



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CYBERSICKNESS EN DIFERENTES TÉCNICAS DE MOVIMIENTO PARA
REALIDAD VIRTUAL

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

JOAQUÍN AGUSTÍN URIBE MONTERO

PROFESOR GUÍA:
NELSON BALOIAN TATARYAN

MIEMBROS DE LA COMISIÓN:
DANIEL CALDERÓN SAAVEDRA
FEDERICO OLMEDO BERÓN

SANTIAGO DE CHILE
2024

Resumen

A pesar de que la realidad virtual es una tecnología que ha aumentado su popularidad en gran medida los últimos años, siendo utilizada tanto para educación como para el entretenimiento, no todas las personas pueden experimentarla de forma agradable debido a que al intentar moverse dentro de un entorno virtual pueden presentar síntomas de cybersickness, un malestar similar a la cinetosis o mareo por movimiento para realidad virtual.

En este trabajo de título se busca encontrar y caracterizar diferentes técnicas para simular movimiento en un entorno virtual, con el fin de hallar factores que produzcan cybersickness y potencialmente formas de reducirlo.

Para esto se desarrolla una aplicación de realidad virtual con el motor de videojuegos Unity en la que se implementan tres técnicas de movimiento y dos escenarios en los que se debe completar un desafío. Utilizando la aplicación se realiza un experimento en el que sujetos de prueba deben utilizar una de las técnicas de movimiento para completar un desafío en uno de los escenarios, de forma en que se estudian todas las combinaciones de técnica de movimiento con escenario posibles. Los datos del experimento se recolectan a través de un cuestionario, la autopercepción del usuario respecto a su malestar en tiempo real y también a través del estudio de las ondas cerebrales de los mismos.

Luego de analizar los resultados se logra concluir que cada técnica de movimiento pareciera tener aspectos positivos y negativos dependiendo del uso que se le esté dando, sin embargo, debido al tamaño de la muestra, las intuiciones obtenidas no son concluyentes. Junto con esto se analiza la utilidad de las formas de medición utilizadas y se hace una reflexión de los aspectos que pudieran mejorar, dejando una base para la realización de nuevos experimentos a futuro.

A Dios, que cumplió la promesa de encargarse de mi educación hasta el final.

Agradecimientos

En primer lugar, me gustaría agradecer a mis padres. Sé que ellos creen que soy capaz de lograr cualquier cosa que me proponga y, aunque quizás yo no me tengo tanta fe a mí mismo, creo que ha sido uno de los grandes motivos por los que ahora puedo permitirme escribir estas palabras. Gracias por permitirme soñar sabiendo que esos sueños pueden estar a mi alcance.

Agradezco a mis dos hermanos Gonzalo Y Beatriz, que me ayudaron, en los momentos de mayor estrés, dando consejos sinceros y muchas veces haciéndome notar que me ahogaba en un charco.

También agradezco a mi pareja, Javiera, quien me ha estado dando ánimos y motivación cada día para escribir este informe. Además de brindarme su compañía durante la realización de los experimentos.

Quiero agradecer también a Franco Balbiano y Gonzalo Mendoza por ofrecer de su tiempo para que tenga la posibilidad de realizar los experimentos, junto a ellos además agradezco a todas las personas que sirvieron como sujeto de pruebas.

Finalmente, agradezco a los profesores Nelson Baloian y Gustavo Zurita por acompañarme a lo largo de todo este proceso.

Tabla de Contenido

1. Introducción	1
1.1. Objetivos	2
1.1.1. Objetivo Principal	2
1.1.2. Objetivos Específicos	3
1.2. Evaluación	3
1.3. Estructura de la memoria	3
2. Estado del arte	5
2.1. Realidad virtual	5
2.1.1. Equipo de VR	5
2.1.2. Motores para desarrollar en VR	6
2.1.3. Herramientas de Unity para VR	6
2.1.4. Técnicas de movimiento en el medio de videojuegos	7
2.1.5. CS y técnicas de movimiento en la literatura	8
2.2. Medición de CS con BCI	10
2.2.1. Trabajo previo	10
3. Antecedentes	13
3.1. Unity	13
3.1.1. GameObjects y Componentes	13
3.1.2. MonoBehaviour	14
3.1.3. Transform	14

3.1.4.	Acciones de input	14
3.2.	Emotiv	16
3.2.1.	Bandas de frecuencia	17
3.2.2.	Performance metrics	17
4.	Análisis y Diseño de la solución	19
4.1.	Diseño del experimento	19
4.1.1.	Obtención de datos	19
4.2.	Requisitos del entorno de pruebas	20
4.3.	Diseño del entorno del pruebas	21
4.3.1.	Técnicas de movimiento	21
4.3.2.	Escenarios	22
4.3.3.	Reducción del scope	24
5.	Implementación	25
5.1.	XR Interaction Toolkit	25
5.1.1.	GameObjects	25
5.2.	Técnicas de Movimiento	27
5.2.1.	Caminata estándar	27
5.2.2.	Caminata con gesto	27
5.2.3.	Propulsión	29
5.2.4.	Gancho	29
5.3.	Escenas	30
5.3.1.	Menú Principal	30
5.3.2.	Laberinto	31
5.3.3.	Pista de carreras	32
5.4.	Recolección de datos	34
5.4.1.	Medidor de malestar	34

5.4.2. Integración de Unity con cortex API	35
6. Experimento	38
6.1. Consideraciones	38
6.2. Procedimiento	38
6.2.1. Participantes	38
6.2.2. Preparación	39
6.3. Actividad	42
6.4. Cuestionario	42
6.5. Dificultades	43
6.5.1. Oculus Quest 2 y conexión a internet	43
6.5.2. Epoch+	43
7. Resultados	45
7.1. Formulario	45
7.2. Medidor de malestar	47
7.3. Ondas cerebrales	48
8. Análisis y Reflexión	52
8.1. Técnicas de movimiento	52
8.1.1. Caminata con gesto	52
8.1.2. Caminata estándar	52
8.1.3. Gancho	53
8.2. Medición	53
8.2.1. Medidor de malestar	53
8.2.2. EEG	54
8.3. Experimento	54
9. Trabajo futuro	55
9.1. Continuar con el mismo experimento	55

9.2. Extender la cantidad de pruebas	55
9.3. Profundizar en causas de CS generales	56
9.4. Detección de CS utilizando BCI	56
10. Conclusión	57
Bibliografía	61
Anexos	62
Anexo A. Transcripción del consentimiento informado utilizado en el experimento	63
title	64
Anexo B. Cuestionario	65
B.1. Caracterización	65
B.2. Síntomas	66
B.3. Autopercepción	66
B.4. Calidad de la aplicación	66
title2	67

Índice de Tablas

6.1. Tabla de síntomas consultados en el cuestionario y su categoría	43
7.1. Porcentaje de identificación de síntomas para cada técnica de movimiento . .	45
7.2. Porcentaje de identificación de síntomas por cada escenario	46
7.3. Porcentaje de recurrencia de cada síntoma para cada TM y Escenario diferente	46
7.4. Promedio del medidor de mareo para grupo con una misma TM y escenario .	47

Índice de Ilustraciones

2.1.	Meta Quest 2 con sus componentes.	6
2.2.	Videojuego Beat Saber, imagen extraída de video de usuario de YouTube Omotea [29]	7
2.3.	Perspectiva de juego de Epic Roller Coasters, extraído de página oficial en Steam [31]	8
2.4.	Jugador de Gorilla Tag desplazándose, extraído de la página oficial en Steam [32]	9
2.5.	Gráfico mostrando relación entre ondas Alpha, Delta y Theta con CS, extraído del trabajo de Krokos et al. [23]	10
2.6.	Gráfico de dos diodos que parecen relacionarse con el CS.	11
3.1.	GameObject de una llave y sus componentes	14
3.2.	Componentes del modelo del objeto llave	15
3.3.	Menú de acciones de Unity	15
3.4.	Distribución de los diodos del Epoc+, extraído del trabajo de Hou et al. [37]	16
4.1.	Bocetos de las técnicas de movimiento desarrolladas	22
4.2.	Bocetos de los escenarios que se planifican desarrollar	23
5.1.	Diagrama de botones disponibles en el XRIT	26
5.2.	Diagrama de árbol de objetos del jugador	26
5.3.	Diagrama simple de la caminata con gesto	28
5.4.	Diagrama de clases de las locomociones implementadas	30
5.5.	Menú principal para seleccionar técnica de movimiento y escenario	31

5.6. Prueba del laberinto en vista ortogonal	32
5.7. Prueba de la pista de carreras en vista aérea	33
5.8. Generador de olas en escena	34
5.9. Medidor de malestar	35
5.10. Comparación de posiciones para la barra de mareo. A la izquierda fija a la muñeca y a la derecha fija a la cámara	36
5.11. Diagrama de estados para crear conexión con la API	37
6.1. Personal médico supervisando el experimento	39
6.2. Cuatro de los participantes del experimento	40
6.3. Interfaz de conexión del EMOTIV Launcher	41
6.4. Emotiv FLEX 2	44
7.1. Gráfico de opiniones respecto a la calidad de la aplicación	47
7.2. Gráfico de cantidad de síntomas en función de valores del medidor de malestar por sujeto	48
7.3. Comparación de poder en ondas theta y alpha de dos sujetos de prueba	49
7.4. Gráficos de pila con proporción de bandas de poder del sujeto 1 (superior) y sujeto 2 (inferior)	50
7.5. Gráfico de líneas comparando la distribución promedio de bandas de poder para diferente cantidad de síntomas percibidos	51

Capítulo 1

Introducción

La realidad virtual (VR, por sus siglas en inglés) es una tecnología que nos permite introducirnos en entornos virtuales (EV) en los que es posible experimentar situaciones que, de forma habitual, no serían posibles de una forma segura, e incluso interactuar con los objetos que las componen. A diferencia de otras tecnologías como la realidad aumentada, en donde se colocan imágenes por encima de lo real para dar más información al usuario, la VR apunta a obtener una inmersión total en la que todo lo que el usuario puede ver ha sido generado de forma artificial por computadora.

En las últimas décadas, la VR ha experimentado un gran crecimiento tanto en su uso comercial como en el mundo de la investigación científica. Esto se ha visto evidenciado en el uso de la VR en diferentes áreas, que no se reduce solamente a la industria del entretenimiento, sino que también a la educación [18], entrenamiento médico [14] e incluso terapia [17], entre otros, generando así un aumento en el interés por usar VR. Sin embargo, es habitual que luego de estar inmerso en un ambiente virtual durante cierto tiempo, el usuario comience a experimentar síntomas usualmente asociados a la cinetosis, empeorando la experiencia y, en muchos casos, haciendo que el usuario se vea forzado a detenerse por completo para descansar.

La cinetosis o “mareo por movimiento” es un trastorno temporal causado por el movimiento, por ejemplo, durante un viaje. Este genera síntomas tales como palidez, náuseas, inquietud, hipersalivación y arcadas [27]. Esto puede ser una barrera de entrada al medio de la realidad virtual que algunas personas no son capaces de superar, pues el malestar es demasiado profundo.

A pesar de que el malestar generado por la interacción en la VR muestre síntomas muy similares a la cinetosis, es importante notar que en un entorno de VR no hay un movimiento real como lo habría, por ejemplo, al estar viajando en un barco. Por lo tanto, no sería correcto llamarlo de ese modo, en su lugar, el fenómeno es conocido como *cybersickness* (CS) [3].

El malestar producido por el CS en diferentes usuarios puede variar para un mismo entorno de VR y, en general, se produce por una disonancia entre lo que percibe el usuario dentro del EV y lo que está ocurriendo en el mundo real. Por lo tanto, el CS se hace notar especialmente cuando el usuario se mueve de alguna forma dentro del entorno, por ejemplo, el caminar dentro de un videojuego de VR. Esto ha producido que a lo largo de los años uno

de los grandes focos de la investigación en el área esté relacionada con encontrar las causas específicas del CS y cómo mitigarlas.

Actualmente, para combatir el CS, se utilizan técnicas como la variación del campo de visión (FOV) [16] o el oscurecimiento de la pantalla cuando el usuario intenta moverse. Una de las formas de movimiento más comunes y amigables con el usuario, aunque poco inmersiva, es el oscurecer la pantalla hasta llegar a negro, para entonces mover instantáneamente la cámara al lugar deseado, para luego volver a aclarar la pantalla lentamente.

Encontrar una forma de moverse por un entorno de VR que genere menos CS podría abrir la puerta a que una mayor cantidad de personas se vea interesada por introducirse en el medio, al no tener la barrera de sentir malestar, logrando así un mayor desarrollo en el uso de la VR que podrá ser útil tanto para ocio como para desarrollo académico de sus usuarios y las otras áreas mencionadas.

En esta memoria se exploran las diferentes formas de representar movimiento en un entorno de VR, para compararlas y encontrar cuáles de estas inducen el menor CS, mientras conserva la inmersión. Las técnicas de movimiento (TM) que se ponen a prueba son seleccionadas entre las ya existentes y documentadas en la literatura, teniendo en cuenta que lo que se busca es encontrar factores que puedan causar el CS.

Para poner a prueba las técnicas de movimiento, se realiza un experimento en un entorno virtual. Mientras un sujeto de pruebas esté inmerso en el EV, se podrán registrar datos, tales como el *neurofeedback* obtenido gracias a un casco de electroencefalografía (EEG), y se interpretarán utilizando una interfaz cerebro-computador (BCI, por su sigla en inglés) pasiva.

Una BCI traduce la actividad cerebral permitiendo identificar estados mentales del sujeto tales como emociones, fatiga mental o estrés [1]. Se llama BCI pasiva a las BCI en las que no se requiere que el sujeto intente controlar su actividad cerebral a conciencia, a diferencia de como se haría, por ejemplo, al intentar controlar un brazo robótico con la actividad cerebral.

Se registra la autopercepción del malestar de los usuarios durante el experimento, utilizando un medidor que se puede mover a voluntad del sujeto y permite indicar la intensidad de CS en cada instante de la experiencia.

También se registra la autopercepción de los participantes posterior al experimento a través de un cuestionario. Estos datos sirven para identificar en qué momentos del experimento y con que forma de moverse se experimenta un mayor CS.

1.1. Objetivos

1.1.1. Objetivo Principal

El objetivo principal de esta investigación es obtener una caracterización de diferentes formas de simular movimiento en un entorno de VR y evaluar la intensidad de los síntomas de CS que generan, para encontrar las ventajas y desventajas de cada una ellas.

1.1.2. Objetivos Específicos

1. Conocer en mayor profundidad cuáles son los diferentes tipos de sistemas de movimiento utilizados en simuladores y videojuegos de VR.
2. Obtener un entorno de pruebas de VR que cuente con todas las técnicas de movimiento y en el que se puedan realizar de forma cómoda experimentos con personas inexpertas en el uso de equipo de VR. No puede tener problemas de rendimiento ni de latencia, y debe representar fielmente como se intenta mover el usuario.
3. Integrar una BCI pasiva que sea capaz de identificar los síntomas de CS.
4. Contar con un cuestionario apropiado para observar la autopercepción de las personas que participen en el experimento.
5. Obtener una clasificación con base en parámetros objetivos de todas las formas de moverse en VR propuestas.
6. Encontrar factores que intensifiquen o que mitiguen el CS al moverse en un EV.

1.2. Evaluación

Se podrá evaluar el trabajo realizado basándose en los resultados de la clasificación final que se obtenga, es decir, en primer lugar se evaluará si es que fue posible generar la caracterización y también se evaluará si las diferencias entre las técnicas son significativas o no.

La evaluación del entorno de pruebas se basará en su capacidad para llevar de forma correcta el experimento. Esto incluye la entrega los datos de forma correcta, que no dependa del examinador para que los sujetos sepan qué hacer y su optimización para garantizar una buena frecuencia de fotogramas por segundo.

1.3. Estructura de la memoria

Lo que resta de este documento se dividirá por capítulos siguiendo la siguiente estructura:

- En el segundo capítulo se muestra el estado del arte respecto a equipos de VR, técnicas de movimiento en entornos de VR, investigación del cybersickness junto con maneras de medirlo y se exponen los resultados obtenidos por pruebas iniciales en busca de indicadores de cybersickness.
- En el tercer capítulo se explican temas cuyo conocimiento previo es necesario para comprender algunas terminologías y la estructura general de la solución. Entre estos temas se encuentra Unity y el uso específico del BCI escogido.

- En el cuarto capítulo se realiza un análisis de lo que se requiere para cumplir con los objetivos, se definen requisitos tanto para la aplicación a desarrollar como del experimento.
- En el quinto capítulo se explica la implementación del entorno de pruebas que se usa para el experimento. Dando detalles de la estructura del código y explicando las particularidades y consideraciones para cada tipo de movimiento.
- En el sexto capítulo se relata el desarrollo del experimento y el método utilizado, para finalizar con una breve reflexión respecto a las dificultades encontradas en el mismo.
- En el séptimo capítulo se revelan los resultados obtenidos y también se muestra la metodología para analizarlos.
- En el octavo capítulo se hace un análisis y reflexión respecto a los resultados obtenidos en el experimento. Revelando las intuiciones obtenidas sobre las diferentes TM, las formas de medir utilizadas y el propio experimento.
- En el noveno capítulo se discuten las posibles maneras de desarrollar trabajos a futuro basándose en esta memoria.
- En el décimo y último capítulo se presentan las conclusiones de todo el trabajo y se expone posibles trabajos a futuro con base en la solución desarrollada.

Capítulo 2

Estado del arte

2.1. Realidad virtual

2.1.1. Equipo de VR

En la actualidad existen múltiples dispositivos de VR disponibles en el mercado, estos se pueden dividir en dispositivos para teléfono móvil, autónomos, para computadora y para consola. [11]

Mientras que los dispositivos autónomos utilizan su propio procesador y tarjeta gráfica, los pertenecientes a las otras tres categorías dependen de su respectivo sistema para funcionar. Por ejemplo, un dispositivo de VR para computadora no podrá funcionar sin estar conectado a una computadora que procese la aplicación que se quiera usar, lo mismo es cierto para los dispositivos de teléfono móvil y de consola.

Para el desarrollo de una aplicación, lo más útil será un dispositivo autónomo o de computadora, sin embargo, los autónomos tienden a ser más económicos y versátiles, pues algunos también pueden conectarse a una computadora.

En cuanto a dispositivos autónomos, la opción más reconocida son los Meta Quest 1, 2, 3 y Oculus Go. Siendo los Quest 2 y 3 los que permiten, además, conectarse a una computadora.

Además de los clásicos controles y visor, también existen más periféricos que permiten interactuar de formas únicas con los EV. Entre estos está el Omni One [36] que permite moverse libremente en el mundo real manteniendo al usuario en su lugar, para una experiencia de movimiento completo.

Se contará con el dispositivo Quest 2 debido a su capacidad de funcionar sin estar conectado a una computadora a través de un cable, siendo este capaz de conectarse vía Wi-Fi, lo que facilitará su uso durante el experimento. El Quest 2 cuenta con un visor y dos controles para las manos, como se ve en la figura 2.1.



Figura 2.1: Meta Quest 2 con sus componentes.

2.1.2. Motores para desarrollar en VR

Existen múltiples opciones para el desarrollo de aplicaciones de VR, entre estas están Unity3D, Unreal Engine, Godot y CryEngine. Siendo que de todas las opciones disponibles, las primeras tres las que el estudiante ha utilizado con anterioridad.

Se decidió descartar opciones como Godot y CryEngine debido a que la comunidad y documentación respecto al trabajo con VR es todavía pequeña en comparación a Unity o Unreal. Por otro lado, Unreal Engine también se descartó debido a la exigencia de hardware que tiene.

Unity3D fue seleccionado como la mejor opción no solo por ser el motor con el que se cuenta con mayor maestría, sino que también por ser el con mayor información disponible en lo que a VR respecta y además por contar con una implementación del BCI que se utilizará en el experimento.

2.1.3. Herramientas de Unity para VR

Actualmente, la solución oficial de Unity para desarrollo de aplicaciones de VR es el *XR Interaction Toolkit* [34] (XRIT), siendo XR un término que significa realidad extendida, el cual sirve para englobar la VR, realidad aumentada y realidad mixta [15, 33].

El XRIT provee un framework el cual permite recibir los eventos de la interacción de los controles de un dispositivo de VR con el entorno en 3D de una aplicación de Unity o también con una interfaz. Además, el XRIT es compatible con los dispositivos Meta Quest.



Figura 2.2: Videojuego Beat Saber, imagen extraída de video de usuario de YouTube Omotea [29]

2.1.4. Técnicas de movimiento en el medio de videojuegos

Como es de esperar, una de las mayores aplicaciones que tienen los entornos de VR es en los videojuegos. Por lo tanto, también es importante revisar las diferentes maneras de moverse presentes en este medio. En esta sección se mencionarán las técnicas de movimiento menos comunes encontradas.

