



Universidad
de La Laguna

Escuela Superior de
Ingeniería y Tecnología
Sección de Ingeniería Informática

Trabajo de Fin de Grado

Desarrollo de videojuegos para la
neurorrehabilitación mediante
eye-tracking

*Implementation of video games for neurorehabilitation
through eye-tracking*

Rebecca Martínez Galán

La Laguna, 4 de marzo de 2015

Dña. **Silvia Alayón Miranda**, con N.I.F. 43.812.596-B profesora Contratada Doctor adscrita al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor

D. **Francisco José Fumero Batista**, con N.I.F. 45.731.321-F personal investigador en formación adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutor

D. **Cristian Modroño Pascual**, con N.I.F. 51.062.048-R investigador del Grupo de Neuroquímica y Neuroimagen de la Universidad de La Laguna

C E R T I F I C A (N)

Que la presente memoria titulada:

“Desarrollo de videojuegos para la neurorehabilitación mediante eye-tracking.”

ha sido realizada bajo su dirección por Dña. **Rebecca Martínez Galán**, con N.I.F. 78.640.697-V.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 4 de junio de 2016.

Agradecimientos

En primer lugar, agradecer a mis tutores, Francisco y Silvia, que me han guiado, orientado y supervisado a lo largo de todo el proyecto, ofreciéndome su ayuda en todo momento.

A mis padres por todo su apoyo que me han dado siempre y por estar ahí en todo momento, confiando en mí y ayudándome a seguir. Por su infinita paciencia y comprensión. Sin ellos esto no hubiera sido posible.

Por último, agradecer a todos los docentes de la universidad que me han ayudado a lo largo de la carrera.

Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento 4.0 Internacional.

Resumen

El objetivo de este trabajo es el desarrollo de un videojuego serio para realizar tareas de neurorehabilitación, que ofrezca la posibilidad de ser jugado tanto con el movimiento de los ojos como con el ratón. Para ello, se ha utilizado un eye-tracker, concretamente The Eye Tribe, un dispositivo que permite registrar el movimiento ocular y saber las zonas en las que el usuario ha detenido su mirada.

Debido a que la rehabilitación es costosa y requiere tiempo, se ha realizado un juego entretenido, pero a la vez sencillo de jugar. Tras valorar varios juegos, finalmente se ha seleccionado el Break Out, un juego clásico de Atari, que cumple con los requisitos mencionados anteriormente.

Como tecnología principal de desarrollo se ha utilizado Unity, un motor gráfico que ofrece numerosas funcionalidades para facilitar el desarrollo de videojuegos.

Este TFG se ha realizado en colaboración con el Grupo de Neuroquímica y Neuroimagen de la Universidad de La Laguna, que serán los usuarios finales del juego diseñado.

Palabras clave: Neurorehabilitación, eye-tracker, The Eye Tribe, videojuego, Break Out, Unity, juego serio.

Abstract

The aim of this final undergraduate project is the implementation of a serious videogame for neurorehabilitation, which offers the possibility of being controlled with the eyes movement and with the mouse. An eye-tracker has been used (specifically The Eye Tribe). An eye-tracker is a device that allows recording the eyes movement and knowing where the user has stopped his gaze.

Neurorehabilitation is a hard task and requires time. Due to this, a funny and easy game has been developed. After evaluating several games, the Break Out was selected. Break Out is a classic game from Atari that accomplishes with the mentioned requirements.

The used technology was Unity, a cross-plataform game engine which offers countless functionalities that makes easier the development of videogames.

This final undergraduate project has been developed in collaboration with the Neurochemistry and Neuroimaging Group of the University of La Laguna, which will be the final users of the designed game.

Keywords: Neurorehabilitation, eye-tracker, The Eye Tribe, videogame, Break Out, Unity, serious game.

Índice General

Capítulo 1. Introducción	1
Capítulo 2. Estado del arte	4
2.1 Videojuegos en el ámbito médico.....	4
2.2 Videojuegos populares en el ámbito de la neurorrehabilitación.....	6
2.3 Herramientas existentes de desarrollo de videojuegos	8
Capítulo 3. Objetivos	9
Capítulo 4. Herramientas y tecnologías utilizadas	11
4.1 The Eye Tribe.....	11
4.2 Unity.....	13
4.3 C#	16
4.4 MonoDevelop.....	16
4.5 Git	16
Capítulo 5. Planificación y ejecución del proyecto.	17
5.1 Fase 1: Reunión inicial	17
5.2 Fase 2: Selección del juego.....	17
Fase 3: Primer prototipo del juego	20
5.3 Fase 4: Juego final.....	20
Capítulo 6. Juego desarrollado	21
6.1 Escena principal	21
6.2 Menú principal.....	22
6.3 Score	23
6.4 Niveles.....	23
6.5 Escena final	25

6.6 Teclado.....	26
Capítulo 7. Validación	27
Capítulo 8. Conclusiones y líneas futuras	28
8.1 Conclusiones.....	28
8.2 Líneas futuras.....	28
Capítulo 9. Summary and Conclusions	30
9.1 Conclusions	30
9.2 Future work	30
Capítulo 10. Presupuesto	32
Apéndice A. Funciones principales	33
A.1. Controlador del personaje.....	33
Apéndice B. Código completo	34
B.1. Repositorio de GitHub.....	34
Bibliografía	35

Índice de figuras

Figura 1.1. Activación de zonas cerebrales motoras cuando el paciente “controla” un objeto virtual con los ojos y/o la mano. Estudio realizado por el Grupo de Neuroquímica y Neuroimagen de la ULL con resonancia magnética funcional.....	2
Figura 2.1. Juego City Salud	4
Figura 2.2. Juego Teen City	5
Figura 2.3. Juego Re-Mission.....	6
Figura 2.4. Primer ejemplo de uso de VirtualRehab.....	7
Figura 2.5. Segundo ejemplo del uso de VirtualRehab	7
Figura 4.1. The Eye Tribe	11
Figura 4.2. Ejemplo de uso del dispositivo The Eye Tribe.....	12
Figura 4.3. Calibración The Eye Tribe.....	12
Figura 4.4. Ventana principal de Unity.....	14
Figura 4.5. Inspector.....	15
Figura 4.6. Ficheros del proyecto	15
Figura 5.1. Space Invaders. En este juego el objetivo es matar a todos los marcianos sin morir.....	18
Figura 5.2. Break Out. En este juego el objetivo es romper todos los bloques sin que la bola caiga al suelo	18
Figura 5.3. Temple Run. Juego que consiste correr la mayor distancia posible, sin caerse, tropezar o ser cogido por el primate que persigue al personaje. 18	
Figura 5.4. Snowball Slalom. En este juego el personaje es una bola de nieve que tiene que ir esquivando obstáculos.....	19
Figura 5.5. Danger bob-omb danger. En este juego el personaje tiene que esquivar todas las bolas y llamaradas de fuego. El objetivo es aguantar el mayor tiempo posible sin morir.....	19
Figura 6.1. Primera escena.....	21
Figura 6.2. Escena de calibración.....	22

Figura 6.3. Menú Principal	22
Figura 6.4. Score	23
Figura 6.5. Primer nivel.....	23
Figura 6.6. Bloques de una vida.....	24
Figura 6.7. Bloques de dos vidas.....	24
Figura 6.8. Bloques de tres vidas	24
Figura 6.9. Bloques irrompibles.....	25
Figura 6.10. Escena final.....	25
Figura 6.11. Teclado	26

Índice de tablas

Tabla 8.1. Tabla de presupuesto	32
---------------------------------------	----

Capítulo 1.

