

# Aplicación de realidad virtual para estimulación de la memoria en un contexto multijugador



Universidad  
del Cauca

Daniel Mauricio Talaga Rodriguez

Director:

M.Sc. Diego Enrique Guzmán Villamarin

Co-director:

PhD. Carlos Felipe Rengifo Rodas

Co-director:

Jeronimo Londoño

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Ingeniería en Automática Industrial

Popayán, Febrero 2022

# Resumen

Este trabajo de grado presenta un juego serio, denominado *Brain Race*, basado en un entorno de realidad virtual inmersiva multijugador para estimular la memoria visual a corto plazo en personas sanas. Su principal aporte consiste en la implementación de tres modalidades de juego diseñadas para dos jugadores y una de un solo jugador, con el fin de estudiar el efecto de estas en la estimulación de la memoria. Para estudiar la incidencia de las modalidades de juego se realiza un experimento probando el juego serio diseñado en dieciséis personas sanas sin antecedentes médicos de alteraciones de la memoria, las cuales se dividieron en grupos de cuatro personas asignando a cada una las modalidades de juego; se aplicó el test de Benton al iniciar y al finalizar el experimento con el fin de evaluar su memoria visual. Mediante los resultados obtenidos en el experimento se obtiene que, no hay un incremento significativo en el porcentaje de aciertos en el test de Benton, en función de la modalidad de juego.

**Palabras clave:** juego serio, memoria visual a corto plazo, realidad virtual, estimulación cognitiva, multijugador.

# Índice general

<b>1. Generalidades</b>	<b>6</b>
1.1. Estado del arte . . . . .	6
1.2. Planteamiento del problema . . . . .	10
1.3. Objetivos . . . . .	11
1.4. Estructura de la monografía . . . . .	11
<b>2. Desarrollo del juego serio</b>	<b>13</b>
2.1. Metodología. . . . .	13
2.1.1. Metodología scrum . . . . .	14
2.2. Tabla de requerimientos . . . . .	16
2.3. Diseño del juego serio . . . . .	18
<b>3. Diseño del entorno 3D</b>	<b>34</b>
<b>4. Experimentación y resultados</b>	<b>49</b>
<b>5. Discusión y conclusiones</b>	<b>58</b>
5.1. Trabajos futuros . . . . .	59
<b>Anexos</b>	<b>67</b>
<b>A. Documentación software:</b>	<b>68</b>
<b>B. Manual de usuario</b>	<b>87</b>

# Índice de tablas

2.1. Diferencias entre las metodologías ágiles y tradicionales . . . . .	14
2.2. Tabla de requerimientos. . . . .	18
2.3. Narrativa primer caso de uso “elegir modalidad“. . . . .	22
2.4. Narrativa caso de uso “crear partida“. . . . .	22
2.5. Narrativa caso de uso “buscar partida“. . . . .	23
2.6. Narrativa caso de uso “jugador listo“. . . . .	23
2.7. Narrativa caso de uso “esperar jugador“. . . . .	23
2.8. Narrativa caso de uso “empezar partida“. . . . .	24
2.9. Narrativa caso de uso “conducir“. . . . .	24
2.10. Narrativa caso de uso “contestar preguntas“. . . . .	24
2.11. Narrativa caso de uso “almacenar datos“. . . . .	25
4.1. Características de la población. . . . .	50
4.2. Resultados test de Benton. . . . .	53

# Índice de figuras

2.1. Proceso metodologia Scrum . . . . .	16
2.2. Mockups (fuente propia). . . . .	21
2.3. Primer diagrama de casos de uso. . . . .	22
2.4. Segundo diagrama de casos de uso . . . . .	23
2.5. Tercer diagrama de casos de uso. . . . .	24
2.6. Tercer diagrama de casos de uso. . . . .	25
2.7. Diagrama de despliegue. . . . .	25
2.8. Diagrama de flujo. . . . .	26
2.9. Librería de mirror en Unity. . . . .	27
2.10. Network Manager. . . . .	28
2.11. Network Identity. . . . .	28
2.12. Network Identity. . . . .	29
2.13. Network Rigidbody. . . . .	29
2.14. Google Cardboard XR Plugin for Unity. . . . .	30
2.15. Cardboard . . . . .	30
2.16. Volante y pedales diseñados. . . . .	31
2.17. Soporte en PVC cabina de conducción. . . . .	32
2.18. Diagrama de circuitos. . . . .	33
2.19. Montaje completo con mecanismo electrónico. . . . .	33
3.1. Terrenos en unity. . . . .	34
3.2. Edición de un objeto tipo terreno. . . . .	35
3.3. Material de un terreno. . . . .	35
3.4. Herramienta paint trees. . . . .	36
3.5. Escenario bosque diseñado para la modalidad individual y competitiva. . . . .	37
3.6. Escenario ciudad diseñado para la modalidad colaborativa. . . . .	38
3.7. Ubicación de la Asset store. . . . .	38
3.8. Aviso de sobre nueva forma de ingreso. . . . .	39
3.9. Asset store desde el buscador de google. . . . .	39
3.10. Resultados de la búsqueda. . . . .	40
3.11. Importar modelo. . . . .	40
3.12. Solicitud de permiso para abrir el modelo 3D desde <i>Unity</i> . . . . .	41
3.13. Información del modelo 3D en <i>Unity</i> . . . . .	41
3.14. Modelo 3D que contiene el paquete seleccionado. . . . .	42
3.15. Escenario de paisaje con atardecer diseñado para la modalidad cooperativa. . . . .	43
3.16. Imágenes modelo 3D de la camioneta GSL 450 Mercedes Benz. . . . .	44

3.17. Objeto tipo textmeshpro. . . . .	45
3.18. Tacómetro diseñado. . . . .	45
3.19. Tacómetro diseñado. . . . .	46
3.20. Plano de la pista. . . . .	46
3.21. Trazos en blender. . . . .	46
3.22. Modelo 3D creados en blender. . . . .	47
3.23. Banderín o meta. . . . .	47
3.24. Modelo 3D de arboles. . . . .	48
3.25. Imágenes que se deben de memorizar. . . . .	48
4.1. Lamina versus reproducción test de Benton. . . . .	51
4.2. Aplicación test de Benton. . . . .	52
4.3. Respuestas encuesta de satisfacción realizada a la población del experimento. . . . .	55
B.1. APK Brain Race. . . . .	87
B.2. Procedencia desconocida. . . . .	88
B.3. Instalación app. . . . .	89
B.4. Abrir aplicación. . . . .	89
B.5. Ventanas inicio app. . . . .	90
B.6. Pantalla de registro. . . . .	90
B.7. Menú principal . . . . .	90
B.8. Elegir individual . . . . .	91
B.9. Elegir individual . . . . .	91
B.10. Elegir individual . . . . .	91
B.11. Elegir colaborativo . . . . .	92
B.12. Elegir rol piloto o copiloto . . . . .	92
B.13. Rol piloto. . . . .	93
B.14. Rol copiloto. . . . .	93
B.15. Comienza la partida . . . . .	94
B.16. Elegir cooperativo . . . . .	94
B.17. Elegir jugador uno o jugador dos. . . . .	94
B.18. Jugador uno modalidad cooperativa. . . . .	95
B.19. Jugador dos modalidad cooperativa. . . . .	95
B.20. Elegir competitivo . . . . .	95
B.21. Elegir jugador uno o jugador dos. . . . .	96
B.22. Jugador uno modalidad competitiva. . . . .	96
B.23. Jugador dos. . . . .	97
B.24. Acelerar. . . . .	97
B.25. Frenar. . . . .	98
B.26. Retroceder. . . . .	98
B.27. Girar el vehículo. . . . .	99

# Capítulo 1

## Generalidades

En el siguiente capítulo se presenta el estado del arte, el cual es fundamental debido a que permite afirmar la formulación del proyecto o tema de investigación. También se presenta el planteamiento del problema de esta investigación y sus objetivos.

### 1.1. Estado del arte

En la actualidad el tratamiento convencional del déficit de la memoria consiste en el uso de diferentes estrategias como las presentadas en [1], algunas de las más utilizadas son: medicamentos para detener la progresión de la pérdida de la memoria [2, 3, 4], ejercicios físicos con el fin de aumentar el flujo de oxígeno en el cerebro [5], tratamientos lúdicos para motivar al paciente [6], alimentación saludable rica en minerales, la cual contribuye a mejorar las conexiones neuronales [7], técnicas para la consolidación del recuerdo [8], vida social activa [9] y entrenamiento cognitivo para estimular el cerebro [9, 10].

Según [11] de las estrategias mencionadas anteriormente, es fundamental el entrenamiento o estimulación cognitiva debido a que esta permite prevenir y tratar el déficit de la memoria, además consideran que debe estar presente en todo tipo de tratamiento cognitivo. En [12] concluyen que este tipo de estímulos permite a las personas reactivar y mantener en funcionamiento sus capacidades mentales, además de regenerar y aumentar la creación de nuevas conexiones neuronales en zonas afectadas [12]. Este proceso de estimulación consiste en la realización constante y apropiada de diferentes ejercicios con los cuales se busca desarrollar cuatro principios generales como: (1) la atención, mediante la cual elegimos o ignoramos información, (2) el sentido o significado que aportamos a cualquier objeto que se pretende memorizar, (3) la organización de datos debido a que esta permite recordarlos fácilmente y por último (4) la asociación, que consiste en relacionar lo que se desea memorizar con algún objeto o concepto aprendido anteriormente [13]. Por otra parte autores como [14] demuestran que mientras las personas buscan soluciones a ejercicios cognitivos utilizan áreas cognitivas como: la memoria, las funciones ejecutivas, el lenguaje, el cálculo, el razonamiento, las apraxias, la orientación y atención.

Algunas de las herramientas convencionales utilizadas por los especialistas para el entre-

namiento de la memoria son: el programa de entrenamiento combinado para la memoria y psicomotricidad SIMA [15], ejercicios de motricidad y memoria para personas mayores [16], Terapia de reminiscencia [16], Taller de memoria (Estimulación y mantenimiento cognitivo en personas mayores) [17], Método UMAM (Unidad de memoria de ayuntamiento de Madrid) [18], Modelo de J.A Yesavage [19], Programa de Lilianne Israel, Grupos de memoria de B. Wilson, Grupos buenos días [20], Grupo de orientación a la realidad, Programas de psicoestimulación cognitiva [21], actividad lúdica de chantón, crucigramas, entre otros [22].

Autores como [23, 24] han realizado investigaciones con las herramientas mencionadas anteriormente y demuestran algunas deficiencias del proceso convencional para tratamiento de la memoria basado en el entrenamiento o estimulación cognitiva. Algunas de las críticas que realizan son: la incapacidad para reflejar la complejidad de los procesos de memorización en la vida diaria, dado que en la realidad hay diferentes distractores y estímulos los cuales difícilmente se pueden recrear en un consultorio. Otra deficiencia que presentan algunos test, es que se dedican a estimular especialmente el almacenamiento de información y no el procesamiento, ocasionando así incertidumbre en el resultado, porque existen pacientes que pueden memorizar ciertos datos y sin embargo no entienden el significado o uso de estos. Por último muchas de estas herramientas se basan en ejercicios de papel y lápiz, lo cual ocasiona poca utilidad práctica para el paciente, debido a que no se desarrollan estrategias que apoyen el proceso de memorización.

Gracias al desarrollo de la tecnología se han diseñado tratamientos no convencionales para la pérdida de la memoria, los cuales buscan solucionar las deficiencias mencionadas anteriormente. Este tipo de tecnologías se denominan juegos serios cuyo objetivo es brindar un entorno lúdico el cual apoye al especialista en la rehabilitación cognitiva de los pacientes. Estas herramientas pueden contener desde agendas electrónicas para recordar datos importantes como lo son futuras consultas o alarmas para medicamentos, hasta ejercicios mentales en realidad virtual o aumentada, donde el paciente interactúa en un ambiente muy semejante al mundo real, con diferentes variables y distractores controlados por el especialista. Autores como [25, 26] consideran que los juegos serios son herramientas en constante desarrollo que no se limitan únicamente al campo de la estimulación cognitiva sino que son utilizadas con diferentes propósitos como: publicidad, concientización, aprendizaje, entrenamiento militar, simulaciones de vuelo, capacitación laboral, rehabilitación, entre otras [27]. Algunos ejemplos de estas herramientas son: Lumosity [28], Attention process training-3 [29], Fit brains trainer [30], Peak [30], Elevate, Ice Cream Selle Test, Virtual store, Virtual action planning supermarket, Virtual mall, Virtual library test [23, 24], etc [31].

Según [31], el uso de estas herramientas ha permitido que las personas estimulen sus capacidades cognitivas de una manera totalmente diferente y lúdica, debido a que realizan ejercicios virtuales como: actividades de memorización de patrones durante un tiempo corto para después recrearlos, esto conlleva a mejorar la percepción visual, recordar la ubicación de los objetivos y entender entornos. También existen actividades de planificación donde el usuario debe de solucionar un problema de como el unir los vértices de una figura, pasando por sus lados una única vez, esto permite desarrollar habilidades de

planificación, reacción al cambio, percepción espacial, agilidad en la toma de decisiones. Otro tipo de actividades que contiene son de enfoque donde se debe de diferenciar que objetos debemos de tocar o evitar, esto conlleva a mejorar la coordinación psicomotriz, ganar velocidad de procesamiento, desarrollar percepción visual y espacial.

Los juegos serios se caracterizan por tener diferentes componentes, configuraciones y alternativas que ayudan a mejorar la experiencia que tienen las personas dentro del entorno. En estudios como [32], mencionan que una de las configuraciones con mayor influencia en el desarrollo de habilidades para la socialización y comunicación es el modo de juego, el cual consiste en elegir el número de jugadores y la forma en que se realizara la actividad propuesta, los diferentes modos de juego son: [33] (i) competitivo, donde los participantes son rivales y deben esforzarse para conseguir el objetivo individualmente, [34](ii) colaborativo, los usuarios deben trabajar juntos para llegar a un mismo fin y [35] (iii) cooperativo, el cual radica en dividir el trabajo entre los participantes del juego, haciendo que cada uno sea responsable de su parte. A continuación se presentaran los beneficios de cada modalidad y algunos ejemplos.

El modo competitivo en los juegos serios ha tenido resultados favorables en el contexto de la capacitación laboral, así lo demuestran en [36] donde se implementa un juego serio competitivo orientado al proceso de formación en el campo de los negocios internacionales y mercadeo de organizaciones con vocación internacional, el propósito de este juego es que las personas adquieran o desarrollen conocimientos y habilidades de negociación en un entorno basado en el juego de mesa *MONOPOLY*, esta herramienta fue implementada en un grupo de personas donde se evidencia que mejoran sus habilidades duras y blandas en el cargo que deben de realizar dentro del juego; además permite que se enfrenten a situaciones semejantes a las del mundo real, donde tiene que tomar decisiones basadas en sus propios conocimientos académicos y laborales. Por otra parte, el modo competitivo también se ha utilizado en el área de la educación escolar y universitaria, en [37] se concluye que la competencia entre los estudiantes aumenta el uso voluntario de las herramientas de aprendizaje y por ende su rendimiento académico, potencializa sus habilidades de comunicación, desarrolla confianza y seguridad en sí mismo, entiende el esfuerzo que conlleva ganar, aprende a perder y a controlar las emociones que esto genera como lo son la rabia y la frustración. Algunos juegos serios con modalidad competitiva son: *Skinquization* [38], *DietaDialisisQuiz* [39], *Cyberspace Odyssey* [40], *Platform wars simulation* [41], entre otros [42].

La modalidad colaborativa en los juegos serios ha sido investigada como herramienta de aprendizaje en el campo de la psicología, la informática y la educación [43]. Algunos ejemplos de estos desarrollos son: *Leoncito y sus amigos* [44], este es un juego diseñado principalmente para que los niños aprendan las vocales por medio de una serie de actividades en grupo, la temática del juego consiste en que cinco jugadores tendrán asignadas cada una de las vocales y deben de realizar una serie de ejercicios sobre el sonido o la grafía de la vocal asignada, cuando cada miembro complete con éxito el nivel ganara una herramienta individual la cual deberá compartir con los demás jugadores debido a que es necesaria para conseguir el objetivo General. Otro juego diseñado bajo esta modalidad es *Nutri-Galaxy* [43], cuyo objetivo es enseñar la función de la nutrición, la temática de

este consiste en que cuatro jugadores conformaran un grupo donde deberán dar el correcto tratamiento a un campesino cuya alimentación está basada en comida rápida y ningún ejercicio físico. Además de los juegos antes mencionados también se encontraron: Saveh [45], The Coco-game borad [46], Miboard [47], Sherlock dengue 8 [48], entre otros [49].

Las investigaciones realizadas con juegos serios colaborativos han permitido demostrar sus ventajas, las cuales consisten en que el proceso de aprendizaje mejora debido a que las personas construyen sus propios conocimientos a partir de que escuchan a los demás y socializan sus ideas [44, 43]. También permite que se desarrolle mejor disposición frente a los procesos de aprendizaje y se incrementa la motivación de las personas ocasionando que dediquen más tiempo a aprender. Igualmente ha presentado ventajas En el área empresarial donde permite que las empresas se acerquen a sus consumidores y conozcan sus necesidades, los entornos colaborativos en el área laboral son importantes debido a que permiten, que las personas trabajen en grupo y discutan sobre las posibles mejoras de los procesos [46].

La modalidad cooperativa ha sido implementada en ámbitos educativos y formativos con excelentes resultados, así lo demuestran en [50] donde se implementan juegos bajo esta modalidad en las clases de español, el objetivo de esta investigación es presentar como este tipo de actividades lúdicas en grupo fomentan la motivación, la atención, la memoria, el interés y otros aspectos en el aula de clases, además investigar sobre las preferencias de los alumnos acerca de la modalidad de juego. Mediante este experimento se concluye que el estudiante en un ambiente cooperativo mejora su autoestima debido a que estar en un grupo genera placer y satisfacción, también permite que cada persona demuestre sus habilidades y conocimientos frente a los demás. Se fomenta la responsabilidad individual y en grupo, se desarrollan habilidades de comunicaciones y compañerismo. Además se evidencia que mediante el trabajo cooperativo se cumplen los objetivos rápidamente y las respuestas mejoran al ser más precisas [50]. Algunos ejemplos de juegos serios cooperativos son: trivirec [51], Escape from Wilson island [52], ParticiPécs [53] , Urban planning board game [54] comFim [55], etc [56].

Por otra parte autores como [50] mediante investigaciones realizadas recomiendan que, los juegos serios sin importar la modalidad que implementen deben de utilizar premios con significado simbólico, debido a que está comprobado que el uso de recompensas económicas, objetos materiales o privilegios para motivar al jugador ocasiona que este se preocupe más por el premio que por el proceso de aprendizaje que contiene el juego. También recomiendan ajustar adecuadamente el tiempo de duración de las actividades, ya que estas deben de ser lo suficientemente extensas para atrapar al jugador, pero no se pueden exceder porque ocasionan frustración y aburrimiento. Por último el objetivo del juego debe de centrarse en el proceso y no en el resultado, de esta manera se quiere destacar que ganar o perder no es tan importante como aprender e ir mejorando con cada intento [50]. además según [57, 58], consideran que los juegos multijugador sin importar la modalidad que se use, representan una alternativa innovadora que permite unir a las personas y proporciona la oportunidad de ayudarse en grupo.

En los párrafos anteriores se demostró que cada modo de juego fortalece una gran variedad de destrezas sin importar el área de aplicación. Sin embargo, en la investigación realizada no se encontraron estudios que comparen las ventajas que podría tener cada modalidad de juego en el contexto de estimulación de la memoria.

## 1.2. Planteamiento del problema

Los juegos serios son herramientas de aprendizaje que poseen entornos pedagógicos, didácticos, autónomos, autosuficientes y reutilizables que permiten al usuario obtener conocimientos, potencializar capacidades y adquirir competencias de manera práctica [59]. Dentro de sus campos de aplicación se encuentra la rehabilitación cognitiva donde es considerada una herramienta de apoyo dirigida al tratamiento y entrenamiento de funciones cognitivas como: la atención, la percepción, el cálculo, el lenguaje y la memoria [60]. Estas tecnologías consisten en entornos virtuales y actividades similares al del mundo real como se presentan en [24]. También permite disminuir los costos de tratamiento, incrementar la accesibilidad del mismo, además poseen el ajuste dinámico de dificultad, feedback automático, utilidad práctica, entre otras ventajas [61] [62].

El desarrollo de juegos serios para la rehabilitación o estimulación de la memoria es una actividad compleja, que requiere procesos de diseño bien establecidos para lograr los objetivos propuestos [63]. Según [64], solamente el 5% de los juegos serios logran tener éxito, debido a que no motivan al paciente y por ende tienen un porcentaje bajo de aceptación. Algunos criterios que deben de cumplir este tipo de herramientas son: basar su contenido en un fundamento teórico de rehabilitación cognitiva, implementar entornos inmersivos y una interfaz humano-computadora clara, guías de manejo, orientar el tratamiento a actividades cotidianas, simplicidad, amigable con el usuario, seguir la evolución del paciente, entre otros [63]. Además debido al impacto que tienen los juegos serios en el tratamiento de la memoria se hace necesario garantizar la calidad de los mismos cumpliendo con: la adecuación funcional, eficiencia, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad [65].

Los juegos serios son herramientas en constante desarrollo que requieren de estudios clínicos para lograr una rehabilitación eficiente [60], si bien estas investigaciones se han centrado en estudiar los posibles beneficios que conlleva utilizar interfaces de realidad virtual o aumentada [66], las ventajas del ajuste dinámico de dificultad [67], que protocolo de estimulación es más eficiente, que factores inciden en el rendimiento de los pacientes [68], etc. Estas investigaciones han olvidado componentes importantes como la interacción social de los pacientes en el tratamiento. Según los resultados encontrados en [6] [17], se evidencia que el tratamiento en grupo mejora el estado de ánimo y el bienestar del paciente, dado que estos expresan cómo las actividades lúdicas y grupales les permiten socializar y crear relaciones interpersonales, por lo cual se ven motivados a participar en cada sesión del tratamiento [17].

Actualmente existen diferentes aplicaciones utilizadas para el tratamiento de la memoria como: Lumosity [29], ADHD Trainer [69], EyeRemember [70], entre otras [30] [24] [23] [31]

[71] [72]. Sin embargo estas aplicaciones han sido diseñadas bajo la modalidad de jugador simple, ignorando que: la interacción social permite crear una relación entre los recuerdos y las emociones [73], gran parte de la información que almacenamos es de contenido emocional [74], el proceso de memorización es selectivo y requiere de la codificación del recuerdo [75] y por ultimo las relaciones interpersonales en el tratamiento incentivan al aprendizaje colaborativo [76].

En complemento a lo anterior, se piensa que los juegos serios pueden abordar el componente grupal. En [32], exponen que dentro de las características que contienen los juegos serios, las ventajas y desventajas de los entornos multijugador es poco investigado y puede ser influyente en la rehabilitación cognitiva debido a que el uso de estas herramientas bajo esta modalidad permite que, el usuario desarrolle vínculos afectivos con las personas que lo acompañan [27].

Este trabajo de grado propone diseñar un juego serio multijugador para estimular la memoria, además pretende analizar qué ventajas y desventajas conlleva realizar el tratamiento de la memoria en un entorno multijugador, lo anterior para dar respuesta a la pregunta ¿Qué ventajas y desventajas conlleva el uso de un entorno multijugador para la estimulación de la memoria, con respecto a aplicaciones de jugador único?

### 1.3. Objetivos

A continuación se presentara los objetivos de este trabajo de grado.

- **Objetivo general**

Proponer una herramienta software para la estimulación de la memoria en un contexto de serious game multijugador.

- **Objetivos específicos**

- Desarrollar un juego serio multijugador para la estimulación de la memoria, consistente en un vehículo que debe llegar a una serie de lugares definidos en un entorno de realidad virtual inmersiva.
- Generar un entorno 3D en él que se utilice un vehículo virtual para evaluar la capacidad que posee el usuario para memorizar lugares, objetos y personas.
- Determinar experimentalmente el progreso de la memoria en un contexto multijugador en un grupo de usuarios sanos, mediante la estimulación de la misma con la herramienta desarrollada.

### 1.4. Estructura de la monografía

Esta monografía se divide en cinco capítulos, el primero consiste en presentar las bases de la investigación, mediante el estado del arte y el planteamiento del problema se identifica la temática en cuestión, se realiza una revisión de los estudios previos para contextualizar el problema. Además de justifica y formula la investigación. En el segundo capítulo se

expone la metodología, diseños y el desarrollo del juego serio, lo que permite establecer los requerimientos y características del aplicativo. En el tercer capítulo se presentan los criterios de diseño y argumentos bajo los cuales se realizaron todos los componentes gráficos del juego serio, por ejemplo: el color, formas, pistas, entornos, letras, objetos, etc. En el cuarto capítulo se desarrolla el experimento de la investigación, donde se elige una población que utilizara el juego serio para estimular la memoria visual, se obtendrán resultados y se analizarán los mismos. Para finalizar, en el quinto capítulo se presentan las conclusiones de la investigación y los trabajos futuros.

# Capítulo 2

## Desarrollo del juego serio

En este capítulo se presentaran las herramientas utilizadas para el diseño y desarrollo del aplicativo, estas permitieron establecer las funcionalidades, requerimientos, arquitectura, interfaces y flujos de información necesarios para el funcionamiento de un juego serio multijugador, cuyo objetivo principal es estimular la memoria visual en un entorno de realidad virtual inmersiva.

### 2.1. Metodología.

La metodología es un conjunto de técnicas empleadas para el desarrollo de cualquier tipo de proyecto, en este caso el diseño de un *software* informático, con el uso de estas técnicas se busca realizar procesos eficientes que reduzcan el nivel de dificultad. Actualmente las metodologías se dividen en dos grandes grupos: Tradicional y Ágil, los cuales tienen características opuestas y se emplean en contextos diferentes.

#### **Metodologías tradicionales**

Las metodologías tradicionales se caracterizan por un desarrollo total del producto, se centra en todo el control del proceso mediante una definición estricta de los roles, actividades, artefactos, herramientas para el modelado y documentación detallada, por lo tanto su adaptación a los cambios presenta gran dificultad y representa un alto costo, haciéndolos no aptos para entornos donde los requisitos pueden cambiar con el tiempo [77].

#### **Metodologías ágiles**

Estas surgen como consecuencia a las metodologías tradicionales, ya que estas permiten realizar modificaciones e incluir la adaptabilidad a cada proyecto, incluyen una comunicación constante y asertiva con el cliente para tener una retroalimentación con este, lo cual permite hacer un desarrollo eficaz. Además debido a la sencillez de su aplicación permite obtener productos de calidad a corto plazo, siendo un proceso incremental (entregas frecuentes con ciclos rápidos).

A continuación en la tabla 2.1 se presentan las diferencias entre las metodologías expuestas [78].

Metodología Tradicional	Metodología ágil
Predictivos	Adaptativos
Orientados a procesos	Orientado a personas
Se concibe como un proyecto	Un proyecto se divide en varios proyectos pequeños
Poca comunicación con el cliente	Comunicación constante con el cliente
Entrega del software al terminar el desarrollo	Varias entregas del software durante el desarrollo
Documentación extensa	Poca documentación

Tabla 2.1: Diferencias entre las metodologías ágiles y tradicionales .

Por lo mencionado anteriormente se decidió implementar un tipo de metodología ágil, debido a que ofrece una respuesta rápida a cambios de requisitos durante el desarrollo del *software*, además permite una comunicación activa con el cliente, mediante reuniones entre este y los desarrolladores, se logra establecer un ambiente colaborativo en el desarrollo, proporciona simplificación en la sobrecarga de procesos ya que se utilizan interacciones cortas, todo esto permite entregar un producto de acuerdo a los requerimientos.

Dentro de las metodologías ágiles encontramos a: *extreme programming (XP)*, *kanban* y *scrum*, a continuación se presentan algunas de las características de cada metodología [79]:

### ***Extreme Programming (XP)***

Esta metodología se caracteriza por incluir: *planning game* (define la fecha de entrega y distribuye las responsabilidades entre el cliente y los desarrolladores), pequeñas entregas, diseño simple, programación en pareja, pruebas, integración continua, ritmo de trabajo continuo, comunicación constante con el cliente, visión clara del sistema y buenas practicas de programación.