Existe una técnica de movimiento en la que el jugador puede moverse en un espacio confinado utilizando su cuerpo real. Un ejemplo de esto es el videojuego Beat Saber (Fig. 2.2). En este juego, debes golpear rítmicamente bloques y, en ocasiones, el jugador debe esquivar murallas que se mueven en su dirección. El movimiento realizado en el entorno virtual será el mismo de la vida real, por lo que está la opción de moverse hacia los lados para esquivar o incluso agacharse y saltar. Sin embargo, no está permitido alejarse demasiado de un área definida.

En el videojuego llamado Epic Roller Coasters (Fig. 2.3), se encuentra la locomoción en rieles, en donde el jugador no puede moverse por su cuenta, sino que está obligado a avanzar de forma automática en por un recorrido predefinido. Esta técnica de movimiento puede ser útil en caso de que se quiera tener un estudio más controlado. Para calcular la posición del jugador se usa la posición del casco de realidad virtual relativa a su posición original.

También existen TM que dependen completamente de las físicas del entorno y que permiten desplazarse por el aire, un ejemplo de esto es el videojuego Gorilla Tag 2.4, en donde los jugadores personifican a un gorila sin piernas, el cual utiliza los brazos para desplazarse. En este caso el usuario debe mover los controles que simulan ser sus manos para que es-



Figura 2.3: Perspectiva de juego de Epic Roller Coasters, extraído de página oficial en Steam [31]

tas impacten con el terreno del mundo virtual, dependiendo de la fuerza con la que golpee, será impulsado en la dirección contraria con una fuerza proporcional, permitiendo al jugador incluso saltar por los aires.

2.1.5. CS y técnicas de movimiento en la literatura

En cuanto a información encontrada en publicaciones científicas, en el estudio de Cherni et al. [7] se revisaron las técnicas de movimiento encontradas a lo largo de 26 publicaciones diferentes, clasificándolas en las que se basaban en el movimiento del cuerpo de los sujetos, las que dependen totalmente de periféricos externos, y las que mezclan ambas opciones.

Ejemplos de las basadas en movimiento son la caminata real, caminata en una posición fija, teletransporte usando gestos de la mano, moverse según la inclinación de la cabeza, balancear los brazos para avanzar, entre otros.

Las TM dependientes de periféricos incluyen moverse con una palanca, teletransporte con un botón de los controles del dispositivo de VR, usar un pedal de aceleración, usar una patineta con vibración o una mezcla entre inclinación del cuerpo y movimiento de palanca.

Se encontró que la caminata en una posición fija, sin periféricos, induce menos CS que la caminata en la que solo se mueven los brazos, aunque produce CS de todos modos. También se puede ver que el teletransporte no presenta CS, debido a que este no simula un movimiento, sino que simplemente mueve al usuario al lugar deseado de forma instantánea. Por el contrario, el uso de periféricos como el Omni One no mitigan el CS de forma notoria.



Figura 2.4: Jugador de Gorilla Tag desplazándose, extraído de la página oficial en Steam [32]

Existe un experimento realizado por Suma et al. [2] donde se compara el CS autopercibido aplicando un cuestionario utilizado para detectar malestar en simuladores, llamado *simulator sickness questionnaire* (SSQ). El cuestionario se utilizó para comparar el movimiento de caminata real, caminata simulada y apuntar para avanzar. Finalmente, se determinó que no existe una diferencia significativa del CS al usar cualquiera de esas técnicas.

También existe un estudio realizado por David Bond y Madelein Nyblom, [4] en el que se pusieron a prueba cuatro tipos de movimiento diferentes, donde se encontró que el que mayor CS generaba era el llamado *Point-Tugging*, el cual consiste en sujetar un punto del espacio 3D y hacer el gesto de tirar para desplazarse, sin embargo, no se sabe con certeza el motivo de que induzca más CS, aunque se sospecha que puede ser debido a que esta técnica permite moverse en una dirección diferente a la cual se está mirando. Además, se encontró que caminar balanceando los brazos, podía producir CS, a pesar de lo estipulado por Coomer et al. [8] quienes estudiaron esta misma técnica de movimiento. En el estudio de Bond, se hace notar la posibilidad de confundir síntomas de CS con cansancio físico o con la sudoración producida por el tiempo de uso del equipo de VR.

Se sabe que la causa principal del CS es la ilusión de movimiento sentida por el usuario y que este se puede detectar en la postura y movimiento, incluso previo a sentir los síntomas [24], además, que ha descubierto que aumenta a un 60% las probabilidades de comenzar a sentir síntomas de CS luego de 20 minutos de exposición a la VR [12].

Existen varios factores que pueden facilitar a que se presenten los síntomas de CS, como el hecho de usar lentes de realidad virtual (HMD, por sus siglas en inglés), además de la latencia del movimiento de cabeza [30, 5], que significa el tiempo que tarda en mostrarse en la simulación el movimiento hecho por el usuario.

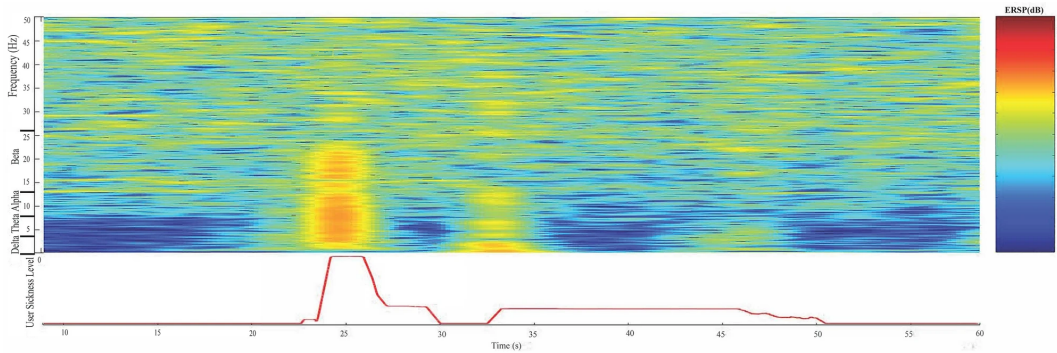


Figura 2.5: Gráfico mostrando relación entre ondas Alpha, Delta y Theta con CS, extraído del trabajo de Krokos et al. [23]

Se conocen varias formas de mitigar el malestar en un EV, como la manipulación del FOV y la estimulación sensorial [16, 19].

2.2. Medición de CS con BCI

Se han podido medir los efectos del CS a través de la EEG [23] y utilizando cuestionarios de autopercepción, como el SSQ y otros, como el *virtual reality sickness questionnaire* (VRSQ) propuesto por H.K. Kim et al. [21], un cuestionario especializado en identificar los síntomas de la cinetosis clásica en entornos de realidad virtual.

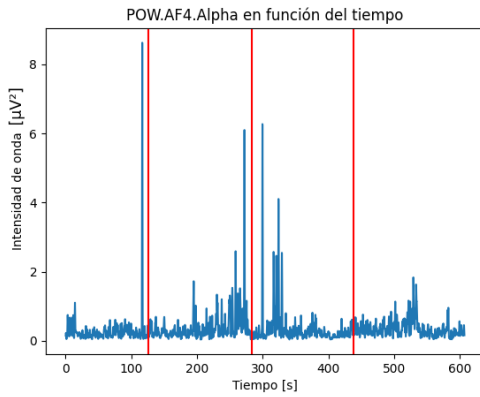
También se ha logrado medir la somnolencia [35] y el estrés con una BCI pasiva [20].

Aunque se detallará con mayor precisión en el tercer capítulo, es importante señalar que las ondas cerebrales pueden clasificarse según su banda de frecuencia. Las ondas Delta (0.5 - 4 Hz), Theta (4 - 8 Hz) y Alpha (8 - 12 Hz) son las de frecuencia más baja. En el estudio de Krokos et al. [23], se encontró una fuerte correlación entre estas tres ondas cerebrales y los picos de malestar autopercebidos por los sujetos de prueba, como se muestra en la figura 2.5, que evidencia un aumento en la actividad de estas ondas. Además, el estudio descarta que el movimiento periódico de la palanca de un controlador durante el experimento influya en los resultados, lo que sugiere la posibilidad de utilizar este movimiento para registrar la autopercepción de los sujetos.

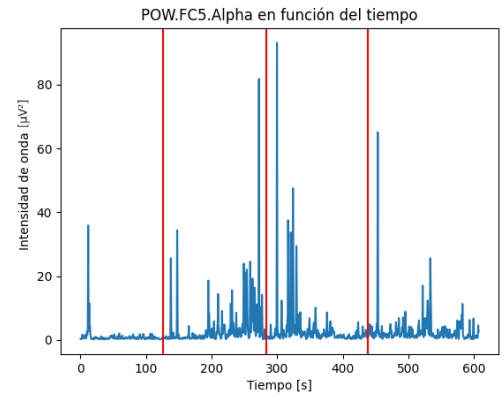
2.2.1. Trabajo previo

Junto con la investigación, también se realizó un breve trabajo previo con el objetivo de confirmar la relación de las ondas mencionadas anteriormente con el mareo.

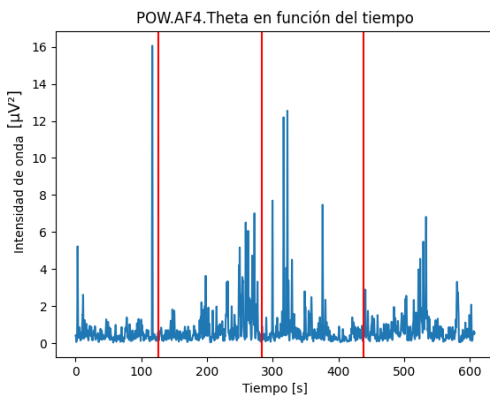
Para hacer el experimento, el estudiante se mantuvo sentado mientras utilizaba un visor de realidad virtual junto con un casco de lectura de ondas cerebrales. Se comenzó a grabar el *neurofeedback* del estudiante durante 10 minutos, al mismo tiempo que este experimentaba



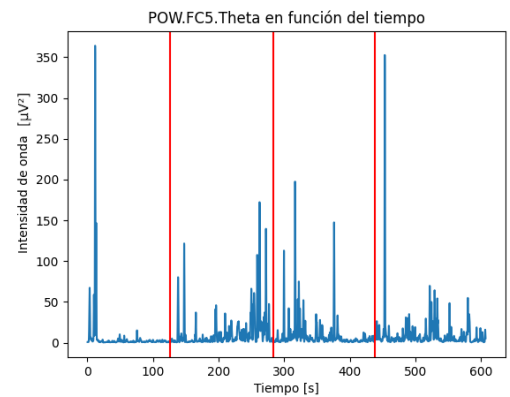
(a) Gráfico de poder en banda Alpha del diodo AF4.



(b) Gráfico de poder en banda Alpha del diodo FC5.



(c) Gráfico de poder en banda Theta del diodo AF4.



(d) Gráfico de poder en banda Alpha del diodo FC5.

Figura 2.6: Gráfico de dos diodos que parecen relacionarse con el CS.

tres vueltas en una montaña rusa simulada de Epic Rollercoasters 2.3.

Al lado del estudiante, se encontraba un voluntario que, manteniendo silencio, se encargó de anotar cualquier comentario por parte del estudiante junto a la marca temporal del momento en el que fueron emitidos, pues durante el desarrollo del experimento, el estudiante emitió varios comentarios en los momentos que consideró pertinente, como cuando este sentía malestar o cuando terminaba una de las vueltas de la montaña rusa.

En las siguientes figuras se pueden apreciar tres líneas rojas verticales. Estas fueron utilizadas para marcar los comentarios emitidos por el estudiante durante la prueba. La primera línea roja indica un momento en el que el estudiante dijo “Me siento mal”, indicando que sentía gran malestar. La segunda línea roja indica el comienzo de la caída de la segunda vuelta del recorrido de la montaña rusa. La última línea indica otro momento en el que el estudiante declaró sentirse muy mal.

El resultado entregado por algunos electrodos de la banda Alpha y Theta que parecen relacionarse con el cybersickness se muestran en la figura (Fig. 2.6).

De aquí, podemos confirmar que, efectivamente las ondas Alpha y Theta parecen aumentar en los momentos en el que el estudiante indico sentir malestar. Sin embargo, también se encuentran varios picos que no parecen tener explicación.

Capítulo 3

Antecedentes

3.1. Unity

El motor de videojuegos seleccionado fue Unity, debido a su facilidad para el desarrollo de aplicaciones de realidad virtual y a la gran cantidad de documentación disponible en línea respecto al tema. Sin embargo, es necesario entender el funcionamiento general del motor para poder comprender las decisiones y la estructura de la implementación.

3.1.1. GameObjects y Componentes

El sistema que usa Unity está basado en el concepto de GameObjects, es decir, estos son la unidad básica de cualquier aplicación [9] que se desarrolle. Un GameObject es la representación de cualquier entidad en una escena y, en caso de que se quiera utilizar código que interactúe con el entorno, este siempre debe estar asociado a un GameObject.

Se utiliza una organización de árbol, en las que cada GameObject puede tener otros como hijos, heredando algunas propiedades y funcionalidades, como lo puede ser las transformaciones.

Cada uno de estos GameObjects está formado por diferentes componentes, los cuales determinan la funcionalidad que tendrá. Por ejemplo, en la figura 3.1 se puede ver el GameObject de una llave.

En esta figura se puede ver como al lado izquierdo se muestra el árbol de jerarquía con todos los objetos presentes en la escena y, en específico, el objeto de la llave que está compuesto por los objetos *key* y *model*. En el lado derecho, se pueden ver los componentes que tiene el GameObject principal, entre los que está *Transform* que indica las transformaciones, un *SphereCollider* que marca el volumen en el que se detectan colisiones con la llave y finalmente un componente *KeyBehaviour* que es un script de C# en el que se especifica el comportamiento de la llave, como su forma de moverse y qué hacer cuando el usuario colisiona con ella.

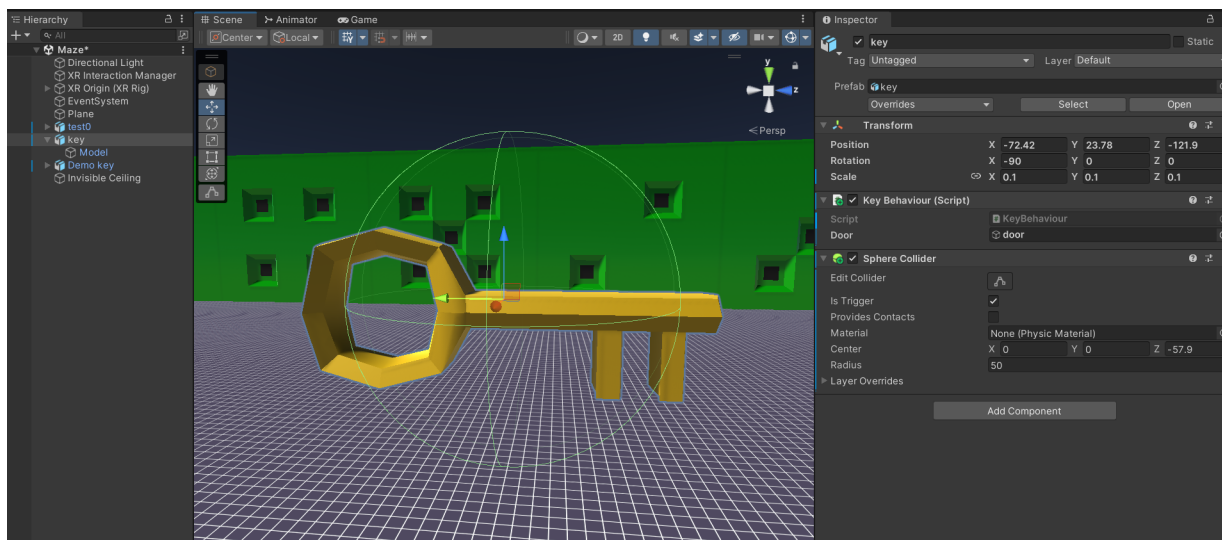


Figura 3.1: GameObject de una llave y sus componentes

También en la figura 3.2 se pueden ver los componentes del GameObject hijo, que representa el modelo de la llave, compuesto por la malla de polígonos y el material.

3.1.2. MonoBehaviour

El MonoBehaviour es la clase base asociada a todo GameObject de Unity y también es la clase que debe extender cualquier script de C#. Este incluye métodos que permiten manejar los objetos, como destruirlos, habilitarlos, deshabilitarlos, etc. También posee métodos que se pueden sobrescribir por las clases que hereden de esta, como el método Update() el cual es llamado cada fotograma, o métodos llamados en la inicialización como Awake() y Start().

3.1.3. Transform

Todos los GameObjects cuentan además con el componente Transform, este componente mantiene información como la posición, rotación y escala de un objeto, es decir, la transformación. Accediendo y modificando los valores de este componente se puede generar movimiento en un GameObject.

3.1.4. Acciones de input

Unity cuenta con un sistema de acciones para manejar el input del usuario. Este sistema permite asociar múltiples opciones de tecla, botón o incluso inputs más complejos, como el giroscopio de algún dispositivo, a una misma acción (Fig. 3.3). De esta forma se obtiene un sistema de controlador extensible.

Luego de agregar las acciones deseadas, se genera automáticamente un script de C# que

