

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



I N F O R M E F I N A L D E T E S I S

**DESARROLLO DE UN VIDEOJUEGO CASUAL MULTIPLATAFORMA PARA
REDUCIR EL ESTRÉS.**

LÍNEA DE INVESTIGACIÓN : Línea 4: Ingeniería De Software – Procesos de desarrollo de software.

ÁREA PRIORIZADA PNCYT : 0403 0302 - Animación para la industria del entretenimiento.

AUTOR : Bach. Iván Marcial Palomino Taype.

ASESOR : Ing. Juan José Oré Cerrón.

ANDAHUAYLAS – APURÍMAC

PERÚ

2016

TABLA DE CONTENIDOS

	Pág.
TABLA DE CONTENIDOS	2
LISTA DE TABLAS	6
LISTA DE FIGURAS	8
RESUMEN	11
ABSTRAC.....	12
INTRODUCCIÓN.....	14
1. PLANTEAMIENTO DEL PROBLEMA.....	15
1.1. REALIDAD PROBLEMÁTICA.....	15
1.2. FORMULACIÓN DEL PROBLEMA.....	20
1.3. OBJETIVOS	20
1.3.1. Objetivo General.....	20
1.3.2. Objetivos Específicos	20
1.4. JUSTIFICACIÓN.....	21
1.5. VIABILIDAD DE LA INVESTIGACIÓN.....	22
1.5.1. Viabilidad Técnica	22
1.5.2. Viabilidad Económica.....	22
1.5.3. Viabilidad Social.....	23
1.6. LIMITACIÓN DEL ESTUDIO	23
2. MARCO TEÓRICO	24
2.1. ESTRÉS	24
2.1.1. Definición.....	24
2.1.2. Tipos de estrés.....	25
2.1.3. Factores que originan el estrés.....	26
2.1.4. Medios emocionales para prevenir y reducir el estrés	27
2.1.5. Manejo de síntomas para reducir el estrés	29
2.1.6. Tratamientos para el estrés.....	30
2.2. COEFICIENTE ALFA DE CRONBACH	32

2.2.1.	Definición.....	32
2.2.2.	Características.....	33
2.3.	VIDEOJUEGOS.....	34
2.3.1.	Definición.....	34
2.3.2.	Videjuegos como entretenimiento.....	35
2.3.3.	Géneros de videjuegos	40
2.3.4.	Desarrollo de videjuegos	43
2.3.5.	Recursos multimedia para el desarrollo.....	44
2.3.6.	Desarrollo nativo vs multiplataforma.....	46
2.4.	METODOLOGÍAS DE DESARROLLO DE VIDEOJUEGOS.....	48
2.4.1.	SUM	48
2.4.2.	GUP (Game Unifed Process).....	53
2.4.3.	XGD (Extreme Game Development).....	57
2.4.4.	Comparación de las metodologías	61
2.4.5.	SCRUM.....	61
2.4.6.	RUP (Rational Unified Process).....	64
2.5.	TELÉFONOS INTELIGENTES.....	69
2.5.1.	Definición.....	69
2.5.2.	Características.....	70
2.6.	HERRAMIENTAS DE DESARROLLO DE VIDEOJUEGOS.....	80
2.6.1.	Motores de desarrollo.....	80
2.6.2.	Lenguajes de programación.....	92
2.6.3.	Herramientas multimedia.....	93
2.7.	CALIDAD DEL SOFTWARE.....	97
2.8.	ESTADO DEL ARTE.....	102
2.8.1.	Experiencias e investigaciones a nivel mundial.....	102
2.8.2.	Experiencias e investigaciones en Latinoamérica.....	104
2.8.3.	Experiencias e investigaciones en Perú.....	105
3.	INGENIERÍA DEL PROYECTO.....	107
3.1.	GESTIÓN DEL PROYECTO	107
3.1.1.	Gestión del alcance.....	107

3.1.2.	Plan de personal	108
3.1.3.	Plan de recursos.....	108
3.1.4.	Plan presupuestal	110
3.1.5.	Cronograma e hitos	111
3.1.6.	Seguimiento del proyecto.....	114
3.1.7.	Gestión de riesgos	115
3.2.	METODOLOGÍA DE DESARROLLO	117
3.2.1.	Concepto.....	117
3.2.2.	Planificación.....	121
3.2.3.	Diseño	135
3.2.4.	Elaboración	154
3.2.5.	Beta.....	158
3.2.6.	Cierre	183
3.3.	HERRAMIENTAS	186
3.3.1.	Software.....	186
3.3.2.	Hardware	187
3.4.	GESTIÓN DE LA CALIDAD DEL VIDEOJUEGO	188
3.5.	DESARROLLO CON OPTIMIZACIÓN EN EL CONSUMO DE RECURSOS.....	198
3.5.1.	Optimización de recursos de audio	198
3.5.2.	Optimización de recursos gráficos.....	198
3.5.3.	Optimización de código fuente.....	199
3.6.	DISTRIBUCIÓN MULTIPLATAFORMA	202
3.6.1.	Distribución para la plataforma Android.....	202
3.6.1.1.	Generación de .apk para teléfonos con SO Android	203
3.6.1.2.	Generación de .apk para Play Store	204
3.6.1.3.	Subir a la Play Store.....	206
3.6.2.	Distribución para la plataforma iOS.....	208
3.6.2.1.	Generación del .app para la teléfonos iPhone	208
3.6.2.2.	Generación del .zip para la App Store.....	216
3.6.2.3.	Subir a la App Store	223
3.7.	METODOLOGÍA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS	228
3.7.1.	Estrategia de la recolección de datos	229

