



MEMORIA DE PRÁCTICA EXTERNA  
ETS DE INGENIERÍA Y SISTEMAS DE  
TELECOMUNICACIÓN  
UPM

Beatriz Ramos García

## 1. INTRODUCCIÓN:

Las prácticas se desarrollan en el Centro de Investigación en Tecnologías Software y Sistemas Multimedia para la Sostenibilidad (CITSEM), situado en el edificio La Arboleda de la ETSIST (Campus Sur). El CITSEM se creó en el año 2011 para promover la investigación, el desarrollo y la innovación (I+D+i). El centro tiene tres grupos de investigación: Grupo de Diseño Electrónico y Microelectrónico (GDEM), Grupo de Tecnología Software y Sistemas (SYST) y Grupo de Aplicaciones Multimedia y Acústica (GAMMA) donde tiene lugar el desarrollo de estas prácticas.

El grupo de investigación GAMMA nació en diciembre de 2015 como un grupo oficial reconocido por la Universidad Politécnica de Madrid (UPM), con sede en la Escuela de Ingeniería y Sistemas de Telecomunicación. En febrero de 2016, el grupo entró a formar parte del CITSEM. Tiene una composición interdisciplinar, contando con investigadores especializados en ámbitos diversos como el procesado de señal, la acústica y la gestión de empresas y la lingüística aplicada a la ciencia y la tecnología. Actualmente sus líneas de investigación se centran en: Gamificación (diseño y desarrollo de juegos serios para su aplicación en entornos educativos), Interfaces naturales inteligentes para personas con discapacidad física, Procesado de imagen médica, Acústica Virtual, Psicoacústica, Procesado de la señal de voz, Gestión del conocimiento, Aprendizaje organizativo y, por último, Innovación en el proceso de enseñanza-aprendizaje de inglés.

Las prácticas en cuestión se encuentran dentro de los campos de investigación de Gamificación e Interfaces naturales inteligentes para personas con discapacidad física. Se basan en la actualización y desarrollo del videojuego terapéutico Phiby's Adventures diseñado con Unity y Blender y utilizando el dispositivo Kinect Xbox One.

Phiby's Adventures es un videojuego diseñado para que personas con discapacidad o debilidad muscular tengan una rehabilitación interactiva a través de las aventuras del personaje principal (Phiby). El juego tiene lugar en una isla donde el protagonista va superando diferentes obstáculos con las indicaciones de personajes secundarios. Los movimientos por el entorno se realizan mediante la Kinect y tienen lugar diferentes ejercicios físicos en forma de minijuegos.

A partir de la última versión disponible del juego se han desarrollado las mejoras y actualizaciones pertinentes. El desempeño de estas prácticas se ha realizado de forma conjunta con otro estudiante de Ing. de Sonido e Imagen, Andrés Ramírez González.

Durante las primeras semanas de trabajo se ha hecho un análisis exhaustivo de todo el funcionamiento y la lógica implementada en la última versión, centrándose en la detección de errores y su corrección. Una vez comprendido el funcionamiento y la estructura del proyecto, se ha pasado a la fase de testing (corrección y depuración de errores). Una vez solucionados estos problemas se han implementado diferentes scripts para mejorar la fluidez de algunos minijuegos. Finalmente se han sustituido algunos scripts sobre los movimientos captados por la Kinect por otros con mejor calibración de movimientos con la ayuda de Eduardo Botija Santamaría.

El seguimiento de estas prácticas lo ha realizado Martina Eckert, tutora profesional con la que han tenido lugar una o dos reuniones semanales para comentar el avance del trabajo realizado.

## 2. OBJETIVOS DE LAS PRÁCTICAS, TAREAS Y ACTIVIDADES REALIZADAS

El videojuego de Phiby's Adventures es un proyecto a largo plazo en el que han trabajado diferentes alumnos de prácticas, becas o PFG.

En mi caso, el objetivo era depurar todos los errores existentes de versiones anteriores, después mejorar la jugabilidad de lo que ya hubiese implementado y por último avanzar con el desarrollo de la historia.

Después de una primera etapa de estudio y comprensión de estructura de carpetas y scripts a partir de la última versión proporcionada, se ha intentado cumplir con los objetivos nombrados anteriormente. A continuación, se detallan las tareas llevadas a cabo:

1. **Familiarización con el entorno:** donde se ha estudiado la estructura de carpetas del proyecto y comprendido el funcionamiento de los diferentes scripts implementados y utilizados.

Para una mejor comprensión de la lógica del juego se ha realizado un diagrama para el control de los checkpoints nombrando los diferentes scripts y líneas de código donde tiene lugar cada cambio de checkpoint.

