

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA
ESCUELA DE SISTEMAS



DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO DE SISTEMAS Y COMPUTACIÓN

“DESARROLLO DE UN VIDEOJUEGO MÓVIL DE REALIDAD
VIRTUAL PARA LA VISUALIZACIÓN EN EL DISPOSITIVO GOOGLE
CARDBOARD.”

NOMBRE:

ANDRÉS SEBASTIÁN BASTIDAS FIERRO

DIRECTOR:

ING. ANDRÉS JIMENEZ

QUITO, 2016

Contenido

Justificación	5
Antecedentes	7
Capítulo 1.....	8
Análisis del entorno actual de videojuegos de Realidad Virtual	8
1.1. Estado actual del desarrollo de videojuegos.....	8
1.2 Estado actual de la tecnología de la realidad virtual	9
1.3 Principales videojuegos móviles de realidad virtual.....	13
Capítulo 2.....	18
Documento de diseño del juego	18
2.1. Descripción general	18
2.1.2. Estilo visual	20
2.1.3. Estilo de audio	20
2.2 Interfaces de Usuario.....	21
2.2.1. Menú principal – selección	21
2.2.2. Inicio del juego y pantalla de introducción	21
2.2.3. HUD y menús dentro del juego	22
2.2.4. Pantalla de pérdida	22
2.2.5. Nivel	23
2.3. Jugabilidad	26
2.3.1. Mecánicas	26
2.3.2. Controles	27
2.3.3. Modos	27
2.3.4. Al ganar	27
2.3.5. Puntaje	27
Capítulo 3.....	28
Desarrollo del prototipo	28
3.1. Modelación tridimensional de assets	28
3.1.1. Modelación tridimensional desde cero	28
3.1.2. Edición de assets existentes	30
3.1.3. Creación de esqueletos	32
3.2. Entorno de desarrollo Unity3D	32
3.3 Codificación de funciones en los lenguajes C# y Javascript.....	34
3.3.1. Generación de obstáculos aleatorios	34
3.3.2. Aumento del puntaje	35
3.3.3. Pérdida al impactar obstáculos	35

3.3.4. Kit de desarrollo de software “GoogleVR”	36
3.3.5. Movimiento del jugador	38
3.3.6. FPS. Eliminación e instancias de escenarios no visibles.	39
Capítulo 4.....	42
Pruebas del videojuego.....	42
4.1 Caso de prueba (C1)	43
4.2. Caso de prueba (C2)	44
4.3. Caso de prueba (C3)	45
Capítulo 5.....	46
Conclusiones y recomendaciones.....	46
5.1. Conclusiones.....	46
5.2. Recomendaciones.....	47
Bibliografía.....	49

Índice de Tablas

Tabla 1. Lista de Assets utilizados en el desarrollo del videojuego.....	32
Tabla 2. Dispositivos de realidad virtual tipo Cardboard.....	43
Tabla 3. Celulares utilizados para casos de prueba.....	44

Índice de Figuras

Figura 1. Tennis For Two. Considerado el primer videojuego. Fuente: Museum of Electronic Games & Art	8
Figura 2. Espada de Damocles. Primer dispositivo de VR. Fuente: A head-mounted three dimensional display, Ivan E. Sutherland, 1968	11
Figura 3. HTC Vive, Oculus Rift, PlayStation VR Fuente: techinsider.io	12
Figura 4. Google Daydream. Dispositivo y Aplicaciones. Fuente: Google VR Daydream.....	14
Figura 5. Bocetos De creación de Personaje Elaborados en: Adobe Illustrator. Autor: Andrés Bastidas F.	18
Figura 6. Boceto Conceptual. Ecuador. Elaborado en: Adobe Illustrator. Autor: Andrés Bastidas F.	19
Figura 7. Diseños Conceptuales de Lava Surf. Cordillera del Ecuador y Monumento Mitad del Mundo. Elaborado en: Blender. Autor: Andrés Bastidas F.....	20
Figura 8. Diseño Conceptual de Menú Principal de Lava Surf. Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.....	21
Figura 9. Distribución Conceptual del Área del Juego Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.	22
Figura 10. Diseño Conceptual de Interfaz de Usuario. Elaborado en: Blender. Autor: Andrés Bastidas F.	22
Figura 11. Pantalla de Pérdida y Menú Exterior Principal Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.	23
Figura 12. Distribución Conceptual de la Visión del Jugador. Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.....	23

Figura 13. Diseño Conceptual Islas Galápagos Elaborado en: Blender. Autor: Andrés Bastidas F.	24
Figura 14. Diseño Conceptual Región Costa Elaborado en: Blender. Autor: Andrés Bastidas F...	24
Figura 15. Diseño Conceptual Volcán en Región Sierra. Elaborado en: Blender. Autor: Andrés Bastidas F.	24
Figura 16. Diseño Conceptual Región Amazonía Elaborado en: Blender. Autor: Andrés Bastidas F.	25
Figura 17. Diseño Conceptual Región Sierra Elaborado en: Blender. Autor: Andrés Bastidas F. .	25
Figura 18. Diferenciación entre Área de Juego o Área de Cielo Elaborado en: Blender. Autor: Andrés Bastidas F.	25
Figura 19. Diseño Conceptual de Área del Cielo. Elaborado en: Blender. Autor: Andrés Bastidas F.	26
Figura 20. Distribución Conceptual de Carriles y Obstáculos Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.	26
Figura 21. Distribución Conceptual de Carriles y Obstáculos Alternativos Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.	27
Figura 22. Distribución Conceptual Objetos Invisibles con Eventos de Aproximación Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.	27
Figura 23. Modelado de Plano, Subdivisión, Randomización y Suavizado Elaborado en: Blender. Autor: Andrés Bastidas F.	29
Figura 24. Modelado basado en la silueta del mapa político del Ecuador Fuente: presidencia.gob.ec Elaborado en: Blender Autor: Andrés Bastidas F.	30
Figura 25. Comparación entre Asset descargado de biblioteca de recursos con forma de Caballo y edición propia con nuevas formas y colores para asemejarse a una Llama. Elaborado en: Blender. Autor: Andrés Bastidas F.	31
Figura 26. Esqueleto de Cóndor. Animación de Vuelo. Elaborado en: Blender. Autor: Andrés Bastidas F.	32
Figura 27. Comparación de Animación: Mono, Brazos Cerrados y Mono Saltando en el Aire, Brazos Abiertos Elaborado en: Unity 3D. Autor: Andrés Bastidas F.	33
Figura 28. Código para generación aleatoria de obstáculos. Elaborado en: MonoDevelop. Autor: Andrés Bastidas F.	34
Figura 29. Diferencia entre la posición y forma de los obstáculos generados aleatoriamente. Elaborado en: Unity3D. Autor: Andrés Bastidas F.	35
Figura 30. Código para aumento de metros recorridos y Visualización Tridimensional del texto. Elaborado en: Unity3D. Autor: Andrés Bastidas F.	35
Figura 31. Código para detección de colisiones y cambio de escena. Elaborado en: MonoDevelop. Autor: Andrés Bastidas F.	36
Figura 32. Colliders visibles de obstáculo y personaje. Elaborado en: Unity3D. Autor: Andrés Bastidas F.	36
Figura 33. Par de Cámaras separadas horizontalmente y Visualización de ambas en una sola pantalla. Elaborado en: Unity3D. Autor: Andrés Bastidas F.	37
Figura 34. Objetos rodeando al personaje posteriormente invisibles. Disparan un código distinto si son observados. Elaborado en: Unity 3D . Autor: Andrés Bastidas F.	38
Figura 35. Funciones y parámetros, verdadero o falso, a dispararse cuando el apuntador ingresa o sale del objeto. Elaborado en: Unity3D. Autor: Andrés Bastidas F.	38
Figura 36. Código para movilización del personaje según la vista del usuario. Elaborado en: MonoDevelop. Autor: Andrés Bastidas F.	39
Figura 37. Objetos Visibles y su función de movilizar al personaje a distintas posiciones. Elaborado en: Unity3D. Autor: Andrés Bastidas F.	39
Figura 38. La dirección de la vista del usuario nos muestra una instancia de cada región, eliminándola posteriormente. Elaborado en: Unity3D. Autor: Andrés Bastidas F.	41
Figura 39. Diferencias entre Cardboard V1.0 (Construido de forma casera) y V2.0 (comprado)...	42
Figura 40. Celulares inteligentes utilizados. Lenovo A5000 y LG G2	43
Figura 41. Lenovo A5000. La falta de Giroscopio impide la jugabilidad. Autor: Andrés Bastidas F.	44
Figura 42. Prueba en LG G2, sin instancias dinámicas del escenario. 23 FPS. Autor: Andrés Bastidas F.	44
Figura 43. Prueba en LG G2, instancias dinámicas del escenario. 43, 36 Y 29 FPS. Autor: Andrés Bastidas F.	45

Justificación

Existiendo el concepto de VR¹ desde finales del anterior siglo, su temprano fracaso se debió a la falta de algunos factores claves que ahuyentaron a sus usuarios como lo son la poca portabilidad de los dispositivos, causando incomodidad para su uso, tecnologías de procesamiento de ambientes tridimensionales demasiado débiles, y un costo elevado de los materiales de fabricación en general.²

Con el aumento de la población mundial que poseen teléfonos inteligentes, dispositivos electrónicos dedicados al entretenimiento y consolas de videojuegos, la demanda por nuevas tecnologías por parte de las masas ha causado un rápido desarrollo en el área de tecnología personal y móvil.

A su vez, la curiosidad por la capacidad de la tecnología cuyo precio se mantenga en un rango aceptable para el consumo del usuario promedio, ha llevado a una adaptación cada vez con mayor facilidad a nuevos conceptos tecnológicos que brindan experiencias de inmersión, como lo evidencia la conquista de la visualización de películas de 3D estereoscópicas en el campo del cine.

Para experimentar la VR, se requiere procesar y renderizar ambientes tridimensionales con el doble de potencia que los tradicionales, dedicando una escena distinta para cada uno de nuestros ojos, por lo cual la VR se ramifica en dos grandes metodologías de procesamiento según el dispositivo de visualización a utilizarse:

- Dispositivos de visualización que utilizan el procesador del computador.
- Dispositivos de magnificación de las pantallas de teléfonos inteligentes.

Los métodos de procesamiento en la actualidad se encuentran lo suficientemente avanzados para demostrar ambientes de carácter realista, en el caso del computador, o de carácter estilizado, en el caso de los teléfonos inteligentes, a diferencia de los procesadores de la década de 1990 que sólo ofrecían escenas de objetos planos.

¹ Realidad Virtual por sus siglas en inglés, "Virtual Reality".

² (Omicrono, 2014)

Digi-Capital, compañía suiza dedicada a la consultoría, inversión y administración del sector de entretenimiento digital, ha avalado al concepto de VR como un campo de inversión que costará cerca de \$150 mil millones en los próximos 5 años.³ Junto a ella, la estadounidense investigadora de tecnologías de la información Gartner, sitúa a la Realidad Virtual como un concepto iniciando su “Cuesta del Esclarecimiento”⁴ y a 5 o 10 años de llegar a la llamada “Meseta de la Productividad”.^{5 6}

Por lo tanto, el desarrollo de software especializado en experiencias de VR es un campo potencial de rentabilidad elevada, con proyectos creativos que interesarán a la juventud y usuarios con facilidad de uso de la tecnología.

³ (Digi-Capital, 2015)

⁴ Categoría utilizada por la empresa Gartner para las ideas que pueden ser ampliamente aceptadas por el público con producción por parte los distribuidores de tecnología. Del inglés “Slope of Enlightenment”.

⁵ Categoría utilizada por la empresa Gartner para las ideas que son completamente aceptadas por el público y para las cuales se desarrollan claros criterios de producción. Del inglés “Plateau of Productivity”.

⁶ (Gartner, 2015; Gartner, Gartner Hype Cycle, s.f.)

Antecedentes

Si bien el término “realidad virtual” fue inicialmente mencionado por el actor francés Antonin Artaud al describir al teatro como “la réalité virtuelle” en el libro *Le Théâtre et son doublé* (1938)⁷ su inicio de popularidad no se dio hasta la década de 1990 gracias al apareamiento de películas futuristas que tocaron el tema, los fallidos proyectos basados en visión estereoscópica y pantallas LCD de las compañías de videojuegos Nintendo y Sega, así como de lo que sería el software precursor del ahora exitoso mundo virtual “Second Life”.

