

ULL

Universidad  
de La Laguna

Escuela Superior de  
Ingeniería y Tecnología

# Trabajo de Fin de Grado

## Representación virtual del cerebro humano

*Virtual representation of the human brain*

Erik Andreas Barreto de Vera

La Laguna, 4 de junio de 2018

D. **Rafael Arnay del Arco**, con N.I.F. 78.569.591-G profesor ayudante doctor adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor.

D. **Niels Janssen**, con N.I.F. XY0263431A profesor contratado Doctor tipo 1 adscrito al Departamento de Psicología Cognitiva, Social y Organizacional de la Universidad de La Laguna, como cotutor

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

Que la presente memoria titulada:

*“Representación virtual del cerebro humano”*

ha sido realizada bajo su dirección por D. **Erik Andreas Barreto de Vera**, con N.I.F. 54.113.602-F.

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 2018

# Agradecimientos

En este camino recorrido hasta la realización de este informe de trabajo de fin de grado debo agradecer la voluntariedad de todas aquellas personas (familiares, profesores de mi colegio, instituto y docentes de la Universidad de La Laguna) que a lo largo de los años han aportado su trabajo y dedicación para formarme como estudiante y como persona.

Por último y no menos importante, quisiera agradecer por su experiencia, consejos, ayuda y por todo el soporte durante el proceso a mi tutor Rafael Arnay del Arco y mi cotutor Niels Janssen, además de darme la oportunidad de hacer un trabajo de fin de grado tan educativo en el que logramos que la tecnología de la realidad virtual fuera una vía alternativa y viable de estudio del cerebro para los alumnos de universidades.

# Licencia



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

## Resumen

*El objetivo de este proyecto es proporcionar a alumnos de enseñanzas de materias de Medicina y Neuro-Psicología una vía alternativa de visualización del cerebro humano a través de la tecnología de realidad virtual, creando para este propósito una aplicación móvil multiplataforma que funciona con gafas de realidad virtual tipo Google CardBoard.*

*Actualmente, los estudiantes de estas materias recurren a métodos de estudio clásicos como libros de texto, atlas, apuntes a color, etc. Y a herramientas más modernas como simuladores 3D [1], sin embargo, la aplicación desarrollada pretende aportar una forma unificada donde visualizar el cerebro, los estudiantes podrán ver un cerebro tridimensional de grandes proporciones con el que pueden interactuar a través de una serie de herramientas que dispone la misma aplicación.*

*Además, el tiempo de aprendizaje de estas materias puede verse acelerado y por tanto es una mejora en la enseñanza tanto para profesores como para estudiantes.*

**Palabras clave:** Unity, Realidad virtual, motor gráfico, Google CardBoard, SDK, IOS, Android, Programación orientada a objetos, C#.

## **Abstract**

*The objective of this project is to provide students of the subjects of Medicine and Neuro-Psychology with an alternative way of visualizing the human brain through virtual reality technology, creating for this purpose a multiplatform mobile application that works with reality glasses virtual type Google CardBoard.*

*Currently, students of these subjects resort to classical study methods such as textbooks, atlases, color notes, etc. And to more modern tools such as 3D simulators [1]. However, the application developed aims to provide a unified form where visualize the brain, students can see a three-dimensional brain of great proportions with which they can interact through a series of tools available in the same application.*

*In addition, the learning time of these subjects can be accelerated and therefore is an improvement in teaching for both teachers and students.*

**Keywords: Unity, Virtual reality, Game engine, Google CardBoard, SDK, IOS, Android, Object-oriented Programming, C#**

# Índice general

<b>Capítulo 1</b>	<b>Introducción .....</b>	<b>1</b>
1.1	Tecnologías en el ámbito educativo.....	1
1.2	Tecnologías de representación del cuerpo humano.....	2
1.3	Antecedentes y estado actual del tema .....	7
1.4	Proyectos similares .....	8
1.5	Motivación del proyecto .....	9
1.6	Objetivo general.....	10
1.7	Objetivos específicos .....	10
1.8	Plan de trabajo .....	11
1.9	Estructura de la memoria.....	12
<b>Capítulo 2</b>	<b>Herramientas, tecnologías y recursos .....</b>	<b>14</b>
2.1	Unity.....	14
2.2	Google CardBoard .....	15
2.3	Brainder .....	17
<b>Capítulo 3</b>	<b>Desarrollo del proyecto.....</b>	<b>18</b>
3.1	Configuración del proyecto.....	18
3.2	Primera aproximación de herramientas y opciones de la interfaz.....	22
3.3	Diseño de la interfaz .....	23
3.4	Implementación de la interfaz .....	25
<b>Capítulo 4</b>	<b>Problemas y soluciones .....</b>	<b>32</b>
<b>Capítulo 5</b>	<b>Conclusiones y líneas futuras .....</b>	<b>35</b>

5.1	Conclusiones .....	35
5.2	Líneas futuras .....	36
<b>Capítulo 6</b>	<b>Summary and Conclusions.....</b>	<b>38</b>
<b>Capítulo 7</b>	<b>Presupuesto .....</b>	<b>39</b>
7.1	Presupuesto.....	39

# Índice de figuras

Figura 1. Dibujo en detalle tórax de ilustración de Leonardo Da Vinci. ....	2
Figura 2. Imágenes capturadas por un microscopio electrónico de transmisión. .....	3
Figura 3. Radiografía tomada por Wilhelm Conrado en 1896. ....	3
Figura 4. Variaciones de ondas magnéticas que se transforman en imágenes. .	4
Figura 5. Atlas de la anatomía humana [6] .....	5
Figura 6. Atlas de la anatomía humana [7] .....	5
Figura 7. Interfaz de la página web [8] .....	6
Figura 8. Cerebro en realidad virtual de la aplicación BrainVR. ....	7
Figura 9. Unity logo. ....	14
Figura 10. InCell VR game .....	15
Figura 11. Componentes necesarios para fabricar unas Google CardBoard. .	16
Figura 12. Google CardBoard con smartphone. ....	16
Figura 13. Instalador de Unity en Mac OS X. ....	18
Figura 14. Prefabs necesarios para ejecutar una escena en realidad virtual. .	20
Figura 15. Cerebro importado en Unity. ....	22
Figura 16. Jerarquía del menú desplegable. ....	24
Figura 17. Configuración de un elemento UI que interacciona con el usuario. .....	25
Figura 18. Resultado del diseño de la interfaz. ....	25
Figura 19. Canvas con la información de la zona seleccionada. ....	27
Figura 20. Modelo de las zonas subcorticales. ....	27
Figura 21. Opciones de manipulación y pintado del cerebro. ....	28

Figura 22. Opciones de pintado. ....	29
Figura 23. Zona del cerebro después de ocultar el cerebro restante. ....	30
Figura 24. Modo a pantalla completa. ....	31
Figura 27. Configuración de una fuente para evitar el pixelado y aumentar la calidad. ....	33
Figura 28. Aspecto de la retícula sin colisión. ....	34
Figura 29. Aspecto de la retícula con colisión más cuenta atrás de interacción. ....	34
Figura 30. Inicio de la aplicación. ....	45
Figura 31. Áreas del cerebro. ....	46
Figura 32. Lista de zonas pertenecientes al área temporal. ....	47
Figura 33. Cerebro con el área frontal pintada de color azul. ....	49

# Índice de tablas

Tabla 2-1. Comparativa de precios de gafas de realidad virtual aproximada.	17
Tabla 7-1. Presupuesto y horas invertidas.....	40

# Capítulo 1

## Introducción

### 1.1 Tecnologías en el ámbito educativo

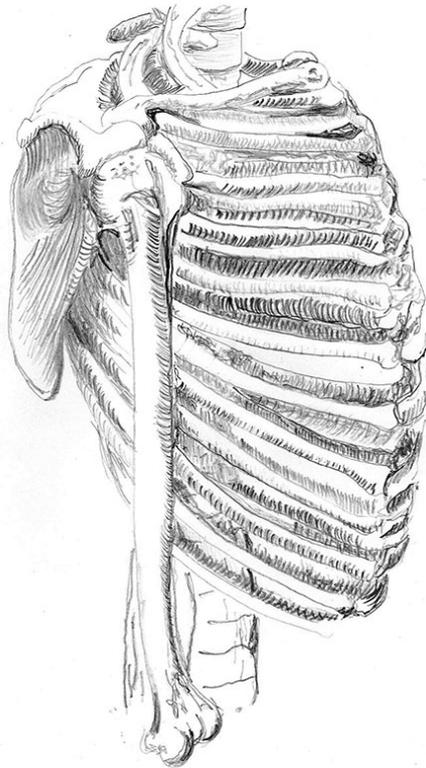
En la actualidad, la mayoría de los jóvenes estudiantes se caracterizan por haber nacido rodeado de productos digitales, por este motivo, se debe pensar en la importancia de un uso adecuado de todas las nuevas tecnologías que se encuentran a su alcance. El hecho de incorporar la tecnología en la educación abre una puerta de ventajas para los estudiantes que se deben aprovechar. Estos jóvenes son capaces de mejorar la eficiencia y la productividad en el aula a través de éstas al mismo tiempo que pueden incrementar su interés por las actividades académicas que realizan [2].

Las ventajas y beneficios que aportan en los procesos educativos se basan en estudios [3] que aseguran que el uso de tecnologías aporta beneficios en el rendimiento en escuelas.

Afortunadamente, muchos ámbitos y carreras profesionales se han beneficiado de estas tecnologías. Hoy en día vemos hospitales llenos de aparatos tecnológicos, programas de gestión de personal, recursos, etc. Pero desafortunadamente, en el ámbito educativo no vemos la adaptación con la misma rapidez. Para ello, este trabajo de fin de grado pretende aportar una aplicación móvil de la que muchos estudiantes pueden sacar beneficio.

## 1.2 Tecnologías de representación del cuerpo humano

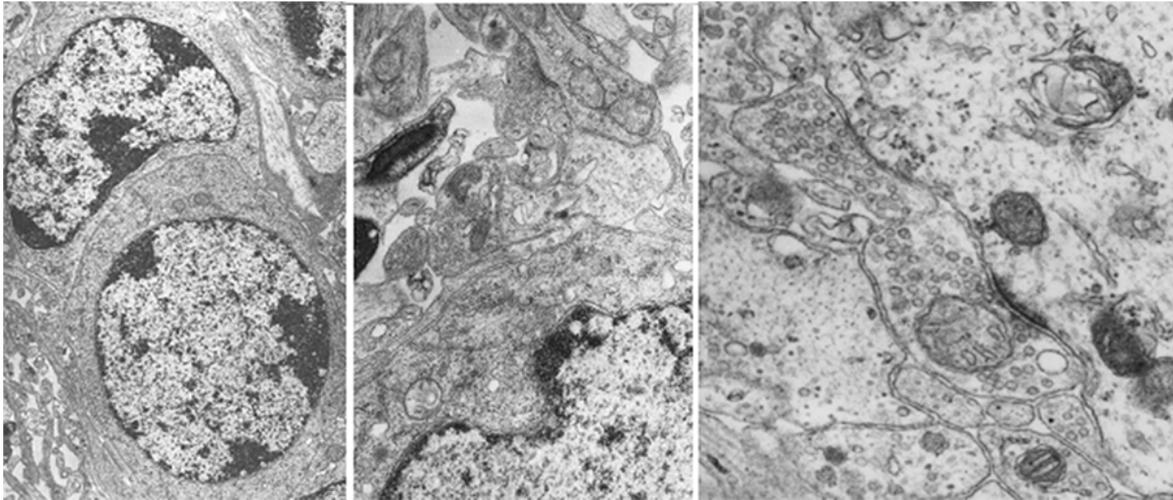
Desde hace mucho tiempo, el interés del ser humano por el conocimiento de su propio cuerpo es innegable. En sus inicios, los únicos métodos para poder comprender la estructura y funcionamiento del cuerpo eran mediante la disección. Vemos a continuación algunos bocetos:



*Figura 1. Dibujo en detalle tórax de ilustración de Leonardo Da Vinci.*

A medida que la tecnología y los conocimientos fueron mejorando, llegaron nuevas herramientas [4]. En 1931, se inventó el primer **microscopio electrónico de transmisión (T.E.M)**, el cual utiliza un haz de electrones en lugar de una luz, lo que logra conseguir un aumento considerable. Este microscopio fue desarrollado por Max Knoll y Ernst Ruska en Alemania. Años después, en 1942, se desarrolló el primer microscopio electrónico de barrido

(SEM).



*Figura 2. Imágenes capturadas por un microscopio electrónico de transmisión.*

Los **rayos X** fueron descubiertos, en primera instancia por el científico William Crookes, en el siglo XIX. Al igual que el microscopio, los rayos X ayudaron a la comprensión de la estructura y la composición del organismo, pero esta vez no se presenta un instrumento para detectar células y bacterias, sino para obtener imágenes internas de tejidos, huesos y órganos del cuerpo humano.