### ***Kanban***

Se caracteriza por centrarse en las personas que componen el equipo (de ellas depende el éxito del proyecto) y la reducción al máximo del número de artefactos producidos, calidad garantizada y la flexibilidad para priorizar tareas según las necesidades del momento.

### ***Scrum***

Implementada en proyectos con un rápido cambio de requisitos, donde el desarrollo de *software* se realiza mediante iteraciones, denominadas *sprints*, con una duración de treinta días. El resultado de cada *sprint* se muestra al cliente con el cual se organizan diferentes reuniones a lo largo de todo el proyecto, haciéndolo parte del equipo de trabajo.

Con el análisis de los tipos de metodologías ágiles realizado anteriormente se decidió implementar la metodología *scrum*, por lo tanto a continuación se profundiza las características de esta.

## **2.1.1. Metodología scrum**

En la actualidad la metodología más utilizada para el desarrollo de software es *scrum*, debido a las ventajas que presenta frente a las demás metodologías. Algunas de estas son: el desarrollo ágil que permite priorizar actividades que aportan valor al producto, esta metodología hace énfasis en las necesidades del usuario y cuenta con frecuente retroalimentación del cliente permitiendo identificar posibles cambios o funcionalidades esperadas, incita el trabajo en equipo, maneja una organización sencilla en el grupo de trabajo y promueve el

desarrollo de software flexible.

*Scrum* se caracteriza por el uso de un marco de trabajo iterativo e incremental, utilizado para el desarrollo de cualquier tipo de proyectos, productos o aplicaciones. La estructura de esta metodología según el autor [79], consiste en ciclos de trabajo denominados *sprints*, estos son de duración fija y suceden uno detrás de otro.

Al principio de cada *sprint*, un equipo multifuncional determina los requisitos del cliente en una lista con prioridades y un tiempo estipulado para la entrega, al final el equipo se reúne a los clientes para revisar y enseñar los avances del proyecto, para recibir comentarios y observaciones que se incorporan al siguiente *sprint*[80].

Esta metodología promueve la colaboración eficaz de los equipos que intervienen en el proyecto, a partir de un modelo de trabajo en equipo basado en la flexibilidad, creatividad y productividad. Además se decide cuál es la persona mas adecuada para desempeñar cada trabajo. A continuación se explican los roles y sus funciones [81]:

- ***Scrum master:*** Es la persona encargada de garantizar que las etapas de Scrum se cumplan y sean entendidas.
- ***Scrum team:*** Son las personas que forman el grupo de desarrollo, su función es entregar el producto.
- ***Product owner:*** Este rol representa el cliente final del producto, es responsable de escribir las historias de usuario, priorizarlas y organizarlas en la lista de producto.

Por otra parte en la metodología *scrum* se utilizan los eventos o actividades de equipo, con el fin de crear regularidad y ahorrar tiempo de trabajo evitando reuniones innecesarias. A continuación se explican eventos y sus definiciones [81]:

- ***Sprints:*** Es la unidad de tiempo empleada para el cumplimiento de una tarea, lo recomendable es que un *sprint* dure entre un y seis semanas, dependiendo de la complejidad y la prioridad de la tarea.
- ***Daily Standup:*** Son reuniones diarias para coordinar las tareas del equipo.
- ***Entregables:*** Forman la parte final del proyecto en el que se debe presentar los documentos según los requerimientos del *Product Owner*. Una vez que estos son aceptados se reinicia el ciclo. Esto se debe a que *scrum* es una metodología pensada para mejoras constantes.

Por ultimo esta metodología implementa artefactos o herramientas. Son técnicas empleadas para comprender el desarrollo del producto y organizar las actividades realizadas y pendientes. A continuación se explican las herramientas y sus definiciones [81]:

- ***Product Backlog:*** Es la lista de tareas necesarias para el desarrollo del producto, estas son responsabilidad del *product Owner*.
- ***Sprint Backlog:*** Es el listado de tareas del *product Backlog* que se utilizaran en un *sprint*.
- ***Incremento:*** Es la lista de actividades del *product Backlog* que se entregaron en el Sprint.

La figura 2.1, muestra de forma gráfica el marco de trabajo propuesto por la metodología Scrum [82].

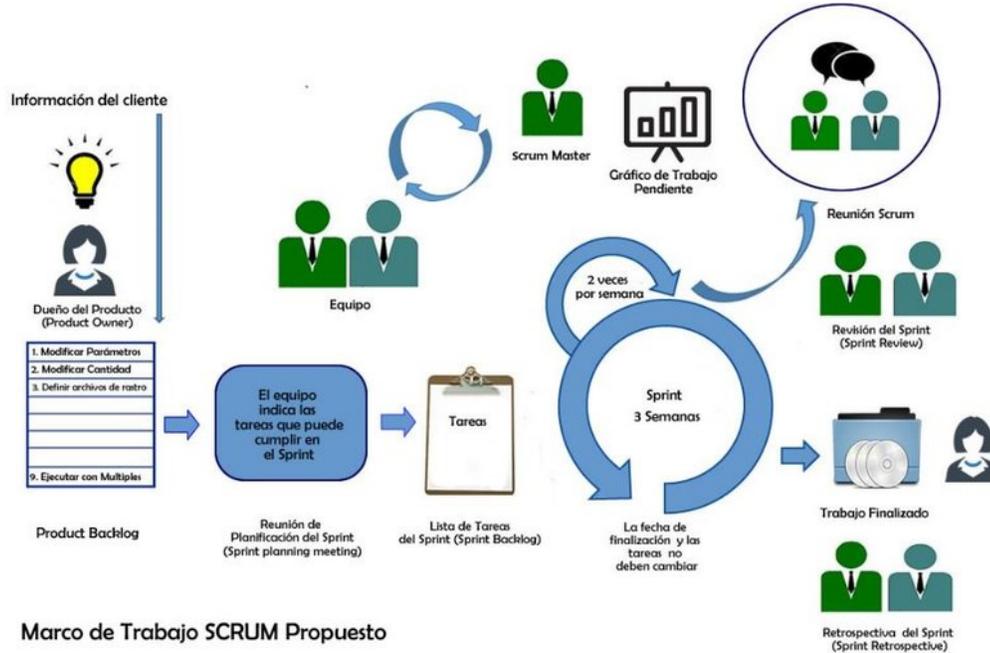


Figura 2.1: Proceso metodología Scrum

## 2.2. Tabla de requerimientos

En la actualidad la primera fase de un proyecto consiste en definir sus requerimientos y con el transcurso del tiempo se evita realizar cambios en estos, debido al incremento en tiempo y costos que conlleva. La metodología *Scrum* es flexible ante los requerimientos del cliente presentado avances del proyecto cada semana, organizados funcionalidades por prioridades y desarrollando *software* flexible. Construir y mantener actualizados los requerimientos del proyecto es una tarea difícil, sin embargo *scrum* implementa las historias de usuario, estas consisten en la descripción de cada uno de los requisitos del cliente, bajo una estructura donde se explique a detalle para quien se va a desarrollar cada funcionalidad del *software* es decir asignar un rol, la funcionalidad que se implementará y los resultados esperados por el cliente o los criterios de satisfacción.

Id	Descripción	Criterios de satisfacción
1.	Yo como fisioterapeuta quiero que dentro del juego se realicen actividades que involucren la atención, percepción, pensamiento y memoria, para que el usuario desarrolle la flexibilidad cognitiva.	1. Imágenes distribuidas en la pista para memorizar. 2. Juego de carreras. 3. Preguntas sobre imágenes que observe durante el recorrido. 4. Conducir el vehículo por medio de una carretera con rectas y curvas. 5. Respuestas de selección múltiple.
2.	Yo como fisioterapeuta quiero que dentro del juego el usuario tenga que plantear una estrategia, para que así él desarrolle la capacidad de resolución de problemas.	1. Diferentes pistas, modalidades y objetivos. 2. El puntaje total dependerá del tiempo de respuestas, número de aciertos y tiempo en la pista. 3. Imágenes aleatorias con diferentes detalles por memorizar.
3.	Yo como fisioterapeuta quiero que dentro del juego el usuario tenga diferentes modalidades de juego multijugador para incentivarlo a socializar.	1. Modalidad individual. 2. Modalidad competitiva. 3. Modalidad cooperativa. 4. Modalidad colaborativa 5. Diferentes roles (copiloto, piloto).
4.	Yo como fisioterapeuta quiero que dentro del juego el usuario tenga que almacenar, organizar y recuperar información para estimular los procesos básicos de la memoria.	1. Se presentan 10 imágenes diferentes por partida, las cuales se deben de memorizar. 2. Preguntas al final de la partida. 3. Preguntas sobre el orden de las imágenes.
5.	Yo como fisioterapeuta quiero que el juego contenga distractores internos y externos para que el usuario registre información a través de sus sentidos.	1. Sonido de aceleración, freno, pitos y choques. 2. Pedales de freno, aceleración y reversa mecánicos. 3. Entorno virtual con múltiples formas y volante eléctrico.

Id	Descripción	Criterios de satisfacción
6.	Yo como fisioterapeuta quiero que el juego serio tenga diferentes imágenes y preguntas, para aportar dificultad al juego.	1. Código con lista de imágenes en la aplicación.
		2. Código con lista de respuestas de selección múltiple.
		3. Código con lista de preguntas en la aplicación
7.	Yo como desarrollador quiero aleatoriedad en las preguntas y las imágenes, para aportar dificultad al juego.	1. Código de aleatoriedad para preguntas, imágenes y respuestas
8.	Yo como desarrollador quiero que el juego serio sea lo más real posible para aportar utilidad práctica al sistema.	1. realidad virtual.
		2. cabina de conducción.
		3. Pistas basadas en el mundo real.

Tabla 2.2: Tabla de requerimientos.

### 2.3. Diseño del juego serio

A partir de los requerimientos planteados en la sección anterior, surge la necesidad de realizar un documentó donde se defina el contenido del juego serio. A continuación se presenta la estructura de diseño implementada.

- a) **Título del juego serio:** Este aplicativo se denomino *Brain race* debido a que su objetivo es entrenar habilidades cognitivas como la memoria, la atención, la percepción, el razonamiento a través de la conducción de un vehículo virtual.
- b) **Descripción:** es un juego serio de carreras multijugador diseñado para personas que buscan desafiar su memoria. El juego esta compuesto por cuatro modalidades:
  - **Modalidad individual:** el jugador debe conducir por una pista empleando el menor tiempo posible y memorizando las imágenes que observa durante el recorrido, una vez terminado el recorrido debe contestar una serie de preguntas relacionadas con el contenido de cada imagen.
  - **Modalidad competitiva:** dos jugadores deben conducir por una pista empleando el menor tiempo posible y memorizando las imágenes que observan durante el recorrido, una vez terminado el recorrido cada uno debe de contestar una serie de preguntas relacionadas con el contenido de cada imagen.
  - **Modalidad cooperativa:** dos jugadores deben conducir por una pista empleando el menor tiempo posible y memorizando las imágenes que observan durante el recorrido, una vez terminado el recorrido los jugadores deben contestar juntos una serie de preguntas relacionadas con el contenido de cada imagen.
  - **Modalidad colaborativa:** un jugador es el encargado de conducir el vehículo mientras el otro observa el mapa y guía al conductor por la pista, indicando donde aparecen las imágenes que deben de memorizar. El conductor debe atravesar cada uno de las estructuras donde están suspendidas las imágenes, al atravesar la novena imagen, el copiloto deberá contestar las preguntas relacionadas con el contenido de cada imagen.

- c) **Mecánicas de juego:** Para iniciar el juego se debe ingresar en la aplicación, elegir la modalidad, en caso de ser multijugador se debe esperar al segundo jugador. Una vez dentro de la partida se instala el celular en la gafas de realidad virtual y se debe acelerar para iniciar el juego, a continuación el jugador comienza a conducir, para lo cual se tienen dos formas de manejo de acuerdo a la modalidad, en ambos casos (individual o multijugador) el conductor debe manipular el acelerador, freno y volante para controlar los movimientos. En la modalidad individual se hace uso de una cabina de conducción en la cual el jugador al pisar el pedal izquierdo acelera el vehículo y pisando el pedal derecho frena, si se mantiene presionado este último el vehículo retrocederá, además deben girar el volante el cual cuenta con un ángulo de 180°, hacia la dirección donde se quiere desplazar. En las demás modalidades se cuenta con un control tipo gamer donde se tienen dos joysticks, el primero está programado para acelerar, frenar y retroceder y el segundo para girar el volante. Una vez finalizada la carrera se debe retirar el celular de las gafas de realidad virtual y contestar las preguntas seleccionando una de las cuatro respuestas que aparecen en la interfaz.
- d) **Temática y género:** teniendo en cuenta las categorías mencionadas en [83], este es un juego serio, de carreras, multijugador en realidad virtual.
- e) **Complejidad del proyecto:** La realidad virtual es la creación de un entorno ficticio con apariencia totalmente real, mediante el uso de dispositivo como gafas o cascos de realidad virtual las personas tienen la sensación de estar dentro del juego. Sin embargo algunas personas al utilizar estos dispositivos presentan mareo, dolor de cabeza y fatiga visual. Por otra parte el uso de los pedales y el volante es considerado como una dificultad para las personas que no manejan o no poseen la coordinación motriz suficiente para manipularlos.
- f) **Ideas y referencias:** En la actualidad se realizan investigaciones dedicadas al diseño de juegos serios en realidad virtual con el objetivo de evaluar o estimular la memoria, a partir de actividades cotidianas que aporten utilidad práctica a los métodos tradicionales, algunos de estos son:
- La investigación realizada por [84], presentan el diseño de un entorno virtual para medir la memoria, el cual consiste en un recorrido de 15 minutos por una ciudad virtual que contiene personas de diferentes géneros y edades, mercados, autos, animales, los cuales deben ser memorizados. En los 15 minutos que dura la prueba el usuario debe desplazarse de un lugar a otro y memorizar no solamente los objetos que encuentra sino el lugar donde los encontró. Algunos de los aspectos que mide este test son: número de correctas e incorrectas, tiempo requerido para recorrer cada zona y tiempo total de la prueba.
  - En [85], exponen el diseño de una ciudad virtual a partir de 1400 fotografías del interior y exterior de centros comerciales en Nueva Zelanda. En esta prueba el participante debe desplazarse en la calle y entrar en los diferentes centros comerciales para hacer 10 mandados que son asignados en una lista. La puntuación en esta prueba dependerá del tiempo y el número de objetos recogidos.
  - En [86], exponen el desarrollo de una ciudad virtual que debe ser recorrida desde un auto donde el usuario puede elegir entre dos roles, el primero consiste en ser el conductor del auto y el segundo en ser el pasajero, durante el recorrido el usuario encontrará diferentes elementos los cuales debe memorizar, acompañado de su ubicación y cuando apareció.

Para el diseño del juego serio *brain race* se tomó como referencia algunas características mencionadas en las investigaciones anteriores como los entornos virtuales basados en escenarios de

la vida real, a partir de estos ejemplos se diseñaron pistas de carreras sobre un bosque, una ciudad y un paisaje nublado, donde los jugadores conducen en un entorno cotidiano y entretenido. Otra característica es el protocolo de estimulación de la memoria el cual consiste en memorizar objetos, personas, imágenes o animales, con estas referencias se diseñó un conjunto de metas distribuidas a lo largo de todas las pistas, en la parte superior de estas estructuras aparece una imagen de diferentes temáticas como: colores, comidas, animales, objetos, etc. Además fue tomado como referencia el tipo de preguntas y respuestas implementado en cada una de estas investigaciones, lo cual se aplicó en el diseño de estas y en la implementación de respuestas de selección múltiple. A su vez permitieron identificar que variables se pueden medir a lo largo del juego como el tiempo en recorrer la pista y en responder las preguntas, además el número de respuestas correctas e incorrectas.

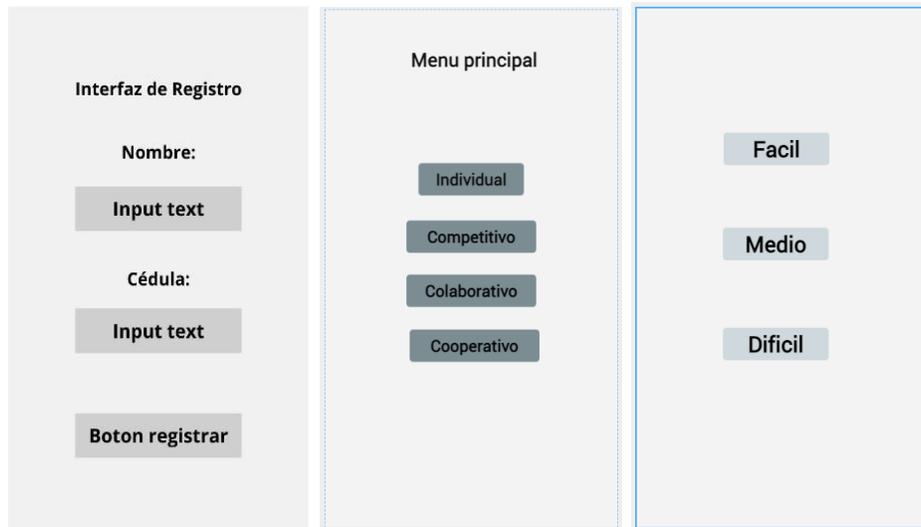
Adicionalmente se encuentra que la memorización de imágenes y la evaluación de la retención de información de estas, es utilizado por diferentes test tradicionales para evaluar la memoria, algunos de estos son:

- **Batería de eficiencia mnésica de signoret:** este test se divide en dos partes con el fin de evaluar el estado de la memoria, la primera es la evaluación del recuerdo diferido y la segunda es la evaluación del aprendizaje serial, para lo cual se hace la presentación y repetición de 24 figuras mediante ejercicios de repetición donde se evalúa el recuerdo inmediato del diseño, el recuerdo de la secuencia de figuras, el recuerdo diferido de los diseños presentados [87].
- **Test de copia de la figura compleja de rey:** este test consiste en copiar y luego reproducir de forma inmediata un gráfico geométrico complejo que evalúa la capacidad perceptiva, organizativa y analítica del paciente. El evaluador mediante el análisis de la forma en que el sujeto reproduce un dibujo evalúa el grado y la fidelidad de su memoria visual, también permite comparar entre la capacidad viso-constructiva y su memoria visual [88].
- **Test de retención visual de Benton:** consiste en presentar una serie de dibujos abstractos al paciente, los cuales debe de memorizar y reproducir una vez desaparezcan de la vista del usuario. En esta prueba se evalúa el número de reproducciones correctas e incorrectas [89].

g) **Mockups:** Una vez definido el contenido del juego es necesario hacer una aproximación del diseño de la interfaz, con el fin de mostrar al cliente las ideas y funcionalidades, esto permite ahorrar tiempo en caso que el cliente no se sienta satisfecho con la estructura propuesta para la aplicación, permite visualizar como se vería en realidad la interfaz del juego y adaptarla a las necesidades del cliente, a continuación se presenta la propuesta inicial del juego serio:

Cuando el usuario ingrese en la aplicación encontrará: Una pestaña de registro compuesta por dos entradas de texto para ingresar el nombre y cédula del jugador, también cuenta con un botón para guardar esta información y seguir a la siguiente pestaña, en la figura a de 2.2 se muestra la estructura de esta interfaz. Después de registrarse el usuario ingresa al menú principal compuesto por cuatro botones para elegir cada una de las diferentes modalidades: individual, competitiva, cooperativa y colaborativa, en la figura b de 2.2 se muestra la estructura de esta interfaz. Si en la anterior interfaz el usuario elige la modalidad individual, encontrará la pestaña para elegir el nivel de dificultad compuesta por tres botones: fácil, medio y difícil, en la figura c de 2.2 se muestra el diseño de esta interfaz; después de elegir el nivel de dificultad, el usuario ingresa en la partida individual donde debe conducir un vehículo virtual. Si el usuario elige la modalidad competitiva, colaborativa o cooperativa ingresa en la interfaz de sala de espera la cual esta compuesta por dos botones, en la figura d de 2.2 se muestra el

diseño de esta pestaña. Estos botones sirven para elegir quien sera el jugador uno y quien el jugador dos, es decir quien es el encargado de crear la partida y quien ingresara a ella. Cuando se conecten los dos jugadores a la partida multijugador, comenzara el juego. Cada vez que se termine la carrera en cualquiera de las modalidades los jugadores ingresan en la pestaña de preguntas, compuesta por un cuadro de texto donde aparecen las preguntas y cuatro botones con las diferentes respuestas, en la figura e de 2.2 se muestra el diseño de esta interfaz. Cuando se contesten todas las preguntas se terminara el juego y se mostrará la interfaz de resultados compuesta por tres cuadros de texto donde aparece el tiempo empleado por el jugador en recorrer la pista, contestar las preguntas y el numero de respuestas correctas obtenidas, en la figura f de 2.2 se muestra el diseño de esta interfaz.



(a) Pestaña de registro.

(b) Menú principal.

(c) Menú de dificultad para la modalidad individual.



(d) Sala de espera para modalidades multijugador.

(e) Interfaz de preguntas.

(f) Interfaz de resultados.

Figura 2.2: Mockups (fuente propia).

h) **Diagramas UML** el objetivo estos diagramas es presentar diferentes perspectivas de un mismo sistema, en este caso del juego serio propuesto, con estos se busca realizar un modelo o representación simplificada de la realidad; en este se describen las funcionalidades del sistema, pero no su implementación. A continuación se presentaran los diagramas UML, utilizados en el diseño de esta aplicación:

- **Diagrama de casos de uso:** según el autor [90], este tipo de diagramas describen el comportamiento del sistema desde el punto de vista del usuario, permite visualizar los diferentes actores que existen y como interactúan con el sistema, también define los límites del sistema y las relaciones entre este y el entorno.

A continuación se ilustra gráficamente como los actores externos interactúan con el sistema:

Al ingresar en la aplicación, aparece un menú con cuatro opciones de modalidad: individual, colaborativa, cooperativa y competitiva, en la figura 2.3 se muestra el diagrama de casos de uso correspondiente al jugador, así mismo, en la tabla 2.3 se describe la narrativa del caso de uso.

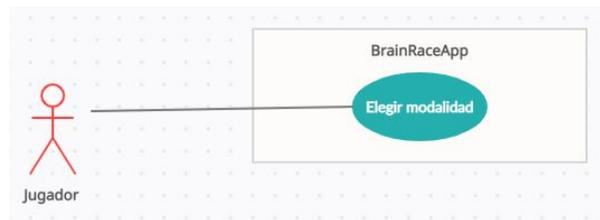


Figura 2.3: Primer diagrama de casos de uso.

<b>Actor:</b>	<b>Jugador</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>Permite al usuario elegir entre las modalidades individual, cooperativa, colaborativa y competitiva.</b>

Tabla 2.3: Narrativa primer caso de uso “elegir modalidad”.

Cuando el usuario ingrese en una modalidad multijugador, aparece una interfaz donde debe crear la partida o unirse a una ya creada, en las tablas 2.4 y 2.5 se describe la narrativa del segundo diagrama de casos de uso, así mismo, en la figura 2.4 se muestra el diagrama correspondiente.

<b>Actor:</b>	<b>Jugador</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>Permite al jugador crear una partida nueva en cualquier modalidad multijugador.</b>

Tabla 2.4: Narrativa caso de uso “crear partida”.

<b>Actor:</b>	<b>Jugador</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>Permite al jugador ingresar a una partida ya creada por otro usuario.</b>

Tabla 2.5: Narrativa caso de uso “buscar partida“.

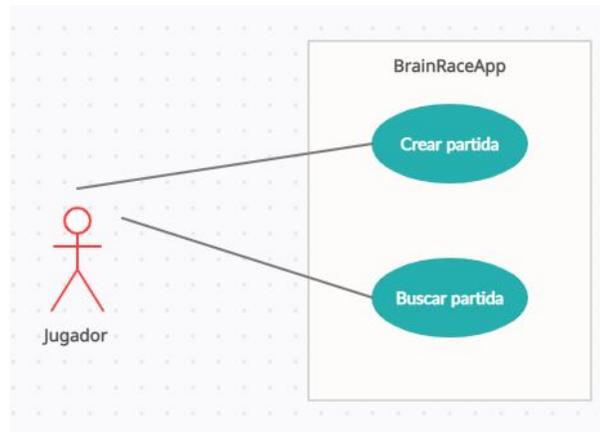


Figura 2.4: Segundo diagrama de casos de uso

En las modalidades multijugador es necesario que se conecten dos jugadores, mientras esto no ocurra no se puede iniciar la partida. en las tablas 2.6, 2.7 y 2.8 se describe la narrativa del tercer diagrama de casos de uso, así mismo, en la figura 2.6 se muestra el diagrama correspondiente.

<b>Actor:</b>	<b>Jugador 1, jugador 2</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>El jugador ingresa en la partida y se dispone para jugar.</b>

Tabla 2.6: Narrativa caso de uso “jugador listo“.

<b>Actor:</b>	<b>Jugador 1, jugador 2</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>El jugador crea la partida y espera a que se conecte el otro jugador.</b>

Tabla 2.7: Narrativa caso de uso “esperar jugador“.

<b>Actor:</b>	<b>Jugador 1, jugador 2</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>Cuando se conecten los dos jugadores comenzara el juego</b>

Tabla 2.8: Narrativa caso de uso “empezar partida”.

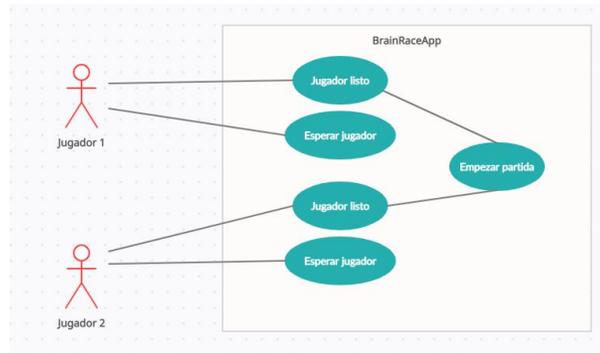


Figura 2.5: Tercer diagrama de casos de uso.

Los jugadores dentro del juego deben de conducir un vehículo y al final del recorrido contestar unas preguntas sobre las imágenes que observaron en el camino. Cada vez que los jugadores finalicen una partida la aplicación genera datos del desempeño de los jugadores, el programador se encarga de almacenar estos datos para su posterior análisis, en tablas 2.9, 2.10 y 2.11 se describe la narrativa del tercer diagrama de casos de uso, así mismo, en la figura 2.6 se muestra el diagrama correspondiente.

<b>Actor:</b>	<b>usuario o jugador</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>el jugador mediante el uso del control o una cabina de conducción debe de controlar los movimientos del vehículo.</b>

Tabla 2.9: Narrativa caso de uso “conducir”.

<b>Actor:</b>	<b>usuario o jugador</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>el jugador en la interfaz de preguntas y respuestas tiene cuatro botones para contestar la respuesta planteada sobre el contenido de la imagen vista en el recorrido del juego.</b>

Tabla 2.10: Narrativa caso de uso “contestar preguntas”.

<b>Actor:</b>	<b>Programador:</b>
<b>Prioridad:</b>	<b>Media</b>
<b>Descripción:</b>	<b>El programador almacena los datos generados por el juego</b>

Tabla 2.11: Narrativa caso de uso “almacenar datos“.

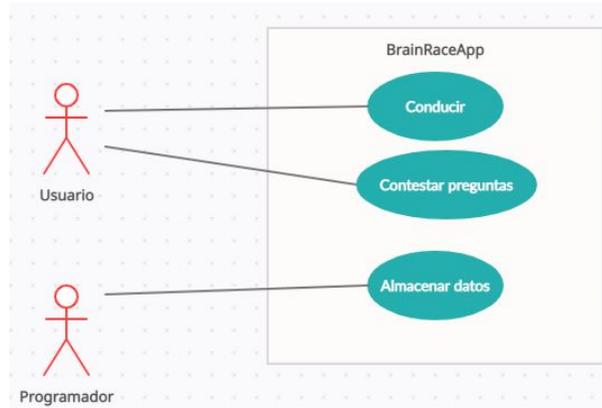


Figura 2.6: Tercer diagrama de casos de uso.