Un avance verdaderamente significativo de la VR no se dio hasta el año 2013 en que la exitosa campaña de financiación colectiva “Oculus Rift” recolectó \$2.5 millones para su desarrollo prototipo. Posterior a esto se reportó la compra de Oculus VR, LLC por parte de la compañía Facebook con un elevado precio de \$2 mil millones en marzo del 2014.⁸

Desde entonces, junto a la revelación de la producción del dispositivo de reconocimiento de movimiento llamado “Oculus Touch”, brindando conceptos similares para VR destinada al uso casero, durante el año 2015 se suman a la apuesta los gigantes tecnológicos como lo son Google con el dispositivo de construcción casera Google Cardboard, Samsung con el producto compatible con sus celulares de alta gama llamado Gear VR, Sony con su producto Project Morpheus, HTC en conjunto con Valve Corporation con el producto HTC Vive, incluyendo sus propios controles de reconocimiento de movimiento, entre otros.

⁷ (Artaud, 1958)

⁸ (Gamerant, 2014)

Capítulo 1

Análisis del entorno actual de videojuegos de Realidad Virtual

En el presente capítulo se analizará el contexto, variantes, tecnología y casos de éxito que rodean a la producción de videojuegos en VR.

Se examinará las metodologías y estándares utilizados para la planificación del desarrollo de videojuegos en general. Además, se va a exponer la tecnología y principales dispositivos disponibles para procesar y visualizar ambientes de VR, junto a un listado de los principales videojuegos específicamente desarrollados para plataformas móviles.

1.1. Estado actual del desarrollo de videojuegos

Desde la creación del primer videojuego de la historia “Tennis for Two” en 1958,⁹ pasando por uno de los peores golpes que ha sufrido este mercado con la Crisis de los Videojuegos ocurrida en 1983,¹⁰ hasta el desarrollo e investigación de juegos dedicados exclusivamente a tratamiento terapéuticos y medicinales,¹¹ la opinión pública con respecto a este medio de entretenimiento ha atravesado por algunos altibajos.



*Figura 1. Tennis For Two. Considerado el primer videojuego.
Fuente: Museum of Electronic Games & Art*

A pesar de las asperezas de las primeras décadas, ahora podemos observar un concepto mucho más positivo con respecto a los videojuegos, pues más del 60% de las

⁹ (BNL, s.f.)

¹⁰ (Oxford, 2011)

¹¹ (AAMC, 2014)

familias en los Estados Unidos tienen por lo menos un miembro que juega videojuegos durante más de 3 horas semanales.¹²

Aparte de la opinión que se tiene sobre la utilización de este medio de entretenimiento, de igual forma se puede analizar la opinión sobre la creación de dichos juegos, obteniendo métricas financieras que comprueban su estado como un negocio rentable y de escalabilidad en la industria. En el 2015 en EEUU, el desarrollo de videojuegos generó más de \$23.5 billones en retorno cuando se consideran las ventas del software, hardware y accesorios.¹³

El desarrollo de videojuegos se enfoca en la recreación de ambientes que le presentan al jugador un número de posibilidades y limitaciones, así como un objetivo a lograr como recompensa de seguir correctamente estas reglas.

Si bien las matemáticas y conocimientos computacionales son una prioridad a la hora de la emulación de distintas realidades contenidas en sus propias leyes físicas, el desarrollo de este producto en la actualidad necesita de variadas áreas de conocimiento, tanto artísticas como técnicas, pues no solo el atractivo visual de los objetos con los cuales va a interactuar el jugador es igual de importante a que la programación del juego sea de calidad. Con un equipo multifacético no solo se logra evitar fallas inesperadas que atentan a la continuidad o inmersión del juego, sino también que gracias a la calidad de audio con música correctamente seleccionada para cada género, o inclusive a la consistencia de la historia y deseos de los personaje, de forma que, inclusive si se presentan cosas que pueden ser irreales en nuestro mundo, a causa de la suspensión de la incredulidad se estaría logrando exitosamente que nuestro cerebro las interprete como reales y por lo tanto entretenidas.¹⁴ De igual forma, se ha comprobado la importancia y estrecha relación entre la educación y el entrenamiento del estudiante, causando que si se siente a gusto y divertido, pueda retener más información que con el estudio tradicional.¹⁵

1.2 Estado actual de la tecnología de la realidad virtual

¹² (Entertainment Software Association, 2016)

¹³ (The NPD Group, 2015)

¹⁴ (Ludosofía, s.f.)

¹⁵ (Pixel, 2015)

A diferencia del desarrollo de videojuegos tradicional, el cual simplemente se encargar de mostrarle al usuario entornos y mundos ficticios, la realidad virtual, o más conocida en la jerga tecnológica como VR (siglas provenientes de su nombre en inglés: Virtual Reality), busca causar que el usuario tenga la sensación de estar inmerso en el interior de estos mundos, siendo él mismo un elemento más en la experiencia.

Si bien se busca engañar al cerebro del usuario en su totalidad para hacerle creer que realmente se encuentra en una realidad generada virtualmente por una computadora, la cual tiene conceptos que serían imposibles de replicar en el mundo real, inicialmente solo con engañar al sentido de la vista puede causar que el usuario se sienta transportado a un nuevo mundo.

La distancia que hay entre los ojos humanos causa que veamos un objeto desde dos ángulos distintos al mismo tiempo, siendo procesadas posteriormente en nuestro cerebro como una sola imagen tridimensional, concepto llamado “percepción de profundidad”.¹⁶

Haciendo uso de dos imágenes que enfocan un mismo objeto desde dos ángulos distintos, técnica llamada “estereoscopía”, desde 1840¹⁷ se logra emular nuestra visión con la ayuda de lentes especiales o métodos de visualización, siendo únicamente posible la percepción tridimensional de escenas estáticas.

Gracias al avance del desarrollo de los gráficos generados por computadora en la década de los 60, no tomó mucho tiempo para unir este concepto con las técnicas de estereoscopía ya existentes para lograr lo que ahora conocemos como “realidad virtual”. Para esta visualización moderna es necesario utilizar de periféricos externos como lo son gafas o cascos con lentes y pantallas, especializados en mostrarle imágenes distintas a cada uno de los ojos del usuario.

¹⁶ (Certimedica, s.f.)

¹⁷ (Brewster, 1856)

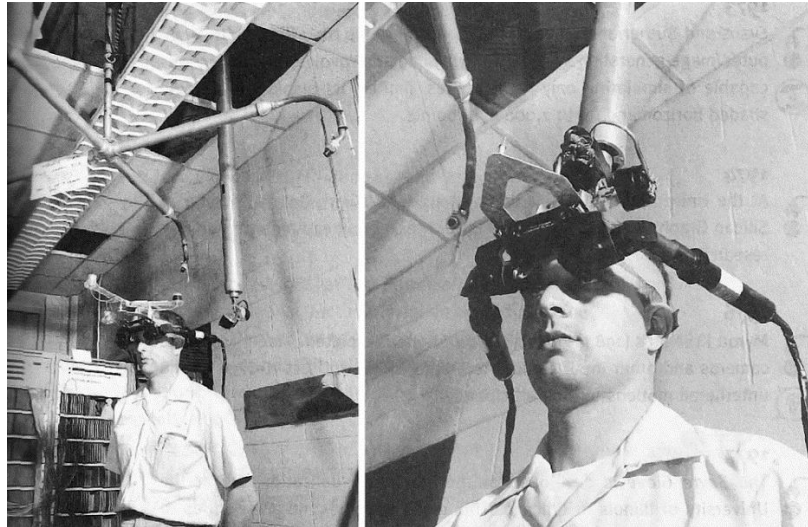


Figura 2. Espada de Damocles. Primer dispositivo de VR.
Fuente: A head-mounted three dimensional display, Ivan E. Sutherland, 1968

A pesar de una extensa campaña publicitaria sugiriendo usos visionarios de esta nueva tecnología, no llamó la atención de la industria durante las primeras décadas de su existencia, a causa de motivos que alejaron al consumidor como lo son la baja calidad de los elementos tridimensionales que eran capaces de generar los ordenadores de la época, o la incomodidad y náuseas que causaba utilizar los gigantescos y pesados dispositivos sobre la cabeza. Por lo que finalmente el desarrollo de productos de realidad virtual sucumbió a mediados de la década de los 90s, inclusive con exponentes de renombre compitiendo como Sega o Nintendo con sus dispositivos Sega VR y Virtual Boy, respectivamente.¹⁸

No es hasta el año 2011, en que el interés del público y la industria de los videojuegos vuelven a considerar a la realidad virtual como un potencial nicho de mercado que explotar. El coleccionista de dispositivos VR, Palmer Luckey, inicia una campaña en línea en busca de fondos para producir en masa el casco Oculus Rift, el cual aprovecha de los avanzados gráficos tridimensionales que son capaces de generar las computadoras de la época moderna, junto a un diseño cómodo, liviano y de fácil acceso para el consumidor. La campaña y una primera versión de prueba generaron tanta atención en la industria tecnológica que en marzo del 2014 se reportó la compra de Oculus VR, LLC por parte de la compañía Facebook con un elevado precio de \$2 mil millones.¹⁹

¹⁸ (KillScreen, 2015)

¹⁹ (Gamerant, 2014)

Desde inicios del siguiente año e inspirados en esta inusual inversión, se pronunciaron dentro de la apuesta a la realidad virtual varios gigantes tecnológicos, entre ellos Google con el dispositivo de construcción casera Google Cardboard compatible con casi cualquier smartphone, LG y Samsung con los productos LG VR y Gear VR respectivamente, que son variaciones de lo propuesto por Google pero únicamente compatibles con sus celulares de gama alta, Razer con la plataforma basada en desarrollo libre OSVR, Sony con su producto PSVR exclusivo para el uso junto a su consola Playstation 4, SulonQ un innovador producto sin necesidad de cables gracias a AMD, StarVR con lentes mucho más grandes buscando proyectar mayor contenidos a los ojos del usuario por parte de Acer, y por último y como la principal competencia contra Facebook y su Oculus Rift, la empresa de telefonía móvil china HTC en conjunto con una de las compañías más reconocidas en el mundo de los videojuegos, Valve Corporation, lanzan el HTC Vive.



*Figura 3. HTC Vive, Oculus Rift, PlayStation VR
Fuente: techinsider.io*

Esta afluencia de productos nos permite separar a los productos en dos categorías muy generales:

a. Dispositivos de visualización que utilizan el procesador del computador.

Estos dispositivos cuentan con el poder superior que ofrecen los procesadores y tarjetas gráficas para generar ambientes tridimensionales a varios cuadros por segundo creando fluidez y realismo en lo que se observa. Inclusive, se implementó una tecnología innovadora que detecta la posición del usuario en un espacio tridimensional no solo permitiéndole caminar en su propio espacio, sino también reconocer la posición de controles en las manos del jugador para interactuar con su ambiente de formas mucho más intuitivas. Sus desventajas están en la falta de movilidad al utilizarlos por los cables que deben estar conectados a la computadora y sus precios muy elevados.

b. Dispositivos de magnificación de las pantallas de teléfonos inteligentes.

Colocamos un smartphone dentro de estos visores y un par de lentes especiales engañan a nuestros ojos haciéndonos pensar que el contenido mostrado por la pantalla celular es nuestra realidad. Estos dispositivos casi no contienen componentes eléctricos para calcular los movimientos del usuario, más bien utilizan la potencia de algunos sensores dentro de los smartphones modernos, como lo son el giroscopio, magnetómetro y acelerómetros. Con ellos, el celular detecta el ángulo en el cual el usuario gira su cabeza, para así poder girar el ángulo de visión del contenido tridimensional. Sus desventajas son el débil procesamiento de elementos tridimensionales y la falta de métodos de interacción con el ambiente a nuestro alrededor.

