

UNIVERSIDAD IBEROAMERICANA DEL ECUADOR

ESCUELA DE SOFTWARE

Trabajo de Titulación para la obtención Del Título de Ingeniería en Software

**“WHISTLE OR DIE” VIDEOJUEGO CON USO DE PATRONES ACUSTICOS
DIRIGIDO A LA POBLACIÓN UNIVERSITARIA DE LA UNIB.E**

Autor:

Accel Leonardo Guamán Chamba

Director:

Rafael Falconí, Magister

Quito, Ecuador.

Agosto 2020

CARTA DE APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN

Mgst.

Yoisy Pérez Olmos

Director(a) de la Carrera de Ingeniería de software

Presente.

Yo, Rafael Eduardo Falconí Salazar, Director(a) del Trabajo de Titulación realizado por Accel Leonardo Guaman Chamba, estudiante de la carrera de Ingeniería de Software informo haber revisado el presente documento titulado "WHISTLE OR DIE" VIDEOJUEGO CON USO DE PATRONES ACUSTICOS DIRIGIDO A LA POBLACIÓN UNIVERSITARIA DE LA UNIB.E, el mismo que se encuentra elaborado conforme al Reglamento de titulación, establecido por la UNIVERSIDAD IBEROAMERICANA DEL ECUADOR, UNIB.E de Quito y el Manual de Estilo institucional; por lo tanto autorizo su presentación final para los fines legales pertinentes.

En tal virtud autorizo a los Señores a que concedan a realizar el anillado del trabajo de titulación y su entrega en la secretaria de la Escuela.

Atentamente,



Mgst. Rafael Falconí

Director(a) del trabajo de titulación

CARTA DE AUTORÍA DEL TRABAJO

Los criterios emitidos en el presente Trabajo de Titulación "WHISTLE OR DIE' VIDEOJUEGO CON USO DE PATRONES ACUSTICOS DIRIGIDO A LA POBLACIÓN UNIVERSITARIA DE LA UNIB.E", así como también los contenidos, ideas, análisis, conclusiones y propuesta(s) son de exclusiva responsabilidad de mi persona, como autor(a) del presente documento.

Autorizo a la Universidad Iberoamericana del Ecuador (UNIB.E) para que haga de éste un documento disponible para su lectura o lo publique total o parcialmente, de considerarlo pertinente, según las normas y regulaciones de la Institución, citando la fuente.



.....

Accel Leonardo Guamán Chamba

No. De cédula de ciudadanía: 1752119972

Quito, 15/9/2020

AGRADECIMIENTOS

Agradezco en primer lugar a mis padres Fredy Guamán y Maribel Chamba por confiar en mí, sin ellos simplemente no me hubiese sido posible llegar a este punto de mi vida, agradezco su enorme entrega en su labor como padres y su lucha por nunca hacernos faltar nada a mí y a mi hermana Kaela Guamán, a quien agradezco por darme la inspiración y aliento necesarios para seguir adelante cada día.

El apoyo de mis familiares y amigos ha sido imprescindible para alcanzar este objetivo, a ellos un inmenso agradecimiento por su cariño y afecto. A mi mejor amigo Sebastián por estar siempre presente en las buenas y en las malas, Gracias.

Así mismo quiero agradecer a las personas que me ofrecieron su guía para poder culminar los estudios, al Mgst. Alirio Mejía por su enorme paciencia y comprensión, a la Doc. Yemala Castillo y Mgst. Yoisy Perez por su entrega y apoyo que brindan a los estudiantes en su formación académica, finalmente un enorme agradecimiento Mgst. Rafael Falconí, su colaboración ha sido indispensable para culminar con este proceso de estudio.

ÍNDICE

CARTA DE APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN	ii
CARTA DE AUTORÍA DEL TRABAJO	iii
AGRADECIMIENTOS	iv
ÍNDICE	v
ÍNDICE DE TABLAS.....	viii
ÍNDICE DE GRÁFICOS.....	x
ÍNDICE DE ANEXOS	xii
RESUMEN.....	xiii
CAPÍTULO 1.....	15
INTRODUCCIÓN.....	15
1.1 Planteamiento del problema	17
1.1.1 Interrogantes	20
1.2 Justificación	20
1.3 Objetivos de la investigación.....	22
1.3.1 Objetivo General	22
1.3.2 Objetivos Específicos	23
CAPÍTULO 2.....	24
MARCO TEÓRICO	24
2.1 Antecedentes de la investigación.....	24
2.2 Bases teóricas	26
2.2.1 Origen de los videojuegos	26
2.2.2 La industria de los videojuegos	27
2.2.3 La industria de los videojuegos en la actualidad	28
2.2.4 Videojuegos.....	28
2.2.5 Clasificación de los videojuegos.....	29
2.2.6 Géneros de los videojuegos	31
2.2.7 Metodología de Software.....	33
2.2.8 COCOMO II	41
2.2.9 Planning Poker	42
2.2.10 Pruebas Unitarias	43

2.2.11 Pruebas de integración.....	43
2.2.12 Herramientas a usar	44
2.2.13. Software a usar	44
2.2.14 SDK Kit de Desarrollo de Software.....	50
2.2.15 Interfaz de Programación de Aplicaciones (API)	50
2.2.16 Licencia de Software	50
2.2.17 GPL Licencia Pública General.....	51
CAPÍTULO 3.....	53
METODOLOGIA EMPLEADA	53
3.1 Metodología de la investigación.....	53
3.2 Población y Muestra	53
3.3 Técnicas de recolección de datos	55
3.4 Validez y confiabilidad	56
3.5 Técnica de análisis de datos.....	56
3.6 Metodología del producto.....	56
CAPÍTULO 4.....	58
DESARROLLO DE LA PROPUESTA.....	58
4.1 Planificación del proyecto de software.....	58
4.1.1 Resultados de la investigación	58
4.1.2 Recursos	69
4.1.3 El documento de diseño	71
4.1.4 Historias de Usuario	74
4.1.5 Estimación de tiempo	75
4.1.6 Estimación de Costos/Esfuerzo.....	75
4.1.7 Product Backlog	77
4.1.8 Diagrama de Gantt	77
4.1.9 Herramientas de gestión de proyecto.....	78
4.1.10 Sistema de control de versionado utilizado	79
4.2 Análisis y Diseño.....	79
4.2.1 Visionamiento y alcance.....	79
4.2.2 Modelo de procesos	83
4.2.2 Diagrama de clases.....	83
4.2.3 Diseño de Base de datos	85

4.2.4 Diseño de interfaz de usuario	85
4.2.5 Diseño de arquitectura	88
4.3 Desarrollo.....	88
4.3.1 Tecnologías utilizadas	88
4.3.2 Producto de software desarrollado	89
4.3.3 Acceso al producto software	91
4.4 Pruebas.....	91
4.4.1 Técnicas de pruebas	91
4.4.2 Herramienta de prueba utilizada	91
4.5 Resultados obtenidos	92
4.6 Manual de usuarios de la aplicación	92
CAPÍTULO 5.....	93
CONCLUSIONES Y RECOMENDACIONES.....	93
5.1 Conclusiones	93
5.2 Recomendaciones	94
GLOSARIO.....	95
BIBLIOGRAFÍA.....	97
ANEXOS	102

ÍNDICE DE TABLAS

Tabla 1. Comparación Metodologías Fuente: (Canós, Carmen, & Patricio, 2012)	35
Tabla 2. Tamaño de la muestra para cada estrato. Elaborado por Guamán (2020)	55
Tabla 3. Pregunta 2. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	59
Tabla 4. Pregunta 11. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	59
Tabla 5. Pregunta 3. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	61
Tabla 6. Pregunta 4. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	62
Tabla 7. Pregunta 5. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	63
Tabla 8. Pregunta 7. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	64
Tabla 9. Pregunta 8. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	65
Tabla 10. Pregunta 9. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	66
Tabla 11. Pregunta 10. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	67
Tabla 12. Pregunta 12. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	68
Tabla 13. Pregunta 13. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	68
Tabla 14. Requerimientos recomendados de sistema. Elaborado por: Guamán (2020)	69
Tabla 15. Herramientas de software a usarse. Elaborado por: Guamán (2020)	70
Tabla 16. Plantilla final del documento de diseño. Elaborado por Guamán (2020)	71
Tabla 17. Estimación de tiempo. Elaborado por Guamán (2020)	75

Tabla 18. Productividad de punto de aplicación. Fuente: (Sommerville, 2011)	75
Tabla 19. Cálculo de puntos de aplicación. Elaborado por Guamán (2020).....	76
Tabla 20. Versión final del Product Backlog. Elaborado por Guamán (2020).....	77
Tabla 21. Glosario de términos. Elaborado por Guamán (2020)	80
Tabla 22. Historia de usuario 01. Elaborado por Guamán (2020)	81
Tabla 23. Historia de usuario 02. Elaborado por Guamán (2020)	81
Tabla 24. Historia de usuario 03. Elaborado por Guamán (2020)	81
Tabla 25. Historia de usuario 04. Elaborado por Guamán (2020)	82
Tabla 26. Historia de usuario 05. Elaborado por Guamán (2020)	82
Tabla 27. Historia de usuario 06. Elaborado por Guamán (2020)	82

ÍNDICE DE GRÁFICOS

Gráfico 1. Modelo de Historia de Usuario. Elaborado por Guamán (2020)	39
Gráfico 2. Proceso de Desarrollo Huddle. Elaborado por Guamán (2020).....	40
Gráfico 3. Tabla de estimación de Poker. Elaborado por Guamán (2020)	42
Gráfico 4 Herramientas a usar en el proyecto. Elaborado por Guamán (2020)....	44
Gráfico 5. Distribución de la muestra según la pregunta 2 y 11. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	59
Gráfico 6. Distribución de la muestra según la pregunta 2 y 11. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	60
Gráfico 7. Distribución de la muestra según la pregunta 3. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	61
Gráfico 8. Distribución de la muestra según la pregunta 4. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	62
Gráfico 9. Distribución de la muestra según la pregunta 5. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	63
Gráfico 10. Distribución de la muestra según la pregunta 7. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	64
Gráfico 11. Distribución de la muestra según la pregunta 8. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	65
Gráfico 12. Distribución de la muestra según la pregunta 9. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	66
Gráfico 13. Distribución de la muestra según la pregunta 10. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	67
Gráfico 14. Distribución de la muestra según la pregunta 12 y 13. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)	68
Gráfico 15. Fórmula de estimación de esfuerzo por composición de aplicación. Fuente: (Sommerville, 2011)	76
Gráfico 16. Diagrama de Gantt del proyecto. Elaborado por: Guamán (2020)	78
Gráfico 17. Modelo de procesos del videojuego. Elaborado por (Guamán 2020)	83
Gráfico 18. Diagrama de clases del juego. Elaborado por Guamán (2020)	84
Gráfico 19. Modelo de base de datos. Elaborado por Guamán (2020)	85

Gráfico 20. Diagrama de navegación del software. Elaborado por (Guamán 2020)	85
Gráfico 21. Pantalla principal del juego. Elaborado por Guamán (2020)	86
Gráfico 22. Interface de pantalla de juego. Elaborado por Guamán (2020)	87
Gráfico 23. Interfaz de la pantalla Fin de Juego. Elaborado por Guamán (2020)	87
Gráfico 24. Arquitectura del videojuego. Elaborado por Guamán (2020)	88
Gráfico 25. Pantalla de inicio del videojuego. Fuente: Videojuego Whistle or Die. Elaborado por Guamán (2020)	89
Gráfico 26. Pantalla de juego. Fuente: Videojuego Whistle or Die. Elaborado por Guamán (2020)	90
Gráfico 27. Pantalla de Fin de juego. Fuente: Videojuego Whistle or Die. Elaborado por Guamán (2020)	90
Gráfico 28. Prueba en modo de juego aprobada. Elaborado por Guamán (2020)	92

NDICE DE ANEXOS

Anexo 1. Encuesta aplicada a los estudiantes de la UNIB.E. Elaborado por Guamán (2020).....	102
Anexo 2. Manual de usuario del videojuego. Elaborado por Guamán (2020)	103

RESUMEN

El presente proyecto tiene como finalidad el desarrollo de un videojuego 3D dirigido a los estudiantes de la Universidad Iberoamericana del Ecuador, con la intención de proveerles un medio de entretenimiento. Para lograr los resultados esperados se usó un enfoque de investigación cuantitativa con un diseño no-experimental de corte transversal, mientras que para el desarrollo del videojuego se usó la metodología Huddle, en primer lugar se recabó la información necesaria para especificar los requerimientos de software, para ello se llevaron a cabo 233 encuestas que fueron completadas satisfactoriamente, los datos aquí recabados definieron aspectos esenciales del videojuego como la temática, el género y las plataformas en las que estaría disponible, esto a su vez permitió pasar a la fase de análisis, donde se modelaron los diferentes diagramas necesarios para proseguir con la programación del videojuego, en la fase de desarrollo se realizó la codificación e integración con los diversos elementos multimedia que se crearon, finalmente se realizaron con éxito las pruebas unitarias y de integración. En conclusión, todas las fases del desarrollo terminaron de forma satisfactoria tras ejecutar los procesos propuestos por la metodología Huddle. En futuras investigaciones o proyectos relacionados a los videojuegos se recomienda usar esta metodología de desarrollo por su versatilidad y adaptabilidad.

Palabras clave: videojuegos, Unity, software, 3D, Huddle.

CAPÍTULO 1

INTRODUCCIÓN

Con el auge tecnológico de las últimas décadas, la población ha logrado tener mayor facilidad de acceso y por ende un mayor acercamiento a la tecnología y dispositivos móviles como herramienta de uso diario, en este contexto, las actividades lúdicas cotidianas se han logrado sistematizar en medios digitales como computadoras, tablets y smartphones, así los videojuegos buscan no solo entretener al usuario sino en algunos casos mejorar ciertos aspectos de sus vidas.

Además, el creciente desarrollo de herramientas para la creación de videojuegos sean IDE's (Entorno de desarrollo integrado), editores de gráficos, aplicaciones de animación, renderizado, e iluminación, edición de audio, modelamiento de gráficos tridimensionales, entre otras; ha dado a los programadores y empresas de software la facilidad para crear sus propios videojuegos logrando reducir drásticamente el tiempo de desarrollo y por ende los costos de producción y distribución, a pesar de que la programación de videojuegos no sea una tarea fácil por la cantidad de disciplinas que se deben manejar.

Por otro lado, en el Ecuador son muy pocas las empresas que se dedican por completo al desarrollo de videojuegos, a pesar de que es una industria que genera miles de millones de dólares al año y de que su oferta de aplicaciones aumenta aceleradamente al igual que sus usuarios (Pallero, 2018), esto hace evidente un gran potencial en esta industria, el cual debe ser explotado con el fin de contribuir al país, tanto en el campo económico como social.

Los videojuegos tienen una gran importancia en el ámbito del entretenimiento, siempre y cuando se los use con moderación pueden aportar grandes beneficios como reducir el estrés, aumentar la rapidez al tomar decisiones y mejorar la capacidad cognitiva. Felicia (2009, pág. 4) recomienda el uso de videojuegos incluso en el área educativa como soporte para la enseñanza, pues estos en el largo plazo permiten a los usuarios mejorar muchas de sus habilidades.

Sin embargo, a pesar de dicha importancia, actualmente la oferta académica en el Ecuador no ofrece carreras universitarias relacionadas con el desarrollo de videojuegos, por ello los estudiantes de carreras relacionadas a las TICs (Tecnologías de la información y comunicación) que quieren desempeñarse en este campo buscan recursos online como herramientas, cursos y soporte por parte de otros desarrolladores.

En virtud de ello es necesario que en el país se empiecen a desarrollar videojuegos para formar parte de la oferta internacional de este tipo de aplicaciones y diversificar los sectores comerciales en el país, por este motivo el presente proyecto busca crear un videojuego con la finalidad de ofrecer a los estudiantes una alternativa gratuita para su entretenimiento y además motivar a los desarrolladores y estudiantes pertenecientes a carreras afines a la informática a crear este tipo de productos de software. El presente documento se divide en 5 capítulos los cuales se mencionan a continuación:

En el primer capítulo se presenta el planteamiento del problema, esto permitirá al lector entender la situación actual del desarrollo de videojuegos y el impacto que tiene en los jóvenes, se presenta también la justificación y objetivos, los cuales permiten comprender por qué y con qué finalidad se realiza la presente investigación.

En el segundo capítulo se provee a los lectores las bases teóricas y antecedentes, esta recopilación de información sirve de premisa para comprender el contexto del proyecto y facilitar la interpretación del proceso de desarrollo del videojuego, esta información está enfocada en el área de software y abarca temas como la metodología, procesos, videojuegos y herramientas a usarse.

En el tercer capítulo se detalla la metodología empleada para la investigación, se abarcan temas como la recolección de datos, la muestra y población y la forma en que esos datos deben ser interpretados para obtener información útil que ayude en la creación del videojuego y en cómo este debe estructurarse con el fin de satisfacer las necesidades de los usuarios a los que está dirigido.

El cuarto capítulo presenta la propuesta de software, es decir se detalla el proceso de desarrollo, cubriendo los pasos de la metodología empleada, es decir en primer lugar se puntualiza la fase de análisis y diseño, posteriormente se detalla el proceso de desarrollo, y por último las pruebas.

Finalmente, en el capítulo quinto se presentan los resultados obtenidos con la investigación y las recomendaciones para futuros proyectos que abarquen la temática contenida.

1.1 Planteamiento del problema

El juego en los seres humanos e incluso animales ha estado presente desde siempre y se ha convertido en una característica fundamental (Huizinga, 1972), pues representa un escape de la cotidianidad hacia un momento de entretenimiento y diversión en el que las personas pueden olvidarse de sus problemas y sentirse cómodas por un instante de tiempo, sin importar su estatus social, religión o edad.

A su vez estos juegos a lo largo de las últimas décadas y gracias al avance tecnológico se han integrado en dispositivos electrónicos, los cuales al igual que los juegos tradicionales buscan entretener al usuario, la diferencia radica en que estos videojuegos o juegos electrónicos se ejecutan a través de una plataforma digital que puede ser una computadora, videoconsola, o máquina arcade, aunque actualmente también funcionan en dispositivos móviles como smartphones o tablets.

Por otro lado, los videojuegos permiten desarrollar las habilidades cognitivas y motoras, aumentar las destrezas y la capacidad competitiva y cooperativa de los jugadores, de acuerdo al género del juego, estos proveen diferentes beneficios formativos (Felicia, 2009), por ejemplo muchas veces la temática se basa en hechos históricos o bases científicas o la jugabilidad a menudo permite al jugador reforzar sus aptitudes mediante la repetición.

Juegos de estrategia como World of Warcraft, por ejemplo, enseñan al usuario un aprendizaje colaborativo, Bioscopia ayuda a los jugadores a entender mejor la

botánica, genética y biología, entre otros videojuegos que mejoran aspectos como el álgebra, liderazgo, ciencias, historia, etc. (Felicia, 2009, pág. 19). Algunos son desarrollados por empresas dedicadas a los Game-based Learning o juegos enfocados en el aprendizaje, mientras que en otros casos el beneficio formativo se da de manera indirecta.

Adicionalmente los videojuegos proporcionan a los jugadores un medio de entretenimiento que a su vez les ayuda a tener un pensamiento crítico pues la constante toma de decisiones a la que se someten los usuarios les obliga a tener que planificar previamente las acciones a tomar, como gestionar y distribuir los recursos disponibles en el juego y resolver problemas rápidamente, esto contribuye con su criterio personal sobre las elecciones frente a diversas alternativas disponibles (Ruiz & Silva, 2018).

Uno de los aspectos principales en cuanto a la calidad del videojuego es la experiencia del usuario, la cual tiende a mejorar significativamente cuando no se limita su interacción con el videojuego (generalmente mediante un mando o pantalla táctil), es decir permitir al usuario controlar a su personaje mediante sensores o accesorios adicionales como puede ser el micrófono, cuyos sonidos entrantes pueden ser analizados mediante un algoritmo para posteriormente transformarlos en acciones que se visualicen en la pantalla.

En la actualidad el proceso de programación de videojuegos viene ligado a la herramienta o motor de juego que se va a usar, entre los más populares se encuentran Unreal Engine, Unity para videojuegos en 3 dimensiones, y LibGdx o Cocos2d para 2 dimensiones, en estos motores mencionados previamente el desarrollo es más profundo, permitiendo tener un alcance más amplio y muy pocas limitaciones, sin embargo, estos requieren de mayores conocimientos sobre codificación y lenguajes de programación.

El lenguaje de programación está relacionado con el motor de juego, ya que este impone la codificación a usarse con base en su propia arquitectura y programación, en el caso de Unity, por ejemplo, el script se basa en Mono y los desarrolladores pueden usar C# o Boo, pero también permite la implementación de librerías

externas en otros lenguajes como C++, Python, Ruby, entre otros. Este tipo de complementos se ven limitados por la plataforma y no se encuentran disponibles en todos los motores gráficos.

Dependiendo de la herramienta a usarse y su licencia, cada una ofrece diferentes beneficios, ventajas y desventajas a la hora de desarrollar videojuegos. El acceso a funcionalidades de animación, renderizado de partículas en 3d y 2d, compresión y empaquetamiento de texturas a través de una interface, son algunos de los beneficios que ofrece Unity, mientras que en otras plataformas como LibGdx estos procesos se los debe realizar de manera manual a través de código.

Por ello, el tiempo de desarrollo depende en mayor magnitud de la plataforma o motor de juego que se vaya a usar, por ejemplo, Unity está en constante crecimiento con actualizaciones periódicas para corregir errores o agregar nuevas funcionalidades, además cuenta con una comunidad muy extensa, lo que facilita el desarrollo, pues ante la posible aparición de errores o problemas de lógica de programación, los foros permiten a los usuarios colaborar entre sí y llegar a una solución rápida.