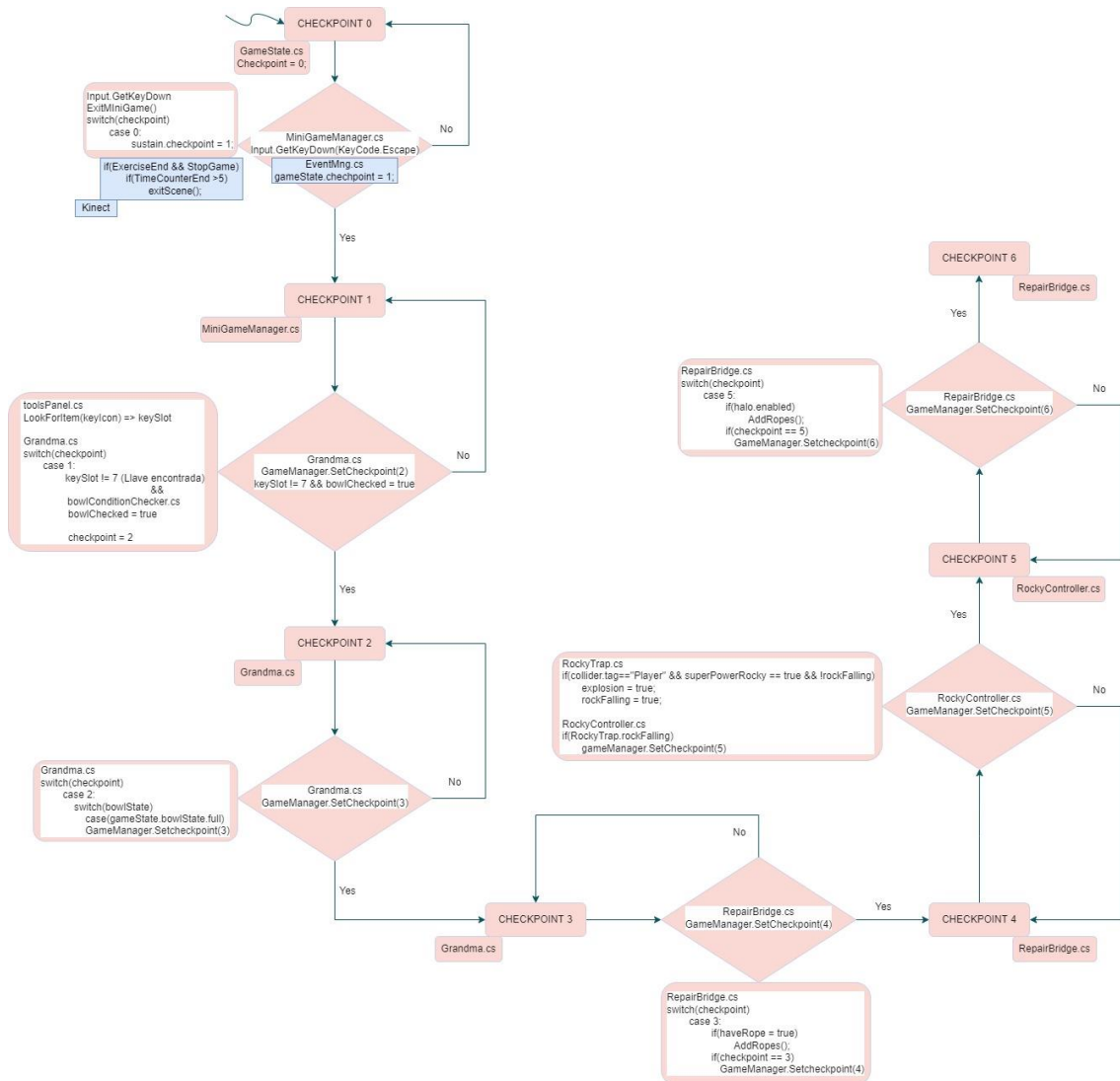
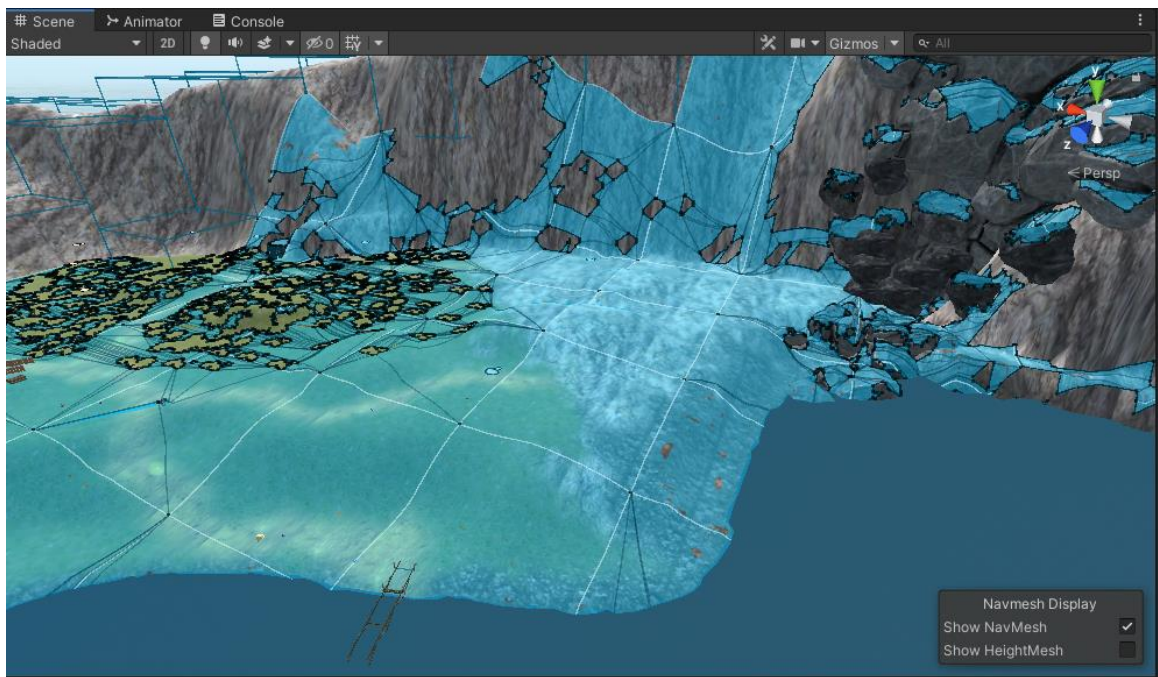


