



Animación3D

Memoria práctica final

Luis Javier Moreno Rojas



Índice de contenidos

1. Introducción	3
2. Modelado	3
2.1. Personaje.....	3
2.1.1. Concepto, referencias, boceto y turn around	3
2.1.2. Proceso de modelado del personaje.....	6
2.1.3. Elementos del personaje.....	6
2.1.4. Mapeado del personaje	7
2.1.5. Modelado final del personaje	7
2.2. Escenario	8
2.2.1. Concepto y referencias.....	8
2.2.2. Elementos del escenario	9
2.2.3. Modelado final del escenario.....	10
3. Texturizado.....	10
4. Iluminación y cámaras.....	11
4.1. Iluminación.....	11
4.2. Cámaras.....	12
5. Renders	12
6. Animación.....	15
6.1. Animaciones del escenario.....	15
6.1.1. Submarino	15
6.1.2. Patos.....	15
6.1.3. Puerta con controlador	15
6.1.4. Árbol.....	16
6.2. Animaciones del personaje	16
6.2.1. Proceso de preparación animación personaje.....	16
6.2.2. Animación inversa y directa (IK-FK).....	17
6.2.3. Uso de mocaps	18
6.2.4. Animaciones paramétricas.....	18
6.2.5. Animación directa	19
7. Exportación a Unity	19
7.1. Exportación del escenario	19
7.2. Exportación del personaje.....	20
7.3. Importación a Unity.....	21
7.4. Animación con físicas	22
7.5. Interactividad con el sistema	22

8. Bibliografía	24
-----------------------	----

Índice de figuras

Ilustración 1: Primera referencia del personaje	4
Ilustración 2: Segunda referencia del personaje	4
Ilustración 3: Boceto con ficha de personaje	5
Ilustración 4: Turn Around del personaje	5
Ilustración 5: Proceso de modelado del personaje	6
Ilustración 6: Coleta del personaje	6
Ilustración 7: Mapeado del personaje	7
Ilustración 8: Modelado final del personaje	7
Ilustración 9: Primera referencia del escenario (Historias corrientes)	8
Ilustración 10: Segunda referencia del escenario	8
Ilustración 11: Farola	9
Ilustración 12: Vallas	9
Ilustración 13: Entrada	10
Ilustración 14: Modelado final del escenario	10
Ilustración 15: Texturas usadas en el editor de materiales	11
Ilustración 16: Esquema de iluminación	11
Ilustración 17: Esquema de cámaras	12
Ilustración 18: Primer render	12
Ilustración 19: Segundo render	13
Ilustración 20: Tercer render	13
Ilustración 21: Cuarto render	14
Ilustración 22: Quinto render	14
Ilustración 23: Animación submarino	15
Ilustración 24: Animación patos	15
Ilustración 25: Animación controlador	16
Ilustración 26: Animación Árbol	16
Ilustración 27: CUT del personaje	17
Ilustración 28: Proceso animación técnica	17
Ilustración 29: Animación inversa	18
Ilustración 30: Ejemplo mocap personaje	18
Ilustración 31: Ejemplo animación paramétrica	19
Ilustración 32: Animación directa	19
Ilustración 33: Ejemplo de exportación	20
Ilustración 34: Proceso de exportación animaciones	20
Ilustración 35: Importación en Unity	21
Ilustración 36: Diagrama de animaciones con variables	21
Ilustración 37: Tela programada con físicas	22
Ilustración 38: Ejemplo script comportamiento	23
Ilustración 39: Controles	23

1. Introducción

En el apasionante mundo de la animación y el modelado 3D, la capacidad de dar vida a escenarios y personajes propios se ha convertido en un desafío creativo y técnico de suma relevancia. Este trabajo se sumerge en la exploración y ejecución de la tarea de modelar y animar un escenario y personaje desde cero, con el objetivo de no solo satisfacer las demandas estéticas, sino también de desentrañar los intrincados aspectos técnicos que subyacen en la creación de mundos digitales inmersivos.

A lo largo de estas páginas, se abordarán los diversos procesos involucrados en la concepción, diseño, modelado y animación, destacando las decisiones críticas tomadas para alcanzar un resultado final coherente y cautivador. Desde la conceptualización inicial hasta la implementación práctica, este trabajo busca analizar los desafíos inherentes a la creación de un escenario y personaje propios con ayuda de diversas herramientas gráficas y de desarrollo como Photoshop, 3Ds Max o Unity, entre otras.

Por decisión propia del desarrollador, se ha decidido no reutilizar prácticas anteriores con la finalidad de presentar un mejor proyecto final con un gran cambio de estética.

La esencia de este proyecto radica en la creación de un personaje femenino con una estética cartoon, acompañado de un escenario colorido y vivo ambientado en un gran parque.

En cuanto al personaje, se hará uso de una paleta de colores cálidos y pastel, y para el escenario, la paleta de colores se diversificará para captar la amplitud de la naturaleza.

Este enfoque en la estética cartoon y el entorno colorido busca no solo entretener, sino también transmitir una sensación de alegría y positividad, creando así una experiencia visualmente atractiva y emocionalmente impactante.

2. Modelado

2.1. Personaje

2.1.1. Concepto, referencias, boceto y turn around

El personaje está basado en un estereotipo típico del cartoon: aquellos personajes que son energéticos, buenos, positivos y que poseen un magnetismo innato. Un personaje con una actitud contagiosa que ilumina el entorno que rodea.

Para ello, se ha contado con una gran cantidad de referentes. A continuación, se muestran algunos de los referentes usados.



Ilustración 1: Primera referencia del personaje

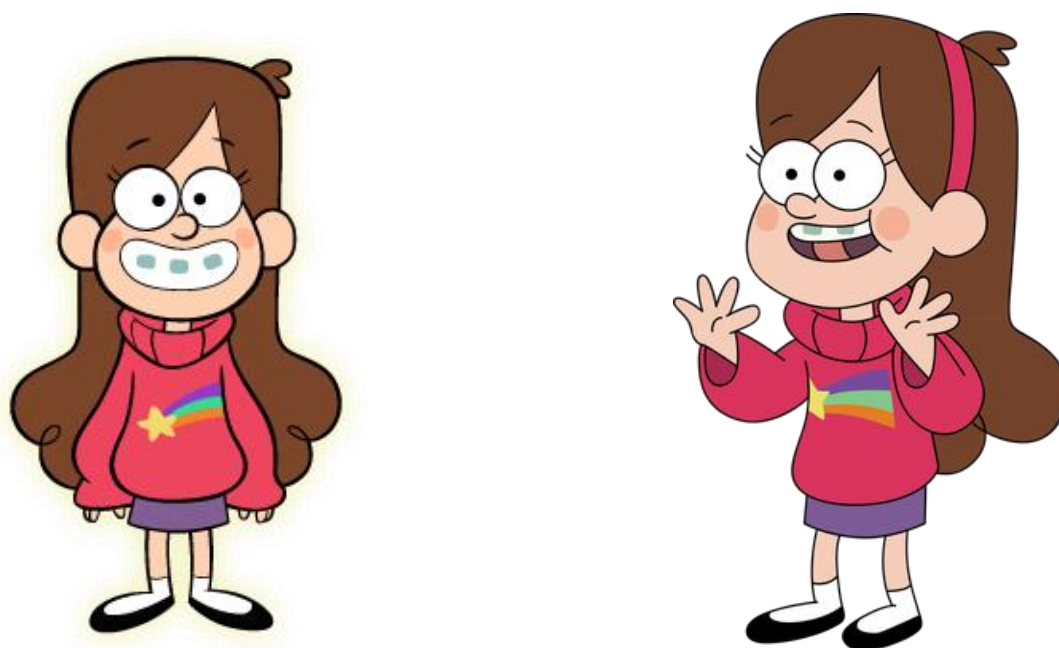


Ilustración 2: Segunda referencia del personaje

El boceto resultante, con su respectiva ficha de personaje:

Nombre: Catalina

Edad: 17

Personalidad: Risueña, decidida, magnética

Físico: Chica joven y de mediana edad.