Por otro lado, se han desarrollado herramientas que permiten a los usuarios crear juegos en tiempos muy cortos y sin necesidad de tener conocimientos de programación tales como: Gamefroot, Ren'Py, RPG Maker, entre otros que usan el enfoque de arrastrar y soltar, sin embargo, son altamente limitados y permiten crear juegos muy específicos, sin dar la posibilidad de agregar características especiales como las requeridas para el presente proyecto, sean control de sensores, animaciones, o la implementación de algoritmos matemáticos.

Por otra parte, con el ritmo de vida actual es muy frecuente que las personas estén en todo momento exigiéndose a sí mismos de mucha actividad física o intelectual, este exceso de trabajo genera estrés en las personas, que cuando se acumula puede llegar a ser muy perjudicial para la salud y resultar contraproducente en las actividades diarias pues el agotamiento mental reduce el rendimiento considerablemente (Burbano, 2015).

De esta forma, también los estudiantes universitarios pueden verse sometidos a situaciones de estrés debido a su proceso de formación profesional, ya sea por la presión de culminar los estudios o por la sobrecarga académica, en especial en los últimos períodos, que es cuando por las diversas labores extracurriculares los estudiantes tienen menos tiempo disponible y por ende las actividades recreativas pasan a representar períodos de muy corta duración (Burbano, 2015).

Por ello el presente proyecto busca conocer las necesidades de los estudiantes para la creación de un videojuego como una alternativa para su distracción, y así disminuir sus niveles de ansiedad, aunque su uso se dé en períodos cortos entre horas de clase o de camino a la universidad, se espera que en este tiempo los usuarios puedan tener un medio de distracción para evitar posibles problemas de salud mental y por ende mejorar su rendimiento académico.

1.1.1 Interrogantes

- ¿Cuáles son los requerimientos de los estudiantes de la Universidad Iberoamericana del Ecuador respecto a la creación de un videojuego?
- ¿Cuál es el diseño arquitectónico más adecuado para los requerimientos establecidos por los estudiantes de la Universidad Iberoamericana del Ecuador respecto a la creación de un videojuego?
- ¿Cómo desarrollar un videojuego 3D con reconocimiento de patrones acústicos en el motor de juego Unity?
- ¿Qué pruebas de desempeño se deben ejecutar en el videojuego para garantizar una óptima experiencia de usuario?

1.2 Justificación

El presente proyecto surge con el fin de brindar entretenimiento a los estudiantes universitarios en cuyo tiempo libre tienen a disposición un Smartphone y acceso a internet, el videojuego que se propone crear busca brindar una alternativa de distracción para períodos de aburrimiento que puedan inducir al usuario a situaciones estresantes como preocupaciones o ansiedad, que en este rango de edad (18-25 años) son bastante comunes, sin embargo no se limita a este grupo y

puede aportar los beneficios de los videojuegos a cualquier usuario con un dispositivo con acceso a internet.

Los aportes en el área de desarrollo de videojuegos con fines de entretenimiento tienen precedentes desde el año 1971 con el primer videojuego comercial denominado "Space Invaders", desde entonces la industria ha ido evolucionando hasta los juegos actuales con equipamientos adicionales como sensores, gafas, joysticks, guantes, entre otros accesorios. Su fin también se ha expandido, con el surgimiento de los Serious Games ha llegado a tener apertura en diferentes áreas como la medicina, enseñanza y el marketing.

Una de las ventajas de los videojuegos es que su fácil distribución les permite ser accesibles de manera global. Por ejemplo, Gy Medic, un sistema desarrollado por Guanoluisa, Pilatasig, Flores, & Andaluz (2019) permite detectar problemas de parálisis en el oro facial mediante la interacción del usuario con una interfaz que detecta patrones de movimiento en puntos estratégicos de la cara del individuo.

En cuanto a juegos que integren el reconocimiento de sonido para mejorar la experiencia de usuario la gama es muy amplia, desde usar el reconocimiento de voz para controlar la navegabilidad hasta usar el sonido de nuestras voces para hacer saltar un personaje o guiar a los supervivientes de un ataque a salir de una nave espacial como en "Mayday Deep Space" (Nelson, 2015).

Por otra parte, Chicken Scream es un juego en el que el usuario controla una gallina que debe llegar a la meta esquivando obstáculos y acumulando puntos, la particularidad de este juego es que el control del personaje se lo realiza mediante los gritos del usuario. Usando el sensor del micrófono el juego analiza la frecuencia del audio y detecta cuando la persona emite un grito, de esta forma el usuario controla el personaje mediante su voz.

Así, con el presente proyecto se prevé brindar a los estudiantes universitarios un medio de distracción basado en un software de entretenimiento que combine los aspectos esenciales de los videojuegos con una modalidad de interacción mediante la cual el usuario a través de la emisión de sonidos podrá ejercer el control del flujo de juego.

En el ámbito académico se espera brindar un marco de referencia para la creación de videojuegos orientado al desarrollo independiente (Juegos Indie) que permita a estudiantes y programadores tener una base en cuanto a la metodología y herramientas a usarse, teniendo en cuenta la poca accesibilidad a un presupuesto, manejo de recursos humanos y materiales limitados, ya sea con fines investigativos o de desarrollo.

En el medio científico y a su vez en un nivel más avanzado se prevé que el algoritmo de reconocimiento de sonidos implementado en el proyecto sirva como base para su integración en diversas áreas en las que se requiera, por ejemplo, en el RAH (reconocimiento automático del habla) cuyo principio se basa en el análisis de las señales de sonido y el reconocimiento de patrones relacionados con el habla.

En el ámbito metodológico se analizó la viabilidad de usar la metodología Huddle para el desarrollo del proyecto, debido a su adaptabilidad a los posibles cambios en los requerimientos del usuario, la investigación pretende mostrar que tan adecuada es esta metodología para el desarrollo de videojuegos. Así también la metodología de investigación cuantitativa para la obtención de los requerimientos de usuario.

Socialmente se espera que el videojuego tenga un impacto positivo en los estudiantes de la Universidad Iberoamericana del Ecuador, pues el objetivo es el de entretener al usuario durante su período de descanso, adicionalmente se busca atraer nuevos jugadores al captar su atención al ofrecer un enfoque de juego innovador en el que el usuario interactúa a través del micrófono y la emisión de sonidos.

1.3 Objetivos de la investigación

1.3.1 Objetivo General

Desarrollar un videojuego 3D multiplataforma con discriminación de patrones acústicos dirigido los estudiantes de la universidad Iberoamericana del Ecuador como medio de entretenimiento digital .

1.3.2 Objetivos Específicos

- Determinar mediante una encuesta los requerimientos de los estudiantes de la Universidad Iberoamericana del Ecuador para un videojuego.
- Realizar el diseño arquitectónico del videojuego como parte del proceso de desarrollo de software.
- Desarrollar el videojuego e integrar el entorno virtual con los scripts de programación en el motor de juego Unity 3D.
- Ejecutar pruebas de desempeño y calidad para mejorar la experiencia de los estudiantes de la universidad Iberoamericana del Ecuador al interactuar con el videojuego.

CAPÍTULO 2

MARCO TEÓRICO

El marco teórico es el conjunto de información disponible relacionada al tema o problema de la investigación, el objetivo de este apartado es sustentar con fuentes confiables y verificadas la teoría de la investigación, para dar una orientación en el estudio, saber cómo desarrollar la investigación en base a los antecedentes y prevenir posibles errores cometidos anteriormente (Hernández, Fernández, & Baptista, 2010).

En el presente capítulo se busca comprender todo el proceso de desarrollo de videojuegos, para lo cual se tiene como premisa la metodología de software a usarse, con base en lo anterior mencionado se abarcarán temas como técnicas de estimación de tiempo y esfuerzo, pruebas, herramientas a usarse, entre otros.

2.1 Antecedentes de la investigación

Los antecedentes son todas aquellas investigaciones previas relacionadas al tema de estudio, conocerlas a fondo permite tener otra perspectiva y facilita el proceso de investigación. Sin embargo, si estas no llegan a fundamentarse como base para el presente proyecto, su utilidad tendrá igual relevancia pues sirve como un punto de partida (Hernández, Fernández, & Baptista, 2010)

La investigación denominada 'Mobile application for vowel learning in children with Down Syndrome "LVDS-App"' de Escobar, Guamán, Montaluisa, Pruna, & Marín, (2017) muestra el despliegue de un videojuego con el objetivo de proveer a los niños una ayuda didáctica para el aprendizaje de las vocales, el videojuego creado en el motor LibGdx fue exportado únicamente la plataforma Android, a pesar de tener soporte para Linux, Windows, e iOS.

Para el proyecto se usó una metodología de software ágil XP (Extreme Programming) y un enfoque cuantitativo para la toma de requisitos y debido a que su susceptibilidad a cambios era bastante alta, la cual en conjunto con las buenas

prácticas implementadas y respetando los valores, características y roles de la metodología, permitió al equipo desarrollar un producto software de calidad.

Como resultado de la implementación del videojuego y parte importante de los factores que contribuyeron a obtener resultados positivos fue la integración de elementos multimedia llamativos como imágenes, sonidos y animaciones, por lo que en el presente proyecto de titulación se pondrá énfasis en dichos componentes.

Por otro lado, Mantilla & Sánchez (2018) desarrollan un videojuego para permitir a los estudiantes de primer y segundo grado del Instituto de Educación Especial Camilo Gallegos mejorar el proceso de aprendizaje de lenguaje de señas, enseñando a los niños las respectivas señales del alfabeto, esto mediante la implementación de minijuegos con una interface llamativa y la utilización de sensores externos como un Leap Motion.

Para el desarrollo del proyecto se escogió una metodología Ágil denominada Huddle (basada en Scrum), esta combina algunas de sus prácticas, pero es orientada a la creación de videojuegos de tal forma que se implementan nuevos roles como el Game Designer, el proyecto tuvo un enfoque cuantitativo y debido a sus resultados positivos se tomó muy en consideración las metodologías de software y de investigación implementadas por Mantilla & Sánchez (2018).

La metodología de desarrollo y la correcta toma de requerimientos hizo que los resultados del proyecto sean los esperados, pues al realizar las encuestas a los profesores que usaron el videojuego como soporte para la enseñanza, estos manifestaron que hubo una disminución en el tiempo de aprendizaje y una evidente mejora en la enseñanza, un factor importante para este resultado fue también el manejo de una herramienta externa como el Leap Motion, lo que motiva a usar este tipo de dispositivos.

Por otro lado, Vaca (2015) como parte de su trabajo de titulación desarrolla un videojuego educativo con el fin de realizar un análisis comparativo de los motores gráficos Three.js y Babylon.js, para lo cual usa la metodología de desarrollo de software OOADM.

En la investigación se evidencia el uso de la metodología OOHDM, la misma que fue considerada para aplicarla en el presente proyecto, sin embargo y debido a que está orientada al desarrollo web, fue omitida puesto que no es el enfoque del presente estudio. Por otro lado para la obtención de requerimientos el investigador usó la técnica de observación para analizar las necesidades de los usuarios finales del videojuego.

Finalmente Vaca (2015) concluye con el desarrollo exitoso del proyecto ofreciendo una guía para desarrolladores de videojuegos, además recomienda el uso de WebGL como plataforma para aplicaciones web, por lo que se tomará en consideración para la implementación del juego en caso de que los estudiantes requieran acceder al juego mediante un navegador de internet.

2.2 Bases teóricas

2.2.1 Origen de los videojuegos

Con fines investigativos se desarrolla el primer juego de la historia en 1971 de nombre Computer Space a manos de estudiantes del MIT, posterior se lanza la versión arcade en 1972, además fue el primer videojuego con fines comerciales, y contenía un sistema de créditos. El mismo año se lanza Table Tennis, un juego para la consola Magnavox Odyssey con casi 300.000 unidades vendidas, en secuencia Atari lanza una consola con el juego Pong (Iglesias, 2011).

En 1977 sale al mercado la Atari 2600, esta consola alcanzo un éxito considerable tanto en ventas, distribución y popularidad, esta fama se debió a los juegos distribuidos como Space Invaders, Missile Command y Pac-Man que le atribuyeron a la empresa cerca de 2.000.000 de dólares. En los siguientes años la presencia de juegos tanto como para consola y arcade tuvieron una mejora significativa en relación a los colores y graficas basadas en vectores como ejemplo se encuentra el juego Asteroids (Wolf, 2008).

Los juegos para computadoras personales comenzaron a tomar notoriedad en gran medida debido a la clonación de juegos exitosos de arcade como Donkey Kong y Pac Man, así surgen revistas y empresas con la finalidad de distribuir los códigos

fuentes y desarrollar los primeros juegos comerciales para PC como: Zork, Roberta Williams Mystery House y King's Quest (Dixon, Computer Nostalgia, 2017).

2.2.2 La industria de los videojuegos

A sus inicios la industria de los videojuegos centraba su idea de negocio en la venta de los cartuchos de juegos, años después la calidad de los mismos se vio afectada ante la gran demanda que existía en el mercado y a secuelas de videojuegos reconocidos, pero de bajo presupuesto, afectando las ganancias considerablemente.

La redención llega por parte de Nintendo y su consola NES (Nintendo Entertainment System) con una arquitectura de 8 bits, rápidamente supero los 60 millones de unidades con juegos como: Super Mario Bros y The Legend of Zelda, estos juegos contaban con una licencia que garantizaban la calidad frente a la competencia y la piratería. Los juegos para computadoras personales también contaban con una licencia denominada "Shareware" que permitía al usuario final hacer uso de una versión reducida del videojuego de forma gratuita (Dixon, Computer Nostalgia, 2015).

Con la aparición de juegos como Wolfenstein 3D y Doom surge el concepto de primera persona un género muy popular hasta la actualidad. En 1992 se liberó el juego "Dune II" el primer juego de estrategia y su tendencia se mantuvo hasta el desarrollo de los renombrados "Warcraft" de Blizzard y "Age of Empires" de Microsoft Games, muchos de estos juegos fueron lanzados como una versión online.

La primera tarjeta gráfica "3dfx Voodoo Chipset" lanzada en 1996, permitió gráficos de calidad y gran resolución, a estas alturas se popularizaron juegos del género multiplayer como: Ultima Online, Star Wars Galaxies y World of Warcraft con millones de jugadores en todo el mundo. La aparición de Java y Flash potencio el uso de navegadores para la ejecución de juegos simples evitando riesgos de seguridad y haciendo la experiencia de usuario más simple, los juegos representativos de esta categoría son: Half Life y Counter Strike.

2.2.3 La industria de los videojuegos en la actualidad

Actualmente la estructura y presentación de los videojuegos se ve limitada por la tecnología, de esta forma hace algunos años:

[...] la escasa capacidad de memoria de los equipos condicionaron la aparición de juegos gráficamente simples, por ello buscaron otros modos de atraer al gran público a través de juegos que estimulaban las habilidades y destrezas óculo-manuales, el resultado fue multitud de juegos adictivos y clásicos que perduran en la actualidad (Nieto, 2006, p. 14).

La famosa PlayStation 2 llega en el año 2000, rompiendo récords en ventas con 100 millones de unidades a su lanzamiento y 150 millones a la fecha, además aún se continúan fabricando, esta consola destaca por incorporar el soporte para DVD y juegos de la PlayStation 1, los juegos más destacados de esta consola son: God of War, Grand Theft Auto: San Andreas, entre otros (Sony, 2020).

Microsoft irrumpe el mercado del año 2001 con el lanzamiento de la Xbox con un procesador Intel Pentium II y un procesador gráfico Nvidia NV2, el esperado "Halo 2" finalmente llega vendiendo 8 millones de copias y la consola alcanza los 28 millones de unidades. Este sería un acontecimiento importante en la historia ya que introduce el concepto de juego online, que permitía interactuar a los jugadores en cualquier parte del mundo (Wolf, 2008).

En la actualidad la industria de los videojuegos genera ingresos millonarios, en el 2018 las ganancias llegaron a los 119.6 billones de dólares que representaron un aumento del 11% en comparación al 2017 (SuperData Research, 2018), demostrando un gran potencial de expansión y generación de empleo, pues es un campo del software que requiere múltiples disciplinas como modelado 3D, programación, animación, producción de audio y video entre otras.

2.2.4 Videojuegos

Los videojuegos han logrado trascender el ámbito del entretenimiento y ocio, pues además de esto:

[...] también han demostrado haber creado nuevas formas de relaciones sociales a través de juegos multijugadores y comunidades de jugadores que comparten e incluso crean contenidos. También han demostrado la capacidad de transmitir mensajes a través de nuevas formas de narrativas y, por otro lado, también han mostrado ser capaces de

perseguir otros objetivos, además del entretenimiento, como lo son la educación y el ejercicio físico (Iglesias, 2011, p. 70).

En la actualidad los videojuegos son la opción más importante para que niños y jóvenes sean parte del mundo de las TIC. Los videojuegos han sido de gran ayuda para que los niños adquieran capacidades y desarrollen diversas habilidades, además de familiarizarse con las nuevas tecnologías. (Belli & Raventós, 2008)

Un juego es una actividad basada en reglas con resultado variable y cuantificable, que requiere del esfuerzo del jugador para obtener dicho resultado. (Iglesias, 2011)

Chris Crawford, diseñador de videojuegos expone su definición de juego mediante cuatro conceptos (Crawford, 2003) Que son:

- Representación: Cuenta con dos definiciones del juego, uno objetivo que trata del sistema de reglas que está conformado el juego, y un aspecto subjetivo que se trata de una versión simplificada y subjetiva de una realidad emocional.
- Interacción: Jugar es un proceso activo.
- Conflicto: Consecuencia por la interacción del jugador.
- Seguridad: Forma segura de experimentar la realidad.

Por otro lado, un videojuego es un juego manipulado bajo el soporte de un dispositivo electrónico que consiste en una actividad donde una o varias personas pueden interactuar a través de un controlador o joystick, mediante un dispositivo que actúa como plataforma, la cual puede estar en un computador, consola, videoconsola o dispositivo móvil (León Mantilla & Sánchez Chacaguasay, 2018).

2.2.5 Clasificación de los videojuegos

Los videojuegos se clasifican en seis grupos (León Mantilla & Sánchez Chacaguasay, 2018):

2.2.5.1 Acción

La característica principal de ese tipo de videojuego es que necesita una gran percepción visual o precisión de los controles, se clasifica en:

- Shooters
- Pelea o lucha
- Plataformas
- Arcade

2.2.5.2 Aventura

Se identifica por contar una historia donde el personaje debe cumplir con objetivos para llegar a la meta, se clasifica en:

- Aventuras convencionales
- Aventura gráfica
- Survival Horror
- Hit and Run (golpe y corre)
- Juego de rol

2.2.5.3 Deportivos

Todo tipo de deporte de la vida real se lleva al mundo real, este tipo de videojuego permite jugar contra la máquina o con otros jugadores, se puede realizar campeonatos, competencias y administrar un equipo.

2.2.5.4 Estrategia

Mediante una planificación se lleva a cabo un orden de para llegar a la meta final, su característica principal es la gran cantidad de interfaces para detallar los recursos y sus acciones, se clasifica en:

2.2.5.5 Simulación

Tiene como objetivo principal recrear actividades o algún funcionamiento específico, se clasifica en:

- Simuladores de naves
- Simuladores de sistemas

- Simuladores sociales

2.2.5.6 Social

Cuenta con dispositivos externos que se comunica con dispositivos externos como cámaras, instrumentos, etc. Tiene como meta reunir a la mayor cantidad de personas para jugar dentro del mismo espacio físico, se clasifica en:

- Juegos de concurso
- Juegos musicales
- Juegos deportivos
- Juegos de movimiento corporal

2.2.6 Géneros de los videojuegos

Los géneros de videojuegos están distribuidos de acuerdo a una serie de elementos comunes a continuación se detalla algunos (Belli & Raventós, 2008).

2.2.6.1 Beat them up (Juegos de pelea a progresión)

Este género tiene similitud con los videojuegos de lucha, con la diferencia que en este género el jugador debe competir con varios oponentes, al mismo tiempo que los niveles avanzan.

2.2.6.2 Lucha

La característica principal de estos videojuegos es la recreación de combates entre personajes inspeccionados tanto por el computador, como por el jugador. El jugador ve a los competidores desde una perspectiva lateral como si fuera parte del público.

2.2.6.3 Juegos de acción en primera persona (FPS)

Opciones básicas como mover el personaje y usar un arma, el jugador aparece en la pantalla en primer plano y puede interactuar con el arma.

2.2.6.4 Acción en tercera persona

Los videojuegos en tercera persona tienen como objetivo con alternar entre pelar, disparos o interacción con el entorno, se juega desde una perspectiva desde atrás y en otros momentos desde una vista isométrica.

2.2.6.5 Plataformas

El jugador debe controlar a un personaje que avanza por escenarios esquivando obstáculos, saltando, escalando o agachándose, se mueve corriendo o saltando.

2.2.6.6 Simulación de combate

Se caracteriza por su extremo realismo en todos los aspectos de cada partida.

2.2.6.7 Arcade

Acción rápida de jugabilidad.

2.2.6.8 Sport

Estos juegos son parecidos a los juegos de deportes reales.

2.2.6.9 Carreras

Juegos que tienen un punto de partida y una meta, teniendo como objetivo llegar antes de los demás competidores.

2.2.6.10 Agilidad mental

Juegos para ejercitar la mente mediante la resolución de ejercicios que a medida que se va pasando los niveles la dificultad crece.

2.2.6.11 Educación

Juegos que además de promover la diversión aportan con la educación de las personas especialmente de los niños.

2.2.6.12 Aventura clásica

El jugador toma vida en un personaje que debe resolver acertijos y rompecabezas con diferentes objetos.

2.2.6.13 Aventura gráfica

"Point and click" juegos que no necesitan de la introducción de comandos, basta con un clic con el puntero.

2.2.6.14 Party games

En este juego cada jugador debe esperar su turno para poder realizar su jugada en un tablero virtual y a medida que avanza el juego ir superando varias pruebas.

2.2.6.15 Juegos On-Line

Este tipo de juegos tiene como objetivo la posibilidad de relacionar a participantes mediante la unión de equipos o la competición con otros jugadores de todo el mundo.