Fig. 1: Diagrama Checkpoint Control

2. **Testing:** detección, corrección y depuración de errores encontrados tanto en la escena principal del juego (Island) como en las diferentes escenas de minijuegos. Se han corregido de forma rápida pequeños fallos de diseño en iconos que aparecían mal o sin transparencia (icono de bol, cuerda, señal de Kinect y flecha minimapa). Mediante la herramienta de depuración Error Pause en la ventana Console de Unity se han ido subsanando los errores que iban apareciendo como errores de lectura en un fichero JSON de parámetros por defecto, modificación de funciones de diferentes scripts, por ejemplo, en RockyTrap.cs para activar bien la animación de las rocas y la aparición del objeto SuperPower o en KeyController.cs el arreglo de frases aleatorias. También se ha modificado el plano del terreno de la isla cercano a la cueva (Nav Mesh) porque el personaje secundario (Rocky) no se movía de forma correcta y no podía seguir al personaje principal.



*Fig. 2: NavMesh modificado de la zona de la cueva*

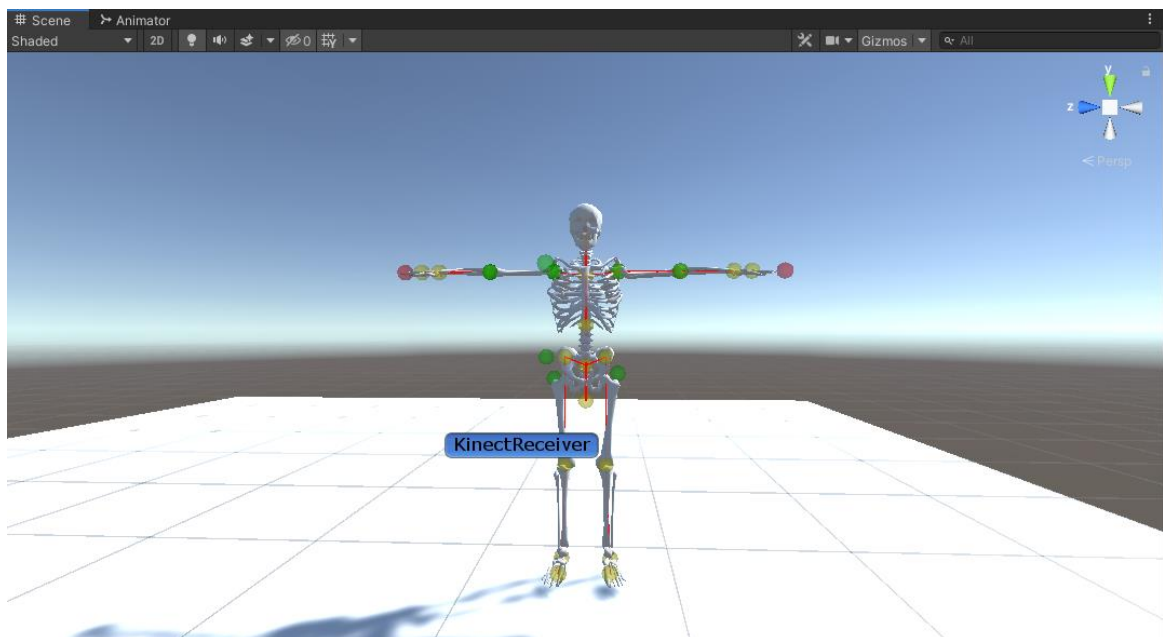
Como se puede apreciar en la Fig.2, el NavMesh ahora cubre prácticamente toda la superficie del suelo de la isla (parte de la imagen que se encuentra coloreada de azul), esto significa que el personaje de Rocky ahora puede seguir a Phiby sin ningún problema (antes se quedaba atrapado o atravesaba zonas del suelo). El NavMesh es un mapa de navegación virtual que se genera de forma automática (aunque se pueden restringir zonas) y que determina por donde pueden pasar los NPCs, queda señalado por la red de color azul claro mostrado en la Fig.2, si el terreno es muy irregular o con obstáculos muy juntos, no se podrá generar un mapa claro y los personajes secundarios tendrán problemas al pasar por ahí.