Gafas, pelo largo y coleta larga.

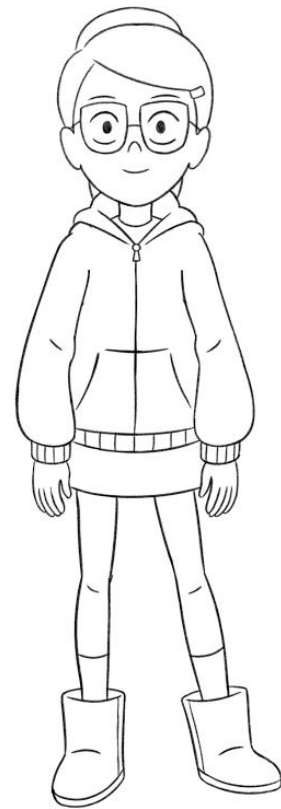


Ilustración 3: Boceto con ficha de personaje

Y, a continuación, se muestra el turna round del personaje:

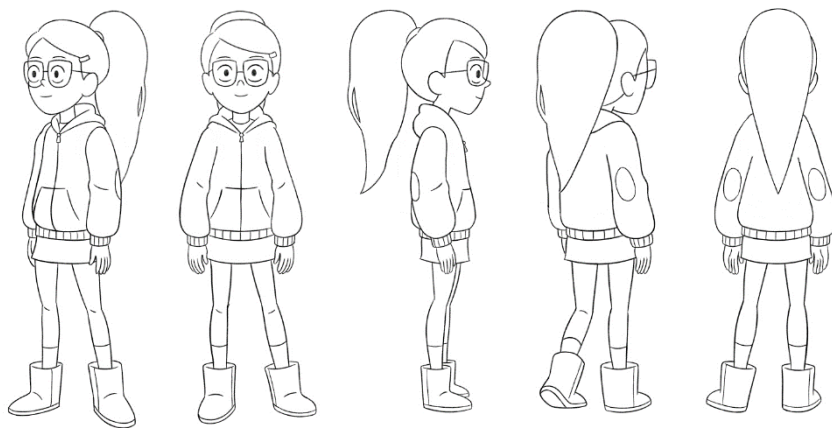


Ilustración 4: Turn Around del personaje

2.1.2. Proceso de modelado del personaje

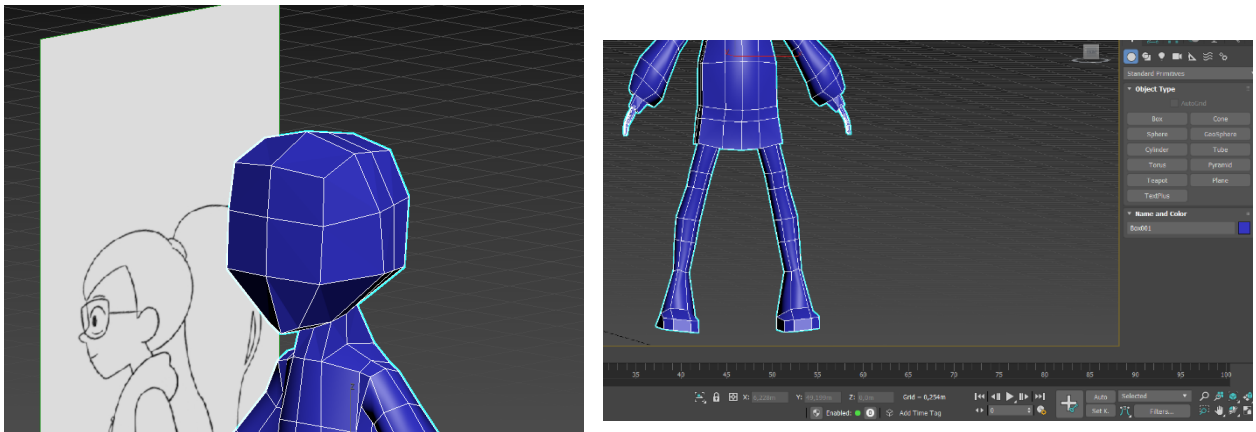


Ilustración 5: Proceso de modelado del personaje

2.1.3. Elementos del personaje

Como principal elemento del personaje, se ha creado una coleta por separado que se podrá usar como objeto a animar.

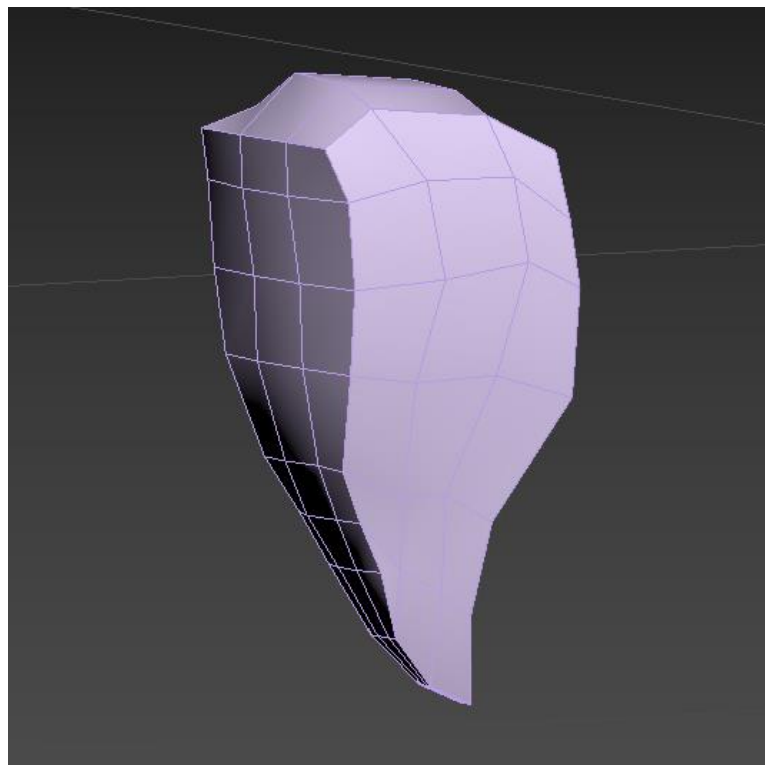


Ilustración 6: Coleta del personaje

2.1.4. Mapeado del personaje

A continuación, se muestra el mapeado del personaje obtenido mediante el modificador unwrap uvw y modificado con costuras propias.

