



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

DISEÑO Y DESARROLLO DE PROTOTIPO DE APLICACIÓN EN REALIDAD
VIRTUAL COMO COMPLEMENTO DE TERAPIAS DE EXPOSICIÓN CONTROLADA
PARA EL MANEJO DE FOBIAS

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

FELIPE JOSÉ ORTÚZAR GARCÍA-REYES

PROFESOR GUÍA:
FRANCISCO GUTIÉRREZ FIGUEROA

MIEMBROS DE LA COMISIÓN:
DANIEL CALDERÓN SAAVEDRA
ANDRÉS ABELIUK

SANTIAGO DE CHILE
2022

Resumen

Las fobias son trastornos de ansiedad que dificultan o imposibilitan a los afectados, al introducir un miedo irracional hacia un objeto, animal o situación. Existen tratamientos para este trastorno que permiten disminuir o eliminar esta falencia, pero debido a que ellos implican la confrontación con el objeto fóbico, existen pacientes que no inician o continúan con el tratamiento. Paralelamente, las tecnologías de realidad virtual han surgido rápidamente en este último tiempo, involucrándose en áreas de simulación, videojuegos, y en particular, la salud. Gracias a su capacidad de simular situaciones o eventos difíciles de replicar en la vida real, ya sea por su coste, imposibilidad o peligro, se han desarrollado aplicaciones que simulan terapias de exposición en ambientes virtuales para el tratamiento de fobias. Estas aplicaciones permiten al paciente tratarse en un ambiente seguro y los resultados en los tratamientos que usan realidad virtual llegan a ser tan efectivos como aquellos realizados en el mundo real. Estas aplicaciones, muchas veces usan dispositivos especializados, de alto costo, y que simulan el tratamiento para una fobia en particular. Es por esto que existe la oportunidad de proveer un simulador para apoyar terapias de fobias en ambientes virtuales que utilice cascos de realidad virtual no especializados, a bajo costo, y que asista en el tratamiento del trastorno para una variedad de fobias y sea capaz de modelar gráficamente al objeto fóbico en sintonía con la gravedad de la ansiedad del usuario.

En el presente trabajo de título se diseñó y desarrolló un prototipo de aplicación en realidad virtual para apoyar terapias de manejo de fobias. Para esto, se eligió el motor de videojuegos Unity y el casco de realidad virtual Oculus Quest 1 para realizar una aplicación donde el usuario debe encontrar a un modelo virtual, representando a su fobia, en un escenario virtual. Usando las herramientas del motor de videojuegos se implementaron las escenas del mundo virtual y se le agregó el comportamiento de interacciones en realidad virtual, mediante el paquete de Unity *XR Interaction Toolkit*. Para obtener los recursos gráficos usados en la aplicación, se utilizó la tienda virtual *Unity Asset Store* y junto a esto se desarrolló el dinamismo visual para modificar el objeto fóbico presente en la escena, a partir de las necesidades psicoterapéuticas específicas del usuario. Una vez desarrollada la aplicación, se realizó una prueba de usuario, para identificar aspectos de funcionalidad, jugabilidad y presencia percibida por los usuarios. Finalmente, se recopilaron los resultados de la prueba y se analizaron para conocer el estado del producto junto con sus aspectos positivos y aquellos que necesitan un trabajo futuro. Los aprendizajes fueron que el aplicativo funciona correctamente y da la sensación de estar inmerso en una realidad virtual. Además, se constató que se necesita disminuir la sensación de mareo, y mejorar la capacidad de interacción de la escena. A partir de lo anterior, se concluye que el trabajo presentado cumple con los objetivos propuestos, pero existen limitaciones a la generalidad de los modelos.

Agradecimientos

En primer lugar, quiero agradecer a mi profesor guía por las reuniones, su conocimiento y los comentarios dados durante todo el proyecto. Su apoyo fue crucial para cubrir las áreas en las que no tenía experiencia, y su rigurosidad y asistencia fueron un pilar en el desarrollo esta memoria.

También me gustaría agradecer a mi familia, quienes me han proporcionado un gran cariño y apoyo siempre, y me han demostrado la capacidad de seguir adelante. Gracias por soportar junto a mi las grandes penurias, pero también vivir conmigo las partes afortunadas de la vida.

Infaltablemente a mi polola, cuyo cariño, compañía y apoyo es algo que me llena de felicidad y me han acompañado siempre en esta relación. Gracias a ella, los momentos buenos fueron muy buenos, y los malos no tanto.

Un agradecimiento a mis amigos del colegio por seguir en las andanzas de los Rebux y seguir siendo un grupo unido. No puedo olvidar al Tata por su compañía y terapias durante la universidad, sus consejos y conversaciones me ayudaron en todos los momentos, y agradezco los buenos ratos pasados pasados en su compañía. Finalmente agradecer a mis más cercanos de la universidad, Pablo y Nico, con quienes siempre agradezco tener una conversación y pasar el rato.

Tabla de contenido

1. Introducción	6
1.1. Problema abordado	7
1.2. Objetivos	9
1.2.1. Objetivo General	9
1.2.2. Objetivos Específicos	9
1.2.3. Evaluación	9
1.3. Resumen de la solución desarrollada	10
1.4. Alcance del proyecto	11
1.5. Estructura del documento	11
2. Estado del Arte	13
3. Diseño del prototipo	16
3.1. Requerimientos psicoterapéuticos	16
3.1.1. Múltiples contextos	16
3.1.2. Realismo dinámico	17
3.2. Múltiples Modelos	18
3.3. Dinámica de la actividad principal	18
3.4. Tutorial	19
3.5. Motor de videojuego	19
3.6. Elementos en la aplicación	20
3.6.1. Casa segura	20
3.6.2. Gráficos	20
3.6.3. Movimiento y física	21
3.6.4. UI, Láser y botón de salida	21
3.6.5. Tiempo de partida	22
3.6.6. Estadísticas de la actividad	22
3.6.7. Estados y transiciones	22
3.6.8. Tutorial	24
3.6.9. Actividad principal	25
4. Implementación de la actividad	26
4.1. XR Interaction toolkit	26
4.1.1. Soporte para diferentes cascos	26
4.1.2. Interacción con el ambiente virtual	27
4.1.3. Interacción con la interfaz de usuario	31

4.1.4. Seguimiento virtual del casco	32
4.2. Recursos gráficos	35
4.2.1. <i>Unity Asset Store</i>	35
4.2.2. <i>ProBuilder</i>	35
4.2.3. Generación de los terrenos	36
4.3. Menú principal	37
4.3.1. Espacio	37
4.3.2. Elementos de la interfaz de usuario	37
4.3.3. Regulación gráfica	38
4.4. Tutorial	40
4.4.1. Terreno y aspecto general	40
4.4.2. Flujo de eventos	41
4.4.3. Problemas con el <i>build</i> al casco	42
4.5. Actividad principal	43
4.5.1. <i>GameManager</i>	44
4.5.2. Casa segura	45
4.5.3. Logros	46
4.5.4. Iluminación	46
4.5.5. Escondites	47
4.5.6. Casa	48
4.5.7. Bosque	49
5. Prueba de concepto y análisis de los resultados	52
5.1. Metodología	52
5.1.1. Participantes	52
5.1.2. Equipo	53
5.1.3. Cuestionarios	53
5.1.4. Procedimiento	54
5.1.5. Recolección y análisis de datos	55
5.2. Resultados y análisis	56
5.2.1. Mareos	56
5.2.2. Jugabilidad	58
5.2.3. Presencia	61
5.3. Piloto de prueba para el bosque	64
5.3.1. Metodología	64
5.3.2. Resultados y análisis	65
5.4. Aprendizajes principales	66
6. Conclusiones y trabajo futuro	68
7. Anexo	75

Índice de ilustraciones

3.1.	Representación gráfica de la máquina de estados	23
4.1.	Componente Grab Interactable	28
4.2.	Capacidad de asociar eventos	29
4.3.	Componente XR Controller	30
4.4.	Mano izquierda en posición de agarre y mano derecha en posición de disparo.	30
4.5.	Componente XR Direct Interactor	31
4.6.	Capacidad de asociar eventos para el usuario	31
4.7.	Láser presionando un botón de la IU	32
4.8.	Diagrama Componentes agregados a VROrigin	33
4.9.	Construyendo una casa usando <i>ProBuilder</i>	36
4.10.	Creación de terreno usando la herramienta Terrain de Unity.	36
4.11.	Escena del menú principal.	37
4.12.	Diagrama de PlayerPreferences	38
4.13.	Modelos de fidelidad para la araña. De izquierda a derecha: <i>Low</i> , <i>Medium</i> , <i>High</i>	39
4.14.	Mapa del tutorial	40
4.15.	Elementos de naturaleza en el tutorial	41
4.16.	Versión del tutorial previo a problemas del <i>build</i>	42
4.17.	Instrucciones iniciales en la IU de la casa	44
4.18.	Diagrama UML del componente	45
4.19.	Escena de logros	46
4.20.	Lugar posible de escondite para Fobos	47
4.21.	Vista del primer piso de la casa	48
4.22.	Vista del segundo piso de la casa	49
4.23.	Vista del mapa de bosque	50
5.1.	Resultados cuestionario jugabilidad (1)	59
5.2.	Resultados cuestionario jugabilidad (2)	59
5.3.	Resultados cuestionario presencia	61
5.4.	Tabla de mediana por categoría	63

Capítulo 1

Introducción

En el Manual de Diagnóstico y Estadísticas de Trastornos Mentales [9], la Asociación Americana de Psiquiatría define la fobia como un tipo de trastorno de ansiedad que consiste en un miedo excesivo y persistente a un objeto, situación o posibilidad. Este trastorno, explican, viene con un cambio en el comportamiento del individuo que lo sufre, lo que incluye algún tipo de respuesta frente a la fobia, como la activación de instintos de lucha y huida, pero también formas más extremas como los ataques de pánico y ansiedad. Para entender las fobias, en el Manual aclaran que el miedo es una respuesta emocional a un peligro real o próximo, y también que la ansiedad es una respuesta de anticipación a una amenaza futura. Por esto, la fobia se separa de estas dos respuestas a medida que se sobreestima el peligro de la amenaza, y este miedo o ansiedad persiste sobre un período de tiempo no apropiado.

Los efectos de la fobia en seres humanos incluyen cambios en su comportamiento natural y en las respuestas fisiológicas del cuerpo. En general, ambos cambios están relacionados entre sí; en particular, el comportamiento normal del individuo que se ve afectado por las respuestas biológicas extremas frente a la fobia. Esto se observa en la huida o evasión de la persona cuando se ve enfrentada a la situación que le causa angustia, incomodidad o alguna respuesta corporal, como por ejemplo: cansancio, sobre-estimulación, propensión a la fatiga, dificultad para concentrarse, tensión muscular, problemas de sueño o irritabilidad [9].

La Asociación define principalmente tres tipos de fobias [9], y cada una de ellas se considera un subtipo de trastorno de ansiedad. En primer lugar, están las fobias específicas, que son aquellas que tratan de un miedo excesivo focalizado a objetos, situaciones o lugares. Las fobias específicas se pueden clasificar a su vez en cuatro: Hacia animales, situaciones, ambientes o inyecciones al cuerpo humano. En segundo lugar, está la agorafobia, que es una fobia relacionada a los espacios. Si bien el ejemplo más conocido es la fobia a los espacios cerrados o “enclaustrados”, también puede referirse a los espacios abiertos, viajes en vehículos de transporte, ponerse en una cola de espera o estar en un ambiente con mucha gente. Usualmente el individuo siente miedo de la dificultad de escapar de esos lugares, o bien de que su fobia se active de manera subconsciente y tenga un ataque de pánico en un lugar complejo o con mucha gente observando. El último tipo es la fobia social o trastorno de ansiedad social. Esta es el miedo o ansiedad de ser juzgado o analizado por otros. Tener una conversación, conocer gente nueva y ser observado haciendo algo pueden ser razones para que la persona

sienta ansiedad o miedo, al existir la posibilidad de ser evaluado negativamente en el ámbito social.

Según la Asociación Americana de Psiquiatría, en Latinoamérica, entre un 2 % a 4 % de las personas sufre algún tipo de fobia específica [9]. Esos números difieren en otros continentes, como Europa o Estados Unidos, con un 6 % a 9 % de padecimiento entre la población. Con respecto al género, las mujeres son el doble de susceptibles que los hombres a generar y mantener una fobia. Etariamente también existen diferencias, pues mientras los infantes tienen un 5 % de posibilidad de generar fobias, las y los adolescentes (13 a 17 años) tienen un 16 %, y en individuos mayores baja a un 3 % – 5 %. Números similares ocurren al comparar con el trastorno de ansiedad social y agorafobia.

1.1. Problema abordado

Los tratamientos principales para las fobias son la terapia de exposición (TE) y la terapia de comportamiento cognitivo (TCC). Por un lado, la terapia de exposición consiste en exponerse al objeto o situación que provoca la fobia de manera incremental, constante y partiendo de manera leve al inicio. Así, el paciente puede ir confrontando el miedo en un ambiente que no lo expone a peligros, por lo cual la persona puede acostumbrarse a la cercanía del objeto o situación con mayor tranquilidad. Esta terapia usualmente dura 1 a 3 horas de exposición y se realiza todas las semanas [41]. Por otro lado, la terapia de comportamiento cognitivo consiste en la terapia de exposición pero junto con otras técnicas que ayudan al paciente a interiorizarse con sus sentimientos y sensaciones que le afectan al sentir la fobia, además de enfocarse en el reconocimiento de la evasión que dicha persona realiza cuando se encuentra en cercanías de lo que le provoca miedo. La evasión muchas veces es el principal factor de disminución a la calidad de vida del paciente, y por esto, que éste pueda aprender su comportamiento y acciones subconscientes que realiza por la fobia suele dar y mejorar la calidad de la terapia [41].

Pese a ser la terapia más eficaz, el gran problema de las terapias *in vivo* es que tienen altos índices (30 %) de abandono o rechazo de la terapia una vez el terapeuta explica de qué se tratará, junto con índices no menores (10 %) de abandono del tratamiento una vez se ha iniciado [18]. Esto ocurre, según Choy et al., [7], debido a que el proceso explicado es muy amenazante o aversivo. Además, pueden llegar a existir problemas entre terapeuta y paciente que se deben a que este último duda o está en contra de la terapia que se le propone. El paciente puede pensar que no se han tomado las consideraciones éticas pertinentes o encontrar la terapia intensa o “extraña” debido a que le provoca excesiva angustia y malestar realizar el tratamiento [31]. Esto ha llevado a provocar que terapeutas tengan una opinión negativa de las terapias que actualmente se usan para tratar la fobia de sus pacientes [10, 36], donde algunos la consideran incluso aversiva [37].

Como respuesta a estas dificultades y reacciones negativas de las TE y TCC, investigadores han experimentado y analizado las terapias de exposición en ambientes virtuales (TEAV). Existen varias ventajas de las TEAV por sobre las TE *in vivo*. En primer lugar, las TEAV son capaces de generar ambientes virtuales que de otra manera serían muy difíciles de conseguir o recrear [32]. Muchas personas padecen de fobia a las alturas, las tormentas eléctricas, volar, multitudes de personas, animales peligrosos, etc [17]. Sería costoso, poco ético y hasta

peligroso montar al paciente en una de esas situaciones, por lo que crear un entorno de realidad virtual que simule esos eventos puede llegar a ser la solución más apropiada para estos casos, puesto que no presentan peligros físicos al paciente. En segundo lugar, las TEAV proporcionan un ambiente virtual que está exento de los peligros reales que una persona teme. Por esto mismo, la preferibilidad de las TEAV sobre las terapias *in vivo* pueden llegar a ser mucho mayores, como fue registrado por García-Palacios et al., [12] al pedirles a 162 alumnos con alto miedo a la arañas si es que preferían realizar TE o TEAV en su investigación. El resultado fue que un 81 % de ellos prefirió TEAV. En tercer lugar, y esto fue en gran medida incrementado por la pandemia del COVID-19, con instrumentos de realidad virtual es posible generar terapias [11] a distancia donde el paciente pueda seguir con su tratamiento sin necesidad de desplazarse ni de tener al terapeuta junto a él.

La efectividad de las TEAV está directamente asociada a tres factores [19]: (1) sensación de presencia en el entorno virtual, (2) provocación y estimulación de emociones y ansiedad, y finalmente, (3) transferencia de esas emociones a la vida real. Estos factores no están completamente resueltos, como fue revisado por Lindner et al. [25] en su estudio sobre las actitudes y familiaridad de terapeutas cognitivo-conductuales ($n = 185$) hacia TEAV. En este documento se muestra que: los pacientes no ven el ambiente virtual como lo suficientemente real; que existe una baja calidad de software; que existen dificultades técnicas asociadas al uso de las herramientas tecnológicas; y que las TEAV no producen resultados que se traduzcan a la vida real. Entre los aspectos positivos se destacan que las TEAV pueden lograr ambientes virtuales que en la realidad serían imposibles, que el ambiente virtual se puede controlar de manera precisa, y que pueden hacer de la terapia de exposición una menos estresante usando componentes lúdicos, entre otros.

Los videojuegos serios son otra herramienta relativamente nueva que ha sido empleada con una frecuencia cada vez mayor en el área de la medicina, y en particular para el uso de un conjunto grande de terapias [13]. Estos son usados en psicoterapia, terapia ocupacional, manejo del dolor, rehabilitación cognitiva y en muchas otras áreas, debido a sus características de entretenimiento e inmersión [13]. Los componentes lúdicos le pueden presentar al usuario contextos nuevos y desafiantes, y su uso en terapias es particularmente fuerte al tener la capacidad de presentar a pacientes ansiosos estímulos aversivos para ayudarlos en su padecimiento, esto gracias a la inmersión a través del juego [13]. No es sorpresa entonces, que ya existan reportes de investigadores y terapeutas que usen la “*gamificación*” [8], que es la utilización de elementos lúdicos en área externas al entretenimiento, en TEAV para aumentar la efectividad e involucramiento de los pacientes en sus terapias. [21, 28]

Trabajar con herramientas de realidad virtual es un desafío técnico, pues requiere resolver un conjunto de problemas que ya se encuentran en herramientas de simulación o videojuegos como lo son: sistemas de *rendering*, audio, manejo de objetos, manejo de colisiones, sistema de animación, administración eficiente de recursos, etc. Si es que se habla de realidad virtual (VR en inglés), entonces sería necesario agregar manejo de input en tres dimensiones con rotación del sistema visual y un sistema espacial de audio. Son estos los nuevos desafíos técnicos presentes en desarrollo de software en realidad virtual, y que han aparecido con el crecimiento de la demanda de estas tecnologías en ámbito del entretenimiento. Por eso, han ido apareciendo más y mejores pautas de diseño y kits de desarrollo para este dominio técnico.

Es en consecuencia de los problemas de altos índices de rechazo y abandono que ocurren en las terapias *in vivo*, que se origina el presente trabajo de título. En este, se propone explorar sobre las TE, TEAV y uso de los *videojuegos serios* para desarrollar un prototipo de *software*. Este, propondrá avances en encontrar una solución a la necesidad de generar herramientas que ayuden a terapeutas y pacientes en la búsqueda de un tratamiento dinámico, general y que no impida al paciente tratarse.

En particular, el alcance específico de la memoria es generar un prototipo de aplicación funcional en realidad virtual que permita regular gráficamente un modelo virtual en función del usuario, y que sea potencialmente útil para el apoyo a terapias de manejo de fobias.

1.2. Objetivos

A continuación se describirán los objetivos generales y específicos se buscan cumplir en la presente memoria.

1.2.1. Objetivo General

El objetivo general de este trabajo consiste en diseñar y construir una herramienta de software que sea potencialmente efectiva en TEAV respecto a su capacidad de generar un estímulo que sea beneficioso a aquellos pacientes sufriendo algún tipo de fobia en específico, pero que tenga suficiente capacidad de generalización a diferentes fobias y modelos gráficos.

Este avance se espera que sea más atractivo para personas que necesiten tratamiento para las fobias, pero que no han podido o querido asistir a un terapeuta por miedo de que la terapia sea muy invasiva y le provoque malestar en exceso.

1.2.2. Objetivos Específicos

1. Diseñar un prototipo de aplicación con un sistema de regulación gráfica de los modelos en el mundo virtual que sean esenciales¹. Esta regulación deberá proporcionar al usuario una potencial sensación de incomodidad que podrá ser beneficiosa en una terapia de exposición.
2. El diseño anterior debe seguir directrices de profesionales en el tema (por ejemplo, psicólogos y psiquiatras), dado el carácter multidisciplinario del tema que se compone de elementos psicoterapéuticos y de diseño de sistemas computacionales interactivos.
3. Construir un software sobre el diseño definido. Este debe incorporar herramientas de inmersión virtual con compatibilidad hacia algún *headset* (lente o casco con capacidad de mostrar una realidad virtual) de realidad virtual. El uso de estas herramientas ayudarían a aumentar la inmersión del software.

1.2.3. Evaluación

La evaluación de la completitud de los objetivos mencionados anteriormente se lista a continuación y en el mismo orden en el que fueron enumerados.

¹Ese decir, modelos que induzcan la fobia al usuario

1. Para comprobar la realización de este punto se debió realizar un análisis sobre la independencia de las tareas o actos que deba realizar el usuario, contra la representación o tipo de modelo. Por ejemplo, estaría mal calificado si el modelo empieza a trepar como si fuese una araña, cuando en realidad es un perro.
2. Se debió validar formalmente la opinión de un experto de dominio en el tema, el que evaluó la interacción del usuario con el modelo (araña, perro, cucaracha, etc.). Por ejemplo, “El modelo se presenta muy lejano y no hay interacción” o “El modelo aparece muy agresivamente en pantalla y debería aparecer de manera más gradual”.
3. El software generado debe ser funcional y operativo, y se debieron hacer pruebas de usuario, que midieron aspectos de usabilidad, jugabilidad y presencia en el mundo virtual para comprobar el sistema, además de seguir los alineamientos definidos en el diseño.

1.3. Resumen de la solución desarrollada

La solución desarrollada es un prototipo de aplicación en realidad virtual, con un modelamiento dinámico y generalidad de elección del objeto fóbico, el que deberá ser buscado por el usuario en un mundo virtual. La exploración realizada por el usuario implica que este debe interactuar con el mundo de una manera activa; debe moverse, observar y buscar entre espacio grandes y pequeños el objeto fóbico (en adelante “Fobos”), y esto genera una sensación de incomodidad que es clave para la superación del miedo. A lo largo de varias sesiones, la inocua incomodidad sentida se espera que se vea reducida gracias a la desensibilización al estímulo del miedo, en otras palabras, se simula una TEAV.

La aplicación contiene principalmente dos módulos: el tutorial y la actividad principal. En el tutorial, el usuario aprende cómo moverse e interactuar con el entorno virtual, y en la actividad principal debe explorar y encontrar a Fobos. En este segundo módulo, el usuario puede elegir a qué especie o modelo visual representa Fobos, por ejemplo, una araña, serpiente, estatua, por nombrar algunos. Pero además, junto a esto, es posible definir qué tipo de realismo virtual tendrá: bajo, medio o alto son las opciones por cada tipo de modelo. Esto afectará la calidad de las texturas utilizadas y la cantidad de vértices que tiene Fobos, con el fin de cambiar qué tan real se ve en el mundo virtual.

Gracias a esta decisión de diseño, el modelo virtual se vuelve independiente del mapa y la interacción de Fobos en el mundo. Junto a esto, la arquitectura está diseñada también con este propósito, puesto que el escalamiento hacia más modelos en el futuro está facilitada.

Estas decisiones de diseño estuvieron sujetas a los comentarios que se recibieron de la experta en el dominio: la Doctora y Profesora Vanetza E. Quezada Scholz ², académica del Departamento de Psicología de la Universidad de Chile, cuya línea de investigación considera el aprendizaje de trastornos de ansiedad y desarrollo del miedo, la que accedió a asumir este rol de guía. Estas mecánicas de interacción de Fobos con el usuario fueron evaluadas y se modificaron gracias a las entrevistas/revisiones al diseño de la solución. Las modificaciones que ocurrieron en la etapa de diseño fueron de manera incremental, es decir, en la medida que se recibieron sugerencias por parte de la experta de dominio se fueron haciendo los cambios.