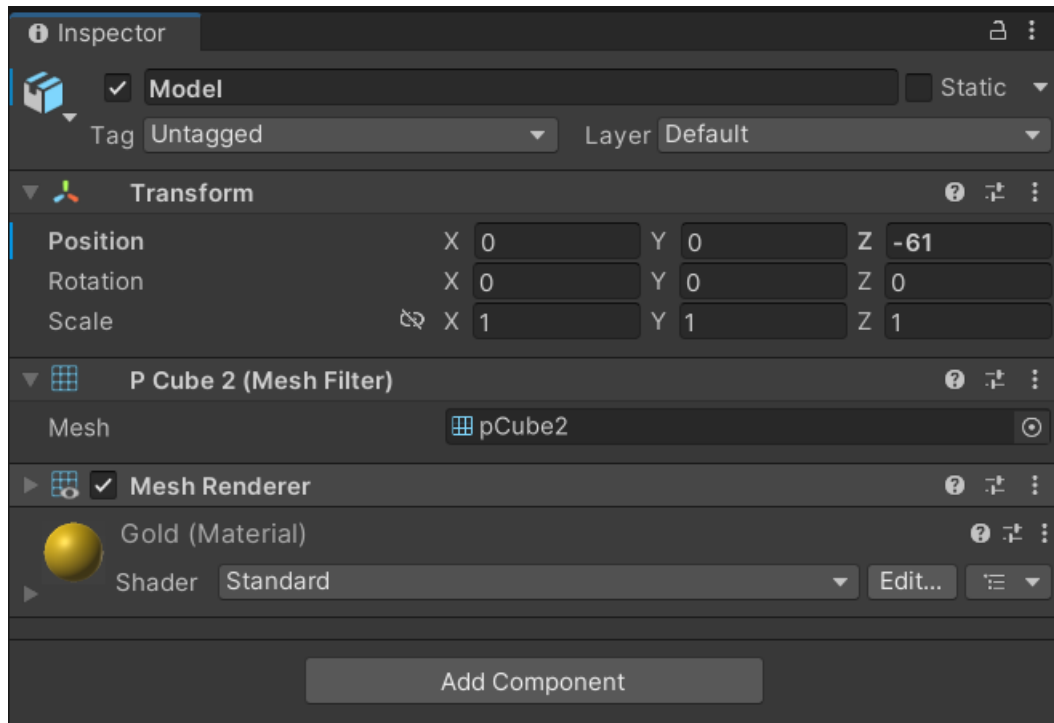


Figura 3.2: Componentes del modelo del objeto llave

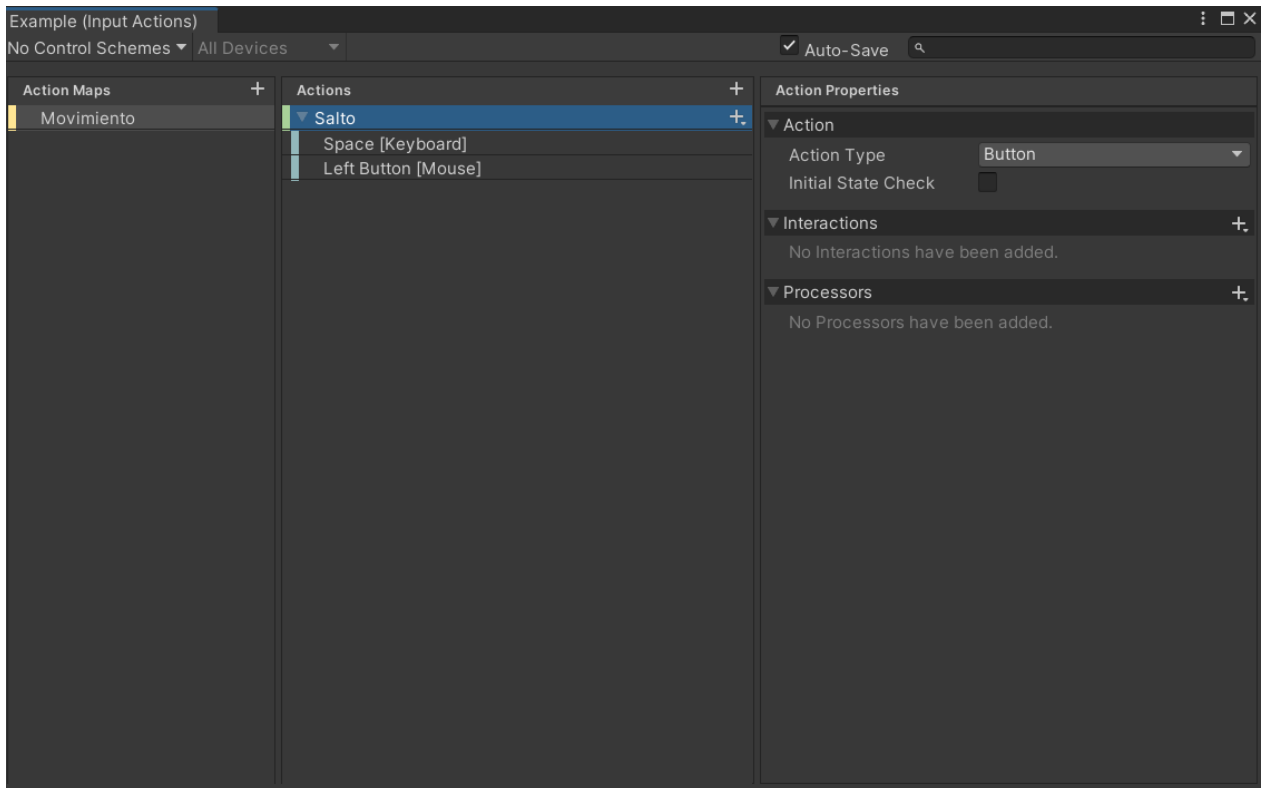


Figura 3.3: Menú de acciones de Unity

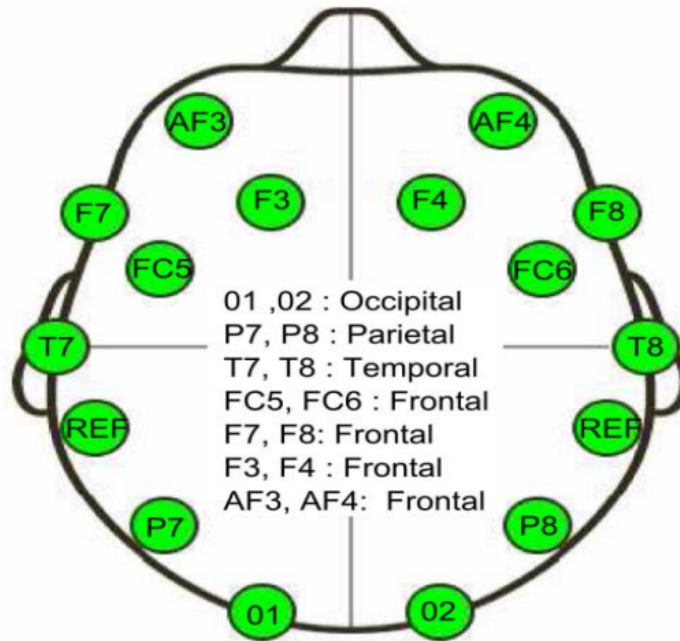


Figura 3.4: Distribución de los diodos del Epoc+, extraído del trabajo de Hou et al. [37]

expone funciones para cada vez que una acción se ejecuta o cancela, en el ejemplo de la figura 3.3, generaría un evento llamado *salto.performed* y *salto.canceled*, permitiendo que luego, en un script se pueda suscribir algún método deseado al evento correspondiente. Siguiendo el ejemplo brindado por la figura, se podría ejecutar el mismo método que haga saltar el GameObject del usuario al presionar tanto la barra espaciadora como al hacer clic izquierdo con el mouse.

3.2. Emotiv

Actualmente, los dispositivos de EEG disponibles en el mercado son los de InteraXon, Muse, Neurosky, OpenBCI y Emotiv [25]. De estos, siendo los de OpenBCI y Emotiv los que pueden entregar más información, pues cuentan con hasta 16 canales cada uno. Los canales son las posiciones en las que se pueden poner los electrodos para obtener los datos de ondas electromagnéticas.

Emotiv entrega también herramientas de investigación accesibles desde Windows, Mac y Linux, por lo que se decidió trabajar con su dispositivo de 16 canales Epoc+.

Los 16 electrodos del Epoc+ se distribuyen como se ve en la figura 3.4. Cada uno de estos electrodos puede recibir una diferente lectura de las ondas cerebrales para un mismo estímulo.

Cortex API de Emotiv

Al realizar una grabación EEG, los datos obtenidos se guardan automáticamente en una base de datos de Emotiv en archivos conocidos como *records*. Para acceder a ellos, se debe hacer a través de la API brindada por la misma empresa, llamada Cortex API, la cual cuenta con las consultas necesarias para crear *records*, consultar los creados por el usuario y exportarlos a un computador local.

3.2.1. Bandas de frecuencia

Emotiv también ofrece una solución que, utilizando Unity, entrega un ejemplo de cómo interactuar con la API. Esta solución, llamada Cortex-Example [13] cuenta con lo necesario para iniciar una grabación por una cantidad fija de tiempo y exportarla al haber terminado, permitiendo escoger que tipos de datos se quieren obtener, como las mediciones de aceleración, el poder las bandas de frecuencia en cada electrodo o las *performance metrics* que son explicadas más adelante.

Las ondas cerebrales de una persona se pueden dividir según su banda de frecuencia, siendo separadas en las Delta, Theta, Alpha, Low Beta, High Beta y Gamma, las cuales se pueden describir de la siguiente manera, basado el trabajo de Kaminska et al. [20].

1. Delta (0.5 - 4 Hz): Estado de sueño profundo sin soñar. Demasiada cantidad pueden indicar daño cerebral.
2. Theta (4 - 8 Hz): Se encuentran al dormir. También pueden ocurrir durante meditación profunda, al estar muy concentrado y al estar bajo mucho estrés.
3. Alpha (8 - 12 Hz): Relacionadas con el estado de relajación. Desaparecen al hacer esfuerzo mental como ejercicios matemáticos.
4. Low beta (15 - 18 Hz): Relacionadas con un estado de concentración intensa, pensamiento lógico y control de emociones.
5. High beta (18 - 28 Hz): Se encuentran al tener un estado de nerviosismo, emoción o irritación. En personas saludables puede ser señal de pánico.
6. Gamma (30 - 200 Hz): Se relaciona la amplitud de estas ondas a cuando el sujeto se enfoca en la fuente de un estímulo. Demasiadas pueden indicar estrés y demasiado pocas pueden indicar depresión.

3.2.2. Performance metrics

Emotiv, junto a la lectura de las bandas de frecuencia, entrega una medida llamada las *performance metrics* [28], las cuales son obtenidas con base en las lecturas de las bandas, y describen el estrés (Stress), retención (Engagement), interés (Interest), emoción (Excitement), atención (Attention) y relajación (Relaxation).

Las *performance metrics* son adaptables para cada usuario, eso significa que mientras más veces una misma persona utilice una misma aplicación de Emovit, se podrán obtener con mayor precisión.

Capítulo 4

Análisis y Diseño de la solución

En este capítulo se discute el diseño que sirve como base para la creación de una aplicación y un experimento que utilice dicha aplicación que tengan la capacidad de obtener la información necesaria para cumplir con el objetivo principal de esta memoria. Es decir, la correcta implementación y aplicación de la solución propuesta debe permitir obtener una noción de las diferencias de múltiples técnicas de movimiento al momento de producir cybersickness.

4.1. Diseño del experimento

Para poder obtener información de diferentes técnicas de movimiento se realizará un experimento en el que, utilizando una técnica de movimiento previamente seleccionada por el examinador, un sujeto de pruebas deba completar un desafío sencillo dentro de un escenario virtual hecho para VR.

Con el fin de poder hacer comparaciones, se debe contar múltiples TM diferentes junto con diferentes escenarios para ponerlas a prueba. Se propone tener 4 técnicas de movimiento y 3 escenarios.

Se espera obtener una muestra de por lo menos tres personas por cada combinación, de forma de que los datos sean comparables. De este modo se requiere un total de 32 personas para cubrir todas las opciones tres veces, aunque más adelante en el capítulo se explicará por qué este número será reducido.

4.1.1. Obtención de datos

Se deben tener múltiples formas de obtener datos de los sujetos, implementadas tanto dentro como fuera de la aplicación.

La aplicación desarrollada debe permitir una obtención de datos sencilla y que no facilite errores de medición. Obteniendo y almacenando de forma automática los datos que sean

posibles de obtener directamente a través de la aplicación.

Finalmente, se planifican tres formas de obtener datos.

Cuestionario de cybersickness

Se le solicitará a los voluntarios rellenar un formulario posterior a la realización del experimento. En este cuestionario se busca identificar que síntomas asociados al CS son autopercebidos por los sujetos.

Autopercepción durante el experimento

Dentro del entorno de pruebas, debe existir alguna forma de que el usuario esté indicando en tiempo real el malestar autopercebido, se propone un medidor con el que se pueda interactuar en todo momento para cumplir con esto.

Datos cerebrales

También se debe poder obtener datos de las ondas cerebrales de los usuarios mientras estén expuestos a los escenarios virtuales. Para esto se debe hacer una implementación de un BCI en la aplicación a desarrollar.

4.2. Requisitos del entorno de pruebas

Con el fin de obtener los datos necesarios para caracterizar las diferentes técnicas de movimiento, se debe desarrollar un entorno de pruebas que permita realizar un experimento de forma cómoda. Para esto se debe desarrollar un entorno el cual logre cumplir con la siguiente lista de requisitos técnicos.

1. Deben existir por lo menos tres técnicas de movimiento suficientemente diferentes, las cuales puedan ser seleccionadas y utilizadas en cada una de las pruebas.
2. Deben existir diferentes pruebas, las cuales sean posibles de superar en un tiempo no mayor a 20 minutos en promedio, esto con el fin de que el experimento no se extienda durante demasiado tiempo.
3. Debe existir un menú inicial el cual permita a los sujetos de prueba seleccionar una técnica de movimiento y un escenario de forma sencilla.
4. El usuario debe ser capaz de mover un medidor de mareo para indicar su condición autopercebida en todo momento.

5. Es necesario que permita comenzar la grabación de datos cerebrales del al mismo tiempo que se estén almacenando, de forma que no se interrumpa el experimento ni se deba extraer al sujeto de pruebas del EV.

4.3. Diseño del entorno del pruebas

Con el fin de poder poner a prueba todas las técnicas de movimiento en las mismas condiciones, se decidió desarrollar una aplicación de VR con todos los escenarios y TM necesarias. De este modo, se tiene un mayor control sobre las diferentes variables, se facilita la extracción de los datos y además se puede hacer una mejor integración con la aplicación del BCI.

4.3.1. Técnicas de movimiento

Inicialmente, se consideraron cuatro técnicas de movimiento que contaban con características únicas que potencialmente podrían desencadenar en una diferencia respecto a la fuerza con la que generan síntomas de cybersickness en sus usuarios.

En la figura 4.1 se pueden ver los primeros bocetos simples hechos para cada una de las TM para planificar su funcionalidad básica.

Caminata estándar

Siendo la más simple de todas, la idea inicial de esta TM es permitir al usuario caminar y correr utilizando un joystick en cualquier dirección. Se consideró debido a que es de las formas de movimiento más comunes y servirá para tener una base con la que comparar las demás.

Propulsión

Consiste en una TM que permite al usuario avanzar de forma vertical y horizontal con propulsión similar a la entregada por un cohete. Se distingue, pues es un movimiento menos controlado que el de caminata estándar y, por lo tanto, podría llegar a generar malestar.

Gancho

En esta TM se simula el movimiento generado por tirar de un gancho adherido a alguna superficie, es en teoría el movimiento más complicado de manejar y que mezcla necesariamente movimiento aéreo con terrestre, por lo que se esperaría un mayor malestar.

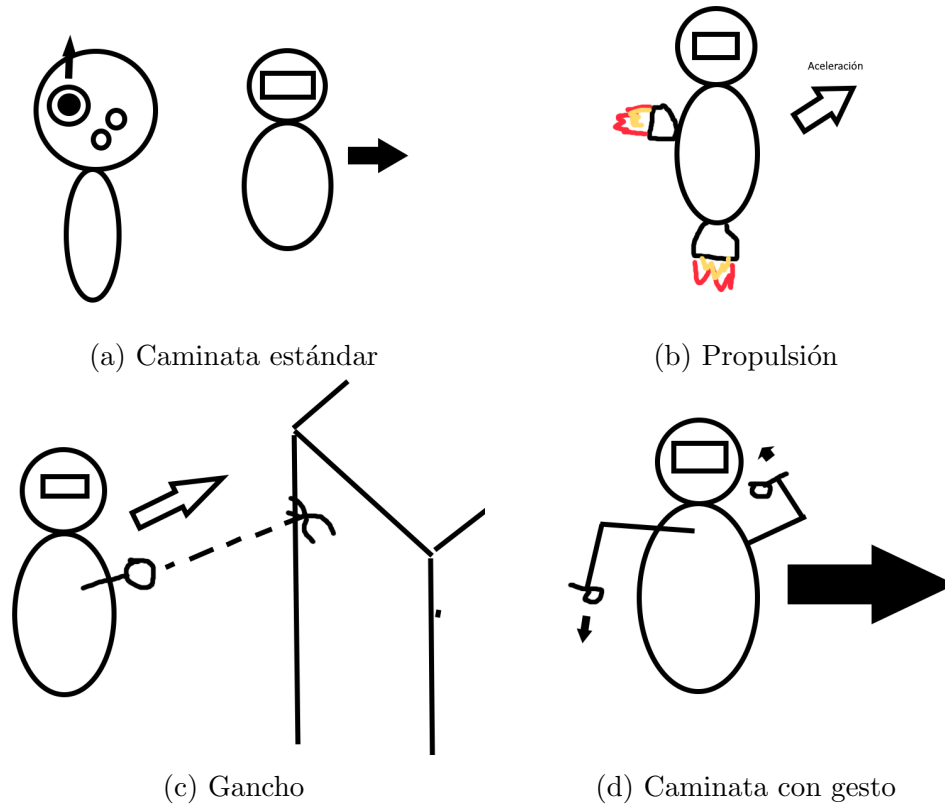


Figura 4.1: Bocetos de las técnicas de movimiento desarrolladas

Caminata con gesto

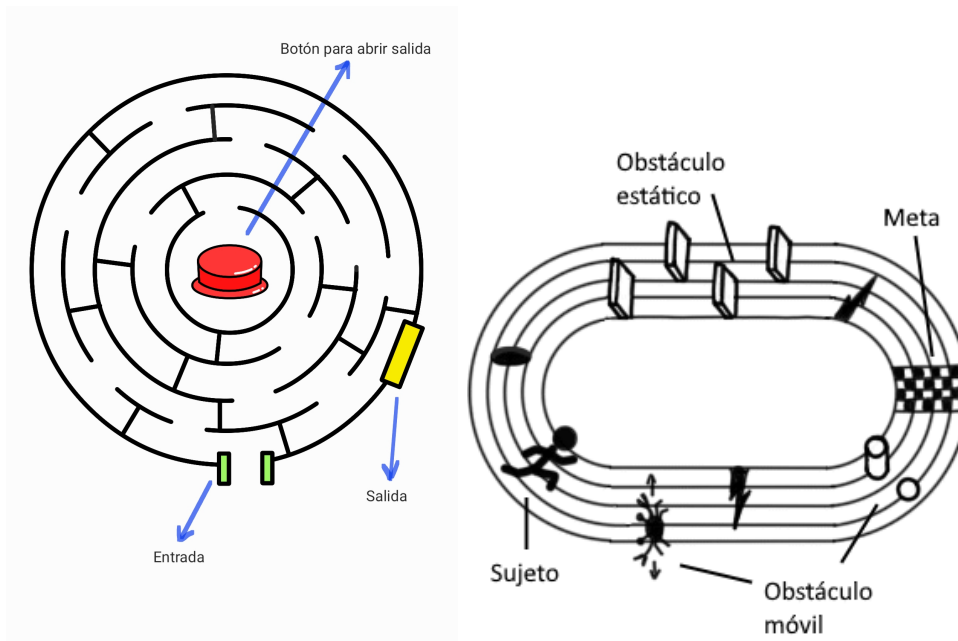
Similar a la caminata estándar, sin embargo, la forma de recibir el input para avanzar, en lugar de usar un botón, es a través del movimiento de los brazos del usuario. Se cree que esta TM puede generar una menor sensación de cybersickness al hacer que el cuerpo del usuario entre en mayor sintonía con la simulación.

4.3.2. Escenarios

Se diseñaron tres escenarios diferentes, los cuales tienen como objetivo impulsar al usuario a moverse de una forma en específico al pedirle completar una prueba dentro de los mismos. En la figura 4.2 se pueden apreciar los primeros bocetos de cada escenario.

Laberinto

Este escenario consiste en un gran laberinto, en donde la meta es encontrar la salida, luego de habilitarla al encontrar un objetivo, como una llave o un botón situado en el centro del laberinto. Este laberinto busca crear la necesidad en el sujeto de cambiar de dirección un gran número de veces.



(a) Laberinto

(b) Pista de carreras



(c) Mundo abierto

Figura 4.2: Bocetos de los escenarios que se planifican desarrollar

Pista de obstáculos

En este escenario, el usuario debe avanzar por una pista de atletismo, esquivando obstáculos situados sobre la misma. El objetivo de esta prueba es hacer que el sujeto deba moverse lateralmente de forma constante para poder avanzar y que no pueda tener un movimiento continuo.

Mundo abierto

Este escenario se ubicará en una amplia zona abierta, en donde se le pedirá al sujeto alcanzar múltiples puntos de control para completarlo. En esta prueba se busca dar una mayor libertad al sujeto al no forzarlo a estar encerrado en un camino predefinido.

4.3.3. Reducción del scope

El comité de ética de la Universidad de Chile solicitó la presencia de personal médico durante las pruebas, dificultando en gran medida la coordinación de horarios y obtención de voluntarios, por lo tanto, se consideró la posibilidad de que se contaría con un bajo número de sujetos de prueba.

Se decidió disminuir la cantidad total de experimentos realizados para satisfacer la baja en voluntarios, por lo que finalmente el escenario de mundo abierto y la TM de propulsión fueron descartadas del experimento. De esta forma, en lugar de tener 12 combinaciones diferentes de escenario con TM, se tienen 6, esperando un total de 18 experimentos.

El mundo abierto se descartó al ser el escenario menos controlado de todos, implicando que sería el que resultaría más difícil de analizar correctamente debido a la gran libertad que tendrían los sujetos de prueba al explorarlo. También añade variables extra a considerar al contar con una gran variedad de modelos en 3D y diferencias en el terreno.

La propulsión fue la TM desechada debido a que se consideró como la TM con menos aspectos únicos. Por ejemplo, el uso de fuerzas difíciles de controlar y movimiento vertical ya viene dado por el gancho, y el uso de las palancas para moverse también se encuentra en la caminata estándar.

Capítulo 5

Implementación

5.1. XR Interaction Toolkit

El XR Interaction Toolkit (XRIT) es un paquete oficial de Unity que provee un sistema de interacción para crear experiencias en VR y AR. Este incluye componentes de Unity que al ser añadidos al GameObject del jugador, se encargan de añadir la funcionalidad básica de VR, por ejemplo, se encargan del manejo del movimiento de la cámara del usuario al mover el casco de VR y también se encarga del movimiento de los controladores, mostrando por defecto un modelo 3D de estos que siguen el movimiento de los controladores en el mundo real.

Junto a los objetos, incluye además un conjunto de acciones asociadas a los botones de un controlador de VR genérico, pues el XRIT está diseñado de forma que se pueda utilizar independientemente del equipo con el que se cuente. Los botones disponibles se pueden observar en el diagrama de la figura 5.1.