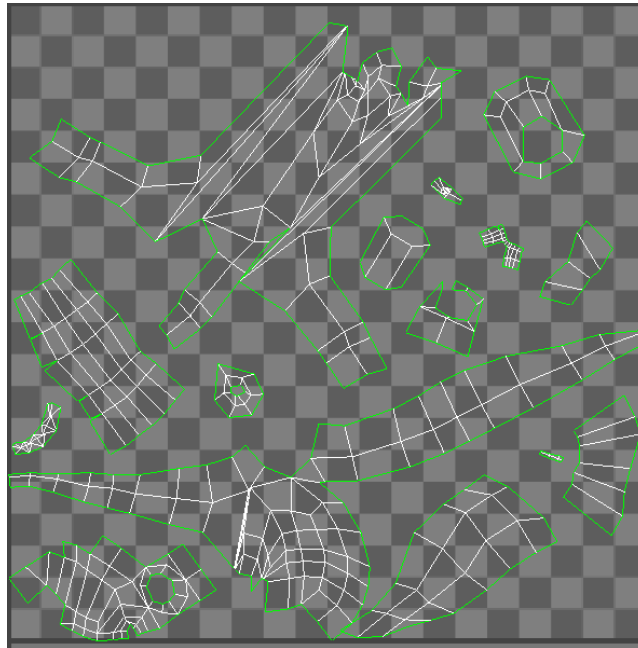


Ilustración 7: Mapeado del personaje

2.1.5. Modelado final del personaje



Ilustración 8: Modelado final del personaje

2.2. Escenario

2.2.1. Concepto y referencias

Siguiendo con la temática cartoon, se ha querido hacer un escenario inspirado en un gran parque de día, con su característica vegetación con colores vivos, sus atracciones para niños y su gran entrada.

Para ello, a continuación, se muestran alguna de las referencias encontradas.



Ilustración 9: Primera referencia del escenario (Historias corrientes)



Ilustración 10: Segunda referencia del escenario

2.2.2.Elementos del escenario

Estos son algunos de los elementos que se han usado para realizar el escenario.



Ilustración 11: Farola

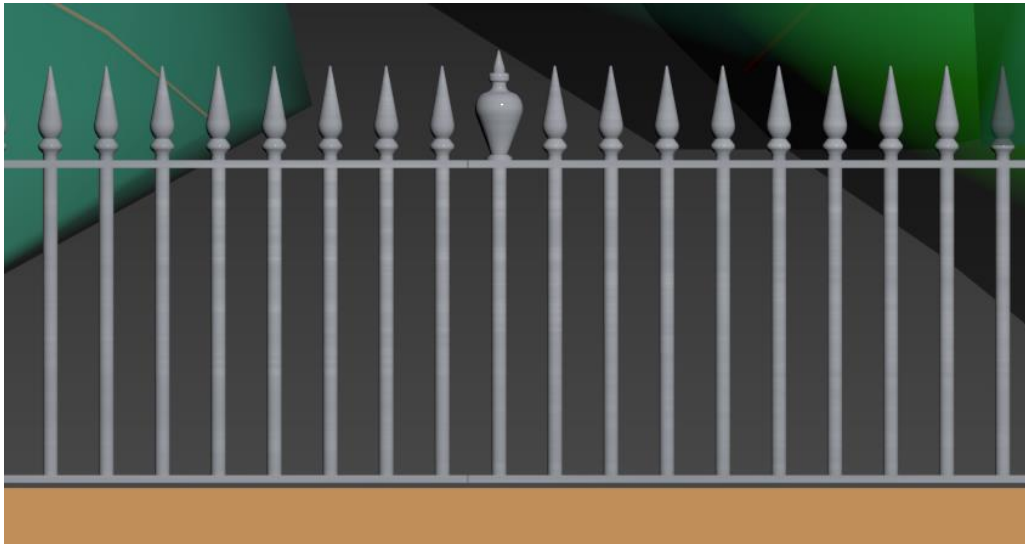


Ilustración 12: Vallas



Ilustración 13: Entrada

2.2.3. Modelado final del escenario

A continuación, se muestra el modelado final del escenario:

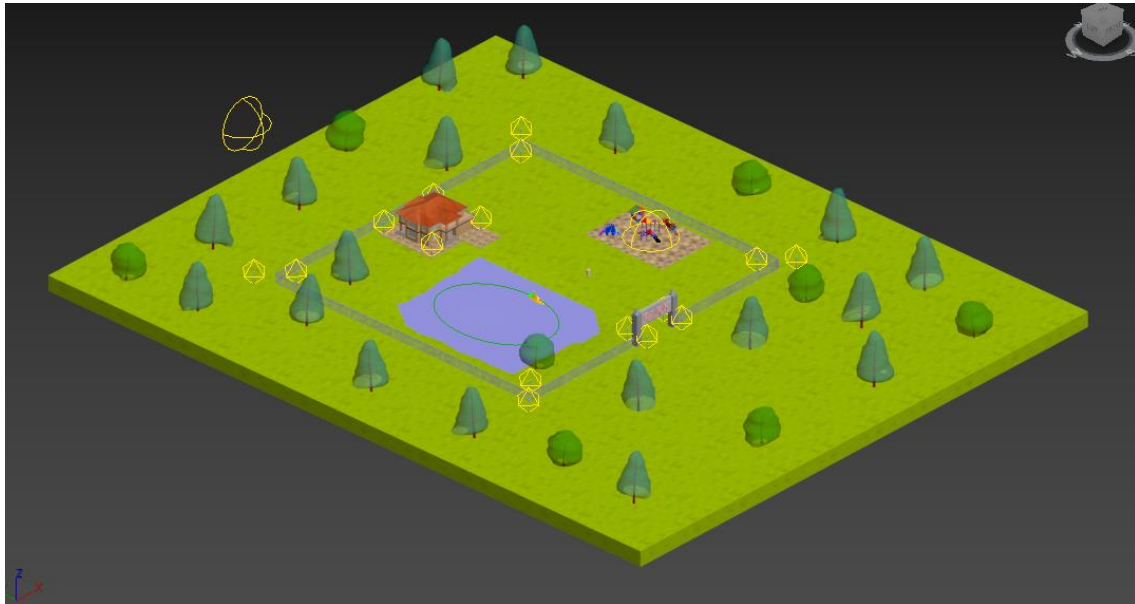


Ilustración 14: Modelado final del escenario

3. Texturizado

Para la fase de texturizado se han usado un total de 12 texturas previamente fabricadas para simular una estética similar a la usada en dibujos animados. A continuación, se muestra una imagen del editor de materiales:

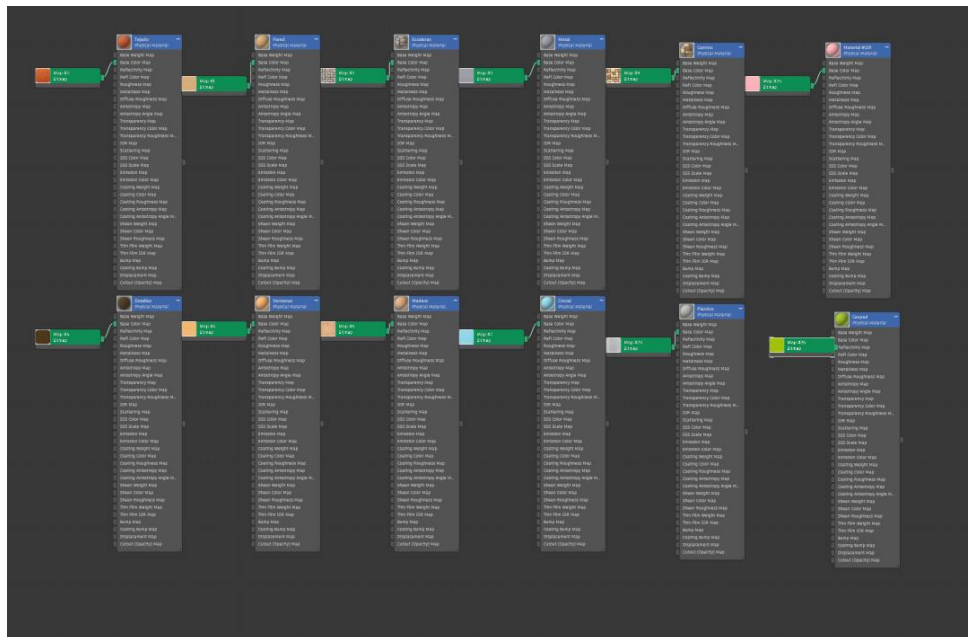


Ilustración 15: Texturas usadas en el editor de materiales

4. Iluminación y cámaras

4.1. Iluminación

Para la iluminación de la escena se han usado dos tipos de luces. En primer lugar, se ha usado una luz de tipo 'Skylight' para simular la luz solar, que ilumina con luz blanca desde el cielo y, en segundo lugar, se han usado dieciséis luces de tipo omni para las farolas que hay colocadas en la escena.