²<http://www.facso.uchile.cl/psicologia/departamento-de-psicologia/57902/vanetza-quezada>

Componentes lúdicos tendrán el objetivo de aumentar la interacción del usuario e impulsarlo a concentrarse en el mundo virtual. Técnicas similares fueron usadas por Levy et al., [21] con objetivo de aumentar la presencia de los jugadores en la tarea que se les encomendaba. De esta manera, el usuario sentirá mayor nivel de interés y entretenimiento en una actividad que lo ayuda a enfrentar su miedo o ansiedad.

Aspectos como la jugabilidad y presencia fueron evaluadas con un cuestionario para cada tema. La medición de la presencia se basó en la *Escala de Presencia Multimodal* [27] de Makranski et al.. Por otro lado, la jugabilidad se midió con la *Gameful Experience Questionnaire* [16] de Högberg et al..

Finalmente, la solución proporciona un desarrollo de una aplicación con tecnologías modernas y relativamente nuevas en el mercado, como son los cascos de realidad virtual y los motores de videojuegos. Y así la solución podrá ser una potencial mejora a las TEAV, gracias a que presenta características de presencia y jugabilidad que la ayudan para su aceptación y realismo hacia posibles personas sufriendo de alguna fobia específica.

1.4. Alcance del proyecto

El alcance del proyecto corresponde a generar un prototipo de aplicación en realidad virtual para apoyar terapias de exposición para el manejo de fobias. Esto incluye encargarse de que los elementos de la aplicación estén correctamente construidos desde un punto de vista funcional y que las decisiones de diseño involucradas tengan el propósito de apoyar las terapias de manejo de fobias. Lo anterior no incluye validar el uso de terapias en ambientes virtuales para el manejo de fobias, puesto que eso ya ha sido validado y no entra dentro de las capacidades esperadas del autor. De la misma manera, en su calidad de trabajo de memoria, tampoco se incluye validación de la aplicación dentro de una perspectiva terapéutica, es decir, no se validará si es que el prototipo desarrollado ayuda a disminuir la sensación de fobia en gente afectada. Esto último se espera que sea el siguiente paso en el desarrollo de este prototipo para un trabajo futuro.

1.5. Estructura del documento

El resto del documento está estructurado de la siguiente forma:

- El Capítulo 2 presenta el estado del arte para entender cómo en la literatura se abordan las Terapias de Exposición en Ambientes Virtuales, y con esto entender cuáles son las necesidades y oportunidades que motivan esta memoria, y qué elementos o características se pueden utilizar para obtener mejores resultados.
- El Capítulo 3 corresponde al diseño de la aplicación, aquí se explica el razonamiento para las diferentes características de la solución, en términos de diseño, elementos psicoterapéuticos y la arquitectura del software.
- En el Capítulo 4 se muestran las características de la solución junto con los elementos que fueron agregados, eliminados o editados en las diferentes etapas de retroalimentación del proceso de desarrollo.

- En el Capítulo 5 está la validación y pruebas de concepto para evaluar la calidad del software y cómo es recibido por diferentes usuarios de prueba, en base a cuestionarios pre y post experimento.
- El Capítulo 6 presenta el fin del documento con las conclusiones que afloraron a lo largo del proceso completo de la memoria, y además se presentan las oportunidades que se abren para un trabajo futuro sobre esta solución.

Capítulo 2

Estado del Arte

En este capítulo se presenta el estado del arte de las TEAV, junto a reportes generales y los diversos mecanismos que los investigadores han agregado a la composición de estas terapias para encontrar variaciones o mejorar la eficacia de los tratamientos previos. Estos avances incluyen elementos como audio espacial, interactibilidad con el medio y *gamificación* para mejorar involucramiento de los pacientes en sus terapias.

Partiendo con lo relativo a las terapias de exposición en ambientes virtuales, Parsons et al. [33] realizó un reporte de las investigaciones sobre la eficacia de las TEAV en ámbitos de investigación, haciendo un filtro sobre aquellas que excedieran ciertos criterios estadísticos: existencia de datos previos y posteriores al uso del software, que sean en inglés, que trabajen sobre un tamaño de muestra mayor a 10 personas, que reporten valores de media y desviación estándar, etc. Este reporte encontró 21 investigaciones que pasaron sus filtros, con un tamaño de muestra total a 300 personas. Las fobias estudiadas fueron fobias sociales, aracnofobia, acrofobia, agorafobia, aviofobia y PTSD. Este meta-estudio concluye que las TEAV pueden ser efectivas desde un punto de vista psicoterapéutico.

Una consideración importante al utilizar tecnologías de realidad virtual es el malestar que podría ser experimentado por los usuarios al usar estas aplicaciones. Este problema fue reconocido hace varios años, en 1997, por Lewis et al. [22] al estudiar los usos terapéuticos de la realidad virtual. Estos investigadores mencionan que las molestias podrían ser problemas oculares, náuseas, perturbaciones en el balance o desorientación. Es por esto que existen recomendaciones para que las aplicaciones sean lo más ameno posible con los usuarios, como por ejemplo: disminuir la velocidad de movimiento de translación, disminuir los movimientos de cabeza requeridos, aumentar gradualmente la exposición e interacción de elementos en el ambiente virtual para usuarios con poca experiencia y utilizar cascos de realidad virtual con ajuste a la distancia inter-pupila.

Otra investigación se enfoca en la opinión de terapeutas comunes que han tenido experiencia con TEAV, y de cuáles han sido sus dificultades y facilidades al trabajar con estas herramientas [25]. Los investigadores entregaron a los terapeutas un cuestionario con diferentes aspectos sobre su usabilidad y eficacia. El estudio, que ya se había hecho antes, mostró que la opinión sobre TEAV ha mejorado con el tiempo y que los terapeutas tienen una opi-

nión en promedio positiva. Pero, todavía existen problemas con el realismo, calidad, costo, usabilidad y obtención del software. En general, los estudios sobre TEAV abarcan resultados positivos en lo que respecta a la eficacia de este tipo de tratamientos y sobre su potencial a futuro [1], pero tienden a ser en gran medida específicos a una fobia particular y los elementos permanecen estáticos a lo largo de la terapia.

Un experimento realizado por Taffou et al. [38] comparó el nivel de estimulación al estar cerca de perros en realidad virtual sobre individuos con Cynophobia (fobia a los perros). Pero lo novedoso de esto fue que compararon el nivel de estímulo provocado por perros en un lugar sin sonidos ambientales y otro con. Los resultados fueron que este sistema de sonido espacial (que se modifica en función de la posición del usuario), aumentó el estímulo recibido por el usuario en comparación a cuando no había sonido. Esto implica que existen técnicas ambientales que se pueden utilizar para aumentar el estímulo percibido cuando es necesario en realidad virtual.

Existe una gran cantidad de investigaciones que ahondan más en distintos escenarios en las que estos ambientes pueden ser útiles: Minns et al. [30] utilizan cinematografía en VR para estudiar los beneficios de esta técnica para tratar la fobia en arañas ($n = 77$); Peskin et al. [34] trabajaron sobre pacientes ($n = 25$) con Estrés Post-Traumático hacia el atentado a las Torres Gemelas en EE.UU, y constataron sobre el beneficio de las TEAV en el síndrome, pero subsecuentemente en el cuadro depresivo que esto les causaba; otro trabajo de Lindner et al. [24] muestra que cascos virtuales de bajo costo y no especializados son también herramientas útiles para las TEAV enfocadas a la ansiedad provocada por hablar en público; Beidel et al. [2] concluyen gracias a sus resultados que el uso de TEAV en pacientes con el Trastorno de Estrés Post-Traumático en veteranos estadounidenses de Iraq y Afganistán es una herramienta que no sólo disminuye las mediciones del trastorno, sino que también disminuye otras variables como la depresión y el manejo de la ira que está presente en estos casos. A modo de cierre con respecto a la actitud de la comunidad académica frente al avance y desarrollo de TEAV como tratamiento eficaz frente a trastornos de ansiedad y en específico a fobias, Powers et al. [35] mencionan que los entornos en realidad virtual permiten al usuario acceder a una experiencia que proporciona una sensación de presencia en el mundo virtual. Junto a esto, este uso se destaca por su desempeño positivo en varias terapias para tratar trastornos de ansiedad, como Trastornos de Estrés Post-Traumáticos, distintos tipos de fobias y trastornos sociales, que su tratamiento ayuda también a mejorar otros aspectos como trastornos depresivos que correlacionan a lo anterior.

Por otro lado, el uso de videojuegos en ámbitos distintos al entretenimiento (“Videojuegos serios”), ha ido aumentando en el área médica a medida que se ha mostrado potencial en su uso [20]. En particular, y con respecto a las TEAV también se han hecho varias investigaciones [4, 21, 26] que han demostrado la utilidad que pueden ofrecer a tratamientos ya existentes, que pueden ser mejorados con mécanicas pertenecientes al área de los videojuegos.

En cuanto a investigaciones específicas que utilicen Videojuegos Serios en las TEAV, Miloff et al. [28] compararon la efectividad de una Terapia de una Sesión tradicional contra una Terapia de una Sesión en realidad virtual con elementos de videojuegos. Lo interesante de esta investigación es que utilizaron situaciones en donde las personas debían realizar tareas al estilo de un videojuego. Además, a la araña, que era el elemento para estudiar la aracnofobia, le

modificaron la apariencia de manera que se hacía más o menos real dependiendo del progreso de la persona en el juego. En esta investigación, por primera vez se observa un elemento dinámico en una TEAV, en este caso el nivel gráfico de la representación de la araña. La limitación de esta investigación consiste en que la interacción con el elemento a temer, es decir, la araña, es un tanto específica con sus características, y no se podría por ejemplo, reemplazar el modelo con el de algún otro animal.

En general, en la literatura revisada no se mencionan las características del hardware, y en particular las del software utilizados en las TEAV. Por lo tanto es difícil hacer un análisis sobre los aspectos de eficiencia, como FPS (*Frames per second*: Cuadros por segundo en inglés) o calidad de la imagen, y desarrollo, como kits de desarrollo, librerías y motores de videojuegos. Respecto a la literatura dónde sí fue mencionada esa información, Bouchard et al. [4] aprovecharon la existencia de un videojuego popular en realidad virtual llamado Half-Life para hacer su investigación, pero no mencionan el *hardware* utilizado. En otra investigación, Botella et al. [3] utilizaron una aplicación de celular en Android para hacer rutinas previas a la TEAV, y así permitir al paciente realizar ciertos avances desde su celular sin necesidad de estar con su terapeuta. Cabe destacar que esta última investigación también ocupó los “Juegos serios” para aumentar la adherencia y aceptación al tratamiento de exposición.

De las investigaciones observadas, la única donde explicitan directamente el hardware utilizado es la desarrollada por Lindner et al., [23], en donde utilizaron un *Samsung Galaxy S6* y *Gear VR Headset*. Además mencionan que los experimentos fueron llevados en el ambiente virtual creado por el videojuego *Itsy*, disponible en el *marketplace* de Oculus¹. En esta investigación, los participantes realizaron TEAV con elementos de videojuegos de manera autónoma, fuera de ambientes de laboratorio y sin estar en presencia de los investigadores. En base a sus resultados, concluyeron que TEAV autónomas presentaron efectividades similares a las obtenidas con acompañamiento de investigadores, y pueden llegar a ser tratamientos prometedores para la aracnophobia.

Revisando lo anterior, se puede observar que existe un interés de investigadores y psicoterapeutas en trabajar con TEAV y elementos de *gamificación* para solucionar algunos problemas presentes en terapias de exposición, como los altos índices de abandono o rechazo a TE por la intensa incomodidad que podrían provocar, o también por la dificultad insertar al paciente en escenarios difíciles o costosos de recrear. Pero si bien existen algunas investigaciones, que realizaron propuestas donde se exhiben características dinámicas en los modelos o mecánicas (cambios en la representación gráfica de la araña) de la simulación, existe la oportunidad de generar soluciones con mayor grado de generalidad y adaptabilidad en el uso de modelos y gráficas para adaptarse a un conjunto mayor de pacientes.

¹https://www.oculus.com/experiences/gear-vr/1046954465328733?ranking_trace=0_1046954465328733_SEARCH_c544d1c4-436b-4f11-b644-2cc7f631ddff

Capítulo 3

Diseño del prototipo

Este capítulo contiene la información del diseño del prototipo. En particular se hablará de cuál será su flujo de acción para el usuario, qué elementos debe tener y cuáles serán las herramientas para el desarrollo.

3.1. Requerimientos psicoterapéuticos

Antes de definir las diferentes decisiones de diseño que nacen a partir de requerimientos psicoterapéuticos, es necesario entender cómo diferentes detalles contextuales influencian la fundación y expresión de una fobia y que, reconociendo esas conexiones, es posible desacoplarlas y disminuir la gravedad del trastorno.

A partir de lo que explican Torrent-Rodas et al., en su investigación *Maximizar la terapia de exposición: Un enfoque basado en el aprendizaje inhibitorio* [39], una fobia puede ir asociada a múltiples detalles dado un contexto; por ejemplo una persona puede tener una fobia a las arañas cuando está en la casa de un familiar, pero no sentir nada cuando está en otro lugar. El problema que surge a partir de esto es que cuando se realiza la terapia de exposición, es recomendable que el ambiente en el cual se realiza el tratamiento sea lo más dinámico posible, esto con el fin de generalizar el tratamiento a todos los contextos posibles. Dando un caso particular para entender este fenómeno, es posible que una persona que va todas las semanas a tratarse su fobia termine por no sentir miedo alguno cuando esté en el lugar donde realiza terapia, pero en el momento que se encuentra en otro lugar o con otras personas vuelva a sentir la ansiedad y miedo de nuevo.

3.1.1. Múltiples contextos

El primer requerimiento para la solución consiste en la existencia de múltiples contextos externos o internos al interactuar con el objeto u animal que causa la fobia. Esto nos hace entender que diferentes lugares, sonidos o compañía afectan cómo el individuo va a reaccionar a un estímulo particular, en nuestro caso, a su fobia.

Torrents-Rodas et al. explican esta primera técnica para maximizar los efectos de terapias

de exposición [39]. Estos diferentes contextos pueden ser la familiaridad con el lugar o las personas presentes, pero también por ejemplo el día de la semana o cualquier otro estímulo o situación. El problema es cuáles de estos estímulos se pueden manejar dentro del mundo virtual, y por ello se eligió modificar el espacio en el que estará el usuario. Al modificar el terreno del ambiente, se está alterando por ello el estímulo visual y la manera de interactuar con el medio; por ejemplo, cómo se puede llegar de un lado a otro, y qué elementos pueden llegar a ser un obstáculo para el movimiento.

Entonces para abarcar la oportunidad de tener múltiples contextos y así mejorar la calidad de la terapia de exposición, se presentó la capacidad de tener varios mapas o entornos virtuales, los cuales son independientes de las mecánicas del mundo o el tipo de fobia que se requiere. En primer lugar, uno de los mapas es una casa de dos pisos. La razón de elegir una casa es debido a que este es uno de los contextos físicos más comunes para una persona. Lo más apropiado hubiese sido que el diseño de la casa fuese su propia casa, pero como no es posible hacer eso por el costo, se diseña una casa lo más estándar posible, con tres dormitorios, una cocina, un comedor, una sala de estar y un baño. Los colores utilizados para la iluminación general de las salas son principalmente colores cálidos para dar un ambiente más acogedor.

El otro mapa que se diseñó para la actividad principal fue el de un bosque. Esto debido a que se necesitaba un entorno virtual que estuviera más asociado a la naturaleza, ya que muchas fobias entran en la categoría de animales u objetos que se pueden encontrar en este tipo de ambientes. Por la misma razón, el bosque permite trabajar con la idea de que los animales pertenecen a la naturaleza y por ello existe una expectativa mayor de encontrar animales en la naturaleza, en comparación por ejemplo a una casa, donde sería más raro ver una serpiente o algún insecto, por nombrar algunos.

3.1.2. Realismo dinámico

En consideración a lo anterior, es decir, a la necesidad de aumentar la generalidad de la solución y del contexto, es que también se decide que el realismo del modelo virtual que representa a Fobos debe también ser capaz de modificarse según los requerimientos del usuario. Esto se decidió incluir en el diseño en consecuencia a las tasas de abandono de las terapias de exposición mencionadas en el capítulo 1 de introducción. Por lo tanto, lo que se espera al introducir diferentes niveles de realismo del modelo virtual es disminuir y poder configurar la pendiente de carga emocional de un posible paciente, para que pueda ajustar la exposición a sus necesidades.

Por lo anterior, el realismo dinámico de los modelos tendrá tres niveles, bajo, medio y alto. El nivel de fidelidad bajo debe tener ser uno en que el modelo virtual de Fobos en la escena sea de bajo detalle y complejidad gráfica, teniendo los elementos mínimos para ser identificado como el objeto fóbico. Por el contrario, el modelo de alta fidelidad gráfica debe acercarse visualmente lo mayor posible a una representación “real” del modelo, en el mundo virtual. Finalmente, el modelo de realismo medio debe ser un punto intermedio entre ambos.

3.2. Múltiples Modelos

Parte importante de la motivación de este trabajo se generó a partir de la importancia de generar una herramienta o software único que permita abarcar una variedad de fobias sin necesidad de cambiar de aplicación. Si bien existen múltiples tipos de fobias (espaciales, sociales, objetos/animales, etc.), el primer acercamiento a esta solución consistió en dirigir la terapia a todos los tipos de fobias que se puedan representar con un modelo virtual, este grupo son principalmente objetos o animales.

Esto significó que la solución permite agregar modelos virtuales sin necesidad de modificar otros aspectos de la dinámica del software. Por ejemplo, de tener un modelo de serpiente en algún software de modelamiento gráfico (como Blender) es posible trasladarlo e incorporarlo fácilmente a la aplicación de Unity para que se incorpore a la solución y de esta manera, personas con fobia a las serpientes puedan utilizar este software.

Esta solución inicial tiene algunas restricciones. En primer lugar, los objetos o animales no pueden ser de un tamaño excesivo, como por ejemplo oso, elefantes, aviones, etc, esto debido al tamaño y la construcción de los mapas, que no pueden contengan objetos de ese tamaño sin salirse de los límites. Por otro lado, actualmente los modelos virtuales sólo presentan una representación gráfica estática del objeto, pero no tiene animación, sonido o inteligencia artificial, puesto que estas características estaban fuera del alcance del proyecto al agregar muchas variables a la terapia y aumentar los tiempos de desarrollo.

3.3. Dinámica de la actividad principal

La manera en que el usuario interactúa con el mundo virtual y Fobos está definido por la dinámica de la actividad principal. Luego de revisar distintas posibilidades, se consideró que un actividad del estilo ‘La escondida’ tiene ciertas ventajas para las necesidades del problema, ya que elementos de exploración, búsqueda y acercamiento, que aparecen también en terapias de exposición.

La idea de la actividad es que el usuario tenga que buscar a Fobos dentro de un mapa virtual. Fobos es un modelo virtual que se encuentra escondido en el mapa, y para encontrarlo el usuario debe ir revisando diferentes áreas y porciones del mapa donde podría estar escondido lo que busca. Esto obliga al usuario a caminar y explorar el mapa con detalle hasta encontrar a Fobos. Inicialmente al usuario se le presenta una imagen referencial de cómo se ve aquello que va a necesitar buscar, pero lo que no sabe es que realmente el modelo que está en la escena del ambiente virtual es de menor calidad que la de la imagen, generando así una violación de expectativas, y esta es una de las estrategias sugeridas en la investigación de Torrent-Rodas para mejorar los resultados de una terapia de exposición.

Como se describe en el Manual de Diagnóstico y Estadísticas de Trastornos Mentales el componente principal en las terapias de exposición *in vivo* es el acercamiento gradual y frecuente al estímulo fóbico. Esto se ve cuando el paciente debe acercarse de a poco y cada vez más a su fobia, actitud que va contra la respuesta innata de huida y evasión propias del trastorno. Tomando este aspecto básico, es que se reconoce la necesidad de que el usuario ‘enfrente’ a aquello de lo que normalmente huye y por esto en la actividad la persona debe

avanzar y acercarse a Fobos. La actividad termina cuando el usuario agarra al modelo virtual de Fobos.

3.4. Tutorial

Para que el usuario aprenda cuáles son todos los controles y acciones que puede realizar en el mundo virtual, es que se prepara un tutorial que puede ser accedido durante el menú principal de la aplicación. En este tutorial el usuario aparece en un mapa más reducido en comparación al resto y en una pantalla se muestran las instrucciones a seguir. Estas le explican al usuario cómo desplazarse y rotar la cámara, junto con los botones que se requieren para tomar objetos en el mundo virtual.

Para asegurarse de que al usuario se le han mostrado todas las acciones y que ha aprendido a utilizarlas, el tutorial va desbloqueando y enseñando qué acciones puede realizar sólo si es que la acción actual se ha realizado. Por ejemplo, en primer lugar el usuario aprende a desplazarse con los controles, y si no lo hace entonces no aparece en pantalla cómo cambiar la rotación de la cámara.

Según lo anterior, el usuario primero tiene que aprender a moverse, luego rotar la cámara, para después aprender a usar el rayo de interacción y el botón de salida. Una vez realizado eso, debe agarrar una bola de un pedestal, y abriendo una puerta poner la bola en otro pedestal. Si el usuario realiza todas estas acciones, entonces el tutorial lo felicita y puede salir, o repetir el tutorial.

3.5. Motor de videojuego

Existen varias opciones para realizar el software propuesto, desde hacer un motor gráfico y de física utilizando librerías *open source*, hasta utilizar un motor de videojuegos gratis y abierto al público. Como hacer un motor gráfico es una tarea titánica y material para una memoria por sí misma, se decide utilizar un motor ya existente.

Existen principalmente tres opciones para motores gráficos gratis en el mercado: Godot, Unity y Unreal Engine. Godot es un motor relativamente nuevo, open-source y que se especializa principalmente en juegos 2D, y por esto último fue descartado. Tanto Unity y Unreal Engine tienen un excelente soporte en 3D, una comunidad que ayuda al aprendizaje de estos motores y lo más importante ambos tienen soporte para desarrollo en realidad virtual con Oculus. Unity tiende a ser más claro y fácil de usar, mientras que Unreal Engine se especifica en el optimizado y calidad del producto final.

Para la construcción del software se utilizó el motor de videojuegos Unity. Esto debido a que es un programa muy conocido y usado para la creación de diversos tipos de videojuegos, ya que tiene soporte y facilidades para programar las gráficas, el audio, la física, animación, interfaces de usuario y VR. Sobre este último, es importante que exista soporte dado que se considera uno de los aspectos técnicos más difíciles para el proyecto. Unity da soporte VR a varias plataformas, siendo una de estas Oculus. Unity además tiene el paquete “XR Interaction Toolkit” que provee un *framework* útil para transformar interacciones del VR hacia eventos

internos de Unity. El hardware utilizado es el *Oculus Quest 1*, y fue proporcionado por el profesor guía. Gracias a esto, se disminuyeron considerablemente los riesgos respecto a la adquisición y costeo del hardware necesario para el trabajo de memoria. Las ventajas de utilizar el casco de *Oculus Quest* para el desarrollo de esta memoria, fue que tiene integración con Unity, específicamente los *plugins* de *Oculus Integration*, *Oculus XR* y *Virtual Reality Toolkit*, y estos ayudaron con el desarrollo del software y su integración en la comunicación con el hardware. Por otro lado, Oculus proporciona documentación, guías y tutoriales disponibles a los desarrolladores, con secciones para empezar, *frameworks*, *SDKs*, ejemplos, y muchas otras herramientas que facilitaron su desarrollo.