1.3 Principales videojuegos móviles de realidad virtual

Lastimosamente, el desarrollo de videojuegos de realidad virtual para plataformas móviles se ha visto opacado por la creación de contenidos interactivos como lo son videos tridimensionales de 360 grados de libertad, aplicaciones educativas, de turismo, arquitectura o inclusive experiencias de terror, siendo varias de ellas aplicaciones por defecto sugeridas por Google, como lo es la aplicación gratuita “Cardboard” cuyo número de descargas se encuentra en un rango entre 5 y 10 millones.²⁰

Anteriormente se habló de que las principales desventajas son la debilidad de los smartphones para procesar elementos tridimensionales y la falta de métodos de interacción con el ambiente a nuestro alrededor. Para solucionar estos problemas el líder en este ámbito Google, con su sistema operativo móvil Android, propuso mantener un estándar mínimo para la construcción de los visores, así como del poder de procesamiento gráfico que ofrecen los celulares, junto a un control que reconoce los movimientos de nuestras manos permitiéndonos apuntar y seleccionar contenidos. Esta estandarización, de nombre Google Daydream, será lanzada oficialmente desde agosto del 2016.²¹

²⁰ (Google, 2016)

²¹ (VentureBeat, 2016)

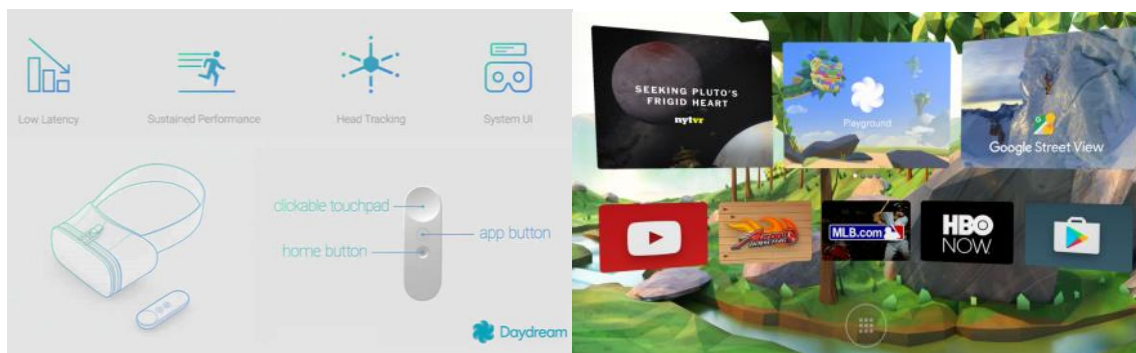


Figura 4. Google Daydream. Dispositivo y Aplicaciones.
Fuente: Google VR Daydream

Aún con estas limitaciones, no se ha impedido el crecimiento en cuanto al desarrollo de videojuegos móviles de realidad virtual un poco más cortos, pero con tareas específicas y rápidas a diferencia de las presentadas en el ambiente de dispositivos ayudados por el procesamiento de computadoras que requieren un mayor involucramiento del jugador. Estas diferencias se pueden comparar a las ya existentes entre el desarrollo de videojuegos tradicional para computadoras o consolas y el desarrollo para móviles, el cual es un mercado que en el 2016 por primera vez superará en ingresos a sus contrapartes.²²

En Google Play, la tienda virtual de aplicaciones para smartphones usuarios del sistema operativo móvil Android, proliferan los videojuegos de realidad virtual gratuitos, en los cuales se observa una diferenciación importante con los juegos tradicionales gratuitos: no se ha esparcido el uso de publicidad dentro del juego. Hasta el momento, ninguna metodología para mostrar publicidad dentro de las experiencias en videojuegos ha tomado importancia, en gran parte debido al cambio completo entre la costumbre de mostrar anuncios en posiciones de la pantalla que no estorben la experiencia y jugabilidad, y el modelo de realidad virtual, en el cual cualquier elemento externo o demasiado llamativo puede romper la inmersión del usuario, incomodándolo mucho más que la atracción que pueda crear al producto publicitado. Los desarrollos de VR han preferido tomar el rumbo de la monetización de sus juegos con una versión demo y su versión final completa pagada, o inclusive incluyendo compras dentro de la aplicación, siendo una solución mucho menos popular.

Existe una lista de títulos sugeridos personalmente por el equipo de Google, los cuales coinciden entre los resultados principales que aparecen en la tienda cuando se

²² (NewZoo, 2016)

hace una búsqueda de título de género “VR”, en la cual podemos ver una mezcla de contenidos pagados, gratuitos y con compras internas.²³

A continuación, un extracto de algunos de los juegos más comentados de dicha lista.

a. Juegos VR Gratuitos

- **Lamper VR: First Flight²⁴**

Desarrollador: Archiact Interactive Ltd.

Número de Descargas: 100.000 – 500.000

Género: Corredor infinito.

Método de Interacción: Movimiento guiado con la dirección de la vista.

Descripción: El juego trata de una abeja que debe esquivar obstáculos a la vez que vuela por ambientes naturales.

Nota:

- **Vanguard V Google Cardboard**

Desarrollador: ZeroTransform LLC

Número de Descargas: 100.000 – 500.000

Género: Disparos sobre carril.

Método de Interacción: Movimiento guiado con la dirección de la vista.

Descripción: En este juego destruiremos asteroides y enemigos a la vista, mientras nos acercamos a la órbita de un planeta en peligro de ser destruido si no lo salva el héroe de la historia.

- **Hardcode (VR Game)**

Desarrollador: Farlenkov

Número de Descargas: 100.000 – 500.000

Género: Disparos en tercera persona.

Método de Interacción: Control bluetooth externo.

Descripción: El juego presenta varios niveles y armas para interactuar.

²³ (Google, 2016)

²⁴ Nota: El desarrollo del prototipo de este trabajo de disertación se inspiró en este videojuego.

b. Juegos VR Pagados

El precio promedio de estos artículos es de \$2.

- **Proton Pulse Google Cardboard**

Desarrollador: ZeroTransform LLC

Número de Descargas: 10.000 – 50.000

Género: Acción rompe-ladrillos.

Método de Interacción: Movimiento guiado con la dirección de la vista.

Descripción: Controlamos una pared invisible con la cual lanzamos un balón direccionado a romper la mayor cantidad de ladrillos posibles y calcular la dirección en la que rebotará.

- **Caaaaardboard!**

Desarrollador: Dejobaan Games, LLC

Número de Descargas: 10.000 – 50.000

Género: Corredor infinito.

Método de Interacción: Movimiento guiado con la dirección de la vista.

Descripción: El juego nos muestra un personaje que está cayendo permanentemente, necesitando esquivar paredes y logrando trucos aéreos en el trayecto.

- **End Space VR for Cardboard**

Desarrollador: Orange Bridge Studios Inc.

Número de Descargas: 10.000 – 50.000

Género: Disparador en primera persona.

Método de Interacción: Movimiento guiado con la dirección de la vista y disparos causados por el ingreso de un botón.

Descripción: Nos encontramos en la cabina de una nave espacial, desde la cual intentamos sobrevivir a una batalla galáctica a nuestro alrededor.

Podemos observar que el número de descargas de los juegos catalogados como gratuitos no disminuyeron de 100.000, pero por otro lado el ingreso de los títulos pagados fue diez veces menos.

Como temática preferida, podemos observar una inclinación por los títulos futurísticos localizados en el espacio exterior.

La última observación interesante que se puede realizar en este extracto de los principales videojuegos móviles de realidad, es que en su gran mayoría se controlan con el limitado uso de la dirección de la vista para interactuar con los elementos alrededor, el cual era un método de interacción considerado entre las principales desventajas de la realidad virtual dedicada a móvil.

Capítulo 2

Documento de diseño del juego

El documento de diseño del juego actúa como nexo para conectar y enlistar todos los aspectos del videojuego. Contiene descripciones, imágenes, gráficos y tablas de la información de cada segmento del videojuego organizando las funcionalidades que se presentarán en el juego y como estas se relacionarán con el resto de elementos presentes. Es una ayuda para mantener a todo el equipo de desarrollo enterado de cuál es la visión y esencia que se busca lograr con la creación de dicho juego.

Para desarrollo más grandes, es importante porque ofrece herramientas y recursos que pueden ser utilizados por el departamento de marketing, con ideas conceptuales que pueden mostrar a inversores o un público expectante o inclusive a recursos humanos pues detalla los posibles puestos vacantes y cualificaciones necesarias para continuar con el desarrollo.

2.1. Descripción general

Lava Surf es un videojuego del género “corredor infinito”, del “endless runner”,²⁵ donde surfearás en el flujo de lava expulsada por un volcán activo, esquivando rocas, saltando deslaves y coleccionando monedas.

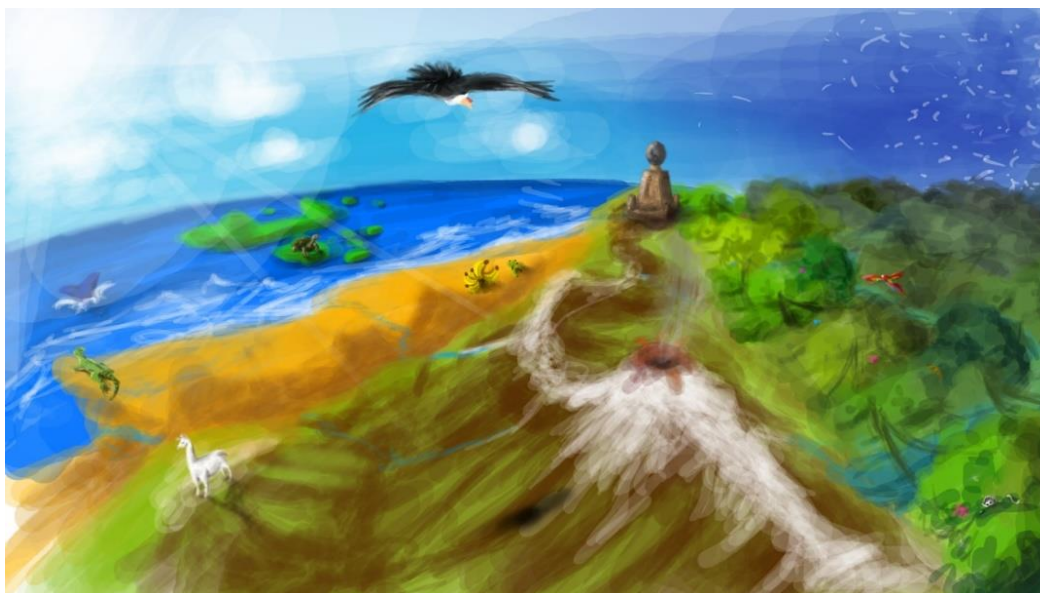
Siendo el personaje jugador un surfista profesional, inicialmente buscabas aventurarte en las famosas olas de las playas del exótico país Ecuador. Por un error de cálculo, terminas en la cima del volcán Cotopaxi, el cual inicia su proceso eruptivo obligándote a escapar de su destrucción haciendo lo que mejor sabes hacer: surfear.



*Figura 5. Bocetos De creación de Personaje
Elaborados en: Adobe Illustrator. Autor: Andrés Bastidas F.*

²⁵ Videojuego “endless runner”: Basado en correr infinitamente a través de un camino esquivando obstáculos.

El recorrido será bastante llamativo, pues mientras te encargas de hacer pequeños movimientos hacia los lados evitando golpearte con rocas o saltando encima de deslaves que te harían caer, el paisaje que te rodeará será de las cuatro regiones naturales ecuatorianas con diversos animales y animaciones individuales.



*Figura 6. Boceto Conceptual. Ecuador.
Elaborado en: Adobe Illustrator. Autor: Andrés Bastidas F.*

2.1.1. Plataforma / Especificaciones recomendadas

Se recomienda utilizar el motor de videojuegos Unity 3D, debido a la cantidad de documentación provista por los desarrolladores demostrando las posibilidades de la herramienta y a la simplificación de su interfaz gráfica de usuario que evita codificación para la manipulación de objetos en la misma. La herramienta también ofrece la posibilidad de publicar un mismo juego en varias plataformas, consolas y sistemas operativos sin la necesidad de crear proyectos externos para cada una de ellas. El presente prototipo será desarrollado para el sistema operativo móvil Android de Google pues su desarrollo y posteriores pruebas en dispositivos físicos pueden ser realizados de forma gratuita, mientras que en el caso del sistema operativo iOS de Apple, es necesario pagar una suscripción anual. Para ser posible la utilización de la tecnología de Realidad Virtual en conjunto con un dispositivo de tipo Cardboard, es indispensable poseer un giroscopio en el teléfono inteligente, y se recomienda una alta cantidad de densidad de píxeles.

2.1.2. Estilo visual

El estilo visual será “Low Poly” minimalista.²⁶ La paleta de color para los objetos estará basada en una selección presente en las guías y buenas prácticas de diseño de interfaces del concepto “Material Design” por Google.²⁷ La iluminación será de ambiente, con una tonalidad de temperatura púrpura para dar la apariencia de que los objetos son caricaturescos o hasta infantiles. Los assets tridimensionales utilizados son modificaciones de elementos anteriormente realizados para el programa de modelación tridimensional Blender, sus licencias de copyright serán libres o de reconocimiento de autoría. La preferencia del uso de Blender se debe a la transparencia que ofrecen los desarrolladores para el uso del programa al ser un proyecto de software libre así como la cantidad de contenido específico para Blender en librerías virtuales generado por usuarios.