3.7.2. Ficha de medición de reducción de estrés en las personas	230
3.7.3. Cálculo de la reducción del estrés	231
4. RESULTADOS	234
CONCLUSIONES	246
RECOMENDACIONES	247
REFERENCIAS BIBLIOGRÁFICAS	248
GLOSARIO DE TÉRMINOS	260
ANEXOS	264

LISTA DE TABLAS

	Pág.
Tabla 1. <i>Propietarios y lenguajes de programación de los sistemas operativos de dispositivos móviles</i>	19
Tabla 2. <i>Subgéneros del videojuego casual</i>	43
Tabla 3: <i>Comparación desarrollo nativo vs multiplataforma</i>	47
Tabla 4. <i>Comparación de metodologías de desarrollo de videojuegos</i>	61
Tabla 5. <i>Sistemas operativos más usados</i>	70
Tabla 6: <i>Clases del motor de videojuegos Corona SDK</i>	84
Tabla 7: <i>Plan de personal para el proyecto</i>	108
Tabla 8: <i>Plan de recursos, recursos por adquirir</i>	109
Tabla 9: <i>Plan de recursos, recursos ya adquiridos</i>	109
Tabla 10: <i>Plan presupuestal por mes</i>	110
Tabla 11: <i>Evaluación de la lista de riesgos</i>	116
Tabla 12: <i>Plan de contingencia para mitigar riesgos</i>	116
Tabla 13: <i>Cronograma de la fase de concepto</i>	122
Tabla 14: <i>Cronograma de la fase de planificación</i>	122
Tabla 15: <i>Cronograma de la fase de diseño</i>	123
Tabla 16: <i>Cronograma de la fase de elaboración</i>	124
Tabla 17: <i>Cronograma de la fase beta</i>	124
Tabla 18: <i>Cronograma de la fase de cierre</i>	125
Tabla 19: <i>Presupuesto detallado del proyecto</i>	125
Tabla 20: <i>Flujo de eventos de la interfaz primer splash</i>	135
Tabla 21: <i>Flujo de eventos de la interfaz de segundo splash</i>	136
Tabla 22: <i>Flujo de eventos de la interfaz menú principal</i>	137
Tabla 23: <i>Flujo de eventos de la interfaz menú de niveles</i>	137
Tabla 24: <i>Flujo de eventos de la interfaz primer tutorial</i>	138
Tabla 25: <i>Flujo de eventos de la interfaz 10 primeros niveles</i>	139
Tabla 26: <i>Flujo de eventos de la interfaz de segundo tutorial</i>	141
Tabla 27: <i>Flujo de eventos de la interfaz 10 últimos niveles</i>	141
Tabla 28: <i>Flujo de eventos de la interfaz over relay de nivel</i>	143
Tabla 29: <i>Relación entre las iteraciones, requisitos e interfaces</i>	154
Tabla 30: <i>Seguimiento de las iteraciones</i>	156
Tabla 31: <i>Versiones del videojuego</i>	157
Tabla 32: <i>Identificación de CE, CEV, CEI</i>	159
Tabla 33: <i>Caso de prueba de la interfaz primer splash (clase válida)</i>	163
Tabla 34: <i>Caso de prueba de la interfaz de segundo splash (clase válida)</i>	163
Tabla 35: <i>Caso de prueba de la interfaz menú principal (clase válida)</i>	164
Tabla 36: <i>Caso de prueba de la interfaz menú de niveles (clase válida)</i>	165
Tabla 37: <i>Caso de prueba de la interfaz primer tutorial (clase válida)</i>	166