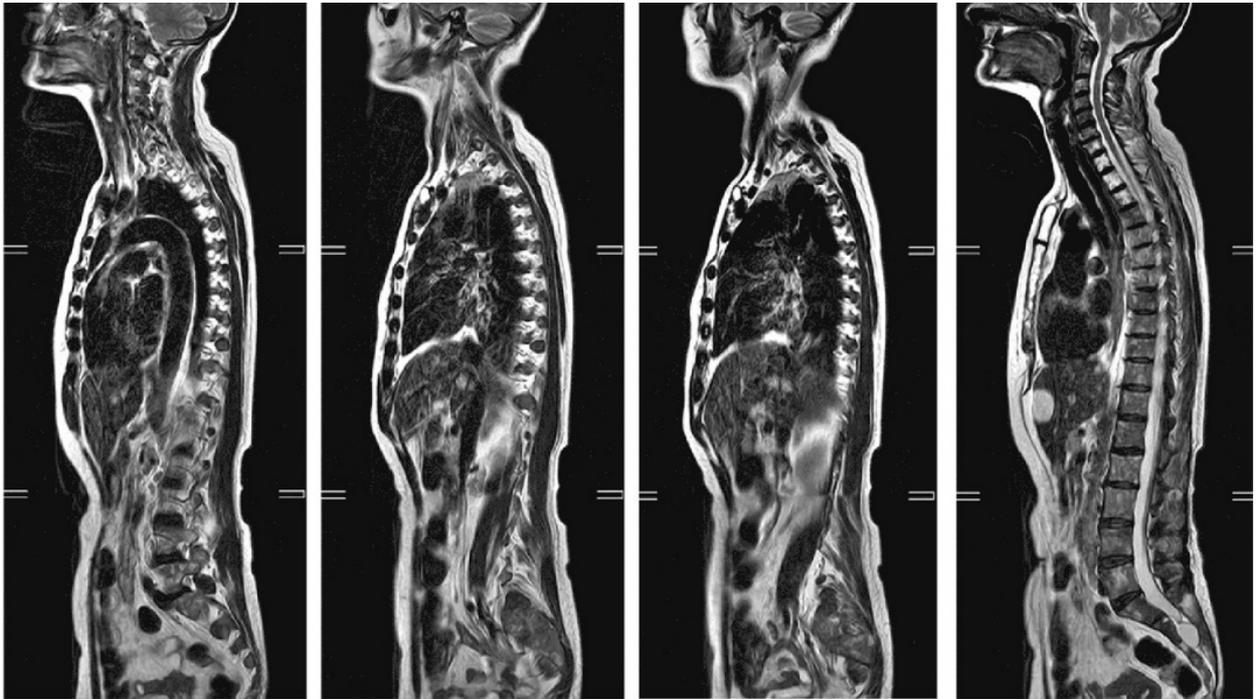
De esta manera, un especialista puede constatar lesiones internas en los órganos y huesos, también se utilizaron para descubrir tumores cancerígenos.



*Figura 3. Radiografía tomada por Wilhelm Conrado en 1896.*

Por otra parte, posterior a los rayos X, se implementó la **resonancia magnética**. Los estudios con imágenes por resonancia magnética (IRM) usan

un gran imán y ondas de radio para observar órganos y estructuras que se encuentran en el interior del cuerpo. Los profesionales de la salud utilizan estas imágenes para diagnosticar una variedad de afecciones, desde rupturas de ligamentos hasta tumores. Las imágenes por resonancia magnética son muy útiles para examinar el cerebro y la médula espinal [5].



*Figura 4. Variaciones de ondas magnéticas que se transforman en imágenes.*

Por último, antes de llegar a formas más complejas de representación, se desarrollaron **ilustraciones** para el entendimiento del cuerpo humano, entre ellas encontramos dos atlas (Sobotta y Netter).

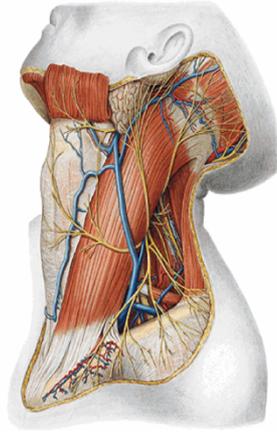


Figura 5. Atlas de la anatomía humana [6]

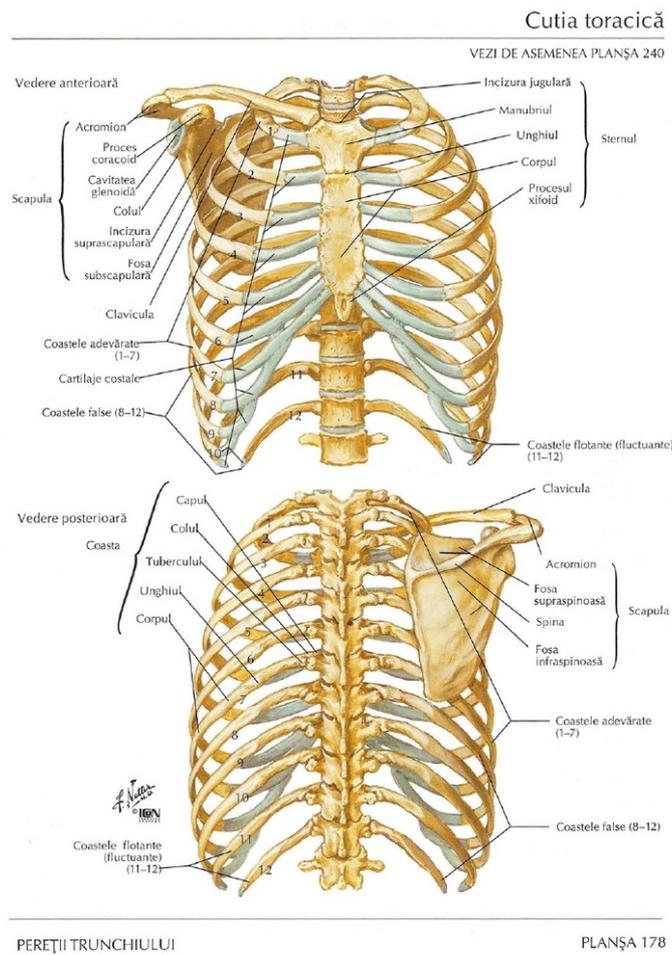


Figura 6. Atlas de la anatomía humana [7]

Contextualizado en tiempos actuales, donde gran cantidad de la información se transmite por internet, se presenta a continuación un sitio web [8] donde se muestran las partes del cuerpo humano dividido primeramente en los distintos sistemas, luego se va adentrando a partes más interiores del cuerpo.

El sitio web [8] presenta imágenes en 3D por lo que se pueden hacer vistas en 360° de la totalidad o zonas específicas del cuerpo. Además, el usuario puede seleccionar distintas partes que se van señalando en el modelo y en algunos casos aparecen definiciones y explicaciones del funcionamiento.

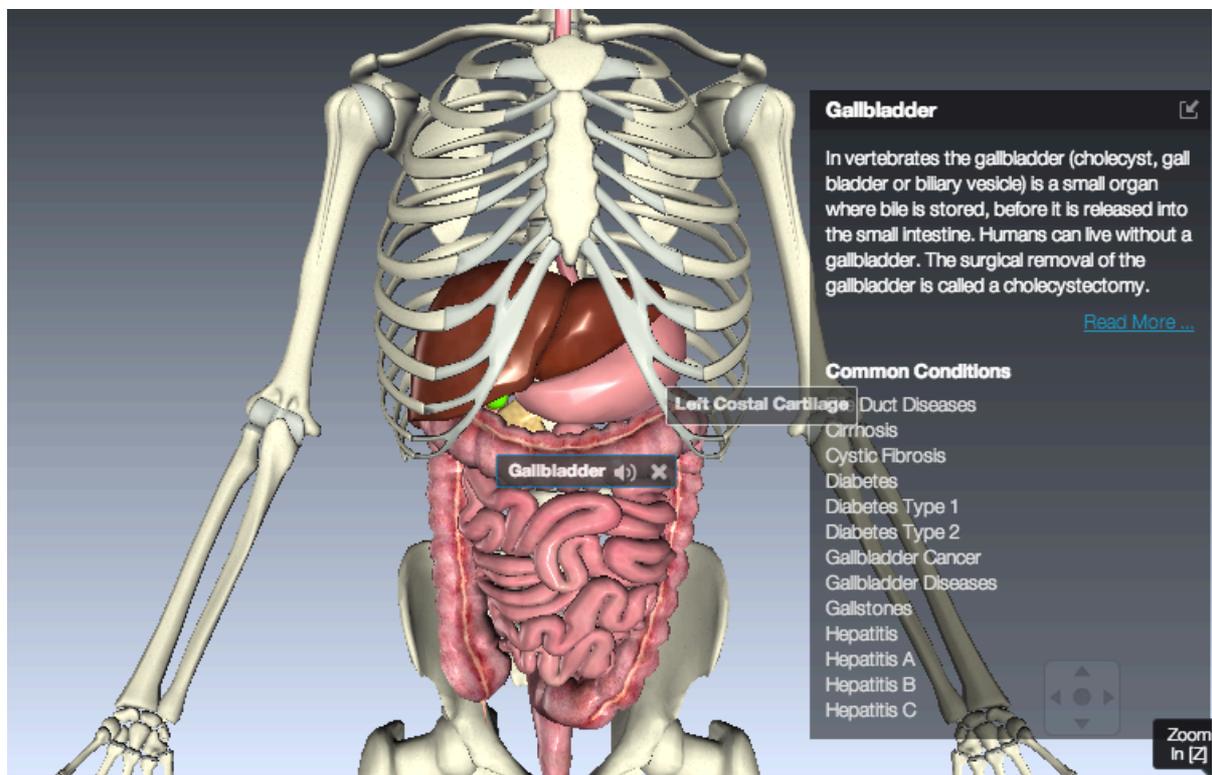
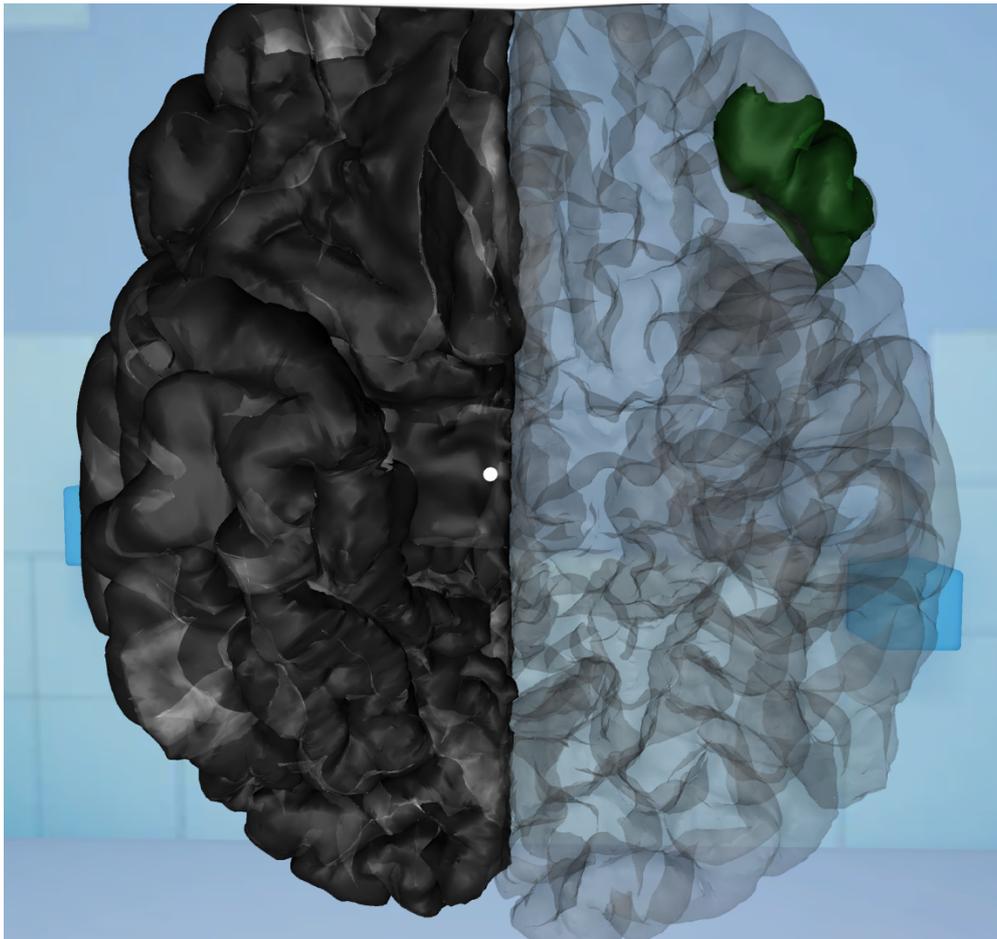


Figura 7. Interfaz de la página web [8]

Para finalizar, la aplicación BrainVR pretende ser el paso a un nuevo salto tecnológico, aprovechando las nuevas capacidades computacionales de los *smartphones* y su potencia para poder visualizar complejos modelos tridimensionales. Veamos a continuación una pequeña muestra de un modelo de un cerebro en realidad virtual de la aplicación:



*Figura 8. Cerebro en realidad virtual de la aplicación BrainVR.*

### **1.3 Antecedentes y estado actual del tema**

El sector de ocio y entretenimiento parecen ser los únicos que se han aprovechado de la realidad virtual, pero lo cierto es que está presente en diferentes disciplinas, como por ejemplo la medicina. De hecho, se está abriendo muchas posibilidades a los profesionales del sector como la formación de estudiantes, operaciones de cirugía o tratamiento de enfermedades.