Por otro lado, Unity tiene un sistema de gráficas que proporciona ayuda en *rendering*, cámaras, post-procesamiento, iluminación, etc. Todo esto ayudó a manipular el realismo de la aplicación. Con estas herramientas, en especial con las técnicas de *rendering*, se maneja el sistema dinámico de representación de los modelos. Y las capas de post-procesamiento e iluminación ayudan con la presencia de la escena.

Unity deja la opción de programar en varios lenguajes: C#, C, C++, IronPython, Lua, Javascript, Boo y Rust. Pero la mejor opción para lo que se realizó es C#. Este lenguaje es el que tiene mayor compatibilidad con las librerías de Unity y con soporte de la comunidad, al ser el más usado. Y entonces ese fue el lenguaje de programación que se utilizó para la construcción de la solución.

3.6. Elementos en la aplicación

Más adelante se presentan los elementos y dinámicas principales presentes en la aplicación.

3.6.1. Casa segura

En una de las reuniones con la experta de dominio, se le recomendó a la solución que tuviera una especie de seguro en el caso de que el usuario entrara en pánico y se quedara congelado en posición, sin poder moverse o sacarse el casco. Por esto, el sistema tiene un sistema de seguridad que chequea el movimiento del usuario cada cierto período de tiempo, por ejemplo 30 segundos, el cual de acabarse, teletransportará al usuario a un lugar seguro. Este lugar debe ser relativamente pequeño y bien iluminado, de manera que si el usuario se congela entonces cualquier estímulo visual negativo desaparezca y sea reemplazado por uno inocuo. Dentro de este lugar, el usuario es capaz de quedarse el tiempo que estime conveniente, y luego puede decidir si continuar donde quedó o salir de la actividad.

3.6.2. Gráficos

El estilo gráfico de la aplicación se define como uno de mediano realismo. Esto es debido a que las texturas que se utilizaron representan la realidad sin ningún tipo de alteración artística de por medio, y cualquier tipo de modelo o textura utilizada no está en resolución alta, sino más bien media.

Todos los tipos de recursos gráficos fueron obtenidos mediante la *Unity Asset Store*, que una tienda virtual donde diferentes creadores de contenido digital ofrecen sus productos al

público. Afortunadamente, existe mucho material gratuito que proporciona una base para recrear espacios básicos como casas o bosques sin que la calidad de los recursos genere un decremento en la calidad visual del mundo virtual. Aquí se pueden conseguir árboles, plantas, puertas, mesas, platos, rocas o cualquier otro material que sea necesario para generar un mapa.

3.6.3. Movimiento y física

Unity tiene implementado de manera nativa el módulo de física en cualquier juego o simulación que se quiera recrear. La condición necesaria para usar el motor físico de Unity es agregar componentes que le permitan al objeto interactuar con la física de la aplicación. Estos componentes son principalmente cuerpos rígidos (que permiten tener masa, aceleración, etc.) y una caja de colisión (que permite agregar colisiones realistas entre objetos con cajas de colisión).

Junto a esto, existen ciertas bibliotecas para utilizar cascos de realidad virtual de manera fácil y simple. Estas bibliotecas hacen la conexión con el casco de realidad virtual y le agrega la sincronización casco-modelo virtual (es decir, si el casco se mueve o rota, el objeto asociado también lo hará) a cierto objeto, el cual después puede ser convertido en un usuario. También será necesario agregar *scripts* que permitan la traslación del usuario en el ambiente virtual, y la representación de las manos como la extensión de los controles en el ambiente virtual.

Además de lo anterior, se deberá agregar la gravedad y la colisión del usuario con el ambiente virtual, de manera que el usuario reaccione de manera correcta cuando existan cambios en el nivel del piso y tampoco pueda traspasar paredes.

3.6.4. UI, Láser y botón de salida

A diferencia de otras aplicaciones en 3D, aquellos que están en VR tienen que solucionar un problema respecto a la IU (Interfaz de usuario). En una aplicación normal, la IU está fija y sigue a la cámara, esto lo manejan prácticamente todas las aplicaciones en 2D y 3D. El problema es que en VR no es recomendable hacerlo de esta manera, puesto que al agregar elementos en pantalla que siguen la rotación de la cámara se genera una sensación de incomodidad visual que se asemeja, por dar un ejemplo, a tener los lentes sucios. Para esta incomodidad, comúnmente en juegos en realidad virtual se soluciona al dejar a la IU como un elemento en el juego, normalmente pegado a una pared o a una pantalla como un elemento del mundo.

No hay mayores problemas al adoptar este tipo de solución al prototipo, pero cuando el personaje quiera salir de la actividad y no esté cerca de una pantalla (en el mundo virtual) entonces no podrá salir. Lo usual sería que existiese un menú de pausa, pero esto no se puede hacer directamente puesto que no se debe poner un IU en la cámara del usuario. Por esto, para salir de la actividad se puede crear un botón que esté fijado a la mano virtual del usuario, algo así como un *smartwatch* interactuable. Este botón, al ser presionado, llevará al usuario al menú principal, y para evitar que esté molestando visualmente durante la actividad es que el botón podrá cambiar su visibilidad con un botón de los controles.

Uno de los problemas comunes que tienen las pantallas IU en el mundo virtual es que es

incómodo interactuar con ellas, puesto que a veces están lejos y para presionarlas habría que acercarse y con la mano presionar algún botón. Esto provoca el mismo problema que si los televisores de hoy en día fueran *Touch*, y por ello cada vez que se necesite cambiar de canal habría que levantarse y tocar la pantalla. Para solucionar esto, se utiliza un láser que permite interactuar a distancia, y gracias a esto es más fácil interactuar con la IU y con los elementos del mundo. Lo anterior puede suceder cuando un objeto está a una distancia grande o en un lugar donde es incómodo tener que agacharse para recogerlo.

3.6.5. Tiempo de partida

Inicialmente se esperaba que en la dinámica de la actividad hubiera un cronómetro que midiera el tiempo que le tomaba al usuario encontrar a Fobos en el mapa. Esto tenía el propósito de incentivar al usuario a que estuviera al tanto de su capacidad de búsqueda a lo largo de varias iteraciones de la actividad y así motivar la mejora de sus resultados. También se podría haber utilizado este tiempo para tener un sistema de puntaje posterior a la finalización de la actividad.

Luego de una reunión con la experta de dominio, se le comentó esta decisión de diseño para la actividad, a lo que aconsejó que era mejor descartar esta mecánica debido a que se incorporaba una sensación de urgencia que podría impedir a que el usuario se concentre en sus emociones internas. Junto a esto, se agregaba una variable más que no era estrictamente necesaria, y podría dificultar el análisis del comportamiento de los usuarios.

3.6.6. Estadísticas de la actividad

Pese a la eliminación de la estadística del tiempo de partida, se agregaron otras que tenían mayor coherencia con el propósito de la actividad. La primera es el tiempo de observación directa, la que es un tiempo que aumenta cuando el usuario está mirando de manera directa al modelo virtual de Fobos. Este tiempo puede ser interpretado como un puntaje, y por ende el usuario estará motivado a incrementar este parámetro y así ‘pasar más tiempo’ en presencia de Fobos. El otro parámetro es la distancia mínima al modelo virtual. Esta estadística está diseñada específicamente para aquellas personas que tengan una fobia más grave y por esto no puedan acercarse a Fobos. Entonces a lo largo de la actividad o sus diferentes iteraciones, podrán ir aumentando el puntaje de cercanía en comparación jugadas anteriores. Estas estadísticas, tiempo de observación directa y distancia mínima, podrían servir para motivar al usuario a mejorar este puntaje y además para usarse como un marcador de su progreso.

3.6.7. Estados y transiciones

Para diseñar la actividad se siguió un modelamiento basado en estado-transiciones, que según Hartson et al. [14], es el procedimiento estándar en diseño y desarrollo de software interactivo. Más adelante se presentan la construcción de los estados y las transiciones de la totalidad de la aplicación, y junto con esto se detallan el diseño para el tutorial y la actividad principal.

A continuación se explicitan los estados en los que podrá estar el usuario mientras juega en

el programa, junto a esto se incluyen las transiciones/requerimientos necesarios para cambiar de un estado a otro. Además se incluye la Figura 3.1 para mayor claridad.

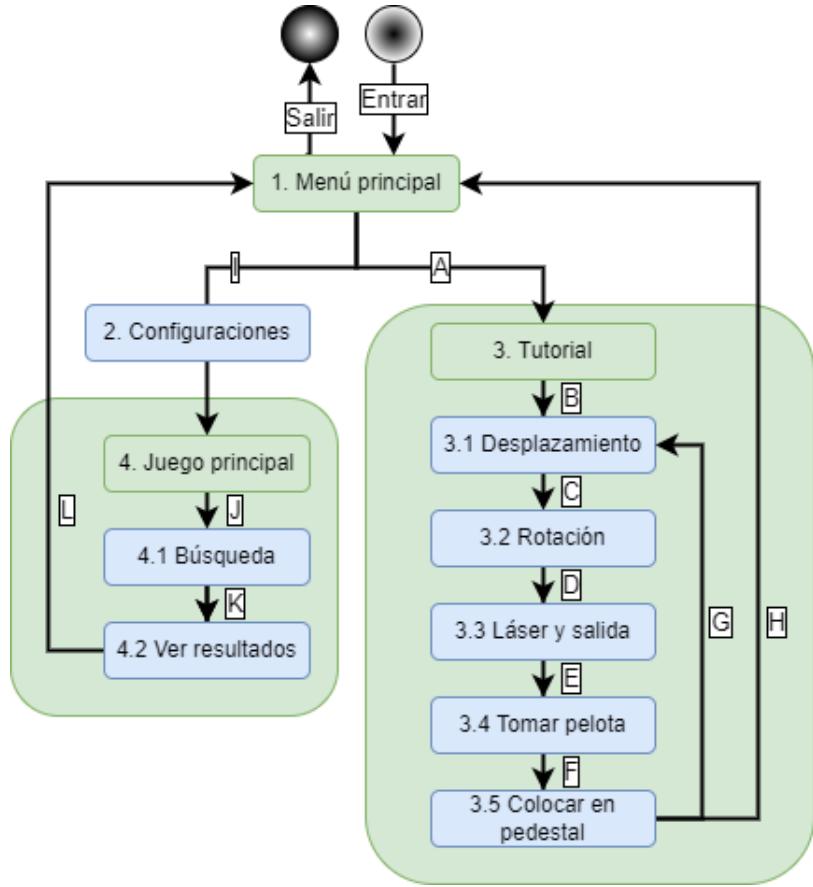


Figura 3.1: Representación gráfica de la máquina de estados.

- Estados

1. Menú principal: Lo primero que se ve al iniciar la aplicación en el dispositivo de realidad virtual. En esta vista, estarán presentes dos botones, uno para iniciar el tutorial y otro para la actividad principal.
2. Configuraciones: En esta parte se elige el grado de autopercepción frente al estímulo fóbico, el tipo de mapa y el tipo de objeto fóbico.
3. Tutorial de uso: El tutorial para aprender a utilizar las mecánicas de la actividad principal.
 1. Interacción para desplazarse.
 2. Interacción para rotar la vista en 45° grados.
 3. Interacción con el láser interactivo y el botón para salir.
 4. Tarea de asir la pelota.
 5. Tarea de llevar la pelota hacia un pedestal, habiendo abierto la puerta en el camino.
4. Actividad principal: La actividad que se utilizará como complemento/soporte a la terapia de fobia.
 1. Realización de la tarea: Búsqueda de Fobos.

2. Observación del puntaje.

- Transiciones
 - a. (1) → (2) : Click en el botón de Jugar.
 - b. (2) → (4) : Editar las configuraciones (1 y 2) y hacer click en “empezar“.
 - c. (1) → (3) : Se hace click en el botón del tutorial.
 - d. (3) → (3.1) : Luego de empezar, automáticamente ocurrirá esta parte.
 - e. (3.1) → (3.2) : Exitosamente se desplazó al usuario.
 - f. (3.2) → (3.3) : Exitosamente se rotó en el plano horizontal la visión del usuario en al menos 45° grados.
 - g. (3.3) → (3.4) : Se clickean ambos botones que hacen aparecer el láser de interacción y el botón de salir.
 - h. (3.4) → (3.5) : Se coge la pelota del pedestal inicial.
 - i. (3.5) → (1.) : Se lleva la pelota al pedestal final tras haber abierto una puerta, y se presiona el botón de salir.
 - j. (3.5) → (3.1) : Se lleva la pelota al pedestal final tras haber abierto una puerta, y se presiona el botón de reiniciar el tutorial.
 - k. (4) → (4.1) : Luego de empezar, se cargará automáticamente esta parte.
 - l. (4.1) → (4.2) : Se inicia la búsqueda de Fobos en el mapa hasta que se encuentra.
 - m. (4.2) → (4.3) : Se realiza la tarea en el tiempo estimado, o terminando el objetivo pedido y se cliquea en salir al menú.

3.6.8. Tutorial

El tutorial tiene como objetivo introducir al usuario al entorno de realidad virtual, y las mecánicas que son necesarias para observar e interactuar con este. Por ello, en esta etapa no están presentes elementos “fóbicos” que puedan interrumpir el aprendizaje del tutorial dada la existencia de elementos que podrían desconcentrar al usuario.

Los puntos específicos a aprender son:

- *Movimiento*: El usuario debe desplazarse usando las palancas del control izquierdo. Basta que camine un poco para desbloquear la siguiente parte.
- *Visión*: Al usuario se le pide rotar la cabeza para ver elementos que no estén en su campo de visión. Usando la palanca del control derecho podrá rotar en 45° grados su visión, al hacerlo puede seguir la siguiente parte.
- *Láser de interacción y botón de salida*: El usuario debe presionar ciertos botones para agregarle al usuario un láser de interacción que le permite interactuar con objetos que estén más lejos y con la IU del mundo virtual. Otro botón tiene como finalidad proveer al usuario de una manera fácil y rápida de salir de la actividad.
- *Interacción con objetos*: El usuario debe tomar una esfera que se encontrará frente a él y llevarlo a un pedestal al otro lado de una muralla. Además de aprender sobre la capacidad de agarre de los controles, debe abrir una puerta que está cerrada para pasar al otro lado.

3.6.9. Actividad principal

La actividad principal consiste en la búsqueda de Fobos dentro de un mapa. El usuario puede caminar y observar el mapa para encontrar a Fobos que se encuentra escondido y es reconocido por su modelo visual que aparece en una de las pantallas.

Existen dos mapas o contextos diferentes, uno pequeño y otro más grande. El pequeño es el interior de una casa, con tres piezas, un baño, una cocina, un comedor y un pasillo. Y el mapa grande es un bosque, con árboles, arbusto y rocas, cuyo tamaño se espera sea 4 veces el tamaño de la casa.

Al inicio de la partida, al usuario se le muestra la forma de Fobos en una imagen del modelo del animal/objeto. Esta imagen hace parecer que Fobos es mayor de lo que es realmente, con el objetivo de amplificar la expectativa del tamaño del objeto, para luego romper esa expectativa una vez que el usuario lo encuentre.

Al final de la partida se recompensa al usuario con tres estrellas dependiendo de sus logros. Una estrella la ganará por agarrar a Fobos en sus manos, otra por observarlo por más de un período de tiempo, y la última estrella por acercarse lo suficiente al modelo (recordando que se puede agarrar a Fobos a distancia con el rayo de interacción).

Capítulo 4

Implementación de la actividad

En este capítulo se discute cómo fue desarrollado el prototipo y sus aspectos técnicos, como la arquitectura, los recursos utilizados, etc. Además se explican los problemas que acontecieron durante esta fase y cuáles fueron las soluciones o desvíos que se debieron tomar para obtener un producto mínimo viable. En este *link* se puede acceder al código de la memoria en *github*, pero por restricciones al tamaño del archivo no se incluyen los recursos visuales importados.

4.1. XR Interaction toolkit

Este paquete¹ provee una base para todas las interacciones necesarias entre el hardware de VR, es decir, el casco montado y los controles, y el software de Unity. La forma en que el proyecto se acopla al *XR Interaction toolkit* es integrando los componentes que provee esta herramienta, en los `GameObjects` del proyecto que tenga relación con el mundo de realidad virtual, por ejemplo: en el personaje, objetos que se puedan interactuar, la interfaz de usuario dentro la aplicación, etc.

Gracias a esto, es posible obtener un *framework* que permite iniciar un desarrollo en VR sin necesidad de generar y mantener la conexión entre el casco y Unity. En consecuencia, esta biblioteca hace los cálculos entre las posiciones locales y globales del VR en sincronía con el mundo virtual, lo que significa que ya no es necesario invertir recursos, como tiempo de desarrollo, en calcular: *¿Dónde está el casco?* *¿Cómo se mueven los controles en relación al casco?* *¿Cómo debería moverse y rotar el casco en casos límites?*, etc.

A continuación se presenta con mayor detalle a las características y ventajas que provee el paquete.

4.1.1. Soporte para diferentes cascos

Esta biblioteca provee varias opciones para el tipo de *hardware* que puede hacer de puente a Unity. Los cascos que tienen el soporte de esta biblioteca son: Meta Quest (todos los

¹<https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.0/manual/index.html>

productos de Oculus Quest, Rift), OpenXR y Windows Mixed Reality. Esto es importante debido a que permitió el desarrollo en el Oculus Quest 1, que de otra manera hubiese requerido buscar otras opciones que mantuvieran la misma calidad.

Además, gracias a esta variedad de hardware permitidos, es que la solución no está delimitada al Oculus Quest 1, sino que fácilmente puede usarse en otros tipos de cascos. Es así que la solución realizada presenta mayor escalabilidad al mercado y al posible uso en cascos de mejor calidad, o bien de precios más accesibles y por ello presentando el apoyo a terapias de fobia a un porcentaje mayor de la población.

Para agregar más tipos de cascos en la compatibilidad del proyecto, basta ir a las configuraciones del proyecto y hacer click en los *checkboxes* para el tipo de casco que se desee agregar soporte. Después de esto, el *build* del proyecto automáticamente genera los componentes necesarios para la compatibilidad en otros casco en el archivo *.apk* de la aplicación.

4.1.2. Interacción con el ambiente virtual

Una de las características principales esperadas por los usuarios es la capacidad de interacción con el ambiente virtual. Este *kit* permite agregar componentes a los controles como a los modelos virtuales para que pueda haber interacción entre ellos. En particular es de interés reconocer cuándo un objeto está siendo tomado por los controles, los que están representados por manos virtuales, y que exista una fuerza que pueda mover estos elementos. En la Figura 4.1 se observa el componente que se debe agregar a las bolas del tutorial que deben ser tomadas y llevadas a otro lugar, y también a las puertas que están en el tutorial y en la casa.

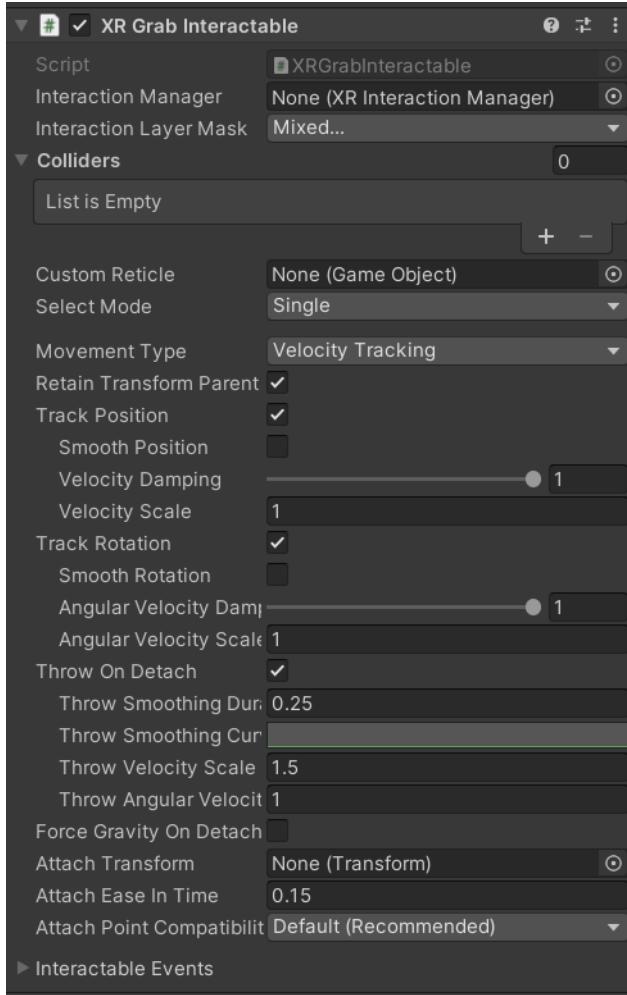


Figura 4.1: Componente Grab Interactable

Este componente permite tener máscaras para decidir quiénes pueden interactuar entre ellos. Por ejemplo, en la variable *Interaction Layer Mask* se define que este *GameObject* es un *Hand Interactable*, que le permite a la aplicación saber que este objeto debe ser tomado sólo por otros *GameObject*s que tengan esta misma máscara, como las manos del usuario.

La variable *Movement Type* define cómo el objeto va a seguir a aquél que lo ha tomado; en particular *Velocity Tracking* aumenta la velocidad del objeto en dirección al que lo está tomando, pero manteniendo las colisiones. Otras opciones no registran las colisiones o el seguimiento es instantáneo, pero para evitar comportamientos no deseados como *clipping* y para poder lanzar la pelota se dejó *Velocity Tracking*. El resto de variables en la Figura 4.1 definen de qué manera el objeto tomado sigue a aquél que lo tomó, y qué hacer cuando termina la interacción de agarre.

Otra capacidad importante que provee el componente *Grab Interactable* es conectar una función al sistema de eventos de Unity, por lo que se puede activar una función específica cuando el objeto en particular es tomado. En la Figura 4.2 siguiente se muestra un caso en donde una función interna se activa cuando el usuario agarra una pelota. Se observa que en el momento en que el usuario selecciona la pelota, es decir, la agarra, se activa un

evento que llama a la función `HasBall` del *script TutorialManager*. Se puede ver además que de ser necesario se podría llamar a múltiples funciones para ese evento, o bien agregar llamados a funciones para otros casos, como en el momento que se suelta el objeto o cuando se agarra/suelta el objeto por primera vez.

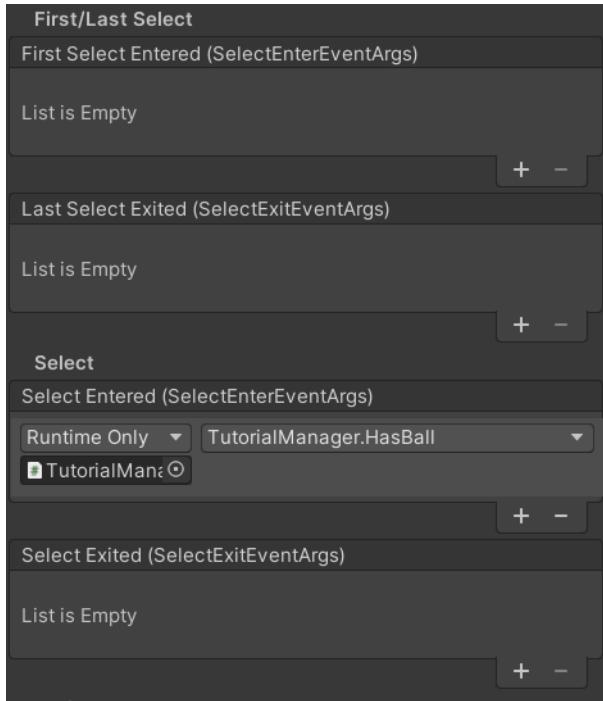


Figura 4.2: Capacidad de asociar eventos

Con relación a las manos del usuario, estas también deben tener dos componentes especiales: **XR Controller** y **XR Direct Interactor**. En caso de no tener estos dos componentes, la biblioteca de *XR Interaction Toolkit* no sabrá que este objeto interactuará de manera especial con otros elementos que sí tienen estas componentes, lo que causa que no muestra funcionalidad alguna.