2.2.7 Metodología de Software

La industria del software es la encargada de desarrollar y poner en el mercado diferentes sistemas relacionados a las tecnologías de la información y comunicación, en este contexto:

La metodología de desarrollo de software son un conjunto de procesos que buscan ayudar a las empresas y desarrolladores a llegar a crear un producto de software, generalmente incluyen cuatro procesos primordiales: especificación, diseño e implementación, validación y evolución del software, aunque según la metodología esto difiera en algunos o muchos aspectos, implícitamente siempre estarán presentes estas cuatro actividades fundamentales (Sommerville, 2011).

El desarrollo de software es una tarea difícil de llevar a cabo y desde su aparición gran parte de los productos salidos al mercado han sido desarrollados sin una metodología, en las últimas décadas y con el fin de evitar los problemas que conllevaba desarrollar software de forma empírica, aparecen dos grandes grupos

de metodologías, por un lado, tenemos las metodologías tradicionales, con un estricto control de actividades y enfocadas en el control del proceso, y por otro las metodologías ágiles centradas en el valor humano, la participación del cliente e iteraciones cortas.

Una metodología de desarrollo debe cumplir los siguientes parámetros:

- Visión del producto
- Vinculación con el cliente
- Establecer un modelo de ciclo de vida
- Gestión de los requisitos
- Plan de desarrollo
- Integración del proyecto
- Medidas de progreso del proyecto
- Métricas para evaluar la calidad
- Maneras de medir el riesgo
- Como gestionar los cambios
- Establecer una línea de meta

Las metodologías ágiles surgen como una alternativa y reacción a las metodologías tradicionales e irrumpen con la burocracia de estas (Tinoco Gómez, Rosales, & Salas, 2010).

Tabla 1. Comparación Metodologías Fuente: **(Canós, Carmen, & Patricio, 2012)**

Metodologías Agiles	Metodologías Tradicionales
Basadas en heurísticas provenientes del desarrollo	Basadas en normas provenientes del entorno de desarrollo
Tolerante a cambios durante el proyecto	Resistencia a los cambios
Impuesta por el equipo	Impuesta externamente
Pocos controles en el proceso y principios	Proceso controlado, cuantiosas normas
Contrato flexible e inexistente	Contrato prefijado
El cliente participa en el desarrollo	El cliente mantiene interacción con el equipo de desarrollo en reuniones
Grupos pequeños menores a diez personas	Grupos grandes
Poca documentación	Alta documentación
Menor realce en la arquitectura de software	La arquitectura de software es esencial

El uso de las metodologías ágiles en empresas de desarrollo de videojuegos ha permitido que los proyectos sean entregados rápidamente cumpliendo con cada iteración propuesta, esto involucra la participación, colaboración y organización de cada miembro del equipo incluido el cliente y generando retroalimentación en cada fase (GaritaHernández, LizanoMadriz, & Cordero Esquivel, 2019). Algunas de estas metodologías ágiles para el desarrollo de videojuegos se mencionan a continuación:

- SUM
- 5M
- Huddle
- VGSC
- Levels

Para el desarrollo del proyecto se utilizará la metodología Huddle en función de los requerimientos del videojuego.

2.2.7.1 Metodología Scrum

Scrum es un marco de trabajo dentro del cual se pueden emplear varios procesos y técnicas, scrum muestra la eficacia relativa de las técnicas de gestión de producto y las técnicas de trabajo, de tal forma que se pueda mejorar continuamente el producto, el equipo y el entorno de desarrollo.

Scrum utiliza un enfoque incremental que tiene como fundamento la teoría de control empírico de procesos. Esta teoría se fundamenta en transparencia, inspección y adaptación; la transparencia, que garantiza la visibilidad en el proceso de las cosas que pueden afectar el resultado; la inspección, que ayuda a detectar variaciones indeseables en el proceso; y la adaptación, que realiza los ajustes pertinentes para minimizar el impacto de las mismas (Navarro Cadavid, Fernández Martínez, & Morales Vélez, 2013, p. 33).

El marco de trabajo Scrum consiste en los equipos Scrum y sus roles, eventos, artefactos y reglas asociadas. Cada componente dentro del marco de trabajo sirve a un propósito específico y es esencial para el éxito de Scrum. Las reglas de Scrum relacionan los roles, eventos y artefactos y rigen las relaciones e interacciones entre ellos (Schwaber & Sutherland, 2017).

2.2.7.1.1 El equipo Scrum

Scrum al igual que diversas metodologías de desarrollo proponen una serie de roles con el fin de organizar al equipo, en el caso de Scrum se tienen las siguientes funciones:

El *Scrum Master* tiene como función asegurar que el equipo está adoptando la metodología, sus prácticas, valores y normas; es el líder del equipo, pero no gestiona el desarrollo. El dueño del producto es una sola persona y representa a los interesados, es el responsable de maximizar el valor del producto y el trabajo del equipo de desarrollo; tiene entre sus funciones gestionar la lista ordenada de funcionalidades requeridas o *Product Backlog*. El equipo de desarrollo, por su parte, tiene como responsabilidad convertir lo que el cliente quiere (Navarro Cadavid, Fernández Martínez, & Morales Vélez, 2013, p. 33).

En recopilación, el equipo Scrum se conforma por:

- Dueño del Producto: responsable de maximizar el valor del producto resultante del trabajo del equipo de desarrollo.

- El equipo de desarrollo: consiste en los profesionales que realizan el trabajo de entregar un Incremento de producto “Terminado” que potencialmente se pueda poner en producción al terminar un sprint.
- Scrum Master: responsable de promover y apoyar Scrum como se define en la Guía de Scrum.

2.2.7.1.2 Eventos Scrum

En Scrum existen eventos predefinidos con el fin de crear regularidad y minimizar la necesidad de reuniones no definidas en Scrum. Todos estos eventos deben ejecutarse dentro de un bloque de tiempo, de tal modo que tienen una duración máxima. Los eventos de Scrum se listan a continuación:

- Sprint: corazón de Scrum, tiene un bloque de tiempo de un mes o menos en el cual se crea un incremento de producto “Terminado”.
- Planificación del Sprint: se planifica el trabajo a realizar, este plan se crea mediante el trabajo colaborativo del equipo Scrum completo.
- Objetivo del Sprint: meta establecida para el Sprint que puede lograrse mediante la implementación de la lista de productos.
- Scrum Diario: reunión con un bloque de tiempo de 15 minutos para el equipo de desarrollo, con el fin de optimizar la colaboración y desempeño del equipo inspeccionando el trabajo avanzado desde el último Scrum diario.
- Revisión de Sprint: al final de cada sprint se lleva a cabo una revisión de Sprint para inspeccionar el incremento y adaptar la lista de producto.
- Retrospectiva de Sprint: es una oportunidad para el equipo de Scrum de inspeccionarse a sí mismo y crear un plan de mejoras que sean abordadas durante el siguiente Sprint.

2.2.7.1.3 Artefactos de Scrum

En Scrum un artefacto representa trabajo o valor, “estos son subproductos de las actividades del marco de trabajo que le brindan dirección y transparencia al equipo. Los artefactos de Scrum son: *Product Backlog*, *Sprint Backlog*, Monitoreo de Progreso e Incremento” (Navarro Cadavid, Fernández Martínez, & Morales Vélez, 2013).

Los artefactos de Scrum representan trabajo o valor en diversas formas que son útiles para proporcionar transparencia y oportunidades para la inspección y adaptación, maximizando la comprensión del artefacto. Se consideran tres artefactos:

- Lista de Producto: es una lista ordenada de todo lo necesario conocido para el producto.
- Lista de Pendientes: es el conjunto de elementos de la lista de producto seleccionados para el Sprint.
- Incremento: es la suma de todos los elementos de la lista de producto completados durante un Sprint y el valor de los incrementos de todos los Sprints anteriores.

La Lista de Productos indica el costo estimado para completar un requisito y además contendrá todo lo que aporte un valor adicional al producto, por lo general la lista de productos contiene los objetivos del producto expresados en historias de usuario.

2.2.7.1.4 Historias de Usuario

En Scrum una historia de usuario consiste en la representación de los requerimientos del cliente, en estas:

[...] se definen las características de una entrega y el número de iteraciones que se necesitarán para terminarla. Para cada iteración el cliente define cuáles de las historias de usuario que componen la entrega funcional desea que se desarrollen. Se pueden crear o modificar historias de usuario en cualquier momento excepto cuando forman parte de una iteración en curso (Navarro Cadavid, Fernández Martínez, & Morales Vélez, 2013, p. 34).

Las historias son las descripciones de las funcionalidades que va a tener el software, estas son el resultado de la colaboración entre el cliente y el equipo, además estas van evolucionando durante toda la vida del proyecto (Trigas Gallego, 2012). Las historias de usuario se componen de tres fases:

- Tarjeta: breve descripción escrita que sirve de recordatorio
- Conversación: servirá para asegurarnos que este todo entendido y concretar el objetivo

- Confirmación: test funcionales para fijar detalles que sean relevantes e indica su límite

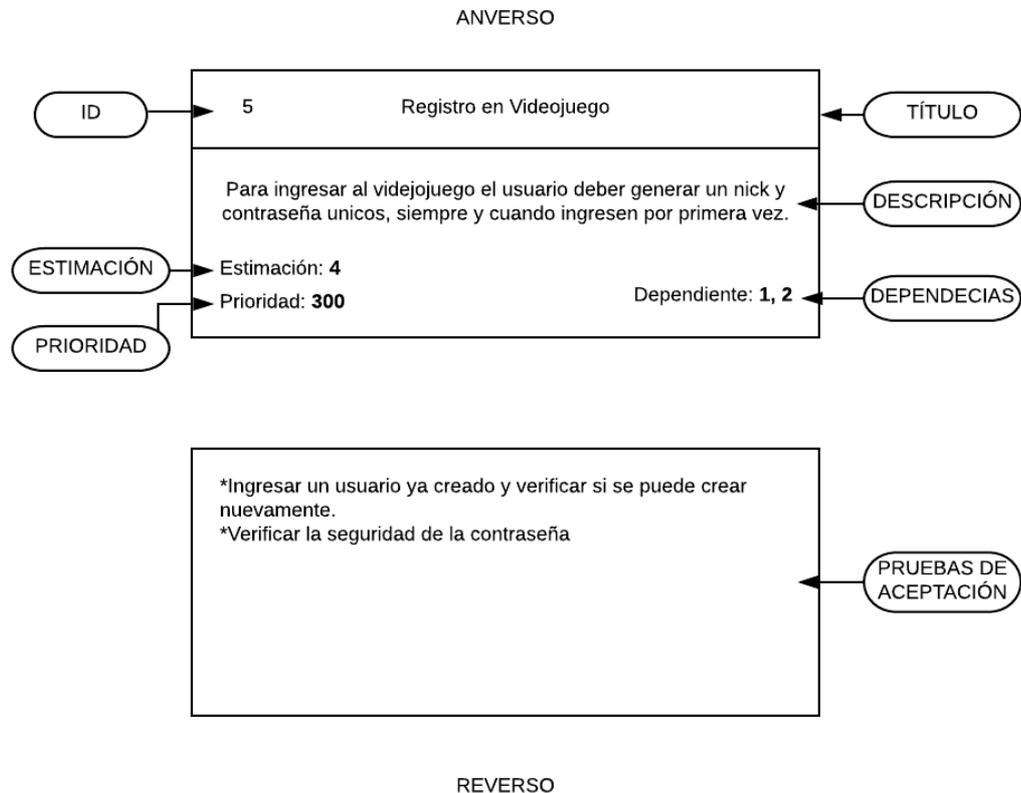


Gráfico 1. Modelo de Historia de Usuario. Elaborado por Guamán (2020)

El gráfico anterior representa el esquema general de una historia de usuario y sus componentes son: 1. *ID*. Es el identificador de la historia de usuario, 2. *Título*. Describe la historia de usuario, 3. *Descripción*. Descripción sintetizada de la historia de usuario, 4. *Prioridad*. Genera prioridad entre las historias de usuario, mientras más historias de usuario más prioridad, 5 *Dependencias*. Las historias de usuario deben ser independientes, pero a veces esto es inevitable.

2.2.7.2 Metodología Huddle

A partir de Scrum derivaron algunas metodologías adaptándose a ciertas áreas específicas del desarrollo de software, así se crea Huddle, como una alternativa para proyectos orientados al desarrollo de videojuegos, esta metodología se denominó de esa forma pues:

Se llama Huddle a la reunión que se realiza en el juego antes de cada jugada en el fútbol americano; la filosofía es que mediante breves reuniones de planeación a corto plazo, se planea cada "jugada" que se inicie; con esto se da un seguimiento más estrecho al avance del proyecto y es posible hacer correcciones tempranas a posibles desviaciones (Morales Urrutia, Nava López, Fernández Martínez, & Rey Corral, 2017, p. 28).

Huddle es una metodología específica basada en Scrum y como tal presenta las siguientes características:

- Ágil
- Equipos de 5 a 10 personas o incluso menor a 5
- Iterativo
- Incremental
- Evolutivo

A diferencia de la metodología Scrum, Huddle considera como sus roles más importantes al Game Designer y Project Manager:

- **Game Designer:** Quien está a cargo de la revisión de los requisitos en la etapa de preproducción, planifica y agenda características que se pueden añadir en los sprints.
- **Project Manager:** Quien está a cargo de actualizar los entregables y mantener la eficacia del equipo para que alcance los objetivos acordados al tiempo establecido

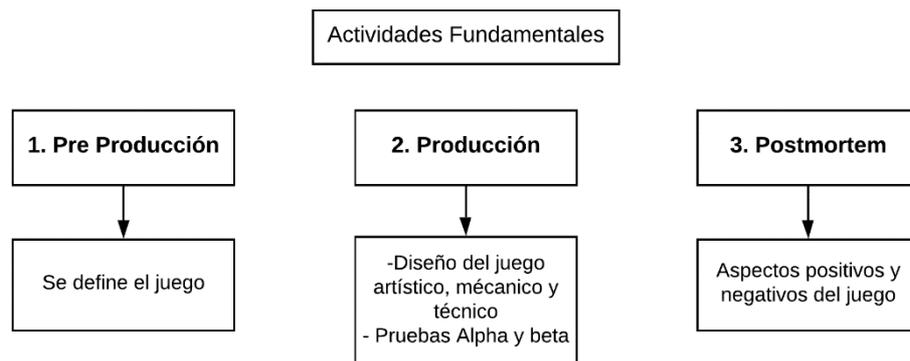


Gráfico 2. Proceso de Desarrollo Huddle. Elaborado por Guamán (2020)

El proceso de desarrollo de un videojuego según la metodología Huddle descrito en la gráfica, dividido en tres actividades: 1) Preproducción: en esta actividad se define el juego y su concepto general, así como sus características más notables y los términos del desarrollo, 2) Producción: una fase muy conflictiva por la

recopilación de diversas actividades y la participación de varios profesionales especializados en cada sub fase, esta actividad contempla el diseño del juego, artístico, mecánico y técnico, la implementación, pruebas alpha y beta, 3) Postmortem: esta actividad analiza los aspectos positivos y negativos del proyecto (Hernández, Pérez, & Correa, 2017).

2.2.8 COCOMO II

COCOMO (Costructive Cost Model) es uno de los modelos empíricos de estimación más populares, propuesto y desarrollado por Barry Boehm en los años 80, está dividido en 3 tipos de modelos:

- Básico
- Intermedio
- Avanzado

En cada modelo de COCOMO los proyectos se clasifican en tres tipos:

- Orgánico(fácil): Son proyectos que se realizan en pequeños grupos de trabajo, desarrollando aplicaciones que les resulta familiares.
- Semi - independiente(intermedio): Trabaja mediando entre proyectos orgánicos y de modo incorporado.
- De modo incorporado(avanzado): Estos proyectos tienen ciertas limitaciones estrictas para poder operar, de acuerdo al tipo de proyecto serán los valores de las constantes que usará la fórmula de COCOMO (Loiza).

Con el paso del tiempo ha crecido la complejidad de los desarrollos de software, se ha intentado abordar estas dificultades mediante el uso de nuevas metodologías, técnicas y métodos, el desarrollo de estos proyectos no solo se trata de actividades de desarrollo, sino también de actividades de gestión de dichos proyectos, entre las actividades se encuentra la estimación de proyectos software. Se han desarrollado algunas técnicas de estimación de esfuerzo de proyectos software, como son Puntos de función y COCOMO II.

Debido a la complejidad que de los proyectos software el modelo original COCOMO fue modificado, pasando a llamarse COCOMO II, este permite estimar el tiempo y esfuerzo de

un proyecto software en dos etapas: diseño temprano y post – arquitectura. COCOMO II permite calcular el tiempo y esfuerzo de un proyecto software a partir de los puntos de función sin ajustar, lo que es una gran ventaja, ya que en la mayoría de los casos es difícil establecer el número de líneas de código que tendrá un nuevo desarrollo, especialmente cuando se tiene poca experiencia previa a proyectos software (Pow Sang Portillo, 2015, pág. 95).

COCOMO es considerado uno de los mejores modelos de estimación de costos, este modelo permite determinar el esfuerzo y el tiempo que se requiere en un proyecto de software a partir de una medida de tamaño del mismo expresada en líneas de código que se estimen generar para la creación del producto software.

2.2.9 Planning Poker

Existen varias metodologías para el cálculo de los Story Points, siendo la más popular la de planning Poker, es una práctica ágil de estimación de software, el objetivo de esta metodología es que todo el equipo participe y llegue a un acuerdo de una manera amena y divertida una actividad que puede llegar a ser aburrida.

El sistema de trabajo de valoración en base a la secuencia Fibonacci 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89. A partir de esta secuencia, se acuerda una adaptación que presenta las siguientes opciones: 0, $\frac{1}{2}$, 1, 2, 3, 5, 8, 13, 20, 40, 100, ?, ∞ , (coffe).

0	1/2	1
2	3	5
8	13	20
40	100	∞
?		

Gráfico 3. Tabla de estimación de Poker. Elaborado por Guamán (2020)

En el grafico se selecciona: 0 cuando la tarea se resuelve con una funcionalidad out of the box, ∞ cuando el esfuerzo para realizar la tarea se considera superior a las opciones anteriores, por lo que la tarea se divide en varias tareas, cuando la respuesta sea “No estoy seguro”, es recomendable que, si en 30 segundos no se puede determinar que esfuerzo requiere, no se debería mostrar ninguna carta.

Coffee:

- Necesito un descanso
- El tiempo para realizar la tarea es el mismo tiempo que el de tomar un café (Prats, 2017).

2.2.10 Pruebas Unitarias

Las pruebas unitarias de software permiten evaluar por separado el correcto funcionamiento de los códigos que lo componen, con el fin de evitar errores a gran escala que con el paso tiempo requieren mayor inversión para su corrección (Villa Betancur & Giraldo Plaza, 2012, p. 147).

Existen dos enfoques principales para el diseño de pruebas unitarias:

- Caja Blanca: orientada a la estructura del código
- Caja Negra: orientada al correcto funcionamiento del código, analizando entradas y salidas que posee y verificando el resultado final.

2.2.11 Pruebas de integración

Las pruebas de integración prueban unidades relacionadas y verifican su relación conjunta, están centradas en la interacción y no en el funcionamiento individual, cada elemento debió haber sido probado, por ende, todo en conjunto debería funcionar bien. Con las pruebas de integración se puede identificar problemas de interfaces entre unidades, la relación entre lo que se espera de una unidad y lo que se ofrece (Peña, 2011).

Estas pruebas se realizan una vez se hayan aprobado las pruebas unitarias, se llevan a cabo para la comprobar que un conjunto de partes de software funciona.

2.2.12 Herramientas a usar

Las herramientas para el desarrollo de software consisten en un conjunto de programas que permiten al usuario crear, mantener y lanzar diversos tipos de productos software. En el ámbito de los videojuegos las herramientas están divididas en dos aspectos esenciales: programación y elementos multimedia, dentro de esta se encuentran las imágenes, animaciones, modelos tridimensionales, sonidos, etc.

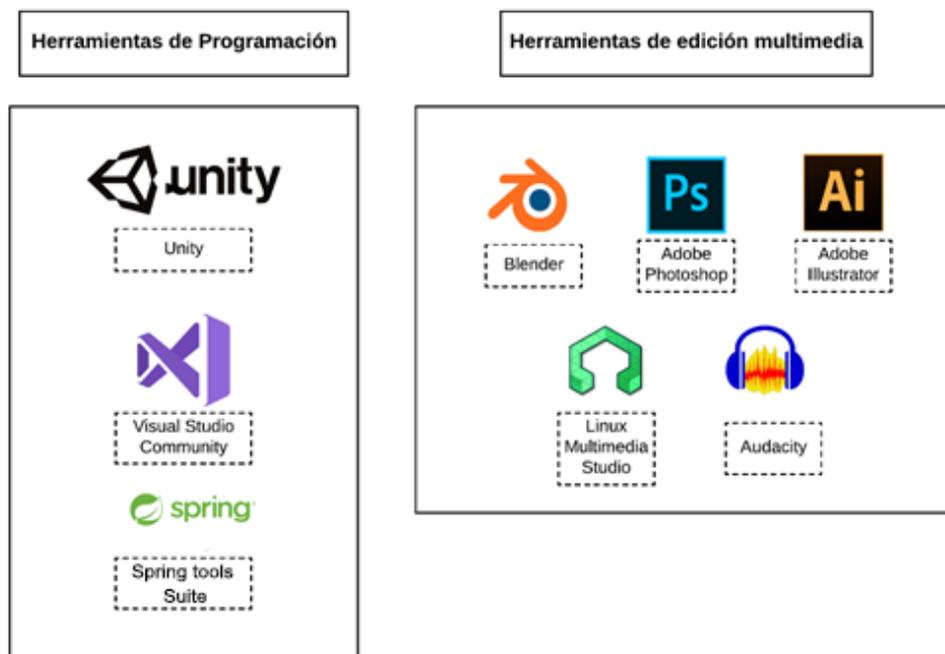


Gráfico 4 Herramientas a usar en el proyecto. Elaborado por Guamán (2020)

2.2.13. Software a usar

Para el desarrollo del proyecto de titulación se considera el uso del software descrito a detalle más adelante:

- Unity
- Spring Tools
- MongoDB

2.2.13.1 Unity

Es un motor de creación de videojuegos en 3D multiplataforma, admite la creación de juegos y varios contenidos interactivos tales como diseños arquitectónicos o animaciones 3D en tiempo real. Este entorno de desarrollo fue creado por la empresa Unity Technologies en 2005, esta empresa se ha esforzado para que cada herramienta de su creación sea accesible y fácil de usar, el equipo de desarrollo de Unity ha optado por mantener el código fuente brindando al usuario una interfaz gráfica completa para el control total del código fuente sin tener que recurrir a la creación de nuevos elementos en el código.