En cuanto a la formación [2], este sistema permite adquirir habilidades manuales a los estudiantes, reconocer la estructura de los órganos y también,

aprender a utilizar diferentes equipos sin necesidad de órganos reales, por ejemplo.

También puede adaptarse al campo de la cirugía [9]. Con ayuda de programadores y modeladores 3D puede reproducirse con exactitud la anatomía del paciente, con la intención de practicar la operación antes de realizarla, se podría repetir el entrenamiento tantas veces como se deseará reduciendo el riesgo de la operación.

En la actualidad, aparecen nuevos proyectos desarrollados específicamente a enfermedades concretas, patologías, fobias o rehabilitaciones cada poco tiempo. Un ejemplo es, Sea Hero Quest VR [10], un videojuego de realidad virtual especializado en detectar el Alzheimer [11], el objetivo del usuario es navegar por un lago con paredes de hielo o rastrear un pantano buscando un monstruo. Sin embargo, detrás de estos desafíos se esconde una prueba que ayuda a los investigadores en el diagnóstico de los primeros indicios de estas enfermedades neurológicas gracias a que esconden tareas de memorización y orientación.

Cabe destacar, que no sólo se está aplicando la realidad virtual al mundo de la medicina, sino también en otros ámbitos, como la rehabilitación deportiva. Investigadores de la Universidad Carlos III de Madrid (UC3M) [11] han desarrollado un sistema de realidad virtual para la rehabilitación motora del hombro. El prototipo, que integra un sensor de movimiento, permite realizar ejercicios controlados en un entorno de juego de fútbol.

## 1.4 Proyectos similares

**Brain VR** [12] es una aplicación de Google Cardboard que estaba disponible en la google store destinada a ver un modelo del cerebro humano. Es

necesario un mando bluetooth o gamepad para navegar alrededor y dentro del cerebro. Actualmente no se encuentra en la tienda, pero se puede encontrar el código en Github.

3D Organon VR Anatomy [14] es el primer atlas de anatomía de realidad virtual completamente equipado del mundo. Es una experiencia inmersiva de autodescubrimiento en el cuerpo humano.

Por último, Polygon medical animation [15] realiza pequeñas experiencias que duran unos pocos minutos, son viajes guiados dentro del cuerpo humano en formato de video a través de Youtube o Facebook. Algunos ejemplos de su trabajo son 360° VR – Inside your amazing cells [16] y 360° Brain virtual reality [17].

## 1.5 Motivación del proyecto

La motivación del proyecto nace de la necesidad de mejorar las herramientas que tienen los estudiantes y profesores a su disposición. El propósito principal es adaptar la realidad virtual a la enseñanza, más en concreto a materias de Medicina y Neuro-Psicología.

Aunque todavía es una tecnología emergente, en relación con su potencial, ya ofrece experiencias a los usuarios que no podíamos lograr con las herramientas de estudio clásicas. Permite a los alumnos experimentar el aprendizaje en escenarios inmersivos muy cercanos a la realidad y con mucho detalle.

Gracias a la realidad virtual, un alumno puede vivir la experiencia de moverse por el cuerpo humano libremente explorando órganos o descubriendo su funcionamiento a través del aprendizaje inmersivo.

## 1.6 Objetivo general

El objetivo general es desarrollar una aplicación móvil multiplataforma (IOS/Android) usando uno de los mejores motores de videojuegos, Unity3D [13], con el fin de proveer una fuente alternativa de enseñanza dirigida específicamente para alumnos de materias de Medicina y Neuro-psicología, de forma que a los alumnos les resulte más fácil comprender la estructura del cerebro.

## 1.7 Objetivos específicos

Para la realización de la aplicación se han desarrollado los siguientes objetivos.

- Aprender el uso de un motor de videojuegos (Unity3D) [13] para el desarrollo de una aplicación móvil multiplataforma.
- Aprender el funcionamiento y uso del SDK de Google [14] para las gafas de realidad virtual tipo Google CardBoard [15]. Hacer uso del SDK en el motor de Unity3D.
- Entender todos los componentes que forman parte de un motor de videojuegos (iluminación, físicas, ...) [16].
- Realizar el diseño de la interfaz de usuario de la aplicación. Incluye el modelaje de la escena, menús, herramientas de los menús, creación de los elementos gráficos (botones, lienzos, paneles, etc.) así como la mejor distribución de estos siguiendo unos patrones de diseño con la finalidad de que la interfaz sea intuitiva y con un diseño simple a la vez que bonito.

- Implementar los *scripts* en C# relacionados con la interfaz de usuario (navegación de los menús, interacción con elementos de la escena y el entorno...) y con la manipulación del cerebro (rotación, aparición y ocultación de determinadas áreas, visualización por zonas, ...).
- Implementar *scripts* en C# para realizar acciones a través de un mando bluetooth tipo gamepad conectado al *smartphone*.
- Buscar, seleccionar, comprender y añadir la información necesaria para cada una de las áreas y zonas del cerebro.
- Hacer tests de usuario para corregir la interfaz de usuario con el feedback de los usuarios.

## 1.8 Plan de trabajo

- **Tarea 0 - Planificación.** La primera tarea es la planificación inicial de todo el trabajo, estableciendo reuniones periódicas entre el tutor, cotutor y alumno para sentar las bases del proyecto.

Para ello se establece un calendario de reuniones, para cada una se lleva un seguimiento del trabajo realizado hasta ese punto y solucionar posibles problemas que hayan surgido, aclarar dudas que se pudiesen encontrar o incluso pensar en alguna modificación que pueda ser útil.

- **Tarea 1 - Formación.** En esta etapa es necesario adquirir una serie de conocimientos básicos sobre motores gráficos de videojuegos, hay que entender el funcionamiento de prácticamente todos los componentes que

forman parte de cualquier videojuego como pueden ser, las físicas, *scripts*, gráficos, iluminación, etc. Básicamente es un período de aprendizaje y adaptación a la herramienta de Unity y el uso del SDK de Google CardBoard.

- **Tarea 2. Configuración del proyecto.** Antes de arrancar las etapas posteriores es necesario ajustar una serie de opciones y recursos para poder ejecutar una aplicación en realidad virtual tanto para la plataforma de Android como para la plataforma de IOS. Esta configuración inicial se ve detallada en el siguiente capítulo.
- **Tarea 3. Diseño.** Una vez familiarizado con Unity y configurado los ajustes básicos, se procede al diseño de la escena y la interfaz de usuario. En primer lugar, se realiza un boceto de una interfaz lo más aproximada a la esperada que tenga la aplicación una vez terminada. En segundo lugar, se crea el escenario que rodea al usuario.
- **Tarea 4. Implementación.** Esta etapa comprende todo el desarrollo funcional de la aplicación, en ella se implementan todos los *scripts* que controlan la interacción entre el usuario y la interfaz, manipulación del cerebro (rotación, pintado, movimiento, etc.), gestión del escenario, etc. Hay que tomar en cuenta que, para poder ejecutar esta tarea, es necesario conocimiento básicos de programación orientada a objetos y de lenguaje C#.
- **Tarea 6. Documentación.** Para terminar, se procede a la documentación del código, de los resultados obtenidos y de la elaboración del informe.

## 1.9 Estructura de la memoria

- En el Capítulo 2 se describen las herramientas y tecnologías utilizadas en el

proyecto.

- En el Capítulo 3 se explican los diversos pasos realizados para la consecución del proyecto.
- En los Capítulos 4 y 5 tratan las conclusiones del proyecto y las futuras líneas de desarrollo.
- En el Capítulo 6 se muestra el presupuesto estimado, con precios de mercado actuales.

# Capítulo 2

## Herramientas, tecnologías y recursos

### 2.1 Unity

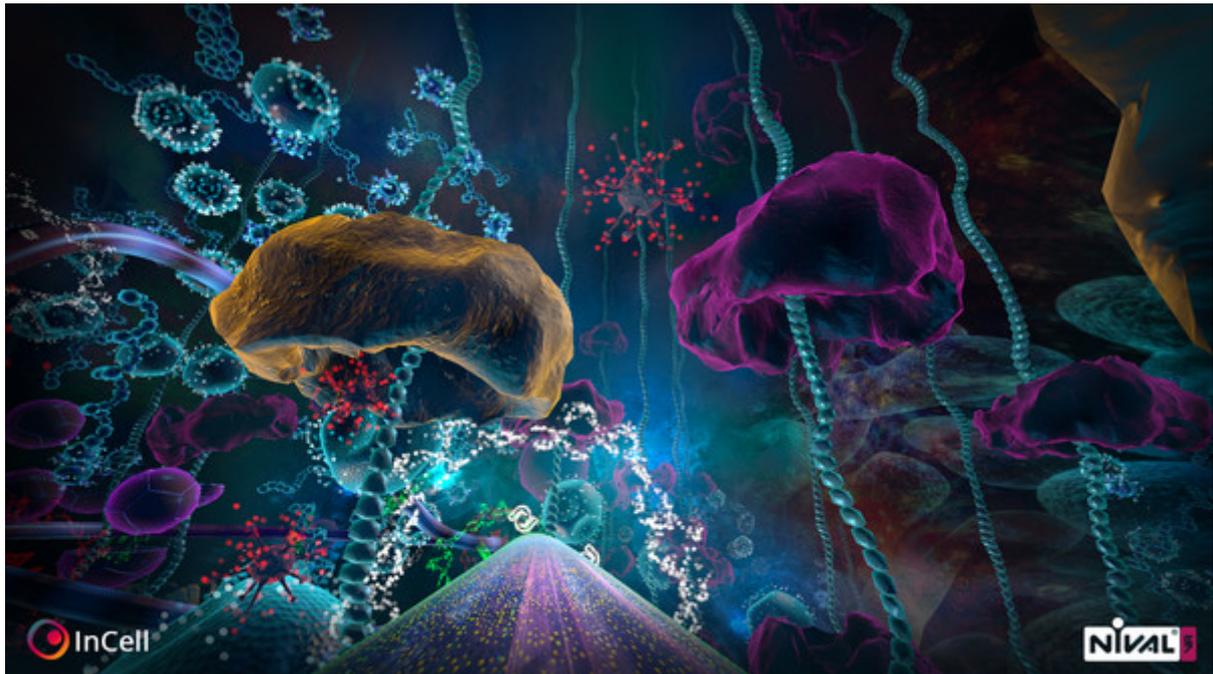
Es un motor de Videojuegos dedicado a la creación de contenido digital para más de 25 plataformas entre móvil, escritorio, consola, televisión, realidad aumentada, realidad virtual, y web. Unity es la herramienta de desarrollo escogida por la mayoría de los creadores de XR (Extended reality).

Algunas de sus características [17] que lo convierten en una magnífica elección están su pipeline de renderizado altamente optimizado de Unity y las capacidades de interacción rápida hacen de este motor una de las mejores elecciones para contenido de realidad virtual. Además, el hecho de ser multiplataforma con soporte nativo hace que llegue al mayor público posible. Por último, posee un versátil API [18] destinada a realidad virtual y realidad aumentada.



*Figura 9. Unity logo.*

Un ejemplo de juego educativo que podemos encontrar en la tienda de Unity es InCell [19]. Es un juego de realidad virtual / acción de carreras con un poco de estrategia. El jugador viaja dentro del micro mundo de las células humanas y para detener el avance del virus.