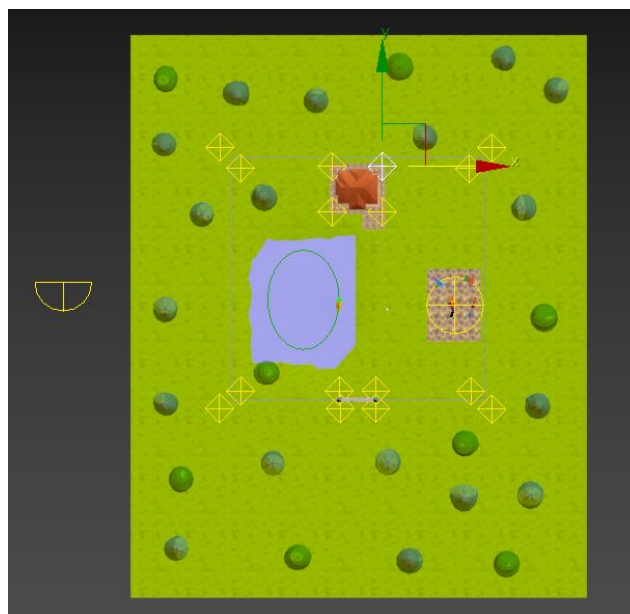


Ilustración 16: Esquema de iluminación



Ilustración 19: Segundo render

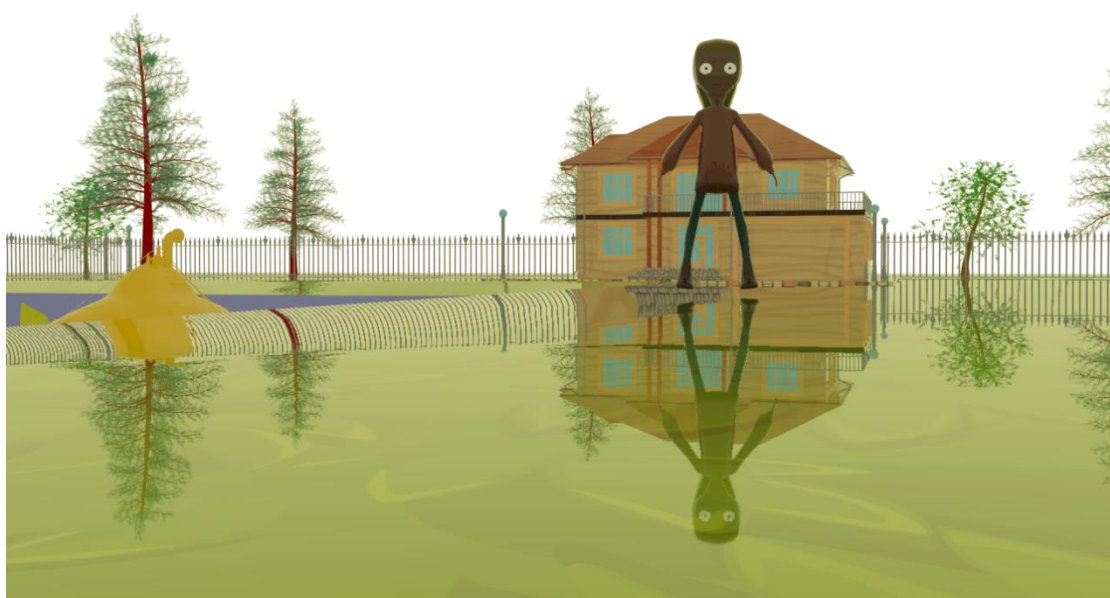


Ilustración 20: Tercer render



Ilustración 21: Cuarto render



Ilustración 22: Quinto render

6. Animación

6.1. Animaciones del escenario

6.1.1. Submarino

Se ha animado el movimiento de las hélices de un submarino mediante curvas de animación y se ha generado un recorrido circular que el submarino sigue (con un helper) mediante paths constraints. Además, el agua también se mueve.

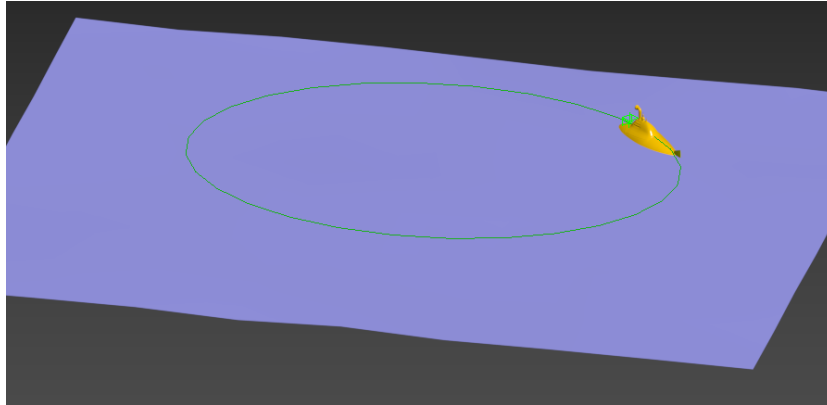


Ilustración 23: Animación submarino

6.1.2. Patos

También se han creado varios patitos de goma mediante el uso de comportamientos y manadas (Crowd y delegados) y se ha diseñado su recorrido mediante paths constraints. El comportamiento elegido para los patos es el de path follow.

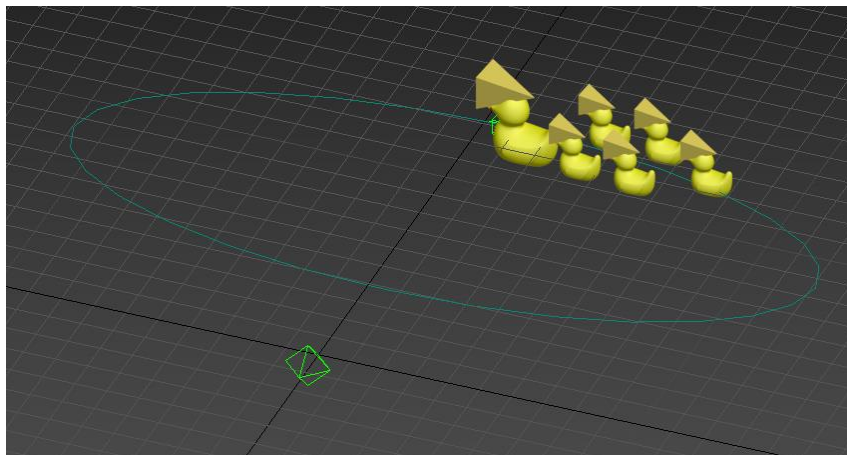


Ilustración 24: Animación patos

6.1.3. Puerta con controlador

Para la creación de la puerta del parque se ha usado un controlador que controla la abertura de la puerta.