Este motor no admite el modelado, pero admite crear escenas que soportan laminación, texturas, cámaras, terrenos. Unity está disponible para plataformas Windows, Mac y Linux, permite crear aplicaciones compatibles con Windows, Mac OS X, iOS, Android, Wii, Playstation 3, Xbox 360, Nintendo, iPad, iPhone con Web.

Gracias a la facilidad de creación de videojuegos en Unity, se ha hecho acreedor a varios premios como ganador del premio de la innovación tecnológica del del Wall Street Journal en la categoría de software en 2010 (Ouazzani, 2012).

2.2.13.2 MongoDB

Es una base de datos de documentos que ofrece una gran escalabilidad y flexibilidad, y un modelo de consultas e indexación avanzado. El modelo de documentos de MongoDB resulta muy fácil de entender y utilizar, además proporciona todas las funcionalidades necesarias para satisfacer los requerimientos más complejos a cualquier escala (MongoDB, 2020). A continuación, se mencionan las características más notables:

- Almacenamiento de datos flexible, similar a JSON
- Asignación de datos en el código de la aplicación
- Consultas, indexación y agregación en tiempo real
- Base de datos distribuida en su núcleo
- Uso gratuito
- Alta disponibilidad con replicación y respaldo integrados

- Escalabilidad horizontal con fragmentación nativa
- Seguridad integral
- Herramientas de gestión de automatización, monitorización y respaldo

2.2.13.3 IDE Entorno de Desarrollo Integrado

Un entorno de desarrollo integrado IDE es un sistema software para el diseño de aplicaciones que combina varias herramientas del desarrollador comunes en una sola interfaz gráfica de usuario (Red Hat, 2020). Un IDE cuenta con las siguientes características:

- Editor de código fuente
- Automatización de compilación local
- Depurador

2.2.13.4 Visual Studio Community 2019

Es un IDE de Microsoft extensible y gratuito para crear aplicaciones, ofrece a los desarrolladores varias herramientas y características que los hace más productivos. Visual se ha convertido en potencia total durante los últimos años a la hora de suministrar herramientas para el desarrollo de software. Está disponible para Windows, Linux y macOS.

Visual Studio ofrece una gran variedad de herramientas como depuradores, editores, diseñadores, además admite programas en diferentes lenguajes como C++, C#, PHP, Python, HTML5, JavaScript, Visual Basic y F#. Proporciona entornos de desarrollo web como ASP.NET MVC, Django, etc., admite el desarrollo de aplicaciones para plataformas móviles Android, IOS y Windows Phone (Hernández & Del Valle, 2018).

2.2.13.5 Spring Tools

Es una suite adaptada para el desarrollo de aplicaciones empresariales con Spring Framework y Spring Boot, además proporciona soporte para las aplicaciones bajo Spring, este software cuya arquitectura se basa en el IDE Eclipse, incorpora tecnologías modernas para los desarrolladores (Vmware, 2020).

Entre sus ventajas se encuentran:

- Integración directa
- Fácil de usar
- Navegación y comprensión intuitiva dentro del código fuente
- Auto completar de código inteligente
- Información enriquecida en tiempo de ejecución

2.2.13.6 Control de versiones

En la fase de desarrollo de un software es de gran importancia disponer de una herramienta de control de versiones que permita el registro de modificaciones que se hayan realizado a las aplicaciones o documentos, y que proporcione las versiones previas del software, evitando problemas en el despliegue del producto final.

El control de versiones es el proceso de almacenar y recuperar cambios de un proyecto de desarrollo, los programas de control de versiones permiten retroceder a versiones anteriores para corregir o actualizar funciones y como característica adicional almacena todas las versiones generadas en una línea de tiempo (Tello Leal, Sosa R., & Tello Leal, 2012, p. 75).

Los principales programas para el control de versiones son:

- Git
- CVS
- Apache Subversion
- Mercurial
- Monotone

Para el manejo de versiones del videojuego propuesto en el proyecto de titulación se utilizará la herramienta Git.

2.2.13.6.1 GIT

En su trabajo Chacon menciona que “Git es un sistema de control de versiones distribuido gratuito y de código abierto diseñado para manejar proyectos escalables con gran velocidad y eficiencia, además es comprensión es fácil y genera un alto rendimiento” (Chacon, 2020).

Originalmente fue diseñado por Linus Torvalds con el objetivo de gestionar los diversos cambios que se realizan sobre los elementos que componen una aplicación, para su creación se consideró la eficiencia y la confiabilidad del mantenimiento de aplicaciones con un gran número de archivos de código fuente, proporcionando así una herramienta para trabajar en equipo de forma inteligente (García, 2015). Sus características principales son

- Gestión de ramas con fluidez
- Gestión distribuida
- Gestión eficiente para proyectos de gran escala
- Re almacenamiento periódico en paquetes

2.2.13.7 Herramientas de edición multimedia

Para la edición de contenido multimedia esencial dentro de un videojuego se lista como opción de uso los siguientes programas:

- Blender
- Adobe Illustrator
- Adobe Photoshop
- Linux Multimedia Studio
- Audacity

2.2.13.7.1 Blender

Es un programa informático gratuito multiplataforma, libre, de tamaño menor al de otros programas con características similares, con varias funcionalidades como modelado, texturizado, iluminación, renderizado, animación y creación de gráficos tridimensionales, además de ser un motor de videojuego que se controla a través de Logic Bricks y programación con el lenguaje de programación Python (Mendez, Obviedo, Fallas, Vega, & Méndez, 2018).

Esta herramienta además se caracteriza “porque al igual que las herramientas de animación incluye cinemática inversa, deformaciones por cuadrícula. Vértices de carga y partículas dinámicas y estáticas, edición y sincronización de audio y video, admite formatos gráficos como TIFF o SGI, TGA, JPG, Iris” (Shibuya, 1995).

2.2.13.7.2 Adobe Illustrator

Es un programa empleado para el dibujo vectorial y al diseño de elementos gráficos casi para todo tipo de dispositivo y soporte, creado hace más de 25 años, toma la forma de un taller de arte en una mesa de trabajo, está enfocado a la creación artística de dibujo y pintura para ilustración. Es desarrollado y comercializado por Adobe Systems, junto con Adobe Photoshop conforman la base de su actual Creative Cloud.

Adobe Illustrator trabaja con imágenes vectorizadas o vectoriales que constituyen puntos en un espacio virtual que se va uniendo mediante trazados, para posteriormente ser rellenados y así obtener imágenes de una muy buena calidad en cualquier tamaño (Carretero, 2014).

2.2.13.7.3 Adobe Photoshop

Dentro de la suite de Adobe se encuentra Photoshop, uno de sus programas más populares:

Adobe Photoshop es un programa que permite la edición de imágenes desarrollado por Adobe Systems Incorporated. Usado principalmente para el retoque de fotografías y gráficos, su nombre en español significa "taller de fotos". Photoshop admite la modificación de imágenes digitalizadas, además permite la creación y edición de imágenes como logotipos y gráficos, se puede editar la forma, la luz, el color, fondo, etc. Disponible para las plataformas Windows y Mac OS (Photoshop, 2019).

A la hora de elegir un software de imágenes digitales, Adobe Photoshop ha sido el estándar durante mucho tiempo.

2.2.13.7.4 Linux Multimedia Studio

Se trata de un entorno de trabajo de audio digital, libre ya que cuenta con una licencia GPL, este programa es multiplataforma disponible para Windows, Mac OS, Linux, ect., LMMS permite producir música profesional de una manera sencilla creando y sintetizando sonidos, tocando el teclado y organizando muestras, editar canciones y de esta manera componer pistas de audio, incluye un editor de compases y bajos para generar ritmos bajos permitiendo al usuario generar procesos de composición en varios niveles (Naranjo, 2018).

2.2.13.7.5 Audacity

Audacity es una aplicación informática muy útil multiplataforma libre y gratuita que se puede usar para grabación y edición de audios en vivo con la ayuda de un micrófono o un mezclador además de que permite digitalizar grabaciones de otros medios y edición de audio, con esta aplicación se puede lograr combinaciones de sonidos en varios formatos de diferentes archivos de uno o varios archivos a la vez, soporta 16, 24 y 32 bits, su interfaz es fácil de usar y entender. Disponible para plataformas Windows, Mac y Linux. Es el editor de audio y sonido más difundido en las distribuciones Linux (Audacity, 2019).

2.2.14 SDK Kit de Desarrollo de Software

Un SDK es un paquete de herramientas y código que facilita a los programadores generar software en un lenguaje, plataforma, aplicación, sistema operativo en específico.

La composición y distribución de un SDK está a cargo del o los desarrolladores originales del paquete interesados en que el software de terceros esté disponible en el mercado para su propio producto o en el lenguaje de programación en cuestión, los SDK son de uso gratuito, o ciertas veces limitados con reglas y licencias (IONOS, 2020).

2.2.15 Interfaz de Programación de Aplicaciones (API)

Una API es una serie de definiciones y protocolos utilizados para el desarrollarlo e integración de un software de aplicaciones, en cuestión una API es un contrato documentado que representa un acuerdo entre las partes involucradas, si una parte genera una solicitud con una estructura en particular, la otra parte responderá en función a esta estructura.

Las API permiten que sus aplicaciones se comuniquen entre si u otros servicios, sin la necesidad de saber cómo están implementados simplificando el desarrollo de las aplicaciones, además de ahorrar tiempo y dinero. Una API otorga flexibilidad, simplifican el diseño, genera innovación, optimiza la administración y el uso de las aplicaciones (Red Hat, 2020).

2.2.16 Licencia de Software

Una licencia de software es un contrato entre el licenciante (autor/titular de los derechos de explotación/distribución) y el licenciatarario (usuario consumidor,

profesional o empresa) del programa informático, para utilizarlo cumpliendo una serie de términos y condiciones establecidas dentro de sus cláusulas.

En el Ecuador la organización encargada de gestionar los registros de la propiedad es el IEPI (Instituto Ecuatoriano de Propiedad Intelectual) donde los desarrolladores y empresas pueden registrar su software a un costo de \$14.00, esta institución además reconoce obras literarias, marcas, patentes, entre otros recursos. (Servicio Nacional de Derechos Intelectuales, 2020).

En la actualidad el número de licencias para lanzar un software al mercado a tenido un incremento exponencial en relación al software libre que es el único que puede ser copiado, usado, modificado y distribuido como desee el usuario final, se consideran la siguiente lista como las principales licencias más usadas:

- Licencias GPL
- Licencia AGPL
- Licencia BSD
- Licencia Apache
- Licencia Creative Commons

Para la distribución del proyecto se utilizará la licencia GPL, conservando los derechos de autor y permitiendo la libre distribución, modificación y uso.

2.2.17 GPL Licencia Pública General

Su origen viene a partir de 1980 en Estados Unidos como una iniciativa de Richard Stallman y otros programadores quienes crean el proyecto GNU que consistía en desarrollar un sistema operativo compatible con Unix bajo la filosofía del software libre, en adición se crea la licencia publica general GPL que declaraba que los programas licenciados por esta cumplen la definición de software libre. Para evitar la creación de otras licencias que puedan modificar este concepto se crea el copyleft, una regla que garantiza las libertades fundamentales y transfiere dichas libertades a los receptores, para su efectividad hace uso de las leyes de propiedad intelectual renunciando a derechos que de ellas se derivan (Brocca & Casamiquela, 2005).

La licencia GPL permite la libre distribución y copia de un software o documentación con o sin ánimo de lucro, además de permitir obras derivadas mientras se

mantenga la misma licencia. GNU (GNU is Not Unix) y la FSF (Free Software Foundation) son estándares de promoción y defensa de la licencia GPL y como tal aconsejan adjuntar la versión completa de la misma en cada uno de los textos licenciados bajo el copyleft, además recomiendan evitar la carga de la licencia en línea por la inestabilidad de la información en red y los cambios que puede tener (Barandiaran, 2020).

CAPÍTULO 3

METODOLOGIA EMPLEADA

3.1 Metodología de la investigación

Para la presente investigación se usó el paradigma positivista, este modelo de investigación se aplica desde inicios de siglo XIX y se fundamenta básicamente en la suposición de la existencia de un mundo exterior (Lorenzo, 2006). Por ello esta investigación se enfocó en este paradigma, pues permite al investigador identificar las variables a ser medidas, y de esta forma discernir entre las preferencias en cuanto a videojuegos de los sujetos de estudio. Puesto que es la primera fase del presente proyecto y busca conocer las necesidades de los estudiantes de la universidad Iberoamericana del Ecuador.

El enfoque a emplear en la investigación es el cuantitativo, este enfoque “usa la recolección de datos para probar hipótesis, con base en la medición numérica y el análisis estadístico, para establecer patrones de comportamiento y probar teorías” (Hernández, Fernández, & Baptista, 2010, pág. 24). Por ende, a través de la medición de variables sobre temas referentes a los videojuegos, se obtuvo la información necesaria para establecer las preferencias de los estudiantes acerca de este tipo de software.

El diseño elegido es no-experimental de corte transversal, pues como definen Hernández, Fernández, & Baptista (2010) este tipo de investigación no manipula variables directamente sino se basa en la observación del fenómeno en su forma natural en un determinado lapso de tiempo. Así la presente investigación busca como su primer objetivo comprender las preferencias de los usuarios referente a los videojuegos durante el periodo en el que se desarrolló el proyecto.

3.2 Población y Muestra

La población es el conjunto de individuos que presentan características similares en algún lugar o tiempo determinado y que se delimitan con el fin de ser estudiados (Hernández, Fernández, & Baptista, 2010), por ello y debido a que la presente

investigación está dirigida a estudiantes de la Universidad Iberoamericana del Ecuador, la población comprende a todos los 588 estudiantes registrados en el periodo Agosto 2020, debido a que las preferencias en cuanto a videojuegos pueden variar según la carrera de cada estudiante, la población fue agrupada en los siguientes estratos: Software, Turismo, Gastronomía, Cosmiatría y Holísticas, Derecho, Enfermería, Nutrición y Dietética, Producción para medios de comunicación y Otras.

La muestra es un subconjunto delimitado de la población que se elige con el fin de ahorrar recursos y materiales (Hernández, Fernández, & Baptista, 2010). Para determinar la muestra primero es necesario definir la unidad de análisis de donde se obtendrán los datos para el estudio, que en el caso de la presente investigación es la población estudiantil de la universidad.

Para la obtención de la muestra se usa la fórmula:

$$n = \frac{N * Z^2 * p * q}{e^2 * (N - 1) + Z^2 * p * q}$$

Dónde: N es población = 588 estudiantes; Z el nivel de confianza de 95% con coeficiente de 1,96; E el error muestral = 5%; P la proporción de éxito= 0.5; y Q es la proporción de fracaso = 0.5.

$$n = \frac{(1.96)^2 * 0.5 * 0.5 * 588}{(0.05^2 * 587) + (1.96^2 * 0.5 * 0.5)}$$

$$n = 232.59 \approx 233$$

El muestreo elegido para la delimitación de la población es probabilístico, pues para este tipo de muestreo todos los participantes deben tener las mismas probabilidades de ser elegidos (Hernández, Fernández, & Baptista, 2010), el

subtipo de muestreo es por estratos, ya que se requiere comparar los resultados obtenidos por grupos, en este caso las diferentes carreras de la universidad.

Al ser un muestreo estratificado, para la presente investigación delimitaron 8 estratos agrupados mediante cada área de estudio de la Universidad Iberoamericana del Ecuador, dando como resultado la Tabla 2. con la proporción y número de muestra por cada estrato.

Tabla 2. Tamaño de la muestra para cada estrato. Elaborado por Guamán (2020)

Estrato	Identificación	Nº	Proporción	Muestra
		sujetos		
1	Ingeniería de software	42	7,1%	17
2	Licenciatura en gastronomía	68	11,6%	27
3	Licenciatura en cosmiatría holísticas e imagen integral	57	9,7%	23
4	Derecho	113	19,2%	45
5	Licenciatura en enfermería	92	15,6%	36
6	Licenciatura en nutrición y dietética	81	13,8%	32
7	Licenciatura en producción para medios de comunicación	80	13,6%	32
8	Otra	55	9,4%	22
	TOTAL		100%	233

3.3 Técnicas de recolección de datos

La recolección de datos tiene como objetivo detallar los procesos que permiten a través de técnicas o instrumentos reunir los datos necesarios para el estudio (Hernández, Fernández, & Baptista, 2010). Por consiguiente, la técnica de recolección de datos elegida fue la encuesta, pues para el desarrollo del videojuego se necesitó conocer las preferencias de los estudiantes en cuanto a videojuegos, como la temática, mecánica de juego, género y plataforma preferida. En este caso

se elaboró una encuesta de 13 preguntas que busca definir los requerimientos de los estudiantes para un videojuego.

3.4 Validez y confiabilidad

Para que la aplicación un instrumento de medición como el cuestionario sea eficiente debe cumplir con dos requisitos primordiales la validez y confiabilidad.

La validez del instrumento se define como el nivel en que este logra medir las variables que se están evaluando (Hernández, Fernández, & Baptista, 2010), por ello para confirmar la validez del instrumento se solicitó la revisión y aprobación de un grupo de expertos en las áreas metodológicas y técnicas. Tras la revisión y respectivas correcciones, se llegó a la versión final mostrada en el Anexo 1.

El segundo requisito para el instrumento es la confiabilidad, esta se define como: “el grado en que su aplicación repetida al mismo individuo u objeto produce resultados iguales” (Hernández, Fernández, & Baptista, 2010). Sin embargo, para el instrumento de la investigación se omitió el cálculo de la confiabilidad debido a que la mayoría de preguntas fueron de selección múltiple y por ende no es posible realizar el cálculo de la varianza para aplicar el alfa Cronbach. Adicional a esto la diferencia entre las áreas de estudio de los estratos seleccionados hacen que los resultados esperados sean altamente variantes.

3.5 Técnica de análisis de datos

El análisis se lleva a cabo una vez que se han obtenido los datos necesarios a través del instrumento, estos datos deben pasar por un proceso previo de verificación y depuración de errores (Hernández, Fernández, & Baptista, 2010), para el análisis se usó Microsoft Excel 2016 el cual permite obtener gráficos estadísticos y los porcentajes en los resultados de cada pregunta, los cuales permitieron definir los requerimientos de usuario.

3.6 Metodología del producto

Para el desarrollo del videojuego se empleó la metodología ágil Huddle, al ser esta una variación de Scrum ofrece tiempos de desarrollo rápidos pues está orientada

al producto, además al estar dirigida al desarrollo de videojuegos incorpora procesos que ofrecen facilidad en esta rama de la programación, de esta forma se convierte en una metodología ideal para el presente proyecto.

Los roles abordados por esta metodología de desarrollo son una adaptación de los planteados en Scrum, con variaciones orientadas al desarrollo de videojuegos, como la incorporación de un diseñador del juego. Las actividades de los roles también tienen ligeros cambios adaptados a sus fases: Preproducción, producción y postmortem.

A pesar de que Huddle está orientada al desarrollo de videojuegos, esta se mantiene muy apegada a los principios de Scrum, uno de ellos es la adaptación, por este motivo se tomaron únicamente las actividades que se consideraron explícitamente necesarias, tomando en cuenta el tamaño, la experiencia y la temática del proyecto se reemplazaron aspectos de Huddle como la elaboración de la plantilla de diseño por las historias de usuario.

El desarrollo general del proyecto siguió también las fases de la metodología, en la etapa de preproducción se establecen los requerimientos mediante la especificación del documento del diseño y la elaboración de las historias de usuario.

En cuanto a la fase de producción, esta es equivalente en Scrum a la implementación, sin embargo, se consideran aspectos esenciales en el desarrollo de videojuegos como la incorporación de los elementos multimedia y su integración con el entorno del juego y los scripts de control de dichos objetos.

Finalmente, en la fase Postmortem propuesta por Huddle también se realizó una adaptación, debido a la complejidad del proyecto se omitió documentación innecesaria y en su lugar se incorporaron los procesos de pruebas unitarias y de integración.

La documentación planteada por la metodología se muestra en sus versiones finales en el siguiente capítulo, pues al ser una metodología iterativa, los documentos como la plantilla del diseño o las historias de usuario sufrieron diversos cambios hasta llegar a su versión final.

CAPÍTULO 4

DESARROLLO DE LA PROPUESTA

4.1 Planificación del proyecto de software

En esta sección se documentan aquellas actividades planteadas por la metodología de desarrollo Huddle concernientes a la planificación del proyecto, sin embargo y debido a que esta metodología se basa en Scrum, fue de aquí de donde se tomaron aquellas tareas que no estaban debidamente especificadas en Huddle o que no se consideraron adecuadas para el proyecto.

4.1.1 Resultados de la investigación

Los resultados son el análisis de los datos obtenidos a través de las encuestas realizadas a los estudiantes de la Universidad Iberoamericana del Ecuador, estas encuestas fueron aplicadas mediante la herramienta de Formularios de Google Docs (ver Anexo 1), allí se contabilizaron un total de 287 encuestas llenas, sin embargo, se tomaron en cuenta únicamente 233 respetando el tamaño de la muestra calculado en el capítulo previo.

Los datos obtenidos a través de esta encuesta permiten tener una visión más amplia acerca de los requerimientos que tienen los sujetos de estudio sobre las preferencias respecto a los videojuegos, esta información es de suma importancia en la fase de planificación de software, debido a que las historias de usuario o plantillas de los documentos de producción se basan en las necesidades del cliente final.