*Figura 10. InCell VR game*

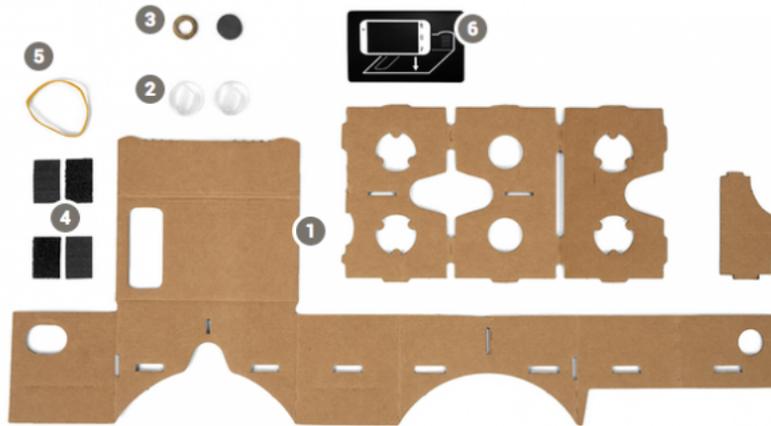
## 2.2 Google CardBoard

Es una solución propuesta por Google para aquellos usuarios que quieren vivir experiencias de realidad virtual, pero a un menor coste que el que tienen las grandes compañías como Oculus, HTC, etc. Es una herramienta ideal para los programadores y usuarios que desean adentrarse en el mundo. Google ha dado una solución casera para que cualquier persona pueda fabricar sus propias gafas a un bajo coste ínfimo.

Estas gafas están hechas de cartón e incluyen dos lentes de 40 mm de distancia focal y dos imanes en los laterales que permiten hacer click. Este efecto se logra gracias al magnetómetro del teléfono, al accionar los imanes, detecta un cambio en el campo magnético, reconociendo de este modo una acción por parte del usuario.

Cualquier usuario puede descargarse este kit a través de la web oficial [20]

junto con las plantillas, prácticas recomendadas y generadores de perfiles de visores correspondientes.



*Figura 11. Componentes necesarios para fabricar unas Google CardBoard.*



*Figura 12. Google CardBoard con smartphone.*

A continuación, veamos una comparativa de precios de las diferentes gafas del mercado para hacer una idea de lo que supone tener esta tecnología tan accesible.

	Tipo	Plataforma	Precio aproximado (menor a mayor)
Google CardBoard	Portátil	Mayoría de <i>smartphones</i>	7,99€ - 14,99€
Samsung Gear VR	Portátil	Mayoría de <i>smartphones</i>	99€
PlayStation VR	Con cable	PlayStation 4	299€
Oculus Rift	Con cable	PC	449€
HTC Vive	Con cable	PC	559€

Tabla 2-1. Comparativa de precios de gafas de realidad virtual aproximada.

Además de las gafas Google provee de un SDK específico [14] para la plataforma de Unity que contiene una serie de assets (*prefabs, scripts, etc.*) necesarios para poder convertir un proyecto en una aplicación de realidad virtual.

## 2.3 Brainder

Brainder es un sitio web [21] dedicado a la investigación de imágenes cerebrales. También se organizan las publicaciones y las charlas del autor, y donde se pueden compartir algunas piezas de software.

Para el desarrollo de esta aplicación se ha descargado un modelo de un cerebro humano real obtenido en el Research Imaging Institute del Centro de Ciencias de la Salud de la Universidad de Texas en San Antonio.

# Capítulo 3

## Desarrollo del proyecto

### 3.1 Configuración del proyecto

1. **Configuración del entorno de desarrollo.** En primer lugar, fue necesario instalar Unity siguiendo los pasos indicados en la página web [22]. Una vez ejecutado el instalador fue necesario marcar los componentes "*Android Build Support*" y "*IOS Build Support*" que sirven para dar soporte a las plataformas respectivas.

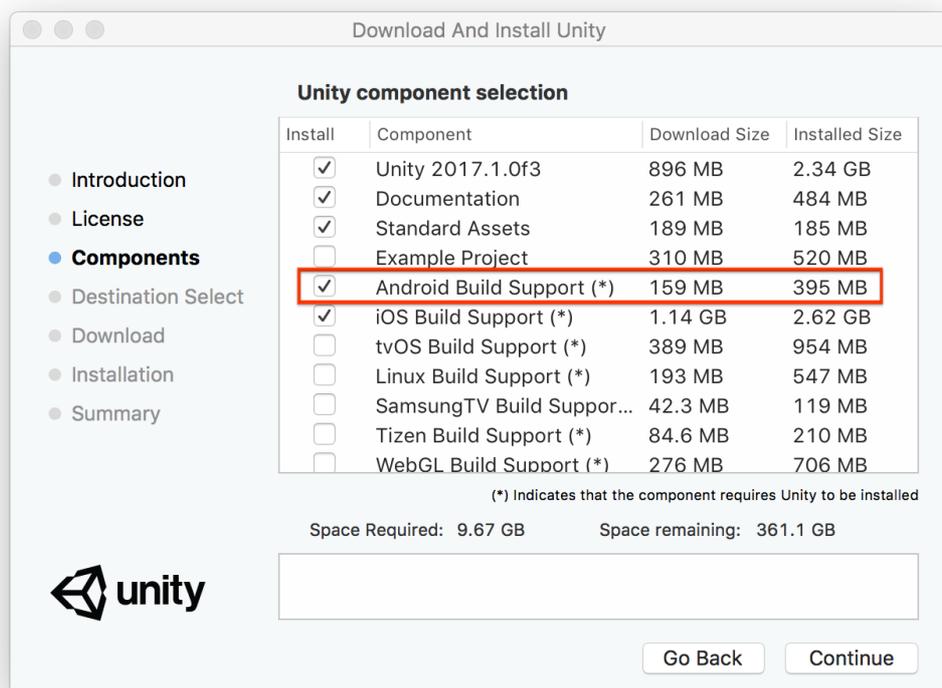


Figura 13. Instalador de Unity en Mac OS X.

2. Creación de un nuevo proyecto en Unity e importación del

**paquete de Google para Unity.** Una vez fue instalado Unity, se tuvo que descargar el SDK de Google para Unity desde el repositorio oficial de Google en Github (5). Acto seguido, fue necesario realizar los siguientes pasos para añadir el SDK. Los pasos fueron:

- Abrir Unity y crea un nuevo proyecto en 3D.
- Seleccionar **Assets > Import package > Custom package.**
- Seleccionar el archivo **GoogleVRForUnity\_\*.unitypackage** que se descargó.
- En el cuadro de diálogo Importar paquete, se hizo clic en **Import.**
- Se aceptó cualquier actualización de API, cuando se solicitó.

**3. Configuraciones de compilación y ajustes del jugador.** Antes de poder visualizar una escena en realidad virtual hace falta activar una serie de ajustes indispensables Veamos:

- **Configuración para Android.**
  - Se seleccionó File > Build Settings.
  - Se seleccionó **Android** y se hizo clic en **Switch platform.**
  - En la ventana **Build settings**, se hizo click en **Player settings.** Se Configuró lo siguiente:
  - **Player Settings > Other Settings > Virtual Reality Supported** -> enabled.
  - **Player Settings > Other Settings > SDKs** -> Click + and select Daydream or Cardboard.
  - **Player Settings > Android > XR Settings > Virtual Reality SDKs > Daydream > Positional Head Tracking** -> Click + and select Disabled, Supported, or Required.
  - **Player Settings > Other Settings > Minimum API Level** -> Daydream: Android 7.0 'Nougat' (API level 24) or higher. / Cardboard: Android 4.4 'Kit Kat' (API level 19) higher.

- **Configuración para IOS.**

- Se seleccionó **File > Build Settings.**
- Se seleccionó **IOS** y se hizo click en **Switch platform.**
- En la ventana **Build settings**, se hizo clic en **Player settings**. Se Configuró lo siguiente:
- **Player Settings > Other Settings > Virtual Reality Supported** -> enabled.
- **Player Settings > Other Settings > SDKs** -> Click + and select Cardboard.
- **Player Settings > Other Settings > Target minimum iOS version** -> 8.0 or higher
- **Player Settings > Other Settings > Bundle Identifier** -> Follow reverse the DNS format suggested by Apple.

4. **Vista previa de la escena en Unity.** Una vez se habían configurado todos los ajustes necesarios, se procedió a añadir los *prefabs* necesarios a la escena tal y como muestra la siguiente ilustración.

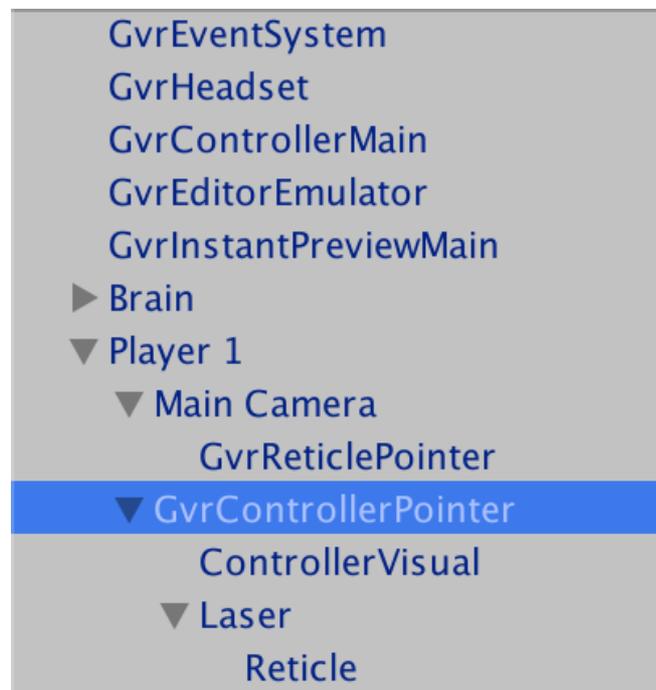
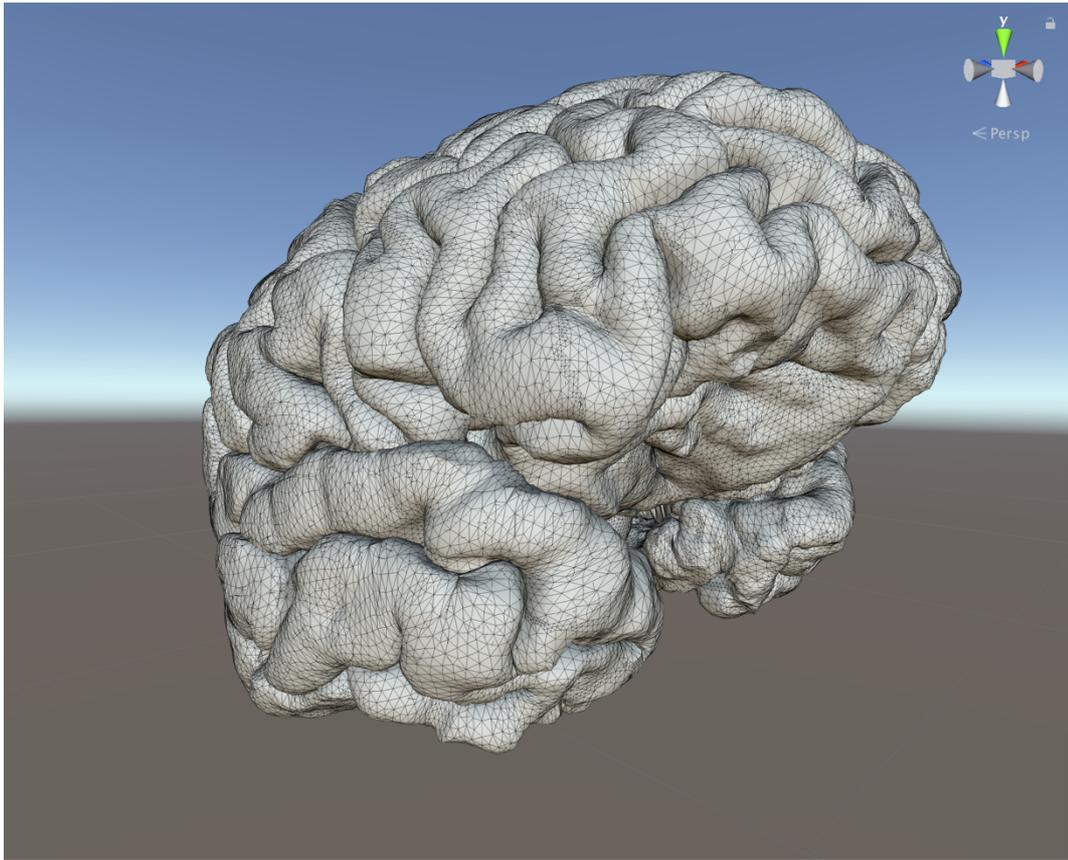


Figura 14. Prefabs necesarios para ejecutar una escena en realidad virtual.