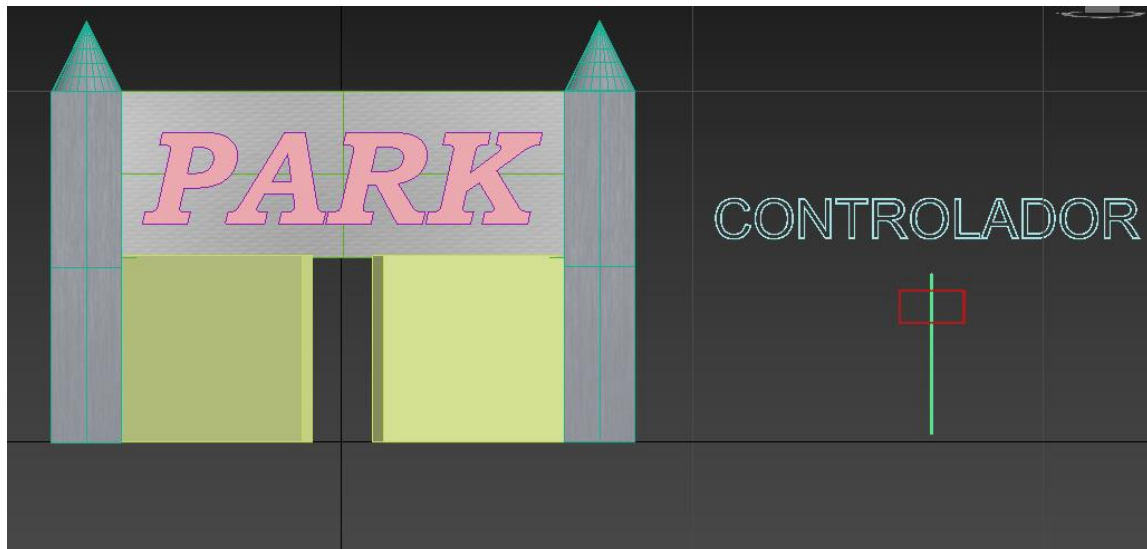


Ilustración 25: Animación controlador

6.1.4.Árbol

Se ha animado el movimiento de un árbol movido por el viento usando el modificador wave. Además, se ha usado un sistema de partículas para simular una caída de hojas movidas por el viento (fuerzas).

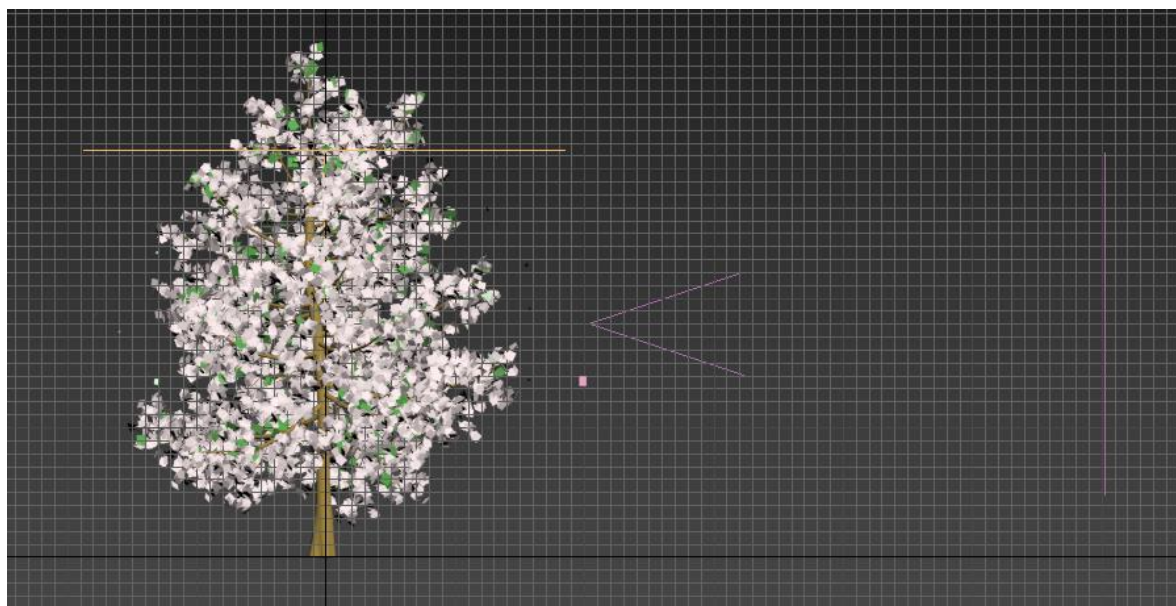


Ilustración 26: Animación Árbol

6.2. Animaciones del personaje

6.2.1.Proceso de preparación animación personaje

Para comenzar la animación del personaje, en primer lugar, se comprobaron que no hubiera fallos de modelado o vértices sobrantes. Finalmente, comenzar con el CUT, usando el modelo como base.

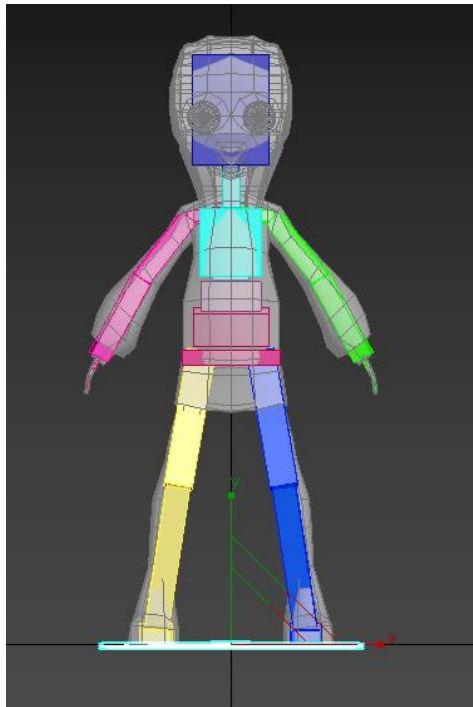


Ilustración 27: CUT del personaje

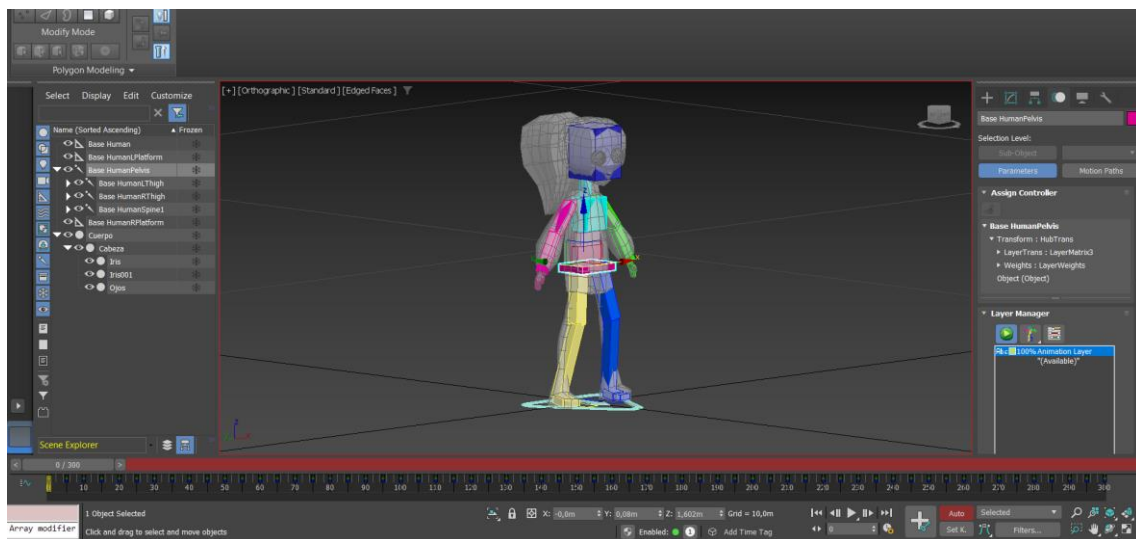


Ilustración 28: Proceso animación técnica

6.2.2. Animación inversa y directa (IK-FK)

Se ha implementado una animación usando la técnica de animación inversa, en la que el personaje desenvainará una espada y amenazará a sus oponentes para que carguen contra ella.