A continuación, se muestra la tabulación de las respuestas obtenidas, y sus respectivos gráficos.

Tabla 3. Pregunta 2. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 2. ¿Hace uso de videojuegos?		
Opciones Pregunta 2	Frecuencia Simple	Frecuencia Porcentual
Si	128	54.9%
No	105	45.1%
Total	233	100%

Tabla 4. Pregunta 11. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 11. ¿Crees que un videojuego te permita distraerte en tu tiempo libre?		
Opciones Pregunta 11	Frecuencia Simple	Frecuencia Porcentual
Sí le permite distraerse	104	81.2%
No le permite distraerse	24	18.8%
Total	128	100%



Gráfico 5. Distribución de la muestra según la pregunta 2 y 11. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

En las tablas y gráficos anteriores se muestran los resultados obtenidos en las preguntas 2 y 11, lo cual muestra que, de las 233 personas encuestadas, un 54.9%, es decir 128 personas hacen uso de videojuegos, de este grupo 94 personas creen que un videojuego les permitiría distraerse en su tiempo libre, lo cual representa un 40.3% del total de los encuestados. Por tal motivo el desarrollo de un videojuego para el uso y distracción de los estudiantes se considera factible pues la mayoría de ellos llegaría a hacer uso de él.

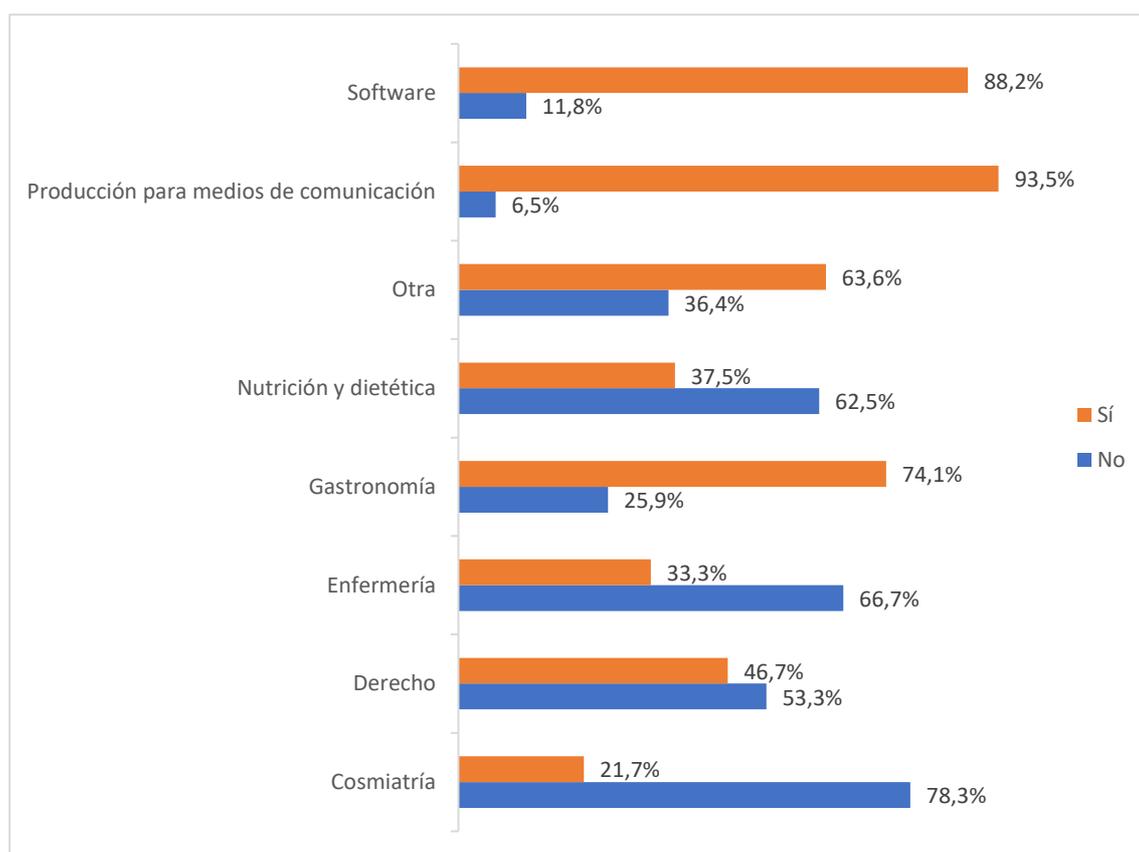


Gráfico 6. Distribución de la muestra según la pregunta 2 y 11. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

Como se puede observar en el Gráfico 6, el uso de los videojuegos en la universidad está altamente sesgado por la carrera universitaria, siendo Ingeniería en Software, Producción para medios de comunicación y Gastronomía las carreras que hacen más uso de videojuegos, por ello para futuras investigaciones y proyectos relacionados se recomienda poner énfasis en estas ramas de estudio.

Tabla 5. Pregunta 3. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 3. ¿Con qué frecuencia juega videojuegos?		
Opciones	Frecuencia Simple	Frecuencia Porcentual
Cada dos días	36	28,1%
Todos los días	38	29,7%
Una vez al mes	14	10,9%
Una vez por semana	40	31,3%
Total	128	100%

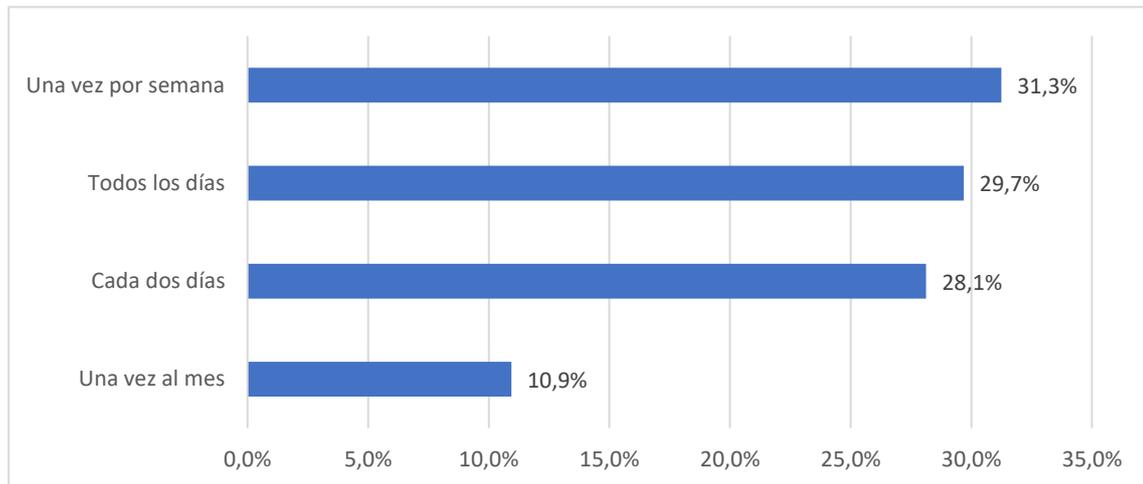


Gráfico 7. Distribución de la muestra según la pregunta 3. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

Como se puede visualizar en la tabla y gráfico anterior, se muestran los resultados obtenidos en la pregunta 3, evidenciando que de los 128 usuarios que hacen uso de videojuegos el 89.1% juega como mínimo una vez por semana, por tal motivo el videojuego debe estar orientado a poder usarse por largos períodos de tiempo sin incidir en la repetición o rutina.

Para lograr esto se debe implementar un factor sorpresa mediante un algoritmo de generación de terreno, que permita al usuario visualizar un entorno virtual distinto en cada partida, de esta forma la ubicación de obstáculos y recompensas será

siempre distinta e impredecible, así se espera evitar que el usuario se aburra fácilmente del juego tras cierto tiempo de uso.

Tabla 6. Pregunta 4. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

Pregunta 4. ¿Qué plataforma prefieres para jugar videojuegos?		
Opciones	Frecuencia Simple	Frecuencia Porcentual
Consola de Videojuego	28	21.9%
PC	44	34.4%
Smartphone	51	39.8%
Tablet	5	3.9%
Total	128	100%

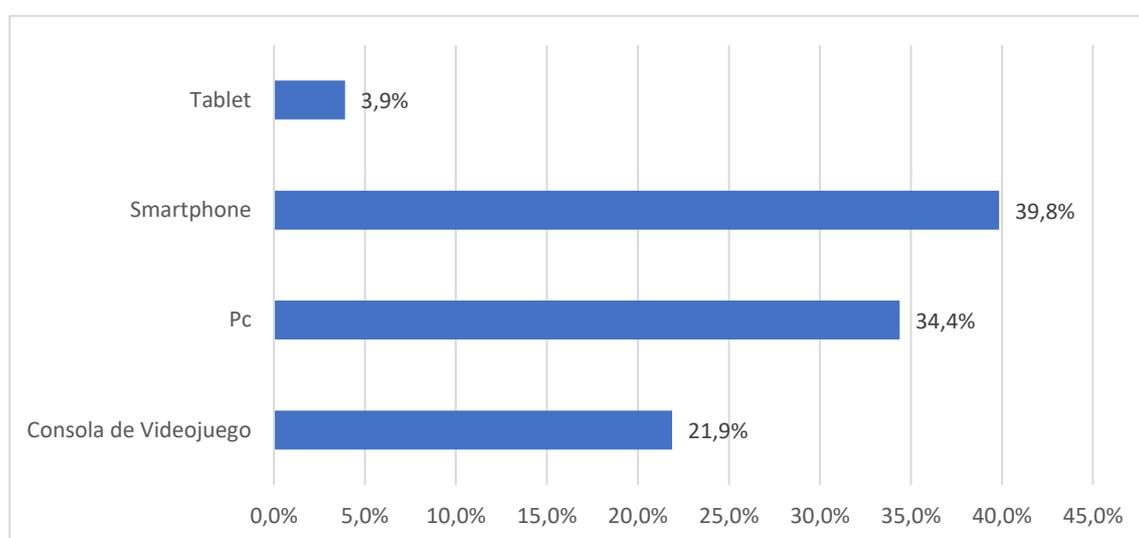


Gráfico 8. Distribución de la muestra según la pregunta 4. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

Como se puede visualizar en la tabla y gráfico anterior se muestran los resultados obtenidos en la pregunta 4, evidenciando que de los 128 usuarios encuestados que hacen uso de videojuegos, la mayoría tienen preferencia hacia dos plataformas, las cuales son: Smartphones y Pc, con un 39.8% y un 34.4% respectivamente. Los resultados demuestran que el videojuego debe programarse para las plataformas Smartphone y Pc.

Tabla 7. Pregunta 5. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 5. ¿El género de simulación y carreras le es llamativo a la hora de jugar un videojuego? (ejemplo: F1, Flight Simulator, Dirt Rally)		
Opciones	Frecuencia Simple	Frecuencia Porcentual
Sí	103	81.3%
No	24	18.8%
Total	128	100%

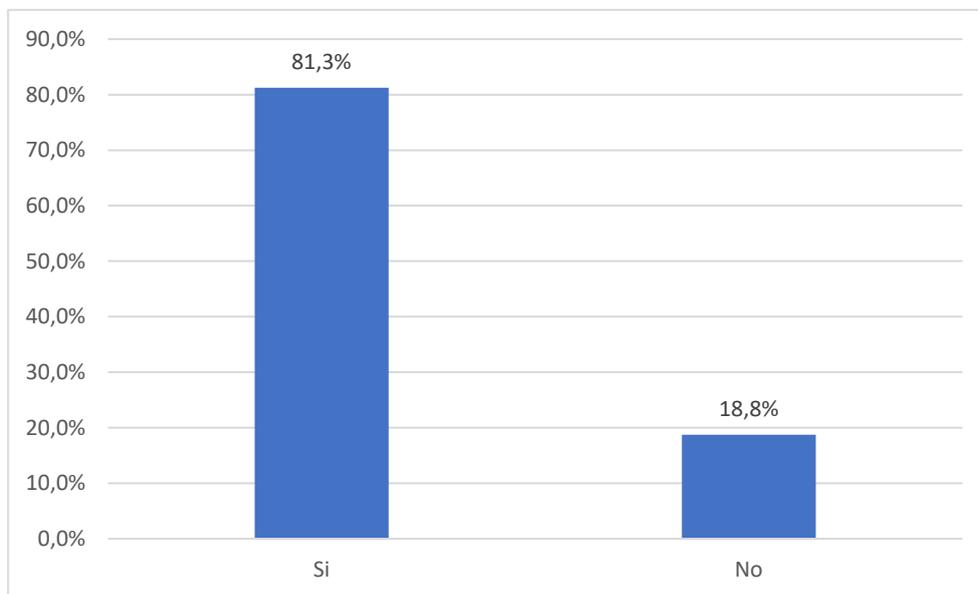


Gráfico 9. Distribución de la muestra según la pregunta 5. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

Como se puede observar en la tabla y gráficos anteriores el género de videojuegos de simulación y carreras le es llamativo para el 81.3% de los usuarios, por ello el objetivo del presente juego consistirá en simular el manejo de un vehículo aéreo, mediante controles táctiles o del teclado, mientras se trata de alcanzar el mayor puntaje posible.

Tabla 8. Pregunta 7. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 7. ¿Consideras indispensable el multijugador en un videojuego?		
Opciones	Frecuencia Simple	Frecuencia Porcentual
Sí	62	48.4%
No	66	51.6%
Total	128	100%

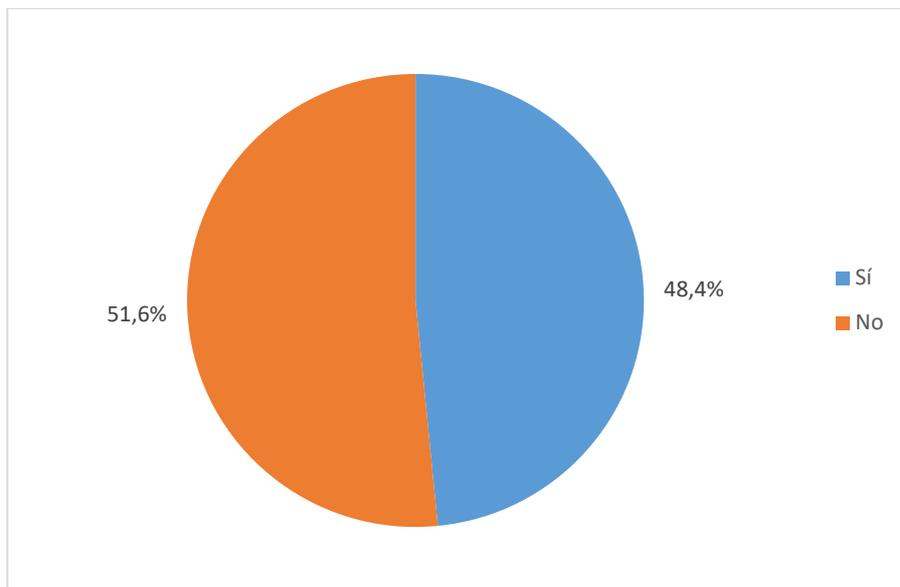


Gráfico 10. Distribución de la muestra según la pregunta 7. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

En el gráfico y figura anterior se puede visualizar la opinión de los usuarios acerca del multijugador en un videojuego, siendo que el 51.6% de ellos no lo considera como un factor indispensable, por esta razón, el proyecto no cuenta con esta funcionalidad, sin embargo, debido a la pequeña diferencia se implementará un sistema de puntuaciones globales, en el que los usuarios podrán competir para estar en los primeros lugares de la lista.

Tabla 9. Pregunta 8. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 8. ¿De las siguientes temáticas de videojuegos cual es de tu preferencia?		
Opciones	Frecuencia Simple	Frecuencia Porcentual
Medieval	21	16.4%
Futurista	22	17.2%
Infantil	3	2.3%
Fantástico	30	23.4%
Guerras	52	40.6%
Total	128	100%

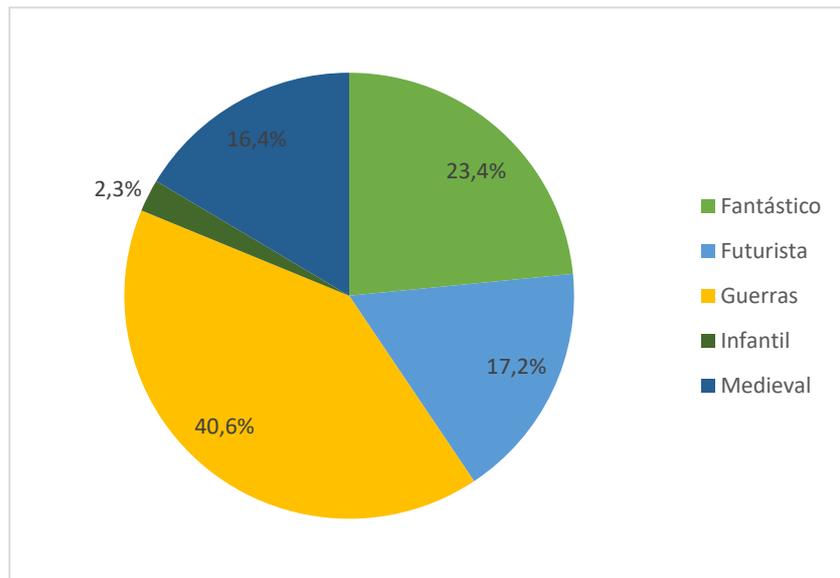


Gráfico 11. Distribución de la muestra según la pregunta 8. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

La tabla y gráfico previos muestran la temática de juego preferida por los usuarios, se puede visualizar que la mayoría, es decir el 40.6% prefiere las guerras, por este motivo el entorno gráfico del videojuego, así como sus personajes y elementos principales deben tener modelos referentes a una guerra, en específico y de acuerdo al género de simulación, el entorno será una ciudad abandonada posguerra.

Tabla 10. Pregunta 9. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 9. ¿Qué tipo de mundo prefieres dentro del videojuego?		
Opciones	Frecuencia Simple	Frecuencia Porcentual
2D	22	17.2%
3D	106	82.8%
Total	128	100%

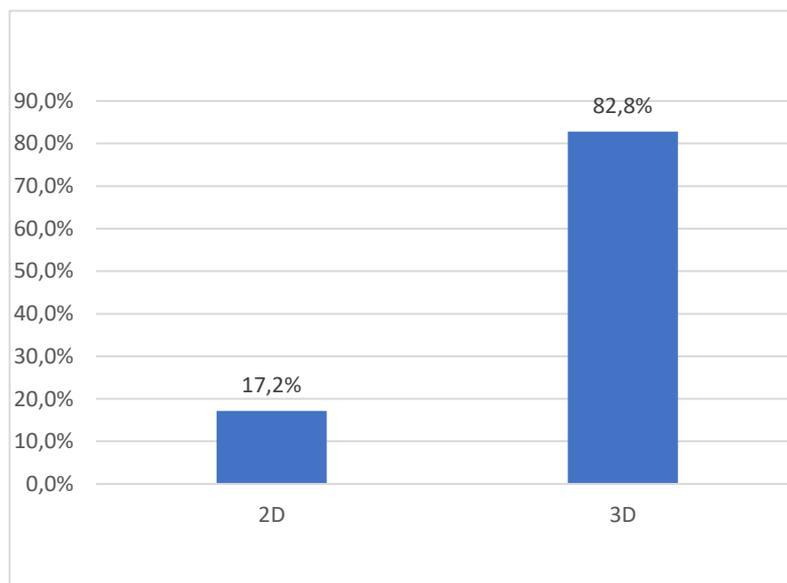


Gráfico 12. Distribución de la muestra según la pregunta 9. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

De acuerdo con la tabla y gráfico anteriores los usuarios de videojuegos tienen una marcada diferencia hacia el tipo de mundo 3D con un 82.8% de preferencia, por este motivo el videojuego se lo debe realizar en 3D con el fin de abarcar una mayor aceptación por parte del público objetivo.

Tabla 11. Pregunta 10. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 10. ¿En un videojuego, qué consideras es lo más importante?		
Opciones	Frecuencia Simple	Frecuencia Porcentual
Efectos de sonido	3	2.3%
Gráficos	72	56.3%
Mecánica de juego	34	26.6%
Temática	19	14.8%
Total	128	100%

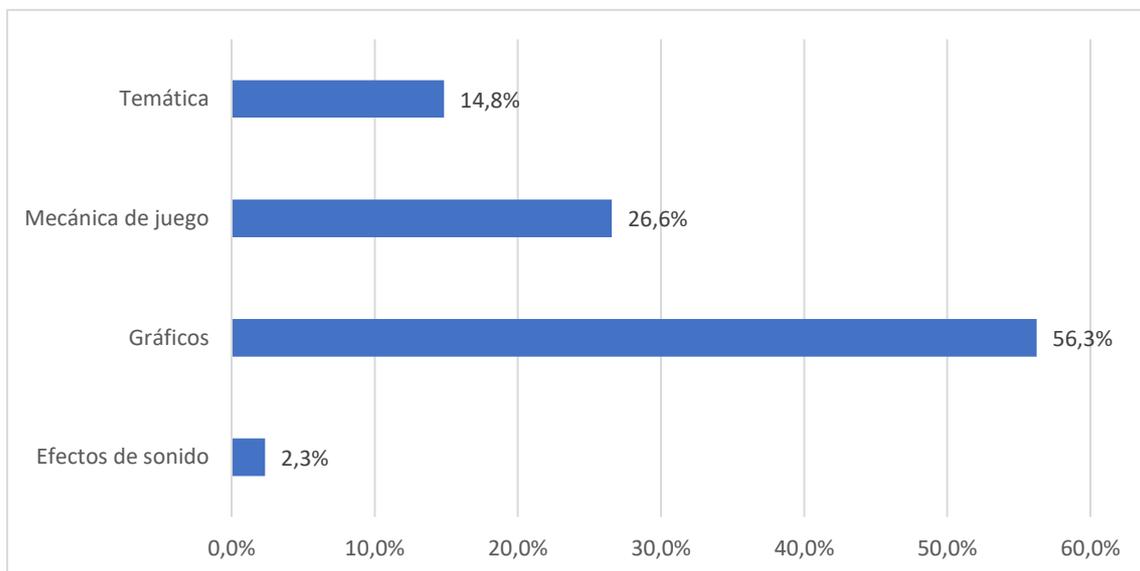


Gráfico 13. Distribución de la muestra según la pregunta 10. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

Como se observa en el gráfico y figura anterior, del total de 128 usuarios que hacen uso de videojuegos, el 56.3% considera que uno de los aspectos más importantes en un videojuego son los gráficos, por este motivo se pondrá énfasis en el área de desarrollo que conlleva el modelado de gráficos.