Introducción

El uso de videojuegos en los distintos ámbitos de la vida cotidiana es cada vez más común debido a la gran cantidad de ventajas que ofrece, como son, por ejemplo, la mejora de las habilidades motoras y visuales o el aumento de la sociabilidad. Además, es una forma común y divertida de pasar el rato para la mayoría de las personas. Por tanto, en este trabajo se pretende aprovechar todas estas ventajas para ofrecer una herramienta de ayuda para la recuperación de las funciones motoras de personas con daño cerebral.

Para ello, se propone la realización de un videojuego que se controle tanto con el ratón como con el movimiento de los ojos, mediante el uso de un eye-tracker, un dispositivo que permite registrar el movimiento ocular con mucha exactitud, y saber el recorrido que la persona ha hecho con los ojos y las zonas en las que ha detenido más su mirada. El que se usará para este proyecto es The Eye Tribe.

A continuación, se explicarán algunos conceptos importantes necesarios para entender bien el problema abordado. Se podría considerar como el más importante de ellos el de daño cerebral, ya que esta es la razón de ser del proyecto.

El daño cerebral es la principal causa de discapacidad del adulto en los países de nuestro entorno. Implica la destrucción o degeneración de células cerebrales, y da lugar a deficiencias o discapacidades que pueden variar mucho en intensidad. Una de las principales consecuencias negativas derivadas del daño cerebral son los déficits motores (presentes en el 70% de los ictus), como por ejemplo, la imposibilidad de mover un brazo. En concreto, las complicaciones que afectan a los miembros superiores están entre las más incapacitantes, llegando a imposibilitar en muchas ocasiones las acciones de la vida cotidiana.

Como solución a este problema está la neurorrehabilitación, un proceso médico centrado en la recuperación del sistema nervioso tras una lesión

neurológica, que tiene como misión minimizar y compensar las alteraciones funcionales.

La idea de este proyecto surge de diversas investigaciones realizadas sobre personas sanas por el *Grupo de Neuroquímica y Neuroimagen de la Universidad de La Laguna, equipo colaborador en el presente TFG*. En los estudios realizados [1], la persona debía controlar un objeto virtual que se movía en una pantalla mediante dos métodos de control diferentes: usando la mano y usando la mirada.



Figura 1.1. Activación de zonas cerebrales motoras cuando el paciente “controla” un objeto virtual con los ojos y/o la mano. Estudio realizado por el Grupo de Neuroquímica y Neuroimagen de la ULL con resonancia magnética funcional.

Los resultados mostraron que utilizar los ojos para controlar un objeto virtual producía una gran activación cerebral en regiones motoras, muy similar a la que se produce cuando el objeto se controla con la mano (ver figura 1). Este hecho, que podría parecer sorprendente, no lo es tanto si se tiene en cuenta que los movimientos de la mano y los ojos están muy acoplados en la vida diaria.

Estos resultados tienen una aplicación potencial en el campo de la neurorrehabilitación, como una nueva forma para generar la activación del sistema sensoriomotor. El entrenamiento de control de elementos virtuales con los ojos podría ayudar a incrementar la actividad cortical de regiones relacionadas con el movimiento de los miembros afectados.

Además, este tipo de rehabilitación ofrece ciertas ventajas entre las que se encuentran:

- Facilidad de uso.
- Permite obtener registros del paciente.
- La terapia física es un tratamiento común para déficits motores, pero no siempre es posible debido a las limitaciones en la extremidad afectada. Con este tipo de rehabilitación no se presenta este problema.
- Las regiones cerebrales activadas son grandes y adecuadas.
- No es necesaria la presencia continua del terapeuta. El sistema propuesto se podrá instalar fácilmente en los centros de rehabilitación o en los domicilios de los pacientes. Esto revertiría en la optimización de los servicios de salud, ahorrando tiempo y dinero a pacientes y personal sanitario.
- La tecnología no es cara. La existencia de equipos de eye-tracking comerciales de bajo coste asegura la disponibilidad del único componente necesario que no suele encontrarse en la casa de la mayoría de las personas. Esto abarata bastante el coste y facilita la proyección a nivel mundial.
- Los fundamentos biológicos sencillos de comprender, facilitando su difusión en el público y sectores implicados (como pacientes, centros sanitarios o empresas de tecnología biomédica).

Capítulo 2.

Estado del arte

En este capítulo se hablará del estado actual de los videojuegos en el ámbito médico. Además, se nombrarán ejemplos de juegos desarrollados para ser usados en rehabilitación y los distintos motores de desarrollo de existentes para videojuegos.

2.1 Videojuegos en el ámbito médico

El uso de videojuegos en el ámbito médico se ha extendido enormemente estos últimos años. Existen dos tipos principales de videojuegos en esta área: aquellos diseñados para la prevención y promoción de la salud (Educación para la salud) y aquellos diseñados para la mejora de la salud, como es el que se pretende implementar en este TFG.

Dentro de los videojuegos de educación para la salud entrarían los videojuegos que fomentan el uso saludable de ejercicio (Exergaming) y aquellos que nos acercan a conductas y conocimientos sobre salud.

Como ejemplo de este tipo de juegos está “City Salud” [2], un juego online que trata la educación para la salud desde múltiples puntos de vista: enseña valores sobre alimentación saludable, actividad física, higiene, descanso, y previene hábitos nocivos como el consumo de tabaco y alcohol (figura 2.1).



Figura 2.1. Juego City Salud

Otro juego de este estilo es “Teen City” [3], un videojuego de simulación basado en el planteamiento de situaciones y toma de decisiones a través de la recreación de rutinas diarias. Según estas decisiones variarán los valores de las barras de estado: comportamiento, educación, amistad, familia, salud, amor, satisfacción y dinero (figura 2.2).



Figura 2.2. Juego Teen City

En cuanto a los videojuegos para mejorar la salud, éstos van más dirigidos a un sector particular de la población, concretamente a las personas que tienen el problema de salud para el que fue diseñado el juego.

Entre este tipo de videojuegos destaca “Re-Mission” [4], creado específicamente para jóvenes con cáncer (figura 2.3). Los jugadores pilotan a Roxxi, un nanobot que en el año 2025 viaja a través de los cuerpos de los pacientes con cáncer para destruir las células cancerosas, luchando contra las infecciones bacterianas y la gestión de efectos secundarios derivados del cáncer y su tratamiento. El videojuego fue probado en un ensayo que pretendía evaluar la eficacia del juego. Fue utilizado por 375 jóvenes ingresados en 24 centros médicos de EEUU, Australia y Canadá. Los datos del estudio fueron publicados en la revista médica Pediatrics [5] y mostraron no sólo una mejora en la calidad de vida, en los conocimientos acerca del cáncer, en su confianza a la hora de hablar de la enfermedad, o en los efectos secundarios, sino que además los pacientes respondieron mejor a la quimioterapia.



Figura 2.3. Juego Re-Mission

2.2 Videojuegos populares en el ámbito de la neurorrehabilitación

Durante la última década, los investigadores han combinado la ciencia y la tecnología para desarrollar nuevos métodos de rehabilitación. Estos incluyen el uso de videojuegos y de realidad virtual, que permite a los pacientes realizar tareas específicas que las terapias tradicionales no proporcionan.