Como se observa en la Figura 4.3, **XR Controller** es un componente que permite asociar un modelo tridimensional a los controles físicos, siendo agregada en la variable *Model Prefab*. Además, en la sección de *Input*, se pueden asociar diferentes *inputs* del usuario a acciones al usar la aplicación: tales como presionar el botón de agarre, mover las palancas, presionar los botones principales, etc. Más aún, en la asociación del modelo virtual, este sigue el desplazamiento y posición del control físico, sin necesidad de desarrollar tal seguimiento, puesto que lo hace de manera automática en la asociación.

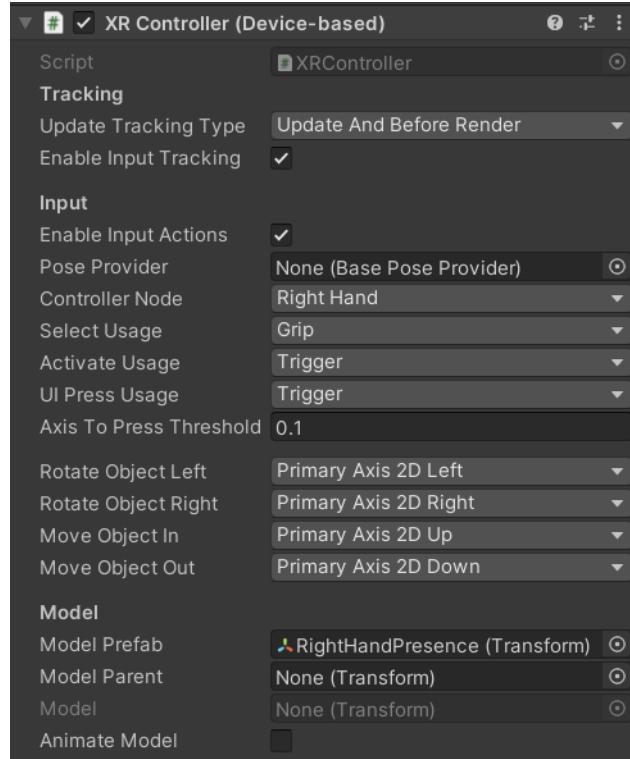


Figura 4.3: Componente XR Controller

En el caso de esta aplicación, se agregaron animaciones para diferentes estados según los *inputs* del usuario. Por ejemplo, cuando el usuario presiona el botón para agarrar un objeto, o apuntar hacia algún lugar, como en la Figura 4.4, la mano se dispone acordemente a la acción realizada.



Figura 4.4: Mano izquierda en posición de agarre y mano derecha en posición de disparo.

Respecto al XR Direct Interactor de la Figura 4.5, este tipo componente es la contraparte de los objetos que se pueden tomar. Es necesario agregarlo al GameObject para inter-

actuar con aquellos que tienen el componente **Grab Interactable**. De la misma manera que el **Grab Interactable**, posee una máscara para acotar las interacciones entre componentes a aquellas en la misma máscara. Nuevamente, existen variables como *Audio Events*, *Haptic Events* o *Interactor Events* que pueden asociar los eventos de interacción (como agarrar o soltar) a función que serán llamadas cuando ocurran estos eventos.

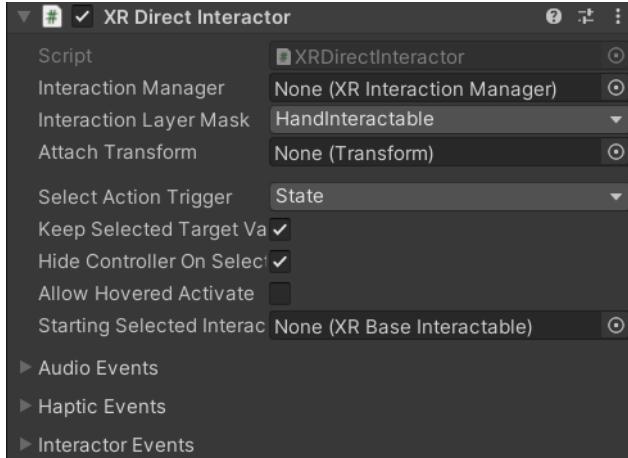


Figura 4.5: Componente XR Direct Interactor

Un ejemplo de esto se muestra en la Figura 4.6, en la que se asocia la función *FobosTomado* del script *GameManager* al evento de tomar un objeto. De esta manera es que se reconoce en la actividad principal, cuándo es que el usuario logra los objetivos.

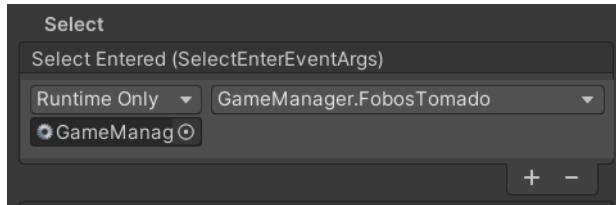


Figura 4.6: Capacidad de asociar eventos para el usuario

4.1.3. Interacción con la interfaz de usuario

XR Interaction Toolkit permite interactuar con la interfaz de usuario en el mundo virtual. Tal como fue dicho en el capítulo 3 de diseño, la forma de interactuar con la interfaz del sistema es diferente, puesto que esta no puede ir asociada al visor de la cámara y debe estar situada en el mundo virtual. Esto provoca un problema, puesto que no es fácil para el sistema saber cuándo el usuario quiere interactuar con alguna interfaz, y es por esto que existe el láser de interacción (entre otras necesidades mencionadas más adelante)

Este láser permite interactuar a distancia con la IU y es el estándar en juegos de realidad virtual. En la Figura 4.7, se muestra cómo es que el usuario interactúa normalmente con la IU. Específicamente, el usuario debe apuntar con sus manos, usando el láser de interacción, a los elementos interactivos de la IU, como los botones, *dropdowns*, *checkboxes*, etc. Una vez

apuntado correctamente, el láser se volverá blanco, y entonces el usuario podrá presionar el botón de disparo para activar presionar dicho elemento.

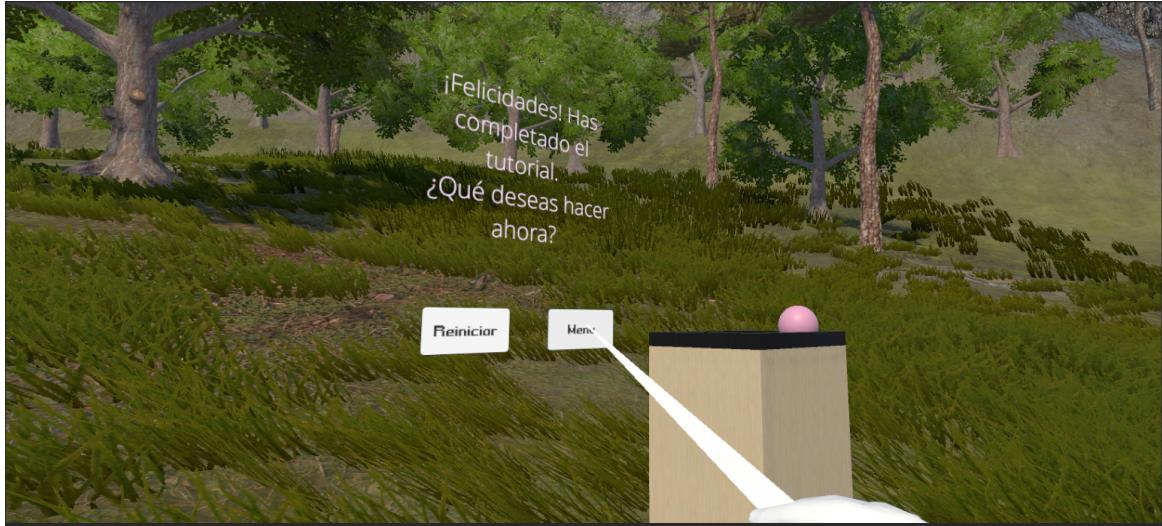


Figura 4.7: Láser presionando un botón de la IU

4.1.4. Seguimiento virtual del casco

El `GameObject` más esencial para la proyección e interacción del usuario en el mundo virtual es el XR Origin. Esta importancia se debe a que es el `GameObject` que se encarga de calcular el seguimiento real del casco y traducirlo al mundo virtual, además se encargar de otras tareas importantes como: (1) ser la cámara principal de la escena y mostrarle al usuario la visión que tiene su personaje, (2) configurar el movimiento de las manos en relación al cuerpo principal y (3) encargar del desplazamiento del personaje en la escena. Esta funcionalidades provienen de los componentes que fueron agregados al `GameObject` del VR Origin. En el Diagrama 4.8 se muestran las características y relaciones de los componentes entre sí, pero para mejorar la legibilidad del diagrama no se muestran los atributos y funciones privadas de *scripts* de paquetes externo, como: Snap Turn Provider, Locomotion System, RigidBody, XROrigin, Character Controller, Transform y Teleportation Provider.

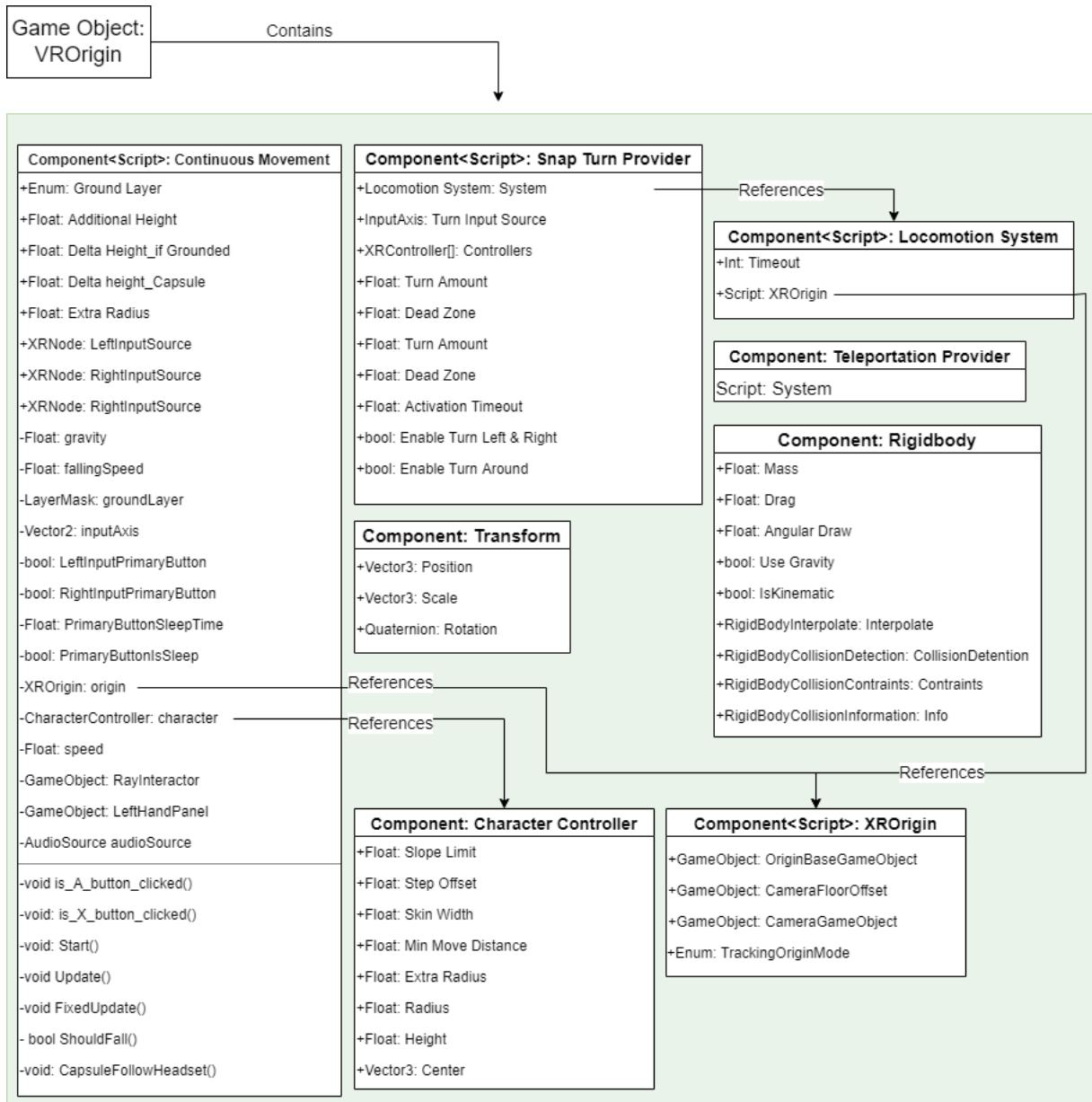


Figura 4.8: Diagrama Componentes agregados a VROrigin

A continuación se esplican los usos de los diversos componentes y por qué estos fueron incluidos.

XR Origin

El componente **XR Origin** hace la conexión directa entre el casco de realidad virtual y Unity, reemplaza la vista 3D de la cámara normal de la escena por una vista especializada y formada específicamente para la estructura de doble lente del hardware que se esté utilizando, puesto que diferentes cascos tienen cada uno su *FOV* (*Field of View*: Campo de Visión) asociado. Junto a esto, el **XR Origin** sigue el movimiento y la rotación del casco del usuario en tiempo real, dando la ilusión de estar en el espacio virtual.

Componentes de movimiento

En primer lugar, **Locomotion System** provee restricciones de movimiento para el **XR Origin**, y en consecuencia el **GameObject** de **VR Origin**. Esto significa que cualquier elemento que implemente la clase abstracta **LocomotionProviders**, como **Teleportation Provider** y **Snap Turn Provider**, puede mover el **XR Origin** cuando **Locomotion System** le provea acceso. Lo anterior tiene como efecto que sólo un **Provider** puede realizar cambios en el personaje en un momento dado.

Snap Turn Provider permite al usuario rotar el ángulo de visión de la cámara por cierta cantidad de ángulos. Sobre las variables que aparecen en el Diagrama 4.8: *Controllers* define qué control puede rotar la visión, y se eligió la mano derecha; las variables booleanas define la libertad de rotación, como si puede rotar hacia la derecha o la izquierda, y si puede dar la vuelta entera. Ambos quedaron verdaderas; *Turn amount* define el ángulo de rotación, y este quedó en 45° grados; *Activation Timeout* define un tiempo de espera antes de otra rotación. A base de pruebas se definió como 0.2 segundos; Y por último, *Turn Input Source* define qué botón del control hace el cambio, y quedó como la palanca, como ocurre generalmente en videojuegos.

Continuous Movement y **Teleportation Provider** son componentes que permiten que el **XR Origin** se mueva por el mapa, pero ambos componentes lo hacen de maneras diferentes.

Continuous Movement permite un movimiento continuo del personaje, esto permite que el usuario se mueva a la velocidad según la intensidad en la que las palancas del control estén siendo operadas. **Teleportation Provider** en cambio, desplaza al usuario según dónde está apuntando el láser de interacción, y haciendo *click* en uno de los botones el usuario aparece ahí. De estos dos métodos, se eligió el primero, desplazamiento continuo, debido a que de esta manera se disminuyen la cantidad de errores o *bugs* que podrían presenciar los usuarios al elegir desplazarse a un lugar no deseado, como fuera del mapa, el techo, sobre muebles, etc. Se decidió mantenerlo como componente del **VR Origin** debido a que en un trabajo futuro se pueda utilizar para disminuir el mareo provocado por la aplicación.

Continuous Movement realiza otras variadas funciones. Una de ellas es definir la altura del personaje, esto lo hace con la función *CapsuleFollowHeadset* que genera una cápsula física que sigue a la cámara del usuario e interactúa por ella con los elementos físicos del mundo, esto permite que la cámara no traspase las paredes, la cámara esté a cierta altura del piso siguiendo la acción de la gravedad. **Continuous Movement** además define la velocidad del personaje y qué botones son asignados a esa funcionalidad. Por otro lado, aprovechando que este componente está siempre presente en el usuario, es que también se le atribuyó la función de manejar el láser de interacción y el botón de salida anexado a la muñeca izquierda del usuario.

Componentes básicos

Los componentes restantes definen funcionalidades básicas de cualquier elemento: *Transform* define la posición, rotación y tamaño de **VR Origin**. *RigidBody* define la masa y la aceleración del objeto al aplicársele una fuerza. Además de definir el roce y la gravedad (en este caso la gravedad se dejó inactivada para que **Continuous Movement** se encargara de ella).

Por último, **Character Controller** trabaja junto a **Continuous Movement** para manejar la colisión del personaje.

4.2. Recursos gráficos

Si bien en un inicio se deseaba modelar en un programa de modelado virtual, como lo es *Blender*², al empezar a trabajar con esta aplicación se observó que la experiencia requerida para modelar objetos virtuales que presentaran una calidad visual apropiada era muy alta. *Blender* requería bastante tiempo invertido en aprendizaje, y esto escapaba del área de trabajo de esta memoria. Es así como para el modelado de objetos, la formación de la arquitectura de la casa y el modelado del terreno se ocuparon otras herramientas o recursos para integrar la parte gráfica de la aplicación sin sobre-invertir tiempo en ellas.

4.2.1. *Unity Asset Store*

Unity Asset Store es una tienda virtual de Unity que funciona como cualquier *e-commerce*; Vendedores ofrecen sus modelos virtuales a cierto precio, y compradores ven las diferentes ofertas del mercado hasta encontrar lo que buscan al precio que necesitan.

Durante la implementación de este trabajo, se hicieron búsquedas de los los modelos o texturas que se necesitaban, por ejemplo: un tostador, sillas, etc. De todos los resultados, se utilizaron aquellos que fueran de carácter gratuito y el estilo visual correspondiera al ya existente en las escenas. De esta manera se pobló la escena con los elementos deseados para el diseño definido.

4.2.2. *ProBuilder*

Para construir la casa virtual en la que el usuario buscaría a Fobos, se utilizó la herramienta integrada de Unity *ProBuilder*. Esta herramienta está diseñada específicamente para hacer estructuras habitacionales como casas y edificios, y está estructurada para que su uso sea simple y fácil de entender.

Este instrumento permite modificar objetos de diferentes maneras. A la izquierda de la Figura 4.9 se observan diferentes funcionalidades que se le pueden aplicar a un objeto creado por *ProBuilder*. En particular para la creación de la casa se usaron los cambios aplicables para planos, por ejemplo: extrusiones, inserciones, escalamientos, rotaciones, pivoteo, etc. Asimismo, los objetos se pueden alterar a partir de sus vértices, aristas o caras, lo que es particularmente útil para las estructuras. Una vez las caras quedan definidas, es posible agregarles texturas externas para agregarle color y realismo al objeto.

²<https://www.blender.org/>

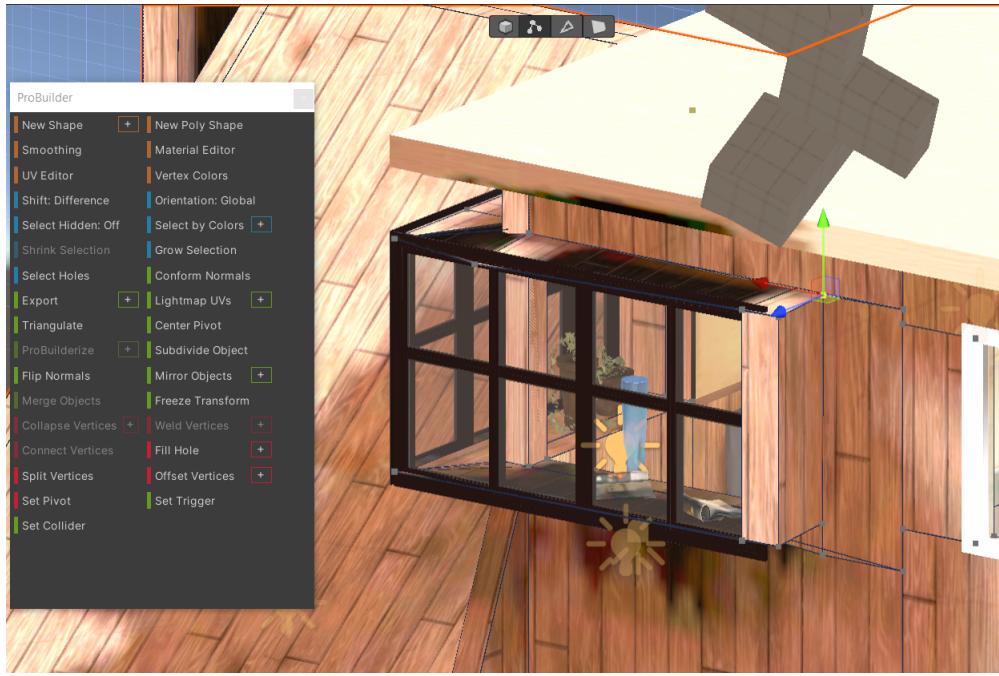


Figura 4.9: Construyendo una casa usando *ProBuilder*

4.2.3. Generación de los terrenos

En particular para el mapa del bosque, la generación del terreno utilizó otra herramienta nativa de Unity: *Terrain Component*. Tal como se observa en la Figura 4.10, este componente permite levantar o disminuir partes de un objeto plano para proveer un terreno de aspecto natural. Esto fue realizado utilizando diferentes pinceles y configuraciones como las que aparecen a la derecha de la imagen, y tales cambios permitieron crear parte de los terrenos del tutorial y el bosque.

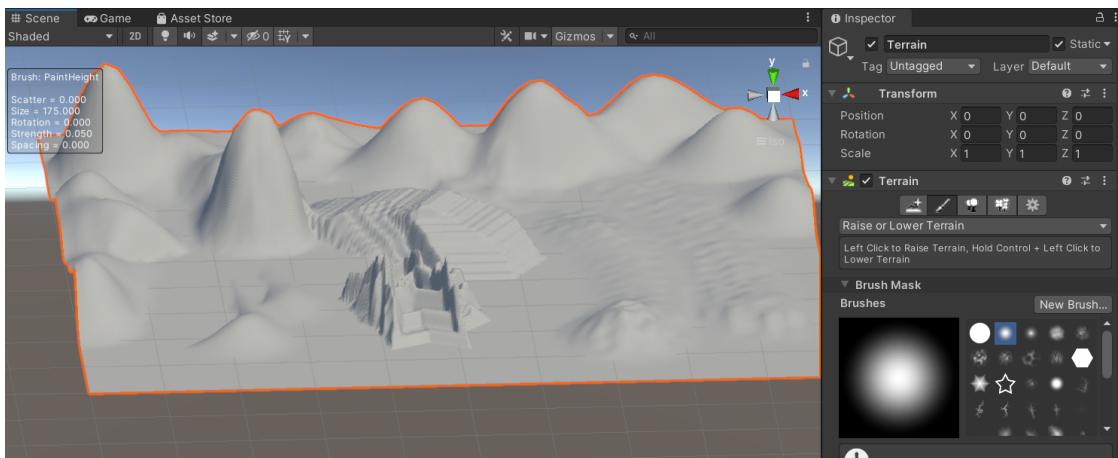


Figura 4.10: Creación de terreno usando la herramienta *Terrain* de Unity.

Junto a esto, el terreno se puede pintar con texturas precargadas, obtenidas gratuitamente de la *Unity Asset Store*, para definir diferentes zonas, como pasto, rocas, tierra, nieve, etc.

Los árboles y arbustos se pueden agregar de manera rápida y simple a la altura del terreno en el sector el cual apunta el puntero del *mouse*.

4.3. Menú principal

El Menú principal es la primera escena a la que accede el usuario cuando abre la aplicación desde el menú de aplicaciones en el Oculus Quest. En este menú, el usuario elige qué acción realizar, ya sea iniciar con el tutorial o bien la actividad principal. De elegir la segunda opción, debe completar una serie de preguntas para configurar su partida a partir de sus necesidades personales.