Tabla 12. Pregunta 12. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 12. ¿Te interesaría un videojuego en el que tengas que emitir sonidos para ganar?		
Opciones Pregunta 12	Frecuencia Simple	Frecuencia Porcentual
Sí	93	72.7%
No	35	27.3%
Total	128	100%

Tabla 13. Pregunta 13. Fuente: Base de datos de las encuestas aplicadas.
Elaborado por: Guamán (2020)

Pregunta 13. ¿Qué sonido preferirías emitir?		
Opciones Pregunta 13	Frecuencia Simple	Frecuencia Porcentual
Gritos	21	8.6%
Silbidos	64	68.8%
Soplos	8	22.6%
Total	93	100%

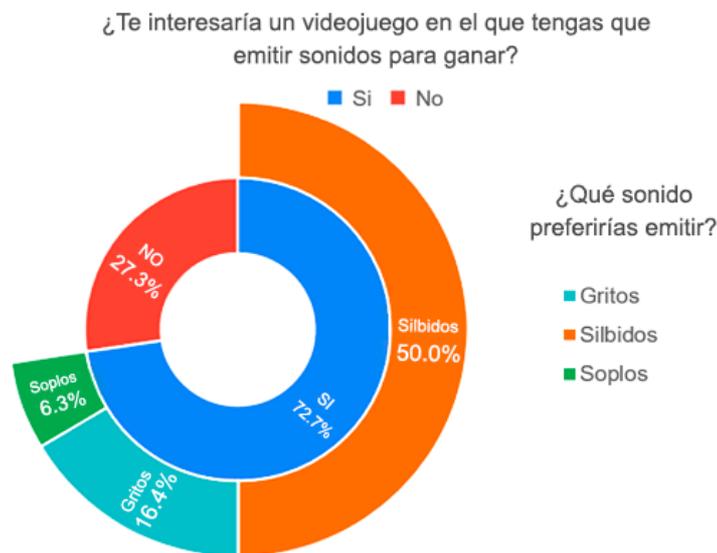


Gráfico 14. Distribución de la muestra según la pregunta 12 y 13. Fuente: Base de datos de las encuestas aplicadas. Elaborado por: Guamán (2020)

Como se puede observar en el gráfico y figura anterior, del total de 128 estudiantes que hacen uso de videojuegos, a un 72.7% le interesaría un videojuego en el que

se tenga que emitir sonidos para ganar, entre los sonidos propuestos para emitir la gran mayoría prefirió los silbidos, por este motivo para controlar el vehículo aéreo del juego, el usuario deberá emitir silbidos constantemente con el fin de poder alcanzar un mayor puntaje.

4.1.2 Recursos

Para el presente proyecto se planteó tres grupos de recursos: humanos, hardware y virtuales. Debido a que la planificación y el desarrollo lo llevó a cabo una sola persona, la asignación de estos recursos fue dirigida hacia el desarrollador, sin embargo, debido a que estos pueden cambiar a lo largo del desarrollo del proyecto, en esa sección se presentan las versiones finales.

En cuanto a recursos humanos el único medio disponible fue la persona encargada del proyecto que pasa a asumir todos los roles que propone la metodología a usar, en caso de que se lleguen a requerir más recursos humanos, se debe considerar aplazar los tiempos del cronograma para lograr cumplir con los objetivos planteados.

Tabla 14. Requerimientos recomendados de sistema. Elaborado por: Guamán (2020)

Recurso	Requerimiento recomendado
Sistema Operativo	Windows 10
Cpu	1,6 GHz o superior compatible con SSE2
Tarjeta gráfica	Soporte para DirectX 9 o 10
Memoria Ram	8GB
Espacio en el disco	Mayor a 40GB

El segundo grupo de recursos se compone por un computador cuyas características recomendadas se detallan en la Tabla 14, estos parámetros se fijan de acuerdo al análisis de los requerimientos de sistema planteados por cada

herramienta de software a usarse. Además de una tableta gráfica para facilitar el modelado y diseño del arte que compone el juego.

Tabla 15. Herramientas de software a usarse. Elaborado por: Guamán (2020)

Herramienta	Versión	Finalidad
Unity	2018.4.24f1 (64-bit)	Desarrollo del videojuego
Visual Studio Community	2017 v15.9.24	Entorno de desarrollo
Adobe Photoshop	CC 2019	Retoque elementos UI
Adobe Illustrator	CC 2019	Ilustración elementos UI
Linux MultiMedia Studio	1.2.2	Producción de sonido
Audacity	2.3.3	Edición de sonido
Blender	2.83.2	Modelado 3D

Finalmente, los recursos virtuales hacen referencia a aquellos medios intangibles necesarios para la ejecución y desarrollo del proyecto, en este apartado se aprecian dos grupos: herramientas de software (Tabla 15) y recursos multimedia. En ambos casos se prefirió usar aquellos de libre acceso o gratuitos.

Debido a que la materia principal del proyecto es el desarrollo de software, aquellos elementos artísticos sean modelos 3D, sonidos e imágenes, que requirieron de mayor esfuerzo, fueron adquiridos de otras fuentes o tiendas como el “Asset Store” de Unity o páginas de contenidos gráficos, con el fin de optimizar tiempo y esfuerzo.

Estos recursos deben estar al alcance del programador y su accesibilidad permite la correcta ejecución del proyecto, en caso de que un recurso no esté contemplado este debe solicitarse posteriormente con una estimación de costo y lapso de entrega.

4.1.3 El documento de diseño

Los resultados de la investigación mostrados previamente permitieron elaborar el documento del diseño, el primer y más valioso artefacto propuesto por la metodología Huddle, en la tabla inferior se muestra la plantilla final del documento de diseño.

Tabla 16. Plantilla final del documento de diseño. Elaborado por Guamán (2020)

CAMPO	DESCRIPCIÓN
CONCEPTO	
Título	Whistle or Die
Estudio/Diseñadores	Accel Leonardo Guamán CHamba
Género	Simulación, Carreras
Plataforma	Smartphone Android, PC windows
Versión	1
Sinopsis de Jugabilidad y Contenido	<p>El videojuego es del género de simulación, con la temática de una clásica carrera infinita, en la que aquel jugador que logre un mayor puntaje sobre los demás se posiciona en los primeros lugares. El juego inicia con un avión que está en constante caída libre, el objetivo es evitar que el avión choque ya sea con el suelo o con los obstáculos. Para evitar que esto suceda el usuario debe mantener siempre a flote al avión, para ello el juego usa el sensor del micrófono para detectar cuando el usuario está silbando, mientras así lo haga, el avión empezará a ascender. A medida que se avanza una mayor distancia, el avión descenderá más rápidamente, lo que hará que el usuario tenga menos tiempo para tomar aire, consecuentemente será más difícil no perder la partida.</p>

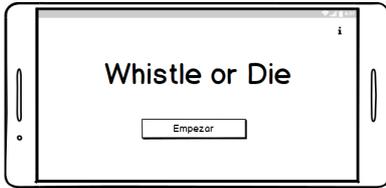
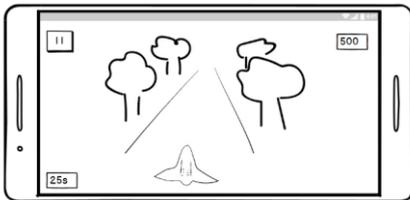
Categoría	Whistle or Die se asemeja a videojuegos como Temple Run o Subway Surfers, con la diferencia que este ocupa el micrófono como control para el personaje.
Licencia	GNU GLP
Mecánica	El jugador puede controlar la navegación horizontal del avión, es decir giros a la derecha o izquierda, mediante los comandos táctiles (smartphone) o mediante el teclado (PC), y la altura se la maneja al emitir silbidos, cuando el jugador silba, el avión asciende y cuando deja de hacerlo, desciende.
Tecnología	Smartphone con sistema operativo Android superior a 5.0 PC con Windows 8 o superior
Público	El juego está dirigido, pero no limitado a los estudiantes de la Universidad Iberoamericana del Ecuador

VISIÓN GENERAL DEL JUEGO

El videojuego tiene como fin proveer de entretenimiento a la comunidad universitaria, al tener un componente innovador como el control mediante silbidos, se espera que tenga gran aceptación por los jugadores.

MECÁNICA DEL JUEGO

Cámara	Perspectiva 3D
Periféricos	Sensor Táctil Teclado Micrófono
Controles	Flechas del teclado Laterales de la pantalla táctil Entrada de micrófono
Puntaje	El puntaje aumenta gradualmente al recorrer más distancia.

	El puntaje aumenta cuando el jugador choca con una recompensa (insignias flotantes).
Guardar/Cargar	El juego únicamente almacena el puntaje cuando el usuario pierde la partida. No se permite guardar partidas.
ESTADOS DEL JUEGO	
Existen tres estados de juego, el menú principal, la pantalla de juego y la pantalla de puntajes o Game Over. El flujo es lineal entre estas tres pantallas.	
INTERFACES	
Menú principal	
Descripción de la pantalla	Esta pantalla muestra el logotipo y permite al usuario ingresar su nombre y empezar un nuevo juego
Estados del Juego	Pantalla de juego
Imagen	
Pantalla de Juego	
Descripción de la pantalla	En esta pantalla se ejecuta el juego y el usuario debe empezar a silbar
Estados del Juego	Menú principal Pantalla de puntajes
Imagen	
Pantalla de Puntajes	
Descripción de la pantalla	Muestra un listado de los puntajes más altos obtenidos

Estados del Juego	Pantalla de juego
Imagen	
Niveles	
Título del Nivel	Nivel Principal
Encuentro	Todas las partidas
Descripción	Único nivel de juego disponible
Objetivos	Recorrer la mayor distancia posible
Progreso	El nivel termina cuando el jugador choca con el avión
Enemigos	Obstáculos
Música y Efectos de Sonido	Música y efectos globales del juego
Detalles de Producción	
Fecha de Inicio	02-04-2020
Fecha de Terminación	28-05-2020

4.1.4 Historias de Usuario

Los datos obtenidos por las encuestas aplicadas a los estudiantes de la universidad permitieron recabar los requerimientos de los mismos en cuanto a videojuegos, los cuales y de acuerdo a la metodología de desarrollo de software a usarse, se detallan en las historias de usuario (mostradas en la sección 4.2.1 Visionamiento y alcance), el propósito de estas es servir de guía tanto para el programador como para los interesados en el proyecto y para el cálculo de la estimación de costos y esfuerzo.

4.1.5 Estimación de tiempo

Tabla 17. Estimación de tiempo. Elaborado por Guamán (2020)

Historia de Usuario	Tiempo estimado
01 – Inicio de juego	1 semana
02 – Juego – contador	3 días
03 – Juego – avance	4 semanas
04 – Juego – generación de terreno	1 semana
05 – Juego – pérdida	1 semana
06 – Pantalla de fin de Fin de Partida	7 días
TOTAL	8 semanas

Para la estimación del tiempo se usó la técnica de Planning Poker, y con la ayuda de otros desarrolladores con conocimientos similares en el área de programación de videojuegos, se estimaron los tiempos mostrados en la Tabla 17, esto después de realizar las respectivas reuniones y discusiones acerca de la complejidad de cada historia de usuario.

4.1.6 Estimación de Costos/Esfuerzo

Para la estimación del costo y esfuerzo se usó el modelo de composición de aplicación de COCOMO II, este método toma en consideración el código reutilizado, la experiencia del desarrollador, la madurez de las herramientas de software a usarse y los factores dentro del desarrollo como el número de pantallas, informes, módulos y las líneas de código.

Tabla 18. Productividad de punto de aplicación. Fuente: (Sommerville, 2011, pág. 639)

Experiencia y habilidad del desarrollador	Muy bajo	Bajo	Nominal	Alto	Muy alto
Madurez y capacidad ICASE	Muy bajo	Bajo	Nominal	Alto	Muy alto
PROD (NAP/mes)	4	7	13	25	50

$$PM = (NAP \times (1 - \%reutilización / 100)) / PROD$$

Gráfico 15. Fórmula de estimación de esfuerzo por composición de aplicación.
Fuente: (Sommerville, 2011, pág. 639)

En la Figura anterior se puede visualizar la fórmula de estimación por composición de aplicación, en donde la variable NAP corresponde al número de puntos de aplicación, este valor se multiplica por el porcentaje de código que no va a ser reutilizado, finalmente se divide por el factor PROD o la productividad, cuyo valor obtenemos de la Tabla 18.

Tabla 19. Cálculo de puntos de aplicación. Elaborado por Guamán (2020)

Punto de aplicación	Número
Pantallas separadas	3
Informes	4
Módulos	2
Número de scripts	20
Bases de datos	1
TOTAL	30

La reutilización de código fue un factor importante pues el algoritmo usado para el reconocimiento de los silbidos del jugador se basó parcialmente en el método usado por (Guamán, Álvarez, Sánchez, & Andaluz, 2018) en el módulo de Formulación Lingüística del videojuego SLT-GAME, en cuya sección del videojuego los niños deben soplar para poder acumular el mayor puntaje posible, en base a esto el porcentaje de reutilización se estimó en un 15%.

Considerando que la experiencia y habilidad del desarrollador y la madurez de las herramientas de software tienen un valor Nominal y que el porcentaje de reutilización de software es del 15%, y con un total de 30 puntos de aplicación, obtenemos el siguiente resultado al reemplazar en la fórmula.

$$PM = \frac{30 \left(1 - \frac{15}{100}\right)}{13} = 1.96$$

El valor obtenido de 1.96 representa el esfuerzo estimado en meses-hombre del proyecto, esto quiere decir que la duración aproximada es de 2 meses, cuyo valor concuerda al cálculo total mediante las historias de usuario.

4.1.7 Product Backlog

Tabla 20. Versión final del Product Backlog. Elaborado por Guamán (2020)

ID	Historia	Estado	Duración	Sprint	Prioridad
01	Inicio de juego	Ejecutado	1 semana	1	Alta
02	Juego – contador	Ejecutado	3 días	1	Media
03	Juego – avance	Ejecutado	4 semanas	1	Alta
04	Juego – generación de terreno	Ejecutado	1 semana	1	Alta
05	Juego – pérdida	Ejecutado	1 semana	1	Baja
06	Pantalla de fin de Fin de Partida	Ejecutado	7 días	1	Alta

4.1.8 Diagrama de Gantt

Un diagrama de Gantt consiste en la representación en forma de barras de los tiempos que tomarán realizar las actividades en un proyecto, detalla también al responsable de dichas actividades y debe mostrar la dependencia entre las tareas programadas (Sommerville, 2011).

Task ID	Task Name	Duration	Start	Finish	Predecessors
1	Diseño del entorno	6 days	Thu 2/4/20	Thu 9/4/20	
2	Diseño de elementos 3D	6 days	Fri 10/4/20	Fri 17/4/20	1
3	Programación de la recursividad del entorno	6 days	Mon 20/4/20	Mon 27/4/20	2
4	Implementación del algoritmo de reconocimiento de silbidos	6 days	Tue 28/4/20	Tue 5/5/20	3
5	Implementación del control del jugador	6 days	Thu 7/5/20	Thu 14/5/20	4
6	Módulo de puntajes	3 days	Fri 15/5/20	Tue 19/5/20	5
7	Módulo de puntajes más altos	3 days	Wed 20/5/20	Fri 22/5/20	6
8	Mejorar El manejo del jugador	3 days	Mon 25/5/20	Wed 27/5/20	7
9	Corrección de errores	4 days	Thu 28/5/20	Tue 2/6/20	8

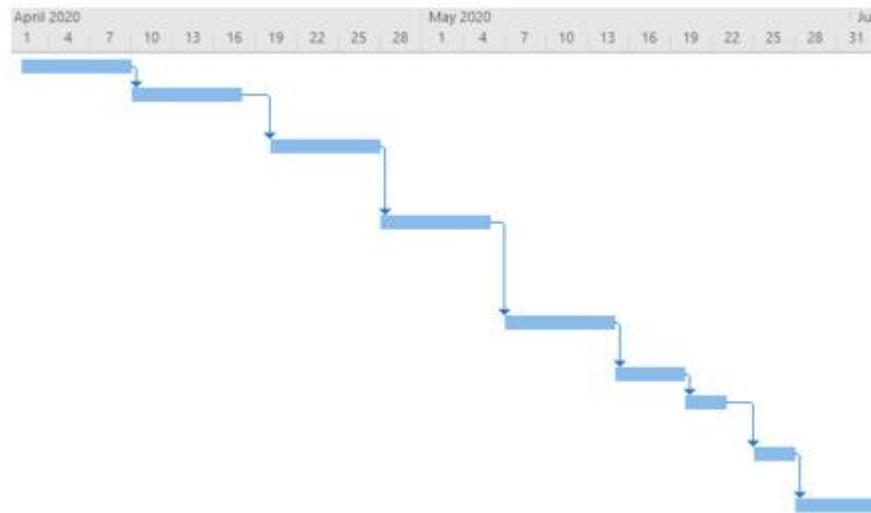


Gráfico 16. Diagrama de Gantt del proyecto. Elaborado por: Guamán (2020)

4.1.9 Herramientas de gestión de proyecto

Como herramienta para la gestión del proyecto se utilizó la herramienta online Jira, en su versión gratuita permite llevar el control de las actividades, así como el seguimiento a errores e incidencias, gracias a sus funciones de control de flujo de trabajo, también fue útil en la gestión de procesos durante toda la etapa del desarrollo del videojuego.

4.1.10 Sistema de control de versionado utilizado

Como sistema de control de versiones se usó Git, debido a su licencia GNU que permite su libre uso, es el software de forja más conveniente, a pesar de que existe un único colaborador para el proyecto, varias funcionalidades como el uso de ramas, resultaron ser muy útiles en el momento del desarrollo.

Además, y en combinación con Git, se usó la plataforma GitHub en donde el código se encuentra alojado, así como todas sus versiones, aquí se pueden hacer nuevos cambios, mejoras o correcciones al videojuego, previa aprobación del creador del proyecto. El criterio para la elección de esta herramienta se basó en el acceso, pues a pesar de ser una plataforma de pago, ofrece muchas facilidades para la creación de repositorios gratuitos.

4.2 Análisis y Diseño

4.2.1 Visionamiento y alcance

En este apartado se pretende mostrar un resumen pragmático del juego, entre otras cosas se mostrarán aquellas funcionalidades que tendrá la aplicación, así como sus limitaciones, es decir los aspectos que no van a estar presentes en el juego, plataformas en las que estará y no estará disponible y público objetivo para el que estará accesible el software.

Tabla 21. Glosario de términos. Elaborado por Guamán (2020)

Término	Definición	Alias
Partida de juego	Lapso desde que el usuario empieza un nuevo juego hasta que lo termina, ya sea por pérdida o retirada	Partida
Entorno de juego	Conjunto de elementos y modelos 3D que conforman el ambiente tales como árboles, terreno, casas, césped, lluvia, etc.	Entorno
Personaje	Ser ficticio que es controlado por el jugador.	PJ
Sesión	Transcurso entre que el usuario abre y cierra el juego.	Sesión

El juego está dirigido a los estudiantes de la Universidad Iberoamericana del Ecuador, como se menciona en el Capítulo 1 el videojuego busca ofrecer a los estudiantes una alternativa de entretenimiento que les permita distraerse en sus tiempos libres a través del uso de la aplicación ya sea mediante un smartphone o una PC y de esa forma reducir sus niveles de ansiedad o estrés.

El producto se denominó Whistle or Die, cuya traducción al español es: silbar o morir, puesto que ese es el objetivo del juego, este nombre hace referencia al modo de juego en el que el usuario deberá silbar para poder mantener su personaje a flote y en caso de que lo deje de hacer, este morirá. El nombre es corto y fácil de recordar y busca llamar la atención de más usuarios.

De acuerdo a las respuestas obtenidas mediante las encuestas aplicadas, se obtuvo la información necesaria para la definición de requisitos mostrados en las Tabla 22 a la Tabla 27, además en base a esto se decidió que las plataformas para las que estará disponible el videojuego serán PC con sistema operativo Windows y Smartphone con sistema operativo Android, sin embargo, para futuros trabajos, queda abierta la posibilidad de exportar el proyecto a otras plataformas como iOS o Web.

Tabla 22. Historia de usuario 01. Elaborado por Guamán (2020)

Historia de usuario	
Número: 01	Nombre: Inicio de juego
Prioridad: Baja	Iteración: 1
Descripción: Se muestra una pantalla inicial con la información del juego, y se dispone de un botón que permite iniciar una nueva partida	
Observaciones:	

Tabla 23. Historia de usuario 02. Elaborado por Guamán (2020)

Historia de usuario	
Número: 02	Nombre: Juego - contador
Prioridad: Media	Iteración: 1
Descripción: Se muestra el entorno del videojuego con un contador regresivo que informa al usuario sobre el inicio de una nueva partida.	
Observaciones: El contador empieza en 3 y termina en 1, acompañado de un efecto de sonido acorde a la temática.	

Tabla 24. Historia de usuario 03. Elaborado por Guamán (2020)

Historia de usuario	
Número: 03	Nombre: Juego - avance
Prioridad: Alta	Iteración: 1
Descripción: El personaje empieza a avanzar hacia el eje z y en caída libre. Cuando el jugador silbe, el personaje empieza a subir lentamente, y si lo deja de hacer, nuevamente empieza a caer.	
Observaciones: El jugador necesariamente debe estar en un lugar con poco ruido.	

