el-impacto-del-motion-design-en-la-experiencia-de-usuario/#:~:text=La%20animaci%C3%B3n%20juega%20un%20papel,del%20usuario%20dentro%20del%20producto.>

Campus, C. (2024a, May 22). *¿Qué es el rigging en animación?* Universidad Europea Creative Campus. <https://creativecampus.universidadeuropea.com/blog/que-es-rigging/>

Campus, C. (2024b, September 4). *Blender 3D: qué es, usos y diferencias con Cinema 4D*. Universidad Europea Creative Campus. <https://creativecampus.universidadeuropea.com/blog/blender/#que-es-blender>

Character Design for Animation and Games - Harvard Division of Continuing Education Course Browser. (2024, October 3). Harvard Division of Continuing Education Course Browser. <https://coursebrowser.dce.harvard.edu/course/character-design-for-animation-and-games/>

Chisaba, C. a. O. (2021, February 22). Captura de movimiento: La tecnología usada en Avatar – Niixer. *Portal De Noticias De Tecnología, Realidad Virtual, Aumentada Y Mixta, Videojuegos*. <https://niixer.com/index.php/2021/02/22/captura-de-movimiento-tecnologia/>

Cooper, J. (2019, October 3). *Making of God of War: Cinematics & Rigging*. Game Anim. <https://www.gameanim.com/2018/06/20/making-of-god-of-war-cinematics-rigging/>

Cortés, J., & Cortés, J. (2024, August 17). *¿Qué es la Animación 3D? Tipos, Técnicas y Programas*. Notodoanimacion.es | Noticias, Recursos, Tutoriales Y Empleo Para Artistas Digitales. <https://www.notodoanimacion.es/que-es-la-animacion-3d-tipos-y-tecnicas/>

DeGuzman, K. (2023, September 7). *What is Mocap — The Science and Art Behind Motion Capture*. StudioBinder. <https://www.studiobinder.com/blog/what-is-mocap-definition/>

Dumitru, C. Z. (2021, June 26). *Where character animation meets human behavior sciences*. Cristina Teaching Art. <https://www.cristinateachingart.com/where-character-animation-meets-human-behavior-sciences/>

Entertainment software Association (ESA). (2023). *Essential facts: About the U.S Video Game Industry*. https://www.theesa.com/wp-content/uploads/2024/02/ESA_2023_Essential_Facts_FINAL_07092023-1.pdf

Euroinnova International Online Education. (2024, September 17). *Conoce qué es una diseñadora gráfica*. <https://www.euroinnova.com/blog/latam/caracterizacion-de-personajes#:~:text=confiar%20en%20Euroinnova%3F-,%20BFQu%20es%20la%20caracterizaci%C3%B3n%20de%20personajes%3F,apreciaci%C3%B3n%20de%20una%20obra%20teatral>

Fernando, & Fernando. (2024, September 15). *Auto-Rig Pro Descargar addon GRATIS*. Sketchitos.com. <https://sketchitos.com/auto-rig-pro-descargar-addon-gratis/>

Floriano, J. F. J., & Floriano, J. F. J. (2022, December 19). *Técnicas de animación 3D | Blog de DSIGNO*. Blog De DSIGNO. <https://www.dsigno.es/blog/disenio-grafico/tecnicas-de-animacion-3d>

Giménez, R. (2024, January 12). *Keyframe Animation: Aplicación en Producción 3D - Animum 3D*. Animum 3D. <https://www.animum3d.com/blog/keyframe-animation/>

God of War Review - IGN. (2018, April 13). IGN. <https://www.ign.com/articles/2018/04/12/god-of-war-review>

GrupoAudiovisual. (2024, July 16). *3D y Animación – Definición, usos y técnicas*. Grupo Audiovisual. <https://grupoaudiovisual.com/3d-y-animacion-definicion-usos-y-tecnicas/>

Kajala, N. (2022). Designing a Combat System for 3D Video Game. *Project: Ripple in Dimensions*. https://www.theseus.fi/bitstream/handle/10024/746542/Kajala_Niilo.pdf?sequence=2

Kyle, J. (2019). Spider-Man's fluid animations make him feel agile and responsive, embodying the essence of the character as a superhero who swings through the city with grace. *Animation and Game Design: The Case of Marvel's Spider-Man*, 12, 45–50.