Figura 7. Diseños Conceptuales de Lava Surf, Cordillera del Ecuador y Monumento Mitad del Mundo. Elaborado en: Blender. Autor: Andrés Bastidas F.

2.1.3. Estilo de audio

La presencia musical en el juego constará de una canción en repetición infinita. El género musical presentado será de fusión urbana, con una mezcla de sonidos tropicales y folclóricos en conjunto a ritmos y características más populares. El sonido de las interfaces, selección de menús y de recolección de objetos colectables será simple y caricaturesco, denotando con facilidad la selección de opciones por parte del usuario. Se utilizará música y sonidos de dominio público.

²⁶ (Thompson, 2007)

²⁷ (Google, 2016)

2.2 Interfaces de Usuario

2.2.1. Menú principal – selección

- Tras la pantalla de bienvenida de Unity 3D, se observará una escena de inicio de fondo espacial, y en frente el logo del videojuego.
- También frente al jugador, pero debajo del logo, se encuentra el botón de Inicio.
- Hacia la derecha, el jugador podrá observar la tabla de puntajes previos más altos.
- Hacia la izquierda, el jugador podrá elegir si utilizar el personaje surfista masculino o el femenino.
- Hacia atrás, si el jugador da una vuelta de 180 grados podrá observar los créditos del juego.



*Figura 8. Diseño Conceptual de Menú Principal de Lava Surf.
Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.*

2.2.2. Inicio del juego y pantalla de introducción

- La activación del botón de Inicio del menú principal cuya es causada en el caso de que el jugador apunte hacia este durante un lapso aproximado de 5 segundos, lo cual se indicará con una animación de progreso. En caso de dejar de ver hacia el botón, este contador para su activación se reinicia.
- Se utilizará el mismo método para la selección del personaje.
- Se inicia el juego con el personaje visible desde una cámara en tercera persona, mientras esté está surfeando sobre un flujo de lava líquida.
- A su alrededor se encuentre una escena inspirada en las regiones naturales de Ecuador.
- El juego entra en marcha siempre y cuando el jugador se encuentre observando al frente suyo.
- El momento en que regresa a ver hacia sus lados, atrás, arriba o abajo, el juego

se pausará automáticamente.



*Figura 9. Distribución Conceptual del Área del Juego
Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.*

2.2.3. HUD y menú dentro del juego

- El HUD, o Visualización Heads-Up por sus siglas en inglés, constará de un pequeño label de texto flotando de forma tridimensional, inicialmente, cerca de la espalda del jugador.
- Su función será para indicar el puntaje actual del jugador, cambiando a la palabra “Pausa” el momento que la visión del jugador salga del área de juego, así como mostrar si el jugador tiene coleccionado o no un corazón.
- Siempre será visible desde cualquier ángulo, es decir que, si el jugador fija su vista en cualquier parte de la zona del área de pausa, seguirá legible.



*Figura 10. Diseño Conceptual de Interfaz de Usuario.
Elaborado en: Blender. Autor: Andrés Bastidas F.*

2.2.4. Pantalla de pérdida

- El momento en que el jugador pierde, el juego retorna al menú principal.
- La diferencia con el menú del inicio del juego es que ahora se muestra información sobre el puntaje recibido en esa partida.
- Existirá un aviso en caso de haber logrado un nuevo record de puntaje.



*Figura 11. Pantalla de Pérdida y Menú Exterior Principal
Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.*

2.2.5. Nivel

Al ser un “endless runner” el juego se mantiene en un mismo nivel mientras que los obstáculos son los que varían según tamaño y dificultad de forma infinita hasta que el jugador pierda. Aun así, el nivel es interactivo según la dirección de la vista del jugador. Aparte del área visual del juego, tenemos el área de pausa, la cual a su vez se divide en otras áreas.



*Figura 12. Distribución Conceptual de la Visión del Jugador.
Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.*

El momento en que el jugador ingresa al área de pausa y redirige su mirada hacia una de las regiones, los elementos en esta se agrandarán y su animación propia iniciará en repetición infinita hasta que el jugador cambie la dirección de su mirada.

Las 5 zonas interactivas, en orden contrario a las agujas del reloj, son estas:

- Galápagos

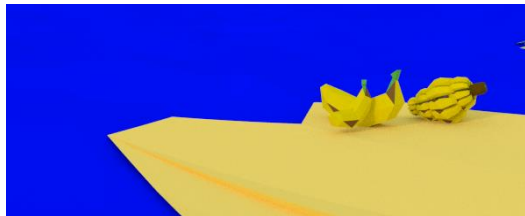
Brisa y aventura. Sus elementos incluyen una tortuga y palmeras. Su animación consta en la tortuga escondiéndose y revelándose en su caparazón y desde él, así como un movimiento de las palmeras causado por viento.



*Figura 13. Diseño Conceptual Islas Galápagos
Elaborado en: Blender. Autor: Andrés Bastidas F.*

- Costa

Arena y relajación. Sus elementos incluyen una mata de bananas, una banana suelta y una ballena. Su animación consta en las bananas rodando mientras que la ballena salta e ingresa nuevamente al agua.



*Figura 14. Diseño Conceptual Región Costa
Elaborado en: Blender. Autor: Andrés Bastidas F.*

- Volcán

Lava y acción. Sus elementos incluyen el volcán, lava y nubes de cenizas. La animación será un estallido de lava causando el flujo de lava por sus laderas y nubes de cenizas subiendo al cielo.



*Figura 15. Diseño Conceptual Volcán en Región Sierra.
Elaborado en: Blender. Autor: Andrés Bastidas F.*

- Amazonía

Humedad y misterio. La zona con mayor ángulo de visión. Sus elementos incluyen el río Napo, orquídeas, árboles, y un mono. Su animación es el flujo del río, la humedad rasante y el mono apareciendo desde su árbol.



Figura 16. Diseño Conceptual Región Amazonía
Elaborado en: Blender. Autor: Andrés Bastidas F.

- Sierra

Frío y paz. Debido al escenario de juego, esta zona sirve como fondo del juego y su ángulo de visión es pequeño. Su animación se activa sólo observando hacia la llama. Sus elementos incluyen la cordillera de los Andes, el monumento de la Mitad del Mundo y una llama. Su animación trata de la llama comiendo bajo sí misma y el globo terráqueo del monumento girando en su propio eje.



Figura 17. Diseño Conceptual Región Sierra
Elaborado en: Blender. Autor: Andrés Bastidas F.

Por último, hay una zona interactiva más a la cual el jugador puede regresar a ver activando la animación de esta. Ubicada en casi toda la zona superior del jugador, el cielo cambiará la hora del día y el origen de la iluminación.



Figura 18. Diferenciación entre Área de Juego o Área de Cielo
Elaborado en: Blender. Autor: Andrés Bastidas F.

- Cielo
Serenidad. Sus elementos incluyen un ave cóndor en vuelo y la “chakana” o cruz andina, representante del sol en varias culturas pre-incaicas.



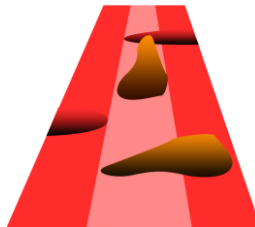
*Figura 19. Diseño Conceptual de Área del Cielo.
Elaborado en: Blender. Autor: Andrés Bastidas F.*

2.3. Jugabilidad

2.3.1. Mecánicas

El juego es modalidad “endless runner” por lo que el objetivo será esquivar obstáculos que se aproximan hacia el jugador mediante movimientos horizontales.

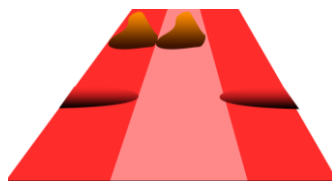
El jugador irá surfeando a través un flujo de lava, el cual constará de 3 carriles en los cuales podrán aparecer obstáculos desde la lejanía y acercándose cada vez más, siendo imposible que un obstáculo “salte” entre carriles.



*Figura 20. Distribución Conceptual de Carriles y Obstáculos
Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.*

Es posible que hasta dos obstáculos aparezcan en el horizonte al mismo tiempo en una “línea”, considerando que la anterior posición “libre” por donde el jugador puede esquivar el nuevo obstáculo, no haya estado en el extremo opuesto, lo cual requeriría que el jugador tenga que movilizarse entre dos carriles en un lapso de tiempo muy corto.

Los obstáculos son de dos clases, rocas y agujeros, cuya única diferencia es estética. A pesar de eso, no se mezclarán las distintas clases de obstáculos en una sola línea. Es decir que puede haber dos rocas o dos agujeros en una misma línea, pero nunca una roca y un agujero.



*Figura 21. Distribución Conceptual de Carriles y Obstáculos Alternativos
Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.*

2.3.2. Controles

El control del jugador estará dictado por la dirección de la vista del jugador. Haciendo uso de 3 objetos invisibles, se determinará si la retícula está apuntando a alguno de ellos, dirigiendo al jugador hacia el respectivo carril al que se observa.



*Figura 22. Distribución Conceptual Objetos Invisibles con Eventos de Aproximación
Elaborado en: Adobe Photoshop. Autor: Andrés Bastidas F.*

2.3.3. Modos

Se presentará la posibilidad de visualizar y jugar sin necesidad de un dispositivo Cardboard, determinando toda la pantalla del celular al juego en modo portarretrato. De igual forma, se puede presentar el modo de visualización pura sin necesidad de jugar, entrando en un estado constante de “pausa”.

2.3.4. Al ganar

Al ser un videojuego de género endless runner, el jugador no consigue un concepto de finalización del juego, sino que más bien será recompensando con un sistema de puntaje que se multiplicará según su tiempo de permanencia y dificultad de la partida.

2.3.5. Puntaje

El puntaje de cada partida será calculado según la cantidad de metros avanzados

por el jugador

Capítulo 3

Desarrollo del prototipo

Si bien la metodología de desarrollo ágil Scrum está enfocada en el desarrollo de productos por parte de un equipo de trabajo, existe una variación nombrada como “Personal Scrum” la cual toma algunos de los aspectos que ayudan a la organización de tiempos del desarrollador y prioridad y evolución de las tareas. Para el desarrollo del presente trabajo se buscó seguir la mayor cantidad de principios y procesos de metodologías ágiles aplicadas al desarrollo de una sola persona, en especial cuando la creación de un videojuego requiere de roles multidisciplinarios que deben llenados en tiempo relativamente simultáneo para lograr que el desarrollo del producto sea fluido en todos los departamentos.²⁸

Por ejemplo, el desarrollo de un videojuego de realidad virtual puede ser seccionado en tres partes de igual de importancia, con las siguientes descripciones generales: La modelación visual de los elementos, la programación y el método para unir ambos.

3.1. Modelación tridimensional de assets

La modelación visual de los elementos a utilizarse, o en inglés denominados como assets, fue el primer paso y prioridad del desarrollo, pues sobre estos se agregaría las animaciones y programación correspondiente.

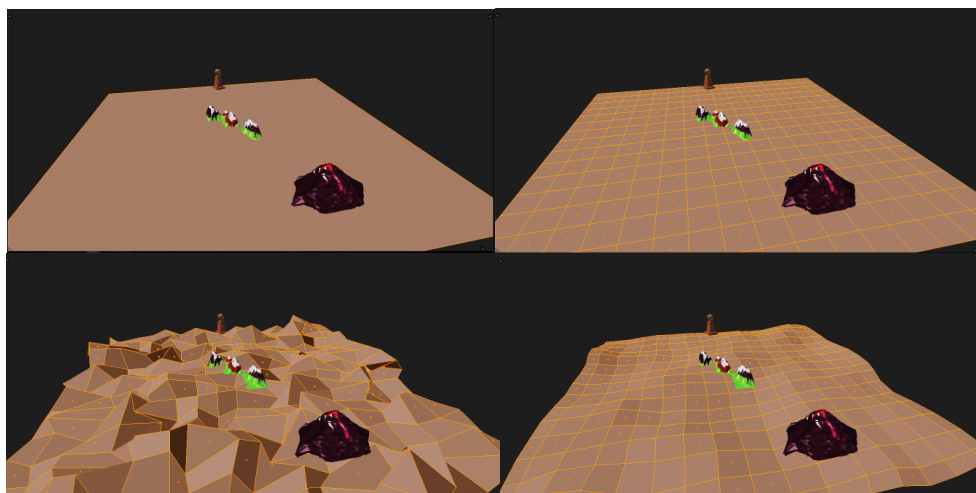
3.1.1. Modelación tridimensional desde cero

Gracias a las herramientas brindadas por el programa de modelación tridimensional gratuito Blender, se logró recrear los elementos necesarios creándolos desde 0 o en varios casos siendo ediciones a assets ya existentes en bibliotecas virtuales que contienen contenidos generados por usuarios, cuyos derechos de uso tienen que ser completamente libres.