Junto con los botones mostrados en la figura, el paquete también incluye las acciones asociadas a la posición, variación de posición, rotación y variación de rotación de los controladores y del casco en el espacio.

Para mantener la extensibilidad de la solución, se respetará la estructura ofrecida por el XRIT en la creación de técnicas de movimiento, la cual se explicará más adelante en la sección de técnicas de movimiento.

5.1.1. GameObjects

En la figura 5.2 se puede ver el diagrama que representa el árbol del GameObject que se utiliza para representar al usuario.

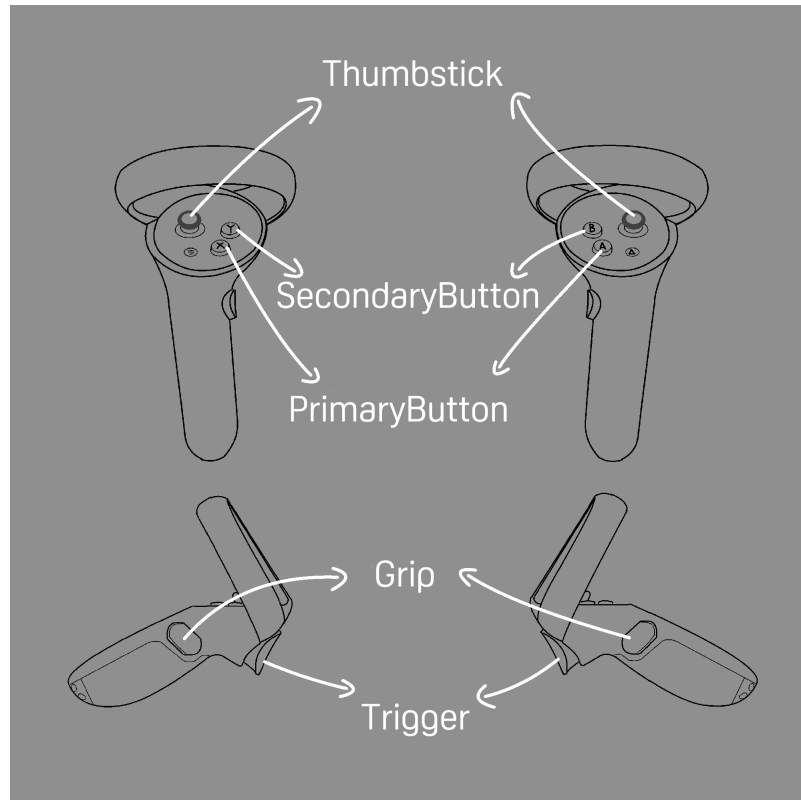


Figura 5.1: Diagrama de botones disponibles en el XRIT

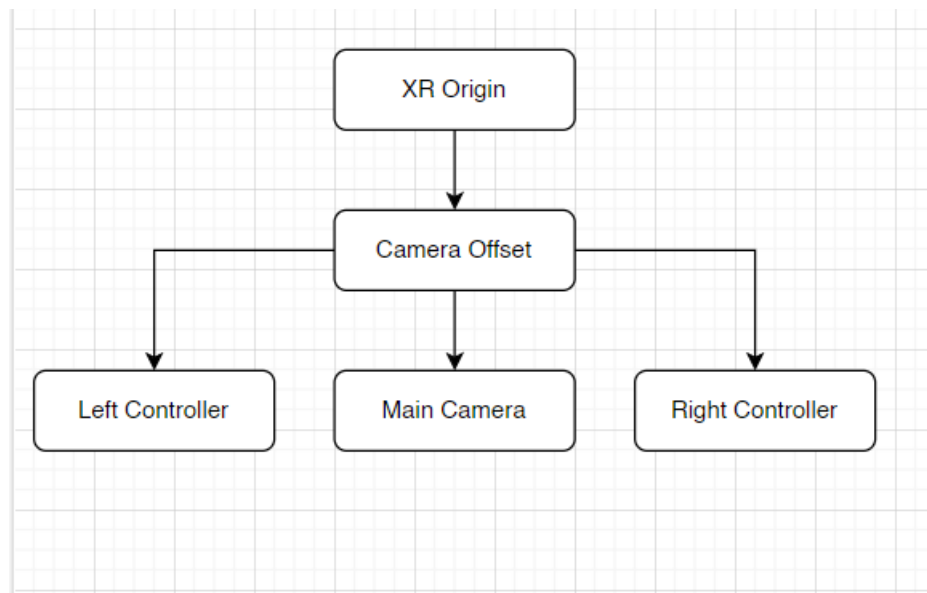


Figura 5.2: Diagrama de árbol de objetos del jugador

5.2. Técnicas de Movimiento

5.2.1. Caminata estándar

La caminata estándar viene incluida por defecto en el paquete de XRIT de Unity, por lo que es un buen momento para explicar la estructura original con la que el paquete maneja las técnicas de movimiento, las cuales para mantener consistencia con el nombre de las clases, serán llamadas locomociones desde ahora en adelante.

La clase `LocomotionProvider` es la clase base para la implementación de las locomociones. Esta mantiene el estado de las locomociones para asegurar que no se esté intentando mover el XR Origin por dos locomociones diferentes al mismo tiempo.

Ofrece los métodos protegidos `BeginLocomotion()`, `CanBeginLocomotion()` y `EndLocomotion()` los cuales deben ser utilizados por las clases que extiendan `LocomotionProvider` para marcar el inicio y el fin de un movimiento.

Utilizando esta base, es que se implementa el `ContinousMoveProviderBase` y su clase hija `ActionBasedContinousMoveProvider`.

Lo más importante del `ContinousMoveProviderBase` es que deja disponible una interfaz para generar movimiento continuo basado solamente en transformaciones, es decir, que no utiliza físicas para el movimiento, como lo es el uso de gravedad, roce, fuerzas, etc. Se sirve de los métodos del `LocomotionProvider` para marcar el inicio y el fin de las locomociones e incluye también la opción de agregar gravedad, movimiento lateral, vuelo y especificar la velocidad de movimiento del usuario.

Finalmente, la clase `ActionBasedContinousMoveProvider` simplemente extiende a la clase anterior y recibe el input de movimiento basándose en las acciones de Unity que fueron explicadas en los antecedentes.

5.2.2. Caminata con gesto

La caminata con gesto consiste en mover hacia adelante al usuario cuando este hace el gesto de caminar balanceando los brazos. Ya que no se necesita el uso de físicas, se creó la clase `PropelLocomotionProvider` la cual extiende el `ContinousMoveProvider`. Como se puede ver en el diagrama de la figura 5.3, se utilizan las acciones de Unity para detectar la variación de posición de cada uno de los brazos del usuario en el eje vertical. De este modo, se fuerza a que el movimiento de los brazos del usuario sea similar a la mímica de caminar, pues no permite mover los brazos horizontalmente de forma arbitraria para que se detecte movimiento falso.

Para calcular la magnitud del movimiento en un fotograma, se utiliza la siguiente fórmula que da como resultado la velocidad total utilizando la variación de posición de ambos brazos:

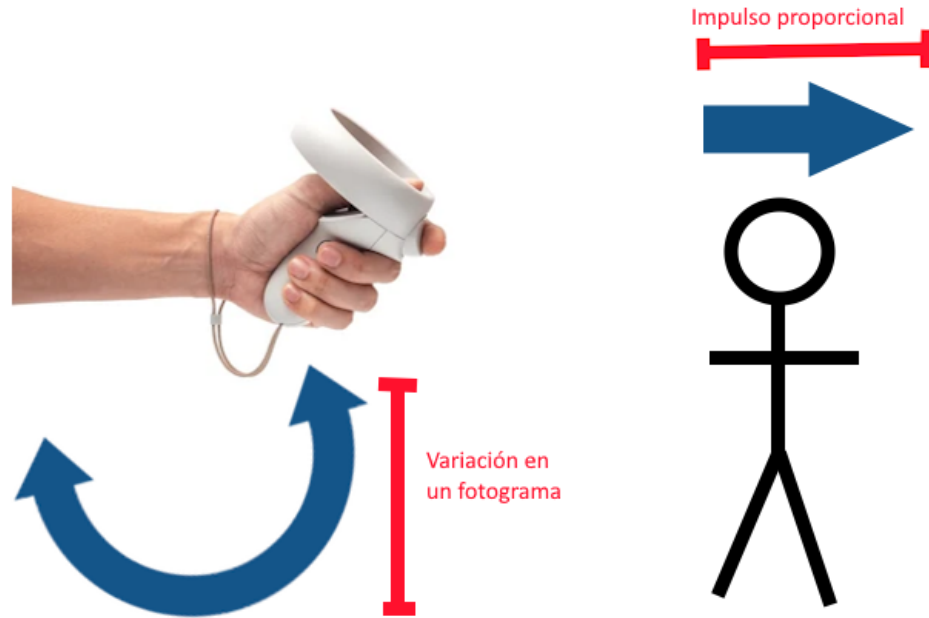


Figura 5.3: Diagrama simple de la caminata con gesto

$$V_f = V_{f-1} + (\Delta R_y + \Delta L_y)S\Delta t - E_l\Delta t^2$$

Donde V_f es la velocidad en el fotograma f , ΔR_y y ΔL_y la variación en el eje vertical de cada respectiva mano, S una fuerza configurable, Δt la variación de tiempo entre el fotograma anterior y el actual y, finalmente, E_l representa la pérdida de velocidad por fotograma, la cual es configurable y se podría considerar como el roce con el viento que genera que se pierda velocidad para que el movimiento eventualmente se detenga.

Luego de hacer el cálculo de la velocidad del fotograma, esta se restringe a una velocidad máxima para evitar que el movimiento pueda llegar a ser demasiado rápido en caso de que el usuario mueva los brazos demasiado rápido. También se impide que la velocidad pase a un valor negativo para que la pérdida de velocidad nunca logre hacer que el movimiento resulte en retroceso.

Para explicar el uso de Δt es útil descomponer la expresión de la siguiente manera.

$$V_f = V_{f-1} + ((\Delta R_y + \Delta L_y)S - E_l\Delta t)\Delta t$$

Se debe multiplicar la velocidad perdida por Δt pues en cada fotograma se demora un tiempo diferente en ser calculado, por lo tanto, si no se multiplicara es posible que en caso de que disminuyan los fotogramas por segundo (FPS), lo que debería ser un movimiento lento se transforme en uno muy rápido, pues existirían menos fotogramas en los que se aplique la reducción de velocidad. Esto no sucede con la variación de posición de las manos, pues por su naturaleza, ya depende del tiempo, ya que en un mismo periodo de tiempo en la vida real

se obtendrá la misma variación, independientemente de la cantidad de fotogramas en las que este movimiento sea revisado.

Además, luego se debe multiplicar toda la expresión por Δt , pues mientras más fotogramas, más veces se moverá al usuario. Entonces, por ejemplo, en caso de que no se haga la multiplicación, si se acelera de 0 a 10 unidades en un solo fotograma, la distancia recorrida sería igual a 10 unidades, pero luego, en caso de que esta misma aceleración se calculara en dos fotogramas, se podría tener $V_0 = 0, V_1 = 5, V_2 = 10$, sumando una distancia recorrida igual a 15 unidades en el mismo periodo de tiempo.

5.2.3. Propulsión

Esta locomoción mueve al usuario de forma similar a la caminata estándar, sin embargo, en este caso el movimiento cuenta con el uso de fuerzas y físicas en lugar de modificar la posición del usuario en cada fotograma. Se le permite al usuario moverse hacia adelante, hacia atrás y también de forma vertical, simulando el uso de un propulsor.

Debido a que el movimiento usa físicas, ya no es apropiado el uso de la clase nativa `ContinuousMoveProvider`. Es por eso que se creó la clase `PhysicsContinuousMoveProvider` que, confía en el uso de un `RigidBody`, un componente nativo de Unity que al añadirlo a un `GameObject` le permite actuar bajo el control de las físicas. Esta clase, de forma similar a su contraparte, hereda del `LocomotionProvider`, tal como se puede ver en la figura 5.4, y se encarga de manejar el inicio y el término de una locomoción.

5.2.4. Gancho

Esta TM también requiere del uso de físicas, al igual que la propulsión, por lo tanto, su provider también debe heredar de `PhysicsContinuousMoveProvider`. Se utiliza el componente de transformación de las manos del usuario para calcular en que dirección están apuntando, y luego, junto con la acción de jalar del gatillo, se impulsa al usuario en la dirección que se haya apuntado.

Para asegurar que al presionar el gatillo, el gancho efectivamente haya impactado en algún objeto, se emite un rayo en la dirección que el usuario esté apuntando y se revisa si el rayo impacta con el colisionador de algún `GameObject`. En caso de que efectivamente el rayo impacte, se revisa si el objeto en cuestión pertenece al grupo de objetos enganchables para evitar que, por ejemplo, el gancho pueda impactar con el suelo o el techo invisible del laberinto que evita que los usuarios escapen de este por arriba.

La velocidad a la que se mueve al usuario una vez tira un gancho depende de que tan lejos esté su objetivo, sumándole, además, un valor fijo en el eje vertical con el fin de generar una sensación de salto al impulsarse.

En pruebas iniciales, el uso de físicas normales hacía que el movimiento fuera incómodo para virar, pues el impulso se mantenía y probó ser difícil detenerse en el momento adecuado para cambiar de dirección. Para arreglar eso, finalmente se decidió que al impactar con el

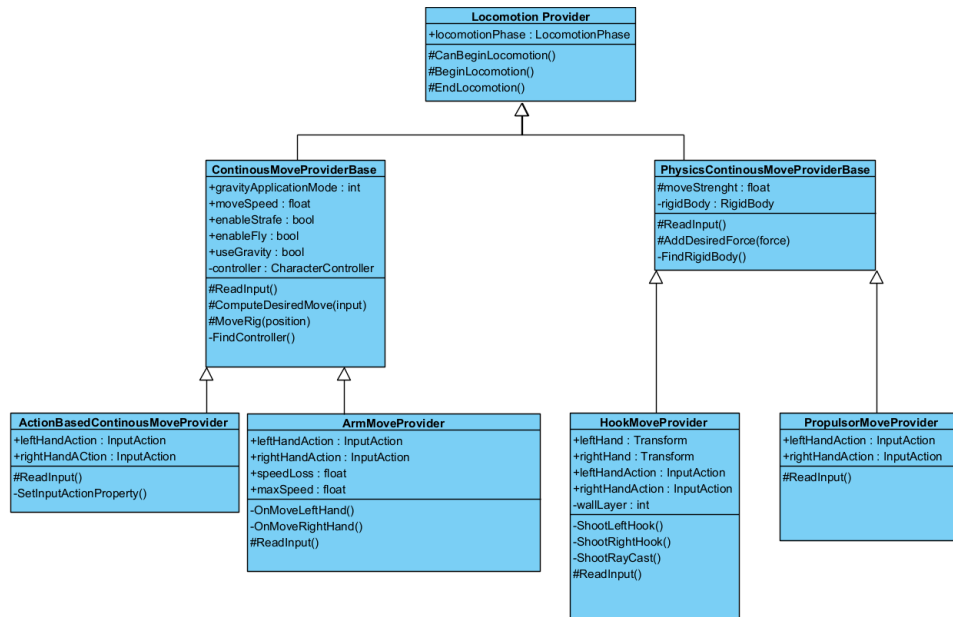


Figura 5.4: Diagrama de clases de las locomociones implementadas

gancho en un nuevo objetivo, la velocidad en el plano horizontal se anulara antes de agregar el nuevo impulso, de esta forma el cambio de dirección puede ser instantáneo.

Debido a que simplemente se aplica fuerza de forma puntual durante el fotograma en el que el gancho impacta, no es necesario considerar los casos en los que existan altas variaciones de fotogramas, esto también gracias a que luego de aplicar la fuerza, se relega la tarea de calcular la posición al motor de físicas de Unity, el cual se encarga de solucionar el problema de forma interna.

5.3. Escenas

5.3.1. Menú Principal

El menú principal muestra una interfaz simple en la cual el usuario no puede trasladarse de ninguna manera. En este menú, se le permite al usuario ver el indicador de mareo para aprender como funciona antes de comenzar el experimento. Además, como se ve en la figura 5.5, se despliega un cuadro con opciones para seleccionar una técnica de movimiento y un escenario.

En esta escena se busca tener la menor cantidad de distracciones disponibles, por lo tanto, el suelo muestra un patrón sencillo de forma infinita y el único objeto con el cual usuario tiene la posibilidad de interactuar, es el menú. Una vez se selecciona la técnica de movimiento, automáticamente se modifica el menú para preguntar por el escenario a utilizar y, en cuanto este es seleccionado, se carga la escena correspondiente al mismo.

A nivel de lógica, se utiliza una clase que sigue el patrón *Singleton* para asegurar su

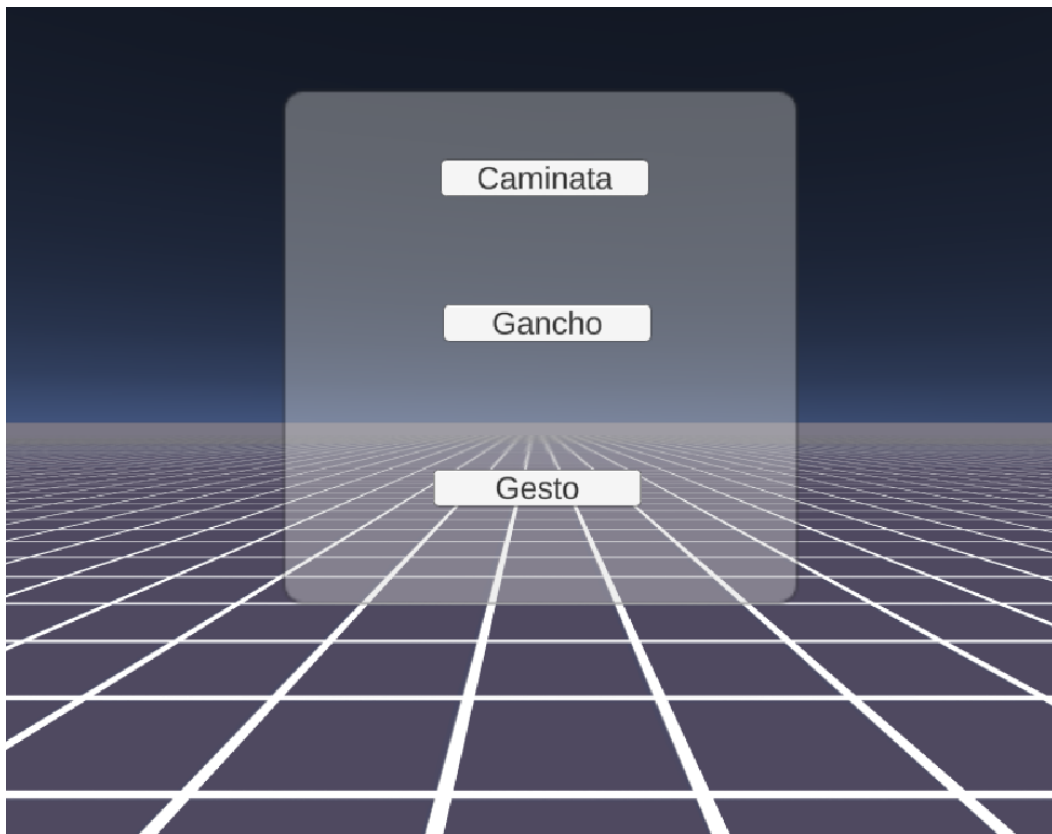


Figura 5.5: Menú principal para seleccionar técnica de movimiento y escenario

unicidad, la cual mantiene el estado del programa. Esta clase llamada `StateManager` almacena la decisión de cuál TM se desea utilizar para así cargarla una vez se cambia a la escena adecuada. Para lograr esto, se utiliza un evento de Unity que tienen disponibles los botones al ser presionados, asociando cada botón a un método del `StateManager` que permite definir la técnica de movimiento seleccionada.

5.3.2. Laberinto

La prueba del laberinto posiciona al usuario a la entrada de un laberinto, permitiéndole entrar por su propia cuenta para que se le haga más fácil recordar cuál es el inicio. En la figura 5.6 se puede ver la disposición de este. El laberinto fue hecho a partir de un modelo extraído de la página Sketchfab, junto con el modelo de los dados que se usa como murallas en el laberinto [10].

Para completar la prueba, se debe obtener la llave dorada que se encuentra en el centro flotando, la cual abre automáticamente la puerta dorada que se ve en la parte superior en cuanto se detecta una colisión entre el usuario y la llave.

Esta llave aprovecha el sistema de componentes de Unity para que se le pueda asignar cualquier `GameObject` como “cerradura”, enlazando la llave con ese objeto y provocando que se destruya al tomar la llave.