- Diagrama de despliegue:** en este diagrama se muestra la arquitectura física de un sistema. Se representan los equipos y los dispositivos, también se visualiza las interconexiones y el software que se encontrará en cada máquina [91]. Para el desarrollo del juego serio es necesario utilizar la librería de *Mirror Networking Multiplayer* que permite realizar la configuración multijugador, las librerías de *VrmodeController* para el diseño del entorno en realidad virtual, la librería de *ArduinoBluetoothAPI* y *System.IO* que permite conectar los componentes electrónicos de arduino que componen la cabina de conducción. En la figura 2.7 se presenta el diagrama de despliegue diseñado para este juego serio.

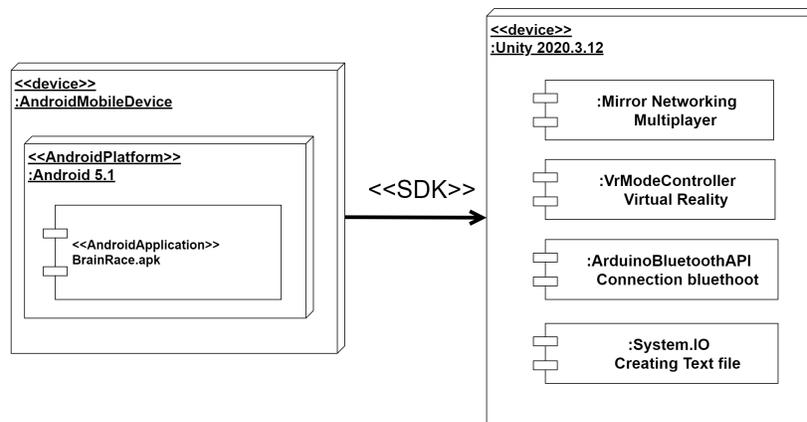


Figura 2.7: Diagrama de despliegue.

- Diagrama de flujo:** este diagrama es utilizado para hacer una representación de algoritmos o procesos, cuenta con una amplia gama de aplicaciones en diseño técnico, comunicación e informes comerciales. Por lo general, algunos cuadros se utilizan para representar varios tipos de operaciones. Los pasos se escriben en los cuadros y luego se conectan mediante líneas con flechas para indicar el orden de ejecución. Estos gráficos son intuitivos y fáciles de entender[91]. en la figura 2.8 se presenta el diagrama de flujo diseñado para esta juego serio:

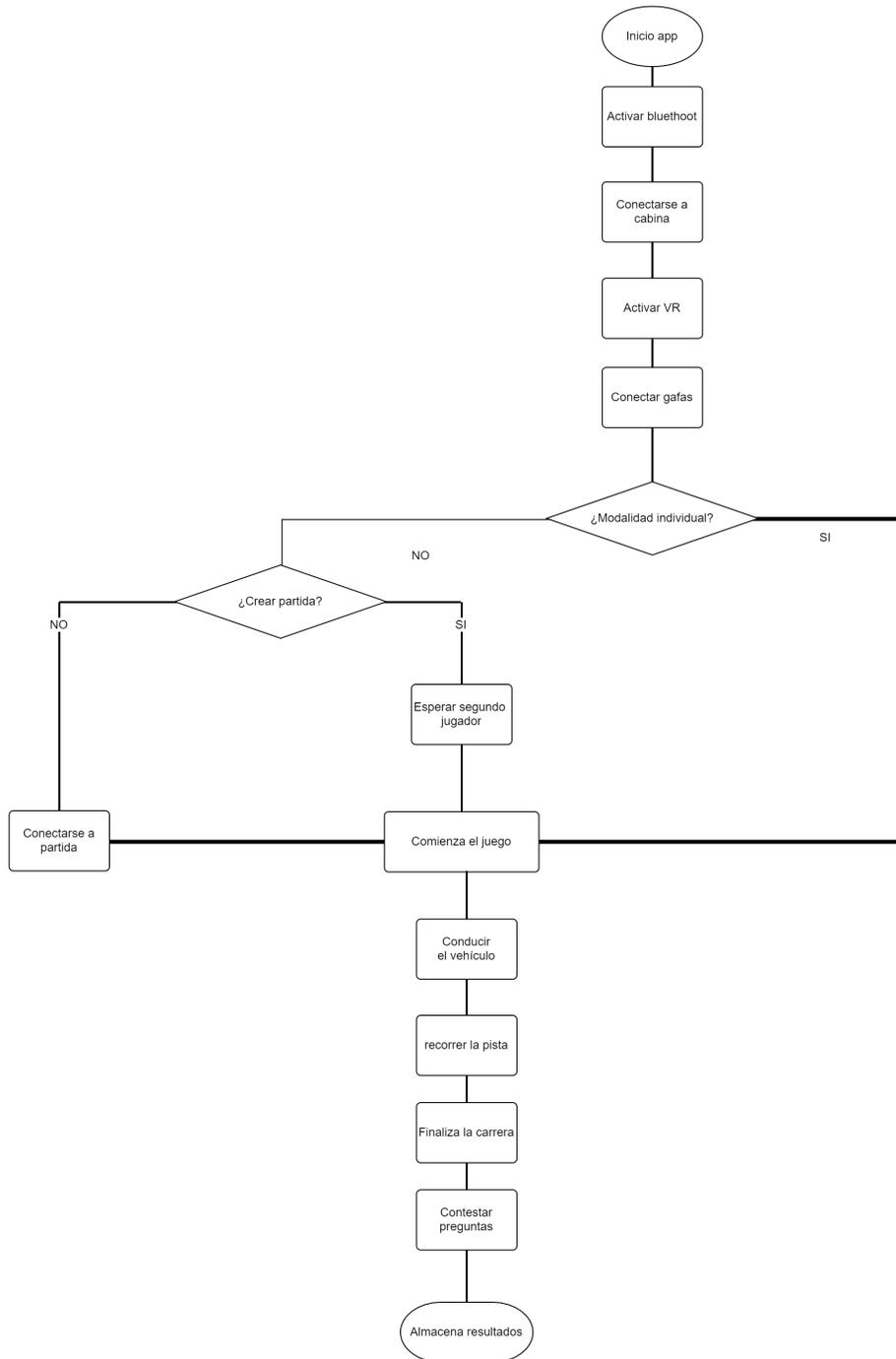


Figura 2.8: Diagrama de flujo.

i) **Descripción del software desarrollado** el diseño del juego serio se realizó en el motor gráfico de *Unity* debido a que permite desarrollar para diferentes plataformas, además cuenta con amplia documentación, foros y comunidades donde explican técnicas y métodos para el desarrollo de todo tipo de videojuegos. A continuación se explica como se desarrolló la configuración multijugador y de realidad virtual en el juego.

- **Configuración multijugador:** actualmente *Unity* no cuenta con una interfaz de programación completa y vigente que permita el desarrollo de juegos multijugador, por ende surge la necesidad de buscar un paquete que permita esta configuración. Se realiza una búsqueda en foros y artículos donde se hayan desarrollado juegos de este tipo, encontrando que en [92] y [93] utilizan el paquete de *mirror* debido a su implementación es concisa y entendible.

*Mirror Networking* usa un diseño donde el cliente y el servidor ejecutan el mismo código, además la comunicación se lleva a cabo por medio de etiquetas. Esta forma de implementación permite simplificar el desarrollo, por esto es ampliamente utilizada en la creación de juegos multijugador. A continuación, se explican las funcionalidades de *mirror* y como permite la interacción entre dos jugadores.

- **Mirror Framework:** Una vez creado el proyecto en *Unity* se debe de instalar *mirror*, para ello se ingresa en la *Asset store* y se descarga con todas sus librerías. *mirror* cuenta con diferentes componentes que se deben de ir añadiendo en los objetos de *Unity* según sus necesidades. A continuación en la figura 2.9 se muestra la librería de *mirror* en la *Asset store*.

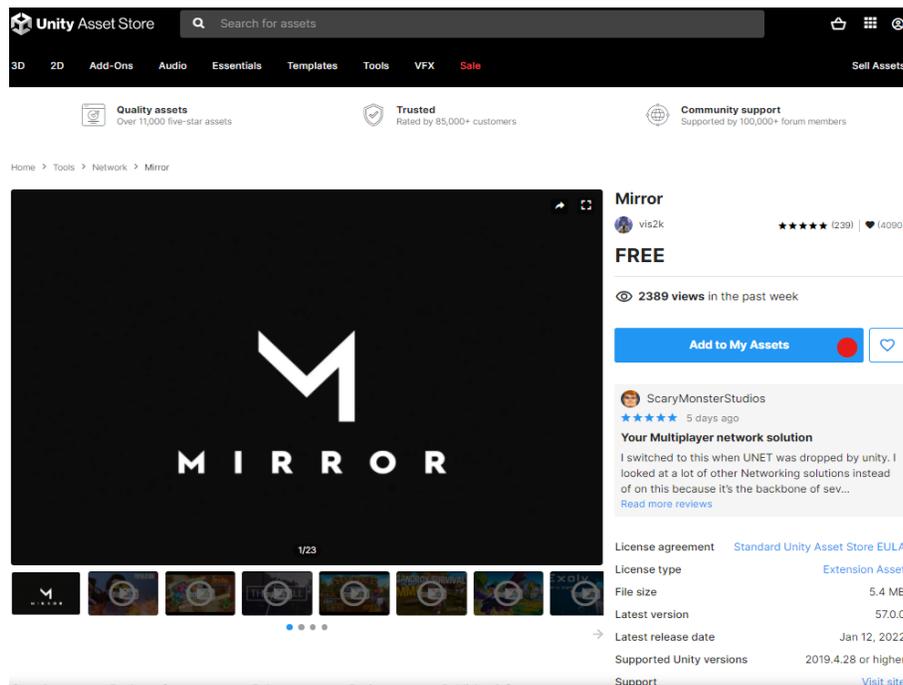


Figura 2.9: Librería de mirror en Unity.

- **Network Manager:** su función consiste en administrar los aspectos de red como: el puerto de transporte, la dirección de red donde deben conectarse los clientes, el número máximo de conexiones permitidas. También permite asignar cual será el objeto que representará al jugador. A continuación en la figura 2.10 se muestra este componente.

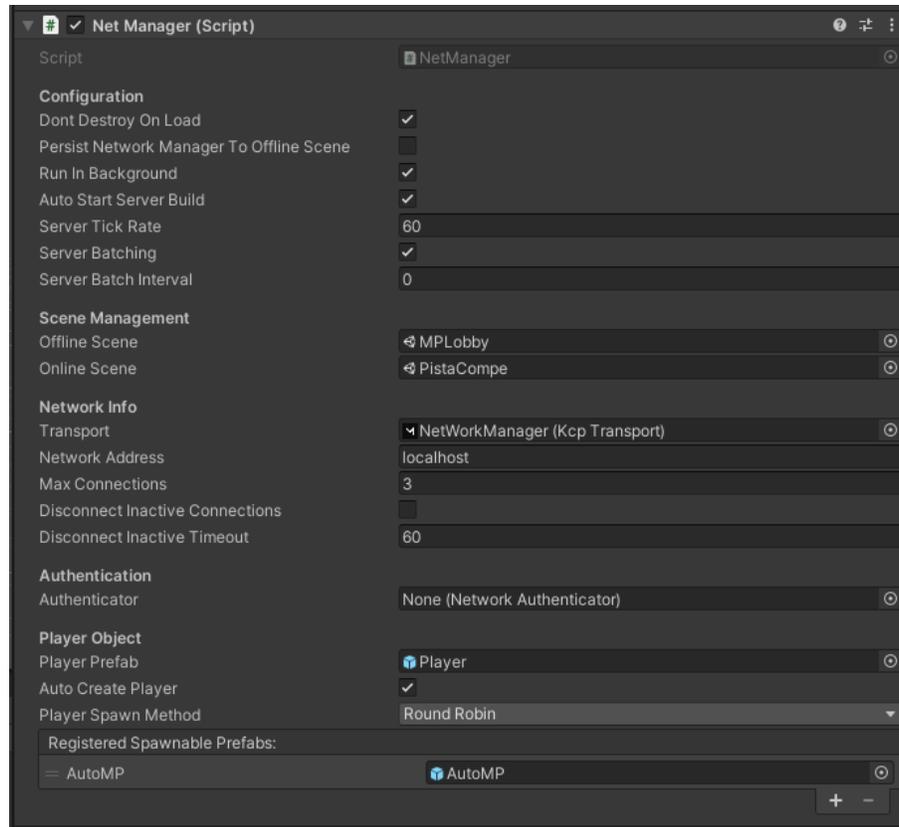


Figura 2.10: Network Manager.

- **Network Identity:** este componente sirve para identificar cada objeto dentro de la red. Gracias a este identificador el sistema es consciente y reconoce cada uno de los objetos que interactúan bajo una misma red. También permite que algunos objetos solo existan para el servidor y sean gestionados por este. A continuación en la figura 2.11 se muestra este componente.

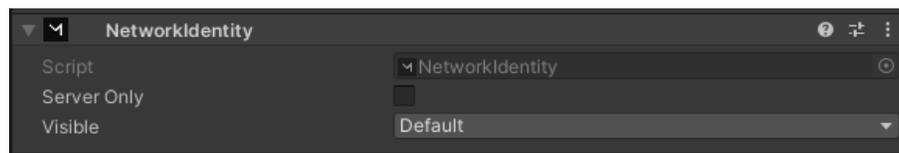


Figura 2.11: Network Identity.

- **Network Transform:** la función de este componente es sincronizar los movimientos, posiciones, rotaciones y escalas dentro de la red. Este componente es importante debido a que los movimientos solo son visibles por el promotor de la acción y en una modalidad multijugador se requiere que, tanto cliente como servidor vean sus movimientos entre sí. A continuación en la figura 2.12 se muestra este componente.

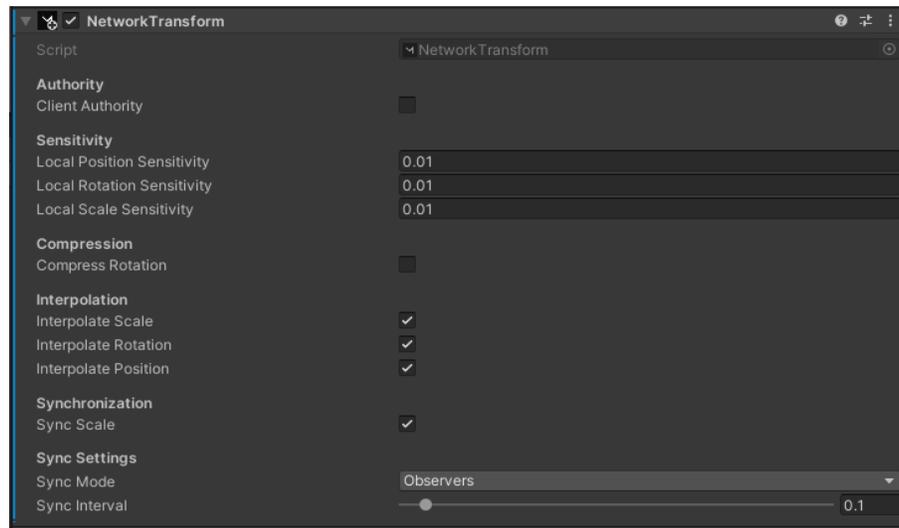


Figura 2.12: Network Identity.

- **Network Rigidbody:** Este componente se encarga de sincronizar las físicas de los objetos y sus interacciones. A continuación en la figura 2.13 se muestra este componente.

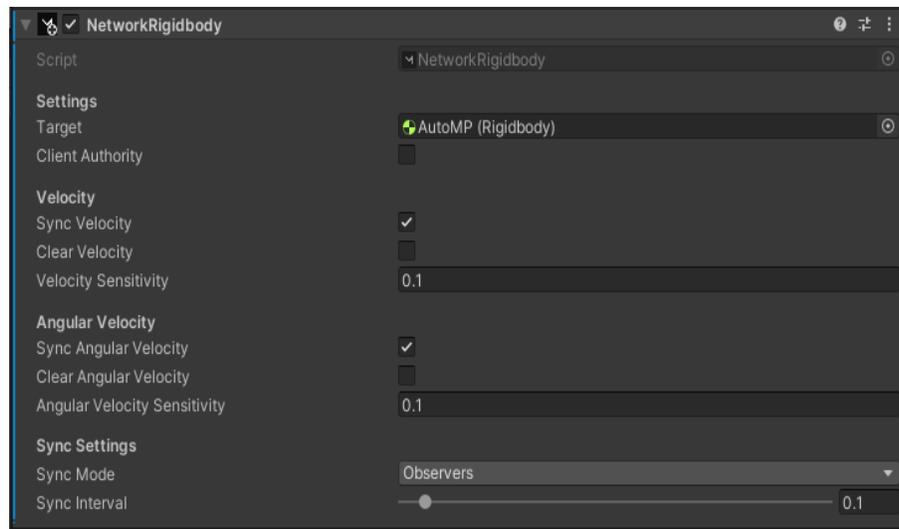


Figura 2.13: Network Rigidbody.

- **Configuración de realidad virtual:** Para generar un entorno de realidad virtual en *unity*, se utilizó el paquete de *Google VR* el cual permite mostrar contenido apto para gafas de realidad virtual, además que contiene diferentes controladores necesarios para su funcionamiento. Dentro de los paquetes de *Google VR* se encuentra *Cardboard*, mediante el cual se genera la realidad virtual. A continuación, se explican las funcionalidades de *Google VR* y como permite generar la realidad virtual.
  - **Importar el paquete de Google VR en Unity:** una vez creado el proyecto, se debe ingresar en *Assets* después seleccionar *Import Package* y finalmente en *Custom Package*. Dentro de la pestaña que se desplegó se debe seleccionar *GoogleVRForUnity* archivo descargado previamente de la página de *Unity*, se selecciona y click en importar. A continuación en la figura 2.14 se muestra este componente.

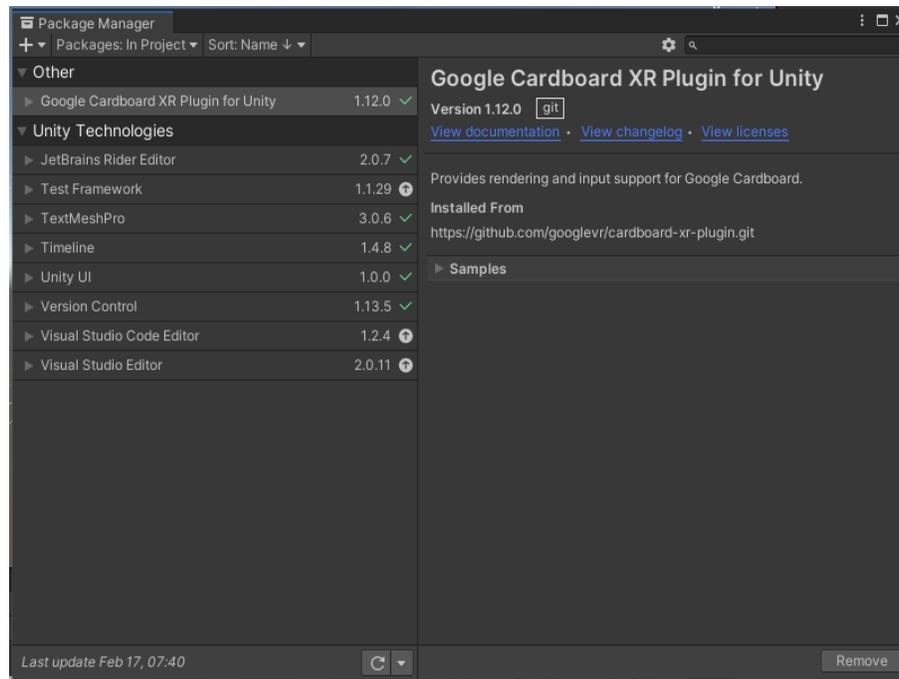


Figura 2.14: Google Cardboard XR Plugin for Unity.

- **Configuración de ajustes para Google VR:** se debe ingresar en *file*, después en *Build Settings*, se activa la plataforma *Android*. Sobre esta misma ventana se da click en *Player Settings* y en *XR plug-in management* se activa el *Cardboard*. A continuación en la figura 2.15 se muestra este componente.

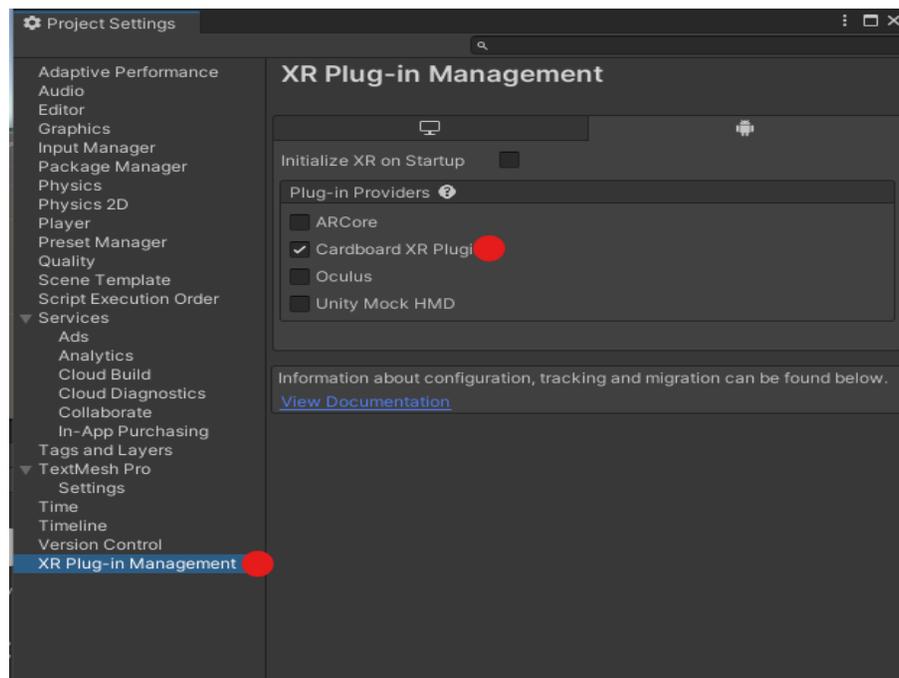
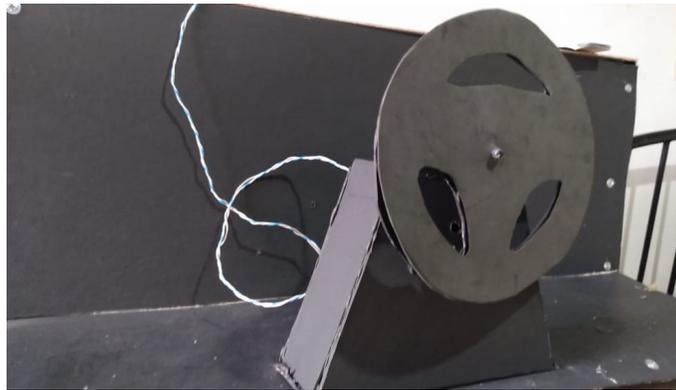


Figura 2.15: Cardboard .

**j) Volante y pedales (hardware)**

Uno de los objetivos de la realidad virtual inmersiva es buscar que las personas entren en un escenario en donde pueden ser protagonistas, es decir, pueden decidir que acciones hacer dentro del juego, todo esto mediante el estímulo sensorial que se consigue a través de los sentidos, como vista, oído, tacto. Por esto se llevo a cabo la construcción de una cabina de conducción con el fin de aportar inmersividad al jugador y ver su destreza motriz.

Esta estructura se compone de una dirección o volante electrónico el cual fue construido a partir de un potenciómetro y unas circunferencias en cartón semejantes a un volante, así se muestra en la figura a de 2.16. además cuenta con dos pedales electrónicos contruidos con dos finales de carrera y cubos de cartón, así se muestra en la figura b de 2.16. los componentes electrónicos se instalaron sobre la estructura hecha de papel cartón y tubos de PVC.



(a) Modelo de volante o dirección en cartón.



(b) pedales en cartón.

Figura 2.16: Volante y pedales diseñados.

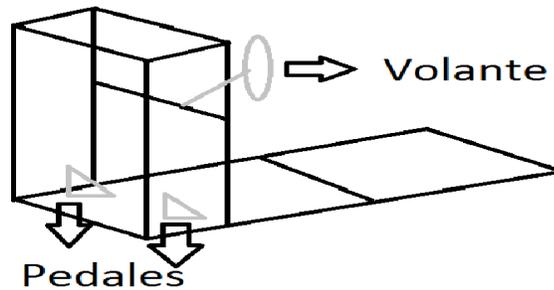
Para diseñar la estructura soporte de la cabina de conducción, se utilizaron dieciocho metros de tubos de PVC con diámetro de una pulgada, estos se cortaron en seis tubos de 90 cm, dos de 80 cm, ocho de 50 cm y doce de 20 cm, en la figura a de 2.17 se muestran los tubos utilizados. Además para armar la estructura y unir los tubos, se usaron dieciocho uniones en forma de "T", cuatro uniones en ángulo de "45°", ocho uniones en ángulo de "90°" ver figura b de 2.17. Se realizó una revisión de las actuales cabinas de conducción *gamer* y se diseñó un plano como guía para unir, cortar y armar una estructura similar a estas, así se ilustra en la figura c 2.17.



(a) Tubos cortados.



(b) Uniones utilizadas en la estructura.



(c) Plano estructura.

Figura 2.17: Soporte en PVC cabina de conducción.

Los componentes electrónicos (volante y pedales) se encuentran conectados a un *arduino* mega por medio de cable UTP. La conexión entre el *arduino* y la aplicación *android* se logra con el modulo bluetooth HC-06, este se encarga de leer los valores digitales (1,0) entregados por cada pedal y los valores analógicos enviados por el potenciómetro (0-1024 bits), estos valores son interpretados por la aplicación para realizar los movimientos del carro como: girar a la derecha, frenar, acelerar, etc. A continuación en la figura 2.18 se presenta el diagrama de conexiones eléctricas entre los dispositivos electrónicos utilizados y en la figura 2.19 el montaje terminado.

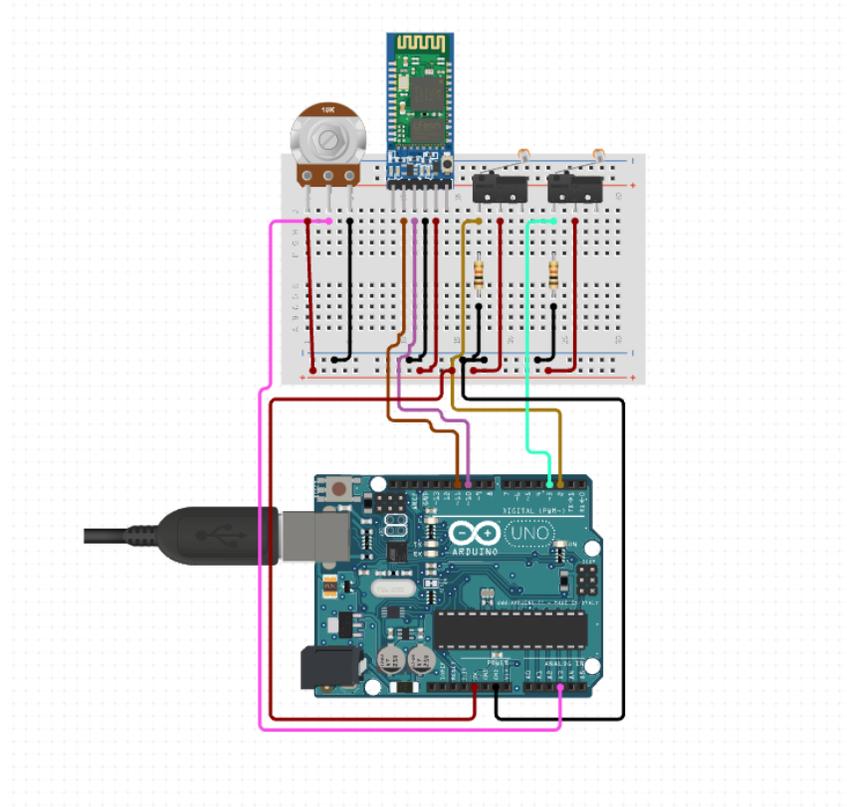


Figura 2.18: Diagrama de circuitos.