Tabla 38: <i>Caso de prueba de la interfaz 10 primeros niveles (clase válida)</i>	167
Tabla 39: <i>Caso de prueba de la interfaz de segundo tutorial (clase válida)</i>	169
Tabla 40: <i>Caso de prueba de la interfaz 10 últimos niveles (clase válida)</i>	169
Tabla 41: <i>Caso de prueba de la interfaz over relay de nivel (clase válida)</i>	172
Tabla 42: <i>Caso de prueba de la interfaz primer splash (clase inválida)</i>	173
Tabla 43: <i>Caso de prueba de la interfaz de segundo splash (clase inválida)</i>	173
Tabla 44: <i>Caso de prueba de la interfaz menú principal (clase inválida)</i>	174
Tabla 45: <i>Caso de prueba de la interfaz menú de niveles (clase inválida)</i>	175
Tabla 46: <i>Caso de prueba de la interfaz primer tutorial (clase inválida)</i>	176
Tabla 47: <i>Caso de prueba de la interfaz 10 primeros niveles (clase inválida)</i>	176
Tabla 48: <i>Caso de prueba de la interfaz de segundo tutorial (clase inválida)</i>	178
Tabla 49: <i>Caso de prueba de la interfaz 10 últimos niveles (clase inválida)</i>	179
Tabla 50: <i>Caso de prueba de la interfaz over relay de nivel (clase inválida)</i>	181
Tabla 51: <i>Lista de errores detectados y solucionados</i>	182
Tabla 52: <i>Medición de la métrica "Complejidad funcional"</i>	188
Tabla 53: <i>Medición de la métrica "Corrección funcional"</i>	188
Tabla 54: <i>Medición de la métrica "Pertinencia funcional"</i>	189
Tabla 55: <i>Medición de la métrica "Comportamiento temporal"</i>	189
Tabla 56: <i>Medición de la métrica "Utilización de recursos"</i>	190
Tabla 57: <i>Medición de la métrica "Capacidad"</i>	190
Tabla 58: <i>Medición de la métrica "Coexistencia"</i>	191
Tabla 59: <i>Medición de la métrica "Capacidad de aprendizaje"</i>	192
Tabla 60: <i>Medición de la métrica "Capacidad para ser usado"</i>	192
Tabla 61: <i>Medición de la métrica "Estética de la interfaz de usuario"</i>	193
Tabla 62: <i>Medición de la métrica "Madurez"</i>	193
Tabla 63: <i>Medición de la métrica "Disponibilidad"</i>	194
Tabla 64: <i>Medición de la métrica "Integridad"</i>	194
Tabla 65: <i>Medición de la métrica "Modularidad"</i>	195
Tabla 66: <i>Medición de la métrica "Reusabilidad"</i>	196
Tabla 67: <i>Medición de la métrica "Capacidad para ser modificado"</i>	196
Tabla 68: <i>Medición de la métrica "Adaptabilidad"</i>	197
Tabla 69: <i>Medición de la métrica "Capacidad para ser instalado"</i>	197
Tabla 70: <i>Optimización de recursos de audio</i>	198
Tabla 71: <i>Optimización de recursos gráficos</i>	199
Tabla 72: <i>Relación entre la variación de los medios y síntomas del estrés y su reducción</i>	232
Tabla 73: <i>Imágenes de las interfaces del videojuego funcionando sobre los sistemas operativos Android e iOS de teléfonos inteligentes</i>	236
Tabla 74: <i>Recursos de hardware consumidos por los videojuegos</i>	241
Tabla 75: <i>Comparación de recursos de hardware consumidos por los otros videojuegos y el videojuego desarrollado</i>	241
Tabla 76: <i>Estrés reducido por medio y síntoma</i>	242

LISTA DE FIGURAS

	Pág.
Figura 1. Incremento en el uso de los smartphones en el mundo	17
Figura 2. Dispositivos móviles que tienen aplicaciones instaladas	17
Figura 3. Videojuego Bejeweled 2	36
Figura 4. Videojuego Peggle	37
Figura 5. Videojuego Bookworm Adventures.....	37
Figura 6: Fases del proceso de entrega de la metodología SUM	50
Figura 7. Metodología GUP	54
Figura 8. Fases de la metodología XGD	57
Figura 9: Fases y disciplinas de RUP	65
Figura 10: Subsistemas del motor de videojuegos Corona SDK.....	83
Figura 11: Diagrama de clases del motor de videojuegos Corona SDK.....	83
Figura 12: Cronograma e hitos, parte 1.....	112
Figura 13: Cronograma e hitos, parte 2.....	113
Figura 14: Seguimiento del proyecto, parte 1	114
Figura 15: Seguimiento del proyecto, parte 2	115
Figura 16. Flujo del desarrollo del concepto	117
Figura 17. Flujo de tareas de la planificación de la administración.....	121
Figura 18. Flujo de tareas de la especificación del videojuego	126
Figura 19: Estimación y priorización de los requisitos parte 1	133
Figura 20: Estimación y priorización de requisitos parte 2	134
Figura 21. Flujo de tareas del diseño de las interfaces y paquetes del videojuego	135
Figura 22: Diseño de la interfaz primer splash	136
Figura 23: Diseño de la interfaz segundo splash	136
Figura 24: Diseño de la interfaz menú principal.....	137
Figura 25: Diseño de la interfaz menú de niveles.....	138
Figura 26: Diseño de la interfaz primer tutorial	138
Figura 27: Diseño de la interfaz 10 primeros niveles	140
Figura 28: Diseño de la interfaz segundo tutorial	141
Figura 29: Diseño de la interfaz 10 últimos niveles	143
Figura 30: Diseño de la interfaz over relay de nivel	144
Figura 31: Flujo alterno OPCIONES.....	144
Figura 32: Flujo alterno REINICIAR	145
Figura 33: Flujo alterno cambiar idioma a Inglés	145
Figura 34: Flujo alterno cambiar idioma a Español	146
Figura 35: Flujo alterno subir volumen.....	146
Figura 36: Flujo alterno bajar volumen	147
Figura 37: Flujo alterno CRÉDITOS.....	147
Figura 38: Flujo alterno SALIR.....	148