4.3.1. Espacio

El espacio del menú es simple, puesto que no debería introducir confusión sobre las acciones a realizar. Es por esto que no existen detalles a excepción de la pared central que contiene los elementos de interfaz de usuario. Tal como se ve en la figura 4.11, los elementos espaciales son únicamente el suelo, las paredes concéntricas al espacio, y la muralla con la interfaz.



Figura 4.11: Escena del menú principal.

4.3.2. Elementos de la interfaz de usuario

En la IU inicial del menú, al usuario se le presentan, mediante botones, las dos opciones a proseguir: el tutorial y la actividad principal. De presionar el botón del tutorial, la escena termina y se instancia la del tutorial, junto a su flujo de acción. En cambio, de presionar “Jugar”, se reemplaza la visualización actual de la interfaz por una nueva, la que consiste en la encuesta SUD [29] de auto-percepción estandarizada que permite medir la sensación de miedo que una persona siente en ese momento. La escala va del cero al diez, y va acompañada de un texto descriptivo para facilitar la elección y mejorar la calidad de la respuesta.

Al responder ese cuestionario y presionar siguiente, vendrá una pregunta sobre el animal al que le tiene fobia el usuario; finalmente se pregunta por el mapa que se quiere utilizar, terminando así la sección de las preguntas e instanciando la actividad principal según las configuraciones del usuario.

Toda la información se recolecta en la clase `PlayerPreferences`, a la cual se le van modificando las variables a medida que se van cambiando en las preguntas al usuario. En el Diagrama 4.12 se muestran las diferentes variables y funciones que guardan las respuestas del usuario de manera estática, esto permite que otras escenas en el futuro puedan acceder a estos valores. En particular, `PlayerPreferences` carga un modelo de Fobos en la función `LoadResources`, con la última información del usuario. Este modelo luego se obtiene con `getFobosModel` y se instancia en la actividad principal. Por otro lado, las funciones que contiene `Changed` en su nombre son las encargadas de actualizar la información de Fobos cuando el usuario hace cambios en la interfaz gráfica del tutorial.

Class: PlayerPreferences	
-string:	<code>QualityLevel</code>
-Object:	<code>FobosModel</code>
-float:	<code>anxietyLevel</code>
-string:	<code>gameMap</code>
-string:	<code>type_of_animal</code>
+float:	<code>getAnxietyLevel()</code>
+void:	<code>setAnxietyLevel(float anxietyLevel)</code>
+Object:	<code>getFobosModel()</code>
+string:	<code>getMap()</code>
-void:	<code>Start()</code>
+void:	<code>ScrollBarChanged()</code>
-void:	<code>LoadResources(string type_of_animal, float amount_distress, string map)</code>
+void:	<code>AnxietyToQuality(float amount_of_distress)</code>
+void:	<code>ModelDropdownChanged(Dropdown dropdown)</code>
+void:	<code>MapDropdownChanged(Dropdown dropdown)</code>

Figura 4.12: Diagrama de `PlayerPreferences`

4.3.3. Regulación gráfica

Uno de los objetivos específicos de la memoria es diseñar un sistema de regulación gráfica de Fobos en el mundo virtual. Esto derivó en tres características de la aplicación.

Complejidad gráfica de Fobos

El modelo de Fobos varía según los resultados de la encuesta SUD realizada en el menú principal; a mayor sensación de miedo del usuario, menor será el realismo gráfico de Fobos. En particular, se definen tres modelos por tipo (o especie) de Fobos: de alta calidad, media y baja.

Un aspecto a mejorar sobre estos tres modelos, es que la diferencia visual no es muy profunda. Esto se debe a que los modelos gratis encontrados en la *Unity Asset Store* no eran muy realistas, por lo que en futuras iteraciones se podría aumentar esta brecha entre los modelos, buscando mejores modelos o bien, creándolos en alguna aplicación gráfica. Para ilustrar el punto anterior, se muestran en la Figura 4.13 los modelos de fidelidad gráfica. Pese lo anterior, gracias a la estructuración del código, basta conseguirse un modelo virtual más apto para su uso y simplemente reemplazarlo en la carpeta con el nombre de “<Low/Medium/High>QualityPhobos”, el código lo reconocerá automáticamente y lo utilizará para el programa. Estos modelos se puede obtener en la *Unity Asset Store* o bien haciéndolos en un programa de modelamiento gráfico, tal como fue construido el modelo de baja fidelidad.



Figura 4.13: Modelos de fidelidad para la araña. De izquierda a derecha: *Low, Medium, High*

Cuando el mapa de la actividad principal se inicializa, la aplicación busca las respuestas del usuario en la configuración del menú principal, busca e instancia el modelo virtual correspondiente a la sensación de miedo en la escena del mapa principal, y además al tipo de modelo de Fobos.

Tipo de modelo de Fobos

El tipo de modelo de Fobos es el animal u objeto que genera miedo al usuario, por ejemplo: arañas, serpientes, perros, gatos, etc. Para que el tipo de modelo sea el correcto, el usuario deberá elegir dicha opción en las configuraciones del menú, cuando se elige poner la actividad principal.

Como fue dicho en el Capítulo 3 de diseño, el desarrollador está limitado por el tamaño del mapa, es decir, no debería agregar tipos de modelos cuyas representaciones reales sean de un gran tamaño, sobre todo en el mapa de la casa. Se podría tratar de disminuir el escalamiento de dicho modelo para que quepa en los lugares de escondite, pero a la pérdida de fidelidad relacional (por ejemplo, sería extraño ver un elefante escondido detrás de una puerta, siendo del tamaño de una pelota de tenis).

Similar al proceso de la elección de fidelidad gráfica de Fobos, para el tipo de modelo se toma la elección del usuario. Esto se hace una búsqueda en las carpetas con el nombre del tipo de modelo. Esto finalmente lleva al sistema a instanciar Fobos con una línea de código que se ve de la siguiente manera: Instancear(“Carpeta de recursos”/“Modelos de Fobos”/Tipo de modelo/Complejidad gráfica).

Aspecto de generalidad

Para finalizar de hablar sobre las características de Fobos, cabe destacar que todo el proceso de diseño e implementación fue modelado parcialmente con el fin de evitar que la aplicación fuera dependiente del modelo virtual de Fobos. Por lo tanto, actualmente la solución permite sacar y reemplazar los archivos *.prefab* (que es aquel con toda la información del modelo virtual) por otro, sin necesidad de cambiar la estructura o la dinámica de la actividad para realizar estos cambios.

4.4. Tutorial

Tal como fue dicho en el capítulo de diseño, el tutorial tiene como objetivo introducir al usuario a las mecánicas comunes del mundo virtual, sobre todo para aquellos que no han tenido la experiencia con realidad virtual, y además enseñarles a usar los controles de la aplicación. De esta manera, al iniciar la actividad principal, los usuarios pueden enfocarse con mayor atención a la tarea que se les propone, y así evitar mayores frustraciones por cualquier tipo de mecanismo del controlador.

4.4.1. Terreno y aspecto general

Las características del terreno deben ser simples y deben guiar al usuario a completar las tareas que se le van asignando a medida que este las realiza.

Bajo esa idea, como aparece en la Figura 4.14, el diseño de este mapa significó el desarrollo de un terreno circular y pequeño, que estuviera dividido por una muralla que sólo se pudiera pasar si es que el usuario es capaz de abrir una puerta que pasa para el otro lado.

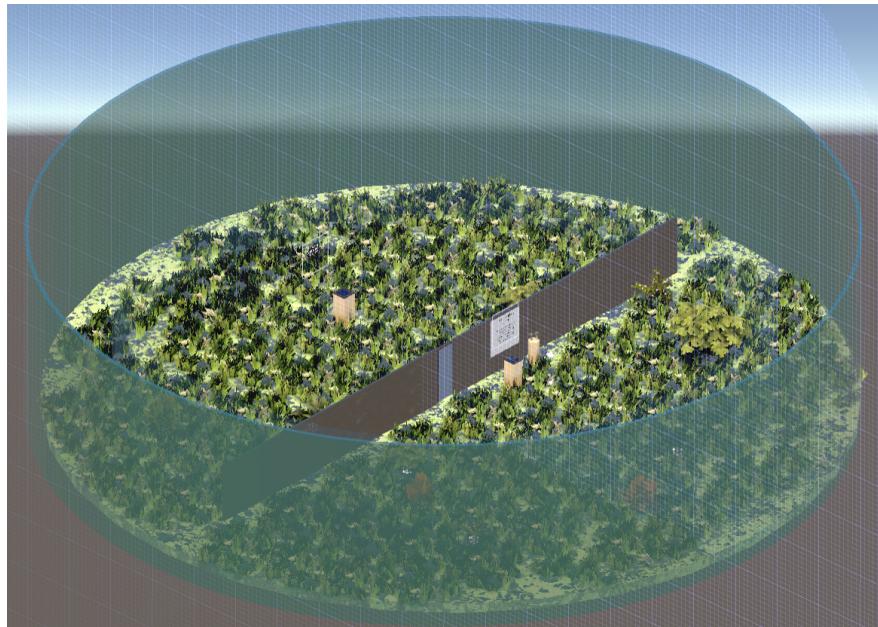


Figura 4.14: Mapa del tutorial

Se agregó, además, un conjunto de plantas como hierbas y arbustos para mejorar el aspecto

del mapa y evitar que el usuario sienta que el espacio virtual en el que se encuentra se ve extraño por la falta de detalles que tiene. Un acercamiento visual, como se ve en la Figura 4.15, muestra el resultado de esta incorporación: pasto en el piso, que evita que el observador vea el piso que plano; arbustos con movimientos de balanceo natural; texturizado con diseño de pasto para mimetizarse con las hierbas que están encima. Esta incorporación también se agregó al mapa del bosque, y resultó parte sustancial del aspecto visual y la sensación de estar en un espacio de la naturaleza.



Figura 4.15: Elementos de naturaleza en el tutorial

4.4.2. Flujo de eventos

Siguiendo el lineamiento del diseño, el personaje debe seguir las instrucciones que aparecen en la pantalla en una de las murallas. Como las instrucciones mantienen relación a la utilización correcta de los controles, en evaluaciones incrementales ejecutadas durante la fase de desarrollo, varias personas requerían sacarse el casco a mitad del tutorial para saber dónde se encontraban los botones que requerían presionar. Para resolver este problema se decidió insertar un modelo virtual de los controles del Oculus Quest 1 dentro de la escena y cerca del lugar de las instrucciones. Además, puesto que el usuario debe resolver las tareas dadas en orden, los botones que debe presionar en el mando real están iluminados y resaltados en los controles virtuales que están de ayuda en la escena, de manera que el usuario puede acudir a ellos para saber a qué botón se refiere el tutorial.

Luego de completar la tarea, el usuario debe aprender otras mecánicas: hacer aparecer el láser de interacción, los botones de salida de la aplicación, y la rotación en 45° grados del campo de visión. Luego de estas tareas, se le pide al usuario una última instrucción, y esta es tomar una de las pelotas de un pedestal y dejarla en el pedestal que está al otro lado del muro. Para completar esta tarea, el usuario, junto con tomar la pelota, debe ser capaz de abrir la puerta que separa ambos espacios, y luego de hacerlo aparece una interfaz que felicita al usuario y le da dos opciones: repetir el tutorial o salir al menú principal.

4.4.3. Problemas con el *build* al casco

Durante la mayoría del tiempo de desarrollo, el mapa lucía diferente. El terreno era más natural y existía un terreno que terminaba bastante más lejos de donde iniciaba el usuario. Junto a esto, la calidad de las texturas era más alta y existían hierbas y pasto de bastante mayor calidad que las que actualmente hay. En la Figura 4.16 se muestra cómo era el terreno en esos momentos: el suelo mostraba un relieve natural, común en espacios de la naturaleza; el pasto era más fino, estaba en mayor cantidad, y se veía mucho más fiel a la realidad; existían árboles en el fondo que se movían como si hubiera viento; el terreno era bastante mayor en espacio, y el fondo estaba bloqueado visualmente con montañas, lo que daba un aspecto visual mejor acabado.



Figura 4.16: Versión del tutorial previo a problemas del *build*

Este fue uno de los mayores contratiempos ocurridos durante el desarrollo de la aplicación, y consistió en la imposibilidad de incorporar la aplicación como un archivo *.apk* en el Oculus Quest, y así dejar una instalación lista que permitiera correr la aplicación de manera independiente al computador que desarrollaba la aplicación. El error específico consistía en que al momento de correr la aplicación en el Oculus Quest, se quedaba cargando en pantalla negra el programa y finalmente el casco volvía a su menú inicial. Cabe recordar que durante todo el período de desarrollo del programa, las pruebas se realizaban con el casco conectado al computador, y el objetivo final es que el casco no tuviera la necesidad de esa conexión.

Se intentó identificar el error utilizando herramientas nativas de Unity para encontrar el problema que estaba causando que el casco no pudiese iniciar la aplicación y terminara con un *crash*. Este problema no se limitaba únicamente al tutorial, y aparecía en ambos mapas, la casa y el bosque. Luego de realizar cambios y a prueba y error, se encontró que la aplicación no podía realizar el *build* si es que ocurrían al menos una de tres situaciones: (1) Existía un terreno nativo de Unity; (2) Se utilizaban ciertos objetos corrompidos en las escenas;

y (3) el archivo `.apk` era muy pesado. Estos errores se identificaron finalmente eliminando aquellos elementos que causaban los problemas, es decir: eliminando el terreno, los `GameObject` corrompidos y disminuyendo el tamaño de las texturas.

Por lo anterior, para el tutorial y el resto de mapas, se eliminó el terreno nativo de Unity, se bajó la calidad de las texturas, y se quitaron ciertos modelos de las escenas con el fin de que la aplicación funcionara correctamente. Las consecuencias que tuvieron estas acciones son que las escenas redujeron su calidad visual en gran medida, y los mapas tuvieron que eliminar sus terrenos y ser reemplazados por otros que consistían en sólo planos y no relieves. La calidad de producto final disminuyó considerablemente en el área gráfica. En un trabajo futuro, en el caso de necesitar mejorar el componente gráfico para comodidad del usuario, se podría generar el terreno en una aplicación gráfica como Blender y reimportar las texturas en su resolución inicial, y observar el comportamiento del *build*.

4.5. Actividad principal

Cuando el usuario ha elegido todas la configuraciones del menú, es ahí cuando es trasladado a una de las dos escenas en donde tiene que encontrar a Fobos. En el lugar de partida, se encuentran las instrucciones de la etapa junto a dos botones: uno para salir del programa y otro para mostrar una imagen de cómo se vería Fobos. Es importante recordar que esta imagen está intencionada para ser una representación virtual de mayor realismo que la que realmente se encuentra escondida en la escena. En la Figura 4.17 se muestra cómo se ve la interfaz del lugar de partida. En futuras iteraciones la imagen de Fobos es preferible que sea reemplazada por otra de mayor realismo, dado que el objetivo es causar un gran impacto de violación de las expectativas del usuario sobre el aspecto de Fobos al momento de encontrarlo. En esta memoria no se pudo realizar este cambio debido al costo de recursos de tiempo en desarrollo y aprendizaje en aplicaciones de modelamiento 3D, y se decidió priorizar los errores de desarrollo que impedían el flujo correcto de acción para el usuario.

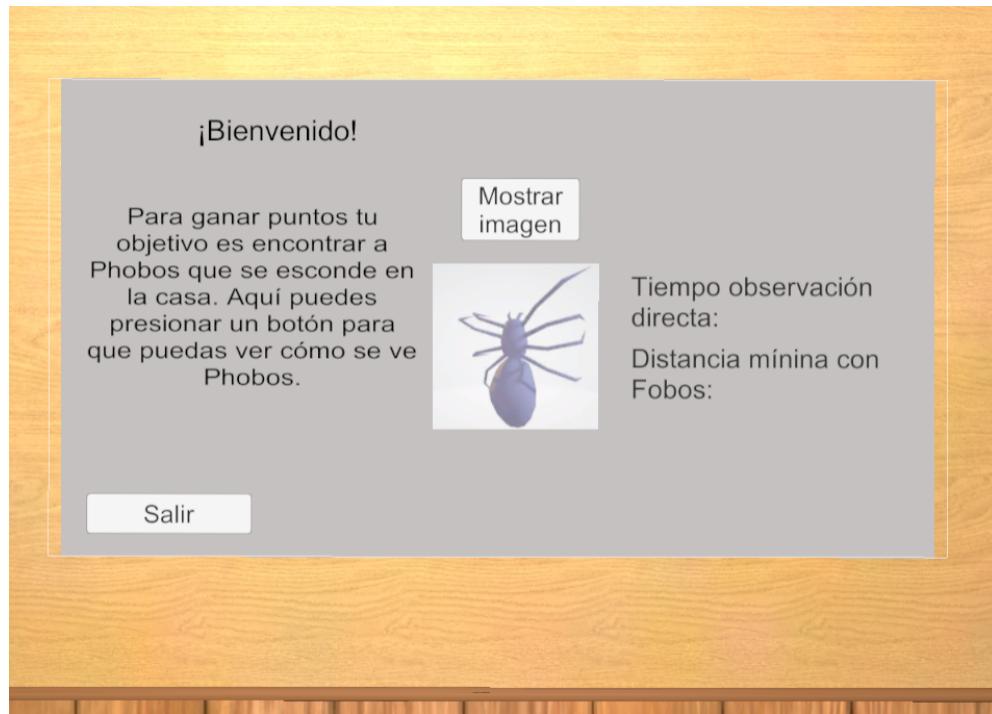


Figura 4.17: Instrucciones iniciales en la IU de la casa

Las variables “Tiempo de observación directa” y “Distancia mínima con Fobos” están para dar una guía al usuario sobre qué tan cercano a estado de Fobos en caso de no encontrarlo. Además sirven para dar un marcador de avance a un usuario que le cueste más energía emocional explorar la casa y le favorezca saber cómo progresá.

4.5.1. GameManager

Existe un **GameObject** en las escenas de casa y bosque que es la encargada de regular las dinámicas del sistema, tales como: condiciones de victoria; carga del modelo correcto de Fobos; cálculo de variables para los logros; teletransporte del usuario a la casa segura; y esconder a Fobos en el mapa.

Este **GameObject** contiene un componente de su mismo nombre que contiene las funcionales mencionadas anteriormente. Si bien se detallan esos puntos en las siguientes subsecciones del documento, en el Diagrama 4.18 se muestran las funciones y variables que se hacen presentes en las escenas de la actividad principal.

Component<Script>: GameManager	
+Text:	TiempoObservacionDirecta
+Text:	DistanciaMinima
+GameObject:	HidePlaces
+GameObject:	Player
-GameObject:	Phobo
-Vector3:	PlayerPosPrevToTele
-Quaternion:	PlayerRotPrevToTele
-float:	timerLeftToMove
-bool:	considerTimer
-float:	maxTimeForStillnes
+GameObject:	safeHouseCenterObject
-float	timeInDirectExposure
-float	minDistanceToPhobo
-void:	Start()
-void:	Update()
-void:	ThrowRayCast()
-void:	TeleportToSafeHouse()
-void:	InsertPhoboinScene()
+void:	ClickedOnContinuar()
+void:	ClickedOnSalir()
+void:	ClickedOnMostrarFobos(RawImage image)
+void:	FobosTomado(SelectEnterEventArgs args)

Figura 4.18: Diagrama UML del componente

En primer lugar, la función *Start* llama a *insertPhoboinScene* para instanciar correctamente a Fobos en la escena y esconderlo. Luego, en el *Update*, se verifican las condiciones para enviar al usuario a una zona segura. Estas condiciones se revisan por cada *frame* del programa, puesto que está en el *Update* que es una función que se llama en cada iteración. Además, en el *Update* se llama también a *ThrowRayCast*, para calcular los si el usuario puede ver a Fobos. Las funciones con el inicio del nombre *ClickedOn* son llamadas cada vez que el usuario interactúa con la IU del ambiente virtual. Y finalmente *FobosTomado* es una función que se llama cuando se activa el evento que ocurre cuando un *GameObject* con el componente *XR Controller* toma un objeto.

4.5.2. Casa segura

La casa segura es un elemento de seguridad para aquellos usuarios que se quedan paralizados por el miedo que le causa Fobos, situación que puede ocurrir cuando el usuario haya encontrado a Fobos como no. Existe un temporizador para el movimiento del usuario, y cuando éste no ha interactuado con el botón de desplazamiento durante un tiempo prudente, por

ejemplo 20 a 30 segundos, entonces el usuario es teletransportado a una habitación pequeña, con el cielo descubierto y bien iluminada. En esta habitación, aparece una interfaz explicando que está en una zona segura y puede permanecer ahí el tiempo que desee, o hasta terminar con la aplicación. En el caso de que el usuario desee continuar con la actividad, entonces puede presionar “continuar” para seguir en exactamente la misma posición y orientación que antes del cambio.

4.5.3. Logros

Cuando el usuario encuentra y toma a Fobos en el mapa, completando así la tarea asignada, se le traslada a otra escena donde aparecen mensajes de felicitaciones y los posibles logros que pudo haber completado. Tal como se observa en la Figura 4.19, estos logros se visualizan a partir de gemas que están flotando sobre unos pedestales, y la interfaz visual explica cuál de estos logros se completaron: cuando el usuario logra tomar a Fobos se gana la estrella de color púrpura; cuando estuvo a menos de un metro de Fobos, se gana la celeste (recordar que se puede tomar a Fobos con el láser de interacción, fallando este logro si se hizo a más de un metro); cuando el usuario se queda en una posición donde puede observar a Fobos durante más de 10 segundos, se gana la estrella naranja.



Figura 4.19: Escena de logros

4.5.4. Iluminación

La iluminación de una escena en Unity puede calcularse de dos maneras: *Tiempo real* o *Cocinada* previamente. A continuación se explicará qué significa cada una.

Si se realiza en tiempo real se tiene la ventaja de que la iluminación se interpreta en cada *tick* del programa, por lo que se modifica según el estado actual de la escena y por lo tanto

existe un mayor nivel de realismo. Por ejemplo, bajo este método, las manos del usuario provocan sombras sobre las superficies que sean bloqueadas de luz por las manos. El problema de usar este método es que necesita calcularse en tiempo real, lo que produce un costo mayor en el casco; en particular para el Oculus Quest 1 puede alterar los *FPS* del programa en mayor cantidad en comparación a otros dispositivos modernos.

Por otro lado, si se “cocina” o precalcula la escena, el gasto de recursos del dispositivo es menor, por lo que las escenas corren sus recursos gráficos y físicos de manera más rápida y la experiencia de la aplicación se vuelve más fluida. Contrario al cálculo en tiempo real, la iluminación queda estática y no cambia según la composición de los objetos en la escena.

La elección para las escenas del ambiente virtual fue la que “cocina” previamente la escena. Esto debido a que es un costo que se puede realizar en etapas de desarrollo, y una vez realizado no es necesario volver a gastarlo. Esto hace el usuario tenga un mejor rendimiento en su programa, y dado que el sombreado en tiempo real no es parte integral de la solución, su pérdida no afecta a la calidad del proyecto en una medida importante.

4.5.5. Escondites

A lo largo de todo el mapa, en el bosque o la casa, se insertaron en lugares donde Fobos podría estar mejor escondido por algún bloqueo del campo visual o es un espacio dónde una persona no miraría normalmente. Estos modelos son normalmente invisibles y no interactúan con los elementos de la escena. En la Figura 4.20 se puede observar cómo están introducidos en el mundo, por ejemplo: escondido entre algunas plantas.



Figura 4.20: Lugar posible de escondite para Fobos

Una vez se inicia el programa, el código elige al azar uno de los lugares donde está una esfera para instancearse. Se elige al azar debido a que si un usuario inicia varias veces la aplicación, entonces, de no ser al azar, sabría exactamente dónde se encuentra Fobos y la experiencia terminaría rápidamente, no dando tiempo para la exploración y la aparición de

emociones relativas a la ansiedad que necesaria para el componente terapéutico de la solución.