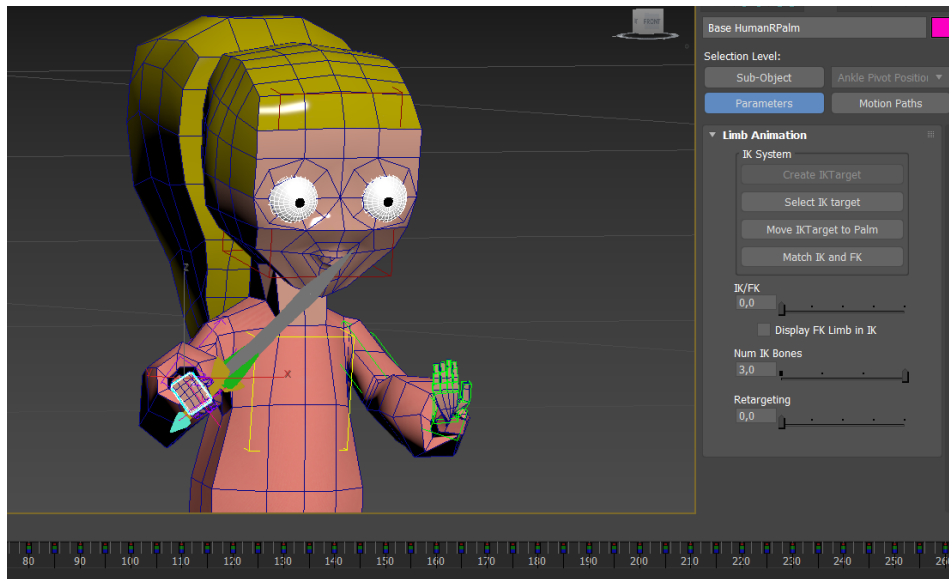


Ilustración 29: Animación inversa

6.2.3. Uso de mocaps

Se han implementado tres animaciones diferentes usando Mocaps. Una en la que el personaje esquiva varios golpes, otra en la que recibe uno y una en la que el personaje estornuda. Para ello, se han modificado fotograma a fotograma cada mocap para evitar resultados indeseados.

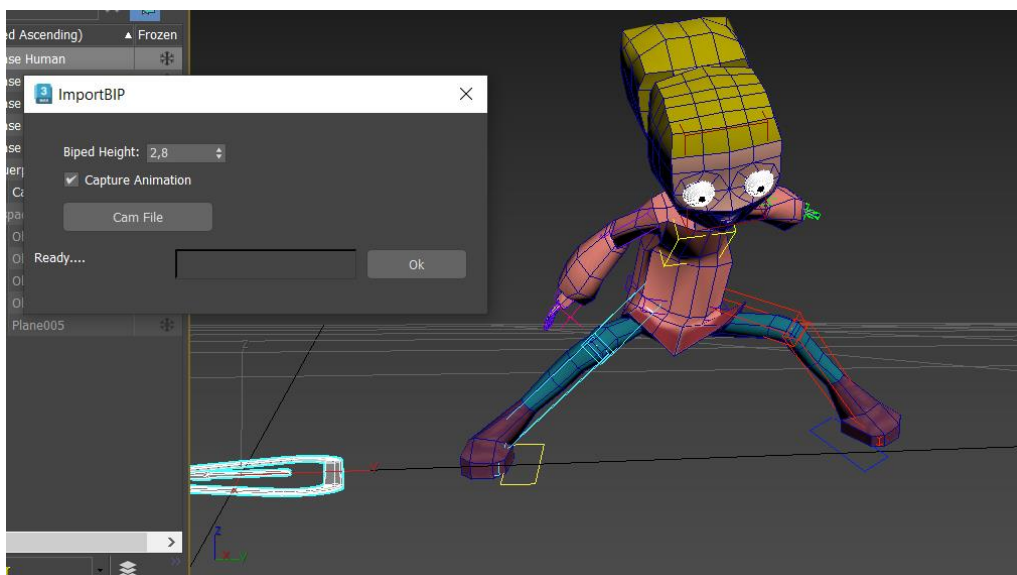


Ilustración 30: Ejemplo mocap personaje

6.2.4. Animaciones paramétricas

Se ha hecho uso de animaciones paramétricas para realizar las animaciones de andar y correr el personaje. Partiendo de las animaciones de correr y andar por

defecto, se han modificado valores (bamboleo de cadera o pecho, separación de pies...) para obtener un resultado buscado.

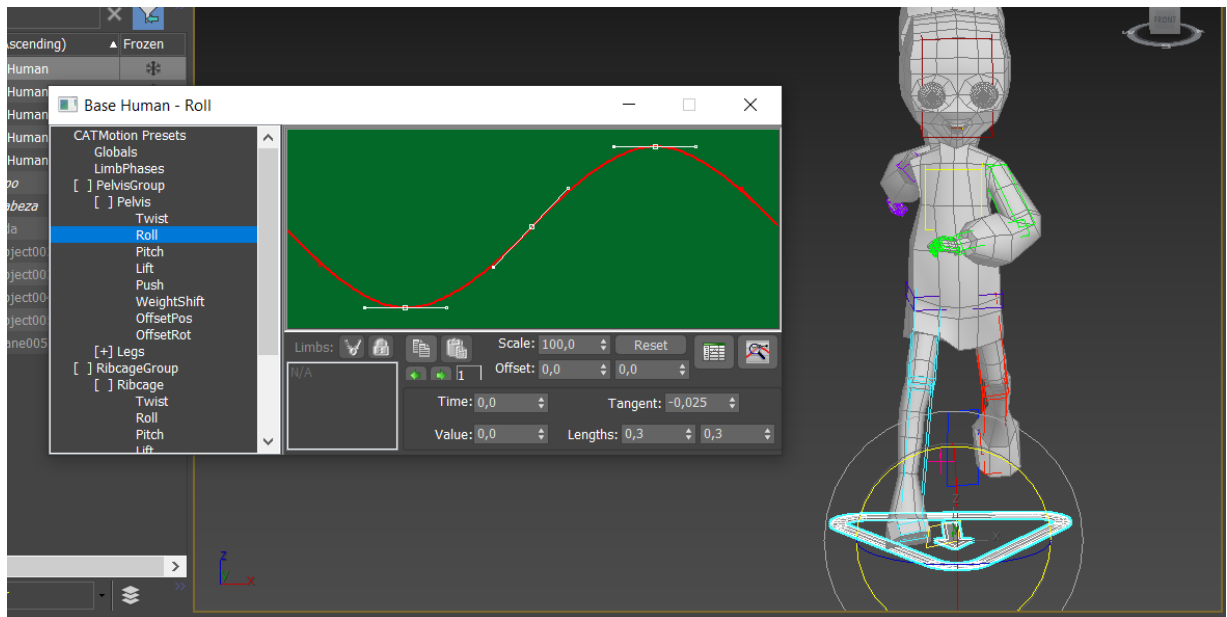


Ilustración 31: Ejemplo animación paramétrica

6.2.5. Animación directa

Se ha hecho uso de una animación directa (frame a frame) para la creación de una animación idle del personaje.