Una de las herramientas más avanzadas en este ámbito es VirtualRehab [6], una herramienta diseñada por expertos en neurorrehabilitación que incluye gran diversidad de programas de rehabilitación para pacientes con diferentes grados de discapacidad física (figuras 2.4 y 2.5). En un entorno basado en tecnología de videojuegos, combinado con sensores de movimiento como Microsoft Kinect y Leap Motion, y tecnología en la nube con Microsoft Azure, el paciente puede realizar los ejercicios de terapia previamente pautados por el terapeuta en el propio centro o también desde casa. VirtualRehab además permite que el paciente sea monitorizado remotamente por su fisioterapeuta para evaluar su evolución.



Figura 2.4. Primer ejemplo de uso de VirtualRehab

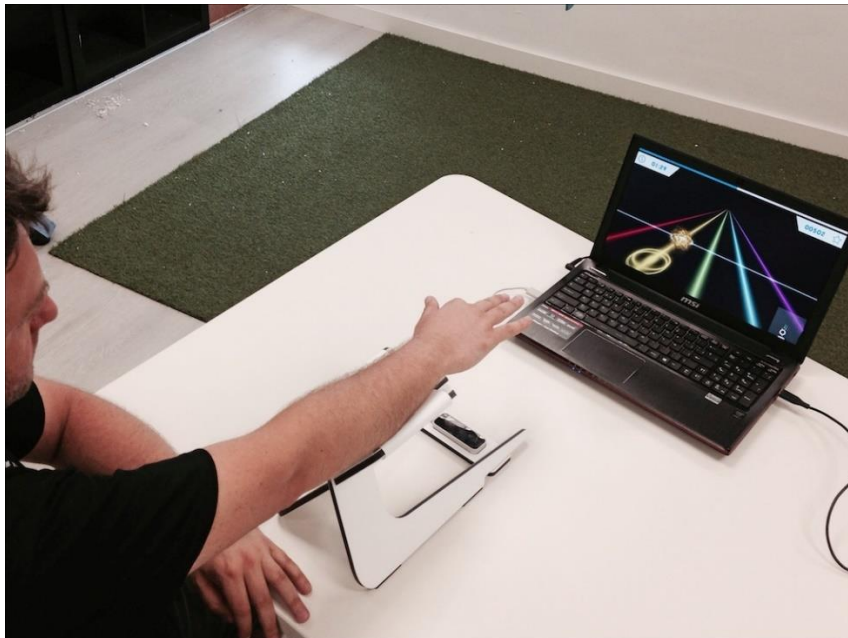


Figura 2.5. Segundo ejemplo del uso de VirtualRehab

VirtualRehab 3.0 está disponible en varias modalidades y paquetes dirigidos tanto a pequeños centros como grandes hospitales. Incluye VirtualRehab Body, con juegos para la rehabilitación de extremidades superiores e inferiores, utilizando el sistema de captura de movimiento Kinect Xbox One de Microsoft y VirtualRehab Hands, la principal novedad de este lanzamiento. Este nuevo módulo incluye nuevos juegos de motricidad fina además de un soporte ergonómico para descansar el brazo y poder realizar los

ejercicios de manera más cómoda con el sensor Leap Motion, que detecta con precisión los movimientos de las manos, dedos y muñecas.

2.3 Herramientas existentes de desarrollo de videojuegos

Existe un gran número de herramientas de desarrollo de videojuegos, tanto gratuitas como de pago. A continuación, se nombrarán las más populares, pero es importante tener en cuenta que existen muchas más.

Unreal Engine [7] y Unity [8] son los motores de desarrollo de videojuegos más populares en la actualidad. Ambos son muy potentes y ofrecen innumerables funcionalidades como el desarrollo de juegos en 2D y 3D. Además, son multiplataforma. Permiten la creación de juegos para diferentes sistemas operativos y dispositivos.

Otros programas de este estilo son:

- Cry Engine [9], un motor gráfico extremadamente potente. Está diseñado para usarse en juegos de PC y consolas, incluyendo PlayStation 4 y Xbox One.
- Gamemaker: Studio [10], una herramienta de desarrollo rápido de aplicaciones, basada en un lenguaje de programación interpretado y un kit de desarrollo de software (SDK) para desarrollar videojuegos, creado por el profesor Mark Overmars en el lenguaje de programación Delphi.
- Blender [11], es un programa informático multiplataforma, dedicado especialmente al modelado, iluminación, renderizado, animación y creación de gráficos tridimensionales. En Blender, además, se puede desarrollar vídeo juegos ya que posee un motor de juegos interno.

Aunque éstos son los más conocidos, existen muchos otros, cada uno con sus ventajas e inconvenientes, en función del juego que se desee implementar.

Capítulo 3.

Objetivos

El objetivo principal de este proyecto es el desarrollo de un videojuego que cumpla las características necesarias para poder ser utilizado en un proceso de neurorrehabilitación.

Entre las características más importantes se encuentra el modo de control del juego. El juego no sólo debe poder jugarse por un medio tradicional como el ratón, sino que también tiene que ofrecer la posibilidad de ser controlado a través de la mirada, mediante el movimiento de los ojos. Ésta podría considerarse la característica más importante de todas, ya que el juego está pensado para ser usado a través del movimiento ocular por personas con problemas motores en los brazos. Este modo de juego beneficiaría potencialmente a la recuperación del movimiento en pacientes con daño cerebral.

Otra característica importante es la jugabilidad. El número de movimientos posibles del personaje debe estar limitado. Esto es debido a que no es posible realizar ciertos movimientos si el personaje se controla con la mirada, ya que lo único que se conoce es hacia donde está mirando el usuario, pero no hay forma de presionar botones ni de recibir ningún otro tipo de información.

También es necesario tener en cuenta que ciertos tipos de juegos pueden ser controlados de forma sencilla con la vista, pero no por eso resultan adecuados. Por ejemplo, los juegos de esquivar obstáculos podrían no ser una buena opción ya que el usuario mirará los obstáculos y esto provocará que el personaje se mueva hacia ellos en lugar de esquivarlos.

Además, como la recuperación es costosa y requiere tiempo, es necesario que el juego sea entretenido. El usuario pasará muchas horas jugando, y mover un objeto virtual con los ojos es algo cansado y requiere más esfuerzo que hacerlo con medios tradicionales, por ello es importante que el usuario no se aburra durante la sesión de rehabilitación.

Una última característica que debe tener el juego es la de guardar en un fichero las posiciones de los ojos mientras el usuario está jugando, para tener una traza del movimiento ocular del paciente. Esta información será revisada y utilizada por los médicos en sus investigaciones.

Capítulo 4.

Herramientas y tecnologías utilizadas

En este apartado se nombrarán todas las tecnologías que se han utilizado durante el desarrollo del videojuego objeto de este TFG. Además, se explicará para qué fueron utilizadas y cuáles son sus principales características.

4.1 The Eye Tribe

Un dispositivo equipado con un eye-tracker permite a los usuarios utilizar su mirada como método de entrada en vez de utilizar métodos tradicionales como pueden ser el ratón o el teclado. Además, su uso se puede combinar con dispositivos de entrada tradicionales, el tacto y los gestos. En este proyecto se ha utilizado The Eye Tribe como dispositivo de eye-tracker.

The Eye Tribe [12] es un sistema de eye-tracking que puede calcular el lugar al que una persona está mirando por medio de la información extraída de la cara y los ojos (figuras 4.1 y 4.2). Las coordenadas se calculan con respecto a una pantalla que la persona está mirando, y están representados por un par de coordenadas (x, y) .