Lapo, M. S., & Lapo, B. S. (2023). *Narrativa, Mecánicas de Videojuegos y Animación como Fortalezas Interactivas para Videojuegos en Móviles*. <https://www.redalyc.org/journal/6887/688775175001/>

Lasseter, J. (1987). Principles of traditional animation applied to 3D computer animation. *ACM SIGGRAPH Computer Graphics*, 21(4), 35–44. <https://doi.org/10.1145/37402.37407>

Lowe, R. K., & Schnotz, W. (2014). Animation principles in multimedia learning. In *Cambridge University Press eBooks* (pp. 513–546). <https://doi.org/10.1017/cbo9781139547369.026>

Manuel, C. M. J. (2015, August 31). *La reinterpretación de los principios clásicos de animación en los medios digitales*. <https://eprints.ucm.es/32970/>

Martínez, G. (2023, May 3). *¿Qué es Unity y para qué sirve?* Ebac. <https://ebac.mx/blog/que-es-unity-y-para-que-sirve>

MasterD. (2019a, November 8). *Qué es Unity y para qué sirve*. MasterD. <https://www.masterd.es/blog/que-es-unity-3d-tutorial>

MasterD. (2019b, November 8). *Qué es Unity y para qué sirve*. MasterD. <https://www.masterd.es/blog/que-es-unity-3d-tutorial>

Maxon - Cinema 4D - Mixamo Control Rig. (2024). Maxon. <https://www.maxon.net/es/cinema-4d/features/mixamo-control-rig>

Melo, C. (2021, November 22). *Animación Blender 3D - MixMamo*. Issuu. https://issuu.com/deridovely/docs/animaci_n_blender_3d_-_mixamo#google_vignette

Moody, A. (2009). *3D Pioneers unveil new company - Mixamo*. Animation World Network. <https://www.awn.com/news/3d-pioneers-unveil-new-company-mixamo>

Murphy, C. (2013). DESIGN BETTER GAMES! FLOW, MOTIVATION, & FUN. *Design and Development of Training Games*. http://www.goodgamesbydesign.com/Files/Chapter5_Flow_Motivation_Fun_Final_WebVersion.pdf

Oliveira, J. (2021, March 29). *¿Qué es un keyframe?, claves para editar tus mejores videos*. <https://www.crehana.com>. <https://www.crehana.com/blog/estilo-vida/que-es-un-keyframe/>

Proskahuer. (2015, September 3). *12 principios de la animación (lo que todo animador debería saber y aplicar)*. BLOG DE PROSKAHUER. <https://proskahuerpro.wordpress.com/2015/09/03/12-principios-de-la-animacion-lo-que-todo-animador-deberia-saber-y-aplicar/>

¿Qué es la animación 3D y qué tipos existen? (2023, May 26). Financial Magazine. https://financiamagazine.es/animacion/animacion-3d-tipos/#Aplicaciones_de_la_animacion_3D

Roca, D. (2021, July 9). *Animaciones para tus proyectos de Unity - Blog Escola Espai*. Blog Escola Espai. <https://www.espai.es/blog/2021/07/animaciones-unity/>

S, M. A. (2019, April 13). Los 12 principios de animación. *HyperBlog*. <https://www.tesseractspace.com/blog/los-12-principios-de-animacion/>

Sheikh, M., & Sheikh, M. (2024, September 8). *Programa de Animación : 15 Herramientas Para Principiantes*. Visme Blog. <https://visme.co/blog/es/programa-de-animacion/#blender>

Statham, N. (2023). Characterization through character design in competitive multiplayer games. *UPPSALA UNIVERSITET*. <https://uu.diva-portal.org/smash/get/diva2:1767167/FULLTEXT01.pdf>