Figura 39: Flujo alternativo intentar acceder a nivel bloqueado	148
Figura 40: Flujo alternativo salir del menú de niveles.....	149
Figura 41: Flujo alternativo no activar movimiento del personaje	149
Figura 42: Flujo alternativo mostrar over relay de pausa	150
Figura 43: Flujo alternativo regresar al menú de niveles desde el over de pausa	150
Figura 44: Flujo alternativo reanudar juego.....	151
Figura 45: Flujo alternativo reiniciar juego desde interfaz de nivel	151
Figura 46: Flujo alternativo reiniciar juego desde interfaz de over de nivel.....	152
Figura 47: Flujo alternativo regresar al menú de niveles desde el over de nivel	152
Figura 48: Diseño de paquetes	153
Figura 49: Flujo de tareas de la planificación de la iteración de la elaboración.....	154
Figura 50: Flujo de la tarea desarrollo de las características.....	156
Figura 51: Flujo de la tarea seguimiento de la iteración	156
Figura 52: Flujo de tareas del cierre de la iteración	157
Figura 53: Flujo de tareas de la planificación de la iteración de la fase beta.....	158
Figura 54: Flujo de la tarea verificación del videojuego	159
Figura 55: Flujo de las tareas de corrección del videojuego	182
Figura 56: Flujo de la tarea evaluación del proyecto.....	184
Figura 57: Flujo de la tarea liberación del videojuego.....	185
Figura 58: Flujo de la tarea despliegue del videojuego	186
Figura 59: Modelo del despliegue del videojuego.....	186
Figura 60: Ruta para compilación para teléfonos Android.....	203
Figura 61: Configuración para compilación para teléfonos para Android	204
Figura 62: Ruta para compilación para Play Store.....	205
Figura 63: Configuración para compilar para Play Store	205
Figura 64: Subir videojuego a Play Store	206
Figura 65: Requisitos para subir el videojuego a Play Store.....	207
Figura 66: Creación de certificado para iOS	209
Figura 67: Solicitar un certificado a una autoridad de certificación.....	209
Figura 68: Generación del certificado para iOS.....	210
Figura 69: Identificador de aplicación para iOS.....	210
Figura 70: Configuración del ID de aplicación para iOS.....	211
Figura 71: Dispositivo (iPhone) donde se instala el videojuego	211
Figura 72: Configuración del dispositivo (iPhone) donde se instala el videojuego	212
Figura 73: Perfil de aprovisionamiento para compilar para dispositivos iPhone.....	212
Figura 74: Selección del tipo de certificado para el Perfil de aprovisionamiento para dispositivos iPhone	213
Figura 75: Selección del ID de aplicación para el Perfil de aprovisionamiento para dispositivos iPhone	213
Figura 76: Selección del certificado para el Perfil de aprovisionamiento para compilar para dispositivos iPhone	214
Figura 77: Selección del dispositivo iPhone donde se instala el videojuego.....	214

Figura 78: Generación del Perfil de aprovisionamiento para compilar para dispositivos iPhone .	215
Figura 79: Ruta para compilar para dispositivos iOS	215
Figura 80: Configuración para compilar para dispositivos iOS	216
Figura 81: Instalación del .ipa en iPhone 3GS	216
Figura 82: Selección del tipo de certificado para el Perfil de aprovisionamiento para subir a App Store	220
Figura 83: Generación del Perfil de aprovisionamiento para subir a App Store	222
Figura 84: Ruta para compilar para subir a App Store	222
Figura 85: Configuración para compilar para subir a la App Store.....	223
Figura 86: Subir videojuego a App Store	223
Figura 87: Enviar .zip del videojuego a la App Store	224
Figura 88: Secciones a configurar para subir videojuego a App Store	224
Figura 89: Configuración de la sección Información de la app.....	225
Figura 90: Configuración de la sección Precio y disponibilidad.....	226
Figura 91: Configuración de la sección Preparar para la presentación	227
Figura 92: Videojuego listo para ser revisado por Apple.....	228
Figura 93: Videojuego subido a Play Store	240
Figura 94: Videojuego subido a App Store	240
Figura 95: Estrés reducido por persona.....	244