3. **Creación de scripts:** basándonos en los scripts implementados en los diferentes minijuegos, mi compañero Andrés Ramírez y yo, hemos desarrollado los scripts ClimbingTrees.cs y ClimbCage\_v2.cs. ClimbingTrees.cs mejora el minijuego de recoger manzanas haciéndolo más fluido y con una velocidad para escalar más adecuada, está basado en el script ClimbAppleTree.cs y ClimbCage\_v2.cs basado en ClimbCage\_v1.cs que mejora el minijuego de escapar de la jaula. En las escenas donde se utilizan ambos scripts, se han modificado los parámetros de calibración para un mejor funcionamiento.

Una de las modificaciones que se han hecho en estos scripts para mejorar su funcionamiento es la utilización de dos objetos vacíos con sus correspondientes colliders esféricos que constituyen el objetivo que debe alcanzar el modelo con la mano para poder escalar. Esto permite simplificar el movimiento de escalar y recoger manzanas o simplemente escalar la jaula. Cada uno de estos objetos vacíos tiene asociado un script ClimbingTrees.cs o ClimbCage\_v2.cs (según la escena), estos scripts se comunican entre sí generando una coordinación y control en el movimiento de ambas manos. El funcionamiento está basado principalmente en el uso de las funciones OnTriggerEnter y OnTriggerExit, empleando así booleanos para comprobar qué mano se ha levantado previamente.

- 4. Kinect:** en esta etapa se han encontrado muchas dificultades a la hora de solucionar los problemas de movimiento del esqueleto, ya que el problema reside en la forma del esqueleto de Phiby, debido a que no se trata de un modelo humanoide (carece de codos y manos). La lógica con Kinect utilizada en el juego es la creada por César Luaces en su PFG (es la que se ha utilizado en las versiones anteriores del juego).

En un intento de hacer la calibración completa desde el principio, tanto mi compañero Andrés como yo, hemos seguido los pasos redactados por César en el Anexo II de su PFG y, en un proyecto a parte, hemos estudiado y practicado la calibración de un esqueleto modelo proporcionado (Fig. 3). Después hemos seguido el mismo procedimiento con el esqueleto de Phiby (Fig. 4) obteniendo los mismos resultados, se tiene una rotación inicial en el brazo izquierdo que modifica el movimiento del modelo cuando se trabaja con él en la escena.