4.5.6. Casa

La casa es el primero de los dos mapas, está diseñado para ser cuatro veces más pequeño que el bosque y ser un estilo de mapa más conocido para el usuario. Creado bajo suelos y murallas de manera, junto a una iluminación cálida, este mapa pretende ser un lugar acogedor y hogareño para el usuario.

Estructura/Terreno

La casa contiene dos pisos: en el primero (Figura 4.21) está la cocina, el comedor, la sala de estar y una pasillo que dirige hacia las escaleras; en el segundo piso (Figura 4.22) se encuentran los tres dormitorios y el baño.



Figura 4.21: Vista del primer piso de la casa



Figura 4.22: Vista del segundo piso de la casa

Inicialmente afuera de la casa existía un terreno con árboles y pasto que podían ser vistos por el usuario a través de las ventanas de la casa. Sin embargo, debido a los problemas con la *build* en Oculus Quest, este terreno tuvo que ser eliminado y de momento no existe nada afuera. Reincorporar el terreno exterior puede ser un trabajo futuro si los usuarios se desconcentran al ver espacio vacío por las ventanas de la casa. Junto a esto, podrían existir usuarios de la aplicación que tengan algún tipo de trastorno de ansiedad espacial que lo incomode al ver que la casa no tiene sus bases en el suelo.

4.5.7. Bosque

El bosque es el segundo mapa y se considera el mapa grande de los dos. La intención de este mapa es ser una escena radicalmente diferente a la casa, para que así exista un mayor contexto físico y ser más eficiente a la hora de ser un complemento de terapias de fobia. Si bien el bosque es más grande que la casa, no necesariamente encontrar a Fobos es más difícil, pero sí puede llegar a tomar más tiempo. Es por esto que se aumenta la velocidad máxima del usuario y así las distancias por recorrer son menos molestas. Lamentablemente, por el hecho de que el terreno nativo de Unity genera que la *build* no funcione, eso provoca que de momento este mapa no puede ocuparse en el Oculus Quest sin tener un cable conectado al computador, y con la aplicación de Unity corriendo el programa. Se espera que en futuras iteraciones se genere un terreno, ya sea usando alguna herramienta nativa de Unity o un programa gráfico como Blender, y este mapa quede integrado con la solución final.

Terreno

El mapa tiene forma rectangular y está compuesto de lomas que tienen vegetación y rocas, como se puede observar en la Figura 4.23. Además, el suelo del terreno está pintado a partir de texturas de hierbas y pasto, pero tiene un camino central que está conectado a todos los

lugares posibles en los que puede estar escondido Fobos. Para evitar que el usuario salga del espacio delimitado para moverse, se insertó una barrera semi-invisible, como un panel de vidrio, que impide al usuario cruzar afuera de los límites permitidos. Finalmente, la IU es idéntica a la casa, junto con su comportamiento hacia el usuario.



Figura 4.23: Vista del mapa de bosque

Shader

Los *shaders* son mecanismos por el cual las texturas se someten a operaciones matemáticas para introducir en la imagen cambios o comportamientos visuales que mejoren la experiencia visual para el usuario. Un *shader* gráfico es el puente que conecta una textura con cómo se verá la misma en la escena, es decir, es aquel proceso que se encarga de tomar la matriz de valores que representa visualmente una textura y la opera matricialmente para representarla en la escena. Existen muchos tipos de *shaders* para diferentes funciones y necesidades, pero en esta memoria se utilizó la herramienta *Shader Graph* de Unity para generar movimiento de viento en los elementos vegetales incorporados en los mapas, sobre todo en el del bosque y el tutorial. Esta herramienta permite hacer operaciones matriciales sobre las texturas usando representaciones visuales de las operaciones, facilitando su uso.

El problema y la razón por la cual se utilizó un *shader* para estos elementos es debido a que los mapas se veían sin vida; las hojas y ramas de los árboles, arbustos y hierbas no se movían. Estos modelos estáticos daban la sensación de que la escena estaba parada y no ayudaba al objetivo de hacer estos tipos de mapas, que es conseguir escenas lo más naturales y coherentes posible.

Entonces se desarrolló un *shader*, cuyo objetivo es hacer que los elementos vegetales se muevan como si hubiera viento en la escena. Este proceso se realiza en tres partes: (1) obtención de los puntos del plano y desplazamiento de estos en función del tiempo, (2) agregar *Perlin Noise* para dar un comportamiento aleatorio natural, y (3) modificar los cambios anteriores para que estén en función de la altura, y los elementos cercanos a la base no se muevan. Este proceso fue tomado de Brackeys, un creador de contenido, y fue

modificado para los objetivos de la memoria. La explicación completa puede ser vista en este *link*.

Capítulo 5

Prueba de concepto y análisis de los resultados

Para estudiar el cumplimiento de los objetivos enunciados al inicio de este trabajo de memoria, se desarrolló una prueba de concepto. En este capítulo, se describe el diseño metodológico de la misma. Luego se presentan y discuten los resultados obtenidos, poniendo énfasis en los aspectos de usabilidad, jugabilidad, y presencia en el mundo virtual de la solución desarrollada. Finalmente, se describen los aprendizajes principales obtenidos al cabo de la evaluación.

5.1. Metodología

El objetivo principal de esta prueba de concepto reside en identificar los aspectos positivos y negativos de la aplicación en base a cuestionarios de jugabilidad y presencia en el mundo virtual. A partir de los resultados, se espera evaluar si la aplicación cumple con los objetivos propuestos, y qué aspectos necesitarán cambios en un trabajo futuro. La prueba de concepto se aplicó sobre un grupo de estudiantes de primer año de ingeniería de la Universidad de Chile, utilizando un tamaño de muestra apropiado, conforme a lo que sugiere Holzinger [15] para este tipo de estudio.

Acerca de la metodología, se discutirán: características de los participantes; el equipo utilizado para la prueba; los instrumentos de medición empleados; el procedimiento de la prueba; y finalmente la recolección y la forma de análisis de los datos.

5.1.1. Participantes

Los participantes fueron alumnos y alumnas de la Universidad de Chile cuya edad varía entre 18 a 19 años. Su reclutamiento se realizó mediante un llamado por el foro del curso *CC1000*, en donde se explicaban los objetivos generales de la memoria y que la prueba consistía en usar un programa en realidad virtual. Para participar debían completar un pre-cuestionario el que obtenía información demográfica, su familiaridad con dispositivos VR, dispositivos favoritos para jugar, junto con la frecuencia de juego. Además, el pre-cuestionario

venía con tres *checkboxes* para entregar su consentimiento informado y autorización para utilizar los datos para su análisis. Por otro lado, y tal como fue dicho en el capítulo 1 de introducción, el grupo etario con mayor incidencia a sufrir trastornos de ansiedad se ubican en la adolescencia y la juventud. Es por esta razón que la muestra de usuarios en este estudio, en particular por su edad, es pertinente para alcanzar un mayor porcentaje de la población a lograrse los objetivos de esta memoria.

5.1.2. Equipo

El equipo utilizado en la prueba de usuario consistió en el dispositivo de realidad virtual: Oculus Quest 1 y su cargador, pañuelo y utensilios para desinfectar y limpiar todos los elementos cuando sean usados por una persona.

El Oculus Quest 1 venía con la aplicación del presente trabajo de título instalada, y por lo tanto se podía acceder al software sin necesidad de alguna conexión con un computador y la sincronización con el motor de videojuegos Unity. Lo anterior permitió independizar los resultados a las características del computador en el que se desarrolló la memoria. Por esto, de identificarse problemas durante las pruebas, estos se asociarán al dispositivo Oculus Quest 1 y no a las capacidades técnicas del computador.

5.1.3. Cuestionarios

Existen dos cuestionarios que debían ser completados por los participantes para obtener la información que se esperaba reunir en esta prueba de usuario. El primero es el cuestionario previo a la actividad, el cual podían realizarlo haciendo *click* en un *link* a un *Google Form*. El segundo es aquel que se debe realizar justo después de probar la aplicación. A continuación se describe cada uno de estos cuestionarios.

Cuestionario previo

Las primeras preguntas son demográficas: Nombre, edad, género. Después se hacen preguntas sobre su familiaridad con el ámbito de videojuegos y de realidad virtual. Además de las preguntas se les pidió completar el consentimiento informado, para asegurarse de que los participantes conozcan las características de la prueba, con sus beneficios y posibles riesgos.

Las preguntas se respondieron eligiendo una de las diferentes opciones que cada pregunta daba acceso. En particular las preguntas fueron:

- ¿Con cuánta frecuencia juegas videojuegos?
- ¿Cuáles son las principales plataformas que usas para jugar videojuegos?
- ¿Qué tipo de mando usas comúnmente para jugar?
- ¿Cuál es tu experiencia usando tecnologías de realidad virtual?

Las preguntas de familiaridad con videojuegos y VR tienen el objetivo de predecir qué tan bien los participantes se manejaban con los controles y dinámicas del entorno virtual. Por ejemplo, alguien acostumbrado a jugar videojuegos con un control de consola, entenderá de manera natural que al mover la palanca hacia adelante, el usuario avanzará en esa dirección.

Cuestionario de salida

Este cuestionario se completó una vez la persona terminó con la experiencia del prototipo usando el casco de realidad virtual.

Se inician las preguntas con el nombre de la persona y qué tan mareado se sintió durante la experiencia. Esto último para determinar si la experiencia previa con VR afecta a la sensación de mareo en la simulación. Esto bajo la hipótesis de que un participante con vasta experiencia en realidad virtual estaría acostumbrado a este tipo de entorno, sintiendo menos mareos que un usuario sin experiencia.

Después viene un módulo de 34 preguntas para evaluar la jugabilidad y los elementos funcionales de la aplicación. Estas preguntas vienen en el libro de Calvillo-Gámez et al, titulado *Game User Experience Evaluation*. [5], específicamente en el capítulo *Assessing the Core Elements of the Gaming Experience* (Evaluando los Elementos Esenciales de la Experiencia de Juego, en español). Este instrumento se seleccionó debido a que evalúa la competencia de los elementos esenciales que debe tener un programa para tener una experiencia de usuario positiva, lo que corresponde al primer objetivo específico de la memoria. Este capítulo provee de un conjunto de preguntas esenciales para la evaluación de la experiencia de jugabilidad que provee la solución. Se utiliza una escala de Likert de siete puntos según recomendación de los autores. Además se omitieron algunas preguntas que abarcaban puntos no desarrollados en este prototipo, como el *soundtrack*. La omisión de preguntas no relacionadas a la experiencia de usuario es una acción recomendada por los autores del cuestionario.

A continuación vienen diez preguntas para evaluar la presencia generada por el ambiente virtual y su implicancias en la interacción con el usuario. Se eligió la selección final de preguntas dadas por Makransky et al., en su investigación para el Desarrollo y Validación de una Escala Multimodal de Presencia para Ambientes de Realidad Virtual. [27] Este instrumento fue seleccionado a causa de medir la presencia en ambientes virtuales, y esto, al estar relacionado al aumento de inmersión en videojuegos, según Weibel et al. [40], constituye al tercer objetivo específico de la memoria.

De los tres tipos de preguntas: presencia de uno mismo, presencia física y presencia social, la social fue omitida por no existir ningún tipo de mecánica social en la aplicación. Por recomendación de los autores, se usó una escala de Likert de cinco puntos.

Finalmente se agregaron tres preguntas de texto abierto para obtener cualquier otro tipo de información que el usuario crea relevante mencionar a partir de su experiencia. Para abarcar la mayor cantidad de opiniones se decidió que las preguntas fueran:

- ¿Qué fue lo que más te gustó de la videojuego?
- ¿Qué fue lo que menos te gustó del videojuego?
- ¿Sugieres algún cambio o mejora a la versión actual del videojuego?

5.1.4. Procedimiento

Luego de diez minutos después de la hora de la citación, a los participantes se les hizo una explicación grupal, general y simple sobre qué trataba el experimento y para qué se

usarían los resultados. Luego se recibieron individualmente al experimentador, según quien se ofreciera primero, para empezar la prueba. Aquellas personas que quedaban esperando (conversando, viendo el celular o en el computador) se encontraban sentadas en puestos a lo largo de la sala en donde se realizaba la prueba. Debido a la distribución de la sala y la limitación de los recursos espaciales, no se evitaron sesgos de observador. Se podrían haber enviado a los participantes en espera afuera de la sala, pero esto podría haber causado que se fueran y no realizaran la prueba, dada la incomodidad de quedarse parados afuera.

Con el usuario sentado en una de las sillas, se le preguntaba si había llenado el cuestionario previo a la actividad. En caso de no haberlo hecho, se le permitía llenarlo en el momento. Lo siguiente era preguntarle si tenía fobia a las arañas y se trataba de reconocer algún tipo de miedo anormal durante la explicación de la tarea. De tener fobia o reconocer que el usuario de prueba mostrara signos de miedo, a este se le pedía terminar con la prueba y se hacía pasar a la siguiente persona. Esto debido a que la prueba era para obtener datos sobre las características funcionales y de presencia de la aplicación, y no para probar las capacidades psicoterapéuticas de esta. Por ello, fue mejor no arriesgar al usuario a un posible daño emocional sin tener apoyo de un experto de dominio al lado.

Preguntado lo anterior, se mantuvo una conversación con el sujeto de prueba para explicarle en qué consistía la prueba y cuál iba a ser su objetivo o misión una vez inserto en el mundo virtual. Junto a esto, se le explicaba que era común sentir mareos usando VR y que en cualquier momento podía terminar la prueba sin la obligación de dar razones, esto en concordancia con los protocolos de experimentación con sujetos humanos, de acuerdo a los estándares de la Asociación de Psicología Americana (APA), ampliamente aceptados y utilizados en estudios de usuarios en el área de Computación Centrada en las Personas.

Antes de que el participante se colocara el casco, se le señaló el nombre de los botones de los controles y se apuntó a su ubicación en el control, puesto que estos nombres aparecían en la aplicación: Botones primarios, palancas, botón de disparo y botón de agarre eran aquellos que necesitaban explicación. Se le pidió al participante que al iniciar la aplicación eligiera primero el tutorial, para así aprender cómo controlar al personaje en el mundo virtual. Además se le explicó que al terminar el tutorial, en el menú eligiera el juego principal, con las configuraciones del juego siendo el mapa: la casa, el modelo virtual: la araña, y la sensación de ansiedad: según su propia auto-percepción.

Una vez encontrada la araña en el mapa, o más de quince minutos de tiempo total transcurridos, o bien el usuario deseara terminar el juego por un malestar de mareo, se le pedía completar el cuestionario post-experiencia en una hoja de papel con un lápiz pasta. Y así terminaba su proceso.

Debido a la contingencia higiénica de la pandemia, cada vez que un usuario terminaba con sus tareas asignadas, se iniciaba la limpieza de los controles, el casco y el lápiz usando toallas desinfectantes.

5.1.5. Recolección y análisis de datos

Los resultados del cuestionario post-experiencia fueron transcritos a una planilla Excel y este fue exportado a un archivo *.tsv* para ser trabajado posteriormente en el lenguaje de

programación R. Se realizó una limpieza de datos sobre el archivo *.tsv*, la cual consistió en la remoción de paréntesis y caracteres insertados por el *Google Form* al exportarse a la planilla.

Luego se realizó una inspección visual de los resultados con el objetivo de internalizar el comportamiento de los datos y facilitar el proceso de trabajo de los datos.

5.2. Resultados y análisis

En esta sección se describen y discuten los resultados obtenidos en la prueba de usuarios y entender el comportamiento observado en la prueba de usuarios.

5.2.1. Mareos

A partir de una revisión de la literatura sobre sobre los mareos, Chang et al. [6] explican que los mareos en VR (o cinetosis en VR) son un conjunto de trastornos que ocurren cuando los diferentes sistemas de percepción reciben información no coherente para el cerebro. Un ejemplo común es el mareo en un automóvil: los sensores de aceleración en el sistema vestibular le dicen al cuerpo que no está en movimiento, pero el sistema visual ve cómo se mueven los objetos afuera de la ventana del auto y estas dos informaciones contradictorias se analizan en el cerebro y ocurre el mareo. Los síntomas son náuseas, dolor de cabeza, sudor frío, entre otros.

El mareo en realidad virtual ocurre por el mismo mecanismo que el del auto, en particular cuando el usuario se desplaza de su posición inicial o cambia el ángulo de visión. Junto a esto, el usuario podría experimentar pérdidas del equilibrio, por lo que la prueba se hizo con la persona sentada.

A continuación se muestran las respuestas que cada uno de los participantes dio sobre el valor auto-percibido de mareo durante o después de la prueba, junto con la experiencia previa usando VR:

Tabla 5.1: Tabla mostrando la mediana de mareo autopercibido agregada a partir de experiencia en VR

Me sentí mareado/a durante o después de jugar.	¿Cuál es tu experiencia usando tecnologías de realidad virtual?
4	Sé qué es, pero nunca lo he hecho.
5	He interactuado con una, pero no poseo una.

De la tabla 5.1, la moda y la mediana es 4, lo que representa “Ni de acuerdo ni en desacuerdo” en la escala de Likert. Hubo dos personas que no sintieron nada o casi nada de náuseas; tres que sintieron la molestia de manera moderada; cuatro quienes lo sintieron de manera más grave. Agrupando según la segunda columna, vemos que los que nunca han ocupado un casco tienen como mediana 4, y los que sí, 5. Además, se destaca la baja experiencia de los participantes en estas tecnologías.

A partir de lo anterior, se obtiene que la solución causó molestias por las náuseas sentidas durante o después de su uso. Existen múltiples posibles factores que pueden haber afectado e incrementado esta molestia, y no es posible discriminar cuáles influyeron en mayor cantidad que otros, dado el alcance del proyecto y el protocolo usado en la prueba. De cualquier forma, que cerca de la mitad de los usuarios presenten síntomas de mareos es un elemento que debe ser tomado en consideración para trabajos futuros. El hecho de que los usuarios al utilizar la aplicación se sientan físicamente incómodos es un problema, a razón de que esta podría ser una razón para evitar utilizar el equipamiento. Considerando que una de las causas por la cual se desarrolla esta memoria es para proveer un prototipo que intente solucionar la tasa de abandono en terapias de exposición en realidad virtual, entonces es un problema que los usuarios también abandonen la aplicación por un malestar físico. En relación a lo anterior, en la revisión de literatura de Chang et at. [6] se mencionan varios factores que inducen el mareo y que están presentes en la solución, los que se discuten a continuación.

En primera instancia, está la información conflictiva entre diferentes aparatos sensitivos del cuerpo y éstos van a influir en las náuseas siempre y cuando exista un desplazamiento o movimiento visual en el mundo virtual que no se corresponda a uno en el mundo real. Examinando la columna de experiencia en tecnologías de realidad virtual de los resultados, no se observa efectos sobre el mareo percibido en los usuarios. Si bien no hubo usuarios con experiencia en VR mayor a leves interacciones, aquellas personas que sí lo hicieron no obtuvieron puntajes menores frente a las náuseas (con mediana 5); contrario a lo esperado, aquellas personas sin experiencia tuvieron un mareo menor al grupo anterior, con una mediana igual a 4. A raíz de lo anterior, los datos no muestran que la experiencia en realidad virtual afecte de manera importante la sensación de mareo percibida por los participantes. Esto tiene como consecuencia que, no se debería esperar a que un posible usuario frecuente de la aplicación sienta menores mareos gracias a su uso constante.

Otro factor que influye en este problema es el realismo virtual de la escena que el usuario observa. Como mencionan Chang et al. a mayor calidad visual del mundo virtual, mayor es la probabilidad de que existan incoherencias virtuales y reales. En consecuencia de lo anterior, el hecho de que la aplicación esté enfocada en aumentar el realismo visual, es posible que este haya aumentado el mareo general en los usuarios, y por ello, no se debería descartar la posibilidad de que para disminuir el nivel de mareo en la aplicación, se deba probar otros estilos visual menos realistas.

¿Qué fue lo que menos te gustó del videojuego?

'El movimiento que hace la cámara al subir y bajar las escaleras y el movimiento estático de la cámara al mover la palanca derecha (como de 60° a 60°) [aquí el participante dibuja un símbolo de un hexágono]'

Cita 1: Participante refiriéndose al cambio brusco del campo de visión

Respecto a los comentarios sobre la aplicación, seis personas destacaron de manera negativa el cambio brusco de 45° grados en la rotación del campo de visión, como se evidencia en la cita 1. Pese a que durante la implementación de la solución, esta forma de rotar la visión sirvió para disminuir las náuseas del desarrollador (en comparación a la rotación continua), esto terminó siendo evaluado negativamente por los usuarios y provocó molestias visuales que

posiblemente se tradujeron en una mayor sensación de mareo.

Similarmente, uno de los participantes comentó una vibración rápida y molesta al subir las escaleras, como se evidencia en la cita 1. Posterior al experimento se identificó tal evento como un error en el desarrollo del sistema físico de la aplicación al subir las escaleras. Esta vibración rápida podría no haber sido comentada por los otros usuarios debido a la posibilidad de que éstos hayan pensado que ese movimiento simulaba la subida de las escaleras. Al igual que con el cambio brusco del cambio de visión, esta vibración visual podría ser también uno de los factores que incrementó la sensación de malestar en los usuarios.

Tanto las molestias por el cambio brusco del campo de visión, y la vibración rápida y molesta al subir las escaleras generaron molestias lo suficientemente importantes para que algunos participantes lo hicieran notar en las respuestas abiertas de la encuesta. Como fue anteriormente, esto implica que los posibles futuros usuarios de esta aplicación eviten su uso debido al malestar físico generado, contrariando el objeto general de la memoria. Es por esto, que en futuro trabajos sobre esta aplicación, este problema debe ser intervenido con alta prioridad.

5.2.2. Jugabilidad

En las figuras 5.1 y 5.2 se pueden observar los resultados de la prueba de usuarios para las respuestas de la temática de jugabilidad usando la escala de Likert de siete puntos. Se utilizó el método de visualización *Violin Plot* para ilustrar de mejor manera el comportamiento de las respuestas por pregunta, ya que se puede observar la dispersión y las preferencias de la muestra en cada pregunta. Además, en el anexo se encuentra la figura 7.1 que contiene la mediana por pregunta y con la agrupación designada por los creadores de este cuestionario [5]. Pero antes se explican los significados de algunos puntos de la columna de agrupación de la tabla, puesto que su significado es más difícil de interpretar.

- *Puppetry*: La interacción del usuario con el juego. Esta se forma por la sensación de control y propiedad sobre el juego o el usuario; esto implica que las acciones del personaje en el juego son internalizadas por el usuario como propias.
- *CEGE*: *Core Elements of Gaming Experience* (o Elementos Esenciales de la Experiencia de Juego, en español) consiste en las condiciones necesarias pero no suficientes para proveer una experiencia positiva mientras se juega videojuegos. Engloba principalmente el área de *Puppetry* y videojuego, junto con la conexión entre ambas partes.
- Videojuego: Consiste en la experiencia intrínseca del juego. Son un conjunto de condiciones o características que proporcionan un ambiente idóneo para incorporación del usuario en el diseño del mundo; sonido, gráficas, reglas del juego, escenario, mecánicas de juego, etc.