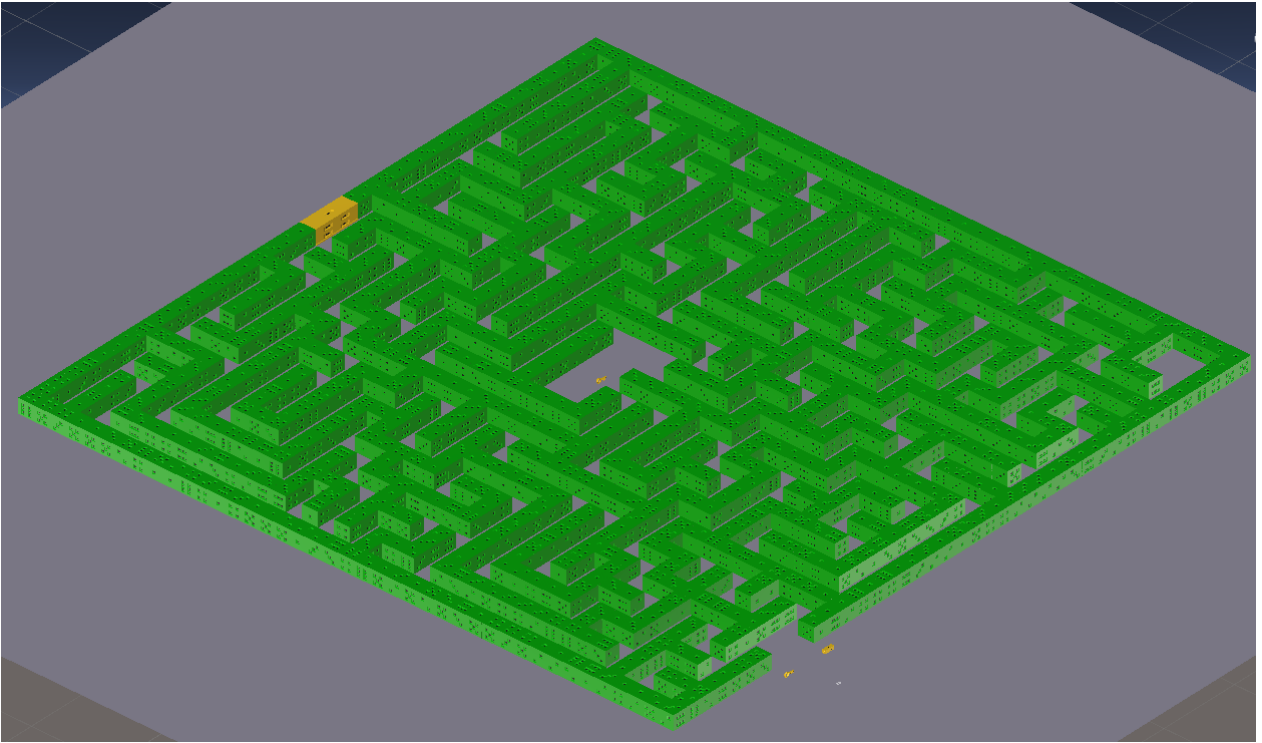


Figura 5.6: Prueba del laberinto en vista ortogonal

En el comienzo del laberinto (Fig. 5.6) hay una pequeña representación de tanto la puerta de salida como de la llave, de esta forma el usuario conoce como se ven de antemano y se evita la preocupación de creer que es posible pasar sin reconocerlas.

5.3.3. Pista de carreras

En esta prueba el usuario aparece al comienzo de una pista rodeada por muros en la que hay solo una dirección posible para avanzar (Fig. 5.7). El objetivo es alcanzar el final de la pista de carreras en el menor tiempo posible.

La pista de carreras consiste en tres secciones, dos de las cuales cuentan con obstáculos estáticos (primera y tercera sección), dejando una sección con obstáculos dinámicos (segunda sección) que empujan al usuario obligándolo a retroceder.

La primera y tercera sección son idénticas, pero reflejadas y están compuestas por dos sectores diferentes. Estos se pueden ver en la figura 5.7, el primer sector se utiliza para forzar un movimiento horizontal, mientras que con el segundo se busca lograr que el usuario se mueva rápidamente para esquivar de forma que no pueda avanzar describiendo una línea recta durante todo el trayecto.

La segunda sección es la que tiene los obstáculos dinámicos y, de estos, existen dos tipos diferentes:

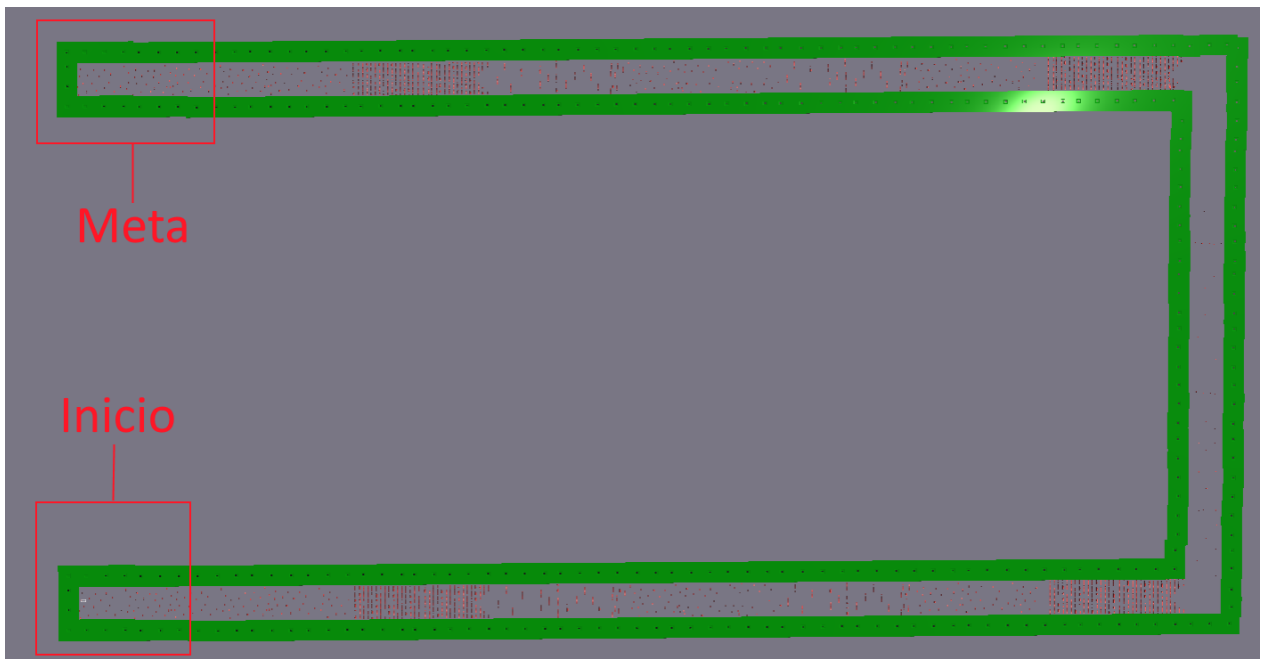


Figura 5.7: Prueba de la pista de carreras en vista aérea

Obstáculo oscilante

Esta clase de obstáculo se mueve siguiendo una trayectoria oscilante sobre su posición inicial. Para describir dicha trayectoria se utiliza la siguiente fórmula.

$$\vec{x}(t) = \vec{x}_i + \vec{r} \cdot \cos t$$

En donde \vec{x} es un vector con las coordenadas de la posición del obstáculo, \vec{x}_i la posición inicial, \vec{r} es un vector que marca la amplitud del movimiento en cada uno de los ejes y, finalmente, t representa el tiempo.

Gracias al vector \vec{r} , es posible configurar el componente de un obstáculo para que describa un movimiento ligeramente diferente. Por ejemplo, si solo se da amplitud en la componente x del vector, el obstáculo solo se moverá en dicho eje. De esta forma se logró obtener una multitud de obstáculos con trayectorias diferentes dentro de la pista.

Obstáculo de olas

Este obstáculo está ideado con la intención de que los usuarios puedan perder progreso si no avanzan con el cuidado de evitarlo. Su funcionamiento consiste en la interacción de dos objetos principales: la ola y el generador de olas.

La ola es el objeto físico que funciona como obstáculo, es decir, el objeto que cuenta con los componentes para generar colisiones, el material, y más importante, el código para definir su comportamiento.

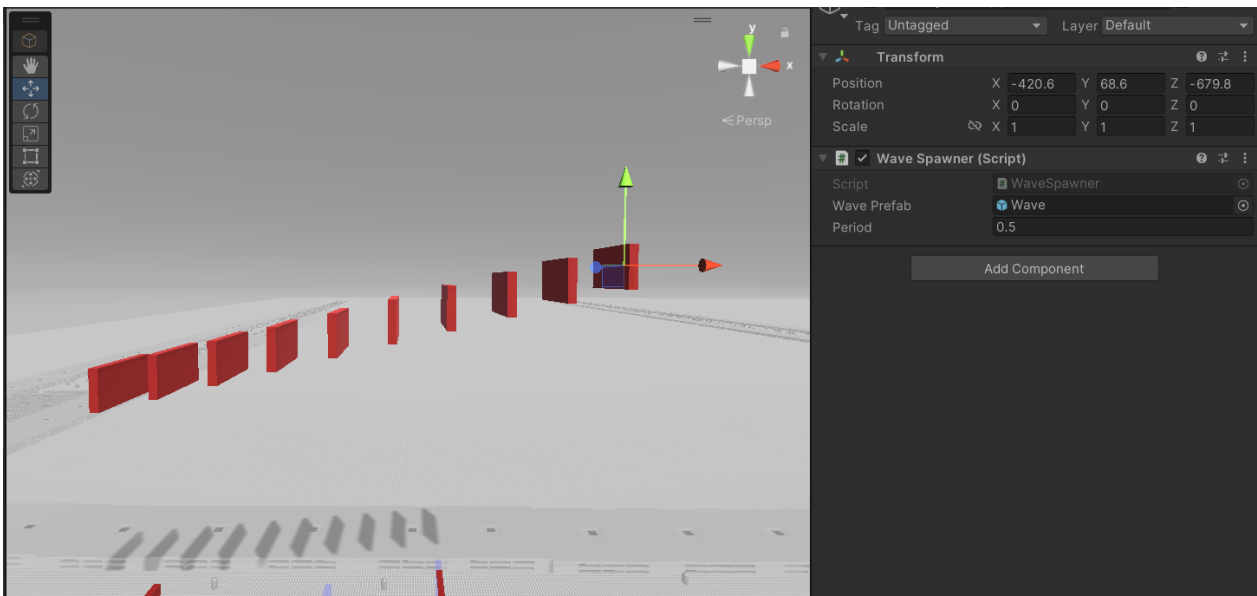


Figura 5.8: Generador de olas en escena

El comportamiento de la ola es simple, esta se mueve en cada fotograma hacia adelante y hacia abajo para empujar al jugador y hacer el efecto de una ola. Una vez la ola queda completamente cubierta por el suelo luego de bajar, esta se destruye para así mantener bien optimizada la aplicación.

El segundo objeto es el generador de olas, este es un objeto vacío que cuenta con dos únicos componentes, el de transformación y un script con su comportamiento.

De forma periódica, este objeto creará nuevas instancias del GameObject de las olas (Fig. 5.8), aunque técnicamente se le puede asignar cualquier objeto. Además, se puede configurar un periodo de tiempo distinto para cada uno de los generadores, de este modo, no se siente que estén sincronizados todos.

Finalmente, para añadir los obstáculos a la escena, se debe usar el editor de Unity para posicionar el objeto del generador de olas en los lugares deseados, como lo es en este caso en la sección inferior de la pista de carreras. Gracias a que el objeto del generador cuenta con el componente de transformación, las olas se generarán con su origen en dichas coordenadas.

5.4. Recolección de datos

5.4.1. Medidor de malestar

Se creó el medidor de malestar (Fig. 5.9) permite a los usuarios modificar en tiempo real su sensación de mareo al mover la palanca del controlador izquierdo. El medidor tiene capacidad para mostrar valores entre 0 y 100, sin embargo, una vez se llena la barra de aún se le permite al usuario aumentar más el valor de mareo en caso de que sienta que todavía aumenta el malestar. La parte visual del medidor se suscribe a un evento que se emite cuando

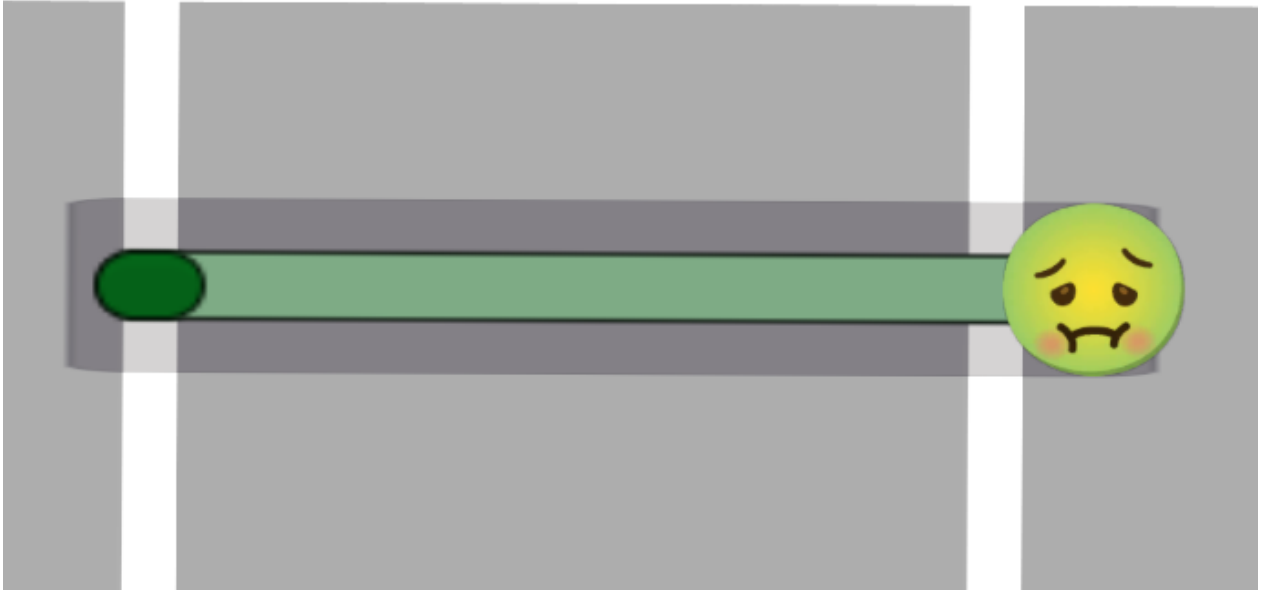


Figura 5.9: Medidor de malestar

el medidor sufre modificaciones por el usuario, de este modo se puede mantener el modelo y la vista separados.

Para guardar la información, se escribe en un archivo de texto cada vez que hay una variación en el valor del medidor junto a una marca temporal.

El medidor de malestar está fijado a la muñeca izquierda del usuario (Fig. 5.10). Esto significa que se ha asociado como hijo del controlador izquierdo en el árbol de escena. De esta manera, al mover el controlador, el medidor se mueve también. Se ha agregado un desplazamiento (offset) para que el medidor quede efectivamente en la muñeca y no directamente en la misma posición del controlador.

La alternativa a esto, la cual fue descartada, es situar el medidor de manera fija como si fuera una interfaz, por ejemplo, en una esquina (Fig. 5.10). Esto puede parecer más correcto en un inicio al verlo desde el editor, sin embargo, al observar a través de los lentes de VR, el medidor queda en la vista periférica y es imposible de enfocar incluso al mover la cabeza, pues está fijo a la cámara. También se descartó la opción de posicionar el medidor más al centro, pues reduce la visión del usuario.

5.4.2. Integración de Unity con cortex API

Emotiv cuenta con un ejemplo de como hacer una integración simple de su API en un proyecto de Unity, por lo que la mayoría de la solución que se describe a continuación está basada en dicha implementación.

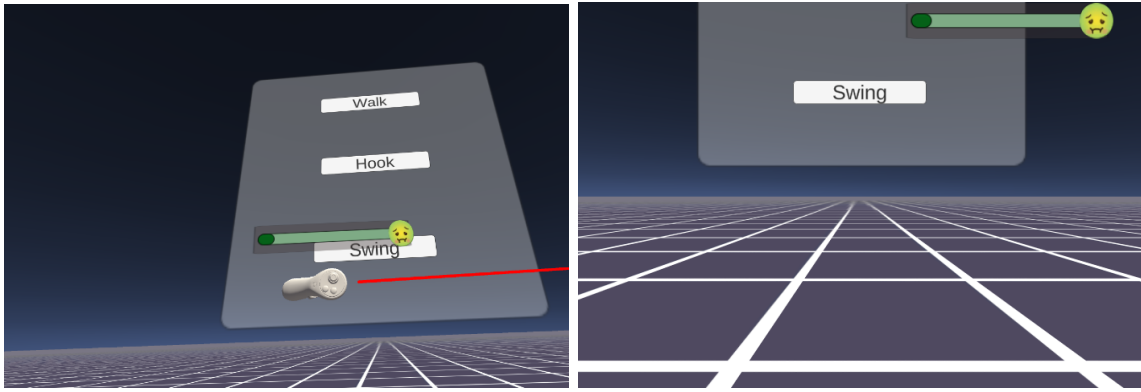


Figura 5.10: Comparación de posiciones para la barra de mareo. A la izquierda fija a la muñeca y a la derecha fija a la cámara

EmotivManager

La clase EmotivManager es una clase creada con el objetivo de mantener toda la información relevante para el uso de la cortex API a lo largo de toda la sesión.

En Unity, la información y el estado de las variables se pierden al cambiar de una escena a otra. Aunque un objeto, como el que representa al usuario, esté presente en dos escenas diferentes, estos serán instancias distintas de una misma clase y, por lo tanto, no compartirán ninguna información entre sí. Además, durante un cambio de escena, los objetos de la primera escena se destruyen, lo que los hace inalcanzables.

Para solucionar esto, se utilizó un método propio de la clase MonoBehaviour llamado DontDestroyOnLoad, que asegura que, al cambiar de escena, el objeto original de la primera escena se mantenga. Además, se implementó el patrón de diseño Singleton para garantizar la unicidad de cualquier instancia existente del EmotivManager.

Detección del BCI

Para sincronizar el BCI con la aplicación de Unity, es necesario en primer lugar encontrarlo, por eso una vez se inicia el simulador, se espera a recibir autorización de las credenciales correspondientes y, a continuación, comienza a escanear el área en busca de cualquier BCI de Emotiv que esté encendido y disponible. Una vez encontrado, se crea una sesión con el identificador correspondiente y queda preparado para funcionar una vez se confirma la creación de la sesión. Todo este proceso se puede ver mejor detallado en la figura 5.11 con un diagrama de estados.

Suscripción de información

Una vez sincronizado, es posible suscribirse a los múltiples tipos de información que son recolectados por el headset, como la de EEG, movimiento, performance metrics, band power, entre otros.

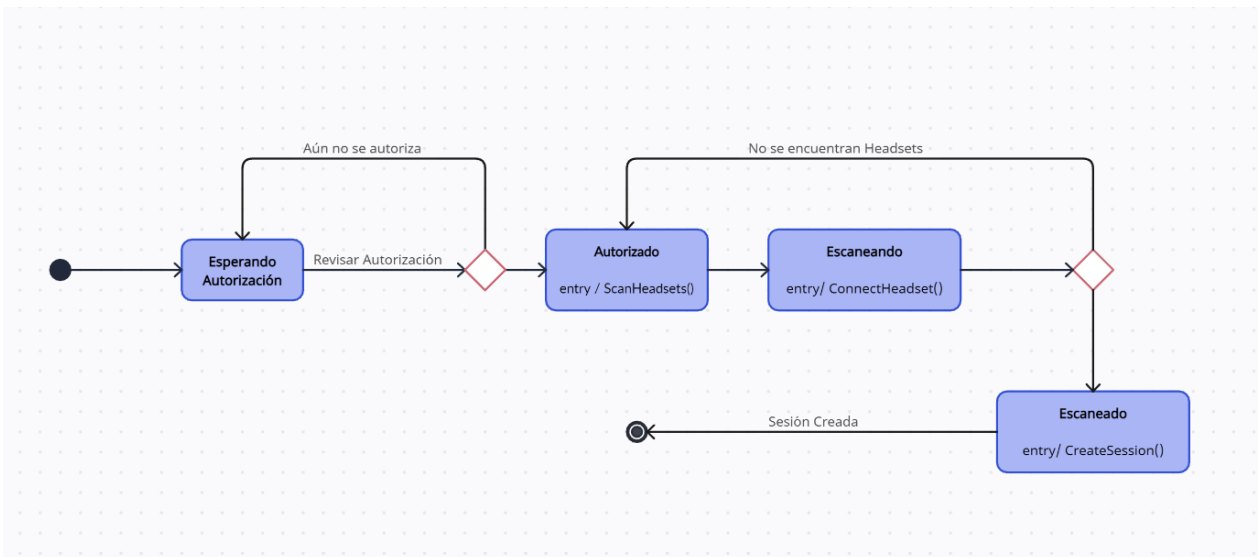


Figura 5.11: Diagrama de estados para crear conexión con la API

En este caso, solo hay interés en la medida de band power, pues esta es la información de la actividad de los diferentes tipos de ondas cerebrales en cada uno de los diodos.