Figura 4.1. The Eye Tribe

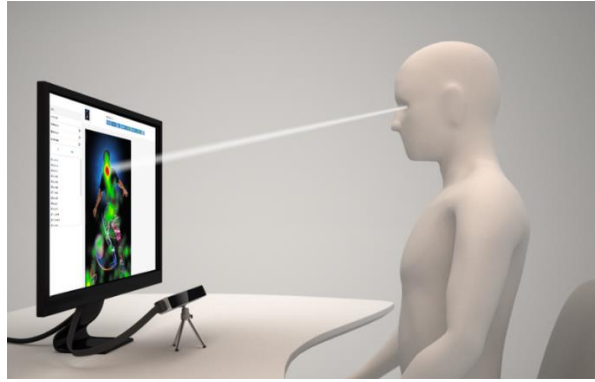


Figura 4.2. Ejemplo de uso del dispositivo The Eye Tribe

Con el fin de realizar un seguimiento de los movimientos oculares del usuario y calcular las coordenadas, el eye-tracker debe colocarse debajo de la pantalla y orientado al usuario (figura 4.2).

Para que el eye-tracker funcione correctamente, y calcule las coordenadas de forma adecuada, es necesario que se realice una calibración inicial (figura 4.3), que consiste en ir mirando unos puntos que van apareciendo en la pantalla. Una vez calibrado el sistema, el eye-tracker calcula la posición de la mirada en la pantalla. Suponiendo que el usuario se encuentra aproximadamente a 60 cm de distancia de la pantalla/eye-tracker, esta precisión se corresponde con un error medio de la pantalla de 0,5 a 1 cm.

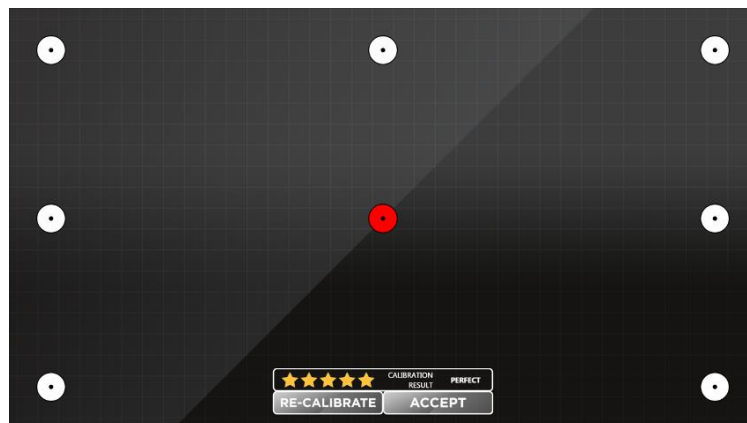


Figura 4.3. Calibración The Eye Tribe

Entre los lenguajes de programación que se pueden usar con este eye-tracker se encuentran C#, C++ y Java.

El principal motivo por el que se usó este eye-tracker y no otro, es por su relación calidad – precio. Se trata de un dispositivo relativamente barato para la precisión que ofrece si lo comparamos con otros del mismo estilo. Su precio es 99\$. Su principal problema es que su precisión disminuye mucho cuando el usuario mueve la cabeza tras la calibración.

El hecho de que sea barato es importante ya que la idea es que los pacientes puedan jugar en casa, y, por tanto, que puedan permitirse comprar un eye-tracker si es necesario.

Recientemente ha salido una nueva versión de The Eye Tribe, que permite al usuario mover la cabeza sin perder precisión con un precio de 199\$.

4.2 Unity

El programa utilizado para la realización del videojuego ha sido Unity, un motor de videojuego multiplataforma creado por Unity Technologies. Unity está disponible como plataforma de desarrollo para Microsoft Windows, OS X y Linux, y permite crear juegos para Windows, OS X, Linux, Xbox 360, PlayStation 3, Playstation Vita, Wii, Wii U, iPad, iPhone, Android y Windows Phone. Gracias al plugin web de Unity, también se pueden desarrollar videojuegos de navegador para Windows y Mac. Unity tiene dos versiones: Unity Professional y Unity Personal.

Es uno de los motores más versátiles del momento, y ofrece multitud de opciones a aquellos que desean iniciarse en el desarrollo de juegos multiplataforma. Además, existe mucha documentación sobre este motor y número ayuda en múltiples formatos para comenzar a desarrollar utilizándolo.

Las características principales de este programa son:

- Posibilita la publicación en numerosas plataformas, sin realizar ninguna tarea de implementación extra.
- Herramientas dedicadas para la creación de contenido 2D y 3D.
- Importación de modelos y animaciones realizadas con otras aplicaciones 3D, como pueden ser Blender, Maya, 3ds Max, Modo, Cinema 4D, etc., en el que Unity realizará y actualizará los cambios en todo el proyecto.

- Construcción rápida de escenas (niveles de juego) para añadir nuestros objetos 2D y 3D.
- Integración con los motores de físicas de NVIDIA(r) PhysX(r) y Box2D.
- Iluminación de sombras en tiempo real, además de proporcionar una herramienta llamada “Particle System”, encargada de simular líquidos, llamas o nubes mediante el uso de pequeñas imágenes 2D en la escena.
- Orientado a componentes, que permitan aumentar los módulos de un videojuego.

La ventana principal de este motor gráfico se muestra en la figura 4.4.