Tabla 25. Historia de usuario 04. Elaborado por Guamán (2020)

Historia de usuario	
Número: 04	Nombre: Juego – generación de terreno
Prioridad: Alta	Iteración: 1
Descripción: A medida que el personaje siga avanzando, el terreno debe seguir generándose.	
Observaciones:	

Tabla 26. Historia de usuario 05. Elaborado por Guamán (2020)

Historia de usuario	
Número: 05	Nombre: Juego – pérdida
Prioridad: Alta	Iteración: 1
Descripción: Cuando el jugador no pueda soplar más, y el personaje haya descendido lo suficiente como para topar el terreno del juego, la partida habrá llegado a su fin.	
Observaciones:	

Tabla 27. Historia de usuario 06. Elaborado por Guamán (2020)

Historia de usuario	
Número: 06	Nombre: Pantalla de Fin de Partida
Prioridad: Alta	Iteración: 1
Descripción: Una vez que el usuario haya topado el terreno o un obstáculo con su personaje, se mostrará una pantalla con el texto de “Fin del juego” y adicional se mostrará la información sobre los puntajes obtenidos.	
Observaciones:	

4.2.2 Modelo de procesos

Un modelo de procesos se define como una representación gráfica de actividades que se ejecutan mediante una secuencia definida a través de los diferentes roles existentes, este modelo se usa para explicar un proceso de software (Sommerville, 2011), para la representación del modelo de procesos del presente proyecto se usó una notación BPMN (Notación de modelado de procesos de negocios).

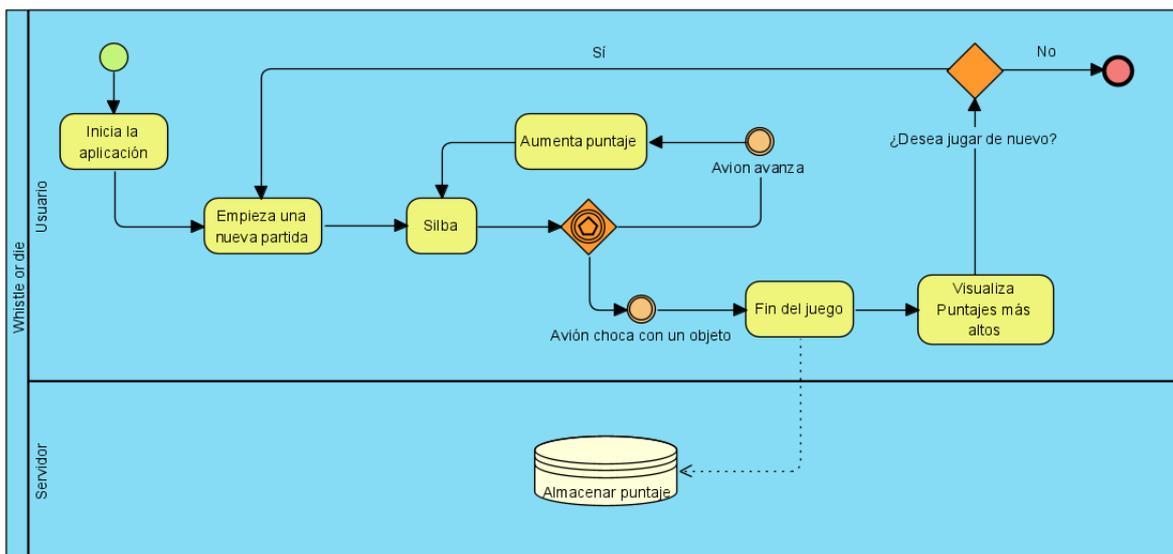


Gráfico 17. Modelo de procesos del videojuego. Elaborado por (Guamán 2020)

Como se muestra en el gráfico anterior, el flujo inicia cuando el usuario abre la aplicación, consiguientemente cuando el usuario empieza a jugar entra en un bucle que termina cuando el jugador pierde la partida o hace colisionar al avión con algún objeto o el suelo, en ese caso se visualiza el listado con los mejores puntajes a la vez que se almacena la puntuación obtenida en esa partida, el ciclo continúa dando al usuario la posibilidad de volver a iniciar un nuevo juego, en caso de ser así se repite el flujo mencionado anteriormente.

4.2.2 Diagrama de clases

Un diagrama de clases es un modelo estructural que describe las clases de un sistema y sus respectivas asociaciones (Sommerville, 2011) para el presente

proyecto se elaboró el diagrama de clases que, debido a la arquitectura basada en Unity, estas pueden ser objetos de juego o Scripts que manejan componentes.

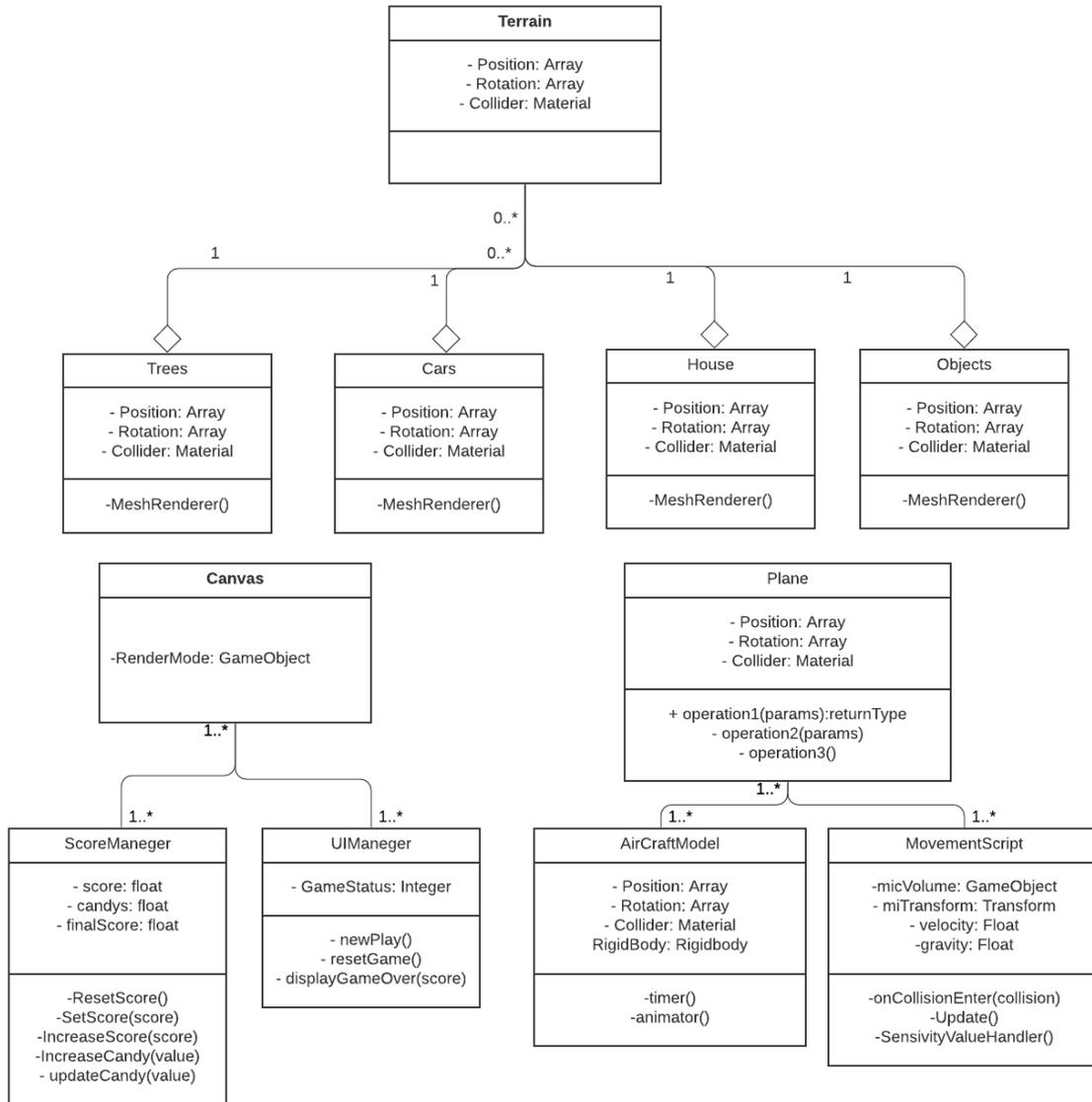


Gráfico 18. Diagrama de clases del juego. Elaborado por Guamán (2020)

Como se muestra en el gráfico anterior, el videojuego está estructurado por tres componentes principales: el Canvas o elemento encargado de renderizar textos y botones, el terreno cuyos elementos conforman el entorno virtual donde se desarrolla el videojuego y el jugador o avión que es controlado por el usuario.

4.2.3 Diseño de Base de datos

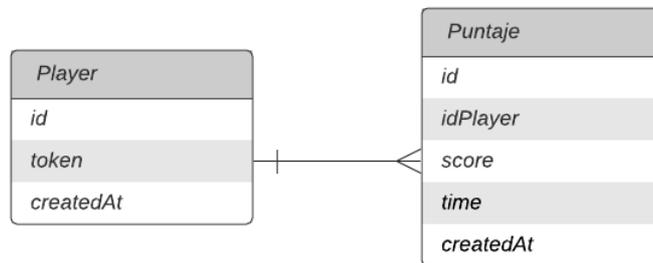


Gráfico 19. Modelo de base de datos. Elaborado por Guamán (2020)

El juego almacena los puntajes de cada jugador y puesto que los datos a almacenarse son sumamente sencillos y muy pocos, por ello el modelo de la base de datos que se presenta en el Gráfico 19 cuenta únicamente con dos entidades.

4.2.4 Diseño de interfaz de usuario

La interface de usuario en el ámbito de desarrollo de videojuegos se puede contemplar desde el punto de vista del jugador final donde se muestra información útil sobre el estado del juego, y desde el punto de vista del personaje, el cual abarca todo el entorno donde se lleva a cabo la jugabilidad.

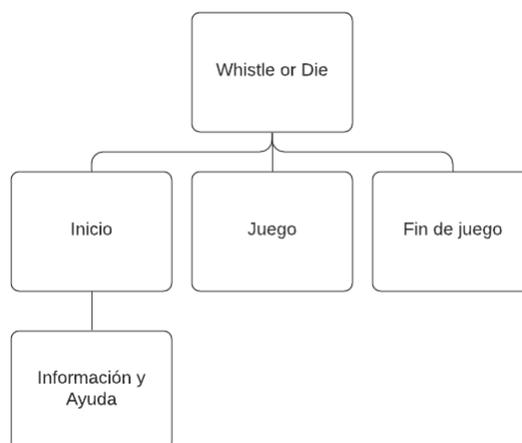


Gráfico 20. Diagrama de navegación del software. Elaborado por (Guamán 2020)

En el documento de diseño mostrado en la Tabla 16 se muestran las pantallas presentes en el videojuego, la navegación entre las mismas se detalla en la figura anterior, y como se puede observar la navegación es lineal y de fácil interacción con el usuario.

El diseño de la interface de usuario se lo realizó con la ayuda de una herramienta de maquetación originalmente orientada para diseños web, sin embargo, no se limita a este ámbito, y puede ser aplicada en muchos casos, los widgets ofrecidos por Balsamiq son de gran ayuda para definir la información que será mostrada y da un breve vistazo a la funcionalidad del juego en general.

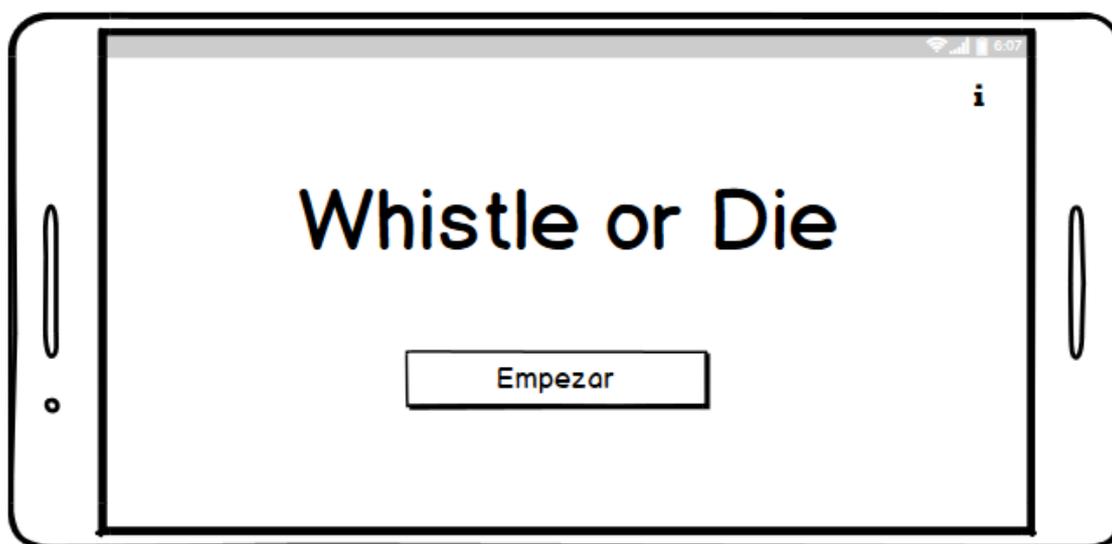


Gráfico 21. Pantalla principal del juego. Elaborado por Guamán (2020)

En el gráfico anterior se detalla el diseño la pantalla principal, la cual se mostrará con una animación al iniciar el juego, aquí se muestra únicamente la opción de iniciar el juego y un botón en la parte superior derecha que ofrece al usuario la información acerca del juego y una pequeña guía de uso y ayuda.

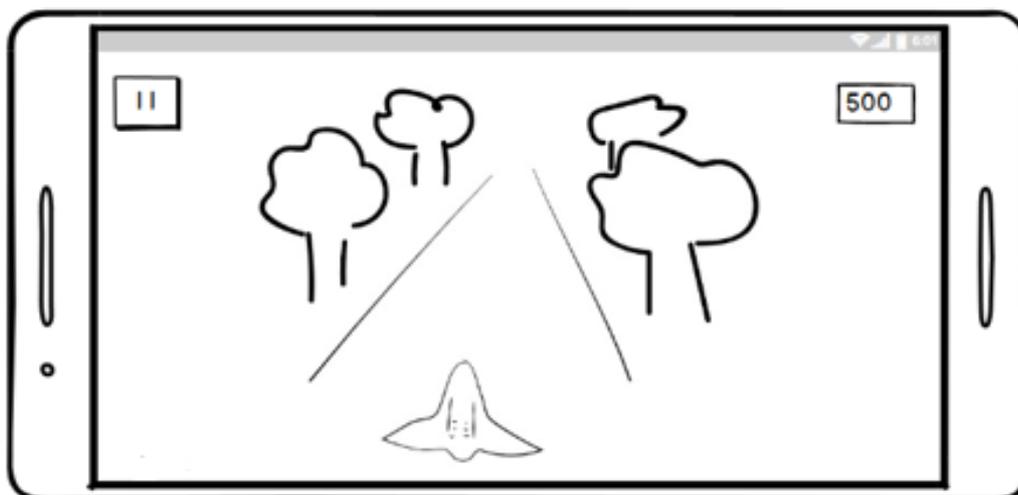


Gráfico 22. Interface de pantalla de juego. Elaborado por Guamán (2020)

En el gráfico previo se muestra el diseño de la pantalla principal de juego, como se puede observar, los elementos son netamente informativos los cuales detallan el puntaje actual y el tiempo de juego, en la parte superior izquierda se muestra un botón que permite pausar el juego, manteniendo los puntajes obtenidos.



Gráfico 23. Interfaz de la pantalla Fin de Juego. Elaborado por Guamán (2020)

Una vez que el jugador haya perdido la partida, se mostrará la interface detallada en el Gráfico 23, esta pantalla muestra información al usuario sobre el puntaje que obtuvo y las mejores puntuaciones globales, además un botón en la parte inferior le permite empezar una nueva partida.

4.2.5 Diseño de arquitectura

El diseño de arquitectura es un proceso en el que se diagrama la estructura de la aplicación con el fin de definir cómo se van a cumplir los requerimientos funcionales y no funcionales (Sommerville, 2011). Para el presente proyecto se diseñó una arquitectura en capas para el videojuego y el API, adicionalmente se muestra la integración de estas mediante la arquitectura cliente-servidor.

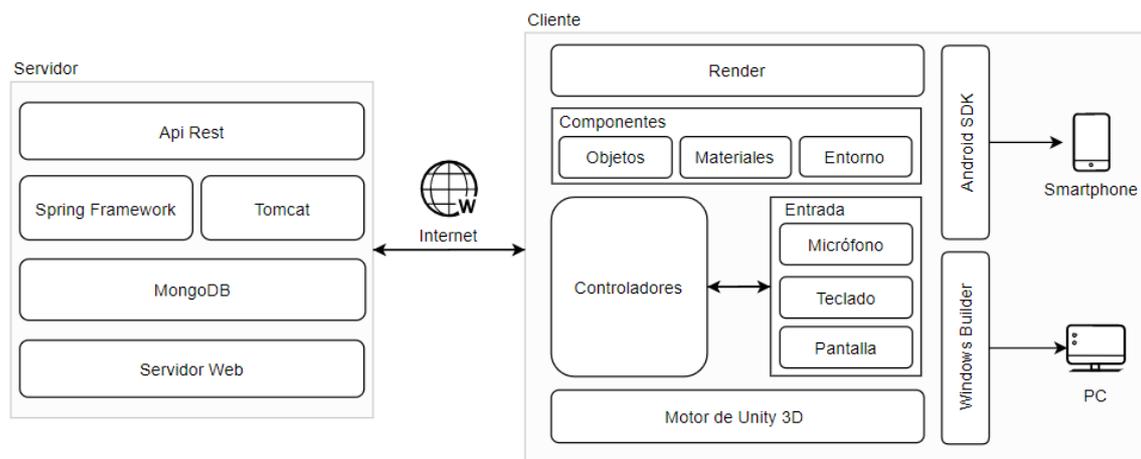


Gráfico 24. Arquitectura del videojuego. Elaborado por Guamán (2020)

Como se muestra en la figura anterior, el proyecto tiene una arquitectura cliente-servidor, sin embargo, para cada uno de estos elementos (cliente y servidor) se propuso una arquitectura en capas, basadas en las tecnologías a usarse (Unity y Spring), las cuales se integran mediante una conexión a internet.

La interacción entre el cliente y servidor se da a través de solicitudes GET/POST realizadas por el videojuego cada vez que se finaliza una partida, mediante estas solicitudes se actualizan los puntajes en el servidor y se obtienen las puntuaciones más altas para ser mostrados en la pantalla de fin de juego.

4.3 Desarrollo

4.3.1 Tecnologías utilizadas

Para el desarrollo del presente proyecto se usaron de preferencia herramientas y tecnologías libres, podemos identificar 2 servicios en la presente arquitectura de la

aplicación: el juego y la interfaz de programación de aplicaciones API, las tecnologías usadas en cada caso se detallan a continuación.

Motor de videojuego: el motor para el videojuego elegido es Unity en su versión 2018, los scripts fueron codificados en lenguaje C#, algunas de sus herramientas internas también fueron usadas, como el Wizard de compilación y construcción para Android y Windows, los cuales permiten generar de manera sencilla un instalador de formato .apk y .exe, respectivamente para cada plataforma.

Para el backend, se usó el framework Spring 1.5.9 con un gestor de paquetes Maven 3.6.3 en el lenguaje java en su versión 1.8, este proyecto se integra con una base de datos MongoDB con un clúster 4.2.8 compartido y 512 MB de ram, lo cual se estima que es suficiente para almacenar los puntajes de los usuarios.

4.3.2 Producto de software desarrollado



Gráfico 25. Pantalla de inicio del videojuego. Fuente: Videojuego Whistle or Die. Elaborado por Guamán (2020)



Gráfico 26. Pantalla de juego. Fuente: Videojuego Whistle or Die. Elaborado por Guamán (2020)



Gráfico 27. Pantalla de Fin de juego. Fuente: Videojuego Whistle or Die. Elaborado por Guamán (2020)

Tras la asignación de las tareas detalladas en el ProductBacklog (Tabla 20), se llevó a cabo la fase de producción, detallada en la guía de Huddle como la etapa en la que se programa el videojuego, y posterior a las revisiones diarias y el refinamiento del Backlog, se obtuvo como resultado los diferentes módulos del videojuego mostrados en los gráficos Gráfico 25, Gráfico 26Gráfico 27.

4.3.3 Acceso al producto software

El proyecto está alojado en la plataforma GitHub, en donde se puede acceder tanto al código como a los respectivos lanzamientos de la app mediante el link: <https://github.com/SoftwareUnibe/WHISTLE-OR-DIE-ACCEL-GUAMAN>

4.4 Pruebas

4.4.1 Técnicas de pruebas

Para el proyecto en Unity se usaron pruebas de integración, debido a que la herramienta que proporciona Unity, se encarga de realizar las pruebas unitarias de manera automática y finalmente las de integración, pues estas ejecutan en modo de juego todos los scripts de manera secuencial.

4.4.2 Herramienta de prueba utilizada

Para la ejecución de las pruebas de integración se usó Unity Test Framework, el marco de pruebas por defecto de Unity, cuyo paquete es una integración de un conjunto de bibliotecas de pruebas orientadas principalmente para lenguajes .NET. (Unity, 2020)

Unity permite ejecutar pruebas en modo juego, estas ejecutan los scripts del juego como una instancia Coroutine, es decir sus funciones y propiedades no quedan expuestas, esto permite realizar un debug más rápido y cómodo de los scripts.