RESUMEN

En el presente informe final de tesis se detalla cómo se desarrolló el videojuego “Casual Labyrinth” de género casual que puede ejecutarse de manera eficiente y sin problemas en teléfonos inteligentes de gama baja con sistemas operativos Android e iOS y que además posee la capacidad de reducir el estrés de las personas. Para dicho desarrollo se utilizó la metodología “SUM” en combinación con las metodologías “GUP” y “RUP” las cuales permitieron realizar un desarrollo más entendible y sólido. Además se utilizó el motor de desarrollo de videojuegos Corona SDK que utiliza el lenguaje de programación Lua.

En la primera fase llamada “inicio” se detalló el concepto general del videojuego y se realizó la designación del personal para el desarrollo del videojuego.

La segunda fase llamada “planificación” se detalló de manera específica la mecánica del videojuego mediante requisitos, además estos fueron priorizados por relevancia.

La tercera fase llamada “diseño” se representó de manera gráfica las interfaces y se detalló el flujo de los eventos realizados por el usuario en cada interfaz, también se especificó la estructura interna de los paquetes que contienen el código fuente del videojuego.

En la cuarta fase llamada “elaboración” se escribió el código fuente mediante iteraciones que dieron vida al videojuego y se realizó la optimización en el consumo de recursos.

En la quinta fase llamada “beta” se realizaron las pruebas de funcionamiento de las interfaces que fueron diseñadas y programadas, para ello se usó casos de prueba.

Y en la última fase llamada “cierre” se realizó el despliegue del videojuego a las plataformas Android e iOS a través de sus tiendas virtuales Play Store y App Store respectivamente.

Los resultados obtenidos permiten concluir que los videojuegos casuales tienen un enorme potencial para reducir el estrés y ello viene siendo descubierto y comprobado en los últimos años. Además se demuestra que sí es posible desarrollar videojuegos multiplataforma, es decir escribir el código fuente una vez y luego compilarlos para varios sistemas operativos de teléfonos inteligentes, y obtener a la vez un desempeño o rendimiento eficiente del videojuego en dichas plataformas de manera equivalente a que si el videojuego hubiese sido programado de manera nativa para un determinado sistema operativo.

PALABRAS CLAVE: videojuegos casuales, teléfonos inteligentes, Android, iOS, gama baja, reducción de estrés.

ABSTRAC

In this thesis details how was development the video game "Casual Labyrinth" of casual genre that can run efficiently and smoothly on smartphones low-end with operating systems Android and iOS and also has the ability to reduce the stress of people.

For this development the "SUM" methodology was used in combination with the methodologies "GUP" and "RUP" which allowed for development more understandable and solid. In addition, the video game development engine Corona SDK that uses the Lua programming language was used.

In the first phase called "start" the general concept of the game is detailed and the appointment of staff to the development of the video game was made.

The second phase called "planning" specifically detailing the mechanics of the video game by requirements in addition these were prioritized by relevance.

The third phase called "design" is represented graphically was interfaces and the flow of events performed by the user at each interface is detailed, the internal structure of the packages containing the source code of the game is also specified.

In the fourth phase called "elaboration" source code was written by iterations that gave life to the game and optimizing resource consumption was performed.

In the fifth phase called "beta", was made testing of operation of interfaces that was designed and developed, test cases was used.

And in the last phase called "closing" was made the deployment of the video game to the platforms Android and iOS through their virtual stores Play Store and App Store respectively.

The results allow to conclude that casual video games have enormous potential to reduce stress and it is being discovered and tested in recent years. In addition it is shown that it is possible to develop multiplatform video games, ie write the source code once and then compile them for multiple operating systems of smartphones, and achieve efficient performance of the video game on these platforms a manner equivalent to that if the video game had been developed natively for a particular operating system.

KEYWORDS: casual video games, smartphones, Android, iOS, low-end, relief stress.

AGRADECIMIENTOS

A Dios por darme la vida y la capacidad de razonar para darme cuenta que ayudar a los prójimos es una de las mejores acciones de la vida.

A mis padres Moisés y Martha, quienes me brindan su cariño y apoyo incondicional en cada momento, por quienes hoy llego a ser quien soy.

A mi hermana Kely, por darme ánimo y ser un ejemplo como persona y como profesional.

A mi amigo Jhancarlo, con quien comparto la misma afición por el mundo videojuegos, con quien además compartí gratas experiencias durante la vida académica.