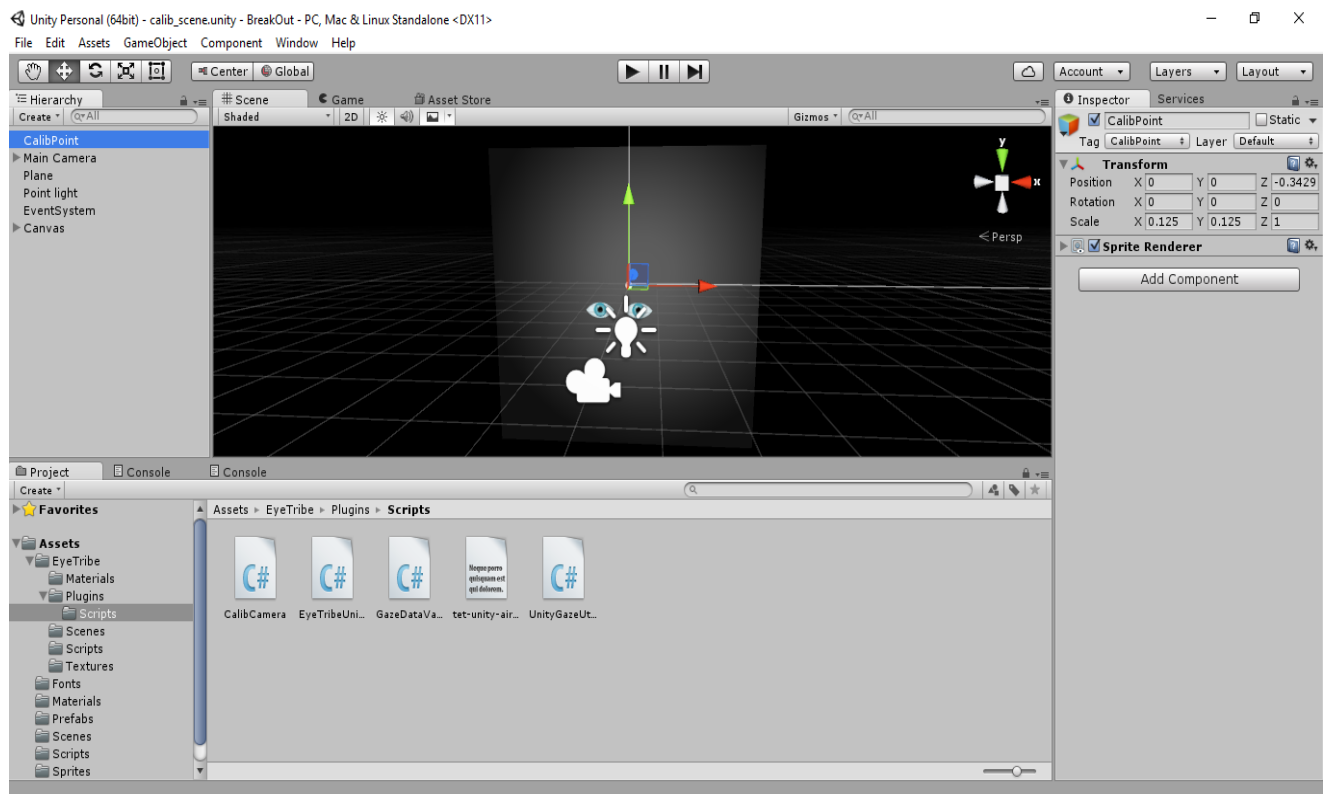


Figura 4.4. Ventana principal de Unity

Desde aquí se implementa la mayor parte del juego. Se crean y editan las escenas, los objetos de cada una de ellas y sus propiedades y componentes.

En la parte izquierda aparecen todos los elementos de la escena actual. Cuando se selecciona uno de esos elementos aparece en la parte derecha el

inspector, que es un menú con sus componentes y las opciones de edición (figura 4.5).

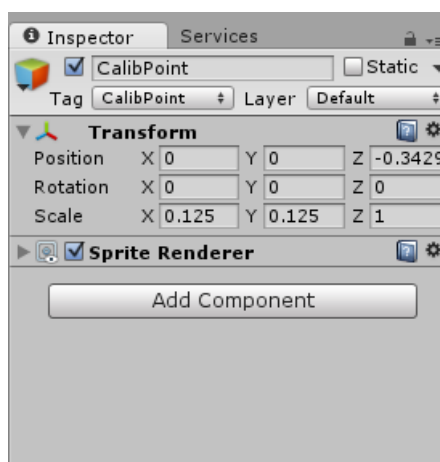


Figura 4.5. Inspector

En la parte inferior están los directorios del proyecto y cada uno de los archivos que hay en ellos. Cuando se selecciona un directorio en particular aparecen todos los elementos que contiene (figura 4.6).

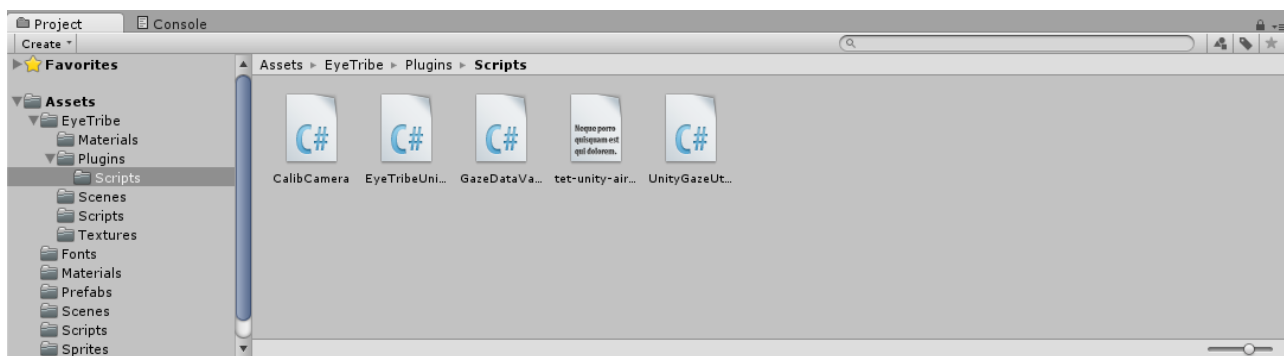


Figura 4.6. Ficheros del proyecto

La razón principal por la que se ha usado este programa es porque facilita el desarrollo de videojuegos. Permite realizar juegos en pocos pasos, que se adaptan a todo tipo de resoluciones, proporcionando un control absoluto sobre las escenas creadas. Además, permite el uso del lenguaje de programación C#, que es uno de los tres que se pueden usar para The Eye Tribe.

4.3 C#

C# [13] es un lenguaje de programación orientado a objetos desarrollado por Microsoft y estandarizado como parte de su plataforma .NET, derivando su sintaxis, al igual que Java, de lenguajes como C/C++.

La razón por la que se ha utilizado como lenguaje de programación para el juego ha sido porque es el que Unity usa como lenguaje para programar los scripts y es uno de los que se puede usar con The Eye Tribe.

4.4 MonoDevelop

MonoDevelop [14] es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows, Linux y Mac. Soporta múltiples lenguajes de programación tales como C++, C#, Visual Basic .NET, F#, Java, Python, etc.

Se ha utilizado para escribir los scripts utilizados en el juego. Viene instalado por defecto con Unity, aunque hay una opción para excluirlo en la ventana de instalación.

4.5 Git

Git [15] es un software de control de versiones pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente.

Se ha utilizado GitHub [16], una plataforma de desarrollo colaborativo que permite alojar proyectos utilizando el sistema de control de versiones Git.

Capítulo 5.

Planificación y ejecución del proyecto.

El desarrollo de este proyecto se ha realizado en varias fases. A medida que se iba pasando de una fase a otra, se iban realizando reuniones con los tutores y el equipo médico colaborador en este TFG para ir mostrando los avances y ver cómo proseguir en la siguiente fase.

5.1 Fase 1: Reunión inicial

La primera reunión se realizó en el Hospital Universitario de Canarias (HUC), donde se explicó en detalle en qué consistía el Trabajo de Fin de Grado. Se habló de la motivación del proyecto y de la tecnología que se iba a utilizar (The Eye Tribe). Además, se analizaron las características que debía tener el juego a desarrollar.

5.2 Fase 2: Selección del juego

Tras la primera reunión, se le presentó al equipo médico una selección de juegos posibles a desarrollar, para determinar cuáles eran los más adecuados para el problema abordado.

Entre los juegos pensados, estaban dos juegos clásicos: “Space Invaders” [17] y Break Out [18] (figuras 5.1 y 5.2 respectivamente).

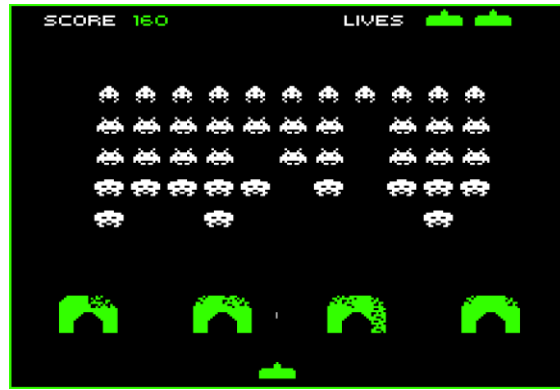


Figura 5.1. Space Invaders. En este juego el objetivo es matar a todos los marcianos sin morir.