Figura 2.19: Montaje completo con mecanismo electrónico.

# Capítulo 3

## Diseño del entorno 3D

En este capítulo se presentan cada uno de los componentes gráficos del entorno 3D como: el automóvil, las pistas, imágenes, casas, edificios, carreteras, curvas, puentes, arboles, etc. También se describe como se realizó el diseño, que componentes se utilizaron, los colores, las fuentes o referencias, software utilizado, etc.

- **Escenarios:** *Brain Race* es un juego serio de carreras el cual tiene tres pistas, cada una se diseño teniendo como referencia un lugar de la vida real, por ejemplo: una ciudad, un bosque o un paisaje. A continuación se presentan los tres escenarios diseñados:
  - **Bosque:** Este escenario se diseño tomando como referencia un bosque. Inicialmente se utiliza la herramienta de terrenos para crear un plano sobre el cual se ubican cada uno de los componentes de la escena, ver figura 3.1.

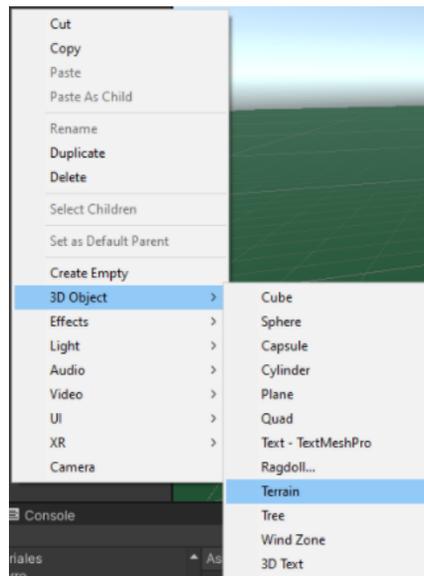
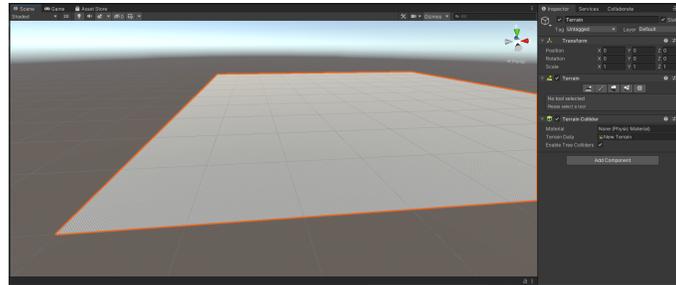
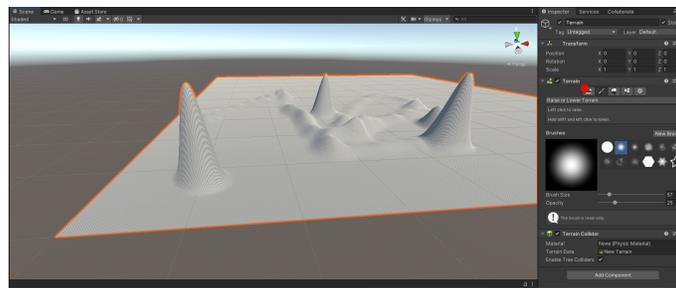


Figura 3.1: Terrenos en unity.

Dentro de un objeto tipo terreno se encuentra la herramienta denominada *create neighbor terrains*, la cual permite modificar las mallas del plano, esta cuenta con diferentes brochas utilizadas para moldear en 3D superficies como: mesetas, montañas, valles, cordilleras, sierras, ver figura 3.2.



(a) Terreno.



(b) Tipos de brochas en objeto terreno.

Figura 3.2: Edición de un objeto tipo terreno.

Adicionalmente el objeto terreno cuenta con una propiedad que permite asociar un material o color al plano, ver figura 3.3. Para saber que color utilizar se realiza una búsqueda de texturas con tonalidades e iluminación que permita visualizar el efecto de un bosque 3D.

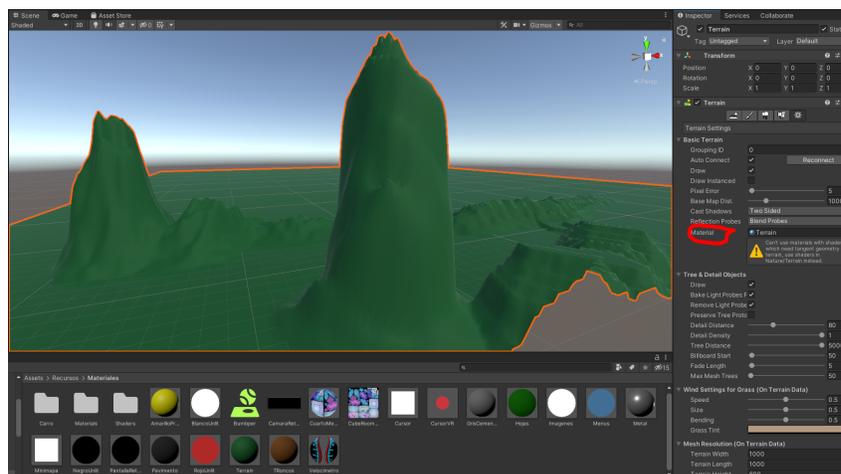
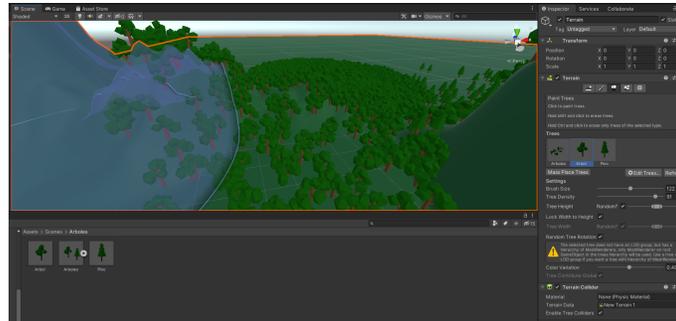
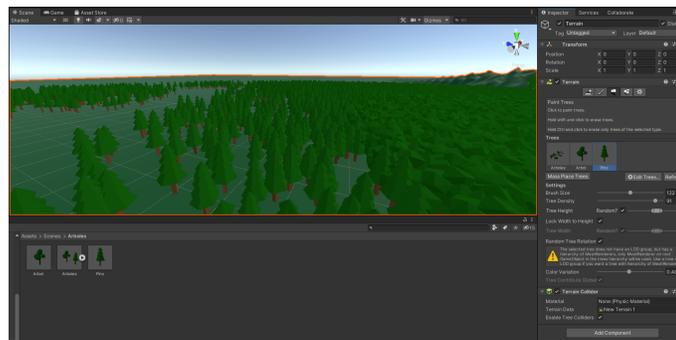


Figura 3.3: Material de un terreno.

Sobre las montañas creadas se utilizó la herramienta *Paint trees*, cuya función consiste en añadir arboles, o dibujar cualquier *prefab* seleccionado sobre un terreno. Los objetos dibujados con esta herramienta se hacen visibles solamente cuando el usuario o cámara están relativamente cerca, ver figura 3.4.



(a) Tipos de arboles.



(b) arboles creados con la herramienta paint trees.

Figura 3.4: Herramienta paint trees.

En este escenario se encuentra una carretera de color negro con muros de contención, este diseño se realizó en la herramienta de *blender*, con el fin de crear una pista con curvas y rectas que aporten complejidad al momento de conducir.

En esta pista se encuentran distribuidos banderines o metas, este diseño se encontró en la *Asset store* de *unity* bajo el nombre de *Standard Assets*. En la parte superior de los banderines se ubican las imágenes, que deben ser memorizadas por el jugador. A continuación en la figura 3.5 se ilustra la pista diseñada para la modalidad individual y competitiva:

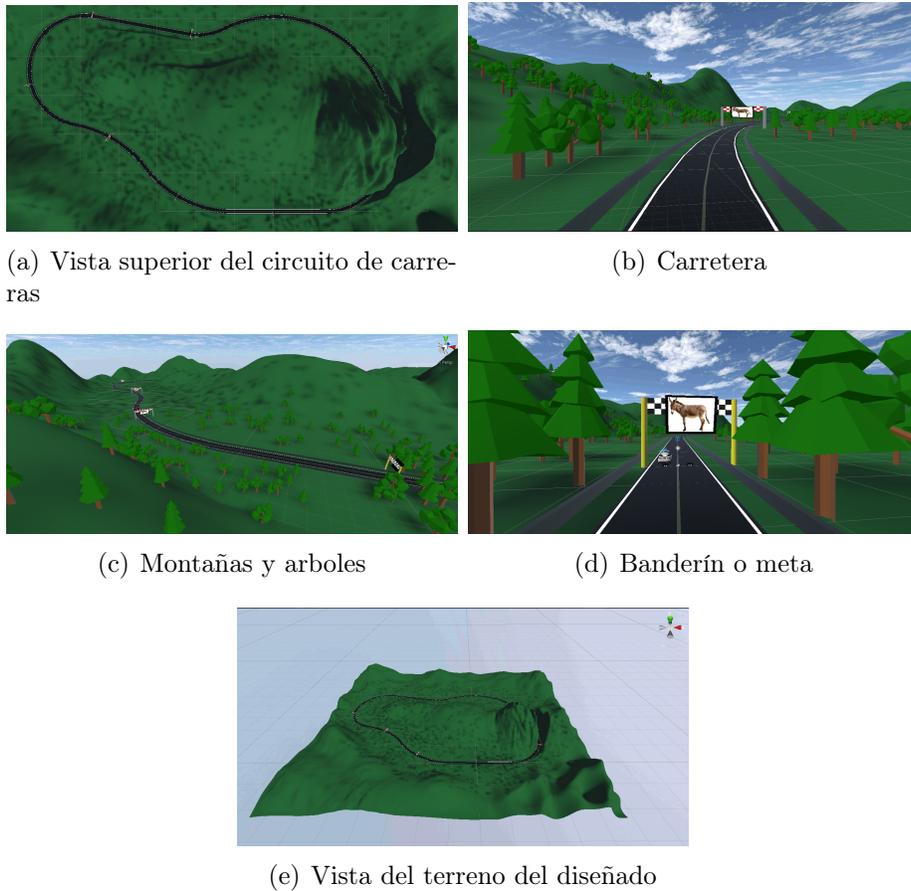
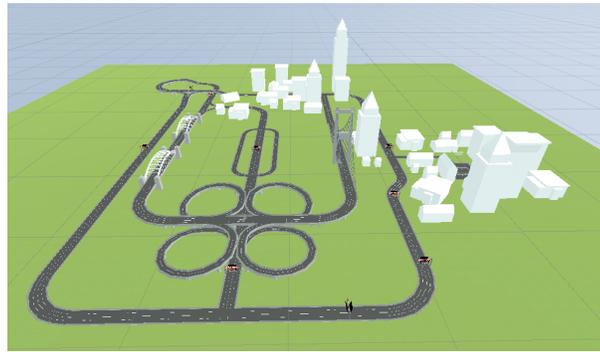


Figura 3.5: Escenario bosque diseñado para la modalidad individual y competitiva.

- **Ciudad:** El diseño de esta escena fue inspirado en una ciudad de la vida real, donde se encuentran diferentes distractores visuales como edificios, puentes, esquinas, semáforos, casas, etc.

Dentro de *Asset store* de *Unity* se encuentran diferentes modelos 3D de ciudades, para elegir cual utilizar se establecieron algunos criterios de selección como: el modelo sea gratis, que utilizaran gráficos 3D de baja resolución, buena iluminación, sin errores de compilación, sin animaciones, funcional para versiones de *Unity* del 2019 en adelante, teniendo en cuenta estas consideraciones se elige el *Asset* denominado *Low Poly Road Pack*. Dentro de este escena fueron distribuidos diez banderines donde aparecen las imágenes que se deben de memorizar. A continuación en la figura 3.6 se muestra la pista diseñada para la modalidad colaborativa:



(a) Vista superior del circuito de carreras en el escenario ciudad



(b) Vista desde el frente del circuito de carreras en el escenario ciudad



(c) Ciudades y casas

Figura 3.6: Escenario ciudad diseñado para la modalidad colaborativa.

Para exportar este modelo 3D de la *Asset store* de *Unity*, se deben de realizar los siguientes pasos:

1. Entrar en la *Asset store* de *Unity*, esta se encuentra ubicada en la parte superior de la escena, como se muestra en la figura 3.7.

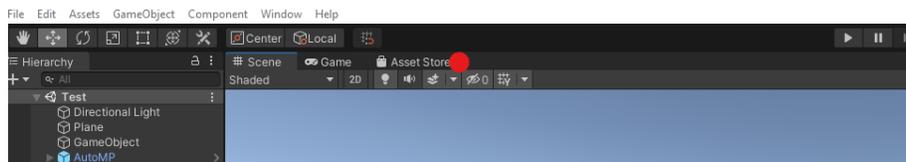


Figura 3.7: Ubicación de la Asset store.

- después de dar click en la *Asset store*, se abre una pestaña notificando que las versiones de *Unity* superiores al 2020.1 deben ingresar desde un buscador, como se muestra en la figura 3.8.

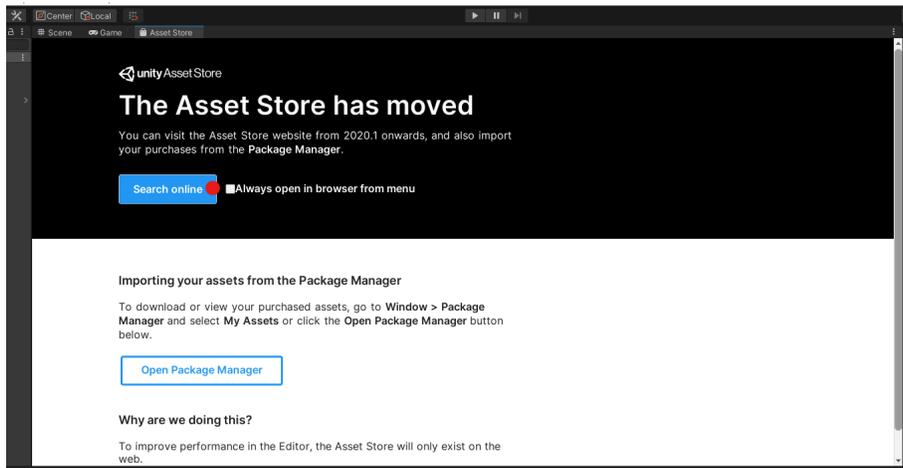


Figura 3.8: Aviso de sobre nueva forma de ingreso.

- después de dar click en la opción *Search online*, se abre la *Asset store* en el buscador de preferencia, como se muestra en la figura 3.9.

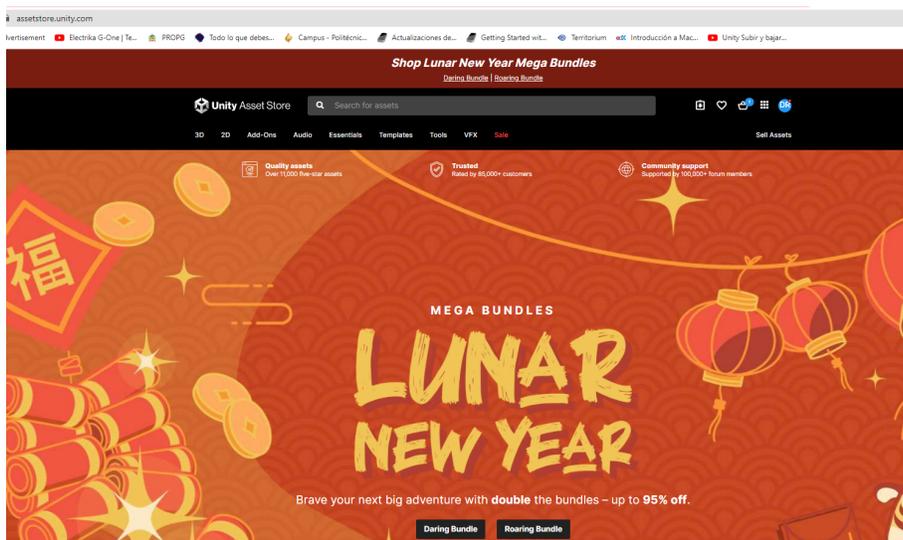


Figura 3.9: Asset store desde el buscador de google.

- después en el buscador de la *Asset store*, se escribe que modelo 3D queremos buscar, en este caso se escribió *low poly road*, encontrando todos los modelos 3D de pistas con gráficos de baja resolución, como se muestra en la figura 3.10.

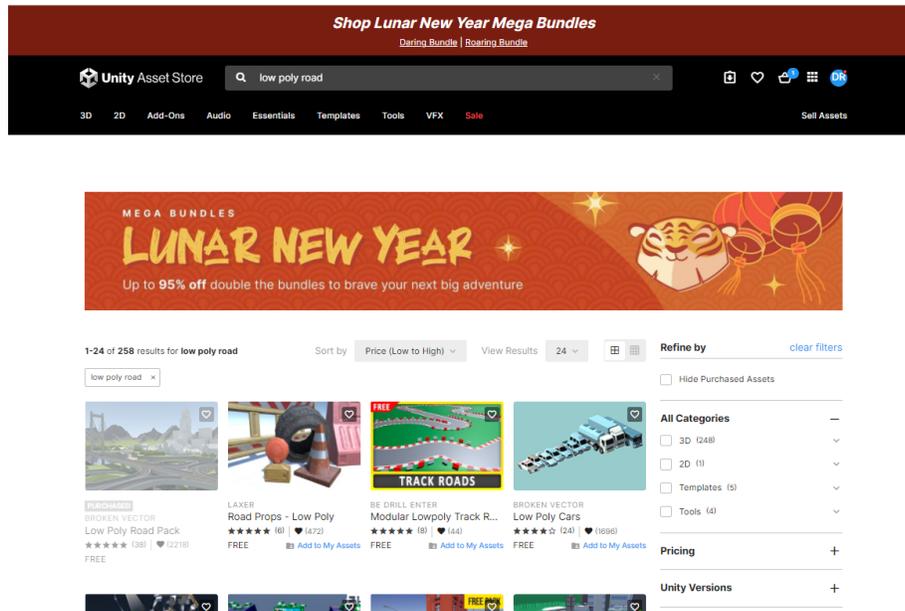


Figura 3.10: Resultados de la búsqueda.

5. Cuando se encuentre el paquete con los modelos 3D que cumplen con los criterios de selección establecidos, se debe abrir el paquete desde *Unity* dando click sobre la opción "*open in Unity*", como se muestra en la figura 3.11.

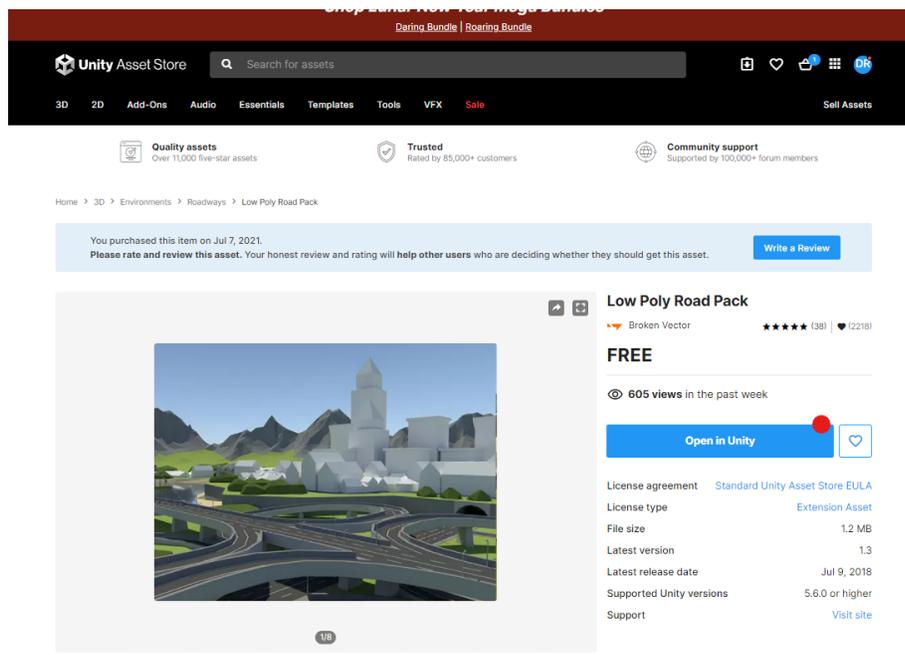


Figura 3.11: Importar modelo.

6. Después de dar click sobre la opción "*open in Unity*", aparece una ventana emergente solicitando permiso para abrir el vinculo desde el programa de *Unity*, a continuación se debe elegir la opción abrir con *Unity editor*, como se muestra en la figura 3.12.

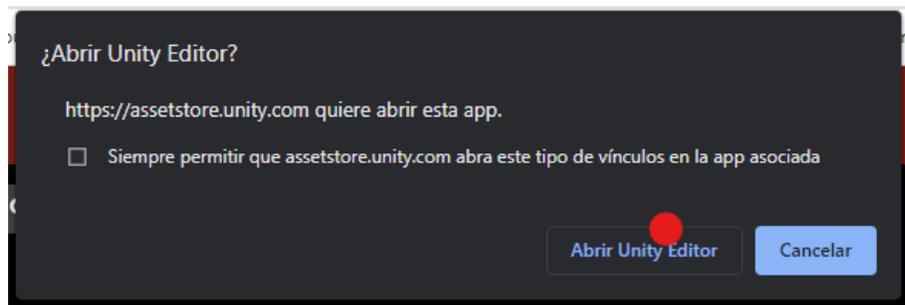


Figura 3.12: Solicitud de permiso para abrir el modelo 3D desde *Unity*.

7. Una vez se abre el paquete seleccionado en *Unity*, aparecerá una ventana donde se muestra toda la información acerca de los desarrolladores, en esta pestaña se debe dar click sobre la opción importar, como se muestra en la figura 3.13.

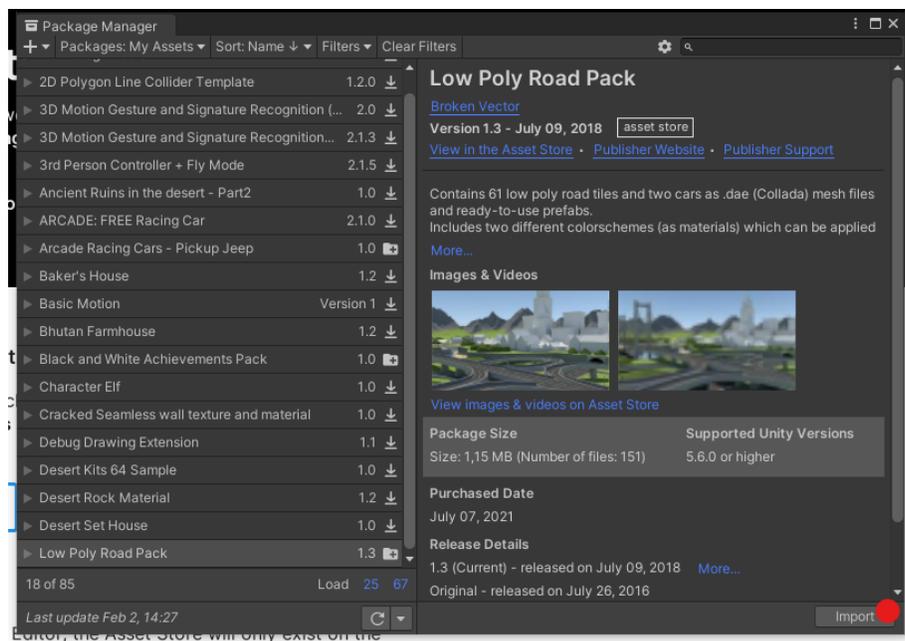


Figura 3.13: Información del modelo 3D en *Unity*.

8. Después de elegir la opción importar, aparece una ventana mostrando los modelos que contiene el paquete, en esta ventana se debe de corroborar si se desea o no importar el paquete, como se muestra en la figura 3.17.

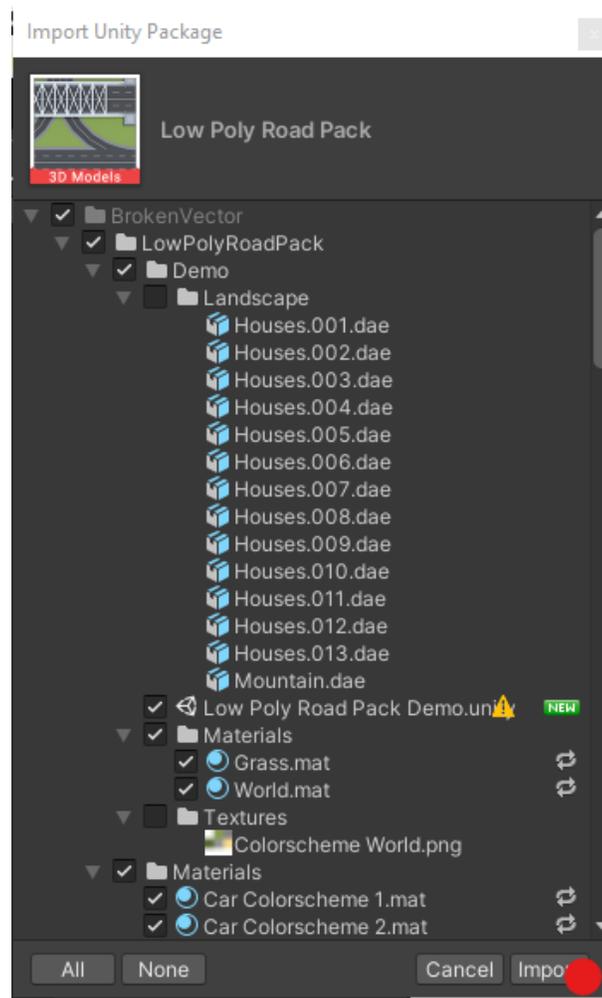
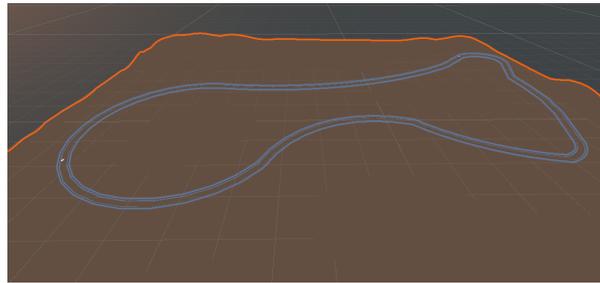


Figura 3.14: Modelo 3D que contiene el paquete seleccionado.

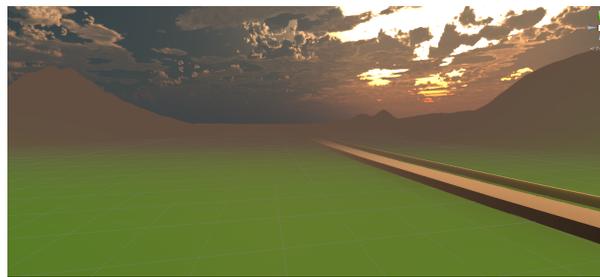
- **Atardecer:** Esta escena se diseño inspirada en un paisaje con un atardecer, mediante la herramienta de terrenos se modelo cada montaña para que encajara perfectamente con la pista realizada en *Blender*. La iluminación de esta escena tiene tonalidades oscuras con el fin de reducir el campo visual de los jugadores, aumentando su atención y concentración en las imágenes de cada ubicadas en cada banderín. Esta es la pista mas grande de todo el juego, al igual que las anteriores contiene diez metas, con sus respectivas imágenes. A continuación en las figuras 3.15 se presenta la pista diseñada para la modalidad cooperativa:



(a) Vista superior del circuito de carreras



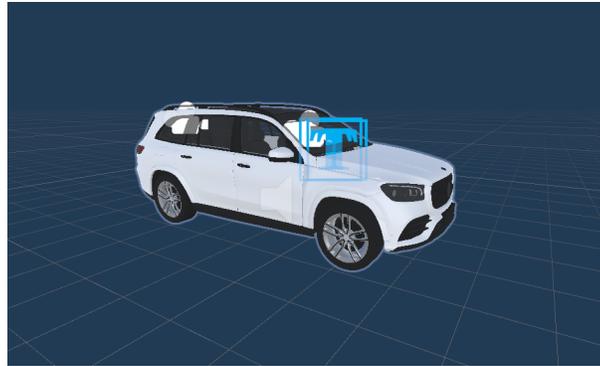
(b) Carretera y banderín



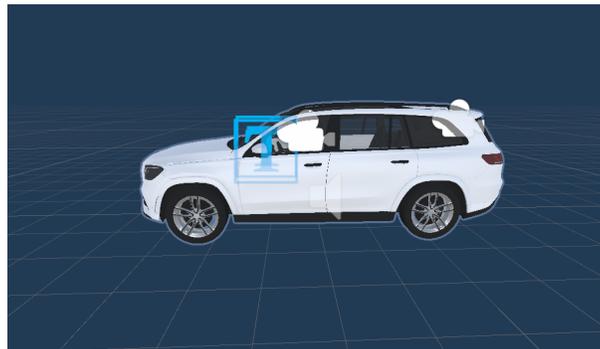
(c) Montañas y atardecer

Figura 3.15: Escenario de paisaje con atardecer diseñado para la modalidad cooperativa.