*Fig. 3: Kinect Receiver de proyecto de prueba de calibración*



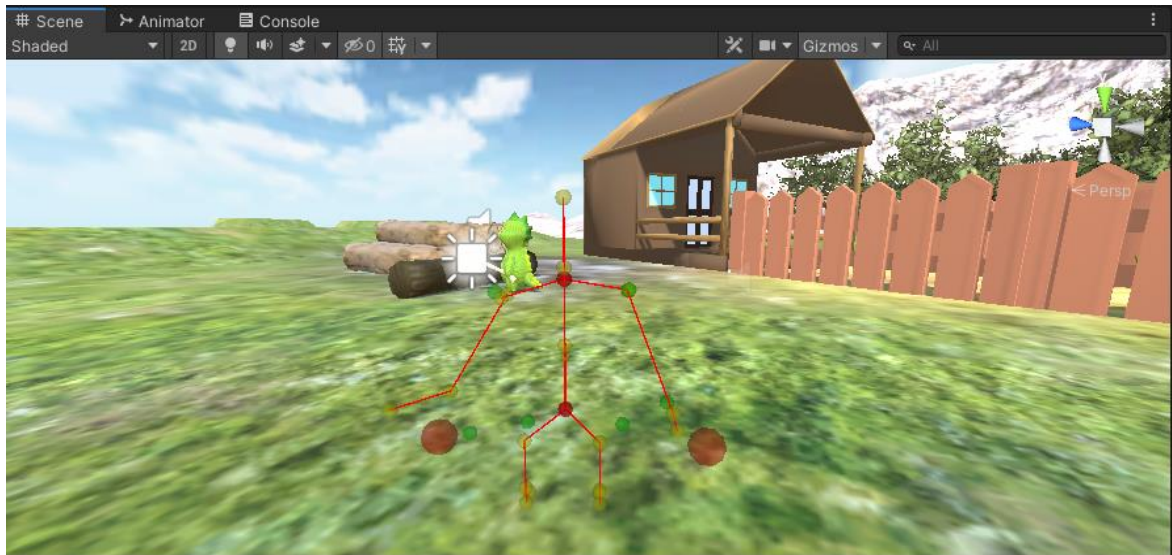


Fig. 4: Kinect Receiver Phiby

Este problema no tiene mucha relevancia en las escenas de escapar de la jaula ni de recoger manzanas, ya que el Kinect Receiver está orientado hacia el eje de coordenadas, el problema se hace visible en la escena de cortar madera donde el modelo no ha sido colocado con la misma orientación. Para solucionar este problema en la escena de cortar madera se ha utilizado un nuevo script de calibración desarrollado por Eduardo Botija Santamaría. Esta calibración es más efectiva para el movimiento de esta escena.