Cabe destacar los cinco primeros *prefabs* encargados de:

Nombre del prefab	Localización en la escena	Descripción
GvrControllerMain	Cualquier lugar de la escena	Responsable de administrar el estado del controlador.
GvrEventSystem	Cualquier lugar de la escena	Permite al controlador Daydream usar el sistema de eventos Unity.
GvrEditorEmulator	Cualquier lugar de la escena	Permite simular el movimiento de la cabeza del usuario con su mouse o su panel táctil.
GvrInstantPreviewMain	Cualquier lugar de la escena	Permite transmitir una vista previa estéreo a su teléfono.
GvrControllerPointer	Junto a la cámara principal	Proporciona un controlador, láser y retícula.

Por último, se tuvo que añadir el modelo del cerebro descargado desde Brainder [23] e importarlo en Unity siguiendo el mismo procedimiento del SDK y se añadió a un *GameObject* de Unity vacío quedando la escena tal como vemos en la figura 15.



*Figura 15. Cerebro importado en Unity.*

## 3.2 Primera aproximación de herramientas y opciones de la interfaz

Fue necesario definir una serie de opciones y herramientas que iba a tener la aplicación antes de proceder al diseño de la interfaz. En primer lugar, se anotaron todas las opciones posibles, puesto que a medida que el proyecto avanzaba, surgían nuevas opciones que implementar.

A continuación, se muestra un resumen del objetivo de cada una de las opciones:

- **Select zone.** Su función principal es seleccionar una zona del modelo de la corteza cerebral o de las estructuras subcorticales. Cuando se realiza la primera interacción, despliega un listado con las áreas del cerebro y

cuando se realiza la segunda interacción se despliega el listado con las zonas pertenecientes al área de la primera interacción.

- **Paint tools.** Su función es colorear zonas y áreas del cerebro. Se permite al usuario poder colorear y reducir el grado de opacidad de zonas y áreas.
- **Manipulation tools.** Incluye las opciones que permiten rotar el cerebro en 360 grados mediante elementos visuales, a través de un gamepad conectado por bluetooth y de rotar alrededor basado en el movimiento de la cabeza.

### 3.3 Diseño de la interfaz

Para el desarrollo de una interfaz en Unity hacen falta dos elementos. Por un lado, está el *Canvas*, que es el área donde todos los elementos de la interfaz de usuario deben estar. Este es un *GameObject* con un componente *Canvas* en él, y todos los elementos UI deben ser hijos. Por otro lado, están los componentes visuales (Textos, imágenes, etc.) y de interacción (botones, *toggles*, *dropdown*, etc.).

En la realidad virtual y en cualquier juego, Unity gestiona la interacción de la cámara con los rayos de la cámara o *raycast*. Para ello, el SDK de Google contiene un *script* llamado *GvrPointerPhysicsRaycaster.cs* que se añade como componente al objeto cámara de la escena.

A continuación, se explica de manera resumida la idea sobre la que se basa la implementación de los menús puesto que explicar todos los detalles se haría demasiado extenso.

El orden de ejecución del primer boceto de interfaz procede de la siguiente manera:

1. Se añadió un *Canvas* en la escena.
2. Se añadieron los botones necesarios (*Menu tools*, *Select zone*, *List of Areas*, etc.).
3. Bajo el botón de *Select zone* se añadió dos contenedores, estos son dos *GameObject* que se caracterizan por tener dos componentes de *layout* (o disposición).
4. Dentro de cada contenedor, se añade una plantilla de botón, este botón se va a replicar tantas veces como áreas haya (si corresponde al contenedor de áreas) o de zonas (si corresponde al contenedor de zonas).
5. En el caso de los botones sin menú desplegable, se obvió el paso 3 y 4.
6. Para cada elemento que recibe la interacción del usuario, fue necesario configurar los disparadores (o *trigger*) y su correspondiente *script*.

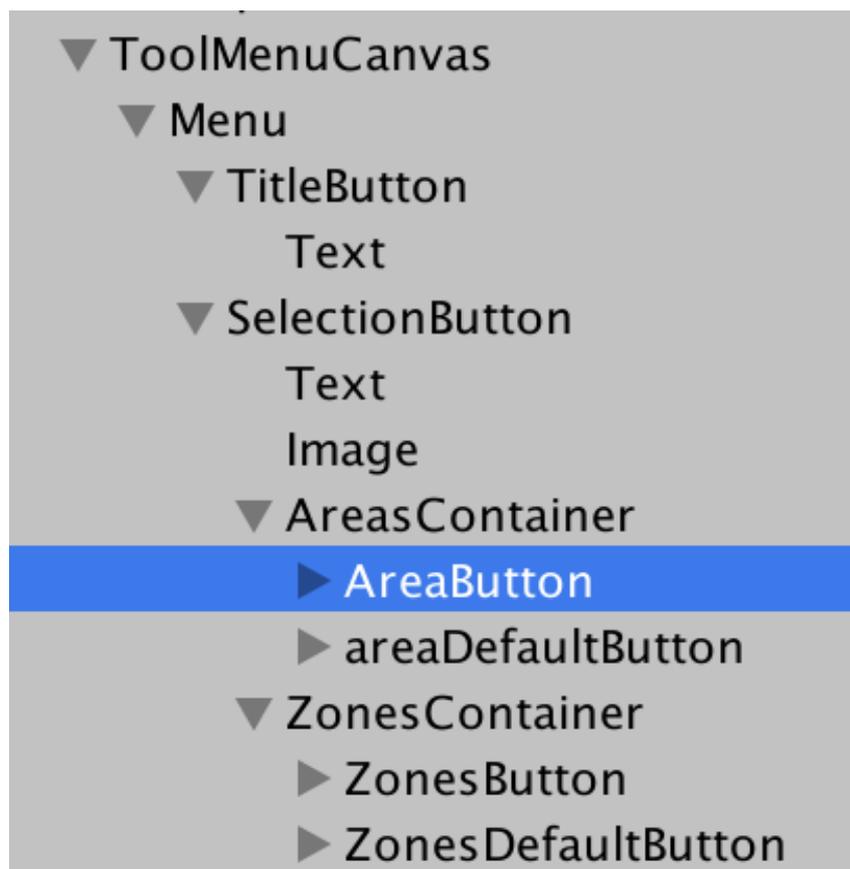


Figura 16. Jerarquía del menú desplegable.

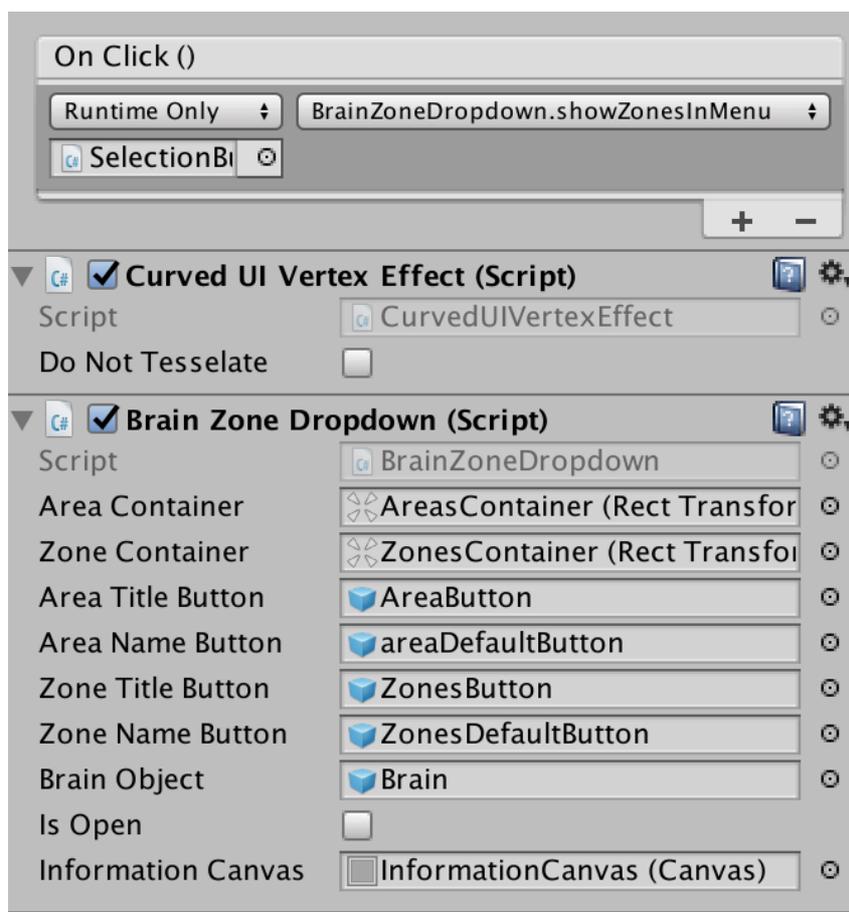


Figura 17. Configuración de un elemento UI que interactúa con el usuario.



Figura 18. Resultado del diseño de la interfaz.

### 3.4 Implementación de la interfaz

Esta etapa reúne el desarrollo de todos los *scripts* necesarios para conectar la interacción del usuario con la interfaz y el cerebro. Antes de comenzar, hay que comentar que no se va a entrar en detalle del código de cada *script* pues son más de 30 y el informe sería demasiado extenso. Veamos para cada una de las opciones del menú, qué pasos se siguieron hasta que fuera funcional.

1. **Select zone.** El botón llama a un *script* llamado *BrainZoneDropdown.cs* que gestiona el menú desplegable.

Para el contenedor de áreas realiza una copia de un botón plantilla por cada área del cerebro y le asigna un *listener* que llama a una función específica para cada área que muestra sus respectivas zonas.

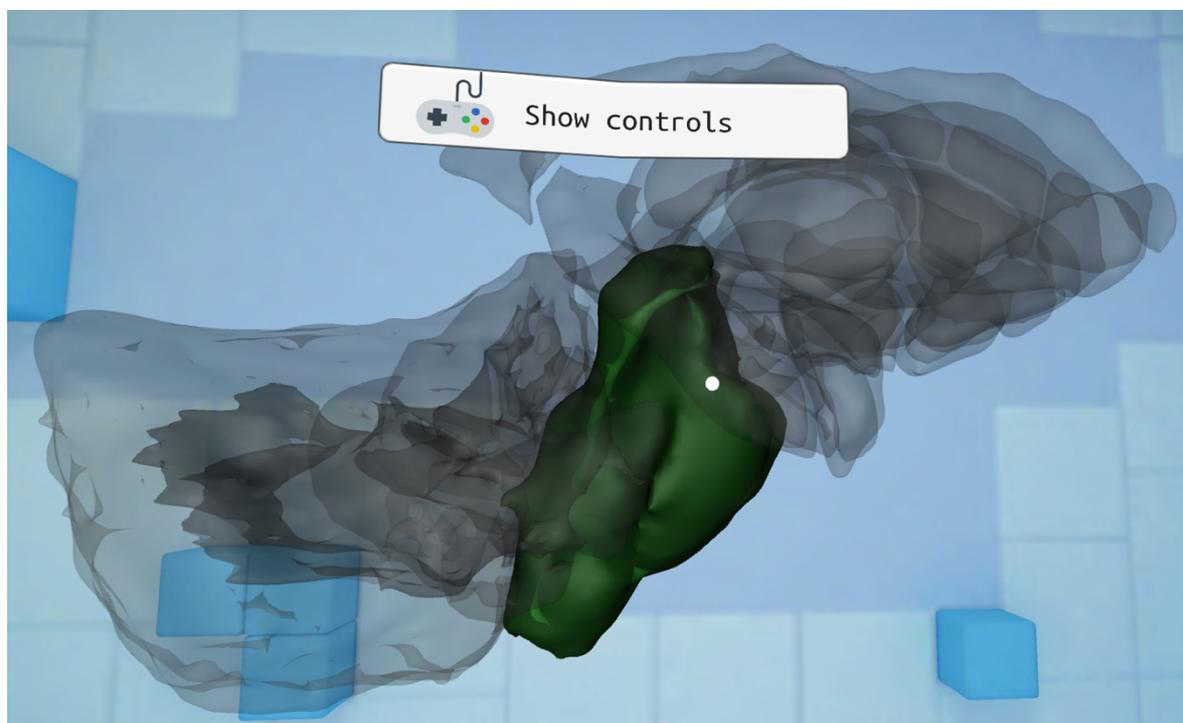
En el caso del contenedor de zonas, es modificado por el *listener* comentado anteriormente, el esquema de funcionamiento es básicamente el mismo, realiza una copia de un botón plantilla por cada zona respectiva, pero, además, le añade un componente de tipo *script* llamado *ShowZoneInBrain.cs* que guarda la zona que se ha seleccionado y su área en un *script* llamado *CurrentSelectedZone.cs* (encargado de almacenar la última zona y área seleccionada).