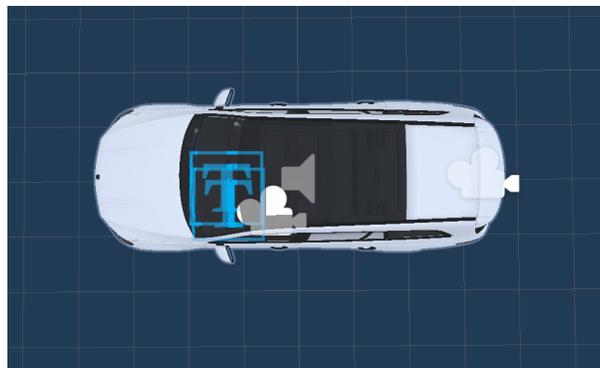
- **Auto:** se realiza una búsqueda de un modelo 3D de un auto para utilizar en el juego, como criterios de selección se tuvo en cuenta que fuera un modelo sin costo, funcional y compatible con el entorno de *Unity*. Se encontró en [94] el modelo 3D de la camioneta GSL 450 Mercedes Benz, este fue exportado en el entorno *Unity* y funciono correctamente. A continuación en las figuras 3.16 se presentan el modelo 3D del vehículo utilizado dentro del juego.



(a) Vista diagonal del modelo camioneta GSL 450 Mercedes Benz.



(b) Vista camioneta GSL 450 Mercedes Benz.



(c) Vista superior camioneta GSL 450 Mercedes Benz.



(d) Vista interior camioneta GSL 450 Mercedes Benz.

Figura 3.16: Imágenes modelo 3D de la camioneta GSL 450 Mercedes Benz.

- **Tacómetro:** Dentro del modelo 3D del vehículo se diseña un tablero donde se visualiza la velocidad del vehículo, el tiempo que lleva manejando y el mapa de la pista. A continuación se explica como se diseño cada componente:

La velocidad del vehículo y el cronometro se diseñaron a partir de dos objetos tipo *text-meshpro*, los cuales tiene la funcionalidad de cuadros de textos donde se puede imprimir cualquier tipo información. en la figura 3.17 se muestra donde se encuentran este tipo de objetos.

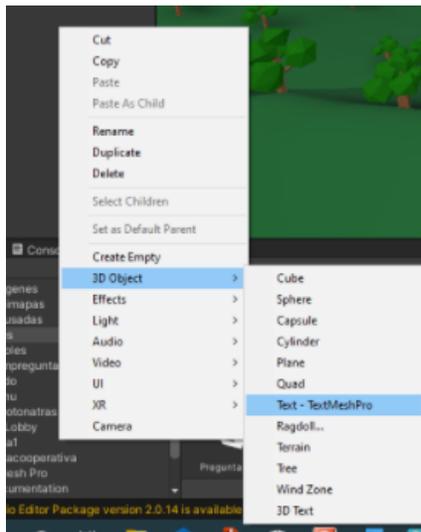


Figura 3.17: Objeto tipo textmeshpro.

Estos cuadros de texto se ubican dentro del vehículo cerca al volante dando la impresión que son el tacómetro, así se puede apreciar en la figura 3.18 .



Figura 3.18: Tacómetro diseñado.

Por otra parte en la herramienta de *Blender* se diseño un plano de cada una de las pistas, el cual fue utilizado para crear un minimapa sobre el cual se muestra la ubicación del jugador en tiempo real, en la figura 3.19 se muestra uno de los minimapas creados.

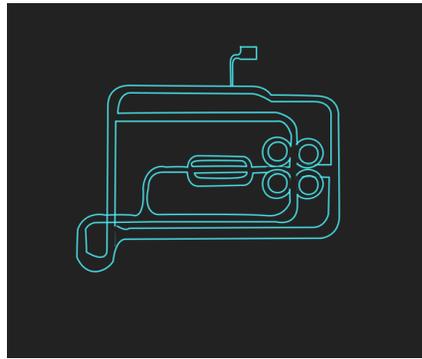


Figura 3.19: Tacómetro diseñado.

- Pista:** Se utilizó el programa de modelado *Blender* para diseñar dos pistas, con el fin de aportar dificultad y motivación al juego. Primero se realiza un plano guía con la forma que lleva la carretera, ver figura 3.20

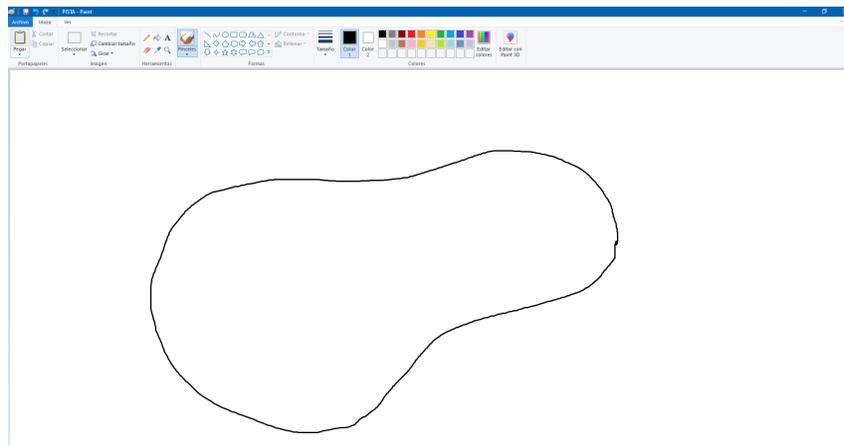


Figura 3.20: Plano de la pista.

después se exporta el plano guía a *blender* para trazar los trayectos de la carretera, ver figura 3.21.

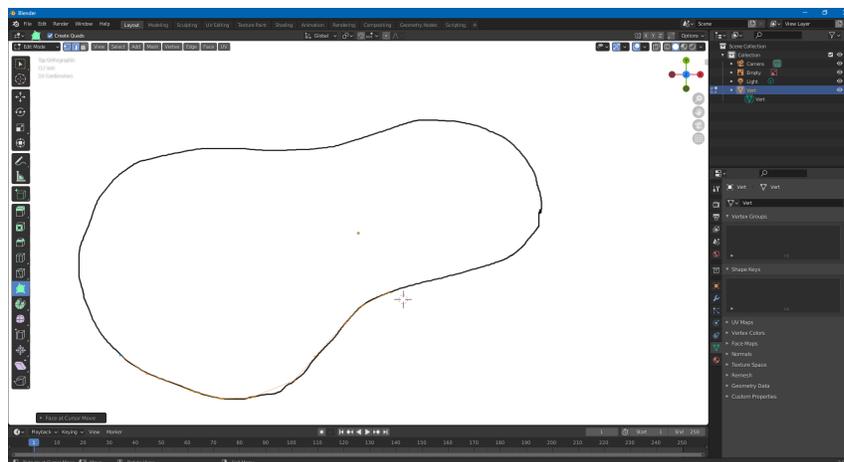
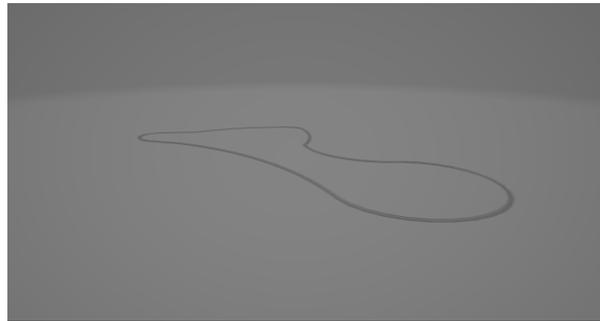
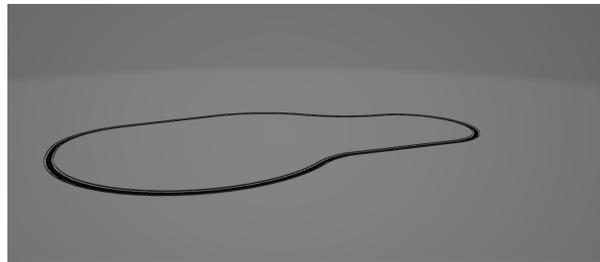


Figura 3.21: Trazos en blender.

A continuación se utiliza la herramienta *skin* de *Blender* para crear una superficie usando los vértices creados sobre el plano guía, finalmente se exporta el modelo 3D que será utilizado en los escenarios de *Unity* como carreteras. En las figuras B.2 se presenta el modelo 3D final.



(a) Pista 1.



(b) Pista 2.

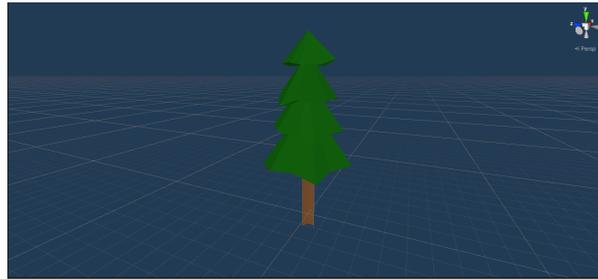
Figura 3.22: Modelo 3D creados en blender.

- **Banderín:** Dentro de todos los escenarios se utilizó el modelo 3D de un banderín o meta de carreras, este diseño se encontró en la *Asset store* de *Unity* bajo el nombre de *Standard Assets* [95], en la figura 3.23 se ilustra el modelo de este componente.



Figura 3.23: Banderín o meta.

- **Arboles:** Dentro del escenario bosque se utilizaron modelos 3D de arboles, estos diseños se encontraron en la *Asset store* de *Unity* bajo el nombre de *Free Trees* [96], en la figura 3.24 se ilustra el modelo de este componente.



(a) Árbol tipo 1.



(b) Árbol tipo 2.

Figura 3.24: Modelo 3D de arboles.

- **Imágenes de lugares, objetos y personas:** En *Unity* fueron importadas setenta imágenes de diferentes temáticas como: objetos, lugares, colores, comidas, personas, etc. Estas imágenes fueron utilizadas por un código que selecciona de manera aleatoria que imágenes aparecerán en cada partida y estas deben de ser memorizadas por el o los jugadores. En la figura 3.25 se muestran las imágenes importadas en el juego.

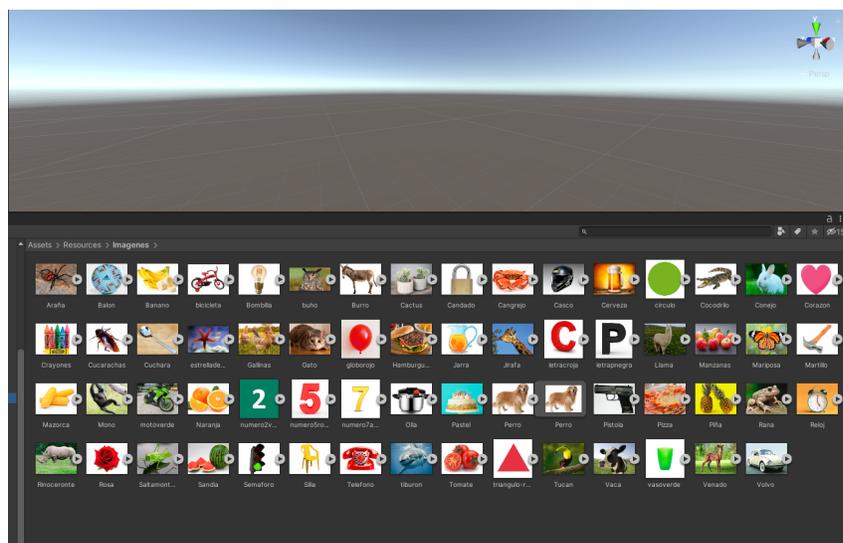


Figura 3.25: Imágenes que se deben de memorizar.

# Capítulo 4

## Experimentación y resultados

En el siguiente capítulo se desarrolla la fase de experimentación, además se presentan los resultados obtenidos y el análisis estadístico de los mismos.

- **Método:** Autores como [13, 97, 98] aconsejan utilizar estrategias o técnicas para mejorar la memoria basadas en: la relación de información con un agente externo como los registros sensoriales (Olfato, vista, gusto, tacto y oído), también incentivar actividades interesantes, emotivas, lúdicas, donde el usuario exprese emociones y sentimientos, debido a que la memoria graba mejor cuando relaciona el recuerdo con un estado de ánimo. Por último recomiendan utilizar actividades que impliquen el uso de las funciones ejecutivas dado que esto contribuye a la optimización de los procesos cognitivos.

El objetivo de este trabajo de grado consiste en analizar que resultados obtienen las personas cuando estimulan la memoria de manera grupal e individual, debido a que de esta forma expresan diferentes emociones y se sienten motivadas a seguir con las sesiones. Por otra parte se quiere analizar si las modalidades de juego: competitiva, colaborativa y cooperativa, tienen algún impacto sobre el proceso de estimulación de la memoria.

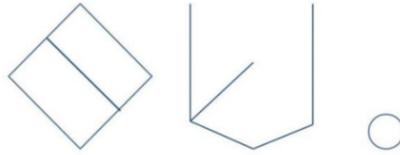
Para este experimento se utiliza el juego serio de realidad virtual inmersiva multijugador "Brain race", el cual fue desarrollado para el presente trabajo de grado, con el fin de que cumpliera todas las características necesarias para llevar a cabo el experimento. En el cual se busca como primera medida estimular la memoria visual a corto plazo, secuencial y sensorial utilizando una pista de conducción, donde a lo largo de esta, se encontraran una serie de imágenes las cuales deberán ser memorizadas por los jugadores. Segundo se tienen diferentes modalidades de juego como individual, cooperativa, colaborativa y competitiva, diseñadas para uno o dos jugadores, donde cada uno desempeña un diferente rol. Tercero el juego cuenta con estímulos auditivos, sensoriales y visuales debido al uso de gafas de realidad virtual y los controles implementados como, la cabina de conducción diseñada en el capítulo dos y controles tipo xbox.

- **Población:** En esta investigación se tomó una muestra de dieciséis personas con edades entre los doce y cuarenta y un años (ocho mujeres, ocho hombres), sin enfermedades cognitivas de base. Se organizaron grupos de cuatro personas donde a cada uno les fue asignado una modalidad de juego: cooperativa, colaborativa, competitiva o individual, en la categoría multijugador se organizaron parejas de juego acorde a su disponibilidad de tiempo. En la tabla 4.1 se describen algunas características relevantes de la población.

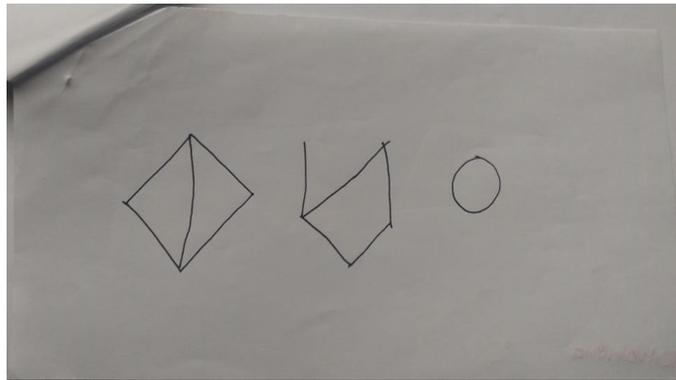
Sujeto	Genero	Edad	Ocupación
1	Femenino	41	Docente
2	Femenino	26	Psicóloga
3	Femenino	23	Estudiante
4	Femenino	23	Estudiante
5	Femenino	22	Estudiante
6	Femenino	23	Estudiante
7	Femenino	27	Estudiante
8	Masculino	22	Estudiante
9	Masculino	25	Técnico
10	Masculino	26	Médico
11	Masculino	33	Auxiliar contable
12	Masculino	26	Estudiante
13	Masculino	13	Estudiante
14	Masculino	16	Estudiante
15	Masculino	24	Estudiante
16	Masculino	23	Estudiante

Tabla 4.1: Características de la población.

- Prueba estándar para evaluar la memoria:** Todos los participantes fueron evaluados antes de comenzar el experimento con el test de **Benton**, herramienta tradicional utilizada para evaluar la habilidad de percepción, memoria visual y otro tipo de habilidades motoras que implican la reproducción y copia de dibujos o formas [99], en la figura 4.1 se muestra una lamina del test de Benton y el dibujo realizado por un participante después de pasar el tiempo de exposición. A continuación se describe la aplicación del test:

**Imagen 10**

(a) Lámina número 10 forma A del test de Benton.



(b) Reproducción realizada por el participante número 4 de Lámina número 10, forma A del test de Benton.

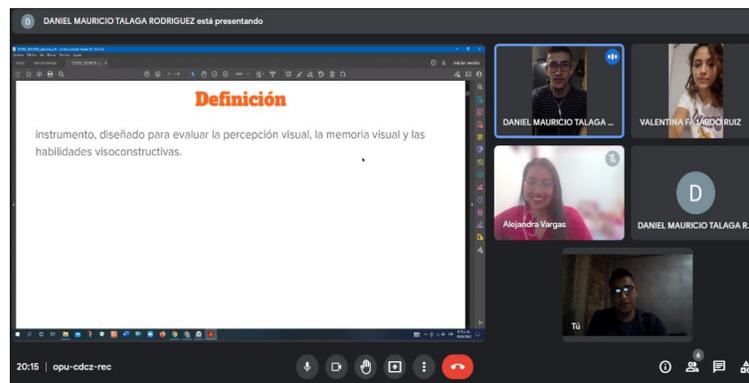
Figura 4.1: Lámina versus reproducción test de Benton.

El test de Benton está compuesto por tres formas (C, D y E) donde cada uno tiene diez láminas, las cuales se administran en diferentes tiempos de la siguiente manera [89]:

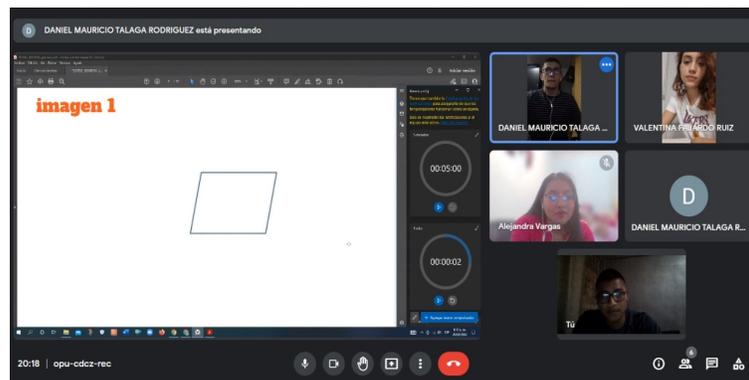
- **Administración A:** 10 segundos de exposición y reproducción inmediata.
- **Administración B:** 5 segundos de exposición y reproducción inmediata.
- **Administración C:** Se copian los dibujos manteniéndolos a la vista del individuo (Para adultos mayores con deterioro cognitivo avanzado.).
- **Administración D:** 10 segundos de exposición y reproducción de memoria después de 15 segundos.

Para evaluar a la población de este experimento se utilizaron los tipos de administración A, B y D. Sin embargo debido a la actual situación de salud pública que se vive a nivel mundial, este test tuvo que ser aplicado de manera virtual a cada uno de los participantes, mediante reuniones de *google meet* donde se presentaban diapositivas con el test de Benton tradicional, se entregaron treinta rectángulos de papel y se solicitó que encendieran las cámaras durante toda la reunión para controlar que los participantes dibujen las figuras justamente después de retirar la imagen, en la figura 4.2 se ilustra este procedimiento.

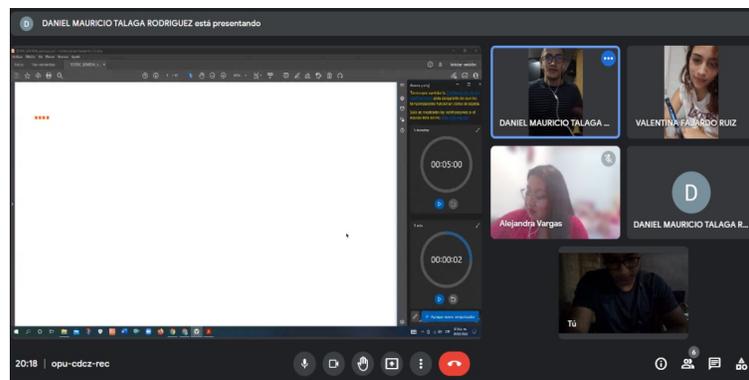
En la **tabla 4.2** se presentan los resultados obtenidos por los participantes en la prueba inicial y final, con el test de Benton.



(a) Introducción al test de Benton por Google meet.



(b) Presentación de lamina.



(c) Reproducción de lamina.

Figura 4.2: Aplicación test de Benton.

Sujeto	Modalidad	Prueba inicial						Prueba final						
		Reproducciones correctas			Valoración de errores			Reproducciones correctas			Valoración de errores			
		Forma A	Forma B	Forma D	Forma A	Forma B	Forma D	Forma A	Forma B	Forma D	Forma A	Forma B	Forma D	
1	Individual	7	7	6	4	4	4	4	9	8	6	1	2	6
2	Individual	7	8	8	3	5	4	4	10	9	8	0	1	3
3	Individual	6	9	9	5	1	1	1	10	9	8	0	1	2
4	Individual	8	9	7	3	1	3	3	10	8	6	0	2	5
5	Competitiva	9	9	8	1	1	3	3	9	8	9	1	3	1
6	Competitiva	5	3	5	6	7	5	5	10	7	7	1	3	3
7	Competitiva	6	6	5	6	6	10	10	10	9	7	0	1	5
8	Competitiva	8	7	8	2	3	2	3	7	6	8	3	4	2
9	Colaborativo	8	7	4	2	4	12	4	10	6	7	0	4	6
10	Colaborativo	8	8	2	2	7	6	7	10	8	6	0	3	5
11	Colaborativo	10	9	10	0	1	1	1	10	8	10	0	2	2
12	Cooperativo	10	10	7	0	0	3	0	10	9	9	0	2	1
13	Cooperativo	10	7	8	0	3	2	3	10	7	8	0	3	3
14	Cooperativo	6	6	7	5	6	3	6	10	9	6	0	2	4
15	Cooperativo	8	5	5	2	6	6	6	9	8	8	1	2	3
16	Colaborativo	10	7	9	0	3	1	3	10	6	7	0	4	3

Tabla 4.2: Resultados test de Benton.

- **Materiales:** Para esta investigación se utilizaron seis gafas de realidad virtual, dieciocho celulares con sistema operativo *android*, seis controles Xbox con *bluetooth*, una silla de escritorio tradicional y la cabina de conducción diseñada. Para iniciar el experimento se instaló el apk en cada uno de los celulares de los participantes, se instruyó en cómo utilizarlo y qué funcionalidades tenía.
- **Diseño:** El experimento consiste en que los participantes deben utilizar el juego *brain race* en la modalidad asignada durante media hora, tres días a la semana, así a lo largo de un mes.

Se realizó un cronograma con cada uno de los participantes para acompañarlos y verificar que cumplieran con las sesiones estipuladas, cada uno fue instruido en las mecánicas del juego, en cómo utilizar los controles para manejar el vehículo, en las funcionalidades de la aplicación y en el uso de las gafas de realidad virtual.

Las pistas de carreras fueron diseñadas sobre entornos familiares como un bosque, una ciudad y un atardecer, con el fin de aportar realismo al juego serio. Las imágenes que se deben memorizar fueron distribuidas manualmente por todas las pistas y cada una de ellas se seleccionó, teniendo en cuenta la calidad de la imagen, su significado, sus colores y temática. Las preguntas y respuestas derivadas de cada imagen se diseñaron de tal forma que el usuario tenga que prestar atención en cada uno de los componentes que posee la imagen.

Los participantes pueden demorarse el tiempo que deseen observando las imágenes sin embargo este tiempo afectará su puntuación en la prueba, también pueden girar su vista libremente y conducir a la velocidad que quieran. La idea del juego serio es incentivar a los usuarios a plantear y planificar soluciones ante el objetivo propuesto.

- **Usabilidad del Software:** Para definir la usabilidad del juego se realizó una encuesta a los dieciséis participantes del grupo muestra del experimento. A continuación, en las figuras 4.3 se presentan los resultados de las preguntas realizadas.

1. ¿Presento dificultades con el uso de las gafas de realidad virtual?
2. ¿Presento dificultades al momento de manejar el vehículo virtual?
3. ¿Fueron claras las preguntas planteadas dentro del juego?
4. ¿Las imágenes que se debían memorizar fueron entendibles?
5. ¿Cómo considera la temática del juego?
6. ¿Considera que el juego plantea un reto para la memoria?

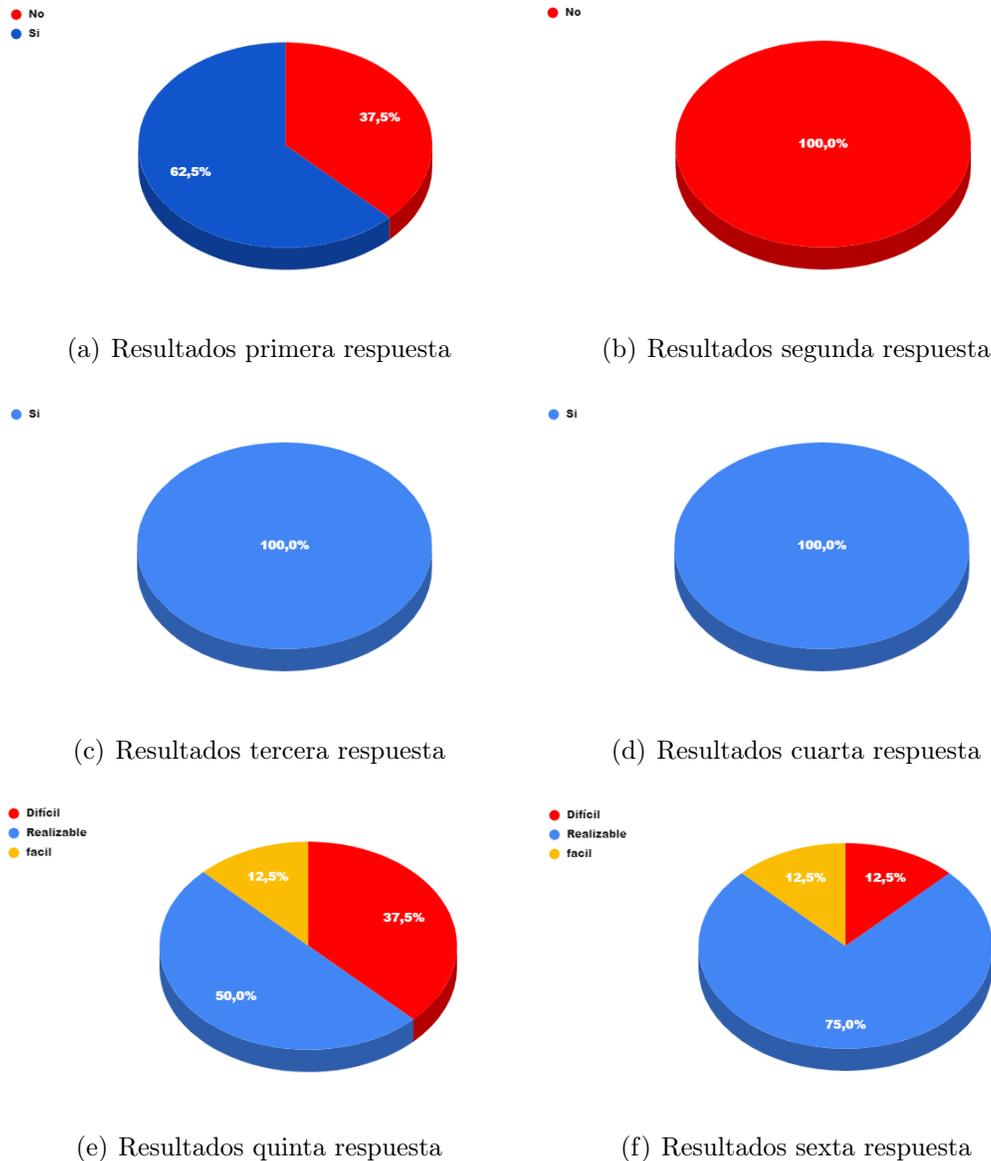


Figura 4.3: Respuestas encuesta de satisfacción realizada a la población del experimento.