El ejemplo de Emotiv incluye métodos que son invocados cuando se recibe información de cada uno de los eventos suscritos, por lo tanto, se aprovechan estos para almacenar la información en un archivo de texto junto a la marca temporal incluida por la misma API para su futuro estudio.

Capítulo 6

Experimento

6.1. Consideraciones

Debido a la naturaleza del experimento, el comité de ética de la Universidad de Chile decidió solicitar la presencia de personal médico durante la realización de los experimentos. Los licenciados en medicina Franco Balbiano y Gonzalo Mendoza dispusieron voluntariamente de su tiempo para hacer posible los experimentos (Fig. 6.1). Sin embargo, esto repercutió en gran medida a la dificultad de coordinar horarios al intentar reclutar voluntarios.

6.2. Procedimiento

6.2.1. Participantes

En el estudio participaron 18 jóvenes estudiantes. De ellos, cinco tienen entre 19 y 21 años, 11 tienen entre 22 y 24 años, y dos tienen entre 25 y 26 años. Quince de estos estudiantes pertenecen a la Facultad de Ciencias Físicas y Matemáticas (FCFM), y también participaron un estudiante de la Universidad Andrés Bello, uno de la Universidad de Santiago de Chile y uno de la Universidad de Harvard.

Respecto a la identificación de género, 14 se identificaron como masculino y 4 como femenino. En la figura 6.2 se pueden observar algunos de los participantes del estudio.

A cada uno de los participantes se les solicitó firmar un consentimiento informado en el que aceptan que su información biológica de ondas cerebrales anonimizada sea extraída para su futuro estudio y además confirman que están conscientes de los síntomas que podrán padecer y la posibilidad de que estos duren durante el resto del día. El documento de consentimiento se puede ver en el anexo A.

Además de esto, se compensó a los voluntarios con \$4.000 pesos chilenos por su participación en el experimento. Ofreciendo adicionalmente como incentivo para la velocidad una

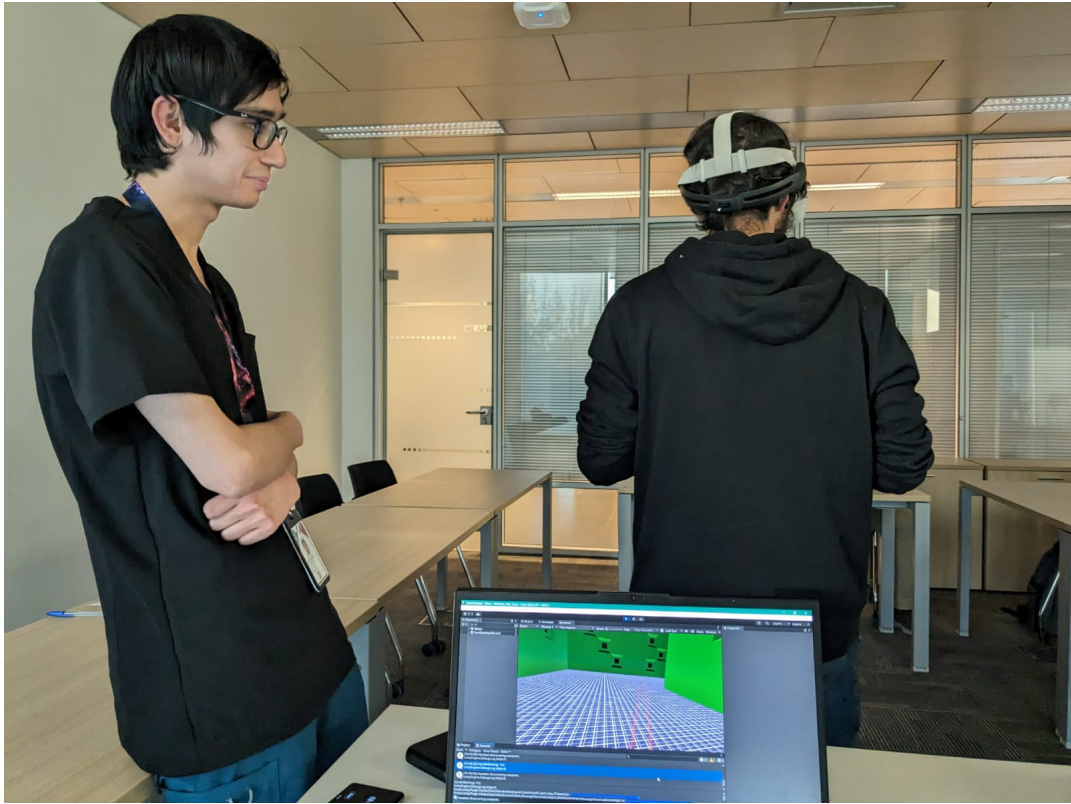


Figura 6.1: Personal médico supervisando el experimento

pizza familiar a la persona que completara su respectiva prueba más rápido que los demás.

6.2.2. Preparación

Previo a comenzar con el estudio es necesario colocar de forma adecuada tanto el Epoch+ como los lentes de VR. El primer paso es colocar cada uno de los diodos en el lugar adecuado, anterior a esto es necesario mojarlos con una solución salina hasta que queden húmedos, cerca de unas 6 gotas por cada uno es suficiente. A continuación se coloca el BCI en la cabeza del sujeto para finalmente mover los diodos hasta el sector del cráneo adecuado, evitando que queden posicionados por encima del pelo. Para lograr esto se utiliza la guía provista por la aplicación oficial *EMOTIV Launcher*. Tal como se muestra en la figura 6.3, la interfaz indica el nivel de contacto de cada uno de los diodos. Para poder avanzar a la siguiente fase se aseguró que el nivel de contacto esté siempre en 100% para garantizar una buena lectura.

A continuación se le acerca el Oculus Quest al sujeto para que se lo ponga de forma que le resulte cómoda por encima del BCI. Muy probablemente hacer esto resulte en la pérdida del contacto completo de los diodos, por lo que antes de avanzar se debe hacer una segunda revisión de este y arreglar cualquier problema que se haya podido generar. Algunas veces es necesario rehumedecer algunos diodos o volver a poner alguna almohadilla que se puede haber caído por el movimiento.

Finalmente, se le da indicaciones básicas al sujeto de cómo sostener los controladores y



Figura 6.2: Cuatro de los participantes del experimento

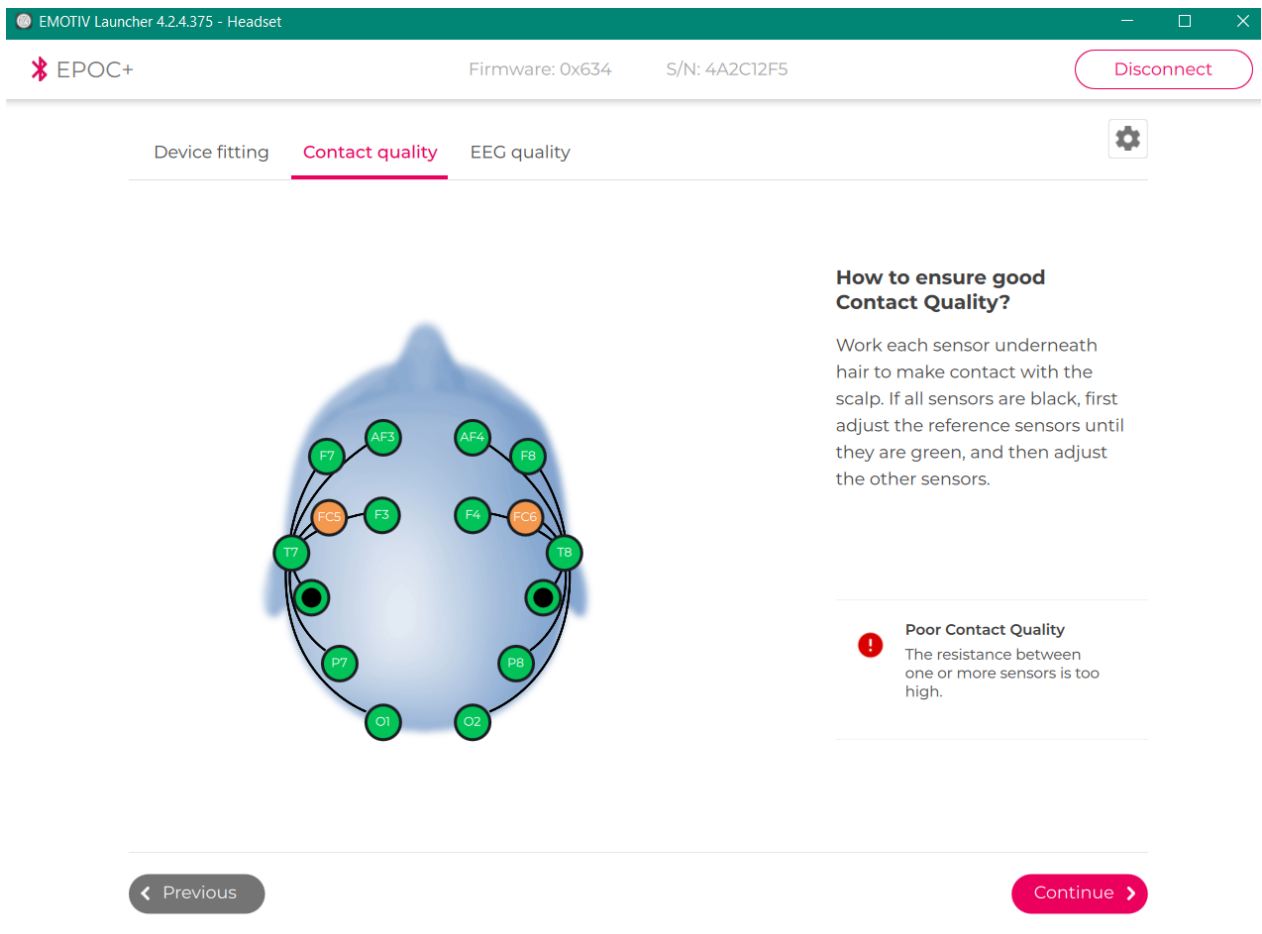


Figura 6.3: Interfaz de conexión del EMOTIV Launcher

cómo utilizar el visor de realidad virtual en general.

Esta etapa del experimento tuvo una duración promedio de 15 minutos, pudiendo variar debido a la dificultad de colocar los diodos de forma correcta, esto fue cierto en especial en las primeras iteraciones del experimento, pues no se contaba con experiencia poniendo el Epoc+, provocando que la etapa de preparación durara incluso hasta 30 minutos.

6.3. Actividad

En esta fase se introduce al sujeto al entorno de pruebas y se le da una sencilla explicación de como manejar la TM que se le ha sido asignada de entre las tres explicadas en los capítulos anteriores, junto a indicaciones de como manejar el indicador de malestar. También se le explica cuál será su objetivo en la prueba (Laberinto o Pista de carreras) y se le recuerda que, junto a tener un tiempo límite de 20 minutos, debe intentar resolver la prueba en el menor tiempo posible.

Una vez el sujeto selecciona en el menú principal la TM y el escenario que se le fueron asignados, es transportado automáticamente a la prueba y se da comienzo al temporizador de 20 minutos, además se le da un recordatorio al participante de que debe intentar moverse lo más rápido posible.

La prueba termina una vez el voluntario alcanza la meta respectiva o si el sujeto no es capaz de superar el escenario en 20 minutos. En ambos casos se cierra el entorno de pruebas y se le retiran tanto el visor de VR como el BCI.

6.4. Cuestionario

Una vez se terminó con la prueba, se solicita al participante completar un formulario lo antes posible. Este formulario incluye la versión en español del Cyber Sickness Questionnaire, obtenida mezclando la traducción de Campo-Prieto, P et al.[6] con el VRSQ de H.K. Kim et al. [21]. En este cuestionario se consulta si se sintió alguno de los síntomas relacionados con el cybersickness, los cuales se pueden ver en la tabla 6.1. El formulario completo se puede ver en el Anexo B.

El formulario también incluye preguntas de caracterización como la edad, género, hábitos de videojuegos, entre otros. En la parte final se le pregunta al sujeto su opinión sobre el entorno de pruebas para poder evaluar su calidad posteriormente.

En la parte final del cuestionario se hacen cuatro preguntas respecto a la calidad de la aplicación y del experimento. Se consulta sobre la dificultad para entender cómo moverse, la calidad de respuesta al usar la técnica de movimiento, la facilidad de usar la aplicación y la claridad del objetivo del sujeto en la prueba. Estas preguntas fueron hechas con el objetivo de validar que el entorno de pruebas ofrezca una experiencia agradable y que de ninguna forma pueda interferir en los resultados obtenidos, ya que, por ejemplo, una técnica de movimiento

Síntoma	Categoría
Malestar general	Oculomotor
Cansancio	Oculomotor
Vista Cansada	Oculomotor
Dificultad para enfocar	Oculomotor
Dolor de cabeza	Desorientación
Pesadez de cabeza	Desorientación
Visión borrosa	Desorientación
Mareo con ojos cerrados	Desorientación
Vértigo	Desorientación

Tabla 6.1: Tabla de síntomas consultados en el cuestionario y su categoría

que no responde correctamente al input del usuario podría empeorar el malestar.

6.5. Dificultades

A lo largo del experimento se presentaron múltiples dificultades y problemas, los cuales pueden ser útiles para tomar en cuenta en caso de que se desee hacer un experimento similar o con base en este.

6.5.1. Oculus Quest 2 y conexión a internet

A pesar de que, en general, funciona correctamente, existen ocasiones en las que se dificulta hacer la conexión entre el visor y el computador con el entorno. En la mayoría de los casos se encontró que estos errores estaban relacionados con que la forma de hacer el enlace fue a través del uso de wifi. Además de estos problemas, se encontró también que los momentos con caídas de FPS eran producidos por una débil conexión a internet en lugar de por las especificaciones del computador que ejecuta el entorno, como se tenía pensado.

Para solucionar estos problemas, existe la opción de tener a disposición múltiples redes de wifi o contar con un cable especializado para Oculus Quest, los cuales son más confiables.

6.5.2. Epoch+

Aunque el Epoch+ logró cumplir su propósito adecuadamente, es cierto que utilizando una versión diferente, el experimento pudo resultar mucho más sencillo de realizar.

En primer lugar, se recomienda escoger una versión más actual, pues con estas es más sencillo obtener una buena lectura de las ondas cerebrales y se asegura un resultado más confiable.

Una segunda recomendación es buscar modelos más flexibles, por ejemplo el FLEX 2 de



Figura 6.4: Emotiv FLEX 2

la figura 6.4. De esta forma, la posterior colocación del Quest no interferirá con la calidad de la lectura. Aunque se debe considerar que este tiene un costo considerablemente mayor que el BCI utilizado en este caso.

Capítulo 7

Resultados

En esta sección se pueden ver los resultados obtenidos a lo largo de todo el experimento, con todas las fuentes de información desarrolladas. Junto a cada resultado se incluye un comentario resaltado lo más interesante de este.

Es importante destacar que, a pesar de que originalmente hayan participado 18 sujetos, debido a fallos con las pruebas iniciales, se debieron descartar los primeros tres resultados, por lo tanto, se cuenta con 15 resultados válidos. De estos, hay 3 de cada TM en el laberinto y 2 de cada TM en la pista de carreras.

7.1. Formulario

En la tabla 7.1 se puede ver el porcentaje de veces que los sujetos identificaron cada uno de los síntomas según técnica de movimiento. Además, en la parte inferior de la tabla se puede ver el promedio de síntomas identificados en total. Aquí, se puede notar que la TM con mayor cantidad de síntomas identificados, en promedio, fue el gancho.

Síntoma	Gesto	Estándar	Gancho
Malestar general	33 %	66 %	50 %
Cansancio	16 %	50 %	16 %
Vista Cansada	33 %	16 %	50 %
Dificultad para enfocar	16 %	16 %	16 %
Dolor de cabeza	16 %	33 %	33 %
Pesadez de cabeza	0 %	33 %	16 %
Visión borrosa	33 %	16 %	16 %
Mareo con ojos cerrados	0 %	16 %	16 %
Vértigo	66 %	16 %	66 %
Promedio de síntomas	2.16	2.66	2.83

Tabla 7.1: Porcentaje de identificación de síntomas para cada técnica de movimiento

Síntoma	Laberinto	Pista
Malestar general	55 %	66 %
Cansancio	22 %	50 %
Vista Cansada	22 %	33 %
Dificultad para enfocar	22 %	0 %
Dolor de cabeza	33 %	33 %
Pesadez de cabeza	11 %	33 %
Visión borrosa	22 %	16 %
Mareo con ojos cerrados	0 %	33 %
Vértigo	44 %	66 %
Promedio de síntomas	2.33	3.33

Tabla 7.2: Porcentaje de identificación de síntomas por cada escenario

Síntoma	Laberinto			Pista		
	Gesto	Estándar	Gancho	Gesto	Estándar	Gancho
Malestar General	67 %	67 %	33 %	0 %	100 %	100 %
Cansancio	33 %	33 %	0 %	0 %	100 %	50 %
Vista Cansada	33 %	0 %	33 %	0 %	50 %	50 %
Dific. para enfocar	33 %	33 %	0 %	0 %	0 %	0 %
Dolor de cabeza	33 %	33 %	33 %	0 %	50 %	50 %
Pesadez de cabeza	0 %	33 %	0 %	0 %	50 %	50 %
Visión Borrosa	33 %	33 %	0 %	50 %	0 %	0 %
Mareo ojos cerrados	0 %	0 %	0 %	0 %	50 %	50 %
Vértigo	67 %	0 %	67 %	50 %	50 %	100 %
Promedio de síntomas	3	2.33	1.67	1	4.5	4.5

Tabla 7.3: Porcentaje de recurrencia de cada síntoma para cada TM y Escenario diferente

Luego en la tabla 7.2 se puede ver nuevamente el porcentaje de identificación de cada síntoma, pero en este caso separado por escenario. En esta se puede ver que hay un claro aumento de síntomas en el escenario de la pista.

Finalmente, en la tabla 7.3 se puede ver en mayor detalle los resultados obtenidos separados por TM y por escenario al mismo tiempo, en donde se puede observar que la caminata estándar y el gancho fueron los que presentaron mayor cantidad de síntomas en promedio al estar en el escenario de la pista, mientras que la caminata con gesto fue la con peor desempeño al estar en el laberinto.

Respecto a la calidad de la aplicación, en la figura 7.1 se pueden observar las opiniones de los usuarios de la misma. Como se puede ver, la respuesta en general fue positiva, siendo la característica mejor evaluada, la facilidad para comprender las técnicas de movimiento y la peor evaluada, la dificultad para usar la aplicación en general.

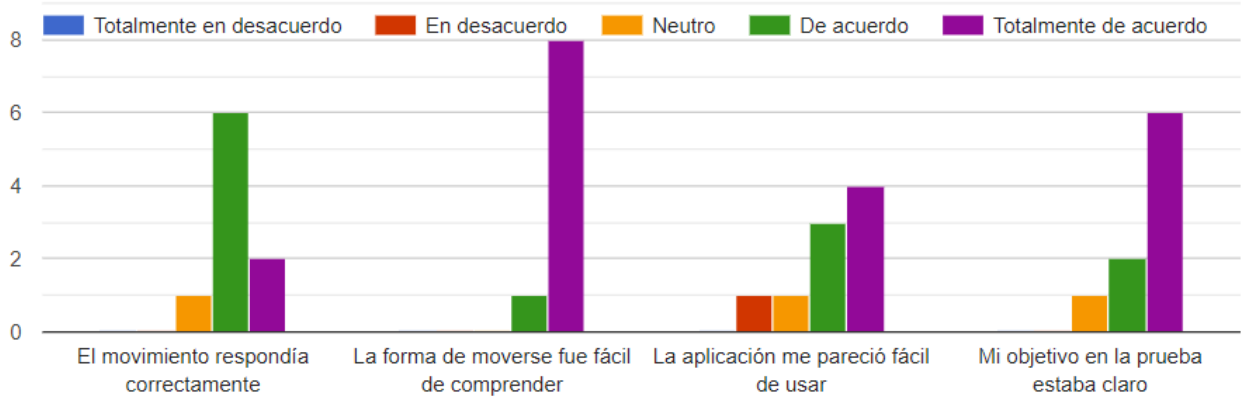


Figura 7.1: Gráfico de opiniones respecto a la calidad de la aplicación

Síntoma	Laberinto			Pista		
	Gesto	Estándar	Gancho	Gesto	Estándar	Gancho
Promedio	24.79	21.36	22.45	1.43	39.11	114.3
Desviación Estándar	30.2	21.36	10.52	15.14	16.83	63

Tabla 7.4: Promedio del medidor de mareo para grupo con una misma TM y escenario

7.2. Medidor de malestar

Respecto a la barra de malestar, se observó que esta era sencilla de olvidar utilizar por los participantes, desencadenando en que algunas de las mediciones queden estáticas durante toda la experiencia.