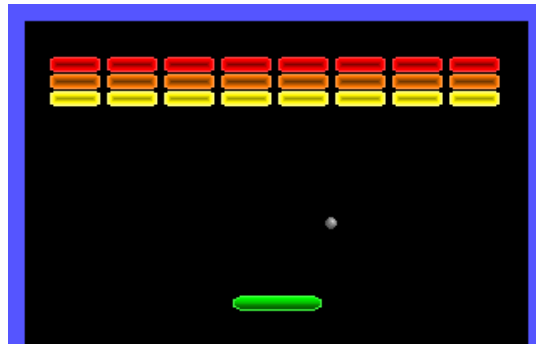


Figura 5.2. Break Out. En este juego el objetivo es romper todos los bloques sin que la bola caiga al suelo

También se pensó en la realización de un infinity runner. Un ejemplo de este tipo de juego sería el “Temple Run” [19] (figura 5.3).



Figura 5.3. Temple Run. Juego que consiste correr la mayor distancia posible, sin caerse, tropezar o ser capturado.

Dos últimos juegos pensados fueron “Snowball Slalom” [20] y “Danger bob-omb danger” [21], ambos minijuegos de Mario Bross (figuras 5.4 y 5.5 respectivamente).



Figura 5.4. Snowball Slalom. En este juego el personaje es una bola de nieve que tiene que ir esquivando obstáculos.



Figura 5.5. Danger bob-omb danger. En este juego el personaje tiene que esquivar todas las bolas y llamaradas de fuego. El objetivo es aguantar el mayor tiempo posible sin morir.

El infinity runner fue descartado ya que se consideró demasiado complejo para jugar con la vista. Sin embargo, se determinó que los juegos más simples de controlar, como el “Space Invaders” y el “Break out”, sí que podrían servir para la neurorrehabilitación. El “Snowball Slalom” y el “Danger bob-omb” también podrían servir, pero habría que tener en cuenta ciertos problemas a

la hora de controlarlos con la vista, ya que al tratarse de juegos en los que hay que esquivar obstáculos, el usuario fijará la vista en ellos, lo que provocaría que el personaje se acercara a ellos, en lugar de esquivarlos. Como posible solución se propuso que el personaje solo se moviera cuando el usuario estuviera mirando dentro de un rango.

Finalmente se decidió empezar por juegos sencillos y, si daba tiempo, pasar a juegos más complicados. Por lo tanto, el objetivo que se propuso para este TFG fue la implementación del juego “Break Out”, un juego en el que se maneja una plataforma y cuyo objetivo es destruir todos los bloques sin que la bola caiga al suelo. Hay un número máximo de veces que la bola puede caer al suelo antes de perder definitivamente. Si se consiguen destruir todos los bloques, se pasa al siguiente nivel.

Fase 3: Primer prototipo del juego

En el primer prototipo desarrollado se implementó el juego completo para ser controlado manualmente con un ratón.

Una vez terminado, los tutores y el equipo médico evaluaron una demo del videojuego para ver si estaba bien implementado y si era necesario añadir o eliminar alguna funcionalidad.

Uno de los puntos analizados fue si ponerle o no sonido al juego. Finalmente se decidió que no era necesario ya que no aportaba nada desde el punto de vista médico. Otro tema que se trató fue el de guardar en un fichero las posiciones de los ojos mientras el jugador está jugando. El médico nos pidió que guardáramos esos datos para poder revisarlos después y así tener una traza del movimiento ocular del paciente.

5.3 Fase 4: Juego final

En esta fase se desarrollaron los cambios mencionados en la fase anterior y se integró el eye-tracker al juego. Para ello, se realizó la programación necesaria para permitir el control del con la vista.

Cuando el desarrollo estuvo completo, se mostró a los tutores y al médico, los cuales, tras realizar varias pruebas con él, dieron su visto bueno.

Capítulo 6. Juego desarrollado

El videojuego desarrollado está formado por varias escenas. En primer lugar, se desarrolló la escena del menú principal y la correspondiente al primer nivel. Seguidamente, se implementó la escena final, que es a la que el jugador va a ser dirigido cuando gane o pierda.

Una vez hechas estas escenas, se procedió a crear más niveles y a añadir nuevas funcionalidades como las vidas y la puntuación. Posteriormente, se añadió un teclado y la funcionalidad de guardar las puntuaciones. La última escena añadida fue la escena principal, donde se ofrecía la posibilidad de jugar con los ojos. Antes de crear esa escena, el usuario sólo podía jugar con el teclado.

A continuación, se describirán con detalle cada una de las escenas del juego final y sus funcionalidades.

6.1 Escena principal

La primera escena que aparece cuando se ejecuta el juego, permite elegir entre jugar con el ratón o jugar con la vista. Además, ofrece la opción de guardar la posición de los ojos del jugador durante el desarrollo del juego en un fichero, en caso de que se vaya a jugar con el eye-tracker (figura 6.1).

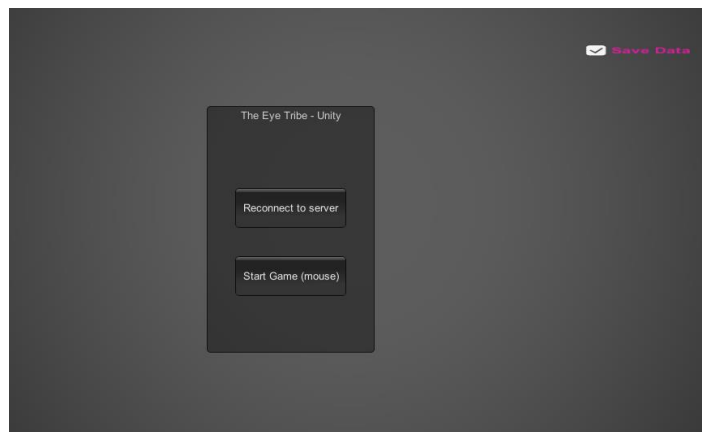


Figura 6.1. Primera escena

Una vez seleccionado el modo de juego, aparecerá el menú principal. Si se ha decidido jugar con el eye-tracker, será necesario hacer una calibración antes de llegar al menú principal del juego (figura 6.2).

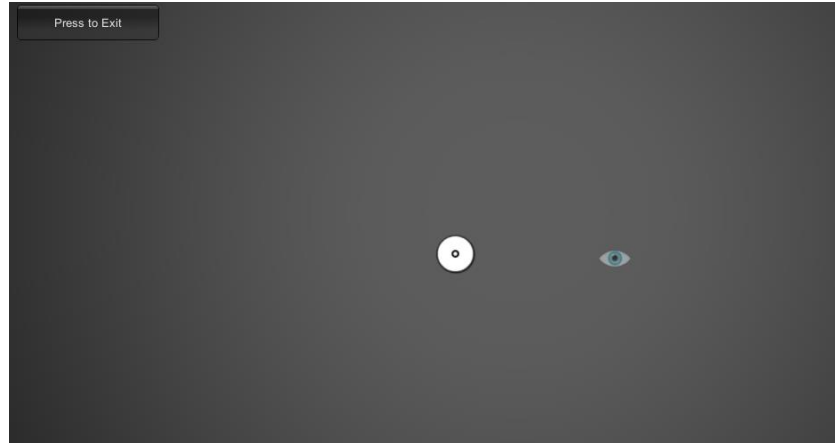


Figura 6.2. Escena de calibración

6.2 Menú principal

En el menú principal aparecen varias opciones: jugar (Play), ver las puntuaciones (Score) y salir del juego (Exit), tal y como se muestra en la figura 6.3.