Por un lado, esta opción pinta la zona del cerebro con un color verde, el área correspondiente de color blanco con un porcentaje bajo de transparencia y todo lo demás de un color negro.

Por otro lado, *ShowZoneInBrain.cs* tiene un método que actualiza la información del *Canvas* de la información asociada a la zona seleccionada. La información de cada zona está almacenada internamente en un *script* llamado *Information.cs* tanto de las zonas de la corteza del cerebro como de las zonas subcorticales.

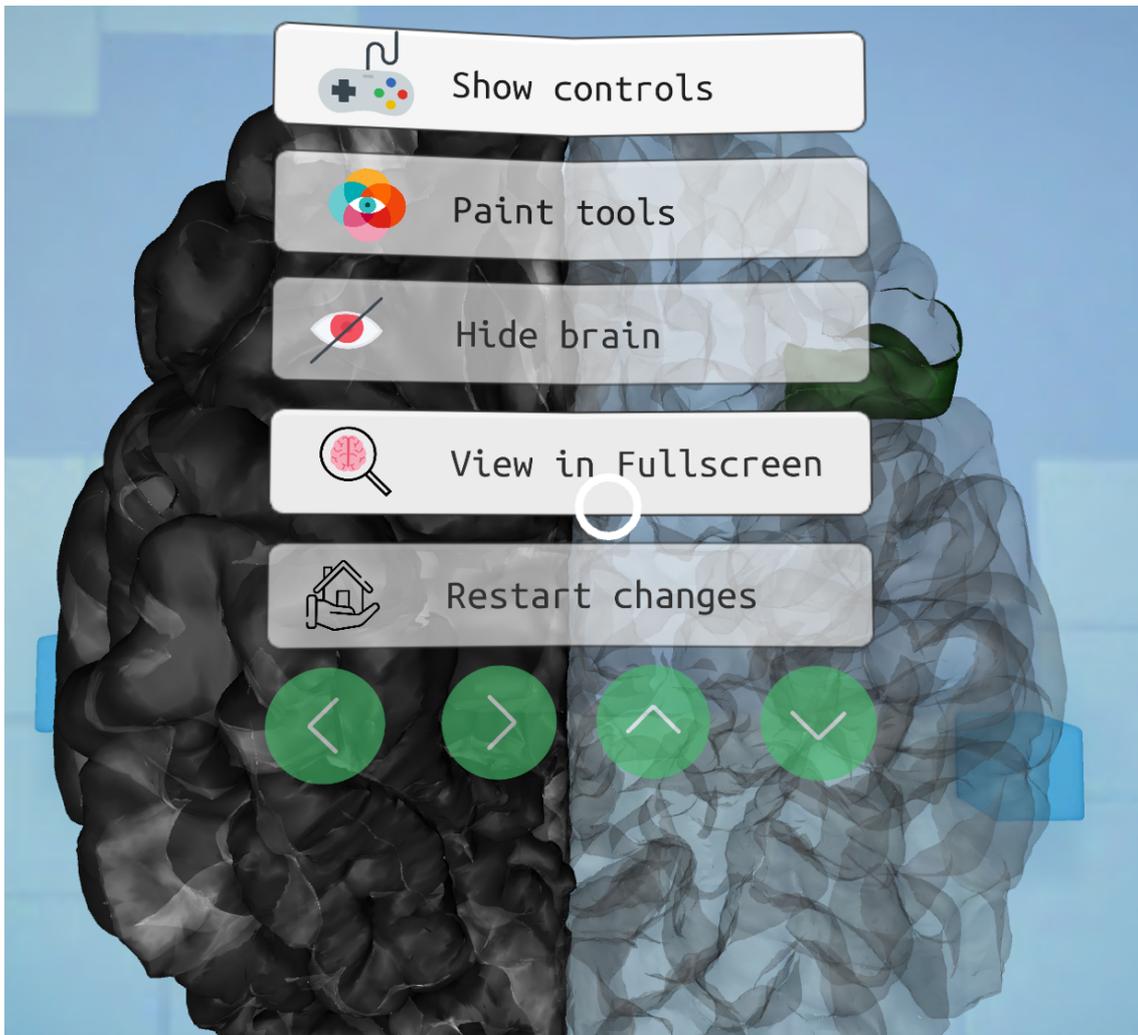


*Figura 19. Canvas con la información de la zona seleccionada.*



*Figura 20. Modelo de las zonas subcorticales.*

2. El resto de las funciones están agrupadas en un contenedor y gestionadas por un botón que se titula *ShowControlsButton*. En este contenedor se encuentran los siguientes botones:



*Figura 21. Opciones de manipulación y pintado del cerebro.*

- **Paint tools.** Su interacción muestra a su vez más botones desplegables que permiten pintar áreas del cerebro tal y como podemos observar en la figura 22. Cada uno de estos botones tiene como componente un *script* encargado de pintar el área correspondiente, así como de decolorarla.

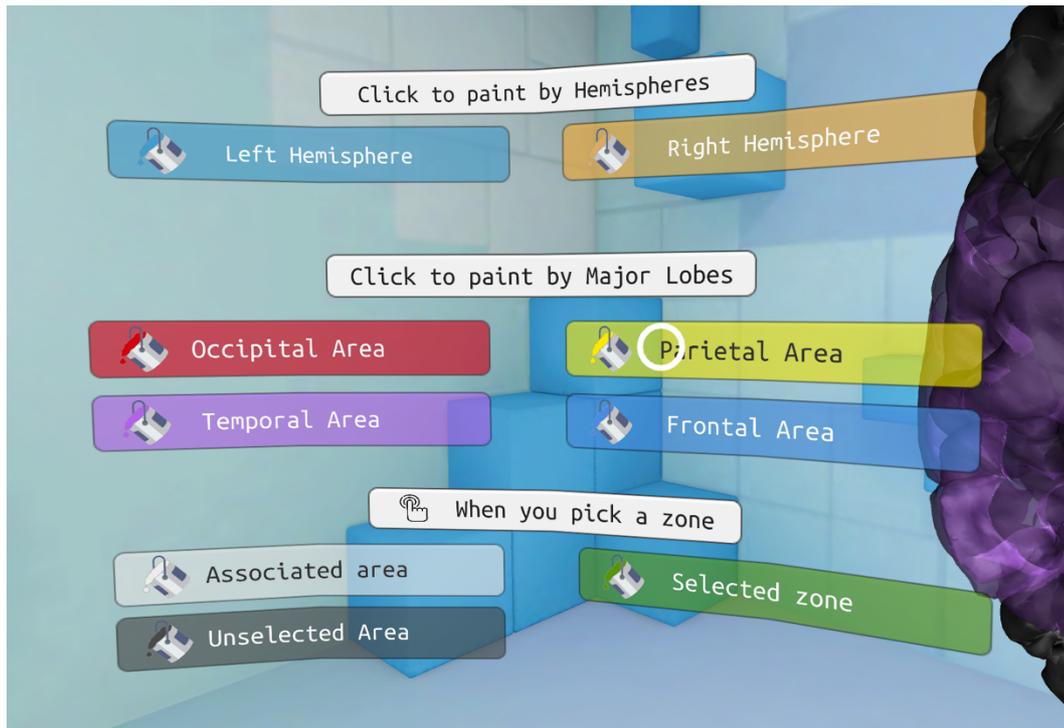


Figura 22. Opciones de pintado.

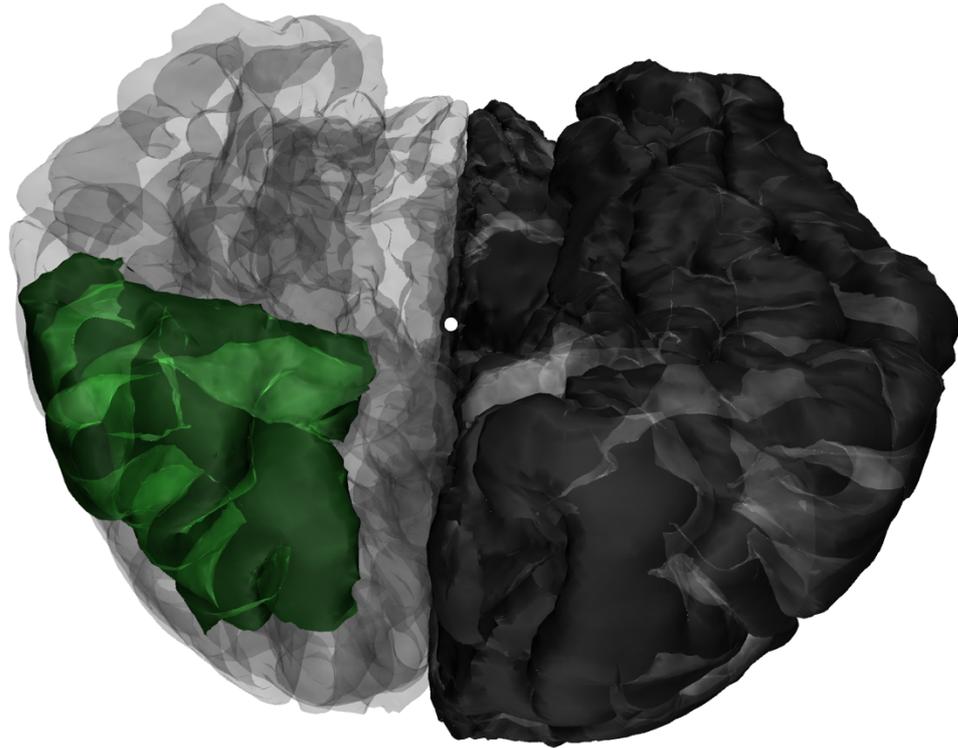
- **Hide Brain.** Tiene como componente el *script HideBrain.cs* encargado de ocultar todas las zonas del cerebro excepto la última zona selecciona que se había almacenado en el *script* de *CurrentSelectedZone.cs*.



Figura 23. Zona del cerebro después de ocultar el cerebro restante.

- **View in fullscreen.** *ChangeToFullScreen.cs* es el *script* encargado de Ocultar el escenario, cambiar el padre del *GameObject* del cerebro por otro llamado *BrainParent* que además pasa a ser hijo de la cámara principal de la escena que tiene un *script* que se activa llamado *RotateBrain.cs* que convierte el movimiento de la cabeza en movimiento alrededor del cerebro. Este gesto es el mismo que ocurre en la realidad cuando cogemos un objeto con las manos y nos movemos con la cabeza alrededor de él.

Para salir de este modo, sólo basta con hacer click en el botón de las gafas y todos los cambios se restauran.



---

*Figura 24. Modo a pantalla completa.*

Para salir de este modo, sólo basta con hacer click en el botón de las gafas y todos los cambios se restauran.

- **Restart changes.** La idea principal es restaurar todos los cambios de rotación realizados en el cerebro y dejarlo en su posición inicial, para ello existe un *script* llamado *RestartChanges.cs* encargado de cumplirlo.
- **Controles de dirección.** Las flechas de dirección son las que realizan la rotación del cerebro en base al origen de este.

# Capítulo 4

## Problemas y soluciones

Durante el desarrollo del proyecto se tuvieron algunos problemas, los cuales se mencionan a continuación.

### **Rotación del cerebro basada en el movimiento de la cabeza**

La principal dificultad fue encontrar documentación relacionada con cómo lograr que el movimiento de la cabeza simulara que el usuario está orbitando alrededor del cerebro.

El problema residía en que la ínfima documentación existente era del SDK de Google antiguo, antes de ser rediseñado. Después de 2016 todos los *prefabs* del SDK cambiaron y con ellos la forma de configurar una escena.

Como solución se procedió a consultar en foros de distintas comunidades como Unity, Stackoverflow o Github. Finalmente, los mismos usuarios de la comunidad de Github respondieron al problema, aportando una solución más simple que cualquiera de las que se podía encontrar por internet, básicamente consistía en coger el *GameObject* del cerebro y hacerlo hijo de la cámara principal de la escena y añadirle a esta un *script* que modificara la rotación del cerebro por la inversa de la rotación de la cámara.

### **Letras de la interfaz pixelada.**

En Unity por defecto al crear un elemento de la interfaz trae el *GameObject Text*, el cual tiene un componente llamado *Text* (que es un *script*) con una serie de atributos públicos accesibles desde la interfaz. Entre ellos, se encuentra *Font*, este atributo muestra la fuente seleccionada.

El problema reside en que el tamaño por defecto con el que viene la fuente es demasiado pequeño. Para solucionarlo, hubo que acceder a la ruta donde estaba situada el archivo con la fuente y modificar el tamaño por defecto como mínimo a un valor mayor que cien.