A los docentes de la Universidad Nacional José María Arguedas, quienes me brindaron su vasto conocimiento y experiencia, así como sus consejos para ser una mejor persona y un mejor profesional, especialmente a la Lic. Frida Fuentes, Ing. Cecilia García, al Ing. Enrique Condor, Ing. Carlos Silva, Ing. Herwin Huillcen, Ing. Julio Huanca, Ing. Daniel Remicio, Ing. Ángel Navarro, Ing. Marlon Canales, Ing. Carlos Castro, Prof. Jorge Vivanco.

A mi asesor Ing. Juan José Oré, por apoyarme durante el desarrollo de este proyecto.

A mi amigo y docente Ing. Jorge Cotrado por mostrarme las ilimitadas posibilidades de los videojuegos.

A los Psicólogos Carmen Salas, Mirella Morales y Efraín Altamirano por apoyarme en el desarrollo del proyecto en lo referente a la medición de reducción estrés.

A mis amigos y compañeros de clase Elizabeth, Rut, Frandio, Pompeyo, Lenin, Roy Iván, Bequer, Emerson, Hans, Ricardo, Juan, con quienes compartí gratas experiencias.

A todas las personas que colaboraron con la Ficha de Medición de Reducción de Estrés.

INTRODUCCIÓN

El constante cambio para adaptarse a una sociedad cada vez más compleja así como el enfrentamiento a los problemas cotidianos y laborales hacen que las personas se estresen y tengan que lidiar con esta situación casi todo el tiempo. Esta situación de estrés afecta a más personas de las que se cree y puede conllevar a situaciones peligrosas, ya que estar en constante estado de estrés hace que el sistema inmunológico no funcione adecuadamente y ello puede desencadenar en estar vulnerable a cualquier enfermedad.

Este proyecto plantea crear un videojuego de género casual para combatir el estrés de las personas, basados en investigaciones realizadas en las universidades del Este de California, Oxford, del Estado de Bridgewater y del Estado de San José, así como el hospital universitario de Bellvitge donde encontraron que los videojuegos tienen el potencial para reducir el estrés.

Por otra parte el avance tecnológico presenta nuevos desafíos, esto en cuanto a las plataformas de los teléfonos inteligentes, donde Android e iOS dominan el mercado seguido de Windows Phone, Symbian, Blackberry y otros. Esto es un reto para los desarrolladores de videojuegos tanto de empresas grandes, medianas y pequeñas puesto que desarrollar de manera nativa para cada plataforma incurre en la multiplicación de recursos necesarios para obtener un videojuego que funcione en esas plataformas móviles de manera eficiente.

Este proyecto pretende utilizar tecnología basada en programación multiplataforma para obtener los instaladores del videojuego para los sistemas operativos móviles más populares a nivel mundial, es decir Android e iOS a partir de un mismo código fuente, sin realizar modificación alguna y garantizar que dicho videojuego sea capaz de funcionar óptimamente sobre ambos sistemas operativos consumiendo menos recursos de hardware que el promedio de otros videojuegos.

Para realizar este proyecto se requiere de un tiempo estimado de 11 meses, así como el personal necesario para el desarrollo del videojuego que incluye un programador, un encargado del sonido, un encargado de los gráficos, un director del proyecto y un revisador externo, también se requiere el apoyo de especialistas en Psicología Humana.

1. PLANTEAMIENTO DEL PROBLEMA

1.1. REALIDAD PROBLEMÁTICA

Junto con el avance de la globalización que afecta los aspectos económicos, sociales, tecnológicos y culturales de la vida de las personas, se incrementa también el estrés, el cual es conocido como la enfermedad del siglo XXI (riniga, 2009; PLENIUM, 2015), la OMS (Organización Mundial de la Salud) define al estrés como el conjunto de reacciones fisiológicas que preparan al organismo para la acción (Idóneos, 2013).

Según el IBR (International Bussines Report) el estrés afecta a las personas de diversos países, dentro de los cuales, los más afectados son: China con 76%, México con 75%, Turquía con 72%, Vietnam con 72% y Grecia con 68% (Bienestar180, 2015). En cuanto al Perú los resultados son similares, dichos resultados proporcionados por el portal de noticias de RPP (2015) (Radio Programas del Perú) indican que aproximadamente 52% de la población peruana masculina vive estresada y 63% de la población femenina también padece esta situación.

La facultad de estudios superiores Acatlán de la UNAM (Universidad Autónoma de México), señala que el estrés se produce por cualquier situación que provoca

emociones, las cuales pueden ser buenas o malas, siendo las emociones malas las que afectan negativamente a la salud de las personas, este estrés negativo es llamado distrés.