²⁸ (Mountain Goat Software, 2015)

El proceso de modelado 3D puede ser descrito en términos simples como una forma de escultura digital donde existen vértices, vectores y planos que pueden ser editados en un ambiente tridimensional según cualquiera de los ejes X, Y o Z. Avanzadas herramientas de automatización incorporadas en los softwares modeladores nos evitan el trabajo de editar cada polígono en un objeto, para poder agregar efectos y distorsiones generales según se desee.

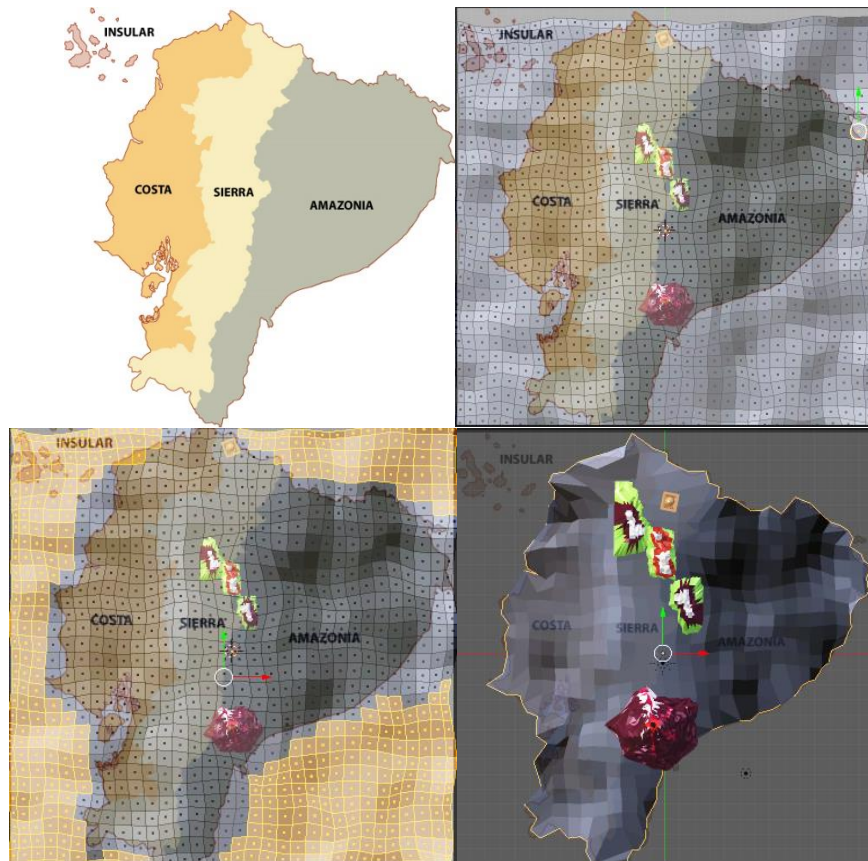
A pesar de esta facilidad, es importante tener en cuenta que la cantidad de polígonos mostrados simultáneamente en cada escena será el factor primordial para determinar el rendimiento del procesador del dispositivo móvil a mostrarse, siendo importante mantener el número más bajo posible ya sea utilizando técnicas de reducción de polígonos tras haber acabado el modelamiento, como por ejemplo el modificador “Decimate” en Blender,²⁹ o también implementando un método para esconder los objetos de forma dinámica si es que la vista del usuario no está en esa dirección.



*Figura 23. Modelado de Plano, Subdivisión, Randomización y Suavizado
Elaborado en: Blender. Autor: Andrés Bastidas F.*

Iniciando por el entorno, fondos y paisajes a ser visualizados en el ambiente, se utilizó como inspiración la silueta del mapa político del país Ecuador. Con elementos importantes para la jugabilidad previamente modelados, como lo son la línea de visión directa del jugador con montañas de la cordillera de los Andes o el emblemático monumento a la Mitad del Mundo.

²⁹ (Blender, s.f.)



*Figura 24. Modelado basado en la silueta del mapa político del Ecuador
 Fuente: presidencia.gob.ec
 Elaborado en: Blender Autor: Andrés Bastidas F.*

3.1.2. Edición de assets existentes

A continuación, continuamos con la modelación del resto de elementos pertenecientes a la geografía y fondo, como lo son las islas Galápagos, la zona Costa y la Amazonía junto a su río.

Agregamos los elementos visuales de fondo que estarán animados y darán vida a cada una de las regiones cuando el jugador regrese su vista hacia ellas. Para este paso, la modelación desde 0 fue insuficiente, y se prefirió utilizar assets ya existentes con modificaciones propias que coincidan con lo propuesto en el documento de diseño del juego, incluyendo la reducción de polígonos.

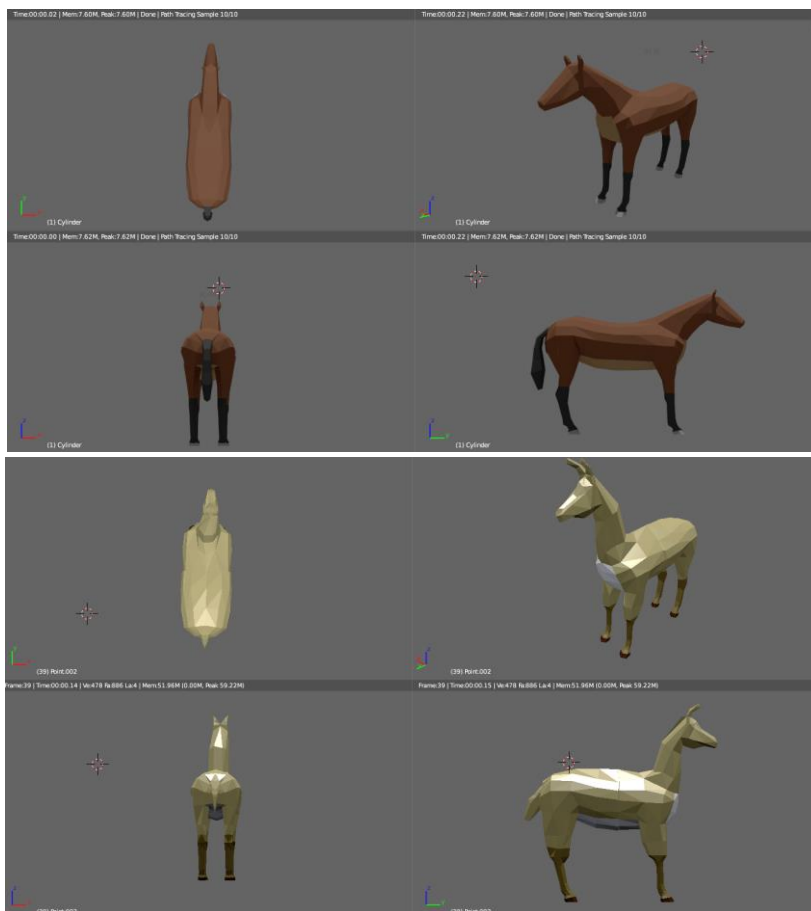


Figura 25. Comparación entre Asset descargado de biblioteca de recursos con forma de Caballo y edición propia con nuevas formas y colores para asemejarse a una Llama.
Elaborado en: Blender. Autor: Andrés Bastidas F.

Tabla 1

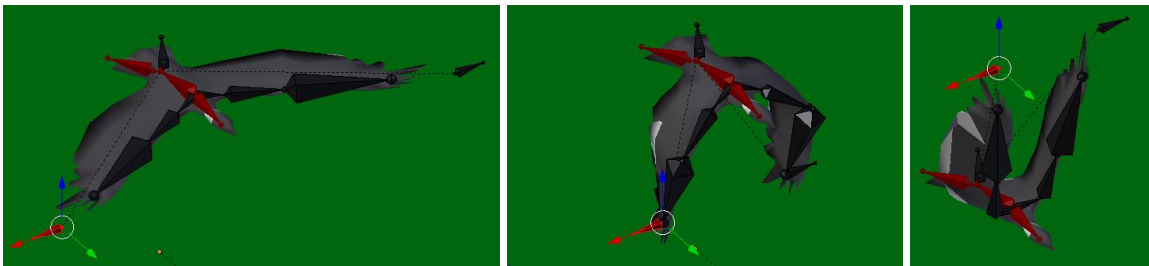
Lista de Assets utilizados en el desarrollo del videojuego. **Fuente:** [www.Blendswap.com](http://www.blendswap.com)

Nombre	Autor	Licencia de Copyright	URL
Caballo	MarkLaBarr	CC Zero	http://www.blendswap.com/blends/view/74846
Cuervo	Nodelete	CC Zero	http://www.blendswap.com/blends/view/61014
Montaña	Druid CZ	CC-BY	http://www.blendswap.com/blends/view/75948
Volcán	PigArt	CC-BY	http://www.blendswap.com/blends/view/74028
Árbol	CactusArtz	CC-BY	http://www.blendswap.com/blends/view/76949
Mono	Josh nwnham	CC-BY	http://www.blendswap.com/blends/view/68638
Ballena	Gabriel	CC Zero	http://www.blendswap.com/blends/view/38458
Tortuga	SSimpossible	CC-BY	http://www.blendswap.com/blends/view/13830
Personaje	Kduribeiro	CC-BY	http://www.blendswap.com/blends/view/12199

3.1.3. Creación de esqueletos

Una vez que se finaliza el proceso de modelado de los objetos, se requiere diferenciar entre aquellos que quedarán como estáticos en la escena y aquellos que tendrán su propia animación, ya sea por motivos decorativos o de acciones a visualizarse en la jugabilidad.

Para lograr estas animaciones se juntan varios vértices del modelo a un subgrupo denominado como “hueso”, los cuales a su vez se encuentran colocados en el interior del modelo en la agrupación conocida como “esqueleto”. La herramienta modeladora nos permite crear poses predeterminadas para el modelo, así como efectos que los movimientos realizados en un lado del esqueleto sean reflejados simétricamente al lado opuesto del modelo, como en el caso de las alas del cóndor.



*Figura 26. Esqueleto de Cóndor. Animación de Vuelo.
Elaborado en: Blender. Autor: Andrés Bastidas F.*

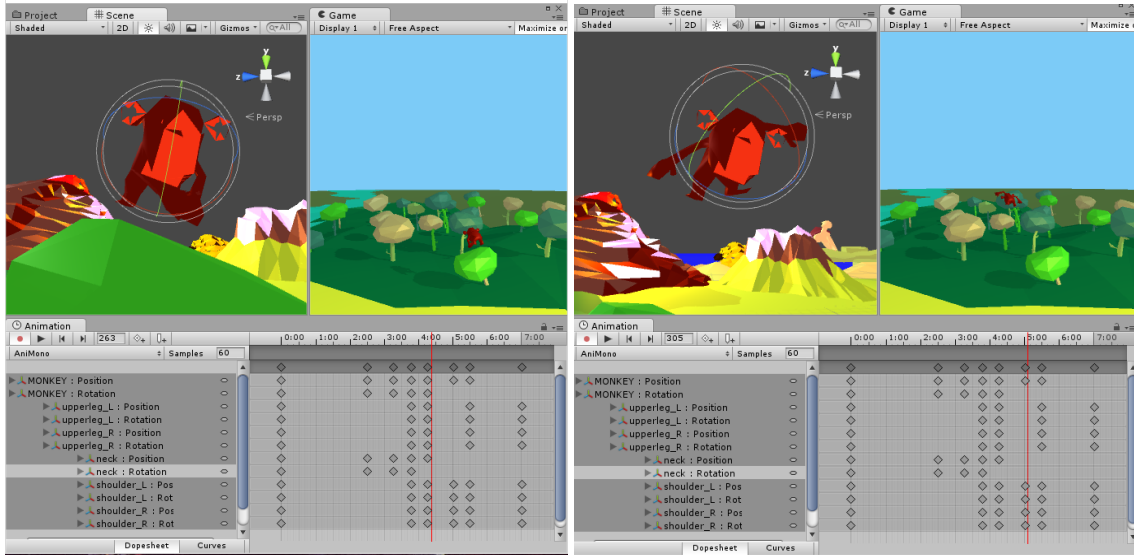
3.2. Entorno de desarrollo Unity3D

El entorno de desarrollo de Unity3D es una herramienta especializada en el desarrollo de videojuegos y ambientes, con la facilidad de poder publicar nuestro contenido en varias plataformas modernas, entre ellas la seleccionada para este proyecto: el sistema operativo móvil Android.