Esto se puede ver en la figura 7.2, donde se muestra un gráfico que compara el promedio de malestar obtenido a través del medidor con la cantidad de síntomas identificados por cada sujeto de pruebas. A pesar de que pareciera tender a que mientras más síntomas, se incrementó el medidor de mareo, existen casos con muchos síntomas que no tienen un valor mayor a 10.

En la Tabla 7.4 se muestran los promedios de malestar para cada TM en los diferentes escenarios y sus desviaciones estándar. Se puede ver la gran diferencia que tiene el uso del gancho en la pista con todo el resto de grupos. Esto coincide con lo observado por el investigador, pues los dos sujetos de prueba que participaron en este grupo fueron quienes más malestar expresaron, llegando al punto de que con uno se debió detener el experimento antes de tiempo. Es importante destacar que, a pesar de que el medidor de mareo soportaba valores entre 0 y 100 antes de llenarse, se le permitió a los sujetos aumentar más aún el valor si lo creían pertinente para que puedan expresar más fielmente como se sentían.

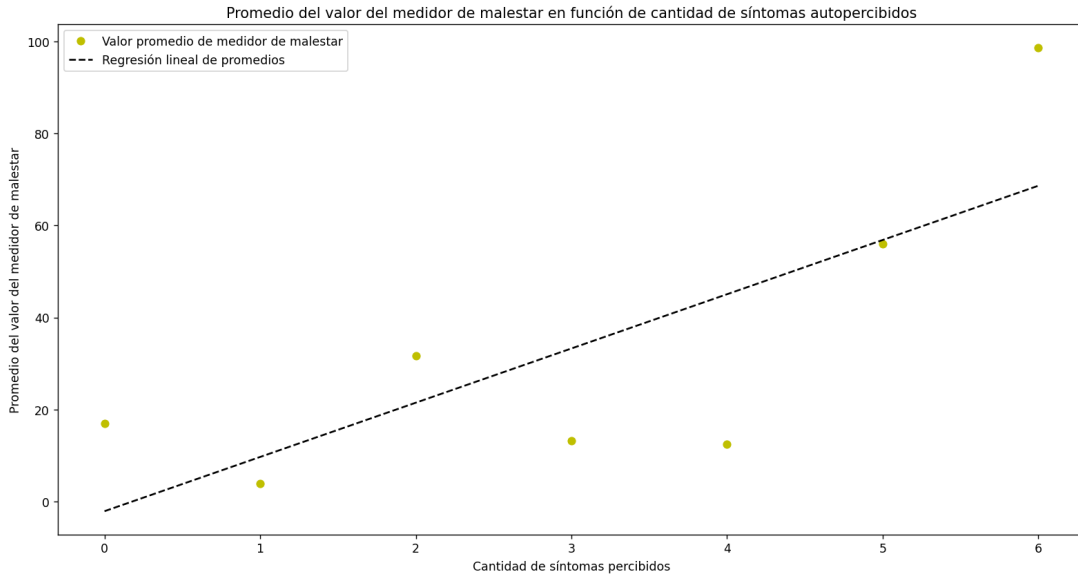


Figura 7.2: Gráfico de cantidad de síntomas en función de valores del medidor de malestar por sujeto

7.3. Ondas cerebrales

Para observar estos resultados se hará un promedio de las ondas alpha, y theta de los 16 diodos del emotiv para compararlas entre diferentes grupos, pues esta es la onda cerebral más relacionada con el malestar del cybersickness, como se explicó anteriormente.

En la figura 7.3 se puede ver una comparación entre el promedio de las ondas theta y alpha de dos sujetos. El sujeto 1 no reportó sentir ningún síntoma de los descritos en la encuesta, a diferencia del sujeto 2 que reportó 6. Sin embargo, en la figura se puede notar como la actividad cerebral en el primer sujeto es considerablemente mayor, llegando al punto de opacar al sujeto 2.

Este fenómeno está relacionado con lo mencionado en el experimento de Marcel, S y R. Millan, J [26]. En donde se postula la diferencia en los patrones de EEG en diferentes personas, cosa que intentan utilizar para hacer un sistema de reconocimiento de identidad.

Tomando en cuenta eso, también se obtiene la figura 7.4 en donde se comparan dos gráficos de pila en los que se muestra la proporción de cada una de las bandas de poder para ambos sujetos. En esta se puede ver que el segundo sujeto muestra una actividad mayoritaria en las bandas de frecuencias bajas, mientras que el primero tiene actividad más distribuida.

En la figura 7.5 se puede ver un gráfico de líneas mostrando el promedio de la distribución de las bandas de poder en función de cantidad de síntomas percibidos por los sujetos de prueba. Sin embargo, el efecto contrario al esperado, pues en este, la distribución de las ondas theta disminuye mientras aumenta para las ondas gamma a medida que los sujetos manifestaron mayor malestar, exceptuando a los sujetos con 6 síntomas. Debido a esto se

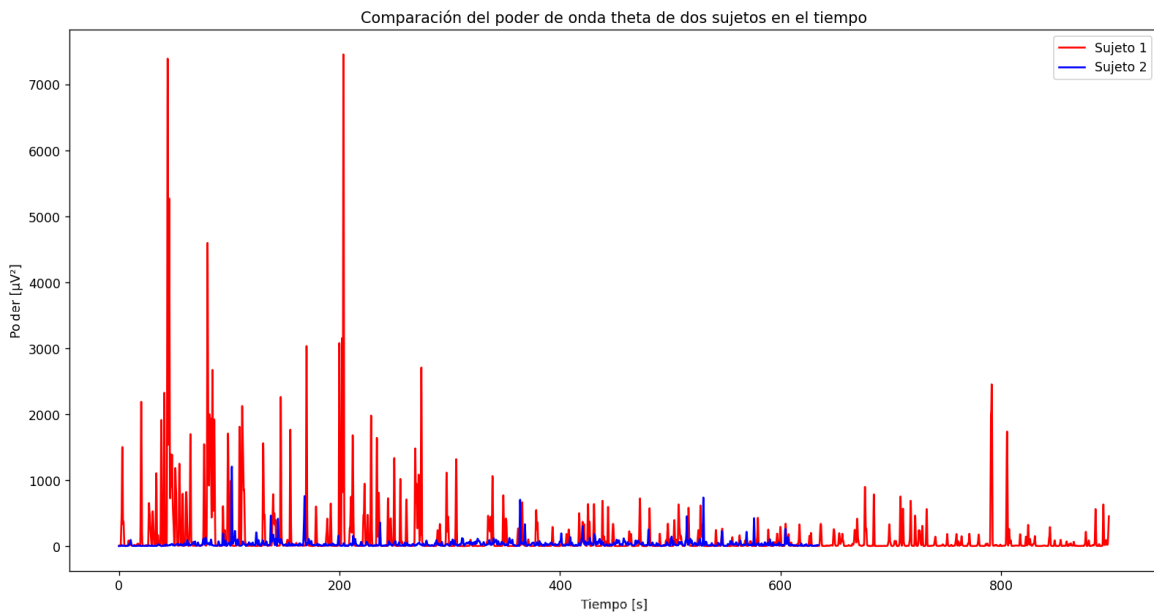
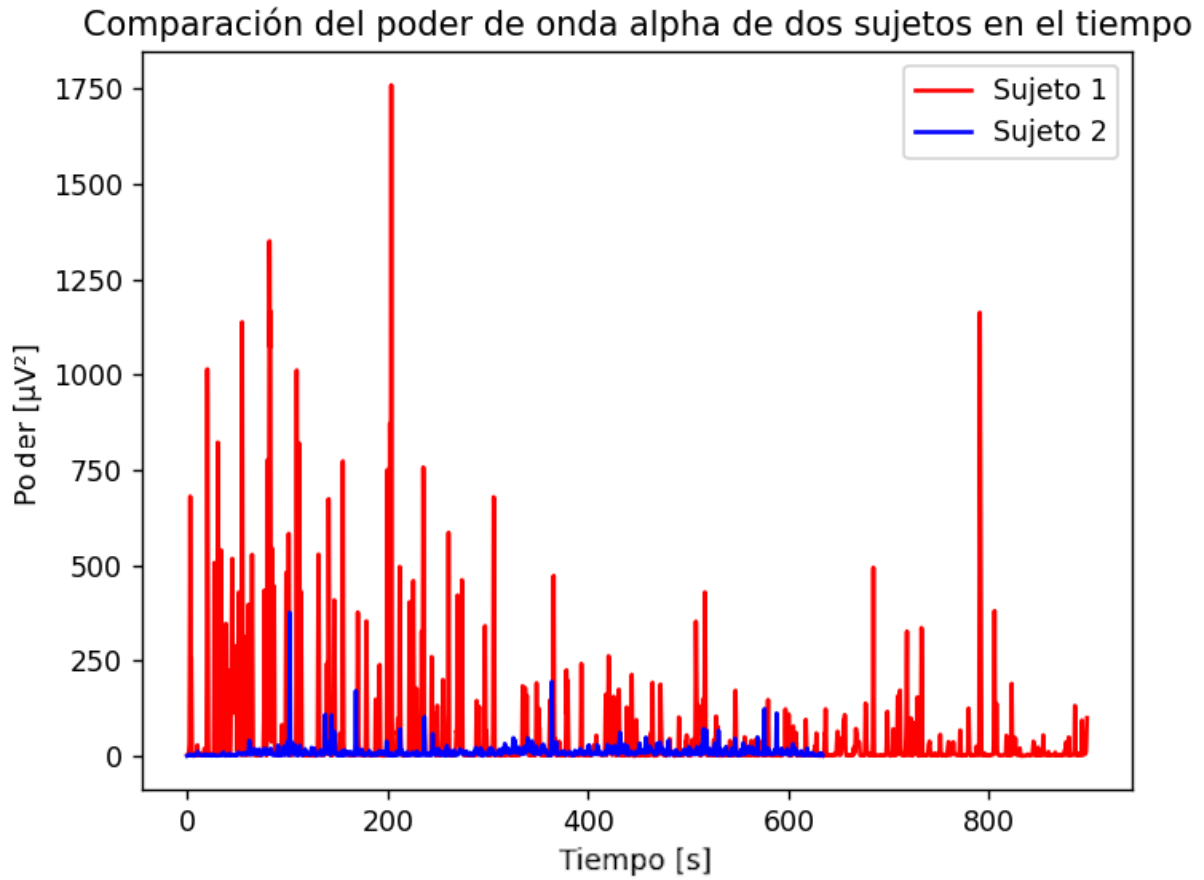


Figura 7.3: Comparación de poder en ondas theta y alpha de dos sujetos de prueba

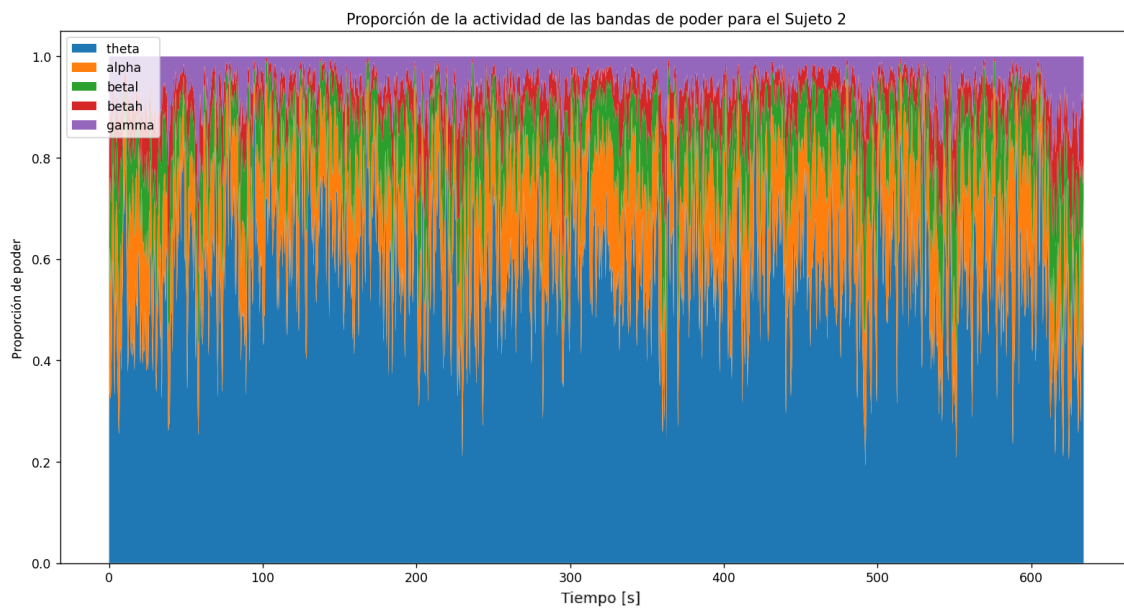
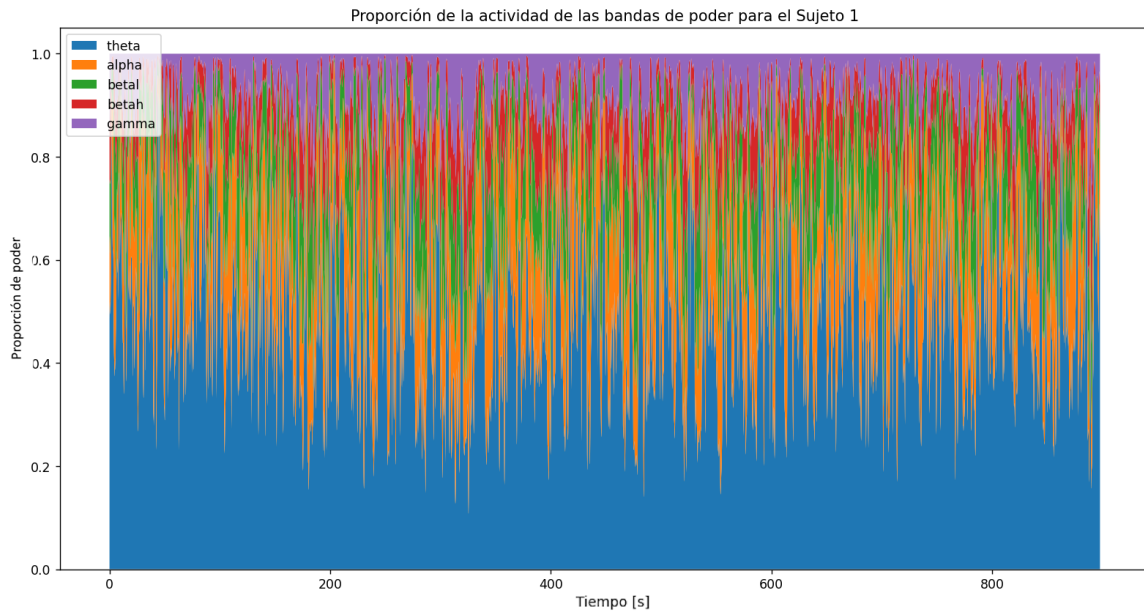


Figura 7.4: Gráficos de pila con proporción de bandas de poder del sujeto 1 (superior) y sujeto 2 (inferior)

Distribución de las bandas de poder según cantidad de síntomas percibidos

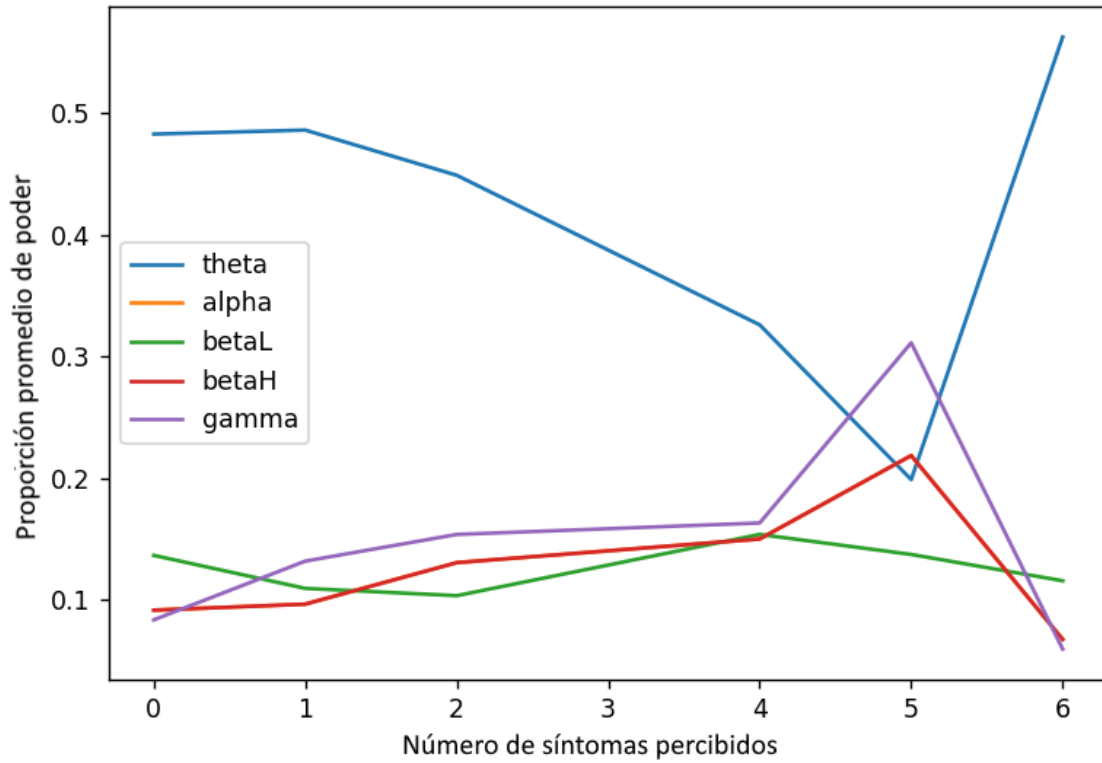


Figura 7.5: Gráfico de líneas comparando la distribución promedio de bandas de poder para diferente cantidad de síntomas percibidos

desconfía de la consistencia de los resultados del EEG y se decide no utilizarlos para el análisis de las TM.

Capítulo 8

Análisis y Reflexión

8.1. Técnicas de movimiento

Basado en lo expuesto en la sección anterior, se logró identificar ciertas características respecto a las TM implementadas y a los síntomas que estas provocan. Encontrando situaciones en las que unas parecen favorecer al usuario y otras en las que lo perjudican.

8.1.1. Caminata con gesto

En promedio, la TM que menos cantidad de síntomas provocó fue la caminata con gesto. Se cree que esto es debido a que genera una mayor conexión cuerpo-movimiento, evitando la disonancia que genera malestar habitualmente. Además, basado en los comentarios de sujetos de prueba, moverse de forma lateral sin mirar es de las acciones que mayor malestar provocó, cosa imposible de realizar con esta TM,

En específico, la caminata con gesto tuvo un muy buen desempeño en la pista de carreras y fue la TM que tuvo el peor en el laberinto. Esta TM es la que presenta la mayor dificultad para virar, debido a que simplemente permite ir hacia el frente, provocando que el movimiento en el laberinto se viera interrumpido con mayor frecuencia y obligando al sujeto de pruebas a mover su campo de visión constantemente para avanzar. Por otro lado, en la pista de carreras no había tanta necesidad de virar, y la mayoría de los obstáculos se podían esquivar con pequeños ajustes a la trayectoria.

8.1.2. Caminata estándar

La caminata estándar es la TM más consistente de todas, debido a que no produce movimiento demasiado repentino y es fácilmente controlable. No fue la técnica con peor desempeño en ninguno de los dos escenarios, sin embargo, la pista de carreras fue el escenario en el que generó más malestar.