- Resultados:** Para el análisis de resultados se compararon los datos obtenidos mediante la evaluación inicial y final de la memoria de los participantes con el test de Benton. Primero se utilizó la Prueba de Kruskal-Wallis, este método no paramétrico es utilizado para probar si un grupo de datos proviene o no de la misma población [100]. A continuación se presentan los resultados obtenidos mediante este método:

#### 1. Prueba de Kruskal-Wallis:

- Prueba de hipótesis para el test de Benton en la forma A:**

**Hipótesis nula 1:** Para el test de Benton en la forma A, el modo de juego no incide significativamente en la diferencia entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta.

**Hipótesis de investigación 1:** Para el test de Benton en la forma A, el modo de juego genera diferencias significativas entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta.

Al aplicar una prueba Kruskal Wallis utilizando el modo de juego como factor de variabilidad, se obtuvo que la probabilidad de que las diferencias entre los modos de juego individual, colaborativo, competitivo y cooperativo se deban al azar es del 35.5 %. Dado que este valor es mayor que el 5 %, no existe evidencia suficiente para rechazar la hipótesis nula 1.

- **Prueba de hipótesis para el test de Benton en la forma B:**

**Hipótesis nula 2:** Para el test de Benton en la forma B, el modo de juego no incide significativamente en la diferencia entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta.

**Hipótesis de investigación 2:** Para el test de Benton en la forma B, el modo de juego genera diferencias significativas entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta.

Al aplicar una prueba Kruskal Wallis utilizando el modo de juego como factor de variabilidad, se obtuvo que la probabilidad de que las diferencias entre los modos de juego individual, colaborativo, competitivo y cooperativo se deban al azar es del 61.7 %. Dado que este valor es mayor que el 5 %, no existe evidencia suficiente para rechazar la hipótesis nula 2.

- **Prueba de hipótesis para el test de Benton en la forma D:**

**Hipótesis nula 3:** Para el test de Benton en la forma D, el modo de juego no incide significativamente en la diferencia entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta.

**Hipótesis de investigación 3:** Para el test de Benton en la forma D, el modo de juego genera diferencias significativas entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta.

Al aplicar una prueba Kruskal Wallis utilizando el modo de juego como factor de variabilidad, se obtuvo que la probabilidad de que las diferencias entre los modos de juego individual, colaborativo, competitivo y cooperativo se deban al azar es del 9.3 %. Dado que este valor es mayor que el 5 %, no existe evidencia suficiente para rechazar la hipótesis nula.

Para el segundo análisis se utilizó la prueba de Wilcoxon, esta es una prueba no paramétrica utilizada para comparar el rango medio de dos muestras relacionadas y determinar si existen diferencias entre ellas[101].

## 2. Prueba de Wilcoxon:

- **Prueba de hipótesis para el test de Benton en la forma A:**

**Hipótesis nula 4:** Para el test de Benton en la forma A, no hay diferencia entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta.

**Hipótesis de investigación 4:** Para el test de Benton en la forma A, el porcentaje de aciertos obtenidos después del entrenamiento con la aplicación propuesta es mayor que porcentaje de aciertos inicial.

Al aplicar la prueba de rangos de Wilcoxon se obtuvo que la probabilidad de que la diferencia de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta se deba al azar es igual al 0.7 % . Dado que este valor es menor que el 5 %, se rechaza la hipótesis nula en favor de la de investigación.

- **Prueba de hipótesis para el test de Benton en la forma B:**

**Hipótesis nula 5:** Para el test de Benton en la forma B, no hay diferencia entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con

la aplicación propuesta.

**Hipótesis de investigación 5:** Para el test de Benton en la forma B, el porcentaje de aciertos obtenidos después del entrenamiento con la aplicación propuesta es mayor que porcentaje de aciertos inicial.

Al aplicar la prueba de rangos de Wilcoxon se obtuvo que la probabilidad de que la diferencia de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta se deba al azar es igual al 16.6%. Dado que este valor es mayor que el 5%, no existe evidencia suficiente para rechazar la hipótesis nula 5.

- **Prueba de hipótesis para el test de Benton en la forma D:**

**Hipótesis nula 6:** Para el test de Benton en la forma D, no hay diferencia entre el porcentaje de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta.

**Hipótesis de investigación 6:** Para el test de Benton en la forma D, el porcentaje de aciertos obtenidos después del entrenamiento con la aplicación propuesta es mayor que porcentaje de aciertos inicial.

Al aplicar la prueba de rangos de Wilcoxon se obtuvo que la probabilidad de que la diferencia de aciertos obtenidos antes y después del entrenamiento con la aplicación propuesta se deba al azar es igual al 21.1%. Dado que este valor es mayor que el 5%, no existe evidencia suficiente para rechazar la hipótesis nula 6.

# Capítulo 5

## Discusión y conclusiones

Gracias al experimento realizado en este trabajo de grado y los datos obtenidos en este, se concluye que:

- Mediante el análisis de los resultados obtenidos al evaluar la memoria de los participantes del experimento, se obtiene que no hay un incremento significativo en el porcentaje de aciertos en el test de Benton, en función de la modalidad de juego; sin embargo mediante el análisis de los resultados obtenidos con la prueba de Wilcoxon y de Kruskal-Wallis, los porcentajes bajo los cuales se rechazan las hipótesis no son lejanos a un porcentaje de aceptación (menor al 5%), teniendo en cuenta que el presente estudio fue realizado con un grupo de dieciséis participantes distribuidos en grupos de cuatro por cada modalidad de juego, esto no permite tener evidencia suficiente para diferenciar los efectos de la estimulación de la memoria con las diferentes modalidades de juego diseñadas, además se utilizó un protocolo de estimulación basado en la memoria visual y un test para evaluación de la misma, limitando la investigación a esta sola variable.
- Los participantes obtuvieron un incremento en el número de aciertos en la prueba A del test de Benton sin incidencia de la modalidad de juego, pero este resultado no se observó en las pruebas B y D, esto se debe a que en la forma de aplicación A del test de Benton los participantes tienen más tiempo para visualizar las imágenes, además de reproducir esta una vez se termina el tiempo de visualización, por otra parte se debe tener en cuenta que el experimento se realiza en personas sanas, las cuales no presentan problemas en la memoria.
- Con el uso de las modalidades de juego multijugador, se logra identificar como ventaja que en este tipo de entornos los participantes se reúnen y se motivan a realizar las sesiones estipuladas, aumentando el porcentaje de permanencia en el uso de la aplicación. Además en estas se evidencia el apoyo entre jugadores lo cual favorece el flujo de la temática y la aceptación de retos sin sentirse agobiados, también permite que los jugadores planifiquen dinámicas y estrategias con el fin de manejar el vehículo rápidamente y memorizar detalladamente cada imagen.
- En el transcurso de las sesiones multijugador cooperativa y colaborativa donde los jugadores deben trabajar juntos para conseguir el objetivo del juego, se observan tres

situaciones: uno de los dos jugadores tiene mejores habilidades de conducción, mayor habilidad para recordar las imágenes o un jugador supera en las dos habilidades al otro, lo cual lleva a concluir que en algunos casos un jugador realiza mayor esfuerzo que el otro.

- En las sesiones multijugador se encuentra una estrecha relación entre el desarrollo del juego y las habilidades de socialización de las personas, puesto que dependiendo de los vínculos afectivos de los jugadores estos interactúan, prologando el tiempo de juego y a su vez interfiere en el proceso de estimulación de la memoria, debido a que se desconcentran.

## 5.1. Trabajos futuros

En esta sección se proponen otros trabajos para llevar a cabo:

- Desarrollar un juego serio multijugador donde se estimulen los diferentes tipos de memoria existentes y validar los resultados con test de atención, percepción, concentración, etc, que permitan obtener resultados en otros campos del área cognitiva.
- Elaborar estudios con un periodo de tiempo mayor, para analizar resultados en la memoria a largo plazo.
- Llevar a cabo estudios con un numero mayor de personas que cumplan con características influyentes en el experimento como: edad, sexo, nivel educativo, deficiencias cognitivas, etc.
- Realizar un estudio aplicando las modalidades de juego multijugador en participantes donde exista una relación interpersonal y otros en los que no, con el fin de analizar su incidencia en el desarrollo del juego.

# Bibliografía

- [1] A. M. Maroto, L. G. Romero, P. S. Marzo, J. M. R. Miguel, F. S. del Corral Usaola, M. R. Toro, and G. B. Zaballo, *Nuevas miradas sobre el envejecimiento*. Insero, 2009, vol. 31005.
- [2] Ó. López Locanto, “Tratamiento farmacológico de la enfermedad de alzheimer y otras demencias,” *Archivos de Medicina Interna*, vol. 37, no. 2, pp. 61–67, 2015.
- [3] M. Portela Romero, J. Pombo Romero, R. Bugarín González, M. Tasende Souto, and S. Represa Veiga, “Utilización de los inhibidores de la acetilcolinesterasa y la memantina para el tratamiento de la demencia tipo alzheimer,” *Revista española de salud pública*, vol. 79, no. 6, pp. 665–672, 2005.
- [4] J. d. J. Llibre Rodríguez and M. A. Guerra Hernández, “Enfermedad de alzheimer: Situación actual y estrategias terapéuticas,” *Revista Cubana de Medicina*, vol. 38, no. 2, pp. 134–142, 1999.
- [5] M. Franco-Martín, E. Parra-Vidales, F. González-Palau, M. Bernate-Navarro, and A. Solís, “Influencia del ejercicio físico en la prevención del deterioro cognitivo en las personas mayores: revisión sistemática,” *Rev Neurol*, vol. 56, no. 11, pp. 545–554, 2013.
- [6] A. R. Cao, I. C. Lacruz, and M. I. T. Pais, “Calidad de vida percibida por las personas mayores. consecuencias de un programa de estimulación cognitiva a través de la motricidad «memoria en movimiento»,” *Revista Española de Geriatría y Gerontología*, vol. 46, no. 2, pp. 74–80, 2011.
- [7] M. Najas, “¿ existe una alimentación adecuada?” *MANUAL DE CUIDADORES*, p. 32, 2002.
- [8] P. Medrano-Martínez and M. J. Ramos-Platón, “Alteraciones cognitivas y emocionales en el insomnio crónico,” *Rev de Neurología*, vol. 62, no. 4, pp. 170–78, 2016.
- [9] L. Berenbauma, M. Rodríguezb, S. Sambánc, S. Eguidazud, and P. Recinielloe, “Taller de estimulación de la memoria: un espacio para las representaciones y el lazo social de adultos mayores,” *Revista Debate Público*, no. 13/14, 2017.
- [10] P. Montejo, “Programa de entrenamiento de memoria para mayores con alteraciones de memoria: resultados y predictores,” *Revista española de Geriatría y Gerontología*, vol. 38, no. 6, pp. 316–326, 2003.
- [11] R. Espert Tortajada, M. d. R. Villalba Agustín *et al.*, “Estimulación cognitiva: una revisión neuropsicológica,” *Therapeia*, no. 6, pp. 73–93, 2014.

- [12] N. A. Davicino, M. S. Muñoz, L. María, and D. Donolo, "El rol psicopedagógico en la estimulación cognitiva de pacientes con demencia tipo alzheimer," *Revista Chilena de Neuropsicología*, vol. 4, no. 1, pp. 6–11, 2009.
- [13] M. Kundera, "La memoria humana," *Caracas: Banco Central de Venezuela*, 2010.
- [14] L. M. J. Madrigal, "La estimulación cognitiva en personas adultas mayores," *Revista cúpula*, pp. 4–14, 2007.
- [15] C. Valencia, E. López-Alzate, V. Tirado, M. Zea-Herrera, F. Lopera, R. Rupperecht, and W. Oswald, "Efectos cognitivos de un entrenamiento combinado de memoria y psicomotricidad en adultos mayores," *Rev Neurol*, vol. 46, no. 8, pp. 465–71, 2008.
- [16] P. G. P. Geis, M. R. C. Rubí, and M. Rubí, *Ejercicios de motricidad y memoria para personas mayores*. Editorial Paidotribo México, 2009.
- [17] V. J. Hernandez, "Metodo uclm para el entrenamiento de memoria en personas mayores." 2005.
- [18] R. M. J. CARRASCO, M. D. A. MONTES, and M. C. MARTIN, "Programas de entrenamiento de memoria. método unam," *Cuadernos de trabajo social*, vol. 14, pp. 255–278, 2001.
- [19] C. M. Carrillo and F. L. Restrepo, "Efectos de un entrenamiento cognitivo de la atención en el funcionamiento de la memoria de trabajo durante el envejecimiento," *Acta Neurológica Colombiana*, vol. 25, no. 4, pp. 244–251, 2009.
- [20] L. Joven Simón *et al.*, "Propuesta de intervención: Psicoestimulación cognitiva en pacientes con demencia leve para mantener el estado cognitivo y mejorar la calidad de vida," B.S. thesis, 2015.
- [21] E. Ostáriz Roca *et al.*, "Las terapias de estimulación cognitiva en la enfermedad de alzheimer," B.S. thesis, 2016.
- [22] M. Boada and L. Tárraga, *Volver a empezar: Ejercicios prácticos de estimulación cognitiva para enfermos de Alzheimer*. Glosa, 1999.
- [23] G. Climent-Martínez, P. Luna-Lario, I. Bombín-González, A. Cifuentes-Rodríguez, J. Tirapu-Ustárroz, and U. Díaz-Orueta, "Evaluación neuropsicológica de las funciones ejecutivas mediante realidad virtual," *Rev Neurol*, vol. 58, no. 465, p. 75, 2014.
- [24] U. Díaz Orueta, G. Climent, J. Cardas Ibáñez, L. Alonso, J. Olmo Osa, and J. Tirapu Ustárroz, "Evaluación de la memoria mediante realidad virtual: presente y futuro," *revista de Neurología*, vol. 62, no. 2, pp. 75–84, 2016.
- [25] F. Diaz, "Los juegos serios y su potencial como dispositivos educativos," *Recuperado en <http://www.eduforics.com/es/los-juegos-serios-y-su-potencial-comodispositivos-educativos>*, 2016.
- [26] C. López Raventós, "El videojuego como herramienta educativa. posibilidades y problemáticas acerca de los serious games," *Apertura (Guadalajara, Jal.)*, vol. 8, no. 1, pp. 0–0, 2016.
- [27] B. E. Marcano Lárez *et al.*, "Juegos serios y entrenamiento en la sociedad digital," 2008.
- [28] V. J. Shute, M. Ventura, and F. Ke, "The power of play: The effects of portal 2 and lumosity on cognitive and noncognitive skills," *Computers & education*, vol. 80, pp. 58–67, 2015.

- [29] S. Zickefoose, K. Hux, J. Brown, and K. Wulf, "Let the games begin: A preliminary study using attention process training-3 and lumosity™ brain games to remediate attention deficits following traumatic brain injury," *Brain injury*, vol. 27, no. 6, pp. 707–716, 2013.
- [30] H. Gil, "As tecnologias digitais e as apps: contributos e valências para idosos com demências," *International Journal of Developmental and Educational Psychology INFAD: Revista de Psicología*, vol. 2, no. 2, pp. 77–88, 2017.
- [31] M. S. Albornoz Pesántez, "Diseño gráfico de una aplicación móvil para el entrenamiento mental en jóvenes," B.S. thesis, Universidad del Azuay, 2015.
- [32] F. Pereira, S. B. i Badia, C. Jorge, and M. da Silva Cameirão, "Impact of game mode on engagement and social involvement in multi-user serious games with stroke patients," in *2019 International Conference on Virtual Rehabilitation (ICVR)*. IEEE, 2019, pp. 1–6.
- [33] F. Mueller, M. R. Gibbs, and F. Vetere, "Taxonomy of exertion games," in *Proceedings of the 20th Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat*, 2008, pp. 263–266.
- [34] J. Roschelle and S. D. Teasley, "The construction of shared knowledge in collaborative problem solving," in *Computer supported collaborative learning*. Springer, 1995, pp. 69–97.
- [35] K. Baur, A. Schättin, E. D. de Bruin, R. Riener, J. E. Duarte, and P. Wolf, "Trends in robot-assisted and virtual reality-assisted neuromuscular therapy: a systematic review of health-related multiplayer games," *Journal of neuroengineering and rehabilitation*, vol. 15, no. 1, p. 107, 2018.
- [36] I. E. Monery Saade, "Aproximación a la negociación internacional mediante el diseño de un juego serio "negotiation game", " *Escuela de Ingeniería de la Organización*, 2018.
- [37] M. El-Beheiry, G. McCreery, and C. M. Schlachta, "A serious game skills competition increases voluntary usage and proficiency of a virtual reality laparoscopic simulator during first-year surgical residents' simulation curriculum," *Surgical endoscopy*, vol. 31, no. 4, pp. 1643–1650, 2017.
- [38] E. F. Schlegel and N. J. Selfridge, "Fun, collaboration and formative assessment: Skinquization, a class wide gaming competition in a medical school with a large class," *Medical teacher*, vol. 36, no. 5, pp. 447–449, 2014.
- [39] J. L. Fernández Benítez, J. L. González Pascual, and E. Puertas Sanz, "Dietadialisisquiz. a multiplayer competitive serious game for learning during dialysis," *Enfermería Nefrológica*, vol. 20, no. 4, pp. 378–381, 2017.
- [40] K. Graham, J. Anderson, C. Rife, B. Heitmeyer, P. R. Patel, S. Nykl, A. C. Lin, and L. D. Merkle, "Cyberspace odyssey: A competitive team-oriented serious game in computer networking," *IEEE Transactions on Learning Technologies*, vol. 13, no. 3, pp. 502–515, 2020.
- [41] J. Sterman, "Interactive web-based simulations for strategy and sustainability: The mit sloan learningedge management flight simulators, part i," *System Dynamics Review*, vol. 30, no. 1-2, pp. 89–121, 2014.
- [42] S. Agudelo-Londoño, R. A. González, A. Pomares, V. Delgadillo, Ó. Muñoz, A. Cortes, and I. Gorbanev, "Revisión sistemática de juegos serios para la educación médica. rol del diseño en la efectividad," *Educación Médica Superior*, vol. 33, no. 2, 2019.

- [43] N. P. Zea, P. Paderewski, F. G. Vela, and N. M. Medina, “Una arquitectura para el desarrollo de videojuegos educativos con actividades colaborativas.”
- [44] N. Padilla-Zea *et al.*, *Metodología para el diseño de videojuegos educativos sobre una arquitectura para el análisis del aprendizaje colaborativo*. Granada: Universidad de Granada, 2011.
- [45] C. S. González, P. Toledo, S. Alayon, V. Muñoz, and D. Meneses, “Using information and communication technologies in hospital classrooms: Saveh project,” *Knowledge Management & E-Learning: An International Journal*, vol. 3, no. 1, pp. 72–83, 2011.
- [46] H. H. Morales, M. C. Plested, and J. E. A. Cobo, “El coco-game un juego de mesa para co-crear, potenciación del trabajo colaborativo y creativo,” *Enl@ ce: Revista Venezolana de Información, Tecnología y Conocimiento*, vol. 12, no. 1, pp. 57–76, 2015.
- [47] K. B. Dempsey, J. F. Brunelle, G. T. Jackson, C. Boonthum, I. B. Levinstein, and D. S. McNAMARA, “Miboard: Multiplayer interactive board game,” *arXiv preprint arXiv:1009.2206*, 2010.
- [48] D. Buchinger and M. da Silva Hounsell, “Sherlock dengue 8: A serious game for teaching about dengue fever prevention with collaboration and competition,” *XIII Simpósio Brasileiro de Jogos e Entretenimento Digital—SBGAMES*, pp. 400–409, 2014.
- [49] V. Wendel, M. Gutjahr, S. Göbel, and R. Steinmetz, “Designing collaborative multiplayer serious games,” *Education and Information Technologies*, vol. 18, no. 2, pp. 287–308, 2013.
- [50] R. Šokčić, “Elementos competitivos y cooperativos de las actividades lúdicas en la clase de español,” Ph.D. dissertation, University of Zagreb. University of Zagreb, Faculty of Humanities and Social . . . , 2020.
- [51] D. V. Rodríguez and J. M. M. Mezquita, “Diseño de juegos serios para reforzar conocimientos: una experiencia educativa en secundaria,” *Profesorado, Revista de Currículum y Formación del Profesorado*, vol. 20, no. 2, pp. 238–254, 2016.
- [52] E. F. W. Island, “Designing collaborative multiplayer serious games for collaborative learning.”
- [53] E. Tóth and A. Poplin, “Participécs—a cooperative game fostering learning about the built environment and urban planning,” in *Proceedings of the 17th AGILE Conference on Geographic Information Science*, 2014.
- [54] A. Orbe, E. Rojí, L. Garmendia, and J. Cuadrado, “Urban planning board game: A cooperative serious game,” in *European Conference on Games Based Learning*. Academic Conferences International Limited, 2018, pp. 509–XXII.
- [55] P. C. Ribeiro, B. B. P. L. de Araujo, and A. Raposo, “Comfim: a cooperative serious game to encourage the development of communicative skills between children with autism,” in *2014 Brazilian Symposium on Computer Games and Digital Entertainment*. IEEE, 2014, pp. 148–157.
- [56] S. Jacobs, “Building an education ecology on serious game design and development for the one laptop per child and sugar platforms: A service learning course builds a base for peer mentoring, cooperative education internships and sponsored research,” in *2010 2nd International IEEE Consumer Electronics Society’s Games Innovations Conference*. IEEE, 2010, pp. 1–6.

- [57] P. M. BARROS and A. G. PANQUEVA, “Juegos multiplayer: Juegos colaborativos para la educación,” *Informática Educativa, UNIANDES-LIDIE*, vol. 11, no. 2, pp. 223–239, 1998.
- [58] T. Rojo López and S. Dudu, “Los juegos serios como instrumento de empoderamiento y aprendizaje socio-laboral inclusivo,” *Revista Fuentes*, 19 (2), 95-109., 2017.
- [59] F. Juca Maldonado, M. B. García Saltos, and O. Burgo Bencomo, “Los juegos serios y su influencia en el uso responsable de energía y cuidado del medio ambiente,” *Revista Universidad y Sociedad*, vol. 9, no. 1, pp. 129–136, 2017.
- [60] J. A. Leyva Regalón and I. Mayol Céspedes, “Los juegos serios en el entrenamiento y la rehabilitación cognitiva,” *Revista Cubana de Informática Médica*, vol. 11, no. 2, pp. 140–157, 2019.
- [61] E. M. Arroyo-Anlló, J. P. Díaz-Marta, and J. Chamorro Sánchez, “Técnicas de rehabilitación neuropsicológica en demencias: hacia la ciber-rehabilitación neuropsicológica,” *Pensamiento psicológico*, vol. 10, no. 1, pp. 107–127, 2012.
- [62] I. Ayed, G. M. Alcover, P. Martínez-Bueso, J. Varona, A. Jaume-i Capó, and A. Ghazel, “Juegos serios para la prevención de caídas en personas mayores mediante el uso de dispositivos rgbd.” in *CEIG*, 2016, pp. 95–100.
- [63] C. Z. Lara, J. M. G. Calleros, J. G. García, Y. N. Rangel, and C. A. C. Ordóñez, “Guías para el desarrollo de espacios virtuales para entrenamiento cognitivo.”
- [64] C. Raymaekers, K. Coninx, and J. M. González-Calleros, “Design and engineering of game-like virtual and multimodal environments,” in *Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems*, 2010, pp. 363–364.
- [65] M. I. Ortega Canalejo, “Qsgame-tool: Herramienta para la evaluación de la calidad de juegos serios,” 2015.
- [66] E. Díaz Pérez and J. A. Flórez Lozano, “Realidad virtual y demencia,” *Revista de Neurología*, vol. 66, no. 10, pp. 344–352, 2018.
- [67] V. MANUEL and Á. PATO, “Estrategia de dificultad adaptativa para un videojuego de dos jugadores.”
- [68] S. Domènech Pou, *Aplicación de un programa de estimulación de memoria a enfermos de Alzheimer en fase leve*. Universitat de Barcelona, 2004.
- [69] G. Ruiz-Manrique, K. Tajima-Pozo, and F. Montañes-Rada, “Case report: ‘adhd trainer’: the mobile application that enhances cognitive skills in adhd patients,” *F1000Research*, vol. 3, 2014.
- [70] T. Wallace and J. T. Morris, “Development and testing of eyeremember: A memory aid app for wearables for people with brain injury,” in *International Conference on Computers Helping People with Special Needs*. Springer, 2018, pp. 493–500.
- [71] D. A. CASTILLO ARMIJOS, “Aplicación móvil de juegos mentales para mantener la estimulación cognitiva en adultos mayores.” B.S. thesis, Quito, 2018.
- [72] H. López de la Nieta Marín, “Mindproblem: desarrollo de una aplicación para la mejora de la memoria en personas de avanzada edad.” B.S. thesis, Universitat Politècnica de Catalunya, 2018.

- [73] M. S. Villanea, "Neuropsicología y envejecimiento: el potencial de la memoria autobiográfica en investigación e intervención clínica," *Revista Costarricense de Psicología*, vol. 31, no. 1, pp. 123–146, 2012.
- [74] P. F. Romeu, "Recuerdo de imágenes emocionales y niveles de procesamiento," *Psicothema*, vol. 14, no. 3, pp. 591–596, 2002.
- [75] N. Justel, M. Psyrdellis, and E. Ruetti, "Modulación de la memoria emocional: una revisión de los principales factores que afectan los recuerdos," *Suma psicológica*, vol. 20, no. 2, pp. 163–174, 2013.
- [76] L. Morales-Maure, O. García-Marimón, A. Torres-Rodríguez, and A. Lebrija-Trejos, "Habilidades cognitivas a través de la estrategia de aprendizaje cooperativo y perfeccionamiento epistemológico en matemática de estudiantes de primer año de universidad," *Formación universitaria*, vol. 11, no. 2, pp. 45–56, 2018.
- [77] R. G. Figueroa, C. J. Solís, and A. A. Cabrera, "Metodologías tradicionales vs. metodologías ágiles," *Universidad Técnica Particular de Loja, Escuela de Ciencias de la Computación*, vol. 9, pp. 1–10, 2008.
- [78] O. T. Gómez, P. P. R. López, and J. S. Bacalla, "Criterios de selección de metodologías de desarrollo de software," *Industrial data*, vol. 13, no. 2, pp. 70–74, 2010.
- [79] J.-P. Subra and A. Vannieuwenhuyse, *Scrum: un método ágil para sus proyectos*. Ediciones ENI., 2018.
- [80] P. Deemer, G. Benefield, C. Larman, and B. Vodde, "Información básica de scrum," *California: Scrum Training Institute*, 2009.
- [81] A. Retamosa Santos, "Scrum. aplicación del método ágil en la gestión de proyectos," B.S. thesis, 2015.
- [82] A. F. Ordoñez. Aplicación de scrum en la construcción de un simulador de redis.
- [83] S. Belli and C. L. Raventós, "Breve historia de los videojuegos," *Athenea Digital. Revista de pensamiento e investigación social*, no. 14, pp. 159–179, 2008.
- [84] A. A. Parsons, Thomas D y Rizzo, "Validación inicial de un entorno virtual para la evaluación del funcionamiento de la memoria: prueba de evaluación del rendimiento cognitivo de realidad virtual," *CyberPsychology Behavior*.
- [85] R. G. Knight, N. Titov, and M. Crawford, "The effects of distraction on prospective remembering following traumatic brain injury assessed in a simulated naturalistic environment," *Journal of the International Neuropsychological Society*, vol. 12, no. 1, pp. 8–16, 2006.
- [86] G. Plancher, V. Gyselinck, S. Nicolas, and P. Piolino, "Age effect on components of episodic memory and feature binding: A virtual reality study." *Neuropsychology*, vol. 24, no. 3, p. 379, 2010.
- [87] A. Leis, M. Iturry, C. M. Serrano, and R. F. Allegri, "Habilidades atencionales asociadas a un mal desempeño en la conducción vehicular del adulto mayor," *Neuropsicología Latinoamericana*, vol. 7, no. 3, 2015.
- [88] A. Rey, "Rey. test de copia de una figura compleja," *TEA ediciones, Madrid*, 2009.
- [89] J. L. B. Vidal and E. R. Campos, "El test de retención visual de benton en lesionados cerebrales adultos," *Quaderns de psicología*, pp. 19–35, 1985.