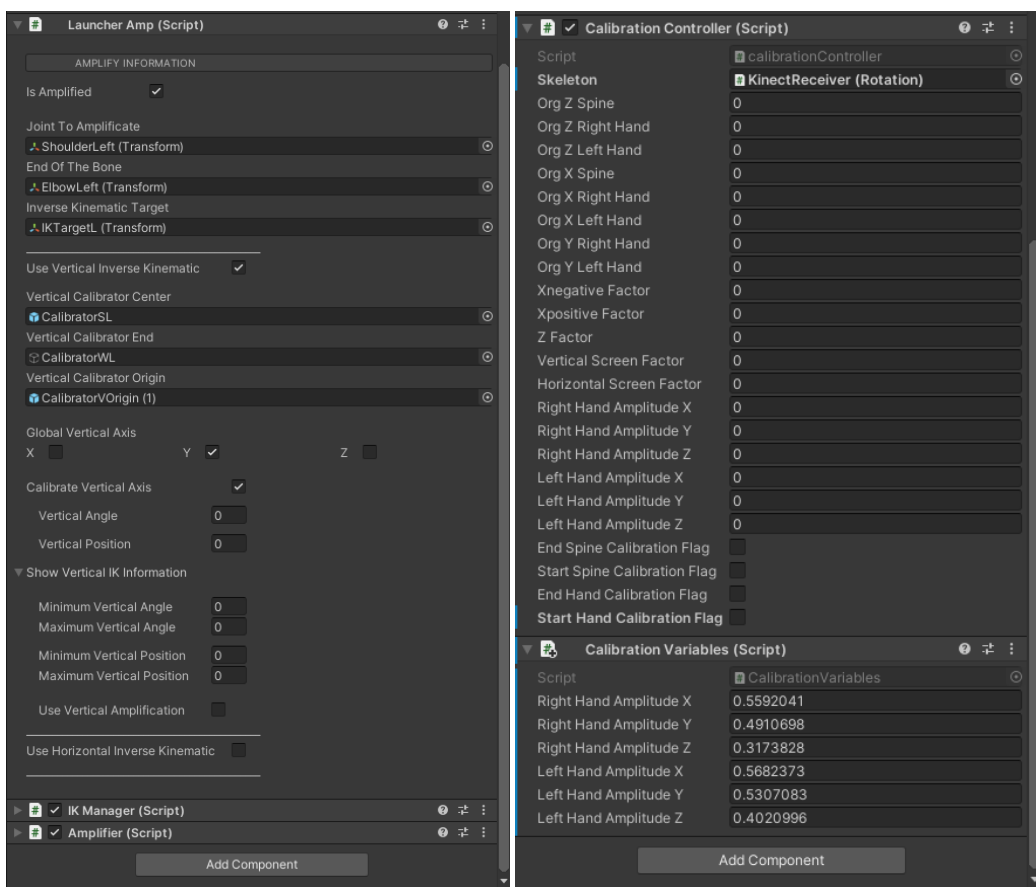
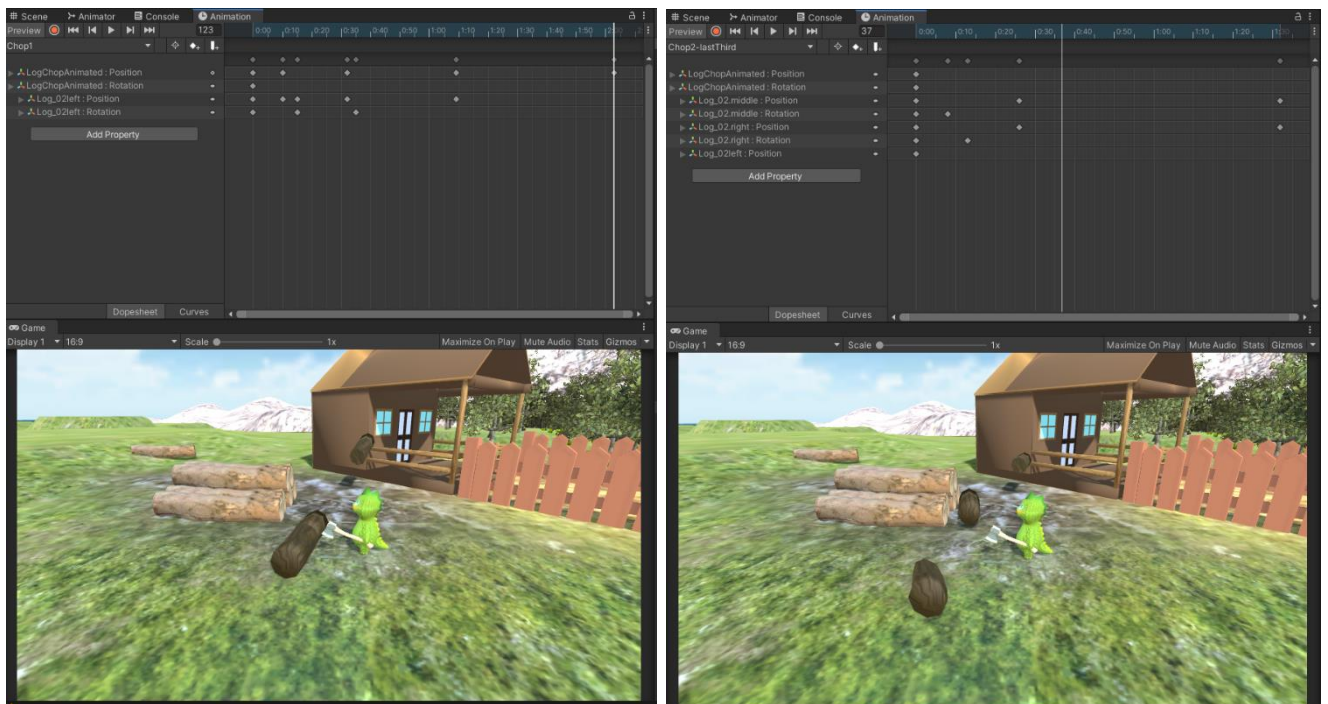


Fig. 5: Pestañas Inspector a la izquierda calibración de César Luaces, a la derecha calibración de Eduardo Botija

Para la calibración de César se necesita cambiar de posición los targets e ir anotando de forma manual los máximos y mínimos (ángulos y posiciones) que alcanza cada articulación a calibrar (sin utilizar Kinect, se realiza todo en Unity). Por el contrario, para la calibración de Eduardo el procedimiento es más sencillo y exacto. Se pulsa por teclado la letra E que inicia la calibración del tronco (Spine Start), el usuario con la Kinect realiza movimientos de tronco hacia delante, detrás y lados. Se pulsa por teclado la letra R para terminar esta calibración (Spine End), después de esto el usuario debe mover los brazos para que la Kinect capte esa calibración. Por último, se pulsa la tecla T para terminar la calibración de los brazos (Hands End). En la Fig. 5 se puede apreciar la diferencia entre ambos parámetros de calibración.