4.5 Resultados obtenidos

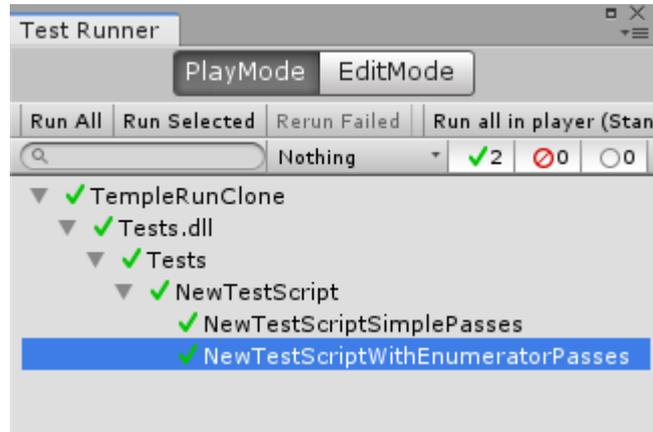


Gráfico 28. Prueba en modo de juego aprobada. Elaborado por Guamán (2020)

En la figura previa se muestran los resultados obtenidos del primer test de integración, este test se lo realizó en modo juego y se evaluaron los scripts desarrollados con una respuesta favorable, pues todos aprobaron la evaluación lo que quiere decir que el juego funciona correctamente.

4.6 Manual de usuarios de la aplicación

El desarrollo del proyecto según la metodología Huddle finaliza con el cierre del mismo y la entrega de la documentación final que incluye el manual de usuario, en este se ofrece al jugador una guía de cómo usar el videojuego, la funcionalidad de cada botón y una explicación de cada elemento que se visualiza en la pantalla, ver Anexo 2.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

Posteriormente a haber llevado a cabo el proceso de investigación y haber ejecutado con éxito el desarrollo del videojuego, se presentan a continuación las conclusiones obtenidas y una serie de recomendaciones dirigidas a aquellas personas o entidades interesadas en el ámbito del desarrollo de software de entretenimiento.

5.1 Conclusiones

El objetivo principal del presente trabajo de titulación fue desarrollar un videojuego que integre las características requeridas por los estudiantes de la Universidad Iberoamericana del Ecuador con el fin de proveerles un medio de entretenimiento.

Teniendo en consideración lo anterior mencionado, se plantearon las siguientes conclusiones:

- El objetivo número uno que refiere a la especificación de requerimientos, se logró mediante la aplicación de una encuesta que, en primera instancia evidenció que un videojuego sería de gran utilidad para la comunidad universitaria, posteriormente demostró que el videojuego debería ser del género de simulación y carreras con temática de guerras y tener características como: perspectiva de cámara en 3D, y un control del personaje mediante comandos de voz y la plataforma en donde estará disponible: Pc y Smartphone.
- Con respecto al segundo objetivo de la investigación, se elaboró satisfactoriamente el diseño del videojuego. Para el cumplimiento del mismo se definió el diagrama de clases, el diseño de base de datos, la interfaz de usuario y finalmente el diagrama de la arquitectura general en donde se detalla la relación entre el videojuego y el acceso a los datos.
- En relación al tercer objetivo se desarrolló el videojuego con la codificación de los scripts en Unity y la integración con los modelos, imágenes y demás elementos multimedia, además la interfaz de programación de aplicaciones

(API) permitió centralizar los puntajes de todos los jugadores, para ello se usaron las tecnologías mencionadas en el capítulo 4.

- El desarrollo del presente proyecto concluyó con la correcta ejecución del cuarto objetivo el cual corresponde a la fase de pruebas, aquí se ejecutaron las pruebas unitarias y de integración de todos los componentes necesarios para el correcto funcionamiento del videojuego, dichas pruebas arrojaron resultados satisfactorios, probando que el proyecto desarrollado cumple con los estándares de calidad necesarios para su salida a producción.

5.2 Recomendaciones

Posterior al desarrollo del videojuego se evidenciaron algunos aspectos que podrían mejorar la calidad de vida de los lectores, a continuación, algunas recomendaciones:

Es pertinente que la Universidad Iberoamericana del Ecuador diversifique las áreas de aplicación de la ingeniería de software, por medio de charlas o cursos gratuitos sobre otros terrenos de programación como videojuegos, aplicaciones móviles, sistemas embebidos, entre otros. Con la finalidad de que los estudiantes puedan especializarse en una rama de su interés desde una etapa temprana de formación y no limitarlo únicamente a tecnologías web o de escritorio.

Los estudiantes de la Universidad Iberoamericana del Ecuador deben estar al tanto de los últimos avances tecnológicos, esto les permitirá siempre tener acceso a herramientas que pueden facilitar sus actividades diarias, o recursos como videojuegos gratuitos de los cuales puedan obtener alguna ventaja, para ello se recomienda realizar una indagación de temáticas sobre su interés, información a la cual pueden acceder de forma gratuita a través de páginas referentes o aplicaciones sobre noticias o investigación y desarrollo.

A la sociedad en general se recomienda hacer uso de videojuegos, ya que estos ofrecen numerosas ventajas, siempre y cuando se los use de manera moderada y sin excesos, para acceder a ellos, pueden visitar periódicamente tiendas de videojuegos como Google Play o Apple Store.

GLOSARIO

API: “Una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. API significa interfaz de programación de aplicaciones” (Red Hat, 2020).

Arquitectura de Software: “Modelo de la estructura y organización fundamentales de un sistema de software” (Sommerville, 2011, pág. 734).

Gestión de versiones: “Proceso de gestionar los cambios a un sistema de software y sus componentes, de modo que sea posible conocer cuáles cambios se implementaron en cada versión del componente/sistema, y también para recuperar y recrear versiones anteriores” (Sommerville, 2011, pág. 738)

Huddle: “Es una metodología apropiada para el proceso de desarrollo de videojuegos; describe los elementos esenciales que se deben llevar a cabo para la implementación de un videojuego” (Hernández, Pérez, & Correa, 2017, pág. 18).

Interfaz: “Especificación de los atributos y las operaciones asociados con un componente de software. La interfaz se usa como medio para acceder a la funcionalidad del componente” (Sommerville, 2011, pág. 739).

Jugabilidad: “Es el grado y la naturaleza de la interactividad que el juego incluye, es decir, cómo el jugador es capaz de interactuar con el mundo virtual” (Hernández, Pérez, & Correa, 2017, pág. 16).

Mecánica de juego: “Representan un componente esencial para la especificación detallada a diferentes niveles de los objetivos, entradas, caminos y salidas, que se pueden originar entre la interacción videojuego–jugador” (Hernández, Pérez, & Correa, 2017, pág. 16).

Metodología de desarrollo de Software: “Es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo” (Hernández, Pérez, & Correa, 2017, pág. 17).

Modelo de proceso: “Representación abstracta de un proceso. Los modelos de proceso pueden desarrollarse desde varias perspectivas y muestran las actividades implicadas en un proceso” (Sommerville, 2011, pág. 742).

Motor de juego: “Un sistema de software diseñado para la creación y desarrollo de videojuegos” (Hernández, Pérez, & Correa, 2017, pág. 14).

Plataforma de software: “El entorno donde se ejecutará el software” (Sommerville, 2011, pág. 39)

Requerimiento de software: “Son descripciones de lo que el sistema debe hacer: el servicio que ofrece y las restricciones en su operación” (Sommerville, 2011, pág. 83).

Scrum: “Es una metodología ágil que sirve para administrar y controlar el desarrollo de un producto. Se caracteriza por cinco valores: compromiso, enfoque, sinceridad, respeto y coraje, que son implementados mediante varias prácticas” (Hernández, Pérez, & Correa, 2017, pág. 17)

BIBLIOGRAFÍA

Bibliografía Impresa

- Barandiaran, X. E. (2020). Cómo y por qué usar licencias Copyleft.
- Belli, & Raventós. (2008). Breve historia de los videojuegos. *Revista de pensamiento e investigación social*, 159-179.
- Belli, S. &. (2008). Breve historia de los videojuegos. *Revista de Pensamiento e Investigación Social*, 159-179.
- Brocca, J. C., & Casamiquela, R. (2005). LAS LICENCIAS DE SOFTWARE DESDE LA PERSPECTIVA DEL USUARIO FINAL. *Pilquen*, 1-10.
- Canós, J. H., Carmen, P., & Patricio, L. (2012). *Métodologías Ágiles en el Desarrollo de Software*.
- Carbonell, J. C. (2015). *ESTÉTICA Y BARRERAS TÉCNICAS EN LOS VIDEOJUEGOS*. Valencia.
- Crawford, C. C. (2003). *Game Design*. New Riders.
- Escobar, I., Guamán, A., Montaluísa, J., Pruna, E., & Marín, Y. (2017). Mobile application for vowel learning in children with Down Syndrome "LVDS-App". *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, 1-5.
- Felicia, P. (2009). *Videojuegos en el aula Manual para docentes*. Bélgica: European Schoolnet.
- GaritaHernández, F. A., LizanoMadriz, F., & Cordero Esquivel, C. M. (2019). Metodologías para el Desarrollo de Videojuegos Serios . *Tecnología Educativa Revista CONAIC*.
- Gregory, J. (2014). *Game Engine Architecture*. New York: A K Peters/CRC Press.
- Guamán, A., Álvarez, M., Sánchez, J., & Andaluz, V. (2018). SLT-Game: Support System for Therapies of Children with Communication Disorders. *Lecture Notes in Computer Science*, 165-175.
- Guanoluisa, G. M., Pilatasig, J., Flores, L. A., & Andaluz, V. H. (2019). GY MEDIC v2: Quantification of Facial Asymmetry in Patients with Automated Bell's Palsy by AI. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics*, 351-361.

- Hernández, A., Pérez, K., & Correa, O. (2017). Marco de trabajo ingenieril para el proceso de desarrollo de videojuegos. *Revista Antioqueña de las Ciencias Computacionales*, 13-26.
- Hernández, R., Fernández, C., & Baptista, M. (2010). *Metodología de la Investigación*. México D.F.: Mcgraw-HILL / Interamericana Editores, S.A. de C.V.
- Huizinga, J. (1972). *Homo Ludens*. Madrid: Alianza.
- Iglesias, A. A. (2011). *Desarrollo de Videojuegos*. Buenos Aires.
- Langlotz, A. C. (2008). Video games industry overview: An analysis of the current market and future growth trends. En *International Business Project* (págs. 4-47).
- León Mantilla, H. M., & Sánchez Chacaguasay, I. D. (2018). *DESARROLLO DE UN VIDEOJUEGO EDUCATIVO PARA EL*. Latacunga.
- Loiza, S. C. (s.f.). ESTIMACION PARA PROYECTOS DE SOFTWARE (TIPOS, MODELO, TECNICAS) Y MODELO COCOMO.
- Lorenzo, R. (2006). Contribución sobre los paradigmas de investigación. *Educação. Revista do Centro de Educação*, 11-22.
- Mantilla, H., & Sánchez, I. (2018). *DESARROLLO DE UN VIDEOJUEGO EDUCATIVO PARA EL MEJORAMIENTO DE LA ENSEÑANZA Y APRENDIZAJE DEL LENGUAJE DE SEÑAS EN LOS ALUMNOS DE PRIMER Y SEGUNDO GRADO DEL INSTITUTO DE EDUCACIÓN ESPECIAL CAMILOGALLEGOS PARA PERSONAS CON DISCAPACIDAD AUDITIVA*. Latacunga: DEPARTAMENTO DE ELÉCTRICA Y ELECTRONICA.
- Mendez, G., Obviedo, E., Fallas, G., Vega, C., & Méndez, A. (2018). *Análisis de las herramientas Unity y Blender para el desarrollo de videojuegos con un enfoque educativo*. San Carlos.
- Morales Urrutia, G. A., Nava López, C. E., Fernández Martínez, L. F., & Rey Corral, M. A. (2017). PROCESOS DE DESARROLLO PARA VIDEOJUEGOS. *Culcyt*.
- Morant, J. (2020). *Desarrollo de un videojugo en Unity con generación de escenario procedural*. Catalunya : Universitat Oberta de Catalunya (UOC).

- Navarro Cadavid, A. D., Fernández Martínez, J., & Morales Vélez, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. *PROSPECTIVA*, 30-39.
- Nieto, D. L. (2006). ANÁLISIS DEL CONTEXTO HISTÓRICO Y TECNOLÓGICO DEL ORIGEN DE LOS VIDEOJUEGOS. *ICONO 14*.
- Ouazzani, I. (2012). *Manual de creación de videojuego con Unity 3D*. Madrid.
- Peña, J. M. (2011). *Pruebas de Integración*.
- Pow Sang Portillo, J. A. (2015). *Estudio de Técnicas basadas en Puntos de Funcion para la Estimación del Esfuerzo en Proyectos Software*. Perú.
- Schwaber, K., & Sutherland, J. (2017). La Guía de Scrum. *La Guía Definitiva de Scrum*.
- Smisek, J., Jancosek, M., & Pajdla, T. (2013). 3D with Kinect. En *Consumer Depth Cameras for Computer Vision* (págs. 3-25). Springer.
- Tello Leal, E., Sosa R., C., & Tello Leal, D. (2012). REVISIÓN DE LOS SISTEMAS DE CONTROL DE VERSIONES UTILIZADOS.
- Tinoco Gómez, O., Rosales, P., & Salas, J. (2010). Criterios de selección de metodologías de desarrollo de software. *Industrial Data*.
- Trenta, M. (2012). Orígenes del videojuego: conexiones históricas y sociales de un producto cultural. *IV Congreso Internacional Latina de Comunicación Social*.
- Trigas Gallego, M. (2012). Metodología Scrum. *Gestión de Proyectos Informáticos*.
- Ulrich, T. (2006). *Casual games and gender*. Munich .
- Vaca, A. (2015). *ESTUDIO COMPARATIVO DE LOS MOTORES GRÁFICOS WebGL 'THREE.JS' Y 'BABYLON.JS' PARA EL DESARROLLO DE UN JUEGO EDUCATIVO*. Sangolquí: Departamento de ciencias de la computación.
- Villa Betancur, A., & Giraldo Plaza, J. E. (2012). Automatización de pruebas unitarias de códigos. *Scientia et technica*, 147-151.
- Wolf, M. J. (2008). *The Video Game Explosion*. ABC-CLIO.

Bibliografía digital

- Audacity*. (2019). Obtenido de <https://www.audacityteam.org/>
- Burbano, A. R. (07 de 06 de 2015). *El estrés incide en el rendimiento académico*. Obtenido de eltelegrafo: www.eltelegrafo.com.ec/

Carretero, A. (2014). *CreativosOnline*. Obtenido de <https://www.creativosonline.org/>

Chacon, S. (2020). *GIT*. Obtenido de <https://git-scm.com/>

Dixon, J. (08 de Octubre de 2017). *Computer Nostalgia*. Obtenido de <http://www.computernostalgia.net/articles/HistoryofComputerGames.htm>

García, A. (16 de 08 de 2015). *códigofacilito*. Obtenido de <https://codigofacilito.com/articulos/que-es-git>

Hernández, & Del Valle, L. (2018). *Programar Fácil*. Obtenido de <https://programarfacil.com/podcast/9-desarrollar-aplicaciones-con-visual-studio-community/>

IONOS. (2020). *Digital Guide IONOS*. Obtenido de <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/software-development-kit/>

MongoDB. (2020). *mongoDB*. Obtenido de <https://www.mongodb.com/es/what-is-mongodb>

Naranjo, D. (2018). *UbuLog*. Obtenido de Linux Multimedia Studio: <https://ubunlog.com/linux-multimedia-studio-una-aplicacion-profesional-de-creacion-de-musica/>

Nelson, J. (7 de Junio de 2015). *'Mayday! Deep Space' is an Incredibly Unique Interactive Story that Uses Apple's Voice Recognition Technology*. Obtenido de toucharcade: <https://toucharcade.com/2015/01/07/mayday-deep-space-now-available/>

Pallero, D. (05 de Noviembre de 2018). *Ecuador crea sus juegos electrónicos*. Obtenido de www.elcomercio.com/: <https://www.elcomercio.com/guaifai/ecuador-crea-juegos-electronicos.html>

Photoshop. (2019). Obtenido de Significados.com: <https://www.significados.com/photoshop/>

Prats, A. (7 de agosto de 2017). *Scrum Poker: evaluando el esfuerzo de tareas*. Obtenido de Piensa en software, desarrolla en colores: <https://blogs.encamina.com/piensa-en-software-desarrolla-en-colores/scrum-poker-evaluando-esfuerzo-tareas/>

Red Hat. (2020). *Red Hat*. Obtenido de <https://www.redhat.com/es/topics/middleware/what-is->

ANEXOS

Anexo 1. Encuesta aplicada a los estudiantes de la UNIB.E. Elaborado por Guamán (2020)

INSTRUMENTO DE COMPETENCIAS INVESTIGATIVAS

Responda las siguientes preguntas tomando en consideración las opciones planteadas, debe responder todos los ítems con un máximo de una respuesta por pregunta.

1. Seleccione su Carrera

- Software
- Turismo
- Derecho
- Producción
- Nutrición y dietética
- Enfermería

2. ¿Hace uso de videojuegos?

- Sí
- No

En caso de que la respuesta sea "No": Finaliza la encuesta

En caso de que la respuesta sea "Sí" continúa:

3. ¿Con qué frecuencia juega videojuegos? *

- Todos los días
- Cada dos días
- Una vez por semana
- Una vez al mes

4. ¿Qué plataforma prefieres para jugar videojuegos? *

- Smartphone
- Consola de Videojuego
- Tablet
- Pc

5. ¿El género de simulación y carreras le es llamativo a la hora de jugar un videojuego? (ejemplo: F1, Flight Simulator, Dirt Rally) *

- Sí
- No

6. ¿Con qué finalidad juega usted Videojuegos? (Selección múltiple) *

- Entretenimiento
- Competitividad
- Socialización
- Desestrés

7. ¿Consideras indispensable el multijugador en un videojuego? *

- Sí
- No

8. ¿De las siguientes temáticas de videojuegos cual es de tu preferencia? (Selección múltiple) *

- Medieval
- Futurista
- Infantil
- Fantástico
- Guerras

9. ¿Qué tipo de mundo prefieres dentro del videojuego? *

- 3D
- 2D

10. ¿En un videojuego, qué consideras es lo más importante? *

- Gráficos
- Mecánica de juego
- Efectos de sonido
- Temática

11. ¿Crees que un videojuego te permita distraerte en tu tiempo libre? *

- Sí
- No

12. ¿Te interesaría un videojuego en el que tengas que emitir sonidos para ganar? *

- Sí
- No

13. ¿Qué sonido preferirías emitir? *

- Gritos
- Silbidos
- Soplos

Anexo 2. Manual de usuario del videojuego. Elaborado por Guamán (2020)



**Whistle or Die
Manual de Usuario**

Versión: 1.0
Fecha: viernes, 7 de agosto de 2020

Versión 1.0

	Whistle or Die Manual de Usuario	Universidad Iberoamericana del Ecuador
---	-------------------------------------	--

HOJA DE CONTROL

Organismo	Universidad Iberoamericana del Ecuador		
Proyecto	Whistle or Die		
Entregable	Manual de Usuario		
Autor	Accel Guamán		
Versión/Edición	0100	Fecha Versión	07/08/2020
Aprobado por		Fecha Aprobación	07/08/2020
		Nº Total de Páginas	7

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
0100	Versión inicial	Accel Guamán	07/08/2020

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos
Accel Guamán

ÍNDICE

1 DESCRIPCIÓN DEL SISTEMA	4
1.1 Objeto	4
1.2 Funcionalidad	4
2 MAPA DEL SISTEMA	5
2.1 Modelo Lógico	5
2.2 Navegación	5
3 DESCRIPCIÓN DEL SISTEMA	6
3.1 Menú principal	6
3.2 Pantalla de Juego	6
3.3 Pantalla de Puntajes	7

2 MAPA DEL SISTEMA

2.1 Modelo Lógico

El videojuego es del género de simulación, con la temática de una clásica carrera infinita, en la que gana aquel jugador que logre un mayor puntaje sobre los demás. El juego inicia con un avión que está en constante caída libre, el objetivo es evitar que el avión choque ya sea con el suelo o con los obstáculos. Para evitar que esto suceda el usuario debe mantener siempre a flote al avión, para ello el juego usa el sensor del micrófono para detectar cuando el usuario está silbando, mientras así lo haga, el avión empezará a ascender. A medida que se avanza una mayor distancia, el avión descenderá más rápidamente. Lo que hará que el usuario tenga menos tiempo para tomar aire, consecuentemente será más difícil no perder la partida.

2.2 Navegación



1 DESCRIPCIÓN DEL SISTEMA

1.1 Objeto

El videojuego tiene como fin proveer de entretenimiento a la comunidad universitaria, al tener un componente innovador como el control mediante silbidos, se espera que tenga gran aceptación por los jugadores.

1.2 Funcionalidad

El jugador puede controlar la navegación horizontal del avión, es decir giros a la derecha o izquierda, mediante los comandos táctiles (smartphone) o mediante el teclado (PC), y la altura se la maneja al emitir silbidos, cuando el jugador silba, el avión asciende y cuando deja de hacerlo, desciende.

3 DESCRIPCIÓN DEL SISTEMA

El juego está dirigido a los estudiantes de la Universidad Iberoamericana del Ecuador. Existen tres estados de juego, el menú principal, la pantalla de juego y la pantalla de puntajes o Game Over. El flujo es lineal entre estas tres pantallas.

3.1 Menú principal

Esta pantalla muestra el logotipo y permite al usuario ingresar su nombre y empezar un nuevo juego. Si el usuario da click en el ícono "I" se desplegará un menú de información y ayuda, si se da click en Empezar, pasa a la pantalla 3.2 y se inicia el juego.



3.2 Pantalla de Juego

En esta pantalla se ejecuta el juego y el usuario debe empezar a silbar, si el usuario está en una PC debe usar las flechas izquierda o Derecha para moverse, si está en un dispositivo móvil deberá perionar los laterales de la pantalla para realizar los movimientos, además de silbar. Si el usuario permite que el avión colisione con un objeto o con el suelo, la partida llega a su fin y pasa a la pantalla 3.3



3.3 Pantalla de Puntajes

Muestra un listado de los puntajes más altos obtenidos, si el usuario da click en el botón de "Nuevo Juego" regresará a la pantalla 3.2 para iniciar una nueva partida.