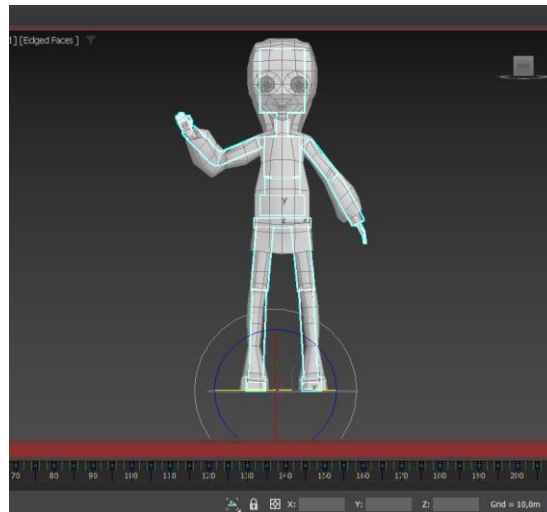


Ilustración 32: Animación directa

7. Exportación a Unity

7.1. Exportación del escenario

Para la exportación a Unity, se han usado varias técnicas diferentes en función de lo que se quería exportar. Por ejemplo, para exportar el árbol con el sistema

de partículas se han bakeado todas las partículas de la animación y se ha introducido todo dentro de un dummy auxiliar para exportarlo todo junto como un fbx. Para exportar la manada de patos, se ha usado un modificador morpher para los patos y luego se ha exportado a fbx.

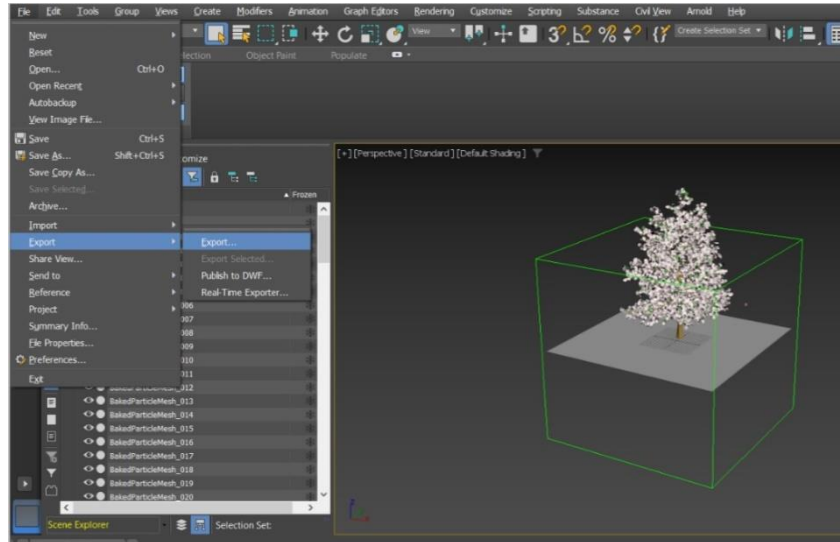


Ilustración 33: Ejemplo de exportación

7.2. Exportación del personaje

Para la exportación de las animaciones del personaje, se han creado varios fbx en los que se encuentran cada animación por separado. De esta forma, se podrán usar las diferentes animaciones en Unity.

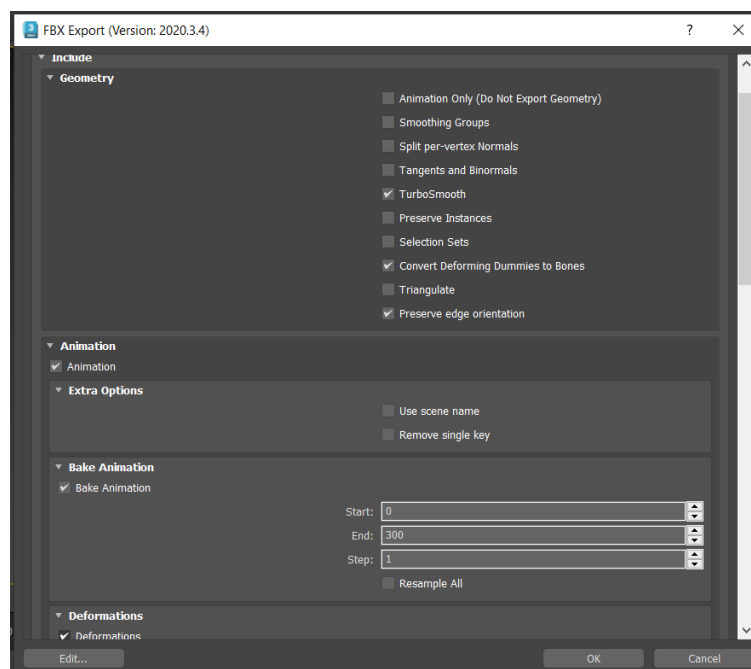


Ilustración 34: Proceso de exportación animaciones

7.3. Importación a Unity

Una vez exportados todos los objetos necesarios como fbx. Lo primero, será crear un proyecto de Unity para nuestro juego y pegar todos nuestros fbx como assets. Para el personaje, será necesario crear un controlador de animaciones y un script de comportamiento.

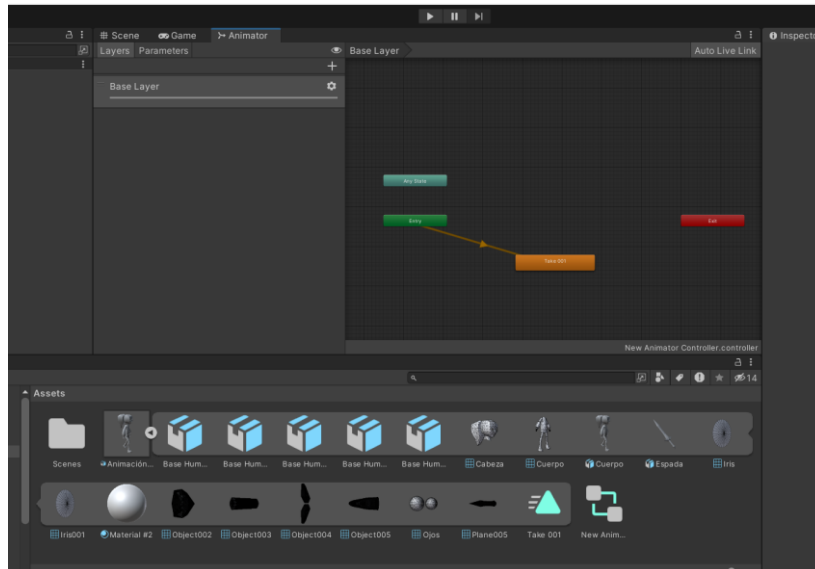


Ilustración 35: Importación en Unity

Con todas las animaciones del personaje exportadas, se ha usado el controlador de animaciones para hacer un diagrama de estados en el que se decide que animaciones pueden saltar entre ellas.