Cuando el distrés está presente de manera prolongada afecta seriamente a las personas, tales efectos fisiológicos son: aumento del ritmo cardiaco, aumento de la presión sanguínea, tensión muscular, dolor de cabeza. También se producen efectos a nivel emocional como miedo, irritabilidad, depresión, ansiedad, enojo y desmotivación. Todo ello conlleva a que las personas no puedan concentrarse, no puedan percibir su entorno de manera adecuada, tengan dificultad para resolver problemas e incluso reduzcan su capacidad de aprendizaje.

Sin embargo según la OMS uno de los factores que más causa estrés en las personas es el estrés laboral (RPP, 2015). El estrés laboral es un patrón de reacciones psicológicas, cognitivas y conductuales que aparecen cuando una persona enfrenta exigencias laborales que no corresponden al campo de su conocimiento, destreza o habilidad (Stavroula, Griffiths & Cox, 2004).

La globalización hace que no solo los países desarrollados como los que pertenecen a la Unión Europea se vean afectados por el estrés laboral, sino también los países en vías de desarrollo, debido a que la globalización genera cambios rápidos en la naturaleza de los trabajos, los cuales giran en torno a la actividad económica.

Debido al problema que genera el estrés en la vida de las personas, se inventan maneras de disuadirlo, evitarlo o atenuarlo, además de los tratamientos médicos se pueden listar los medios de entretenimiento complementarios que hacen frente al estrés que se genera cada día, dichos medios de entretenimiento son la industria del cine, de la música, de los videojuegos y demás. Sin embargo tener acceso a estos medios de entretenimiento requiere disponer de varios factores, tales como tiempo, medios (un DVD con la película, una radio, un teléfono móvil), dinero, etc. Esto hace que las personas tengan que elegir un medio de distracción, rápido, eficiente y barato.

La masificación del uso de los smartphones (teléfonos inteligentes) a nivel mundial, creció tal como se muestra en el siguiente gráfico proporcionado por el diario El Comercio (2014a):

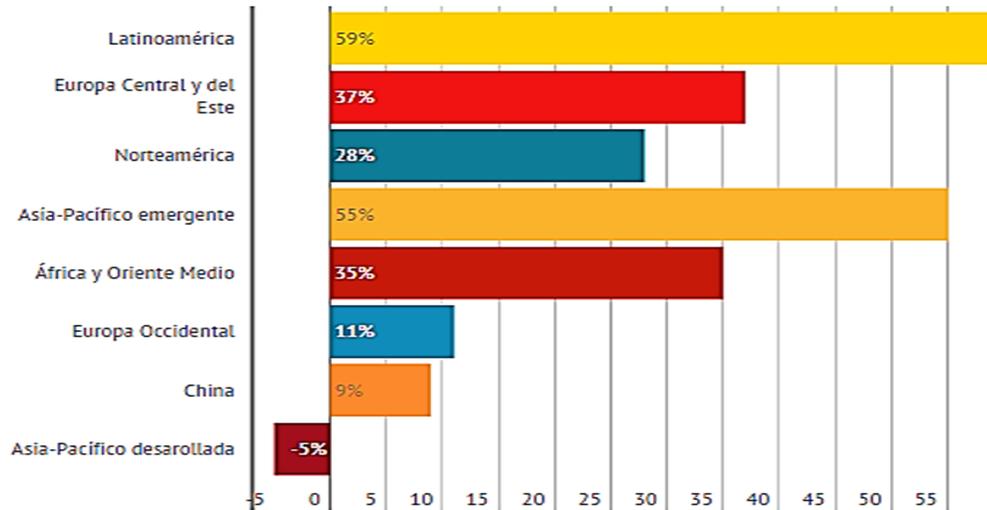


Figura 1. Incremento en el uso de los smartphones en el mundo
 Fuente: <http://elcomercio.pe/paginas/smartphones-tablets/latinoamerica-comando-aumento-smartphones-mundo-noticia-1792109>

Donde se puede apreciar el aumento general que tuvo el uso de los smartphones en cada uno de los continentes.

También se incrementó el uso de aplicaciones móviles en los últimos años, el siguiente gráfico obtenido del diario La República muestra el porcentaje de personas que tiene aplicaciones en sus dispositivos móviles (mayores porcentajes), esto para países latinoamericanos (Amaro, 2015).