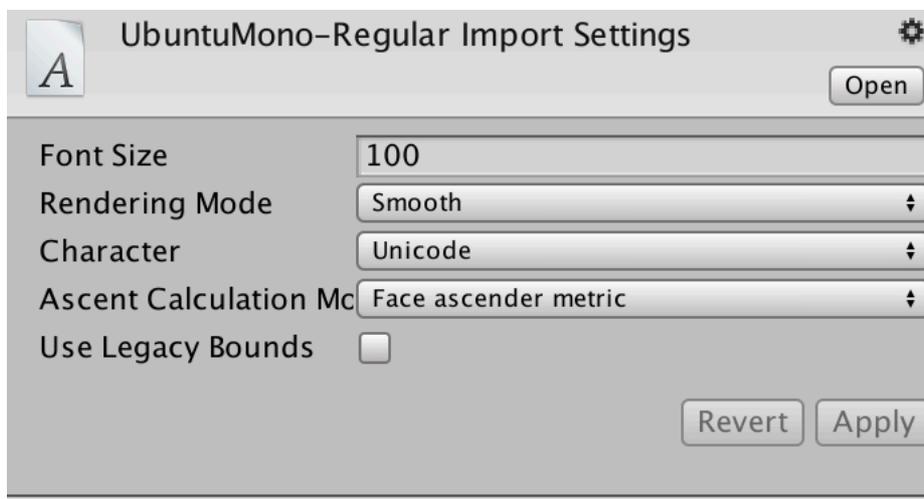


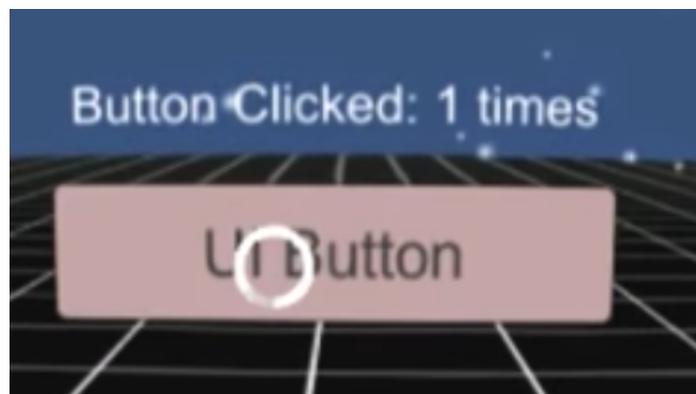
Figura 25. Configuración de una fuente para evitar el pixelado y aumentar la calidad.

## Gafas de realidad virtual sin botón

Existe un número de posibles usuarios que tuvieran alguna gafa tipo Google CardBoard comprada años anteriores y sean de versiones sin ningún botón que simule el click. Para este tipo de gafas hay una solución estándar basada en el *input timed gaze* o entrada de mirada temporizada si lo traducimos al castellano. Esto consiste en que el usuario mira fijamente a un punto durante un determinado tiempo y esto se traduce en una interacción por ejemplo de click. En la figura 28 se puede ver cómo se visualiza la retícula sin colisionar con ningún objeto. Y en la figura 29 el aspecto cuando colisiona con uno. Desde el momento que colisiona el programa lleva una cuenta atrás del tiempo marcado por defecto y tras pasar ese tiempo realiza la acción descrita en el *script* correspondiente.



*Figura 26. Aspecto de la retícula sin colisión.*



*Figura 27. Aspecto de la retícula con colisión más cuenta atrás de interacción.*

# Capítulo 5

## Conclusiones y líneas futuras

### 5.1 Conclusiones

El uso de las tecnologías en la enseñanza es una alternativa frente a los métodos tradicionales de estudio. Cualquier ayuda que le aporte al estudiante alguna ventaja debe ser considerada. En el caso de la realidad virtual, es una herramienta muy versátil que permite, por un lado, a los profesores servir contenido adicional a sus alumnos, capaz de reforzar los contenidos explicados en las clases y por otro lado, ayudar a los alumnos a expandir y reforzar sus conocimientos.

Por el momento, el consumo de esta tecnología tiene una serie de inconvenientes, para empezar, es cierto que hoy en día casi todas las personas jóvenes poseen un *smartphone*, pero no hay que dejar de comentar que para ejecutar una aplicación de realidad virtual es necesario poseer un *smartphone* o un PC. Esto supone partir de un precio base de 100€ a 200€ como mínimo en el caso de un *smartphone* y de 500€ a 800€ como mínimo en el caso de un ordenador de sobremesa, a todo esto, hay que sumarle el precio de las gafas.

Además, a todo lo anterior hay que sumarle que la mayor barrera reside, no tanto en la tecnología, sino en la decisión de millones de profesores de impulsarla dentro de sus aulas.

Para concluir, el reto es crear contenidos de realidad virtual que sean aliados del proceso educativo junto con la implicación del profesorado, centros e instituciones educativas. Una actitud participativa en un nuevo marco en que el sistema educativo adopte la práctica experiencial y aproveche las nuevas

oportunidades tecnológicas al servicio de la enseñanza y el aprendizaje.

## 5.2 Líneas futuras

La tarea más importante que necesita la aplicación es añadir un modo que permita evaluar los conocimientos adquiridos. Para un alumno no sólo es importante tener acceso a contenido, sino también practicar con él de alguna forma. BrainVR no trabaja ese aspecto. Sería interesante añadir a la aplicación alguna opción donde el usuario pueda examinarse.

Como solución, se han planteado algunas ideas como realizar exámenes tipo test. Por un lado, se le puede pedir al usuario escoger cuál de varias zonas coloreadas es aquella por la que se le pregunta y, por otro lado, que se le pida encontrar una zona específica en el cerebro y el usuario tenga que señalarla en el cerebro.

Otra forma de añadirle algún atractivo a este modo examen sería la gamificación o aprendizaje divertido. Hoy en día son muchas las aplicaciones que se centran alrededor de este concepto para motivar al usuario a seguir usando la aplicación y mejorando. Para ello, se recompensa con puntuación cada vez que el usuario acierta y va escalando niveles.

Otro punto interesante, sería añadir la opción al usuario de poder guardar sus propios apuntes. Esto significa que la información que aparece asociada a cada zona del cerebro pueda ser editada por el usuario, pudiendo subir la información que se considere importante. Para realizar esta funcionalidad, haría falta el uso de una base de datos que almacene la información para cada usuario, un sistema de autenticación y de alguna aplicación web donde el usuario pueda loguearse y acceder a todos sus ajustes para modificarlos.

El último punto es adaptar la aplicación a otras plataformas además de

las móviles (IOS y Android). Para conseguir esto haría falta hacer uso de otros SDK como los de Oculus o HTC Vive.

# Capítulo 6

## Summary and Conclusions

The use of technologies in teaching is an alternative to traditional methods of study. Any help that gives the student any advantage should be considered. In the case of virtual reality, it is a very versatile tool that allows, on the one hand, teachers to serve additional content to their students, able to reinforce the contents explained in the classes, on the other hand, to help the students to expand and reinforce their knowledge.

At the moment, the consumption of this technology has a number of drawbacks, to begin with, it is true that today almost all young people have a *smartphone*, but we must not forget that to run a virtual reality application is necessary own a *smartphone* or a PC, this means starting from a base price of € 100 to € 200 at least in the case of a *smartphone* and € 500 to € 800 at least in the case of a desktop computer, all this is that add the price of the corresponding glasses.

In addition, to all the above we must add that the biggest barrier lies not so much in technology but in the decision of millions of teachers to promote it within their classrooms.

To conclude, the challenge is to create virtual reality contents that are allies of the educational process together with the involvement of teachers, schools and educational institutions. A participatory attitude in a new framework in which the educational system adopts the experiential practice and takes advantage of new technological opportunities at the service of teaching and learning.

# Capítulo 7

## Presupuesto

Este capítulo describe el coste de cada una de las tareas de desarrollo del proyecto, en un presupuesto estimado con los precios de mercado actuales.

### 7.1 Presupuesto

Para la realización de este proyecto, el desarrollo ha sido dividido en una serie de tareas detalladas en la sección 1.8. En el desarrollo del proyecto se emplearon una cantidad de horas para cada una de las tareas, como se puede ver en la tabla 7-1.

Estipulando que el precio aproximado por hora de trabajo de un ingeniero informático es de 16 €, el presupuesto del proyecto es:

Tareas de desarrollo	Número de horas
Tarea 0. Planificación	8h
Tarea 1. Formación	60h
Tarea 2. Configuración del proyecto	2h
Tarea 3. Diseño	30h
Tarea 4. Implementación	100h
Tarea 5. Test de usuario.	20h
Tarea 6. Documentación.	20h
<b>Número de horas de trabajo</b>	<b>240h</b>

<b>Presupuesto en horas de trabajo</b>	4800€
Google CardBoard	7,99€
Gamepad	14,99€
Smartphone	250€
<b>Presupuesto en materiales</b>	272,98€
<b>Presupuesto total</b>	5.072,98€

*Tabla 7-1. Presupuesto y horas invertidas.*

# Bibliografía

- [1] V. Duque, «ELSEVIER,» 05 2018. [En línea]. Available: <https://www.elsevier.es/corp/generacionelsevier/formas-practicas-estudiar-anatomia/>.
- [2] A. García, «labrechadigital,» [En línea]. Available: <http://www.labrechadigital.org/labrecha/Articulos/los-beneficios-de-la-tecnologia-en-la-educacion.html>. [Último acceso: 05 2018].
- [3] A. a. B. R. a. K. S. Balanskat, «The ICT impact report,» *European Schoolnet*, vol. 1, pp. 1--71, 2006.
- [4] M. W. María Michelle Camus, «Representación gráfica del cuerpo humano,» [En línea]. Available: [https://wiki.ead.pucv.cl/T%C3%ADtulo\\_I:\\_Representaci%C3%B3n\\_gr%C3%A1fica\\_del\\_cuerpo\\_humano#Tecnolog.C3.ADas\\_de\\_representaci.C3.B3n\\_del\\_cuerpo\\_humano](https://wiki.ead.pucv.cl/T%C3%ADtulo_I:_Representaci%C3%B3n_gr%C3%A1fica_del_cuerpo_humano#Tecnolog.C3.ADas_de_representaci.C3.B3n_del_cuerpo_humano). [Último acceso: 05 2018].
- [5] M. Plus, «MedlinePlus,» [En línea]. Available: [https://medlineplus.gov/spanish/mriscans.html#cat\\_82](https://medlineplus.gov/spanish/mriscans.html#cat_82). [Último acceso: 05 2018].
- [6] J. Sobotta, Atlas of human anatomy, vol. 1, Hafner Pub. Co., 1986.
- [7] F. H. Netter, Atlas of Human Anatomy, Professional Edition E-Book: including NetterReference. com Access with Full Downloadable Image Bank, Elsevier Health Sciences, 2014.
- [8] BioDigital, «BioDigital,» [En línea]. Available: [www.biodigitalhuman.com](http://www.biodigitalhuman.com). [Último acceso: 05 2018].
- [9] D. C. LaFuente, «RFI,» [En línea]. Available: <http://es.rfi.fr/ciencia/20170825-la-realidad-virtual-inmersa-en-cirugia>. [Último acceso: 05 2018].
- [10] S. H. Q. VR, «Sea Hero Quest VR,» [En línea]. Available: <http://www.seaheroquest.com/site/es/credits>. [Último acceso: 05 2018].