Superior, E. F. P. (2023, August). *¿Qué es el rigging en animación 3D?* ESIC. <https://www.esic.edu/rethink/tecnologia/que-es-rigging-animacion-3d-c>

Tai. (2022, April 21). *¿Qué es la animación 3D y para qué sirve?* TAI ARTS. <https://taiarts.com/blog/que-es-animacion-3d-y-para-que-sirve/#:~:text=La%20animaci%C3%B3n%203D%20va%20mucho,est%C3%A1%20cambiando%20la%20creaci%C3%B3n%20audiovisual>

Technologies, U. (2013). *Unity - Manual: Visión general del Sistema de Animación*. <https://docs.unity3d.com/es/530/Manual/AnimationOverview.html>

Technologies, U. (2015). *Unity - Manual: Animator component (Componente animator)*. <https://docs.unity3d.com/es/530/Manual/class-Animator.html#:~:text=El%20componente%20Animator%20es%20utilizado,hacer%20una%20transici%C3%B3n%20entre%20estos>.

Technologies, U. (2016a). *Unity - Manual: Integración de Animación*. <https://docs.unity3d.com/es/530/Manual/UIAnimationIntegration.html>

Technologies, U. (2016b). *Unity - Manual: Visión general del Sistema de Animación*. <https://docs.unity3d.com/es/530/Manual/AnimationOverview.html#:~:text=Mecanim%20proporciona%3A,animaciones%20creadas%20dentro%20de%20Unity>

The Legend of Zelda: Breath of the Wild Review - IGN. (2024, June 27). IGN. <https://www.ign.com/articles/2017/03/02/the-legend-of-zelda-breath-of-the-wild-review>

The National Centre for Computer Animation, & Anderson, E. F. (2021). *Bournemouth University Research Online [BURO] - Real-Time character animation for computer games*. <https://eprints.bournemouth.ac.uk/10263/>

Tipos y técnicas de animación 3D. (n.d.). DesarrolloWeb.com. <https://desarrolloweb.com/articulos/tipos-tecnicas-animacion-3d>

Tomb Raider Review - IGN. (2018, December 13). IGN. <https://www.ign.com/articles/2003/10/04/tomb-raider-review>

Torres, A. (2024a, August 5). *¿Qué es la animación 3D y qué tipos existen?* ESDESIGN. <https://www.esdesignbarcelona.com/actualidad/animacion/que-es-la-animacion-3d-y-que-tipos-existen>

Torres, A. (2024b, September 2). *¿Qué es el Motion Design y para qué sirve?* ESDESIGN. <https://www.esdesignbarcelona.com/actualidad/motion-design/que-es-el-motion-design-y-para-que-sirve>

Triviño, M. (2016, May 3). *Sobre la caracterización de los personajes y su influencia en el jugador*. <https://deusexmachina.es/la-caracterizacion-los-personajes-influencia-jugador/>

Unity. (2024). *Cómo animar personajes con el rigging de animación | Unity*. <https://unity.com/es/topics/rigging-in-animation>

Vista de Modelado y Animación 3D. (n.d.). <https://revistas.utp.ac.pa/index.php/el-tecnologico/article/view/1286/1724>

Vittar, J. R. (2023, September 9). *Animación con Blender: El Arte en Movimiento*. Fanáticos Del 3D. <https://fanaticosdel3d.com/animacion-con-blender/>

Wirtz, B. (2023a, July 4). Game Design with Impact: The Psychology of Video Game Immersion (Bonus: List of the Most Immersive Games). *Video Game Design and Development*. <https://www.gamedesigning.org/learn/game-immersion/>

Wirtz, B. (2023b, July 4). Game Design with Impact: The Psychology of Video Game Immersion (Bonus: List of the Most Immersive Games). *Video Game Design and Development*. <https://www.gamedesigning.org/learn/game-immersion/>

Yuquilema, M. P., Vita, D. F., & López, M. Á. (2022). *Técnicas y principios de animación: Caso de estudio serie Fantasmagorias, La Llorona*. Dialnet. <https://dialnet.unirioja.es/servlet/articulo?codigo=8635282>