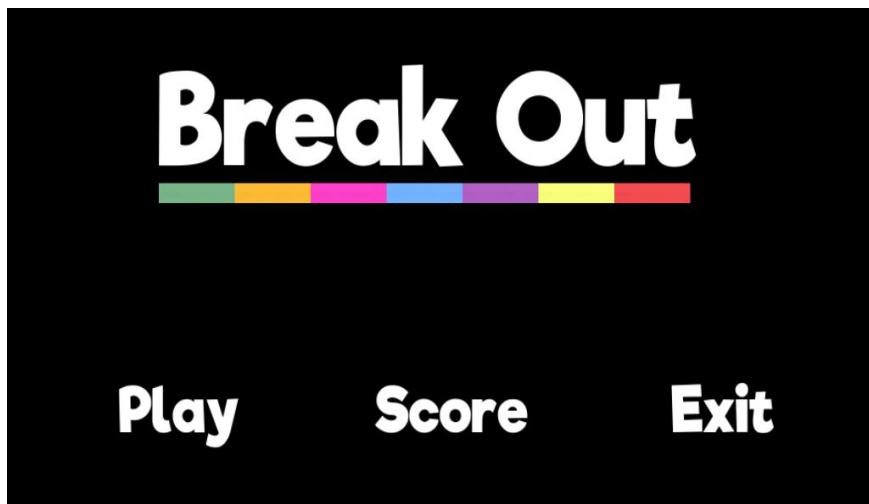


Figura 6.3. Menú Principal

6.3 Score

Si se selecciona la opción de “Score” (figura 6.4), aparecerán las puntuaciones de las personas que han jugado y han decidido guardar su puntuación, ordenada de mayor a menor. Las puntuaciones se guardan de forma local, por lo tanto, las que aparecen dependerán del ordenador en el que se juegue.

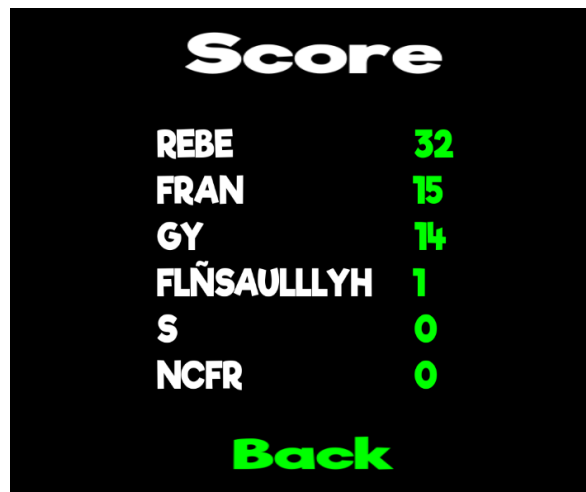


Figura 6.4. Score

6.4 Niveles

Si se selecciona la opción de “Play”, aparece el primer nivel del juego (figura 6.5). El juego está compuesto por cinco niveles.

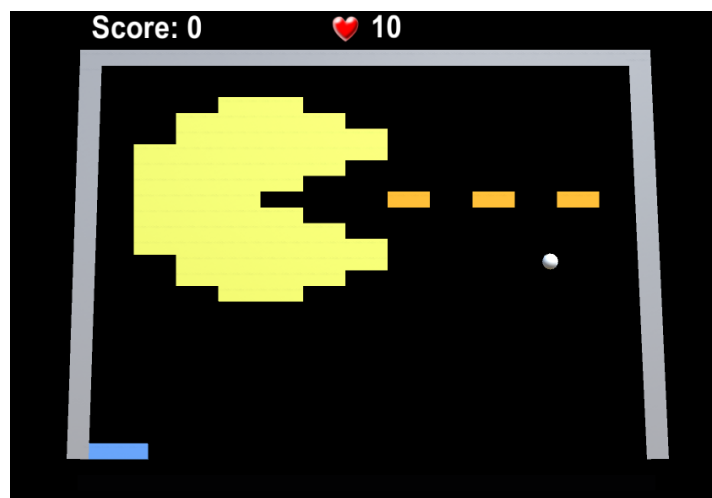


Figura 6.5. Primer nivel

Cuando se comienza el juego, se tienen 10 vidas y 0 puntos. Cada vez que se da a un bloque, se gana un punto y cada vez que se pierde una vida se pierden cinco puntos.

Cada nivel tiene los bloques distribuidos de forma diferente. Además, a medida que se va aumentando de nivel aparecen distintos tipos de bloques.

El juego consta de 4 tipos de bloques:

- Bloques de 1 vida: sólo hay que darles una vez con la bola para que se rompan. Se gana un punto cuando se rompe el bloque (figura 6.6).



Figura 6.6. Bloques de una vida

- Bloques de dos vidas: hay que darles dos veces con la bola para que se rompan. Cada vez que se les da un golpe se gana un punto, y cuando se les da la primera vez cambian de color y se transforman en uno de los de una vida (figura 6.7).



Figura 6.7. Bloques de dos vidas

- Bloques de tres vidas: hay que darles tres veces con la bola para que se rompan. Cuando se le da por primera vez, cambian de color y se transforma en un bloque de dos vidas (figura 6.8).



Figura 6.8. Bloques de tres vidas

- Bloques irrompibles: estos bloques no se pueden romper, cada vez que la bola choca con ellos, simplemente rebota, como cuando choca con una pared. No dan puntos (figura 6.9).



Figura 6.9. Bloques irrompibles

6.5 Escena final

Cuando se pierde o se gana el juego, se llega a la escena final (figura 6.10), donde aparecen los puntos conseguidos. Existe la opción de escribir el nombre del jugador para que se guarde su puntuación, pulsando el teclado o fijando la mirada sobre la imagen del teclado. También se puede regresar al menú principal sin introducir el nombre.



Figura 6.10. Escena final

6.6 Teclado

Si se selecciona la opción de guardar la puntuación, aparecerá un teclado para escribir el nombre del jugador (figura 6.11).



Figura 6.11. Teclado

Una vez escrito el nombre, es necesario pulsar “Save”, lo que devuelve al jugador a la escena final, mostrada anteriormente. Si el usuario se ha equivocado y desea modificar su nombre, sólo tiene que volver a esta escena y volver a escribirlo. Cuando se selecciona la opción de “Back to menu” de la escena final se guarda la puntuación en el fichero. Por lo tanto, es posible modificar el nombre las veces que se desee mientras no se regrese al menú principal.

Capítulo 7.

Validación

Una vez finalizado el videojuego, se hizo un análisis de su usabilidad y se comprobaron ciertos parámetros como la dificultad y la calidad de la calibración.

Tras haber jugado varias veces, durante bastante rato, se obtuvieron los siguientes resultados:

- La calibración está programada de forma adecuada, aunque depende mucho del usuario. Si el usuario realiza la calibración de forma adecuada, es decir, no mueve la cabeza y mira a los puntos de la calibración cuando debe mirar, se obtiene una precisión bastante alta.
- El eye-tracker funciona correctamente a lo largo del juego, independientemente de la duración, aunque después de un cierto tiempo jugando (alrededor de 10 minutos) comienza a calentarse.
- En un principio, jugar con la vista es bastante complicado, principalmente debido a la falta de costumbre. Tras haber jugado varias partidas (unas 5), el usuario se empieza a adaptar y consigue mejores resultados.
- Las primeras veces que se juega, la vista se cansa rápidamente, pero a medida que se va jugando, se va adaptando y se cansa cada vez menos. Hacen falta muchas horas de juego para que la vista tarde en cansarse.
- Escribir con el teclado es un poco complicado las primeras veces, principalmente debido a la falta de costumbre, pero si la calibración se ha realizado de forma correcta, se puede escribir con relativa facilidad con la práctica. Sin embargo, si la calibración no se realizó adecuadamente o el usuario ha movido la cabeza a lo largo de la partida, escribir puede llegar a ser una tarea complicada.