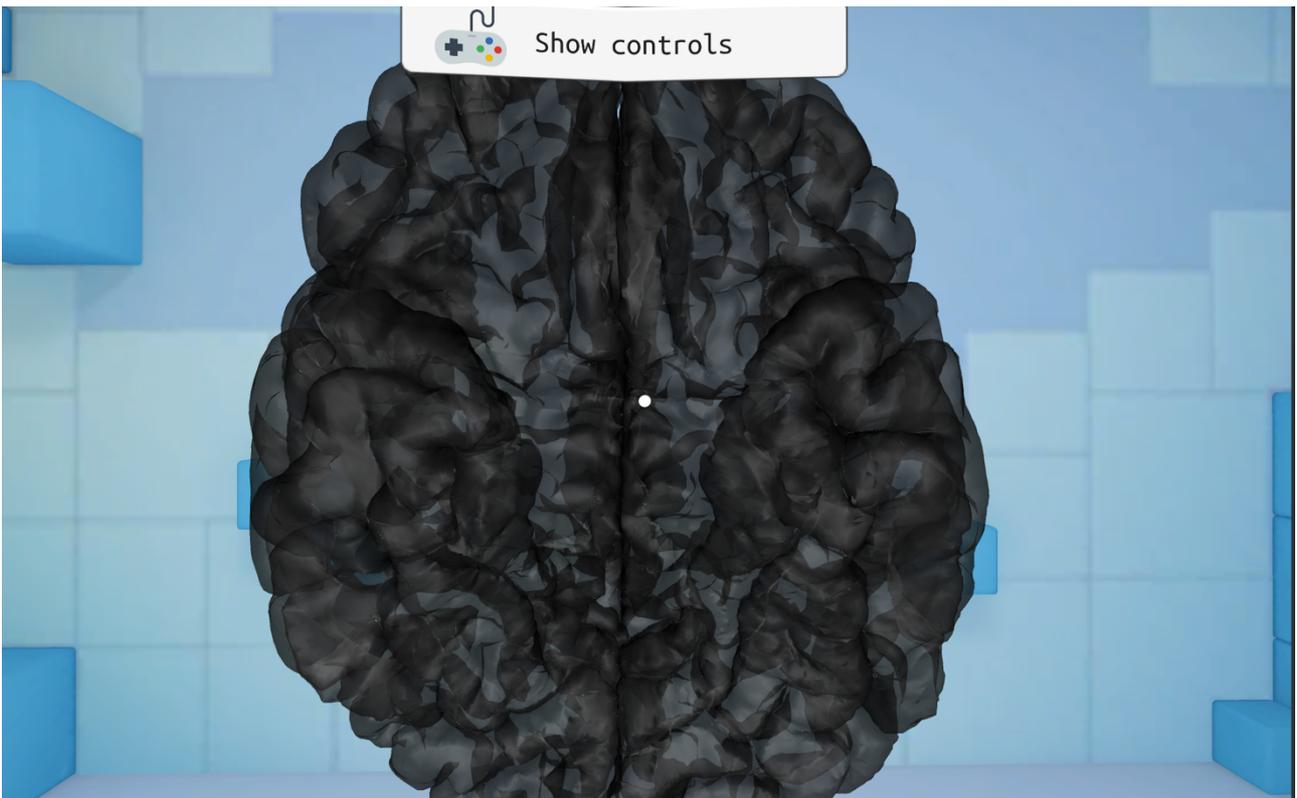
- [11] CEAFA, «CEAFA,» [En línea]. Available: <https://www.ceafa.es/es/que-comunicamos/noticias/realidad-virtual-para-detectar-el-alzheimer>. [Último acceso: 05 2018].
- [12] U. C. I. d. Madrid, «uc3m,» [En línea]. Available: [https://www.uc3m.es/ss/Satellite/UC3MInstitucional/es/Detalle/Comunicacion\\_C/1438738168390/1371216052710/Realidad\\_virtual\\_para\\_la\\_rehabilitacion\\_motora\\_del\\_hombro](https://www.uc3m.es/ss/Satellite/UC3MInstitucional/es/Detalle/Comunicacion_C/1438738168390/1371216052710/Realidad_virtual_para_la_rehabilitacion_motora_del_hombro). [Último acceso: 05 2018].
- [13] P. Mercurio, «THYRD informatics,» [En línea]. Available: <http://thyrd.info/brainvr>. [Último acceso: 05 2018].
- [14] M. Media, «viveport,» [En línea]. Available: <https://www.viveport.com/apps/10fc4de2-db6f-49ed-996f-9048e67c509b>. [Último acceso: 05 2018].
- [15] P. m. animation, «Polygon medical animation,» [En línea]. Available: <https://www.polygonmedical.com/inside-your-cells-virtual-reality>. [Último acceso: 05 2018].
- [16] P. m. animation, «Youtube,» [En línea]. Available: <https://www.youtube.com/watch?v=136cE452fGc>. [Último acceso: 05 2018].
- [17] P. m. animation, «Youtube,» [En línea]. Available: <https://www.youtube.com/watch?v=cnAwR61qdJk>. [Último acceso: 05 2018].
- [18] U. 3D, «Unity 3D,» [En línea]. Available: <https://unity3d.com/es>. [Último acceso: 05 2018].
- [19] Google, «Google VR,» [En línea]. Available: <https://developers.google.com/vr/develop/unity/get-started-android>. [Último acceso: 05 2018].
- [20] G. Cardboard, «Google Cardboard,» [En línea]. Available: <https://vr.google.com/cardboard/>. [Último acceso: 05 2018].
- [21] Wikipedia, «Wikipedia,» [En línea]. Available:

- [https://es.wikipedia.org/wiki/Motor\\_de\\_videojuego](https://es.wikipedia.org/wiki/Motor_de_videojuego). [Último acceso: 05 2018].
- [22] U. 3D, «Unity 3D,» [En línea]. Available: <https://unity3d.com/es/unity>. [Último acceso: 05 2018].
- [23] U. 3D, «Unity Documentation,» [En línea]. Available: <https://docs.unity3d.com/Manual/VROverview.html>. [Último acceso: 05 2018].
- [24] Luden.io, «Luden.io,» [En línea]. Available: <https://luden.io/incell/>. [Último acceso: 05 2018].
- [25] Google, «CardBoard,» [En línea]. Available: [https://vr.google.com/intl/es\\_es/cardboard/manufacturers/](https://vr.google.com/intl/es_es/cardboard/manufacturers/). [Último acceso: 05 2018].
- [26] Brainder, «Brainder,» [En línea]. Available: <https://brainder.org/research/brain-for-blender/>. [Último acceso: 05 2018].
- [27] U. 3D, «Unity 3D,» [En línea]. Available: <https://unity3d.com/es/get-unity/download>. [Último acceso: 05 2018].
- [28] Brainder, «Brainder,» [En línea]. Available: <https://brainder.org/2012/05/08/importing-freesurfer-subcortical-structures-into-blender/>. [Último acceso: 05 2018].
- [29] [En línea]. Available: <https://github.com/mercurio/brainvr>.
- [30] [En línea]. Available: <https://brainder.org/research/brain-for-blender/>.
- [31] [En línea]. Available: <https://github.com/googlevr/gvr-unity-sdk/releases>.
- [32] [En línea]. Available: <https://unity3d.com/es/get-unity/download>.
- [33] [En línea]. Available: <https://brainder.org/2012/05/08/importing-freesurfer-subcortical-structures-into-blender/>.
- [34] [En línea]. Available: [https://vr.google.com/intl/es\\_es/cardboard/manufacturers/](https://vr.google.com/intl/es_es/cardboard/manufacturers/).

# Anexo 1

## Tutorial de la aplicación

El primer paso es iniciar la aplicación. Una vez abierta, lo primero que vemos es lo que se muestra en la figura 30. Después se van a explicar las diferentes opciones que podemos realizar.



*Figura 28. Inicio de la aplicación.*

### 1. Seleccionar una zona del cerebro.

En el lado derecho de la sala donde nos encontramos, aparece un menú con diferentes opciones, entre ellas está **Select zone**. Esta opción despliega una serie de áreas que vemos en la figura 31.



Figura 29. Áreas del cerebro.

A continuación, para escoger una zona del cerebro, debemos pulsar el botón de las gafas o el botón de *submit* del *gamepad bluetooth* conectado al *smartphone* (Cada *gamepad* puede tener los botones con distinta distribución, para ello ver el manual de usuario del dispositivo en uso).

Una vez hecho, se despliega el listado de zonas pertenecientes al área escogida. En la siguiente figura 32 se muestran todas las zonas pertenecientes al área temporal.

Para escoger la zona, debemos realizar el mismo proceso que para el área, se debe hacer uso del botón de las gafas de realidad virtual. Después la zona elegida se muestra en el cerebro de un color verde, su hemisferio de un color blanco transparente y el otro de un color negro tal y como se muestra en la figura 20.



*Figura 30. Lista de zonas pertenecientes al área temporal.*

## **2. Cambiar entre el modelo de la corteza cerebral y el modelo subcortical.**

Para cambiar entre el modelo de la corteza cerebral y el modelo subcortical, tenemos que ir al menú de la derecha de la sala, donde tenemos la opción de seleccionar la zona (**Select zone**). Todas las opciones pertenecen al modelo de la corteza excepto la última. Ésta permite cargar el modelo subcortical, mostrar sus zonas y escoger una tal y como vimos en la figura 20.

## **3. Ver la información de la zona seleccionada.**

En primer lugar, debemos haber realizado el paso 1. En segundo lugar, debemos mirar hacia el lado izquierdo de la sala, allí veremos un panel con el nombre de la zona que hemos seleccionado e información relativa a la misma. El resultado de este paso se mostró en la figura 19.

#### 4. Ver en pantalla completa.

El botón **Show controls** despliega una serie de opciones que vimos en la figura 21. De ellas, debemos escoger **Show in FullScreen**, después tenemos como resultado lo visto en la figura 24.

#### 5. Mostrar/ocultar el cerebro, excepto la zona seleccionada.

El botón **Show controls** despliega una serie de opciones que vimos en la figura 21. De ellas, debemos escoger **Hide brain**, después tenemos como resultado lo visto en la figura 23.

#### 6. Rotar el cerebro.

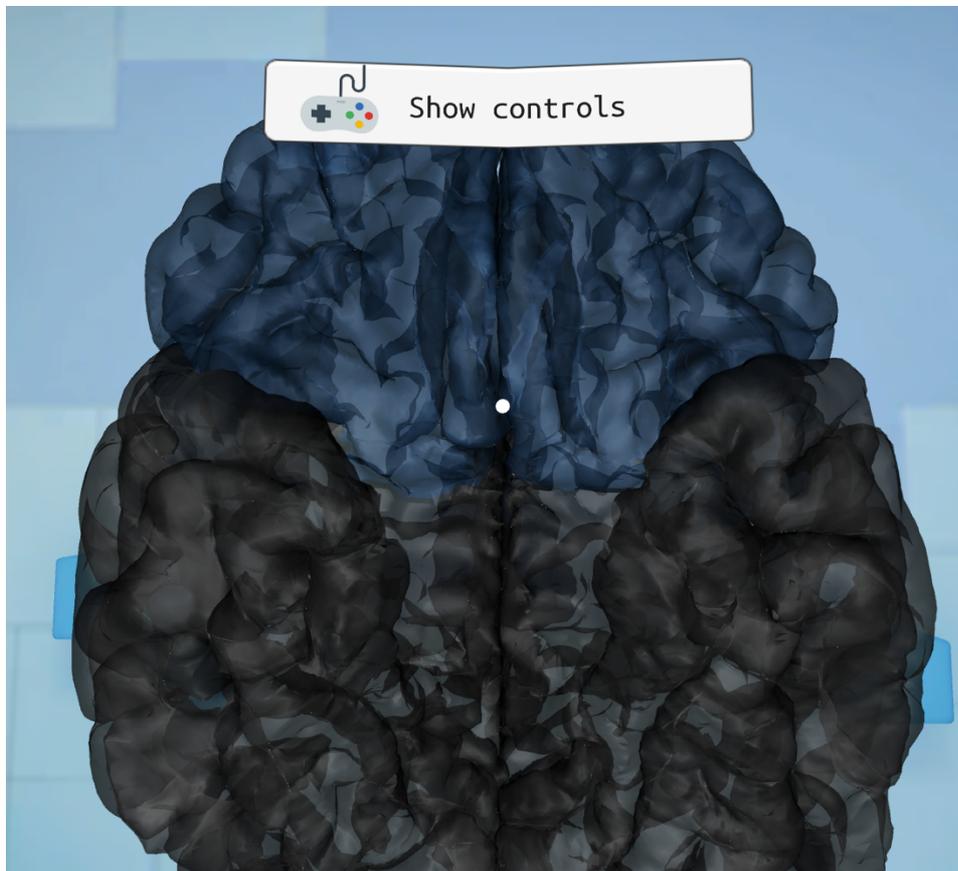
El botón **Show controls** despliega una serie de opciones que vimos en la figura 21. Después de todas las opciones, tenemos cuatro flechas de color verde que apuntan a diferentes direcciones, para rotar sobre una sólo debemos pulsar el botón de las gafas sobre cualquiera de ellas.

#### 7. Restaurar la rotación del cerebro a la original.

El botón **Show controls** despliega una serie de opciones que vimos en la figura 21. De ellas, debemos escoger **Restart changes**, después el cerebro volverá a la posición original antes de haber realizado ningún cambio.

#### 8. Pintar las áreas del cerebro.

El botón **Show controls** despliega una serie de opciones que vimos en la figura 21. De ellas debemos escoger **Paint tools**, después tenemos dos grupos, uno por hemisferios y otro por lóbulos mayores. Para pintar todos los hemisferios debemos pulsar en el botón blanco que dice **click to Paint by hemispheres** y para pintar los cuatro lóbulos mayores debemos pulsar en el botón blanco que dice **click to Paint by Major Lobes**. Dentro de cada uno de estos grupos existe varios botones de colores, el color del fondo de cada uno nos indica el color con el que va a pintar esa área en el cerebro, haciendo uso del botón podemos colorear cada una de las áreas. Veamos en la figura 33 el resultado de pintar el lóbulo frontal.



*Figura 31. Cerebro con el área frontal pintada de color azul.*