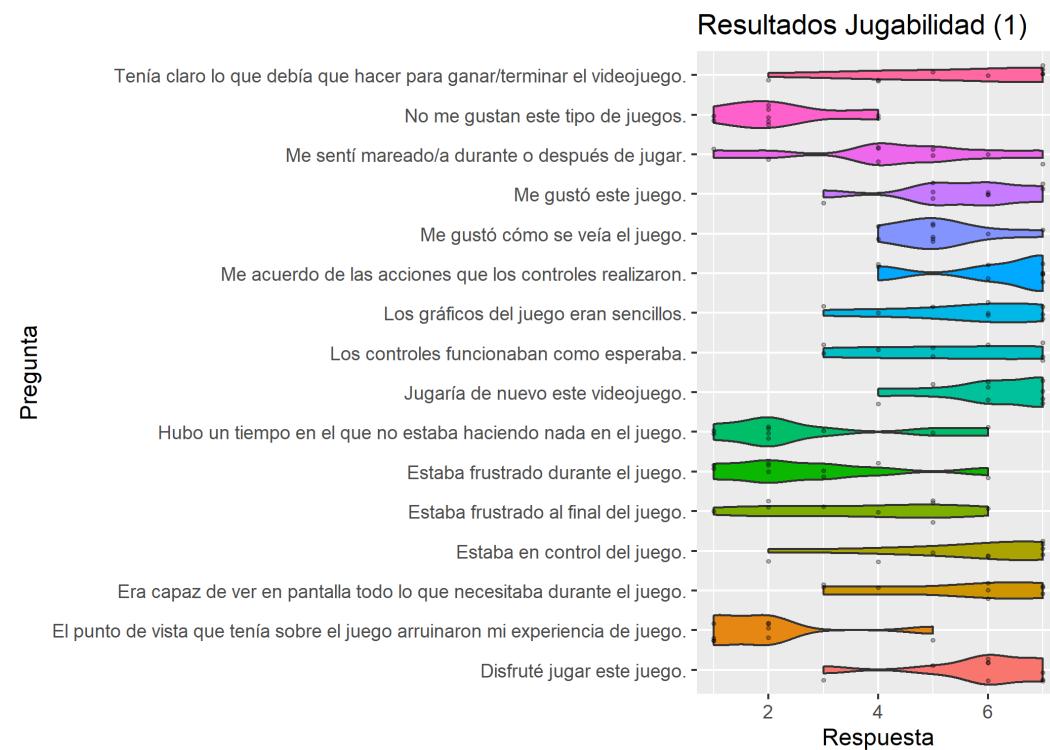


Figura 5.1: Resultados cuestionario jugabilidad (1)

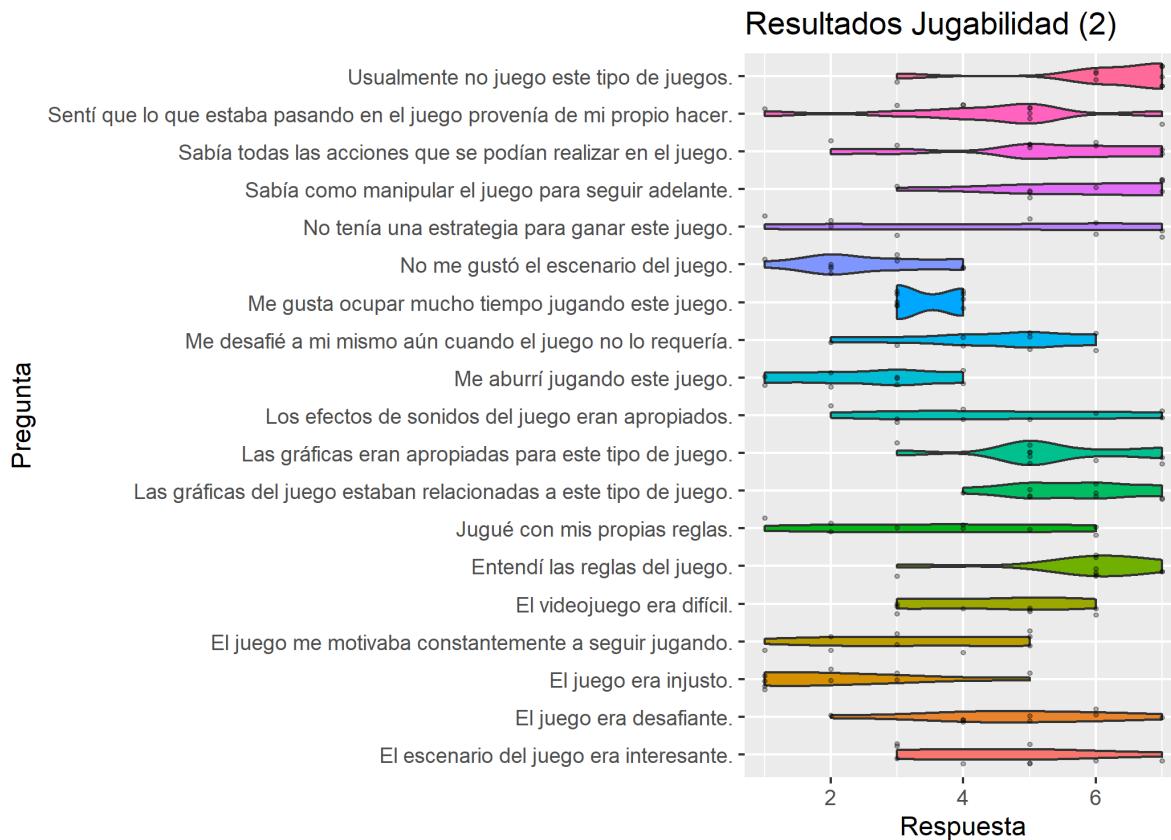


Figura 5.2: Resultados cuestionario jugabilidad (2)

En la figura 5.1 y en la tabla 7.1, podemos observar aceptación y disfrute en la experiencia de jugabilidad, puesto que preguntas como “Disfruté jugar este juego”, “Me gustó este juego”, “Jugaría de nuevo este videojuego” fueron valoradas con altos valores en la escala; todas siendo 6, lo que es “En acuerdo”. Los resultados anteriores se traducen en un grado alto de aceptabilidad de la aplicación para los participantes. Ellos disfrutaron la experiencia y les gustaría repetirla, ya sea debido a que: sienten que lo pueden hacer mejor una siguiente vez; les hubiese gustado observar mejor el ambiente virtual; o simplemente les gustó la experiencia y dinámicas experimentadas en la aplicación. Además, se destaca que la aplicación podría atraer a aquellas personas con fobias a interactuar de nuevo el programa, como se evidencia en las respuestas de los participantes sobre si jugarían de nuevo el juego.

Se observa un aumento en la frustración percibida a lo largo de la experiencia. Puesto que la frustración durante el juego marca un puntaje de 2, y al final un 4. Esto parece relacionarse a la dificultad de la actividad, a causa de que se le consideró ligeramente desafiante y difícil según los usuarios; pese a esto, en la figura 5.2 la mediana respecto a la pregunta “El juego era injusto” es 2, lo que junto a las otras respuestas, implica que la dificultad de la actividad no se debe a un desbalance de las reglas del mismo. Los resultados anteriores tienen como consecuencia que, los participantes consideran que la actividad es justa, y por ende sus logros y fracasos se deben a su propio desempeño dentro del videojuego. Si bien existe un aumento de la mediana de la frustración a lo largo de la prueba de usuario, si se revisan los resultados por persona, se observa que algunos disminuyen su frustración al final del juego, lo que podría deberse a que completaron el juego exitosamente. Por esta razón, es que se cree que cuando un usuario logra agarrar a Fobos y completar la experiencia, se disminuye la sensación de frustración. En ese sentido, al cuestionario le faltó componer esa pregunta para comprobar si realmente existe una relación entre la completitud de la tarea y la diferencia de frustración a través del proceso.

Existe coherencia respecto a la experiencia previa de los usuarios con tecnologías de realidad virtual, considerando que la pregunta del pre-cuestionario “¿Cuál es tu experiencia usando tecnologías de realidad virtual?” mostró baja experiencia, y la pregunta del post-cuestionario “Usualmente no juego este tipo de juegos” resultó en una mediana 7 para los usuarios. Es por este motivo que cree que esta coherencia en los resultados implica que los participantes respondieron honestamente y manteniendo la concordancia entre respuestas que apunta a temas similares. Por ejemplo, si un participante dijera que no tiene experiencia jugando juegos en realidad virtual, pero al mismo tiempo respondiera que juega usualmente ese tipo de juegos, entonces se podría pensar que ese tipo de participante no está siendo coherente con sus respuestas.

Si bien las respuestas a las preguntas favorecen la implementación de la solución, a partir de la figura 5.2, en la pregunta “El juego me motivaba constantemente a seguir jugando” se obtuvo una mediana igual a 3, lo que se traduce a “Ligeramente en desacuerdo”. Este resultado ejemplifica una falla de diseño para la dinámica de la actividad, en consecuencia a la monotonía que se siente al buscar a Fobos en el mapa. Esta monotonía se podría haber evitado insertando algún tipo de mecánica que agregue dinamismo a la búsqueda, por ejemplo; Estrellas o puntaje escondidos en rincones que premien la exploración del usuario, o agregar sonidos que den pistas sobre el paradero de Fobos. En el peor de los casos, si es que el usuario no puede encontrar a Fobos, este debe sentir que la búsqueda en sí misma valió la

pena, puesto que el objetivo es que el usuario afrente su miedo y busque a Fobos.

En relación a las preguntas de la agrupación de Disfrute, se esperaba hacer una comparación entre usuarios con experiencia o sin; pero todos los participantes tenían una muy baja experiencia con estos tipos de tecnologías. Pese a no tener esta comparación, hay que tener cuidado con el sesgo que podrían tener los usuarios al ser nuevos experimentando VR, en la medida en que el disfrute de la aplicación se puede deber a la novedad y primeros usos de la aplicación. Un usuario más experimentado podría haber analizado con mayor detenimiento las características de la solución y, haciendo una comparación con otras experiencias, haber sido más riguroso con su puntaje.

Hubo un comentario dentro de la categoría *GECE + Puppetry* y trataba sobre la interacción del usuario con las puertas. Se mencionó que las puertas eran complicadas de abrir e interactuar; Si bien fue sólo una persona que comentó esto durante la prueba, durante el período de desarrollo, varios *testers* de etapas iniciales tuvieron problemas con las puertas. Mucho de estos problemas iniciales eran errores técnicos, pero luego de arreglarlos todavía quedaba una incomodidad con la mecánica de las puertas. Se cree que el problema principal de estas ocurre debido a que abrir una puerta es una tarea de coordinación compleja, en el sentido de que no sólo requiere el movimiento del brazo para desplazar la puerta, si no que además requiere un desplazamiento del cuerpo principal en dirección a la puerta, permitiendo abrir la puerta en todo su rango. El problema de esta interacción en VR, y sobre todo en el período de pruebas, es que el usuario está sentado y por lo tanto no tiene toda la capacidad y el rango para abrir una puerta de manera normal y fácil, necesitando mayor experiencia e intentos.

5.2.3. Presencia

A continuación se muestran los resultados de presencia del cuestionario.

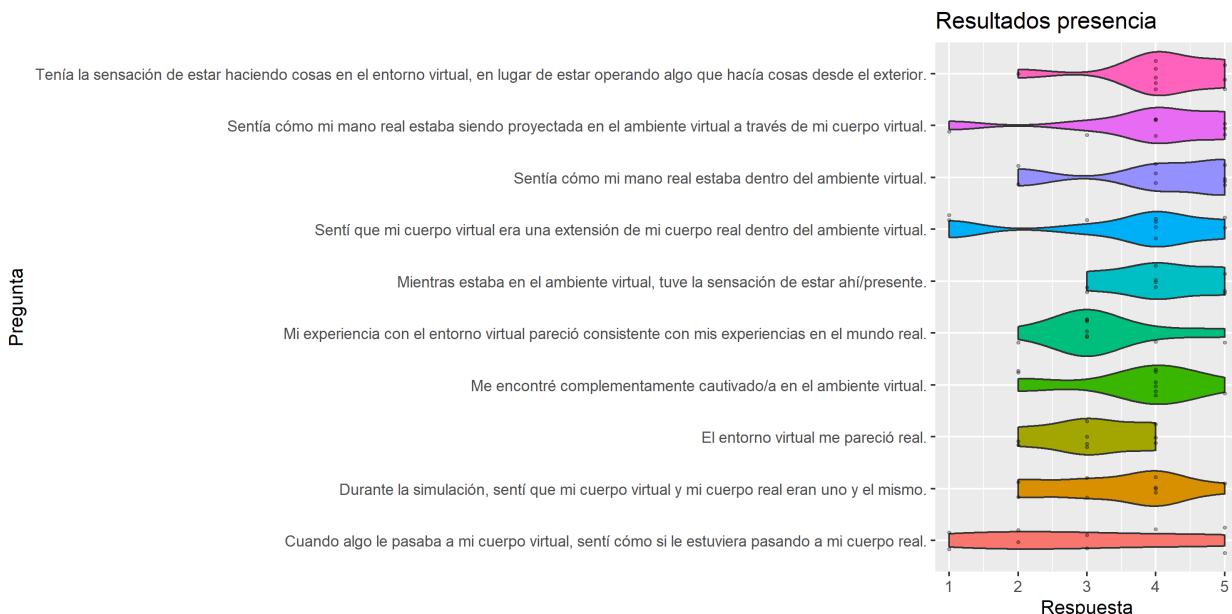


Figura 5.3: Resultados cuestionario presencia

A partir de la figura 5.3, podemos observar bajo las cinco primeras preguntas, que la sensación de extensión del cuerpo real al cuerpo virtual, y el tipo de respuesta esperada del usuario sobre sus acciones realizadas concuerdan con lo que esperan que ocurra, dado que la mayoría está "De acuerdo" con estas afirmaciones. Estas preguntas apuntan a entender cómo se siente el usuario frente a las interacciones dentro del mundo virtual, independiente de cómo se vea y qué gráficos tenga.

A partir de lo anterior, la aplicación logra traducir y proyectar las intenciones de la persona a cargo. Esto se puede ver en la aplicación en momento en los que la persona mueve las manos reales, y las manos virtuales se mueven con la misma velocidad, posición y rotación que su contraparte real. Además, si un usuario pensaba que podía agarrar algo, y lograba obtenerlo, eso generaba consistencia entre el aparato visual y el modelo virtual. Esto último implicó que la segunda y tercera pregunta hayan obtenido puntajes con mediana cuatro, que significa que están "De acuerdo" con esos puntos.

La sexta pregunta obtuvo resultado más neutrales, siendo "Ni de acuerdo ni en desacuerdo" el valor de la mediana. Los usuarios no lograron sentir consistente las experiencias entre el mundo virtual y el real. Una causa de esto podría ser la falta de elementos interactivos en la escena, ya que los usuarios intentaron tomar elementos como libros, platos, ventanas, u otros elementos, y no ocurría nada cuando intentaban interactuar moviendo los objetos. Esto causa un quiebre a la reglas del mundo virtual que recuerda al usuario que está en uno.

Por otro lado, los usuarios se encontraron cautivados por el ambiente virtual, siendo la mediana "De acuerdo". Esto se puede deber a que la mayoría de las personas tenían baja o nula experiencia utilizando dispositivos de realidad virtual, y por lo tanto su primer encuentro tiende a ser más emocionante por esa razón.

Pese a que se encontraban cautivados por el ambiente virtual, no lo encontraron particularmente real, puesto que en la pregunta ocho los usuarios se encontraron "Ni de acuerdo ni en desacuerdo" con respecto a este punto. Esto se cree se debe a que la aplicación no estaba hecha para parecer real y "confundirse" con el mundo real, puesto que las gráficas utilizadas fueron principalmente de baja o media resolución para mantener la resolución y el refresco de la pantalla dentro de la comodidad del usuario.

Finalmente, la última pregunta destaca por lo disperso de los resultados, pese a que la mediana se mantiene en "Ni de acuerdo ni en desacuerdo". Esta dispersión se cree que ocurrió debido a la manera de interpretar la pregunta para las diferentes personas, puesto que esta pregunta está hecha para una aplicación con mayores dinámicas para el modelo virtual del usuario, en donde tal vez sea posible que el ambiente virtual o alguno de sus entidades interactúe y aplique alguna acción sobre el usuario. En el caso de este prototipo, no existía tal mecanismo por lo que los usuarios podrían estar confundidos de a qué se refiere esta preguntar en el contexto de la prueba.

La tabla 5.4 muestra la mediana agrupada por las categorías definidas por Makransky et al [27].

Figura 5.4: Tabla de mediana por categoría

Agrupacion	Mediana
No consciente de la mediacion espacial	4
No estando atento al ambiente real	4
Realismo fisico	3
Sensacion de estar en el mundo	4
Sensacion de extension del cuerpo	4
Sensacion de la conectividad del cuerpo	3

De la tabla 5.4, se observa homogeneidad en las medianas por categoría, siendo el valor 3 y 4 las que se repiten, representando a “Ni de acuerdo ni en desacuerdo” y “En acuerdo” respectivamente. Estos resultados muestran que los aspectos de presencia se pueden mejorar, en particular en aquellas agrupaciones cuyas medianas dieron 3, tales como “Realismo Físico” y “Sensación de la conectividad del cuerpo”. Por lo anterior, el prototipo requiere mejorar los aspectos de realismo físico y sensación de conectividad del cuerpo presente en el ambiente virtual de la aplicación. Las causas para estos resultados también se pueden encontrar en las respuestas abiertas que completaron los participantes. A continuación, en la cita 2, se detallan mejor los problemas con el realismo físico observado por varios usuarios.

¿Qué fue lo que menos te gustó del videojuego?

‘La poca interacción con objetos. Como por ejemplo gabinetes, cajones, no poder agarrarlos para ver más de cerca.’

Cita 2: Participante refiriéndose a la falta de interacción con objetos

Con este comentario, es posible discutir la necesidad de agregar mayor interacción con los diferentes objetos presentes en la escena. Se espera que al mejorar este aspecto del proyecto, el mundo virtual sea más consistente con el comportamiento esperado por el usuario, y esto cause un aumento en el realismo físico percibido en la aplicación.

Durante esta prueba de usuarios es posible que los resultados, en particular los de presencia hayan sido afectados por un sesgo causado por los otros participantes. Esto a causa de que la prueba se tomó en una sala de clases donde los participantes no estaban separados de manera física ni acústicamente de aquellos que estaban a la espera de su turno. Esto implicó que existieran ruidos por las conversaciones y sonidos generados por estos últimos, lo que puede haber alterado la sensación de estar en un mundo virtual. Se podría hipotetizar que los sonidos externos le permitían a los usuarios recordar que había una realidad al otro lado del mundo virtual, y por ende su mente disminuía el foco y la sensación de estar ahí de los participantes. Esta forma de sesgo se podría haber evitado utilizando dos salas diferentes, una para esperar el turno para la prueba, y otro cuando sea momento de realizarla. Respecto a los resultados, se cree que este sesgo podría haber disminuido el puntaje en las preguntas de presencia, y por lo tanto haber afectado negativamente al cómo se integra un usuario al mundo virtual.

En consideración a la agrupación de “Realismo físico”, que contiene las preguntas con unos

de los valores más bajos, se deducen algunas causas para estos resultados. Partiendo por los comentarios abiertos, leyendo la cita 2 o la 3, escrita por los participantes, fue recurrente la recomendación de hacer más objetos del ambiente virtual interactuables, es decir, que se pudieran tomar y mover con la mano. Cabe recordar que las puertas eran objetos que tenían esta condición, y por lo tanto un usuario esperaría que otros objetos en el mundo virtual tuvieran esta capacidad, al generalizar la capacidad de interacción hacia más objetos.

¿Sugieres algún cambio o mejora a la versión actual del videojuego?

'Añadir más objetos para interactuar'

Cita 3: Participante refiriéndose a la falta de interacción con objetos

5.3. Piloto de prueba para el bosque

Durante las etapas finales del desarrollo, en particular al realizar el *build* de la aplicación, se encontró un problema que consiste en el quiebre de la aplicación cuando está exportada en el casco. Luego de un proceso de investigación se encontró que el problema se relaciona a la existencia de *Unity Terrain* al ocupar dispositivos de realidad virtual. Si bien existen múltiples razones por la cual este problema ocurre, según diversos foros en línea¹², al parecer existe un problema recurrente con la compresión de las texturas al insertarse en el terreno.

Si bien en el tutorial y en el mapa de la casa se corrigió este error al eliminar el terreno y reemplazarlo por una versión más simple, esto no se pudo realizar en el mapa del bosque debido a ser parte esencial en los elementos del mapa y su reemplazo con algún otro terreno significaría utilizar gran parte de tiempo en el desarrollo de otro terreno desde cero utilizando otro métodos. Es por lo anterior que se recomienda que en un trabajo futuro se aborde este problema para considerar este problema, considerando maneras de solucionar el error o bien creando un mapa utilizando *Computer Shaders* o utilizando herramientas gráficas como Blender para crear y exportar el modelo a la escena del bosque.

Pese a lo anterior, el mapa del bosque si se puede utilizar y es funcional respecto al flujo de acción de la aplicación, por lo que de igual manera se realizó una prueba de concepto de menor escala ($n=4$) para verificar que el programa y sus mecánicas funcionen adecuadamente, junto con obtener comentarios de los usuarios de prueba.

5.3.1. Metodología

Los participantes fueron personas jóvenes, entre 24 y 30 años, donde tres eran mujeres y uno hombre. Para cada persona, se le llevó a una sala donde se encontraba una mesa con un computador y un Oculus Quest 1, y una silla donde se debían sentar. Al estar sentados se les explicó el objetivo de la memoria, y que la metodología de obtención de datos sería *Thinking Aloud*, es decir, que ha medida que interactuaban con la aplicación debían comentar aquellos pensamientos sobre la aplicación y sobre lo que les iba ocurriendo. Después de probar el programa se les realizó tres preguntas: ¿Qué fue lo que más te gustó de la aplicación?, ¿Qué

¹<https://forums.oculusvr.com/t5/Oculus-Go-Development/Unity-Terrain-crashes-Gear/td-p/307926>

²<https://forum.unity.com/threads/gear-vr-terrain-crashing.432467/>

fue lo que menos te gustó de la aplicación?, ¿Qué cambios te gustarían tener en la aplicación?

Primero se les pidió realizar el tutorial y luego la actividad principal. A los participantes se les pidió elegir la cantidad de malestar al objeto fóbico que quisieran, pero eligiendo la araña como objeto fóbico y el bosque como mapa. Las condiciones de término de la aplicación fueron dos: el usuario encontraba a Fobos, o le costaba mucho y se decidía terminar la prueba para que no durara más de 15 minutos en la actividad principal. Una vez terminada la actividad principal se continuaba con las preguntas al usuario y se recogían también cualquier comentario que se considerara importante.

5.3.2. Resultados y análisis

A continuación se comentan los resultados obtenidos en esta prueba de usuarios, haciendo énfasis en los comentarios recibidos por los participantes y la manera en que interactuaron con la aplicación.

Aspectos positivos

En primer lugar destacaron algunos usuarios es que la actividad a realizar es fácil de entender y partir con la acción, en ningún momento fue necesario recordar las reglas del juego ni dar pistas de cómo deberían lograr su objetivos. Estos comentarios corroboraron que la actividad de buscar a Fobos en el mundo virtual fue una elección que impidió a los usuarios sentirse confusos sobre qué hacer en el mundo virtual, además de darles una motivación para explorar el ambiente virtual y aumentar la sensación de presencia.

Junto a lo anterior, un usuario comentó que se sentía presente en el mundo virtual y que consideraba el entorno como “muy real”. Además, los usuarios comentaron que el ambiente del bosque era visualmente bonito y entretenido de explorar; que los gráficos estaban bien hechos; y que la vegetación era un componente agradable de observar.

Varios usuarios comentaron que la interfaz era fácil de usar y clara al momento de avanzar a la siguiente parte del programa.

Considerando que una de las necesidades que motivaron el trabajo de esta memoria era la falta de involucramiento para incurrir en terapias de exposición para el manejo de fobias, el hecho de los todos los usuarios mostraron emoción al probar la aplicación se considera un logro para motivar a personas con alguna fobia a tratarse con este tipo de tecnologías y métodos novedosos.

Aspectos negativos y por mejorar

El aspecto que más se repitió dentro de los comentarios de los participantes fue que el objeto fóbico era muy difícil de encontrar. Esto se cree que se debe a que el contraste de Fobos con el terreno era muy leve, y que para encontrarlo se debía estar bastante cerca al modelo virtual. Además, el tamaño del terreno podría ser un factor importante también, puesto que los usuarios podrían disminuir su tiempo de observación en el lugar en el que están, para no perder mucho tiempo y poder alcanzar a revisar todo el mapa en un tiempo más corto.