De igual forma, nos permite importar desde Blender todos los assets de forma nativa e individual y hacer cambios superficiales o agregar propiedades sin perder la habilidad de editar su modelo tridimensional posteriormente.

La animación en Unity3D es controlada a través de una línea de tiempo en la cual seleccionamos las posiciones, rotaciones y tamaños en los que queremos animar el esqueleto de un determinado modelo en distintos momentos, la traslación entre un punto

y otro se realiza automáticamente gracias al motor. Al finalizar la animación posponemos deliberadamente su finalización, para crear un bucle de forma que si se repite la animación no se vea repetitivo para el usuario.



*Figura 27. Comparación de Animación: Mono, Brazos Cerrados y Mono Saltando en el Aire, Brazos Abiertos
Elaborado en: Unity 3D. Autor: Andrés Bastidas F.*

Se puede crear varias animaciones para un solo modelo, y seleccionar cualquiera de ellas de forma dinámica según se requiera. Esto es útil en el caso de que el personaje jugador tenga varias acciones, siendo en el caso del videojuego Lava Surf con la diferencia entre la animación por defecto cuando el personaje surfea o cuando este pierde al impactar contra un obstáculo frente a él.

3.3 Codificación de funciones en los lenguajes C# y Javascript

Las funcionalidades principales de un videojuego “endless runner” son simples. Se necesita codificar un ambiente en el cual el personaje del juego aparente estar en constante movimiento hacia adelante mientras que obstáculos se aproximan hacia él desde el horizonte, en posiciones aleatorias, de forma que puedan ser esquivados una vez que estos llegan al personaje. No existe un estado final del juego, brindándole al jugador la posibilidad de jugar infinitamente mientras que no choque contra alguno de los obstáculos, y aumentando su puntaje indefinidamente según la cantidad de tiempo o “distancia” recorrida.

3.3.1. Generación de obstáculos aleatorios

Lava Surf, específicamente, presenta tres carriles por los cuales aparecen los obstáculos, o en los cuales se puede posicionar al personaje para esquivarlos.

```
6 public float createCadaSegundos = 5f;
7 public GameObject[] obs = new GameObject[3];
8 float[] positions = new float[3];
9 float pos = 0;
10
11 void Start() {
12     InvokeRepeating("CrearObstaculo", 0, createCadaSegundos);
13     positions[0] = 0.15f;
14     positions[1] = 1.6f;
15     positions[2] = -1.1f;
16 }
17
18 void CrearObstaculo() {
19     int rand = Random.Range(0, positions.Length);
20     while (pos == positions[rand]) {
21         pos = positions[Random.Range(0, positions.Length)];
22     }
23     pos = positions[rand];
24
25     Instantiate(    obs[Random.Range(0,obs.Length)],
26                   new Vector3(pos, 6.25f, -50f) ,
27                   Quaternion.identity
28                 );
29 }
30
```

Figura 28. Código para generación aleatoria de obstáculos.
Elaborado en: MonoDevelop. Autor: Andrés Bastidas F.

Como podemos observar en la **Figura 28**, al inicializarse el código, se llama a invocar la repetición de la función “CrearObstaculo()” en intervalos separados por 5 segundos. En la **línea 25** se elige aleatoriamente uno de los 3 obstáculos prefabricados, para colocarse en la posición igualmente aleatoria para el eje X según lo descrito en la **línea 26**.



Figura 29. Diferencia entre la posición y forma de los obstáculos generados aleatoriamente. Elaborado en: Unity3D. Autor: Andrés Bastidas F.

3.3.2. Aumento del puntaje

El ojo humano no es capaz de enfocar y ver correctamente objetos demasiado cercanos³⁰, por lo tanto, se cambia el uso habitual de una interfaz gráfica de usuario estática por una interfaz que nos muestre información relevante pero que se encuentre de igual forma en el ambiente tridimensional del juego.

En el juego desarrollado, la puntuación siempre está visible en forma de texto flotante detrás del personaje y representa la cantidad de metros en aumento que el personaje ha surfado sin chocar. El contador se inicializa al empezar el juego y llama a repetirse la función “metroPorSegundo()” encargada de aumentar el integral global “puntaje” cada medio segundo y cambiar el texto del campo flotante, siempre y cuando el juego no se encuentre en un estado de pausa o pérdida, lo cual se revisa gracias a los booleanos globales “pausado” y “perdido”.

```
IEnumerator metroPorSegundo(){
    while(!pausado || !perdido) {
        TextMesh textObject = GameObject.Find("TEXT").GetComponent<TextMesh>();

        puntaje++;
        textObject.text = puntaje + "m";
        yield return new WaitForSeconds(0.5f);
    }
}
```

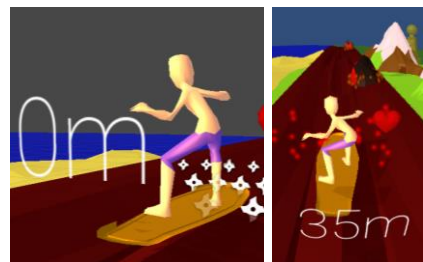


Figura 30. Código para aumento de metros recorridos y Visualización Tridimensional del texto. Elaborado en: Unity3D. Autor: Andrés Bastidas F.

3.3.3. Pérdida al impactar obstáculos

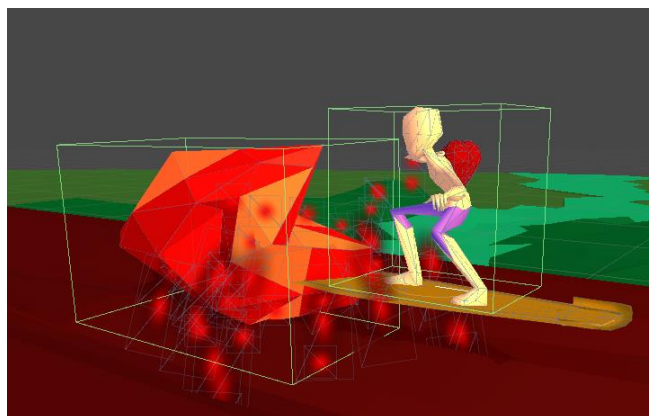
³⁰ (Unity3D, 2016)

El momento en que el personaje impacta contra uno de los obstáculos el juego termina y se inicia la pantalla de menú principal donde el jugador puede observar cual fue su puntuación, así como cuál ha sido la puntuación más alta hasta ahora y por último repetir el juego nuevamente.

Para calcular si un objeto ha impactado con otro es demasiado demandante para un dispositivo móvil si utilizamos los polígonos de los elementos para comparar si se encuentran intersecando en una misma posición. Para solucionar esta falta, se utilizan elementos adjuntos a su modelo llamados “Box Colliders” que toman una forma cúbica mucho más simple alrededor de los modelos tridimensionales. Una vez que entran en colisión entre un collider y otro, se activa el código que empezará por cambiar la animación del personaje, así como el texto a mostrar y después de una espera de pocos segundos, envía al jugador directamente al menú principal.

```
void OnCollisionEnter (Collision col) {
    if(col.gameObject.tag == "Obstaculo") {
        StartCoroutine( perdiste() );
    }
}
IEnumerator perdiste(){
    animPersonaje.runtimeAnimatorController = animacionPerdiste as RuntimeAnimatorController;
    TextMesh textObject = GameObject.Find("TEXT").GetComponent<TextMesh>();
    Globales.perdido = true;
    textObject.text = "Perdiste";
    yield return new WaitForSeconds(2f);
    Application.LoadLevel (Application.loadedLevel);
}
```

*Figura 31. Código para detección de colisiones y cambio de escena.
Elaborado en: MonoDevelop. Autor: Andrés Bastidas F.*



*Figura 32. Colliders visibles de obstáculo y personaje.
Elaborado en: Unity3D. Autor: Andrés Bastidas F.*

3.3.4. Kit de desarrollo de software “GoogleVR”

Una vez que se ha codificado la estructura, elementos e interfaces en el juego, es necesario codificar el método de interacción a través del cual el usuario podrá comunicarse con el juego y representar sus acciones.

En este paso el desarrollo de un videojuego de realidad virtual difiere del desarrollo de un juego tradicional, pues con la necesidad de utilizar un dispositivo externo con lentes especiales para aumentar lo mostrado en cada mitad de la pantalla del smartphone, se pierde el principal medio de interacción que posee un celular, su pantalla táctil. Como solución, y gracias a que la realidad virtual es una emulación de lo que nuestros ojos observan, se utiliza un método que es mucho más intuitivo para el humano, nuestra vista.

El kit de desarrollo de software, o SDK por sus iniciales en inglés “software development kit”, “Google VR” brindado de forma gratuita y listo para implementarse en cualquier proyecto de Unity3D nos da la facilidad de agregar código para visualizar dos ángulos cámaras en una sola pantalla. Las cámaras son casi idénticas solo siendo horizontalmente separadas por una pequeña distancia (emulando la separación entre los ojos humanos). Una vez instalado el proyecto en el dispositivo móvil, el código también se encarga de recibir los datos enviados por el giroscopio del celular, indicándole cuál es su rotación con respecto a la gravedad de la tierra, de forma que la dirección hacia la que apuntan las cámaras pueda ser controlado por la rotación de la cabeza del usuario.

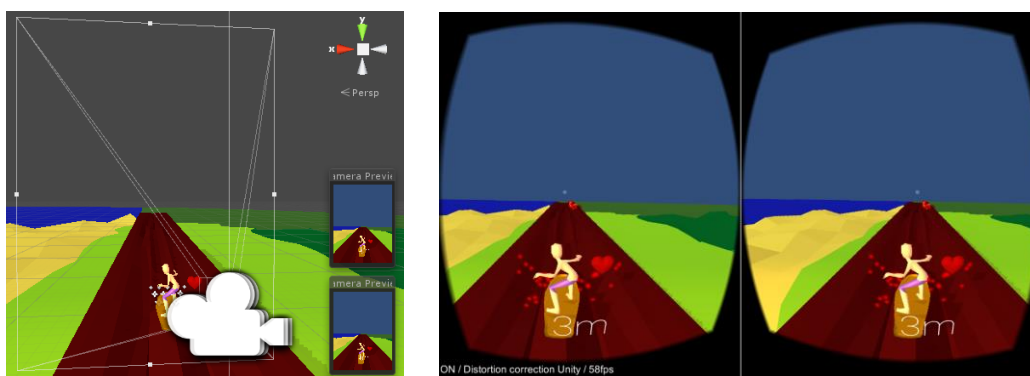


Figura 33. Par de Cámaras separadas horizontalmente y Visualización de ambas en una sola pantalla.
Elaborado en: Unity3D. Autor: Andrés Bastidas F.

La última funcionalidad que brinda el SDK de realidad virtual, nos permite conocer cuando la vista de un usuario está centrada en algún objeto, de forma que podemos codificar funciones en el caso de que el usuario esté observando o no a un objeto inclusive si este objeto es invisibilizado de la escena.

De esta forma se decidió crear un sistema de cubos para lograr dos funcionalidades principalmente: el movimiento del personaje, un método para pausar el juego y la eliminación e instanciación dinámica de los escenarios alrededor del personaje solamente cuando el usuario dirija su vista hacia ellos.

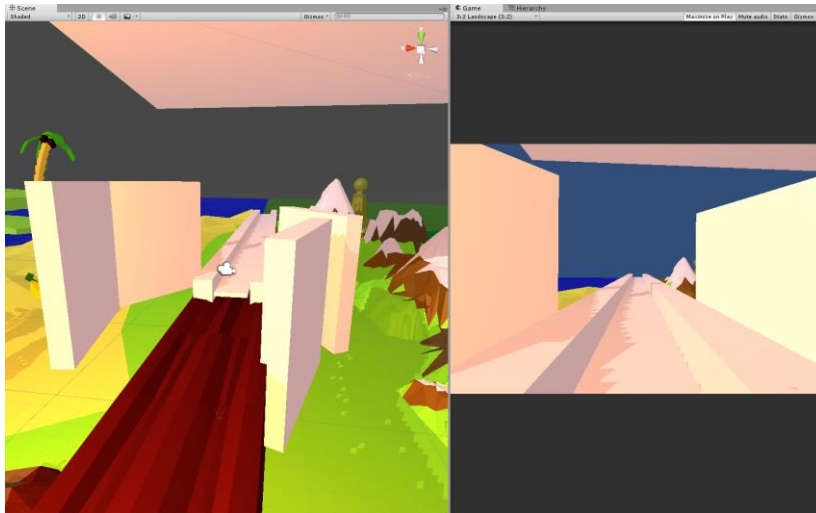


Figura 34. Objetos rodeando al personaje posteriormente invisibles. Disparan un código distinto si son observados. Elaborado en: Unity 3D . Autor: Andrés Bastidas F.