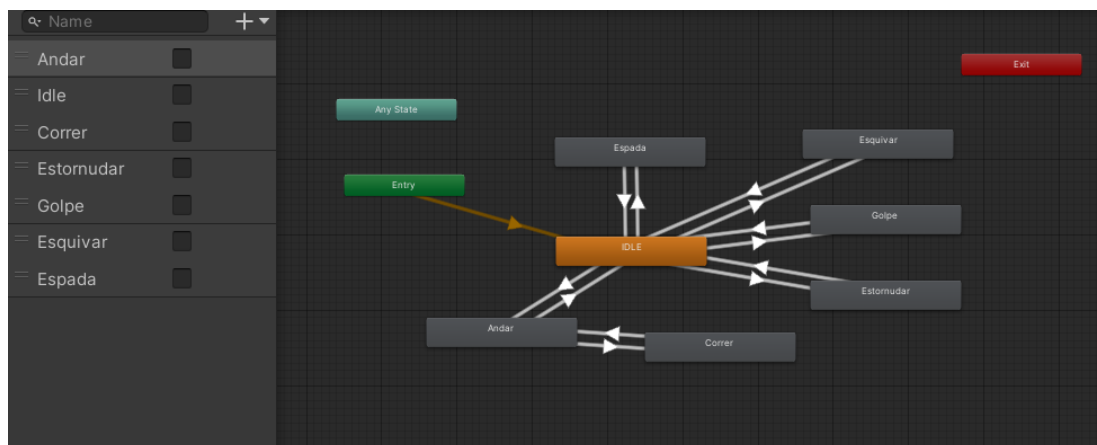


Ilustración 36: Diagrama de animaciones con variables

7.4. Animación con físicas

Se ha hecho uso de las técnicas de animaciones con físicas para añadir una animación con físicas a la escena. En el escenario, se podrá observar una bandera movida por el viento situada justo encima de la casa.

Para la creación de esta animación se han usado 5 scripts en los que convertimos un plano normal en una tela simulada con ayuda del calculo de nodos y muelles de este.

El objeto tela lleva consigo un script 'MassSpringCloth' cuyas características más importantes son el viento con su intensidad y vector de dirección, la rigidez de la malla o el amortiguamiento de esta.

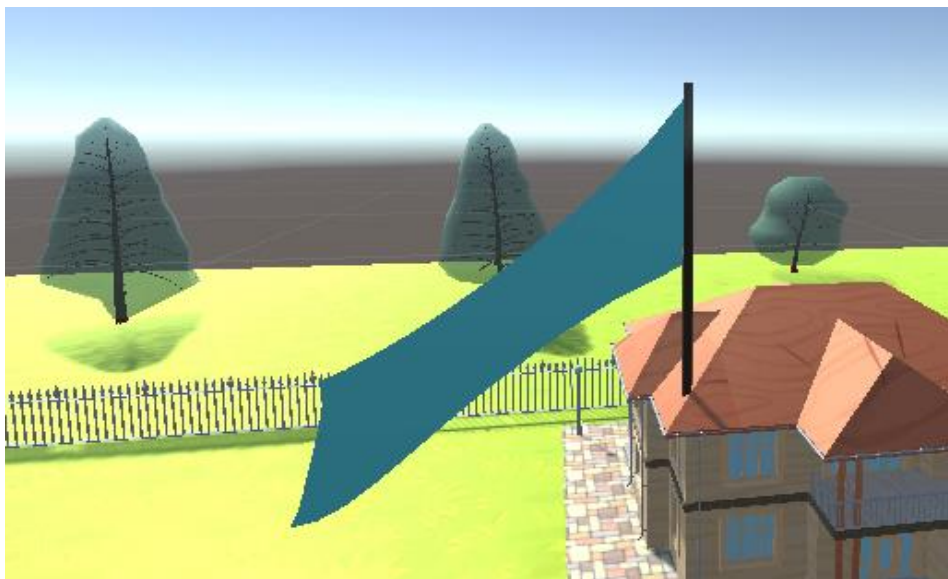


Ilustración 37: Tela programada con físicas

7.5. Interactividad con el sistema

Una vez han sido exportadas y añadidas a la escena todas las animaciones, se han creado varios scripts para dar interactividad a la escena y, en concreto, al personaje.

En primer lugar, el script 'Player' es el encargado de permitir a nuestro personaje moverse por la escena. Para ello, el personaje cuenta con una velocidad de movimiento y rotación. Para hacer esto es importante aplicar un capsule collider a nuestro personaje y un box collider a el suelo de nuestro escenario o si no el personaje se caerá de la escena.

Para permitir pasar de unas animaciones a otras ha sido necesario usar el animator controller del personaje junto con sus animaciones. Se ha creado un script de comportamiento por cada animación y se ha programado su contenido en función del diagrama de estados con ayuda de variables booleanas. Algunas de las cosas más interesantes han sido ocultar el objeto espada en animaciones que no requerían su uso, aumentar la velocidad de movimiento del personaje cuando este corre o encontrar las teclas concretas para cada animación.

```

// Script de Unity | 0 referencias
public class RunBehaviour : StateMachineBehaviour
{
    GameObject player;
    // OnStateEnter is called when a transition starts and the state machine starts to evaluate this state
    // Mensaje de Unity | 0 referencias
    override public void OnStateEnter(Animator animator, AnimatorStateInfo stateInfo, int layerIndex)
    {
        player = GameObject.FindGameObjectWithTag("Player");
        player.GetComponent<Player>().espada.SetActive(false);
        player.GetComponent<Player>().velocidadMov = 10;
    }

    // OnStateUpdate is called on each Update frame between OnStateEnter and OnStateExit callbacks
    // Mensaje de Unity | 0 referencias
    override public void OnStateUpdate(Animator animator, AnimatorStateInfo stateInfo, int layerIndex)
    {
        if (Input.GetKeyUp(KeyCode.LeftShift))
        {
            animator.SetBool("Andar", true);
            animator.SetBool("Correr", false);
            player.GetComponent<Player>().velocidadMov = 5;
        }
    }
}

```

Ilustración 38: Ejemplo script comportamiento

Por último, se deja un breve esquema mostrando los controles para mover al personaje (importante mantener pulsadas las teclas necesarias durante cada animación o se cortarán de forma abrupta para volver a su estado anterior).

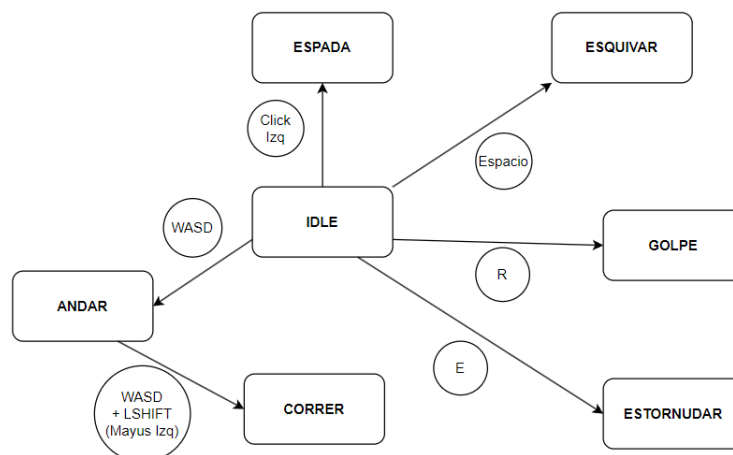


Ilustración 39: Controles

8. Bibliografía

https://regularshow.fandom.com/es/wiki/Historias_Corrientes

<https://www.aulavirtual.urjc.es/moodle/course/view.php?id=205564§ion=4#tabs-tree-start>

<https://www.youtube.com/watch?v=Thvc696ipYs&t=44s>

<https://youtu.be/LewXWM7HDd8>