Para lo anterior se consideran varias recomendaciones para disminuir lo difícil que es completar la actividad. En primer lugar el objeto fóbico podría ser de mayor tamaño y tener colores más destacable, pero habría que tener cuidado de qué tan grande sería apropiado en una perspectiva terapéutica de la aplicación. Esto último es importante, puesto que uno de los principios terapéuticos usados como estrategia para ayudar en el manejo de fobias es la violación de las expectativas del objeto fóbico, y si Fobos es más grande de lo que el usuario cree que es, entonces podría llegar a ser peligroso para el usuario. Es por esto, que este cambio debiese ser consultado con un experto de dominio.

En la misma línea, algunos usuarios propusieron ideas de que Fobos podría aparecer en varios lugares a lo largo del mapa, y así facilitar su búsqueda. Otros usuarios comentaron que podrían haber pistas en el mapa, como por ejemplo telarañas en el caso de la araña, que ayudaran a dar ideas de dónde podría estar el modelo en el mapa. Un participante comentó que le hubiese gustado que hubiera sonidos en la aplicación, por ejemplo, el sonido del viento, las ramas o del mismo Fobos.

Por otro lado, algunos usuarios comentaron que el cambio de ángulo de visión de manera brusca en 45° grados les causaba mareos e incomodidad. Pero por otro lado, otro usuario comentaba que ese movimiento era preferible por sobre otros movimientos graduales. Es por esto que se considera que debe existir una configuración manual para el usuario que le permita definir de qué manera prefiere rotar su visión: de manera gradual o por un cierto ángulo. Junto a esto, los usuarios mostraron un cierto nivel de mareos al final del uso de la aplicación, salvo uno, que no tuvo ningún problema de inicio a fin.

Finalmente, algunos usuarios comentaron que se les olvidaba a veces qué hacían ciertos controles y que necesitaba practicar más para poder aprendérselos. En particular, comentaron que los botones y los *trigger* se les confundían ciertas veces. Pero con las palancas nunca hubo problema, y fue un movimiento que les resultaba natural. Es por lo anterior que se recomienda agregar en futuras iteraciones algún tipo de recordatorio para las posibles interacciones con los controles durante el uso de la aplicación. Esto puede ser algún tipo de botón de pausa en donde salgan imágenes de los botones y qué hacen dentro del programa.

5.4. Aprendizajes principales

En base a los resultados, se validaron y analizaron los componentes del diseño de la solución.

En este análisis se reconocen como aspectos positivos: el disfrute de los usuarios durante el uso de la aplicación; la capacidad de involucrarse con un ambiente virtual; y las mecánicas que el usuario debe experimentar para cumplir con los objetivos. En base a los valores obtenidos, a los usuarios les gustó la actividad y el ambiente virtual en el que se insertaron, además la encontraron desafiante y sabían cómo proseguir para lograr sus objetivos.

Por otro lado, se reconocen los aspectos a mejorar en la aplicación: es necesario encontrar maneras de disminuir los mareos para aquellos usuarios más susceptibles a este problema; se debe considerar aumentar la capacidad de interacción de la utilería en la escena para el usuario; y cubrir los errores de desarrollo presenciados por los usuarios, tales como vibraciones

en segmentos específicos, problemas para abrir las puertas, regular los cambios del campos de visión. Además, en el nivel del bosque, es necesario disminuir la dificultad de encontrar a Fobos, ya sea incrementando su número en el mapa, o dando pistas que guíen al usuario.

En relación a los objetivos propuestos inicialmente, estos fueron cumplidos en su mayoría, puesto que el objetivo relativo a generalizar a distintos tipos de fobias se ve acotado a objetos fóbicos cuyo modelo virtual sea pequeño y fácil de esconder. En particular, se logró diseñar e implementar un prototipo de una aplicación en realidad virtual con un sistema de regulación gráfica para los modelos del mundo virtual. Lo anterior se pudo ver en acción cuando a los participantes se les asignaba un modelo con la fidelidad visual según las respuestas del cuestionario SUD en el menú principal. Además, los usuarios mostraron entusiasmo y goce al interactuar con el mundo y buscar a Fobos. Finalmente, durante la etapa de diseño e implementación se aplicaron cambios a partir de reuniones con la experta de dominio, que terminaron beneficiando el potencial componente psicoterapéutico de la aplicación.

Capítulo 6

Conclusiones y trabajo futuro

A continuación se presentan las conclusiones del desarrollo de la memoria, junto con el cumplimiento de los objetivos y los cambios que tendrían los trabajos a futuro del proyecto.

Durante el desarrollo de esta memoria, fue posible observar la multidimensionalidad de un prototipo de aplicación en realidad virtual con componentes lúdicos. En particular, las dimensiones involucradas en este desarrollo abarcaron desde la jugabilidad y aspectos gráficos y técnicos/computacionales, hasta la presencia, interacción humano-computador, y elementos psicoterapéuticos. Es por ello que la dificultad de este proyecto radicó en la integración de todos estos elementos para cumplir con los objetivos propuestos. Por lo tanto, este trabajo consistió en obtener un equilibrio entre las partes y saber priorizar el manejo del recurso del tiempo de desarrollo. En este balanceo, fueron críticos los múltiples cambios incrementales y pruebas informales con usuarios, para reconocer cuándo era necesario aumentar la prioridad de ciertas dimensiones sobre otras.

En el progreso de este trabajo, fue necesario aprender sobre las distintas dimensiones del proyecto. Inicialmente se definió un plan de trabajo y luego el diseño de la solución. Este último se basó en las características técnicas de Unity como motor de videojuegos, y en el paquete de *XR Interaction Toolkit*. Pero además, las necesidades psicoterapéuticas, como la inclusión de múltiples contextos, definieron los objetivos de la solución. En base a esos dos componentes, es que se formaron las dinámicas de la actividad, las escenas del mundo virtual (como el tutorial, la casa y el bosque) y las capacidades de interacción del usuario con el mundo virtual. Junto a los aprendizajes anteriores, y debido al carácter multidisciplinario de este trabajo de título, se necesitó investigar sobre el trastorno de ansiedad que es la fobia, su tratamiento, y junto a las recomendaciones y la validación de la experto de dominio en psicoterapia, se logró incorporar el componente terapéutico a la solución.

A partir de la prueba de concepto realizada, se evaluó la aplicación para corroborar que la solución final corresponde a un producto mínimo viable para el potencial uso terapéutico con componentes de variabilidad gráfica. Junto a esto, se evidenció la aceptación general de la solución y sus componentes de jugabilidad, presencia y de funcionalidad. Los usuarios sintieron que el programa proponía un entorno con reglas y mecánicas claras, y consideraron que la experiencia durante el uso de la aplicación fue positiva.

Aún cuando los resultados cualitativos obtenidos en la prueba de usuario son útiles y provechosos para validar y mejorar la aplicación, estos no están libre de limitaciones debido al reducido tamaño de la muestra analizada.

Finalmente, el aspecto negativo que resaltó en los resultados es la sensación de mareos. Esto ocurrió debido al carácter exploratorio y visual de la aplicación, y por lo tanto era esperado que los usuarios presentaran mareos. En un futuro, aquellos elementos de la aplicación causantes de esta condición deben ser tomados en consideración para las futuras intervenciones en el proyecto. Además, la dificultad del juego, en particular en mapas de tamaño grande como el bosque, debe ser disminuido para hacer el objetivo de la actividad más realizable en un tiempo razonable.

A partir del desarrollo del proyecto y los resultados obtenidos en las pruebas de usuario, se concluye que los objetivos específicos han sido cumplidos, pero el objetivo general se cumplió con la restricción de que los modelos fóbicos deben ser de un tamaño pequeño. En particular, se diseñó e implementó un prototipo de aplicación con un sistema de regulación gráfica en realidad virtual, el que fue probado y construido en un Oculus Quest 1. Además, el diseño de la aplicación siguió las directrices de una psicóloga e investigadora enfocada en trastornos de ansiedad, y por ello se presenta como la experta de dominio. Los comentarios recibidos fueron incorporados durante el inicio y el desarrollo de la solución para incrementar la eficiencia para una herramienta potencial de terapia de manejo de fobias.

A lo largo del desarrollo del prototipo y a partir del análisis de los resultados, se identificaron ciertos aspectos del proyecto que necesitarían ser mejorados y presentan una oportunidad de mejora para la solución en futuras iteraciones.

En primer lugar, es necesario aumentar la diferencia entre la complejidad gráfica entre los modelos de Fobos bajo la misma especie. Para este cambio es recomendable que se tome en cuenta la opinión de un experto en el área de diseño en modelado 3D para delimitar bien los extremos gráficos. Sería necesario, previo al desarrollo de los modelos, tener una idea clara de cuán fidedigno a la representación real debe quedar el modelo para que no sea contraproducente el realismo. Este modelo también requerirá la validación y guía de un experto de dominio del área de los trastornos de ansiedad. A partir de lo anterior, será importante saber desde qué momento un modelo virtual deja de ser útil para la solución, por su bajo nivel de realismo gráfico. Si bien esto no se realizó durante la memoria, debido a que no se encontraron los modelos apropiados en el *Unity Asset Store* y el costo de hacerlo es muy alto, se recomienda trabajar con una persona con experiencia en modelamiento gráfico de 3D para obtener mejores resultados.

Similarmente, el avance natural de este proyecto sigue en dirección a generar más modelos virtuales de otras especies, y así aumentar el potencial de uso hacia una población mayor de personas con algún trastorno fóbico. Además, sería interesante estudiar el efecto que tendría el agregar más características a Fobos, por ejemplo, animaciones y sonido. Estas propiedades podrían ser agregadas al modelo de Fobos con mayor fidelidad visual, o bien dejarlo como una configuración en el menú principal.

Se propone además hacer un estudio para disminuir las posibles causas del mareo percibido durante la prueba de usuarios. Un primer acercamiento para resolver el problema sería

cambiar el tipo de movimiento de uno continuo, a otro basado en teletransportación usando el componente **Teleportation Provider** y además hacer que la rotación del campo visual cambie a uno continuo, esto sería posible ajustando los parámetros del componente **Snap Turn Provider** en el **GameObject VR Origin**.

Otro alcance para futuras iteraciones podría abarcar el arreglar el *build* para los mapas con terreno nativo de Unity. Esto permitiría aprovechar el desarrollo del mapa del bosque y anteriores iteraciones del tutorial. Junto a lo anterior, se podrían utilizar las texturas nativas, con mayor resolución, para aumentar la calidad visual del proyecto y así mejorar la experiencia visual del usuario en el ambiente virtual.

Con respecto a la metodología seguida durante la prueba de usuarios, podría haber ocurrido un sesgo para la encuesta de presencia, debido al ruido de los otros participantes a la espera de su turno. Por lo tanto, se debe tener en consideración este problema al realizar futuras pruebas de usuarios o cuestionarios para la investigación de posibles efectos terapéuticos de la aplicación.

Finalmente, este trabajo se enfoca en un prototipo inicial para una herramienta de *software* potencialmente efectiva para complementar terapias de manejo de fobias, pero esto no limita al proyecto a extender su aplicación a involucrarse más activamente en el tratamiento de fobias. Junto con los trabajos futuros recomendados, se entrega el presente proyecto al profesor guía para que esta manera el proyecto pueda crecer y ser algún día una ayuda para aquellas personas sufriendo alguna fobia.

Bibliografía

- [1] Anderson, Page L y Anthony Molloy: *Maximizing the impact of virtual reality exposure therapy for anxiety disorders.* Current Opinion in Psychology, 36:153–157, 2020, ISSN 2352-250X. <https://www.sciencedirect.com/science/article/pii/S2352250X20301974>, Cyberpsychology.
- [2] Beidel, Deborah C, B Christopher Frueh, Sandra M Neer, Clint A Bowers, Benjamin Trachik, Thomas W Uhde y Anouk Grubaugh: *Trauma management therapy with virtual-reality augmented exposure therapy for combat-related PTSD: A randomized controlled trial.* Journal of anxiety disorders, 61:64–74, 2019.
- [3] Botella, C., J. Breton-López, S. Quero, R.M. Baños, A. García-Palacios, I. Zaragoza y M. Alcaniz: *Treating cockroach phobia using a serious game on a mobile phone and augmented reality exposure: A single case study.* Computers in Human Behavior, 27(1):217–227, 2011, ISSN 0747-5632. <https://www.sciencedirect.com/science/article/pii/S0747563210002396>, Current Research Topics in Cognitive Load Theory.
- [4] Bouchard, Stéphane, Sophie Côté, Julie St-Jacques, Geneviève Robillard y Patrice Renaud: *Effectiveness of virtual reality exposure in the treatment of arachnophobia using 3D games.* Technology and health care, 14(1):19–27, 2006.
- [5] Calvillo-Gamez, Eduardo H., Paul Cairns y Anna L. Cox: *Assessing the Core Elements of the Gaming Experience*, páginas 37–62. Springer International Publishing, Cham, 2015, ISBN 978-3-319-15985-0. https://doi.org/10.1007/978-3-319-15985-0_3.
- [6] Chang, Eunhee, Hyun Taek Kim y Byounghyun Yoo: *Virtual Reality Sickness: A Review of Causes and Measurements.* International Journal of Human–Computer Interaction, 36(17):1658–1682, 2020. <https://doi.org/10.1080/10447318.2020.1778351>.
- [7] Choy, Yujuan, Abby J Fyer y Josh D Lipsitz: *Treatment of specific phobia in adults.* Clinical psychology review, 27(3):266–286, 2007.
- [8] Deterding, Sebastian, Dan Dixon, Rilla Khaled y Lennart Nacke: *From Game Design Elements to Gamefulness: Defining "Gamification".* En *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, MindTrek '11, página 9–15, New York, NY, USA, 2011. Association for Computing Machinery, ISBN 9781450308168. <https://doi.org/10.1145/2181037.2181040>.

- [9] Edition, Fifth *y cols.*: *Diagnostic and statistical manual of mental disorders*. Am Psychiatric Assoc, 21, 2013.
- [10] Feeny, Norah C, Elizabeth A Hembree y Lori A Zoellner: *Myths regarding exposure therapy for PTSD*. Cognitive and Behavioral Practice, 10(1):85–90, 2003.
- [11] Freeman, Daniel, Polly Haselton, Jason Freeman, Bernhard Spanlang, Sameer Kishore, Emily Albery, Megan Denne, Poppy Brown, Mel Slater y Alecia Nickless: *Automated psychological therapy using immersive virtual reality for treatment of fear of heights: a single-blind, parallel-group, randomised controlled trial*. The Lancet Psychiatry, 5(8):625–632, 2018.
- [12] Garcia-Palacios, Azucena, Hunter G Hoffman, Sheree Kwong See, AMY Tsai y Cristina Botella: *Redefining therapeutic success with virtual reality exposure therapy*. CyberPsychology & Behavior, 4(3):341–348, 2001.
- [13] Griffiths, Mark, Daria J Kuss y Angelica B Ortiz de Gortari: *Videogames as therapy: A review of the medical and psychological literature*. Handbook of Research on ICTs and Management Systems for Improving Efficiency in Healthcare and Social Care, páginas 43–68, 2013.
- [14] Hartson, H. Rex y Deborah Hix: *Human-Computer Interface Development: Concepts and Systems for Its Management*. ACM Comput. Surv., 21(1):5–92, mar 1989, ISSN 0360-0300. <https://doi.org/10.1145/62029.62031>.
- [15] Holzinger, Andreas: *Usability Engineering Methods For Software Developers*. Commun. ACM, 48:71–74, Enero 2005.
- [16] Höglberg, Johan, Juho Hamari y Erik Wästlund: *Gameful Experience Questionnaire (GAMEFULQUEST): an instrument for measuring the perceived gamefulness of system use*. User Modeling and User-Adapted Interaction, 29, Julio 2019.
- [17] Iancu, Iulian, Jennifer Levin, Pinhas N Dannon, Amir Poreh, Yoram Ben Yehuda y Moshe Kotler: *Prevalence of self-reported specific phobia symptoms in an Israeli sample of young conscripts*. Journal of anxiety disorders, 21(5):762–769, 2007.
- [18] Issakidis, Cathy y G Andrews: *Pretreatment attrition and dropout in an outpatient clinic for anxiety disorders*. Acta Psychiatrica Scandinavica, 109(6):426–433, 2004.
- [19] Krijn, M, P.M.G Emmelkamp, R.P Olafsson y R Biemond: *Virtual reality exposure therapy of anxiety disorders: A review*. Clinical Psychology Review, 24(3):259–281, 2004, ISSN 0272-7358. <https://www.sciencedirect.com/science/article/pii/S0272735804000418>.
- [20] Lau, Ho Ming, Johannes H. Smit, Theresa M. Fleming y Heleen Riper: *Serious Games for Mental Health: Are They Accessible, Feasible, and Effective? A Systematic Review and Meta-analysis*. Frontiers in Psychiatry, 7:209, 2017, ISSN 1664-0640. <https://frontiersin.org/article/10.3389/fpsyg.2016.00209>.

- [21] Levy, Fanny, Pierre Leboucher, Gilles Rautureau, Odile Komano, Bruno Millet y Roland Jouvent: *Fear of falling: efficacy of virtual reality associated with serious games in elderly people*. Neuropsychiatric disease and treatment, 12:877, 2016.
- [22] Lewis, Christopher H y Michael J Griffin: *Applications of Virtual Reality*. Virtual reality in neuro-psycho-physiology: Cognitive, clinical and methodological issues in assessment and rehabilitation, 44:35, 1997.
- [23] Lindner, Philip, Alexander Miloff, Camilla Bergman, Gerhard Andersson, William Hamilton y Per Carlbring: *Gamified, Automated Virtual Reality Exposure Therapy for Fear of Spiders: A Single-Subject Trial Under Simulated Real-World Conditions*. Frontiers in Psychiatry, 11:116, 2020, ISSN 1664-0640. <https://www.frontiersin.org/article/10.3389/fpsyt.2020.00116>.
- [24] Lindner, Philip, Alexander Miloff, Simon Fagernäs, Joel Andersen, Martin Sigeman, Gerhard Andersson, Tomas Furmark y Per Carlbring: *Therapist-led and self-led one-session virtual reality exposure therapy for public speaking anxiety with consumer hardware and software: A randomized controlled trial*. Journal of anxiety disorders, 61:45–54, 2019.
- [25] Lindner, Philip, Alexander Miloff, Elin Zetterlund, Lena Reuterskiöld, Gerhard Andersson y Per Carlbring: *Attitudes toward and familiarity with virtual reality therapy among practicing cognitive behavior therapists: a cross-sectional survey study in the era of consumer VR platforms*. Frontiers in psychology, 10:176, 2019.
- [26] Lindner, Philip, Alexander Rozental, Alice Jurell, Lena Reuterskiöld, Gerhard Andersson, William Hamilton, Alexander Miloff y Per Carlbring: *Experiences of Gamified and Automated Virtual Reality Exposure Therapy for Spider Phobia: Qualitative Study*. JMIR Serious Games, 8(2):e17807, Apr 2020, ISSN 2291-9279. <http://games.jmir.org/2020/2/e17807/>.
- [27] Makransky, Guido, Lau Lilleholt y Anders Aaby: *Development and Validation of the Multimodal Presence Scale for Virtual Reality Environments: A Confirmatory Factor Analysis and Item Response Theory Approach*. Computers in Human Behavior, 72, Febrero 2017.
- [28] Miloff, Alexander, Philip Lindner, William Hamilton, Lena Reuterskiöld, Gerhard Andersson y Per Carlbring: *Single-session gamified virtual reality exposure therapy for spider phobia vs. traditional exposure therapy: study protocol for a randomized controlled non-inferiority trial*. Trials, 17(1):1–8, 2016.
- [29] Milosevic, Irena y Randi E McCabe: *Phobias: The psychology of irrational fear: The psychology of irrational fear*. Abc-Clio, 2015.
- [30] Minns, Sean, Andrew Levihn-Coon, Emily Carl, Jasper AJ Smits, Wayne Miller, Don Howard, Santiago Papini, Simon Quiroz, Eunjung Lee-Furman, Michael Telch y cols.: *Immersive 3D exposure-based treatment for spider fear: A randomized controlled trial*. Journal of anxiety disorders, 61:37–44, 2019.

- [31] Olatunji, Bunmi O, Brett J Deacon y Jonathan S Abramowitz: *The cruellest cure? Ethical issues in the implementation of exposure-based treatments.* Cognitive and Behavioral Practice, 16(2):172–180, 2009.
- [32] Oskam, Paul: *Virtual reality exposure therapy (VRET) effectiveness and improvement.* En *2nd Twente University Student Conference on IT, Enschede, The Netherlands.* CiteSeer, 2005.
- [33] Parsons, Thomas D y Albert A Rizzo: *Affective outcomes of virtual reality exposure therapy for anxiety and specific phobias: A meta-analysis.* Journal of behavior therapy and experimental psychiatry, 39(3):250–261, 2008.
- [34] Peskin, Melissa, Katarzyna Wyka, Judith Cukor, Megan Olden, Margaret Altemus, Francis S Lee y JoAnn Difede: *The relationship between posttraumatic and depressive symptoms during virtual reality exposure therapy with a cognitive enhancer.* Journal of anxiety disorders, 61:82–88, 2019.
- [35] Powers, Mark B y Barbara O Rothbaum: *Recent advances in virtual reality therapy for anxiety and related disorders: Introduction to the special issue.* Journal of anxiety disorders, 61:1–2, 2018.
- [36] Prochaska, James O y John C Norcross: *Systems of psychotherapy: A transtheoretical analysis.* Oxford University Press, 2018.
- [37] Richard, David CS y Andrew T Gloster: *Exposure therapy has a public relations problem: A dearth of litigation amid a wealth of concern.* En *Handbook of exposure therapies,* páginas 409–425. Elsevier, 2007.
- [38] TAFFOU1a, Marine, Emmanuelle Chapoulie, Adrien David, Rachid Guerchouche, George Drettakis y Isabelle Viaud-Delmon: *Auditory-visual integration of emotional signals in a virtual environment for cynophobia.* Annual Review of Cybertherapy and Telemedicine 2012, página 238, 2012.
- [39] Torrents-Rodas, David, Miquel A. Fullana, Bram Vervliet, Michael Treanor, Chris Conway, Tomislav Zbozinek y Michelle G. Craske: *Maximizar la terapia de exposición: Un enfoque basado en el aprendizaje inhibitorio.* Revista de Psicopatología y Psicología Clínica, 20(1):1–24, abr. 2015. <https://revistas.uned.es/index.php/RPPC/article/view/14403>.
- [40] Weibel, David y Bartholomäus Wissmath: *Immersion in Computer Games: The Role of Spatial Presence and Flow.* Int. J. Comput. Games Technol., 2011, jan 2011, ISSN 1687-7047. <https://doi.org/10.1155/2011/282345>.
- [41] Wells, Adrian: *Cognitive therapy for social phobia.* Handbook of brief cognitive behaviour therapy, página 141, 2004.

Capítulo 7

Anexo

Tabla 7.1: Tabla de la mediana según preguntas del *CEGEQ*, junto con su tipo de agrupación

Valor	Agrupación
6	Disfrute
4	Frustracion
2	Frustracion
6	Disfrute
6	Disfrute
6	CEGE+Puppetry
5	CEGE+Puppetry
7	CEGE+Puppetry
6	CEGE+Puppetry
2	CEGE+Puppetry
6	CEGE+Puppetry
2	CEGE+Puppetry
5	CEGE
6	CEGE
2	CEGE
3	CEGE
3	CEGE
7	CEGE
5	CEGE
3	CEGE
5	CEGE
3	CEGE
4	CEGE
6	CEGE
5	CEGE+Videojuego
6	CEGE+Videojuego
4	CEGE+Videojuego
2	CEGE+Videojuego
6	CEGE+Videojuego
5	CEGE+Videojuego
5	CEGE+Videojuego
5	CEGE+Videojuego
2	CEGE+Videojuego
5	CEGE+Videojuego
5	CEGE+Videojuego
76	El escenario del juego era interesante.
No me gustó el escenario del juego.	