Parece ser una buena opción en general para avanzar por cualquier clase de escenario, aunque en caso de necesitar deslizarse lateralmente puede aumentar el malestar generado.

8.1.3. Gancho

La TM basada en gancho fue la que obtuvo en promedio el peor de los desempeños entre todas. Esta TM es la única que genera que el usuario se despegue del suelo y utiliza un movimiento basado en físicas, por lo que genera un movimiento más impredecible que las demás.

En el laberinto generó poco malestar, debido a que en este hay mayor facilidad para impactar el gancho y, además, se tiene mucho espacio para moverse libremente. Sin embargo, en la pista, fue la TM con peor desempeño y además fue la combinación que resultó en el mayor malestar de un sujeto particular. Esto debido a que la pista de carreras, el movimiento, se cancela constantemente con los obstáculos y también, como reportaron múltiples sujetos, por el movimiento lateral.

8.2. Medición

Algunas de las formas de medir el malestar de los sujetos de prueba presentaron múltiples problemas, los cuales serán discutidos a continuación.

8.2.1. Medidor de malestar

El medidor de malestar demostró ser preciso y confiable en la mayoría de los casos, sin embargo, durante la realización del experimento se notó que algunos de los sujetos se encontraban confundidos por el sistema, mientras que otros olvidaban con frecuencia actualizarlo.

En primero lugar, las personas que se confundieron fue debido a que este presenta un emoji de mareo al final del indicador, cuya intención era indicar que mover la barra hacia esa dirección significa un mayor malestar, sin embargo, algunos de los sujetos interpretaron el emoji como el indicador de malestar en sí. Generando confusión, pues estos presuponían que el medidor no se estaba moviendo correctamente y que comenzaba lleno.

Lo segundo, y más importante, es el hecho de que los usuarios se olvidaran de mover el medidor luego de unos cuantos minutos de estar introducidos en el entorno de pruebas. Se cree que esto se puede haber producido por dos factores: La visibilidad y la naturaleza de las pruebas.

Ya que el medidor fue situado en el brazo izquierdo del usuario, este puede desaparecer de la vista fácilmente si se sitúa el brazo fuera de la cámara, situación que ocurrió más de lo esperado. Además, la prueba exige concentración al caminar para no chocar con obstáculos

o para recordar el camino correcto, generando que sea más sencillo de olvidar la existencia del medidor.

Para explotar todo el potencial de esta forma de medición, se debe diseñar una prueba en la que la única acción disponible para el usuario sea manipular el medidor y que este siempre esté visible. También puede ser una buena solución que se hagan recordatorios periódicos.

8.2.2. EEG

La medición de ondas cerebrales finalmente debió ser descartada como fuente de información debido a las imprecisiones encontradas. Esto pudo ocurrir debido a múltiples factores, como lo son errores al momento de colocar el BCI en los sujetos, fallos del mismo BCI debido a su uso, problemas con diodos oxidados, des posicionamiento de los diodos producto del movimiento de los sujetos y diferencias en los patrones de EEG entre diferentes sujetos.

Para poder utilizar exitosamente esta forma de medir se debe hacer en un experimento con menos variables, es decir, uno en el que los sujetos no se deban estar moviendo constantemente e, idealmente, utilizando un equipo BCI más compatible con unos lentes de realidad virtual.

Además, es importante recalcar que el uso de tecnología como esta, para medir el mareo, es mucho más efectiva para realizar análisis en una misma persona. Ya que los patrones de ondas cerebrales de cada persona son diferentes, se complica la comparación entre diferentes sujetos de prueba. Para usar correctamente un BCI se debería poner a un mismo sujeto en múltiples situaciones y registrar la variación de su reacción a las mismas, sin embargo, para poder ejecutar un experimento como este se requiere de una gran cantidad de tiempo, ya que lo ideal sería que un sujeto espere un día completo entre pruebas.

8.3. Experimento

Los resultados obtenidos por este experimento pueden ayudar a vislumbrar las diferencias que pueden producir en el usuario la elección de que forma de moverse, sin embargo, no pueden ser tomados como una regla debido a la baja cantidad de muestra en la cual se basan los resultados.

A pesar de que las TM realmente parecen tener diferentes características que las hacen mejores en ciertos escenarios que en otros, es muy probable que la resistencia al mareo específica de cada sujeto pueda afectar a los resultados obtenidos, por lo tanto, sería adecuado aumentar la muestra para evitar que factores como estos afecten al resultado.

Capítulo 9

Trabajo futuro

Del trabajo realizado en esta memoria se obtienen nociones que pueden resultar útiles para el desarrollo de más investigaciones y trabajos a futuro. A lo largo de este capítulo se discuten rutas que se podrían tomar para profundizar todavía más en lo encontrado en esta memoria.

9.1. Continuar con el mismo experimento

En primer lugar, está la opción de continuar con el experimento actual intentando aumentar la muestra de sujetos de prueba. De esta forma, los resultados obtenidos tendrán mayor confiabilidad, pues la muestra con la que se cuenta en esta memoria es demasiado baja. Además, es importante corregir los errores descritos anteriormente y mejorar el diseño experimental junto con pulir más la aplicación desarrollada.

En caso de mantener un diseño experimental similar, sería recomendable desistir del uso de un BCI, pues este demostró no ser un buen indicador de mareo al comparar entre diferentes sujetos de prueba.

9.2. Extender la cantidad de pruebas

Usando la base de aplicación desarrollada, es posible programar nuevas técnicas de movimiento y escenarios con facilidad, por lo que también se puede seguir avanzando el proyecto al poner a prueba un mayor número de TM y escenarios, encontrando nuevas características para cada una de las TM. De este modo, se podría comenzar a esclarecer los motivos por los cuales algunas TM tienen un mejor desempeño en algunos escenarios y peor desempeño en otros, pues se podría generalizar los tipos de escenario en los que generan menor CS.

9.3. Profundizar en causas de CS generales

Se encontraron posibles factores generales que parecieran generar CS, como por ejemplo, moverse en una dirección que no se está mirando o colisionar con obstáculos del entorno. Sin embargo, para poder analizar de mejor manera estos fenómenos, es necesario diseñar experimentos específicos que expongan sujetos de prueba a estos. De esta forma se podría asegurar o desmentir la incidencia de estos factores en la generación de CS.

9.4. Detección de CS utilizando BCI

La detección de CS utilizando un BCI demostró ser un terreno todavía poco explorado, por lo tanto, es necesario realizar experimentos que estén completamente enfocados a esto, a diferencia del trabajo realizado.

Se puede profundizar específicamente en como cuantificar el CS con un BCI, permitiendo de esta forma la comparación entre diferentes sujetos con un valor completamente objetivo.

Capítulo 10

Conclusión

En esta memoria, el objetivo principal fue encontrar una caracterización de diferentes formas de simular movimiento en un entorno de realidad virtual y evaluar la intensidad de los síntomas de CS que generan, para encontrar las ventajas y desventajas de cada una de ellas. Para lograrlo se implementó con éxito un entorno de realidad virtual en el cual realizar las pruebas, utilizando el motor de desarrollo de videojuegos Unity. El entorno resultó ser cómodo para obtener los resultados de forma ordenada y también fue validado por los sujetos de prueba. A continuación se planificó un diseño experimental que fue suficiente para obtener información respecto al malestar de los usuarios y la actividad cerebral de los mismos, utilizando tres formas diferentes para la recolección de datos.

Cumplir los objetivos específicos, probó ser útil para avanzar en la dirección correcta para alcanzar el objetivo principal de la memoria. Se investigó del estado del arte respecto a técnicas de movimiento en el medio, seleccionando técnicas de movimiento que parecieran ser útiles en la vida real, además de suficientemente diferentes entre sí. Junto con esto, también se encontraron buenas formas de medir el cybersickness como una encuesta especializada, un medidor de malestar y el uso de EEG, aunque este último no dio los resultados esperados debido a la limitación de tiempo y personas.

Ya que se logró obtener una caracterización de tres técnicas de movimiento, encontrando escenarios en las que son mejores y peores, junto a una intuición del motivo por el cual este puede ser el caso, se puede dar por cumplido el objetivo de este trabajo de título, aunque claramente queda mucho espacio para mejorar estos resultados y hacerlos más concluyentes.

La clasificación obtenida revela que la mejor técnica de movimiento fue la de caminata con gesto, seguida por la caminata estándar, dejando en último lugar al gancho, sin embargo, se identificó situaciones en las que estas pueden ser mejor que las demás, independientemente de su clasificación. Destacando de cada una el movimiento frontal, la consistencia, y el movimiento en espacios amplios, respectivamente.

Respecto a factores de mareo, por sobre todo se identificó que moverse en una dirección sin estar mirando al mismo tiempo genera mucho malestar. Esto se mantuvo consistente en diferentes escenarios, siendo comentado por los mismos sujetos de prueba.

Además de la clasificación, este trabajo de título queda como una buena base para la realización de múltiples experimentos en el futuro, exponiendo las dificultades que se pueden encontrar al trabajar con las diferentes formas de medición, al planificar una metodología, pero también proponiendo formas de solucionar estas dificultades para asegurar un resultado mucho más robusto.

El estudio de ondas cerebrales no parecer ser muy confiable para comparar el mareo entre diferentes personas, no solo por la gran cantidad de factores y estímulos que pueden desencadenar en la variación de potencia en las ondas cerebrales, sino que también porque cada persona tiene un patrón diferente. En caso de que aun así se quiera utilizar tecnología similar, es recomendable usarlo para comprobar el comportamiento de las ondas de una misma persona bajo distintas situaciones controladas en una misma instancia. También puede ser favorable recolectar los datos cerebrales de los sujetos en reposo previamente y posterior al experimento.

Para futuros experimentos también puede ser recomendable reducir la cantidad de técnicas de movimiento a comparar, de esta forma es más sencillo alcanzar una muestra representativa y, además, de esta forma se pueden analizar en mayor detalle.

Finalmente, se dejan propuestas múltiples rutas para avanzar con la investigación del CS, tanto usando un BCI como sin uno.

Todavía quedan muchos factores y variables a explorar en el área de la realidad virtual, por ejemplo, se podría hacer un experimento completo tan solo poniendo a prueba una misma técnica de movimiento en diferentes escenarios. Pero también por eso, el espacio a mejorar la experiencia de los usuarios al utilizar realidad virtual también es grande.

Bibliografía

- [1] Maryam Alimardani and Kazuo Hiraki. Passive brain-computer interfaces for enhanced human-robot interaction, Aug 2020.
- [2] Evan A. Suma Sabarish Babu and Larry F. Hodges. Comparison of travel techniques in a complex, multi-level 3d environment. *3D User Interfaces*, 03 2007.
- [3] Patricia Bockelman and Deanna Lingum. Factors of cybersickness. *Communications in Computer and Information Science*, page 3–8, 2017.
- [4] David Bond and Madelein Nyblom. Evaluation of four different virtual locomotion techniques in an interactive environment, 2019.
- [5] D. A. Buker, T. J. Vincenzi and J. E. Deaton. The effect of apparent latency on simulator sickness while using a see-through helmetmounted display: Reducing apparent latency with predictive compensation. *Human Factors: The Journal of the Human Factors and Ergonomics*, 2012.
- [6] P. Campo-Prieto, G. Rodríguez-Fuentes, and J Cancela Carral. Traducción y adaptación transcultural al español del simulator sickness questionnaire (translation and cross-cultural adaptation to spanish of the simulator sickness questionnaire): Ssq versión española. *Retos*, 2022.
- [7] Heni Cherni, Natacha Métayer, and Nicolas Souliman. Literature review of locomotion techniques in virtual reality. *International Journal of Virtual Reality*, 20(1):1–20, 2020.
- [8] Noah Coomer, Sadler Bullard, William Clinton, and Betsy Williams-Sanders. Evaluating the effects of four vr locomotion methods: Joystick, arm-cycling, point-tugging, and teleporting. In *Proceedings of the 15th ACM Symposium on Applied Perception, SAP '18*, New York, NY, USA, 2018. Association for Computing Machinery.
- [9] Documentación de Unity sobre GameObjects. <https://docs.unity3d.com/Manual/GameObjects.html>.
- [10] Modelo del laberinto y dados. <https://sketchfab.com/models/704f4acfa7504bb3ae5c51ee75ae5a8b/embed>.
- [11] Jun 2018. DeuSens. DISPOSITIVOS VR — ¿Qué gafas de Realidad Virtual comprar en 2022? <https://deusens.com/es/blog/dispositivos-vr-gafas-realidad-virtual>.

- [12] Regan E. and Price K. The frequency and occurrence and severity of side-effects of immersion in virtual reality. *Aviation Space and Environmental Medicine*, 1994.
- [13] Emotiv. <https://github.com/Emotiv/cortex-example/tree/master>.
- [14] Jannat Falah, Soheeb Khan, Tasneem Alfalah, Salsabeel F. Alfalah, Warren Chan, David K. Harrison, and Vassilis Charissis. Virtual reality medical training system for anatomy education. *2014 Science and Information Conference*, 2014.
- [15] Mana Farshid, Jeannette Paschen, Theresa Eriksson, and Jan Kietzmann. Go boldly! *Business Horizons*, 61(5):657–663, 2018.
- [16] Ajoy S Fernandes and Steven K. Feiner. Combating vr sickness through subtle dynamic field-of-view modification. *2016 IEEE Symposium on 3D User Interfaces (3DUI)*, 2016.
- [17] Rus-Calafell Mar Gutiérrez-Maldonado José, Magallón-Neri Ernesto and Peñaloza-Salazar Claudia. Virtual reality exposure therapy for school phobia. *Anuario de Psicología*, 2009.
- [18] Sandra Helsel. Virtual reality and education. *Educational Technology*, 1992.
- [19] Sungchul Jung, Richard Li, Ryan McKee, Mary C. Whitton, and Robert W. Lindeman. Floor-vibration vr: Mitigating cybersickness using whole-body tactile stimuli in highly realistic vehicle driving experiences. *IEEE Transactions on Visualization and Computer Graphics*, 27(5):2669–2680, 2021.
- [20] Dorota Kamińska, Krzysztof Smółka, and Grzegorz Zwoliński. Detection of mental stress through eeg signal in virtual reality environment. *EBSCO*, 2021.
- [21] Hyun K. Kim, Jaehyun Parkb, Yeongcheol Choi, and Mungyeong Choe. Virtual reality sickness questionnaire (vrsq): Motion sickness measurement index in a virtual reality environment. *Applied Ergonomics*, 2018.
- [22] Eric Kirchgessner, Matías Sothers, Valentina Aravena, Nelson Baloian, and Gustavo Zurita. *The Mind in Virtual Meetings: Comparing VR and Video Conferencing Environments Through Experiential Impact Assessment and EEG Analysis*, pages 255–267. 11 2023.
- [23] Eric Krokos and Amitabh Varshney. Quantifying vr cybersickness using eeg. *Springer*, 2021.
- [24] Thomas A. Stoffregen L. James Smart, Jr. and Benoît G. Bardy. Visually induced motion sickness predicted by postural instability. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 2002.
- [25] John LaRocco, Minh Dong Le, and Dong-Guk Paeng. A systemic review of available low-cost eeg headsets used for drowsiness detection. *Frontiers in Neuroinformatics*, 14, 2020.
- [26] S. Marcel and J. R. Millan. Person authentication using brainwaves (eeg) and maximum a posteriori model adaptation. *IEEE Transactions on Visualization and Computer Graphics*, 2007.

- [27] S. Martín-Aragón and J. Benedí. *Vértigo y mareo. Farmacia profesional*, 2004.
- [28] ByEmy Nguyen. Performance metrics, Jul 2023. <https://www.emotiv.com/knowledge-base/performance-metrics/>.
- [29] Omotea. <https://www.youtube.com/@omotea1926>.
- [30] M. D. Patterson, R. Winterbottom and B. J. Pierce. Perceptual issues in the use of head-mounted visual displays. *Human Factors*, 2006.
- [31] Epic Rollercoasters. https://store.steampowered.com/app/787790/Epic_Roller_Coasters/.
- [32] Gorilla Tag. https://store.steampowered.com/app/1533390/Gorilla_Tag/.
- [33] Unity Technologies. Xr. <https://docs.unity3d.com/Manual/XR.html>.
- [34] Unity Technologies. Xr interaction toolkit manual. <https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.5/manual/index.html>.
- [35] Asis Kumar Tripathy, Suchismita Chinara, and Mahasweta Sarkar b. Biocybernetics and biomedical engineering. *Biocybernetics and Biomedical Engineering*, 2016.
- [36] Virtux. The future of gaming. <https://omni.virtuix.com/>.
- [37] Hou Xiyuan, Liu Yisi, Sourina Olga, and Müller-Witting Wolfgang. Cognimeter: Eeg-based emotion, mental workload and stress visual monitoring. 2015.

Anexos

Anexo A

Transcripción del consentimiento informado utilizado en el experimento

Consentimiento

Objetivo:

El objetivo de esta actividad es recopilar datos de ondas cerebrales relacionados con el CyberSickness, utilizando un dispositivo de interfaz cerebro-computador (BCI). Esto se hace con el propósito de identificar que forma de moverse en un entorno virtual genera mayor intensidad de este malestar.

Durante la actividad, se llevarán a cabo unas pruebas de orientación usando un dispositivo BCI y un casco de realidad virtual. Para ello, se introducirá al sujeto a un entorno de realidad virtual con el BCI puesto, registrando las ondas cerebrales durante la realización de las pruebas.

Se pondrá a prueba una técnica de movimiento al azar de la siguiente lista: Caminata estándar, Caminata con gesto, Gancho.

Posibles riesgos:

Durante las pruebas se **espera** que el sujeto experimente síntomas como náuseas, malestar general, palidez, aumento del ritmo cardiaco o dolor de cabeza.

Es posible que, luego de terminar el experimento, alguno de estos síntomas se mantenga hasta el final del día en el peor de los casos.

Actividad:

La actividad seguirá los siguientes pasos:

1. Calibración del dispositivo: En esta etapa, se colocará el dispositivo y se verificará la calidad de la señal detectada por los sensores para asegurarse de que sea suficiente.

2. Uso del entorno de pruebas: Se llevarán a cabo las pruebas en las que el sujeto debe llegar a una meta.
3. Se realizará un cuestionario de autopercepción de los síntomas de CyberSickness.

Datos personales:

Nombre Género: Edad: Carrera estudiada:

Uso de datos: Los datos entregados en conjunto con los que se medirán, serán únicamente para el uso de la obtención de datos para este estudio, de forma anónima y no serán entregados en ningún contexto fuera de esta investigación.

¿Acepta el uso de sus datos con el propósito señalado anteriormente? (Sí / No)

¿Está consciente y acepta los síntomas y malestares que podrá sentir durante las pruebas?
(Sí / No)

Anexo B

Cuestionario

B.1. Caracterización

1. Nombre
2. Edad
3. Carrera
4. ¿Qué escenario usaste?
 - Laberinto
 - Pista de carreras
5. ¿Qué técnica de movimiento usaste?
 - Caminata normal
 - Caminata con gesto
 - Gancho
6. ¿Qué tan propenso eres a marearte al usar lentes de realidad virtual?
 - Nunca había usado uno
 - Poco
 - Normal
 - Mucho
7. ¿Qué tan seguido sueles jugar videojuegos?
 - Nunca
 - Un par de veces al mes
 - Semanalmente
 - Diariamente

B.2. Síntomas

¿Qué síntomas sentiste? (Selección múltiple)

- Malestar general
- Cansancio
- Vista cansada
- Dificultad para enfocar
- Dolor de cabeza
- Pesadez de cabeza
- Visión borrosa
- Mareo con ojos cerrados
- Vértigo

B.3. Auto percepción

1. ¿En qué momento fue que sentiste un mayor malestar?
2. ¿Qué elementos fueron los que más te marearon?

B.4. Calidad de la aplicación

Responde sin considerar la etapa de preparación (Conexión de los diodos y del casco de realidad virtual)

1. Responde que tan de acuerdo estás con las siguientes afirmaciones
 - (a) El movimiento respondía correctamente
 - Totalmente en desacuerdo
 - En desacuerdo
 - Neutro
 - De acuerdo
 - Totalmente de acuerdo
 - (b) La forma de moverse fue fácil de comprender
 - Totalmente en desacuerdo

- En desacuerdo
 - Neutro
 - De acuerdo
 - Totalmente de acuerdo
- (c) La forma de moverse fue fácil de comprender
- Totalmente en desacuerdo
 - En desacuerdo
 - Neutro
 - De acuerdo
 - Totalmente de acuerdo
- (d) Mi objetivo en la prueba estaba claro
- Totalmente en desacuerdo
 - En desacuerdo
 - Neutro
 - De acuerdo
 - Totalmente de acuerdo

2. ¿Tienes algún comentario adicional respecto a la aplicación?