- 5. Mejora escena:** en la escena de cortar madera (ChopWoodPhiby) creada por Marta Sanz se han realizado cambios a raíz de los problemas comentados en el apartado anterior. La calibración utilizada en esta escena es más eficiente, pero sigue habiendo problemas con la orientación del modelo al ejecutar la escena (se rota apuntando hacia el eje), por tanto y como solución provisional, en vez de adaptar a Phiby a la escena, se ha adaptado la escena a Phiby. Se han cambiado los troncos de posición y modificado las animaciones respecto a la nueva posición del modelo.



*Fig. 6: Animaciones de la escena ChopWoodPhiby, a la izquierda animación 1 y a la derecha animación 2*

Al modificarse el modo de calibración de César Luaces por el de Eduardo Botija, de momento cada vez que se accede a este minijuego se debe realizar una calibración previa.

Para que de tiempo a realizar dicha calibración se ha eliminado del script ChopMovementController.cs, la cuenta atrás para salir del minijuego. En un futuro se debe implementar un script o añadir una función en alguno existente que cuente los movimientos buenos y malos y adaptar el contador de energía (como en el resto de los minijuegos). De esta manera saldremos del minijuego bien cuando la energía

se haya agotado o bien cuando se hayan conseguido los troncos requeridos. Se ha creado una función en este script, a la cual se llama en el script LogAnimated.cs, para sumar un tronco cada vez que se realiza un movimiento correcto. Así, cuando se alcanza el máximo de troncos se sale del minijuego. Este funcionamiento es provisional.

Se está trabajando en la creación de un JSON donde se guarden los parámetros de calibración de cada jugador y que sea cargado en escena de acuerdo con el usuario que va a realizar el ejercicio. Este funcionamiento será provisional hasta que esté funcionando de forma correcta la escena de calibración. A esta escena de calibración se accederá desde el menú y se deberá realizar antes de empezar a jugar. Ahora mismo se está utilizando la calibración de César Luaces en todo el juego salvo en la escena de cortar troncos (ChopWoodPhiby) que se utiliza la calibración de Eduardo Botija.

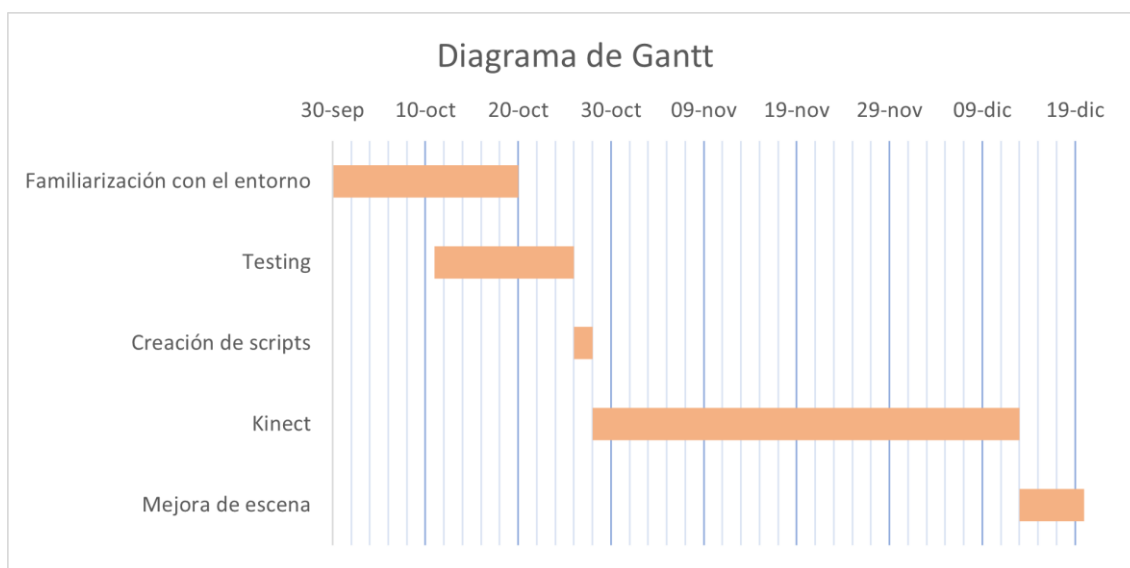


### 3. EVOLUCIÓN CRONOLÓGICA DE LAS ACTIVIDADES REALIZADAS Y LOGROS OBTENIDOS DURANTE EL PERÍODO,

De acuerdo con las tareas realizadas en el apartado 2, a continuación, se muestra la cronología de dichas actividades mediante una tabla y un diagrama de Gantt:

Nombre actividad	Fecha inicio	Duración	Fecha fin
Familiarización con el entorno	30-sep	20	20-oct
Testing	11-oct	15	26-oct
Creación de scripts	26-oct	2	28-oct
Kinect	28-oct	46	13-dic
Mejora de escena	13-dic	7	20-dic