Capítulo 8.

Conclusiones y líneas futuras

Este capítulo recoge una breve reflexión sobre el trabajo realizado y cómo podría seguir mejorándose en el futuro.

8.1 Conclusiones

Mediante la realización de este proyecto se ha conseguido crear un videojuego entretenido y relativamente sencillo de jugar, que puede ser jugado tanto con el ratón como con la vista.

El producto final desarrollado cumple con todos los objetivos planteados inicialmente en el apartado 3, e incluso se ha podido añadir alguna funcionalidad extra, como la posibilidad de escribir el nombre y guardar las puntuaciones.

En lo que respecta a la obtención de conocimientos que se han experimentado con las tecnologías y herramientas descritas en el capítulo 4, cabe destacar todo lo aprendido con respecto a Unity, ya que la autora de este TFG no poseía conocimientos previos sobre esta herramienta.

Por otra parte, enfrentarse a la planificación de este proyecto, su desarrollo y su puesta en marcha junto con el conjunto de decisiones tomadas que influyeron directamente sobre el mismo, ha sido una de las experiencias más valiosas de este proyecto.

8.2 Líneas futuras

A pesar de que la aplicación, una vez finalizado su desarrollo, cumple con los requisitos establecidos al inicio del proyecto, es cierto que pueden añadirse múltiples funcionalidades que pueden convertirlo en un juego más completo.

Una mejora a aplicar en un futuro sería la de modificar el juego para que pueda ser utilizado con otros eye-trackers, que ofrezcan una mayor precisión.

También puede ser interesante añadir nuevas funcionalidades como sonido o la posibilidad de añadir más bolas a la partida.

Otra posible mejora sería permitir la modificación los parámetros del juego, como son, por ejemplo, velocidad de la bola, de la barra o el número de vidas.

Una última mejora, para la línea de investigación desarrollada por el equipo médico colaborador, sería la de implementar nuevos juegos. Esto es importante ya que el usuario va a pasar muchas horas jugando, por ello, cuanta más variedad de juegos tenga, menos se aburrirá.

Capítulo 9.

Summary and Conclusions

This chapter offers a brief summary of the developed work and how it could be improved further in future.

9.1 Conclusions

In this project a fun and relatively simple game, that can be played with either the mouse or the gaze, has been implemented.

The final game accomplishes with all the objectives initially suggested in chapter 3 and even includes some extra functionalities, such as the ability to write the name and save the scores.

Regarding the knowledge acquisition that the student has experimented about the technologies and tools described in chapter 4, Unity can be highlighted, due to the lack of previous knowledge about this tool.

In addition, one of the most valuable experiences of this project has been to deal with the project planning, development and implementation, and with the set of decisions that influenced this process.

9.2 Future work

Although the implemented application accomplishes with the established requirements at the beginning of the project, multiple features could be added to complete it.

A future improvement would be the adaptation of the game to other more accurate eye-tracker devices.

It could be also interesting to add new functionalities, such as sound, or the possibility to play with more balls during the game.

Another possible improvement would be to allow the modification of the game parameters such as, the ball and bar speed or the number of lives.

A final improvement for the research line developed by the medical staff of this project would be the implementation of new games. This fact is important because the users will spend many hours playing, therefore, the more variety of games they have available, the less they get bored.

Capítulo 10.

Presupuesto

En este capítulo se hace una estimación del presupuesto del TFG. Teniendo en cuenta que el precio por hora de trabajo es de 20€, la tabla de presupuesto es la siguiente:

	Precio
Eye-tracker	99€
Horas de trabajo	$307 * 20€ = 6140€$
Total	6239€

Tabla 10.1. Tabla de presupuesto.

Apéndice A.

Funciones principales

A.1. Controlador del personaje

```
/******  
*  
* Bar.css  
*  
*****  
*  
* DESCRIPCION: Este código se encarga de calcular la posición a la que se tiene que  
* mover la barra, tanto si se juega con el ratón o con la vista. Además, en caso de  
* que se juegue con la vista, se guardarán las posiciones x e y de los ojos, si esta  
* opción fue seleccionada.  
*  
*****  
*  
*Point2D gazeCoords=GazeDataValidator.Instance.GetLastValidSmoothedGazeCoordinates();  
* Vector3 mouse;  
* if (!Global.useMouse) {  
*   Point2D gp = UnityGazeUtils.GetGazeCoordsToUnityWindowCoords (gazeCoords);  
*   mouse = new Vector3 ((float)gp.X, (float)gp.Y, CamaraPosition.z);  
*   mouse = Camera.main.ScreenToWorldPoint(mouse);  
*     if (Global.saveData == true) {  
*       SaveMedicalData.instance.Save ((float)gp.X, (float)gp.Y);  
*     }  
* else {  
*   mouse = Camera.main.ScreenToWorldPoint(Input.mousePosition);  
*}  
*  
*****/
```

Apéndice B.

Código completo

B.1. Repositorio de GitHub

Todo el código desarrollado se encuentra alojado en un repositorio de GitHub. A continuación, se proporciona el enlace al dicho repositorio:

<https://github.com/RebeccaMartinez/BreakOut>

Para poder abrir el juego con Unity únicamente hay que descargarse el repositorio, abrir Unity, seleccionar la opción de abrir proyecto existente y seleccionar el repositorio descargado.

Bibliografía

- [1] C. Modroño, J. Plata-Bello, F. Zelaya, S. García, I. Galván, F. Marcano, G. Navarrete, O. Casanova, M.Mas and J.L. González-Mora. Enhancing Sensorimotor Activity by Controlling Virtual Objects with Gaze. *Plos One*, 10(3): e0121562. doi:10.1371/journal.pone.0121562, 2015.
- [2] City Salud: <https://www.citysalud.es/>
- [3] Teen City: http://www.dip-alicante.es/iter/teencity/teen_city_es.html
- [4] Re-Mission: <http://www.re-mission.net/>
- [5] Pamela M. Kato, Steve W. Cole, Andrew S. Bradlyn, Brad H. Pollock. A Video Game Improves Behavioral Outcomes in Adolescents and Young Adults With Cancer: A Randomized Trial. *Pediatrics*. Volume 122, 2008. <http://pediatrics.aappublications.org/content/122/2/e305>
- [6] Virtual Rehab: <http://www.virtualrehab.info/>
- [7] Unreal Engine: <https://www.unrealengine.com/>
- [8] Unity: <https://unity3d.com/es>
- [9] Cry Engine: <https://www.cryengine.com/>
- [10] Gamemaker: Studio: <http://www.yoyogames.com/gamemaker>
- [11] Blender: <https://www.blender.org/>
- [12] The Eye Tribe: <https://theeyetribe.com/>
- [13] C#: https://es.wikipedia.org/wiki/C_Sharp
- [14] MonoDevelop: <https://es.wikipedia.org/wiki/MonoDevelop>
- [15] Git: <https://git-scm.com/>
- [16] Github: <https://github.com/>
- [17] Space Invaders: https://es.wikipedia.org/wiki/Space_Invaders
- [18] Break Out: [https://es.wikipedia.org/wiki/Breakout_\(videojuego\)](https://es.wikipedia.org/wiki/Breakout_(videojuego))
- [19] Temple Run: https://en.wikipedia.org/wiki/Temple_Run
- [20] Snowball Slalom: http://www.mariowiki.com/Snowball_Slalom

[21] Danger bob-omb danger: http://www.mariowiki.com/Danger,_Bob-omb!_Danger!