- [90] A. Canchala, C. de Uso, S. de Uso, D. d. C. de Uso, D. de Estructura Estática, D. de Interacción, and D. de Estados, “Uml, ejemplo sencillo sobre modelado de un proyecto,” *Developer Network*, 2004.
- [91] A. T. Jiménez, “Uml.”
- [92] Ó. Santiesteban González *et al.*, “Top run: Juego de plataformas multijugador,” 2020.
- [93] G. Lee Kâm Châk, “Melilac: desarrollo de un videojuego de combate en tercera persona con opción de multijugador en línea,” Ph.D. dissertation, Universitat Politècnica de València, 2021.
- [94] goldenflies, “Mercedes benz gls 580 2020 modelo 3d.”
- [95] U. Technologies. (8) Standard assets (for unity 2018.4).
- [96] Ada<sub>K</sub>ing.(4)*Freetrees*.
- [97] M. Bentosela and A. E. Mustaca, “Efectos cognitivos y emocionales del envejecimiento: aportes de investigaciones básicas para las estrategias de rehabilitación,” *Interdisciplinaria*, vol. 22, no. 2, pp. 211–235, 2005.
- [98] J. M. R.-S. de León, “Estimulación cognitiva en el envejecimiento sano, el deterioro cognitivo leve y las demencias: estrategias de intervención y consideraciones teóricas para la práctica clínica,” *Revista de logopedia, foniatría y audiología*, vol. 32, no. 2, pp. 57–66, 2012.
- [99] A. L. Benton, “A visual retention test for clinical use,” *Archives of Neurology & Psychiatry*, vol. 54, no. 3, pp. 212–216, 1945.
- [100] “Aprender y comprender la prueba de análisis de varianza por rangos unidireccional de kruskal-wallis para las diferencias entre tres o más grupos independientes.”
- [101] V. Berlanga and M. J. Rubio Hurtado, “Clasificación de pruebas no paramétricas. cómo aplicarlas en spss,” *REIRE. Revista d’Innovació i Recerca en Educació*, 2012, vol. 5, num. 2, p. 101-113, 2012.

# Anexos

# Anexo A

## Documentación software:

A continuación se describen los *Scripts* mas importantes utilizados en la programación del juego serio *Brain Race*:

### Programación multijugador

Para la programación de las modalidades multijugador es necesario entender que la librería de *mirror* implementa dos acciones remotas: la primera son comandos que se llaman desde el cliente y se ejecutan en el servidor, y la segunda consiste en llamar desde el servidor y se ejecuta en el cliente. A continuación se presentan las etiquetas que permiten configurar el cliente y el servidor en *mirror*:

#### Command [Command]:

Para invocar una función en el servidor, es necesario utilizar la etiqueta [Command] antes de definir la función, además todas las funciones deben tener como nombre el prefijo **Cmd** para identificar que son funciones del servidor, si no se cumplen estos requisitos la función no se ejecutará en el servidor.

#### ClientRpc [ClientRpc]:

Esta etiqueta es utilizada para llamar a una función en todos los clientes al mismo tiempo. Para definirla se debe añadir la etiqueta [ClientRpc] antes de declarar la función y posteriormente definir el nombre de la función con el prefijo **Rpc**.

#### Server [Server]

Mediante esta etiqueta se define el comportamiento de una función para que se ejecute solo, y únicamente en el servidor, aunque un cliente llame a esta función, solo se ejecutara en el servidor.

- **EntrarVr()** Esta corrutina se conecta con la librería **UnityEditor.XR.Management**, entra en sus configuraciones, las instancia e inicia el modo de realidad virtual.

```
1 IEnumerator EntrarVr() {
    yield return XRGeneralSettings.Instance.Manager.
        InitializeLoader();
3     XRGeneralSettings.Instance.Manager.StartSubsystems();
    }
```

- **OnDestroy():** Esta función es utilizada para conectarse con la librería **UnityEditor.XR.Management** y detener la realidad virtual instanciada.

```

private void OnDestroy()
2   {
      if(XRGeneralSettings.Instance.Manager.isInitializationComplete)
      {
4       XRGeneralSettings.Instance.Manager.StopSubsystems();
        XRGeneralSettings.Instance.Manager.DeinitializeLoader();
6     }
  }

```

- **Controlador individual:** Cada vez que se entre en la escena de la modalidad individual se ejecutara este código. El cual se encarga de tomar aleatoriamente imágenes de una lista de imágenes y cargarlas en una lista tipo *sprite*, estas imágenes se cargan como texturas dentro del juego y serán las que debe memorizar el jugador.

```

1 private void Awake()
  {
3     if (Controlador.multiplayer)
        return;
5     Sprite [] imagenes = Resources.LoadAll<Sprite>("Imágenes");
        Debug.Log("Cargadas " + imagenes.Length + " Imgs");
7     int numImgs = Controlador.dificultad == 1 ? 5 : Controlador.
        dificultad == 2 ? 7 : 9;
        string loaded = "";
9     while (listaImgs.Count < numImgs)
        {int num = Random.Range(0, imagenes.Length);
11         if (!listaImgs.Contains(imagenes[num]))
            {
13             listaImgs.Add(imagenes[num]);
                loaded += imagenes[num].name + ", ";
15         }
        }
17     StartCoroutine(Inicializar());
  }

```

- **Void start:** Se llama a *Start* en el marco cuando se habilita un *script* justo antes de que se llame por primera vez a cualquiera de los métodos *Update*. También tiene un condicional donde se van contando los *checkpoint* o metas que se atraviesan con el vehículo, el encargado de hacer esto es el servidor. una vez se atraviesan los *checkpoint* se lleva un contador para guardar el registro, se desactiva el *checkpoint* que se atravesó y se activa el siguiente.

```

void Start(){
2     imagenes = Resources.LoadAll<Sprite>("Imágenes"); //Carga las
        imagenes
        Debug.Log("Cargadas " + imagenes.Length + " Imgs");//Muestra el
        numero de cargadas
4
        miniM = FindObjectOfType<Minimapa>();
6
        if(isServer
8         while (indices.Count < 10){
            int num = Random.Range(0, imagenes.Length);
10            if (!indices.Contains(num))
                indices.Add(num);
12        }

```

```

14         listaChecks.AddRange(checkpoints.GetComponentsInChildren<
15             CheckpointColaborativo>(true));
16
17         Debug.Log($"Indices {indices.Count} Chekcs {listaChecks.Count}"
18             );
19         for (int i = 0; i < 10; i++){
20             listaImgs.Add(imagenes[indices[i]]);
21             listaChecks[i].SetImagen(imagenes[indices[i]]);
22             listaChecks[i].gameObject.SetActive(false);
23         }
24         if (isClientOnly)
25             textoConteo.text = $"Conectado, iniciando con {indices.
26                 Count} indices";
27     }

```

- **OnCoopilotoConectado:** Esta función se llama desde el cliente para notificar que se conecto en la partida, para ello envía un *true*.

```

1     public void OnCoopilotoConectado(bool espera){
2         if(espera)
3             textoConteo.text = "Conectado";
4         else
5             RpcActivarPclient(Random.Range(0, listaChecks.Count));
6     }

```

- **Checkpoint:** se encarga de destruir el *checkpoint* atravesado, desde el cliente y registra el conteo.

```

2     public void Checkpoint(CheckpointColaborativo c){
3         Debug.Log("Llamado a checkpoint " + c.name);
4
5         if (!isServer)
6             return;
7         DestruirCheckPclient(listaChecks.IndexOf(c));
8     }
9
10 }

```

- **DestruirCheckPclient:** se encarga de destruir el *checkpoint* atravesado desde el servidor y registra el conteo.

```

1     [ClientRpc]
2     void DestruirCheckPclient(int c){
3         Debug.Log("Llamado a destruir el indice " + c);
4         Destroy(listaChecks[c].gameObject);
5         listaChecks.RemoveAt(c);
6         if (isServer)
7             if (listaChecks.Count > 0)
8                 {
9                     RpcActivarPclient(Random.Range(0, listaChecks.Count));
10                    //Le manda a decir al cliente que lo agregue
11                }
12            else

```

```

12 |         {
13 |             Debug.Log("entrooooo finalizar" + Controlador.
14 |                 tiempoCarrera.ToString("F1"));
15 |             NetworkManager.singleton.ServerChangeScene("
16 |                 Preguntascolab"); }
17 |     }
18 |     [ClientRpc]
19 |     void RpcActivarPclient(int i) {
20 |         Debug.Log($"Llamado a RPCactivarP con {i} faltan {listaChecks.
21 |             Count}");
22 |         listaChecks[i].gameObject.SetActive(true);
23 |     }
24 | }

```

- **OnFinish():** Una vez se atraviesa el ultimo *checkpoint*, se carga la escena de preguntas.

```

1 [Command(requiresAuthority =false)]
2 public void OnFinish()
3 {
4     NetworkManager.singleton.ServerChangeScene("
5         Preguntascooperativas");
6 }

```

- **Void Start preguntas colaborativas:** Se llama a *Start* en el marco cuando se habilita un *script* justo antes de que se llame por primera vez a cualquiera de los métodos *Update*. En esta función *Start* se desactiva la funcionalidad de los botones para el servidor.

```

1 void Start()
2 {
3     if (isServer && Controlador.modo == Controlador.Modos.
4         colaborativo)
5     {
6         foreach (Button botones in FindObjectsOfType<Button>())
7             Destroy(botones);
8     }
9 }

```

- **Respondido():** En esta función se sincronizan las preguntas del servidor y el cliente, además se guarda la respuesta de cada jugador.

```

1 public void Respondido(int i)
2 {
3     if (Controlador.modo == Controlador.Modos.colaborativo){
4         Sinc_opciones(i);
5         preguntascolab.Respuesta(i); }
6
7     else {
8         if (isServer)
9             if (respExterna) Rpc_Contestando();
10            else { Rcp_Sinc_Contestado_cliente();
11                respPropia = true;
12                valorResp = i;
13            }
14        if (isClientOnly)
15            if (respExterna) Cmd_Contestando();

```

```

14 |         else { Cmd_Sinc_Contestado_servidor();
16 |             respPropia = true;
18 |             valorResp = i;
20 |         }
    |     }
    | }

```

- **SincroPreguntas:** Estas funciones se encargan cargar las respuestas del cliente y visualizar la respuesta seleccionada en la interfaz del cliente.

```

public class SincroPreguntas : NetworkBehaviour
2 {
    [Command(requiresAuthority = false)]
4     public void Cmd_Sinc_Contestado_servidor()
    {
6         respExterna = true;
    }
    [Command(requiresAuthority = false)]
8     public void Cmd_Contestando()
10    {
12        preguntascolab.Respuesta(valorResp);
14        respPropia = false;
16        valorResp = 0;
18        respExterna = false;
20    }
    [ClientRpc]
22    public void Rcp_Sinc_Contestado_cliente()
    {
24        respExterna = true;
    }
    [ClientRpc]
26    public void Rpc_Contestando()
    {
28        preguntascolab.Respuesta(valorResp);
30        respPropia = false;
32        valorResp = 0;
34        respExterna = false;
    }
    [Command(requiresAuthority = false)]
    public void Sinc_opciones(int i)
    {
        preguntascolab.Respuesta(i);
        Debug.Log("sincro");
    }
}

```

- **SincroPreguntasCoop:** Las siguientes funciones permiten que el cliente y el servidor contesten las preguntas, además de que cada uno visualice las mismas preguntas, las mismas opciones de respuesta y la elegida.

```

public class SincroPreguntasCoop : NetworkBehaviour
2 {
    public bool respExterna = false;
4     int valorResp;
    public preguntascooperativas preguntasCoop;

```

```

6      public List<Image> botones;

8      private void Start(){
9          preguntasCoop = FindObjectOfType<preguntascooperativas>();
10     }

12     public void Contestando(int i){
13         valorResp = i; //Se almacena la respuesta
14         if (respExterna){
15             OnAmbosRespondidos();
16         }
17         else {
18             OnPrimeroEnResponder(i);
19         }
20     }

22     void OnAmbosRespondidos() { //Se llama cuando respuesta
23         Externa es true
24         preguntasCoop.Respuesta(valorResp); //Se manda la respuesta
25         propia
26
27         if (isServer) RpcResponder(); //Se le dice al otro que
28         mande la suya
29         else CmdResponder();
30
31         respExterna = false;
32         ResetBotones();
33     }

34     void OnPrimeroEnResponder(int i) { //Se llama cuando respuesta
35         Externa es false
36         ResetBotones(); //Se resetean el color de
37         los botones
38         botones[i - 1].color = Color.cyan; //Se marca la respuesta
39
40         if (isServer) Rcp_Sinc_Contestado(i); //Se marca que se
41         respondi
42         else Cmd_Sinc_Contestado(i);
43     }

44     [Command(requiresAuthority = false)]
45     public void Cmd_Sinc_Contestado(int i){
46         ResetBotones();
47         botones[i - 1].color = Color.cyan;
48         respExterna = true;
49     }

50     [ClientRpc]
51     public void Rcp_Sinc_Contestado(int i){
52         if (isServer) return;
53         ResetBotones();
54         botones[i - 1].color = Color.cyan;
55         respExterna = true;
56     }

```

```

56 |
57 |     [Command(requiresAuthority = false)]
58 |     public void CmdResponder(){
59 |         preguntasCoop.Respuesta(valorResp);
60 |         ResetBotones();
61 |     }
62 |
63 |     [ClientRpc]
64 |     public void RpcResponder(){
65 |         if (isServer) return;
66 |
67 |         preguntasCoop.Respuesta(valorResp);
68 |         ResetBotones();
69 |     }
70 |
71 |     void ResetBotones(){
72 |         foreach (Image boton in botones)
73 |             boton.color = Color.white;
74 |     }
75 | }

```

- **Preguntas:** En este código se crea una lista de preguntas y respuestas las cuales se organizan de acuerdo al orden de la lista de imágenes.

```

1 | public class Preguntas : MonoBehaviour
2 | {
3 |     public GameObject obPregs, obOrden, obimagenes, obFinal;
4 |     public static int correctas = 0, incorrectas = 0;
5 |     public bool verdadero = true;
6 |     public int bandera = 1;
7 |     public int banderacompetitiva = 0;
8 |     public int [] resultados;
9 |     public float tiempoRespuestas;
10 |
11 |     List<Pregunta> pregs = new List<Pregunta>();
12 |     public Text textoPreg;
13 |     public Text textoCorrectas, textoIncorrectas;
14 |     public Text [] opcs;
15 |     public TextMeshProUGUI textoFinal;
16 |
17 |     List<Pregunta> utilizadas = new List<Pregunta>();
18 |
19 |     public class Pregunta
20 |     {
21 |         public string texto, nombreTextura;
22 |         public string [] opciones;
23 |         public int correcta;
24 |
25 |         public Pregunta(string textura, int c, string text, string []
26 |             rtas)
27 |         {
28 |             texto = text;
29 |             nombreTextura = textura;
30 |             correcta = c;
31 |             opciones = rtas;

```

```
33     }
35     void Start()
36     {
37         if (Random.Range(0, 3) == 1){
38             MezclarLista();
39             verdadero = false;
40         }
41
42         agregarPreguntas();
43         BuscarPreguntas();
44         StartCoroutine(MostrarPreg());
45     }
46
47     public void Respuesta(int opc)
48     {
49         if (opc == utilizadas[0].correcta)
50         { correctas++; textoCorrectas.text = "Correctas: " + correctas;
51           }
52         else
53         { incorrectas++; textoIncorrectas.text = "Incorrectas: " +
54           incorrectas; }
55
56         utilizadas.RemoveAt(0);
57
58         if (utilizadas.Count > 0)
59             StartCoroutine(MostrarPreg());
60         else
61         {
62             SecuenciaPreguntas();
63             obPregs.SetActive(false);
64             obOrden.SetActive(true);
65         }
66     }
67
68     IEnumerator MostrarPreg()
69     {
70         textoPreg.text = utilizadas[0].texto;
71         opcs[0].text = utilizadas[0].opciones[0];
72         opcs[1].text = utilizadas[0].opciones[1];
73         opcs[2].text = utilizadas[0].opciones[2];
74         opcs[3].text = utilizadas[0].opciones[3];
75         foreach (Text o in opcs)
76             o.transform.parent.gameObject.SetActive(false);
77         yield return new WaitForSeconds(1);
78         foreach (Text o in opcs)
79             o.transform.parent.gameObject.SetActive(true);
80     }
81
82     void BuscarPreguntas()
83     {
84         if(Controlador.multiplayer)
85             foreach (Sprite text in SincronizadorImgs.listaImgs)
86             {
87                 foreach (Pregunta preg in pregs)
88                 {
89                     if (preg.text == text)
90                     {
91                         utilizadas.Add(preg);
92                         pregs.Remove(preg);
93                     }
94                 }
95             }
96     }
97 }
```

```

87         if (text.name == preg.nombreTextura)
88             {
89                 utilizadas.Add(preg);
90             }
91     else
92     foreach (Sprite text in ControlImgs.listaImgs)
93     {
94         foreach (Pregunta preg in pregs)
95             if (text.name == preg.nombreTextura)
96                 {
97                     utilizadas.Add(preg);
98                 }
99     }
100 }
101
102
103 void SecuenciaPreguntas() {
104     if (!Controlador.multiplayer)
105     foreach (Image im in obimagenes.GetComponentInChildren<
106         Image>()) {
107         if (ControlImgs.listaImgs.Count > im.transform.
108             GetSiblingIndex())
109             { im.sprite = ControlImgs.listaImgs[im.transform.
110                 GetSiblingIndex()];
111             }
112         else im.gameObject.SetActive(false);
113     }
114     else
115     foreach (Image im in obimagenes.GetComponentInChildren<
116         Image>())
117     {
118         if (SincronizadorImgs.listaImgs.Count > im.transform.
119             GetSiblingIndex())
120             {
121                 im.sprite = SincronizadorImgs.listaImgs[im.
122                     transform.GetSiblingIndex()];
123             }
124         else im.gameObject.SetActive(false);
125     }
126 }
127
128 void MezclarLista()
129 {
130     if (!Controlador.multiplayer)
131     for (int i=0; i< Random.Range(4, 6); i++)
132     {
133         int a = Random.Range(0, ControlImgs.listaImgs.Count);
134         ControlImgs.listaImgs.Add(ControlImgs.listaImgs[a]);
135         ControlImgs.listaImgs.RemoveAt(a);
136     }
137     else
138     for (int i = 0; i < Random.Range(4, 6); i++)
139     {
140         int a = Random.Range(0, SincronizadorImgs.listaImgs.
141             Count);

```

```

135         SincronizadorImgs.listaImgs.Add(SincronizadorImgs.
           listaImgs[a]);
           SincronizadorImgs.listaImgs.RemoveAt(a);
137     }
139 }
140
141 void PuntajeFinal()
142 {
143     if (Controlador.multiplayer)
144     {
145         FindObjectOfType<PuntajeFinal>().OnFinalizado(correctas,
           incorrectas);
146         return;
147     }
148
149     obOrden.SetActive(false);
150     obFinal.SetActive(true);
151     tiempoRespuestas = Time.timeSinceLevelLoad;
152     textoFinal.text = "Tiempo de carrera: " + Controlador.
           tiempoCarrera.ToString("F2");
153     textoFinal.text += "Seg \nTiempo de respuesta: " +
           tiempoRespuestas.ToString("F2");
           textoFinal.text += "Seg \nRespuestas correctas: " + correctas;
           //textoFinal.text += "\nPorcentaje correctas: " + (correctas/(
           correctas+incorrectas)*100).ToString("F2");
155     textoFinal.text += "\nDificultad: ";
156
157     switch (Controlador.dificultad)
158     {
159         case 2:
160             textoFinal.text += "Medio";
161             break;
162         case 3:
163             textoFinal.text += "Difícil";
164             break;
165         default:
166             textoFinal.text += "Facil";
167             break;
168     }
169 }
170
171 public void RespuestaSecuencia(bool resp) {
172     if (verdadero == resp) {
173         bandera = 0;
174         correctas++;
175         textoCorrectas.text = "Correctas: " + correctas;
176     }
177     if (verdadero != resp) {
178         bandera = 0;
179         incorrectas++;
180         textoIncorrectas.text = "Incorrectas: " + incorrectas;
181     }
182 }
183
184 //if (bandera==0) { Debug.Log("banderaaa"); Write_txt(+
           correctas+";"+incorrectas); }