*Tabla 1: Desarrollo de actividades y duración en días de cada una*



*Gráfica 1: Planificación de actividades mediante diagrama de Gantt*

#### 4. TECNOLOGÍAS Y MEDIOS TÉCNICOS UTILIZADOS

Para el desarrollo y ejecución de las prácticas se han utilizado los siguientes recursos de software y equipamiento:

- **Unity:** se trata de un software de desarrollo de videojuegos en tiempo real, es un motor de juego multiplataforma pensado para la implementación de juegos en 2D y 3D. Ha sido creada por Unity Technologies y está disponible para Windows, Mac OS y Linux, permitiendo el desarrollo de videojuegos para más de 25 plataformas distintas. La interfaz de programación de aplicaciones de Unity está definida en C# (C Sharp), lenguaje utilizado para la creación de scripts mediante Visual Studio. Todas las tareas realizadas durante el período de la práctica se han realizado en esta plataforma.
- **Middleware K2UM:** desarrollado por César Luaces Vela en su PFG, es un protocolo de comunicación entre la plataforma Unity y la Kinect. Esta herramienta permite transformar la información capturada por la Kinect en paquetes UDP con la información de cada articulación y traducirla en el movimiento y rotación del modelo dentro de la escena.
- **Ordenador sobremesa del despacho 46 (CITSEM):** las actividades realizadas han tenido lugar en el ordenador proporcionado por el centro, aunque algunas se han realizado desde el ordenador de sobremesa personal en casa.
- **Kinect para Xbox One:** es una versión mejorada de la Kinect para Xbox 360, recibe el nombre de Kinect 2.0. Dispone de una cámara de resolución 1080p con un nivel de precisión y reconocimiento superior al de la Xbox 360. Permite procesar 2GB de datos por segundo y captura 30 imágenes por segundo. Es compatible para PC ya que Xbox One comparte un sistema operativo basado en Kernel de Windows 10.
- **Unity Scripting API:** documentación proporcionada por Unity. Se ha utilizado para consultar dudas sobre funciones ya implementadas y para la creación de nuevos scripts.
- **Asignatura SAI:** documentación proporcionada por la asignatura Síntesis y Animación de Imágenes.
- **PFG y Memorias de prácticas:** documentación de Proyectos Fin de Grado y memorias de prácticas de alumnos que han trabajado sobre este proyecto con anterioridad.

## 5. COMPETENCIAS Y HABILIDADES ADQUIRIDAS CON LAS PRÁCTICAS

Competencias y habilidades que se han adquirido durante las prácticas:

- C\_GEN\_02: Capacidad de búsqueda y selección de información, de razonamiento crítico y de elaboración y defensa de argumentos dentro del área.
- C\_GEN\_04: Capacidad de abstracción, de análisis y de síntesis y de resolución de problemas.
- C\_GEN\_05: Capacidad de trabajo en equipo y en entornos multidisciplinares.
- C\_GEN\_08: Capacidad de organización, planificación y de toma de decisiones.
- C\_BAS\_02: Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- C\_SI\_05: Capacidad para crear, codificar, gestionar, difundir y distribuir contenidos multimedia, atendiendo a criterios de usabilidad y accesibilidad de los servicios audiovisuales, de difusión e interactivos.

## 6. CONCLUSIONES

La oportunidad de poder realizar las prácticas externas en el centro de investigación de la escuela me ha aportado conocimientos y valores positivos. Me ha permitido adquirir gran parte de los conocimientos que permiten trabajar en un proyecto de desarrollo de un videojuego en un entorno de trabajo en grupo, así como un desarrollo a nivel profesional y personal.

En cuanto al entorno de trabajo, ha sido inmejorable, la ayuda de antiguos alumnos que trabajaron en el proyecto con anterioridad y el ambiente acogedor y de compromiso e ilusión en el que me he encontrado. Me he sentido muy compenetrada con mi compañero Andrés Ramírez y con Martina Eckert (coordinadora del proyecto y también compañera).

Martina ha sido un apoyo constante durante este período y sobre todo en los momentos en que nos sentíamos atascados y desmotivados por no comprender la lógica de ciertas partes del proyecto.

Aunque ha sido una experiencia enfocada a la resolución de fallos presentes en el proyecto y no he podido avanzar con el desarrollo del juego en sí, era necesario hacer este proceso de depuración para la mejora general del juego.

Para terminar, me gustaría dar las gracias a todos los compañeros con los que he tenido el placer de trabajar y que me han ayudado a llevar este período con ganas e ilusión.