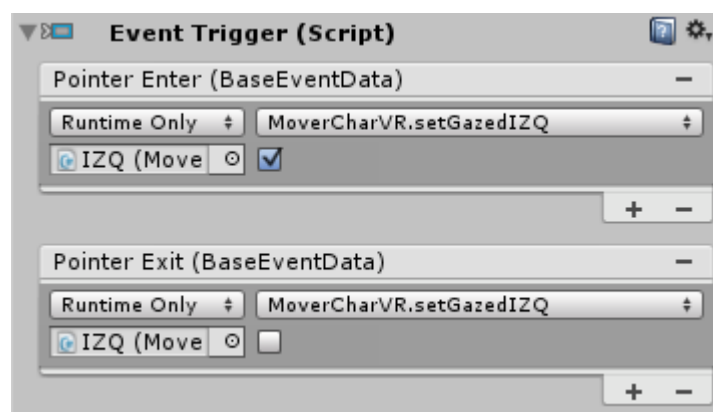


Figura 35. Funciones y parámetros, verdadero o falso, a dispararse cuando el apuntador ingresa o sale del objeto. Elaborado en: Unity3D. Autor: Andrés Bastidas F.

3.3.5. Movimiento del jugador

Para lograr que el jugador pueda esquivar los obstáculos debe poder colocarse en uno de los 3 carriles que se encuentren vacíos, esta dirección se logró utilizando 3 objetos invisibles que disparan un código que trasladando al personaje a distintas posiciones según la mirada del jugador.

```

76 public GameObject personaje;
77 float[] carriles = new float[3];
78
79 void Start() {
80     carriles [0] = 1.3f;
81     carriles [1] = 0;
82     carriles [2] = -1.3f;
83 }
84
85 public void setGazedIZQ(bool gazed){
86     Globales.verIZQ = gazed ? true : false;
87 }
88 public void setGazedCTR(bool gazed){
89     Globales.verCTR = gazed ? true : false;
90 }
91 public void setGazedDER(bool gazed){
92     Globales.verDER = gazed ? true : false;
93 }
94
95 void Update(){
96     int i = 1;
97
98     if (Globales.verIZQ) {
99         i = 0;
100     } else if (Globales.verCTR) {
101         i = 1;
102     } else if (Globales.verDER) {
103         i = 2;
104     }
105
106     personaje.transform.position =
107     Vector3.MoveTowards( personaje.transform.position, new Vector3(carriles[i], -0.04f, -8.84f), (Time.deltaTime*2) );
108 }

```

Figura 36. Código para movilización del personaje según la vista del usuario.
Elaborado en: MonoDevelop. Autor: Andrés Bastidas F.

Como podemos observar en la **Figura 36**, entre las líneas 85 y 93, tenemos tres funciones distintas que cada una convierte en verdadera o falsa distintas variables globales booleanas llamadas verIZQ, verCTR y verDER. Más adelante, entre las líneas 98 y 104, dentro de la función de actualización repetida automáticamente “Update” se asigna una posición que será la encargada de decidir la posición hacia la que el personaje deberá moverse en la **línea 107**.

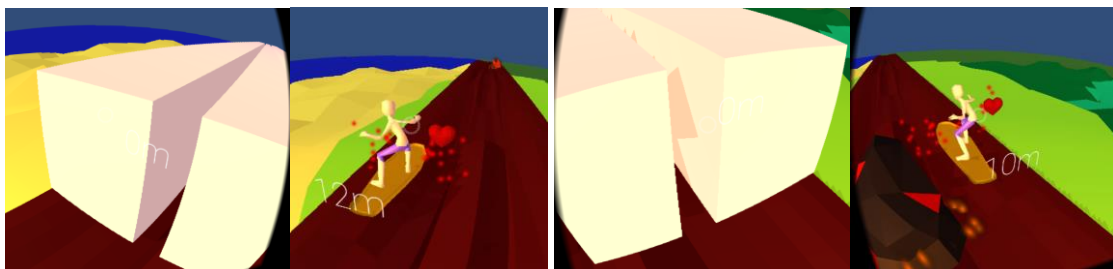


Figura 37. Objetos Visibles y su función de movilizar al personaje a distintas posiciones.
Elaborado en: Unity3D. Autor: Andrés Bastidas F.

3.3.6. FPS. Eliminación e instancias de escenarios no visibles.

Una vez finalizada la codificación del prototipo se decidió implementar una función más relacionada con el mantenimiento y desempeño de la aplicación para que el usuario tenga una experiencia óptima.

El ojo y cerebro humano utilizan varias técnicas para reconocer que un objeto se encuentra en movimiento y cuál es su velocidad. La animación puede ser emulada en ambientes artificiales si se muestra una rápida sucesión de imágenes en un espacio de un segundo, de forma que esta métrica se llama fotogramas por segundo, o también FPS por su abreviación del inglés de “frames per second”.

El ojo promedio puede ser engañado que se encuentra viendo un objeto animado si es que el video supera los 25 fotogramas por segundo, mientras que el cerebro promedio detecta que en realidad se tratan de varias imágenes sucesivas si este número es menor.³¹

El caso de la realidad virtual tiene un acercamiento totalmente distinto en cuanto a los FPS necesarios en comparación con la visualización tradicional de video digital. Esta diferencia se basa en que no solo estamos mostrándole a los ojos una animación en un dispositivo que se encuentra en un dispositivo estático frente a nosotros, sino que intentamos engañar al cerebro que la realidad a su alrededor es otra a la que ha estado acostumbrado toda su vida, la cual no tiene una limitación de cuadros por segundo.

Es por esto, que los FPS mínimos recomendados para un ambiente de realidad virtual está en un rango de 45 a 75 cuadros por segundos.³² El procesador gráfico de los dispositivos móviles debe encargarse de generar esta cantidad de imágenes en menos de un segundo para proyectarla al ojo humano sin que el usuario sienta disconformidad o, peor aún, náuseas.

La cantidad de objetos mostrados en una sola escena es inversamente proporcional a la capacidad del procesador para generar más FPS en un dispositivo, es por esto que se decidió implementar la funcionalidad de eliminar todos los objetos del paisaje externos a la necesidad y jugabilidad del juego y únicamente traerlos a escena cuando el jugador dirija su vista hacia donde estos se encontraban originalmente.

³¹ (British Broadcasting Corporation, 2008)

³² (MTech Games, 2015)

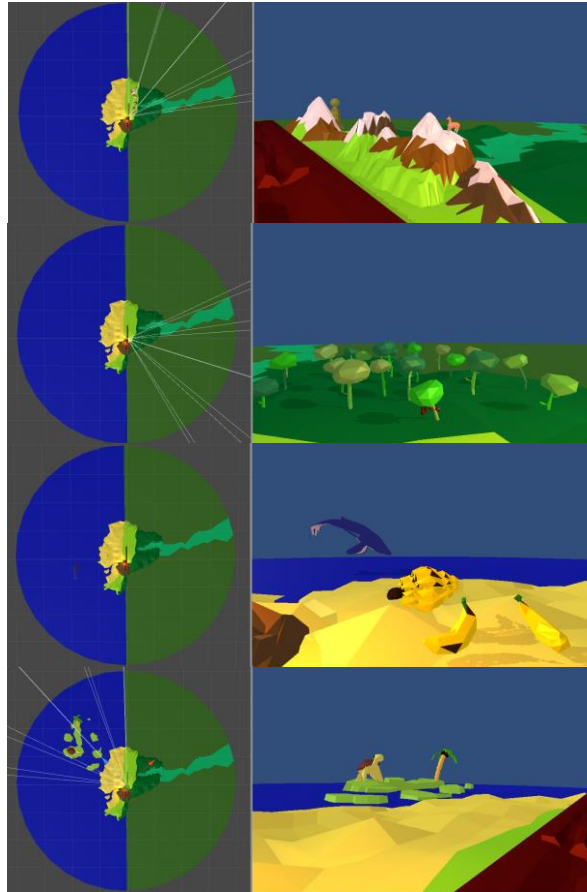


Figura 38. La dirección de la vista del usuario nos muestra una instancia de cada región, eliminándola posteriormente.
Elaborado en: Unity3D. Autor: Andrés Bastidas F.

El aumento en el desempeño del juego, es decir su cantidad de FPS, fue bastante considerable como se demuestra en el capítulo siguiente.

Capítulo 4

Pruebas del videojuego

Con el juego ya desarrollado, se procedió a documentar el proceso de pruebas por el que atravesó el videojuego para definir cuál sería la experiencia óptima para el usuario. La metodología utilizada fue de pruebas de caja negra, basadas en la comparación entre el desempeño de distintas versiones del videojuego en una combinación de dos celulares y dos dispositivos de realidad virtual tipo Cardboard.

Tabla 2
Dispositivos de realidad virtual tipo Cardboard

Versión	Diámetro de Lente	Distancia Focal	Material
1.0	25mm	45mm	Cartón Corrugado
2.0	45mm	50mm	Cartón Couché

La facilidad brindada por la empresa Google de proveer los planos y tamaños para imprimir y construir un dispositivo Cardboard de forma casera demuestra que este producto es accesible para casi cualquier usuario de celular inteligente. De igual forma, para evitarse el trabajo de la construcción manual, siendo el material principal el cartón, su precio es muy bajo en el mercado, enfatizando la accesibilidad del producto. La única especialidad del producto serían los lentes, diferenciando a las versiones 1.0 y 2.0 pues el diámetro de lente superior del segundo caso nos permite utilizar celulares con una pantalla más grande de 5.5 pulgadas, mientras que la diferencia entre la distancia focal se refiere a la distancia que habrá entre el lente y el celular colocado para proyectar las imágenes de realidad virtual.



Figura 39. Diferencias entre Cardboard V1.0 (Construido de forma casera) y V2.0 (comprado)

Tabla 3
Celulares utilizados para casos de prueba

Nombre	Potencia del Procesador	Versión del sistema operativo Android	Giroscopio Incluido
Lenovo A5000	1.3Ghz	4.2.2	No
LG G2 D800	2.26Ghz	6.0.1	Sí



Figura 40. Celulares inteligentes utilizados. Lenovo A5000 y LG G2

Aparte del uso de un celular de gama baja (Lenovo A5000) y de uno de gama alta (LG G2), se decidió comprobar la diferencia entre el juego utilizando la eliminación e **instancias dinámicas** de los escenarios no visibles, descrito en la sección 3.3.6., así como sin este método.

4.1 Caso de prueba (C1)

Descripción: Prueba en Lenovo A5000.

Capturas:



*Figura 41. Lenovo A5000. La falta de Giroscopio impide la jugabilidad.
Autor: Andrés Bastidas F.*

Fotogramas por Segundo: Un rango entre 10 - 30 FPS.

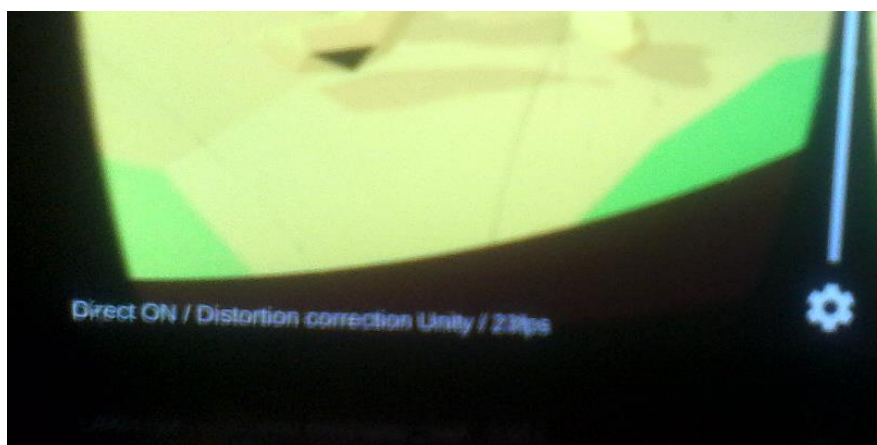
Estado: Desaprobado

Comentario: El celular pudo instalar correctamente la aplicación, pero fue el momento de intentar cambiar de escena a través de la rotación de la cabeza que no permitió un correcto funcionamiento pues el celular no cuenta con un giroscopio incluido, mostrando únicamente el cielo del videojuego.