```

```
185 //Add - Panel de tiempo y correctas
186 PuntajeFinal();
187 }
188
189
190
191 void agregarPreguntas()
192 {
193     string [] respuestas;
194     string texto;
195
196     respuestas = new string [] { "Ninguna", "3", "4", "Muchas" };
197     texto = " Cuantas manchas ten a el Venado de las im genes?";
198     pregs.Add(new Pregunta("Venado", 4, texto, respuestas));
199
200     respuestas = new string [] { "Rama", "Carretera", "Esta volando",
201         "Arbusto" };
202     texto = " Sobre qu estaba el tuc n de las im genes?";
203     pregs.Add(new Pregunta("Tucan", 1, texto, respuestas));
204
205     respuestas = new string [] { "Cafe", "Negra", "Blanco y negro",
206         "blanca" };
207     texto = " Qu colores ten a la vaca que viste en las
208         im genes?";
209     pregs.Add(new Pregunta("Vaca", 3, texto, respuestas));
210
211     respuestas = new string [] { "1", "2", "3", "4" };
212     texto = " Cuntas jirafas aparecan en la imagen de este
213         animal?";
214     pregs.Add(new Pregunta("Jirafa", 1, texto, respuestas));
215
216     respuestas = new string [] { "Dormido", "jadeando", "muerto", "
217         sentado" };
218     texto = " En qu estado se encontraba el perro de las
219         im genes?";
220     pregs.Add(new Pregunta("Perro", 2, texto, respuestas));
221
222     respuestas = new string [] { "Azul", "Negro", "Negro y rojo", "
223         Cafe" };
224     texto = " Qu colores ten a la araa que viste en las
225         im genes?";
226     pregs.Add(new Pregunta("Araa", 3, texto, respuestas));
227
228     respuestas = new string [] { "Futboll", "Basketball", "Rugby", "
229         Volleyball" };
230     texto = " A qu deporte pertenece el bal n que viste en las
231         im genes?";
232     pregs.Add(new Pregunta("Balon", 1, texto, respuestas));
233
234     respuestas = new string [] { "1", "2", "3", "mas de 3" };
235     texto = " Cuantos bananos aparecan en la imagen de esta
236         fruta?";
237     pregs.Add(new Pregunta("Banano", 4, texto, respuestas));
238
239     respuestas = new string [] { "Fundido", "Apagado", "Roto", "
```

```
Encendido" };
231 texto = " Cul era el estado del bombillo de las im genes?";
pregs.Add(new Pregunta("Bombilla", 2, texto , respuestas));

233 respuestas = new string [] { "Acostado", "Dormido", "Parado", "
    No habia un burro" };
235 texto = " Cul era la posici n del burro de las im genes?";
pregs.Add(new Pregunta("Burro", 3, texto , respuestas));

237 respuestas = new string [] { "No aparecia candados", "Solamente
    1", "2", "3" };
239 texto = " Cuantos candados aparec an en la imagen de este
    objeto? ";
pregs.Add(new Pregunta("Candado", 2, texto , respuestas));

241 respuestas = new string [] { "Cangrejo", "Escorpion", "Mantis
    religiosa", "Langosta" };
243 texto = " En las im genes que viste , que animal ten a
    tenazas?";
pregs.Add(new Pregunta("Cangrejo", 1, texto , respuestas));

245 respuestas = new string [] { "Botas", "Guantes", "Casco", "
    Rodilleras" };
247 texto = " En las im genes aparec a un elemento de
    protecci n para los motociclistas cu l es? ";
pregs.Add(new Pregunta("Casco", 3, texto , respuestas));

249 respuestas = new string [] { "Lata", "Botella", "Jarra", "Vaso"
    };
251 texto = " En las im genes aparec a una cerveza , en qu
    presentaci n estaba?";
pregs.Add(new Pregunta("Cerveza", 4, texto , respuestas));

253 respuestas = new string [] { "1", "2", "3", "mas de 3" };
255 texto = " Cuntos cocodrilos aparec an en la imagen de este
    animal?";
pregs.Add(new Pregunta("Cocodrilo", 1, texto , respuestas));

257 respuestas = new string [] { "Negro", "Gris", "Blanco", "Negro y
    blanco" };
259 texto = " Qu colores ten a el conejo que viste en las
    im genes?";
pregs.Add(new Pregunta("Conejo", 3, texto , respuestas));

261 respuestas = new string [] { "si , un dibujo a blanco y negro ",
    "no", "si , una foto del organo", "si , una ilustracion
    animada" };
263 texto = " En las im genes que visualizaste , encontraste un
    coraz n?";
pregs.Add(new Pregunta("Corazon", 4, texto , respuestas));

265 respuestas = new string [] { "1", "2", "3", "6" };
267 texto = " Cuntos crayones hab a en la imagen de este objeto
    ? ";
pregs.Add(new Pregunta("Crayones", 4, texto , respuestas));
```

```
269 respuestas = new string [] { "1", "2", "3", "mas de 3" };
    texto = " Cuntas cucarachas hab a en la imagen de este
          animal?";
271 pregs.Add(new Pregunta("Cucarachas", 1, texto , respuestas));

273 respuestas = new string [] { "Madera", "Plastico", "Metal", "
    Cristal" };
    texto = " De qu material era la cuchara que aparec a en las
          im genes?";
275 pregs.Add(new Pregunta("Cuchara", 3, texto , respuestas));

277 respuestas = new string [] { "Negro", "Amarillo", "Gris", "
    Manchado" };
    texto = " Qu colores ten a el gato que viste en las
          im genes?";
279 pregs.Add(new Pregunta("Gato", 3, texto , respuestas));

281 respuestas = new string [] { "Doble carne", "Doble queso", "
    Tocineta", "Tenia lechuga y tomate" };
    texto = " La hamburguesa de las im genes ten a algo
          diferente? ";
283 pregs.Add(new Pregunta("Hamburguesa", 4, texto , respuestas));

285 respuestas = new string [] { "Estaba vacia", "Limonada", "
    Gaseosa", "Agua" };
    texto = " En las im genes aparec a una jarra , que liquido
          ten a adentro?";
287 pregs.Add(new Pregunta("Jarra", 2, texto , respuestas));

289 respuestas = new string [] { "Cafe", "Negra", "Gris", "Blanca"
    };
    texto = " Qu colores ten a la llama que viste en las
          im genes?";
291 pregs.Add(new Pregunta("Llama", 4, texto , respuestas));

293 respuestas = new string [] { "1", "2", "3", "mas de 3" };
    texto = " Cuntas manzanas aparec an en la imagen de esta
          fruta?";
295 pregs.Add(new Pregunta("Manzanas", 4, texto , respuestas));

297 respuestas = new string [] { "Azul", "Cafe", "Verde", "Naranja"
    };
    texto = " Qu colores ten a la mariposa que viste en las
          im genes?";
299 pregs.Add(new Pregunta("Mariposa", 4, texto , respuestas));

301 respuestas = new string [] { "Martillo", "Pala", "Metro", "
    Destornillador" };
303 texto = " En las im genes aparec a una herramienta de
          construccion , cu l era?";
305 pregs.Add(new Pregunta("Martillo", 1, texto , respuestas));

307 respuestas = new string [] { "Lata", "Desgranada", "Cocinada", "
    Sin cascara" };
```

```
309     texto = " En qu  presentaci n estaban las mazorcas de las
        im genes?";
    pregs.Add(new Pregunta("Mazorca", 3, texto , respuestas));

311
313     respuestas = new string [] { "Colgado de una rama", "Acostado",
        "Parado", "Sentado" };
    texto = " en qu  posici n estaba el simio de las im genes?"
        ;
    pregs.Add(new Pregunta("Mono", 1, texto , respuestas));

315
317     //Revisar Imagen
    respuestas = new string [] { "Partidas por la mitad", "Peladas",
        "Normal", "Rodajas" };
    texto = " En qu  forma estaban las naranjas de las im genes?"
        ";
    pregs.Add(new Pregunta("Naranja", 4, texto , respuestas));

321
323     respuestas = new string [] { "Olla", "Sarten", "Estufa", "
        Licuadora" };
    texto = " En las im genes aparec a un objeto de la cocina ,
        cu l era?";
    pregs.Add(new Pregunta("Olla", 1, texto , respuestas));

325
327     respuestas = new string [] { "Sin decoracion", "Velas", "Velas
        de Feliz cumplea os", "Decoraci n con crema" };
    texto = " En las im genes aparec a un pastel, ten a alg n
        tipo de decoraci n?";
    pregs.Add(new Pregunta("Pastel", 2, texto , respuestas));

329
331
333     respuestas = new string [] { "1", "2", "3", "mas de 3" };
    texto = " Cuntas  pi as aparec an en la imagen de esta
        fruta?";
    pregs.Add(new Pregunta("Pi a", 2, texto , respuestas));

335
337     respuestas = new string [] { "Escopeta", "Fusil", "Rifle", "
        Pistola" };
    texto = " En las im genes aparec a un arma, que tipo era?";
    pregs.Add(new Pregunta("Pistola", 4, texto , respuestas));

339
341     //Revisar
    respuestas = new string [] { "Pepperoni y queso", "Pollo", "
        Champi ones", "Pi a" };
    texto = " En las im genes aparec a una pizza, que
        ingredientes ten a?";
    pregs.Add(new Pregunta("Pizza", 1, texto , respuestas));

343
345
347     respuestas = new string [] { "Cafe", "Amarilla", "Negra", "
        Blanca" };
    texto = " Qu  colores ten a la Gallina que viste en las
        im genes?";
```

```
349     pregs.Add(new Pregunta("Gallinas", 1, texto, respuestas));
351
352     respuestas = new string[] { "Naranja", "Amarilla", "Azul", "
353         Verde" };
354     texto = " Qu  colores ten a el sapo que viste en las
355         im genes?";
356     pregs.Add(new Pregunta("Rana", 4, texto, respuestas));
357
358     respuestas = new string[] { "Arena", "Sol", "Digital", "Analogo
359         " };
360     texto = " Qu  tipo de reloj era el que aparec a en las
361         im genes?";
362     pregs.Add(new Pregunta("Reloj", 4, texto, respuestas));
363
364     respuestas = new string[] { "Tierra", "Arena", "Matera", "No
365         habian cactus" };
366     texto = " En que estaba plantado el cactus de las imagenes?";
367     pregs.Add(new Pregunta("Cactus", 3, texto, respuestas));
368
369     respuestas = new string[] { "Rinoceronte", "Unicornio", "Toro",
370         "Bambi" };
371     texto = " Habia un animal con un cuerno en las imagenes, cual
372         era?";
373     pregs.Add(new Pregunta("Rinoceronte", 1, texto, respuestas));
374
375     respuestas = new string[] { "2", "4", "6", "8" };
376     texto = " Cuantas patas tenia el saltamontes de las im genes?
377         ";
378     pregs.Add(new Pregunta("Saltamontes", 3, texto, respuestas));
379
380     respuestas = new string[] { "una Partida y una entera", "Entera
381         ", "Sin cascara", "No habia sandia" };
382     texto = " De qu  forma estaba la sand a de las im genes?";
383     pregs.Add(new Pregunta("Sandia", 1, texto, respuestas));
384
385     respuestas = new string[] { "Rojo", "Amarillo", "Naranja", "
386         Verde" };
387     texto = " En que color estaba el semaforo de las imagenes?";
388     pregs.Add(new Pregunta("Semaforo", 4, texto, respuestas));
389
390     respuestas = new string[] { "Roja", "Negra", "Rosada", "Azul
391         " };
392     texto = " De que color era la rosa de las imagenes?";
393     pregs.Add(new Pregunta("Rosa", 1, texto, respuestas));
394
395     respuestas = new string[] { "Verde", "Naranjas", "Rojos", "
396         Negros" };
397     texto = " De que color eran los tomates que habian en las
```

```
393     imagenes?";
pregs.Add(new Pregunta("Tomate", 3, texto, respuestas));

395     respuestas = new string [] { "Negro", "Rojo", "Blanco", "Azul"
397     };
texto = " De que color es el carro que aparece en las imagenes
?";
pregs.Add(new Pregunta("Volvo", 3, texto, respuestas));

399

401     respuestas = new string [] { "Amarilla", "Negra", "Blanca", "
Azul" };
texto = " Qu colores ten a la silla que viste en las
im genes?";
403     pregs.Add(new Pregunta("Silla", 1, texto, respuestas));

405

407     respuestas = new string [] { "Negro", "Rojo", "Azul", "Naranja"
};
texto = " Qu colores ten a el tel fono que viste en las
im genes?";
409     pregs.Add(new Pregunta("Telefono", 2, texto, respuestas));

411     respuestas = new string [] { "Trucha", "Atun", "Tiburon", "
Salmon" };
texto = " En las imagenes que viste habia un pescado, que
clase de pez era";
413     pregs.Add(new Pregunta("Tiburon", 3, texto, respuestas));

415

417     //new
419     respuestas = new string [] { "Roja", "Azul", "Verde", "Negra" };
texto = " En las imagenes que viste habia una bicicleta, de
que color era?";
421     pregs.Add(new Pregunta("bicicleta", 1, texto, respuestas));

423     respuestas = new string [] { "Volando", "Comiendo", "Dormido", "
Observando" };
texto = " En las imagenes que viste habia una buho, que estaba
haciendo?";
425     pregs.Add(new Pregunta("buh", 4, texto, respuestas));

427     respuestas = new string [] { "Rojo", "Verde", "Azul", "Ninguno"
};
texto = " En las imagenes que viste habia una circulo, de que
color era?";
429     pregs.Add(new Pregunta("circulo", 2, texto, respuestas));

431     respuestas = new string [] { "Estrella de mar", "Estrella fugaz"
, "Estrella dibujada", "Ninguna" };
texto = " En las imagenes que viste habia una estrella, de que
tipo era?";
```

```
433     pregs.Add(new Pregunta("estrellademar", 1, texto, respuestas));
435     respuestas = new string[] { "Naranja", "Gris", "Negro", "Rojo"
        };
        texto = " En las imagenes que viste habia una globo, de que
            color era?";
437     pregs.Add(new Pregunta("globorojo", 4, texto, respuestas));
439     respuestas = new string[] { "Azul", "Roja", "Amarilla", "Verde"
        };
        texto = " En las imagenes que viste habia una letra C, de que
            color era?";
441     pregs.Add(new Pregunta("letracroja", 2, texto, respuestas));
443     respuestas = new string[] { "Azul", "Roja", "Amarilla", "Negra"
        };
        texto = " En las imagenes que viste habia una letra p, de que
            color era?";
445     pregs.Add(new Pregunta("letrapnegra", 4, texto, respuestas));
447     respuestas = new string[] { "verde", "Roja", "Amarilla", "Negra"
        };
        texto = " En las imagenes que viste habia una moto, de que
            color era?";
449     pregs.Add(new Pregunta("motoverde", 1, texto, respuestas));
451     respuestas = new string[] { "blanco", "Rojo", "Amarillo", "
        Verde" };
        texto = " En las imagenes que viste habia un numero 2, de que
            color era?";
453     pregs.Add(new Pregunta("numero2verde", 1, texto, respuestas));
455     respuestas = new string[] { "blanco", "Rojo", "Amarillo", "
        Verde" };
        texto = " En las imagenes que viste habia un numero 5, de que
            color era?";
457     pregs.Add(new Pregunta("numero5rojo", 2, texto, respuestas));
459     respuestas = new string[] { "blanco", "Rojo", "Amarillo", "
        Verde" };
        texto = " En las imagenes que viste habia un numero 7, de que
            color era?";
461     pregs.Add(new Pregunta("numero7amarillo", 3, texto, respuestas)
        );
463     respuestas = new string[] { "blanco", "Rojo", "Amarillo", "
        Verde" };
        texto = " En las imagenes que viste habia un triangulo, de que
            color era?";
465     pregs.Add(new Pregunta("triangulo-rojo", 2, texto, respuestas)
        );
467
469     respuestas = new string[] { "blanco", "Rojo", "Amarillo", "
        Verde" };
        texto = " En las imagenes que viste habia un vaso, de que
```

```

        color era?";
        pregs.Add(new Pregunta("vasoverde", 4, texto, respuestas));
471     }
473
475 }

```

- **InputHelper:** su funcionalidad consiste en recibir los valores enviados por el *bluetooth* e interpretarlos como movimientos del auto.

```

1  public class InputHelper : MonoBehaviour
3  {
5      public static float direccion, aceleracion;
6      public static bool freno;
7      public Text textoblue;
8      BT bt;
9
10     private void Start()
11     {
12         bt = GetComponent<BT>();
13     }
14
15     void Update()
16     {
17         if (Application.platform == RuntimePlatform.Android)
18             getBT();
19         else
20             getEditor();
21     }
22
23     private void getEditor()
24     {
25         direccion = Input.GetAxis("Horizontal");
26         aceleracion = Input.GetKey("w") ? 1 : Input.GetKey("s") ? -1 :
27             0;
28         freno = Input.GetKey("space");
29     }
30
31     private void getBT()
32     {
33         direccion = bt.hor*0.006f;
34         aceleracion = bt.acelerador? 1 : 0;
35         freno = bt.freno;
36         //textoblue.text = $"dir{direccion} ac {aceleracion} fr {freno}";
37     }
38 }

```

- **IniciarCarrera():** LA funcionalidad de este código consiste en los movimientos del vehículo.

```

2  void IniciarCarrera(){ canMove = true;}
    void FinCarrera() {

```

```
4      canMove = false;
5      foreach (WheelCollider llanta in llantas)
6          llanta.brakeTorque = 8000;
7  }
8
9  void Update()
10 {
11     if (canMove)
12         foreach (WheelCollider llanta in llantas){
13             llanta.motorTorque = InputHelper.aceleracion * 600;
14
15             if (InputHelper.freno)
16             {
17                 if (GetRpm() < 20) { llanta.motorTorque = -400;
18                     llanta.brakeTorque = 0; }
19                 else llanta.brakeTorque = 8000;
20             }
21             else llanta.brakeTorque = 0;
22         }
23
24     // quitar la sensibilidad a la direccion
25     llantas[0].steerAngle = InputHelper.direccion * 5;
26     llantas[1].steerAngle = InputHelper.direccion * 5;
27 }
```

# Anexo B

## Manual de usuario

En la siguiente sección se explicara como instalar y utilizar el juego serio Brain race:

- **Instalación:** En esta sección se explicara a detalle cuales son los pasos a seguir para instalar la aplicación Brain race desde un dispositivo android.

Antes de instalar esta aplicación se deben de tener en cuenta algunos requisitos del hardware como: un sistema operativo android versión 10.0 o superiores, las dimensiones de la pantalla deben ser como mínimo 1512 x 720 pixeles, debe de contar con un espacio en memoria interna de mínimo 100 megabytes y para su ejecución la memoria RAM debe ser de mínimo 2 gigabytes. También debe de contar con modulo bluethoot, modulo wiffi y giroscopio.

**Pasos de instalación** A continuación se describen los pasos de instalación:

- Descargar el instalador presionando sobre el *APK* enviado, ver figura B.1.

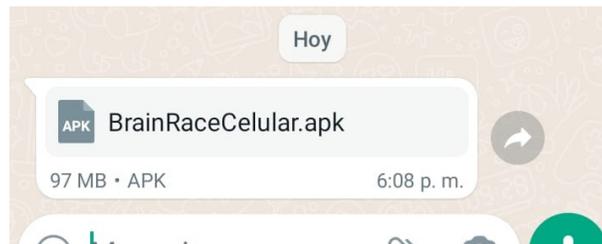
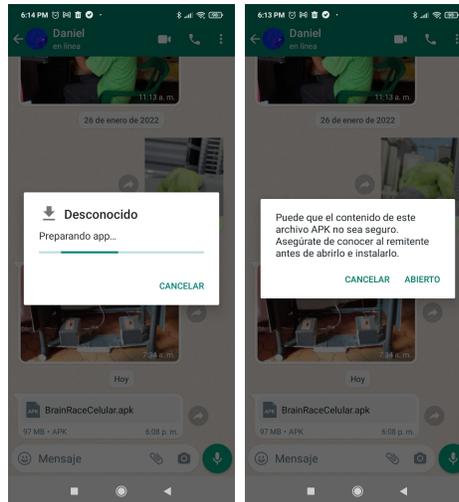


Figura B.1: APK Brain Race.

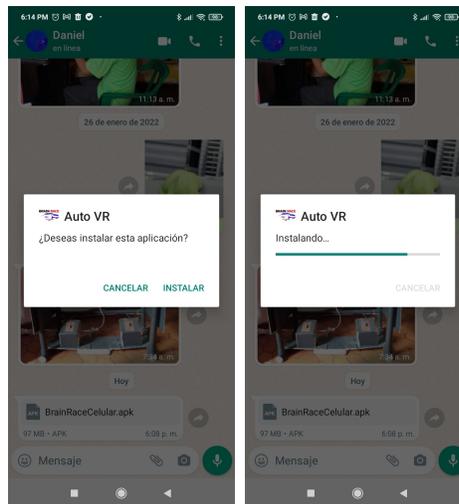
- Cuando se descargue y se presione sobre el instalador, saldrá un aviso notificando que la procedencia del instalador es desconocida, dentro de esta ventana aparecen dos opciones y se debe elegir la denominada abierto, ver figura B.2.



(a) Aceptación de la procedencia de la app. (b) Aviso de notificación de la procedencia de la app.

Figura B.2: Procedencia desconocida.

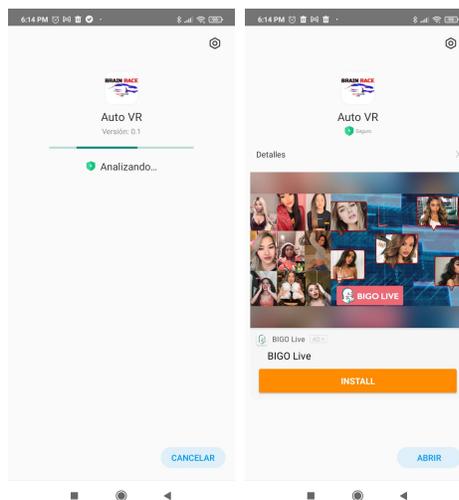
- Después de aceptar que la procedencia del archivo es confiable, saldrá una ventana emergente notificando si se desea instalar esta aplicación, para esto aparecen dos opciones donde se debe elegir instalar, ver figura B.3.



(a) Mensaje de instalación. (b) Carga de la instalación.

Figura B.3: Instalación app.

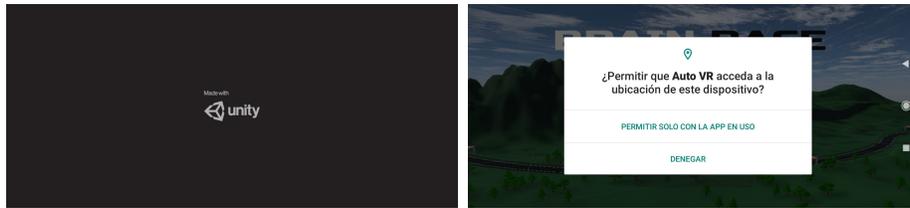
- Una vez terminada la instalación, se procede a abrir la aplicación, ver figura B.4.



(a) Analizando app. (b) Abrir app.

Figura B.4: Abrir aplicación.

- Una vez abierta la aplicación saldrá una ventana emergente solicitando ¿Permite que Auto VR acceda a la ubicación de este dispositivo? se debe elegir "Permitir solo con app en uso", esta opción es necesaria puesto que de ello depende que tenga los permisos necesarios para funcionar en el dispositivo, ver figura B.5. Después de realizar estos pasos ya se puede utilizar la aplicación Brain Race desde el dispositivo. Es necesario aclarar que los mensajes en las ventanas emergentes pueden variar según el dispositivo donde se instale, sin embargo los pasos a seguir son los mismos.



(a) Ventana emergente mientras carga la aplicación.

(b) Mensaje de permisos.

Figura B.5: Ventanas inicio app.

- **Uso de la aplicación** En esta sección se explicara como utilizar la aplicación *Brain Race*.

**Pantalla de registro** Cuando se ingrese en la aplicación se encontrará con una pantalla de registro ver figura B.6, esta cuenta con dos cuadros de texto donde se para ingresar el nombre y cédula del jugador, una vez ingresados se debe presionar el botón de registrar.



Figura B.6: Pantalla de registro.

**Pantalla principal** Después de presionar el botón de registrar, se encontrará con un menú de modalidades donde se debe elegir cual utilizar ver figura B.7, dependiendo de la modalidad seleccionada se deben realizar ciertos pasos, a continuación se explican cada uno de acuerdo a la opción elegida.



Figura B.7: Menú principal

- **Modalidad individual:** En esta modalidad un solo jugador debe de conducir por una pista memorizando las imágenes que se encuentra en el camino. Al finalizar el

recorrido debe contestar una serie de preguntas sobre el contenido de las imágenes que visualizo.

- Para ingresar en esta modalidad se debe presionar sobre la primera opción, denominada "individual", ver figura B.8



Figura B.8: Elegir individual

- Dentro de la modalidad individual se encuentra un menú para elegir el nivel de dificultad, en éste se encuentran las siguientes opciones: nivel fácil donde aparecerán cinco imágenes para memorizar, nivel medio donde aparecen siete imágenes y el difícil donde aparecen nueve, en la figura B.10 se muestra la interfaz.



Figura B.9: Elegir individual

- Después de elegir el nivel de dificultad se ingresa en la partida, dentro de está el dispositivo activa de manera automática el *bluetooth* y la realidad virtual. A continuación el usuario debe instalar el celular en las gafas de realidad virtual y cuando este listo para comenzar debe de presionar el acelerador y comenzara la partida.

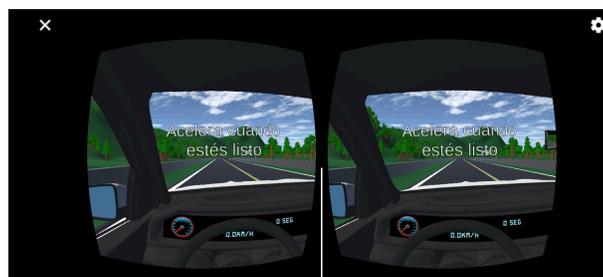


Figura B.10: Elegir individual

- **Modalidad colaborativa:** en esta modalidad cada jugador tiene asignada una tarea y deberán trabajar juntos para alcanzar el objetivo del juego. Mientras un jugador se encarga de conducir el vehículo, el otro deberá guiarlo por la pista para encontrar las imágenes que deben de memorizar. al finalizar el copiloto es el encargado de contestar las preguntas.
  - Para ingresar en esta modalidad se debe presionar sobre la segunda opción, denominada "colaborativo", ver figura B.11.



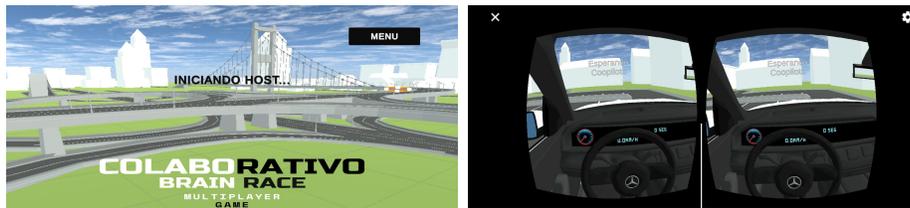
Figura B.11: Elegir colaborativo

- El siguiente paso es elegir el rol bajo el cual se quiere jugar. El piloto sera el encargado de conducir el vehículo, mientras el copiloto debe observar el mapa y guiar hacia donde están ubicadas las imágenes que se deben de memorizar, ver figura B.12.



Figura B.12: Elegir rol piloto o copiloto

- El usuario que eligió la opción piloto es el encargado de crear la partida y debe esperar a que se conecte el copiloto, mientras esto ocurre debe de instalar el dispositivo en las gafas de realidad virtual, ver figuras B.13.

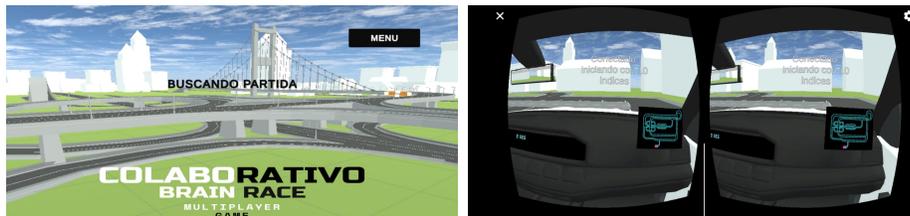


(a) Mensaje iniciando host.

(b) Esperando copiloto.

Figura B.13: Rol piloto.

- El usuario que eligió la opción copiloto debe buscar la partida creada por el primer jugador, mientras esto ocurre debe de instalar el dispositivo en las gafas de realidad virtual, ver figuras B.14.



(a) Mensaje buscando partida.

(b) Inicia el juego.

Figura B.14: Rol copiloto.

- Cuando estén conectados los dos jugadores comienza la partida. En esta modalidad se debe de conducir el vehículo por en medio de las metas, debido a que estas se encuentran ocultas y solo aparecerá la siguiente si se atraviesa la actual, ver figura B.15.

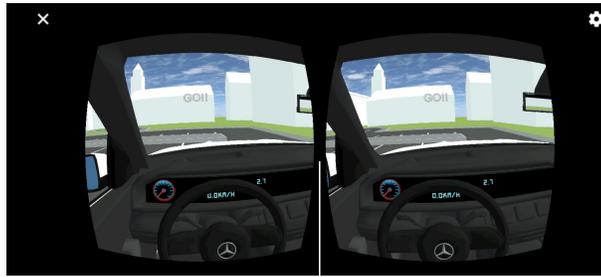


Figura B.15: Comienza la partida

- **Modalidad cooperativa:** en esta modalidad cada jugador tiene un vehículo y deberán acompañarse a lo largo del recorrido de la pista, memorizando las imágenes entre los dos, para que al finalizar el recorrido juntos contesten las preguntas.
  - Para ingresar en esta modalidad se debe presionar sobre la tercera opción, denominada "cooperativo", ver figura B.16.



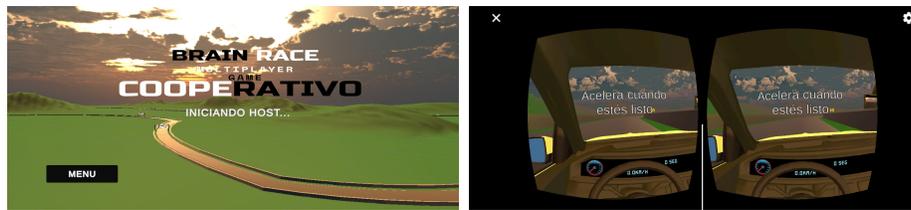
Figura B.16: Elegir cooperativo

- El siguiente paso es, elegir el jugador con el cual se quiere ingresar en la partida. En esta modalidad se tiene la opción jugador uno y jugador dos, ambos tienen las mismas funciones, ver figura B.17.



Figura B.17: Elegir jugador uno o jugador dos.

- El usuario quien eligió la opción jugador uno es el encargado de crear la partida y debe esperar a que se conecte el segundo jugador, mientras esto ocurre debe de instalar el dispositivo en las gafas de realidad virtual, ver figuras B.18.

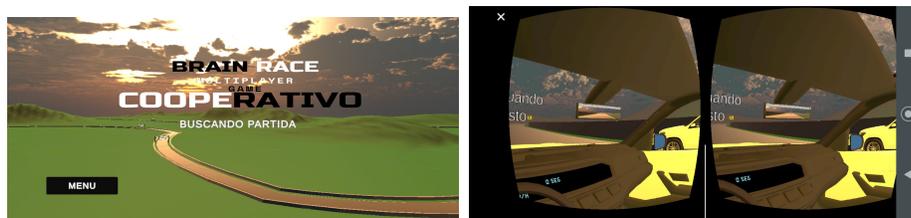


(a) Mensaje iniciando host.

(b) Inicia el juego.

Figura B.18: Jugador uno modalidad cooperativa.

- El usuario quien eligió la opción jugador dos debe buscar la partida, mientras esto ocurre debe de instalar el dispositivo en las gafas de realidad virtual, ver figuras B.23.



(a) Mensaje iniciando host.

(b) Inicia el juego.

Figura B.19: Jugador dos modalidad cooperativa.

- **Modalidad competitiva:** en esta modalidad cada jugador tiene un vehículo, con el cual deben recorrer la pista y memorizar las imágenes distribuidas a lo largo de esta. Al finalizar el recorrido cada jugador debe contestar una serie de preguntas de acuerdo a el contenido de las imágenes vistas. Ganara quien menor tiempo tenga en la pista y mayor numero de respuestas correctas tenga.
  - Para ingresar en esta modalidad se debe presionar sobre la cuarta opción, denominada "competitivo", ver figura B.20.



Figura B.20: Elegir competitivo

- El siguiente paso es, elegir el jugador con el cual se quiere ingresar en la partida. En esta modalidad se tiene la opción jugador uno y jugador dos, ambos tienen las mismas funciones, ver figura B.21.



Figura B.21: Elegir jugador uno o jugador dos.

- El usuario quien eligió la opción jugador uno es el encargado de crear la partida y debe esperar a que se conecte el segundo jugador, mientras esto ocurre debe de instalar el dispositivo en las gafas de realidad virtual, ver figuras B.22.



(a) Mensaje iniciando host.

(b) Inicia el juego.

Figura B.22: Jugador uno modalidad competitiva.

- El usuario quien eligió la opción jugador dos debe buscar la partida, mientras esto ocurre debe de instalar el dispositivo en las gafas de realidad virtual, ver figuras B.23.



(a) Mensaje iniciando host.

(b) Inicia el juego.

Figura B.23: Jugador dos.

- **conducción** En esta sección se explicara como conducir el vehículo virtual.

- **Acelerar:** Para acelerar el vehículo se debe de presionar el pedal izquierdo de la cabina de conducción o mover el joystick hacia arriba, como se puede apreciar en la figura B.24.



(a) Pisar pedal.

(b) Mover el joystick hacia arriba.

Figura B.24: Acelerar.

- **Frenar:** Para frenar el vehículo se debe de presionar el pedal derecho de la cabina de conducción o mover el joystick hacia abajo, como se puede apreciar en la figura B.25.



(a) Pisar pedal.

(b) Mover el joystick hacia abajo.

Figura B.25: Frenar.

- **Retroceder:** Para retroceder el vehículo se debe de mantener presionado el pedal derecho de la cabina de conducción o mantener el joystick hacia abajo, como se puede apreciar en la figura B.26.



(a) Pisar pedal.

(b) Mover el joystick hacia abajo.

Figura B.26: Retroceder.

- **Girar:** Para girar la dirección del vehículo se debe de girar el volante de la cabina de conducción o mover el joystick hacia los costados, como se puede apreciar en la figura B.27.



(a) Mover volante.

(b) Mover el joystick hacia la izquierda.

(c) Mover el joystick hacia Derecha.

Figura B.27: Girar el vehículo.