4.2. Caso de prueba (C2)

Descripción: Prueba en LG G2, sin instancias dinámicas del escenario.

Capturas:



*Figura 42. Prueba en LG G2, sin instancias dinámicas del escenario. 23 FPS.
Autor: Andrés Bastidas F.*

Fotogramas por Segundo: Un rango entre 15 - 35 FPS.

Estado: Desaprobado

Comentario: Al estar todos los escenarios presentes al mismo tiempo, el celular tuvo dificultades para mostrar el juego con un alto rendimiento. Apenas obtuvo mejores resultados cuando se apuntó al cielo donde únicamente tuvo que renderizar el cóndor en el aire mientras que tuvo un peor desempeño cada vez que se instanciaban los obstáculos aleatorios. Es posible jugar, pero la falta de fotogramas por segundo es suficiente para causar inconformidad en el usuario hasta inclusive náuseas al intentar un uso más prolongado.

4.3. Caso de prueba (C3)

Descripción: Prueba en LG G2, instancias dinámicas del escenario.

Capturas:



*Figura 43. Prueba en LG G2, instancias dinámicas del escenario. 43, 36 Y 29 FPS.
Autor: Andrés Bastidas F.*

Fotogramas por Segundo: Un rango entre 25 - 45 FPS.

Estado: Aprobado

Comentario: Igualmente hubo rebajas en los fotogramas por segundo el momento que se instanciaban los obstáculos aleatorios, pero su desempeño fue mucho superior y no causo inconformidades.

Capítulo 5

Conclusiones y recomendaciones

5.1. Conclusiones

- La capacidad potencial de la realidad virtual de emular cualquier ambiente añadido de elementos que serían imposibles en nuestro entorno real, generará una demanda por contenidos y experiencias de todo tipo, en el cual el área de desarrollo informático puede aprovechar en el futuro venidero, preparándose en las multidisciplinarias tareas necesarias para desarrollar con esta nueva tecnología.
- En cuanto al desarrollo, un documento de desarrollo de juego es prioridad, pues permite tener constancia de lo planeado todo en un medio que puede ser comprendido por cualquier integrante del equipo para referencias futuras.
- Como particularidad de la realidad virtual, es notable la importancia de la optimización de polígonos para tener un desempeño óptimo a la hora de realizar ambientes tridimensionales, en especial si se desarrolla para plataformas móviles.
- El paradigma de diseño de interfaces de usuario tradicionales tiene que ser eliminado para proyectos VR, pues será necesario mantener al usuario informado a través de texto que se encuentra de forma tridimensional en la escena como si fuera un elemento más de nuestra nueva realidad.

- La capacidad de inmersión del usuario es importante, pues la interacción con los objetos a su alrededor no tiene que ser simplemente con botones, como se mantiene en el desarrollo tradicional, sino que tiene que ajustarse a acciones mucho más simples e intuitivas para el ser humano, como lo es la vista y el tacto.
- Con el crecimiento exponencial, el estado actual de la realidad virtual no tardará en evolucionar para implementar nuevos métodos para que el usuario se sienta inmerso dentro de la experiencia, iniciando por el control de como generamos audio para asemejarse a como nuestro cerebro percibe la distancia de una fuente de sonido procesando la diferencia entre la información recibida por cada uno de nuestros oídos, para luego pasar a generar sensaciones táctiles haciéndonos creer que estamos tocando elementos suaves o duros, fríos o calientes, y así continuar con el resto de nuestros sentidos.

5.2. Recomendaciones

- Al igual que cualquier otro proyecto de desarrollo de software, es importante mantener una planificación organizada y bien documentada, de forma que no existan detalles que puedan ser pasados por altos.
- Inclusive tratándose de una metodología ajustada específicamente para cada proyecto, siempre es preferible mantener una guía metodológica en la cual basarse a lo largo del desarrollo del proyecto sin importar la cantidad de desarrolladores envueltos o las diferencias entre las actividades a realizarse.
- Si bien el desarrollo de cualquier producto informático necesita de conocimientos matemáticos o de programación, es notable que para el campo de desarrollo de realidad virtual es necesario tener por lo menos un conocimiento superficial de manejo de modelos y ambientes tridimensionales, pues las posiciones y formas de los objetos tendrán que ser implementadas a través de código y debe haber una fácil comunicación entre el departamento visual y el funcional.
- Se recomienda bastante optimización de polígonos de los elementos tridimensionales, de forma que no sea necesario procesar absolutamente todos

los objetos que vayan a utilizarse en algún momento, sino únicamente aquellos que están a la vista del usuario y son importantes para la jugabilidad.

- Se recomienda mantener una interfaz simple pues puede causar disconformidad al usuario si este se encuentra con una aplicación que le muestre pocos fotogramas por segundo, o de igual forma que abrume al usuario al sentirse rodeado de demasiada información incomprensible a primera instancia, por lo que se recomienda incorporar la menor cantidad de texto posible. El objetivo de un proyecto de realidad virtual trata de emular la realidad agregándole elementos que no puede ser replicados en la realidad, por lo que la principal prioridad es que el usuario se sienta cómodo con su ambiente y entorno, creyendo encontrarse realmente dentro de este.
- En etapas futuras, se recomienda agregar conceptos utilizados en el desarrollo de videojuegos tradicional para mantener enganchado al usuario, como es un detallado sistema de accesorios y personalización del personaje, nuevas ubicaciones para jugar, potenciadores generados aleatoriamente para variar la dinámica del juego, una base de datos para comparación de puntajes a través de redes sociales, entre otros.
- Mantener claro un potencial modelo de negocio para promocionar y generar ingresos en la aplicación de forma que no sea únicamente una experiencia, sino que pueda ser mantenida constantemente.
- Por último, se recomienda mantener una mente creativa, de forma que se pueda desarrollar mejores métodos para la interacción entre el usuario y el juego, asemejándose a la interacción que el humano tiene con las herramientas utilizadas diariamente en su entorno.

Bibliografía

- AAMC. (Junio de 2014). AAMC. Obtenido de <https://www.aamc.org:https://www.aamc.org/newsroom/reporter/june2014/384790/technology-medical-education.html>
- Artaud, A. (1958). *The Theatre and its Double Trans. Mary Caroline Richards*. New York: Grove Weidenfeld.
- Blender 3D. (s.f.). Obtenido de <http://blender3d.es/>
- Blender. (s.f.). *Blender Decimate Modifier*. Obtenido de <https://www.blender.org/manual/modeling/modifiers/generate/decimate.html>
- BNL. (s.f.). *History: The First Video Game?* Obtenido de BNL: <https://www.bnl.gov/about/history/firstvideo.php>
- Brewster, S. D. (1856). *The Stereoscope; its History, Theory, and Construction, with its Application*. Londres.
- British Broadcasting Corporation. (Septiembre de 2008). Obtenido de <http://downloads.bbc.co.uk/rd/pubs/whp/whp-pdf-files/WHP169.pdf>
- Certimedic. (s.f.). Obtenido de <http://www.certimedic.es/percepcion-distancia-profundidad/>
- Digi-Capital. (Julio de 2015). *The 7 drivers of \$150 billion augmented/virtual reality*. Obtenido de <http://www.digi-capital.com/news/2015/07/the-7-drivers-of-150-billion-augmentedvirtual-reality>
- Entertainment Software Association. (2016). *www.thesa.com*. Obtenido de Thesa: <http://www.thesa.com/wp-content/uploads/2016/04/Essential-Facts-2016.pdf>
- Gamerant. (2014). *Oculus Rift Will Be Cheaper Thanks To Facebook; Aiming For 2015 Release*. Obtenido de <http://gamerant.com/oculus-rift-cheaper-after-facebook-buyout/>
- Gartner. (18 de Agosto de 2015). *Gartner's 2015 Hype Cycle for Emerging Technologies Identifies the Computing Innovations That Organizations Should Monitor*. Obtenido de <http://www.gartner.com/newsroom/id/3114217>
- Gartner. (s.f.). *Gartner Hype Cycle*. Obtenido de Gartner Hype Cycle: <http://www.gartner.com/technology/research/methodologies/hype-cycle.jsp>
- Google. (s.f.). Obtenido de <https://material.google.com/>
- Google. (Junio de 2016). Obtenido de <https://play.google.com/store/apps/details?id=com.google.samples.apps.cardboarddemo>
- Google VR. (2016). Obtenido de <https://developers.google.com/vr/unity/>
- Heavy. (2015). Obtenido de <http://heavy.com/tech/2015/07/best-vr-virtual-reality-headset-glasses-goggles-oculus-rift-specs-review/>
- Kickstarter. (s.f.). *Oculus Rift: Step Into the Game*. Obtenido de <https://www.kickstarter.com/projects/1523379957/oculus-rift-step-into-the-game>
- KillScreen. (2015). *KillScreen*. Obtenido de <https://killscreen.com/articles/failure-launch/>
- Ludosofía. (s.f.). Obtenido de <https://ludosofia.com/2008/09/29/la-suspension-de-la-incredulidad/>
- Mashable. (s.f.). *iOS*. Obtenido de <http://mashable.com/category/ios/>

Microsoft. (s.f.). *Visual C#*. Obtenido de <https://msdn.microsoft.com/es-ec/library/kx37x362.aspx>

Mountain Goat Software. (6 de Enero de 2015). Obtenido de <https://www.mountangoatsoftware.com/blog/how-i-work-and-use-scrum-personally>

MTech Games. (9 de Diciembre de 2015). *MTech Games*. Obtenido de <http://mtechgames.com/2015/12/09/physics-and-frame-rate-beating-motion-sickness-in-vr/>

NewZoo. (21 de Abril de 2016). Obtenido de <https://newzoo.com/insights/articles/global-games-market-reaches-99-6-billion-2016-mobile-generating-37/>

Oculus Rift España. (s.f.). Obtenido de <http://www.oculusriftespana.es/>

Omicrono. (15 de Diciembre de 2014). *5 razones por las que la realidad virtual fracasó en los 90 y ahora lo peta*. Obtenido de <http://www.omicrono.com/2014/12/5-razones-por-las-que-la-realidad-virtual-fracaso-en-los-90-y-ahora-lo-peta/>

Oxford, N. (21 de Septiembre de 2011). *Ten Facts About The Great Video Game Crash Of '83*. Obtenido de IGN: <http://www.ign.com/articles/2011/09/21/ten-facts-about-the-great-video-game-crash-of-83>

PlanetVB. (18 de Noviembre de 2013). *Nintendo introduces video game players to three-dimensional worlds with new virtual reality video game system*. Obtenido de <http://www.planetvb.com/modules/advertising/?r17>

Radar, T. (18 de Diciembre de 2014). *Google Cardboard: everything you need to know*. Obtenido de <http://www.techradar.com/news/phone-and-communications/mobile-phones/google-cardboard-everything-you-need-to-know-1277738>

Realidad, V. (s.f.). *Que es la realidad virtual*. Obtenido de <http://www.realidadvirtual.com/que-es-la-realidad-virtual.htm>

The NPD Group. (2015). *NDP*. Obtenido de NDP: <https://www.npd.com/wps/portal/npd/us/news/press-releases/research-shows-15.39-billion-dollars-spent-on-video-game-content-in-the-us-in-2013-a-1-percent-increase-over-2012/>

Thompson, J. (2007). *Game Design: Principles, Practice, and Techniques - The Ultimate Guide*. John Wiley & Sons.

Unity. (s.f.). *LA MEJOR PLATAFORMA DE DESARROLLO PARA CREAR JUEGOS*. Obtenido de <https://unity3d.com/es/unity>

Unity3D. (2016). Obtenido de <https://unity3d.com/es/learn/tutorials/topics/virtual-reality/user-interfaces-vr>

VentureBeat. (2016). Obtenido de <http://venturebeat.com/2016/05/21/google-daydream-standards/>

VentureBeat. (21 de Mayo de 2016). Obtenido de <http://venturebeat.com/2016/05/21/google-daydream-standards/>

Wearable. (16 de Septiembre de 2015). *The best VR headsets*. Obtenido de <http://www.wearable.com/headgear/the-best-ar-and-vr-headsets>

Xataka, A. (8 de Febrero de 2011). *¿Qué es Android?* Obtenido de <http://www.xatakandroid.com/sistema-operativo/que-es-android>