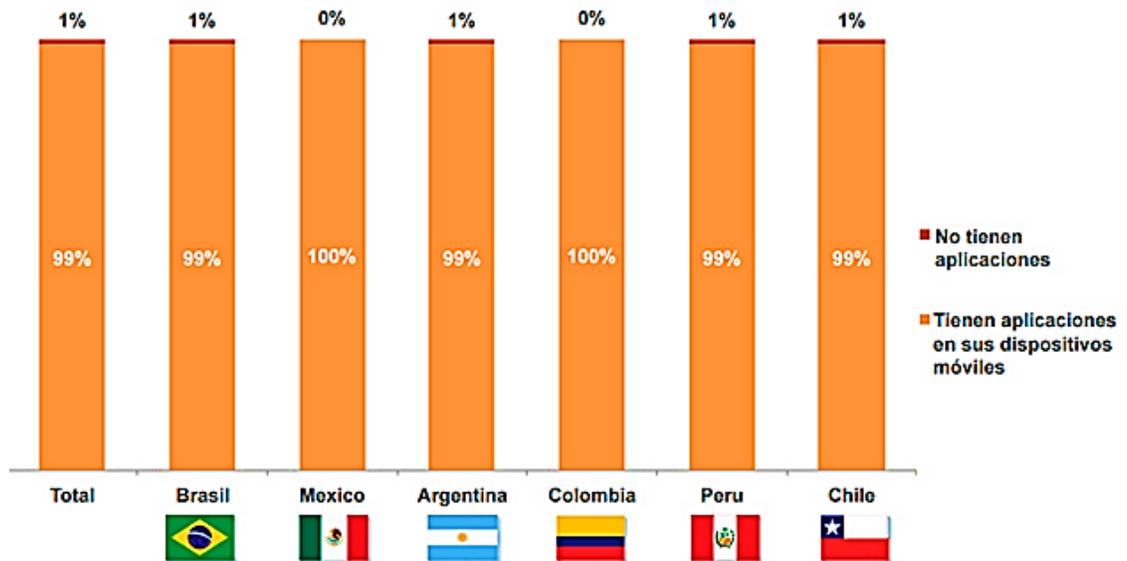


Figura 2. Dispositivos móviles que tienen aplicaciones instaladas
 Fuente: <http://larepublica.pe/marketing/4540-uso-de-smartphones-en-peru-es-uno-de-los-mas-crecientes-en-latinoamerica>

Debido a los datos estadísticos mostrados anteriormente se ve la gran oportunidad de optar por la industria de los videojuegos en teléfonos inteligentes como medio de entretenimiento para combatir el estrés.

Sin embargo esto conlleva a una nueva problemática vinculada a las empresas desarrolladoras de videojuegos así como a desarrolladores independientes, y es que el portal de noticias El País, publicó un informe donde menciona que los teléfonos inteligentes más usados en Sudamérica son los de gama baja y media (Fernández, 2015). De manera similar en Perú se registró la entrada de 2.9 millones de teléfonos inteligentes, de los cuales el 39.7% pertenecen exclusivamente a la gama baja y 51,7% pertenecen a la gama media-baja, con lo cual se puede ver que el mercado está dominado por este tipo de teléfonos inteligentes (Netdreams, 2015). Dichos teléfonos inteligentes poseen bajos recursos de hardware.

Sumado a esto las personas adquieren sus teléfonos inteligentes con la finalidad de poder disfrutar de ellos, así que instalan muchas aplicaciones, almacenan fotos, videos, música y con el paso del tiempo el teléfono se va haciendo más lento hasta el punto de que es necesario reiniciarlo para seguir usándolo o se ponen a desinstalar las aplicaciones, borrar las fotos, música y video o incluso instalar aplicaciones para optimizar el rendimiento que en realidad son drenadoras de recursos, todo ello es molesto y esto es corroborado por el reporte de la empresa que desarrolló el antivirus AVG (2015), donde para el primer trimestre del 2015 lista aplicaciones y videojuegos que prácticamente hacen añicos al teléfono inteligente en lo que a consumo de recursos se refiere, es decir que consumen gran cantidad de batería, memoria RAM y espacio de almacenamiento, recursos que no son abundantes en un teléfono inteligente de gama baja. Algunas de dichas aplicaciones perjudiciales son: Spotify Music, Line, BBC News, Netflix, Facebook, Facebook Messenger, WhatsApp, OLX, Weather and Clock Widget Android, etc. En cuanto a los videojuegos se tienen los de 2D y 3D como: Hay Day, Farm Heroes Saga, Candy Crush Saga, Clash of Clans, My Talking Tom, My Talking Angela, Solitaire, Pet Rescue Saga, 8 Ball Pool, Temple Run 2, etc. Y dentro de las aplicaciones para optimizar el rendimiento y antivirus se tiene: Clean Master, AppLock, Security Police Updates, Lockout Security and Antivirus, etc.

Además los desarrolladores se ven ante una nueva dificultad debido al avance tecnológico, el cual hizo que ahora se tengan distintos SO (sistemas operativos) para los smartphones, esto hace que programar para un determinado sistema operativo

incluya necesariamente aprender su lenguaje correspondiente o nativo, esto se muestra en la siguiente tabla, proporcionada por la empresa de desarrollo de aplicaciones Digimark (2012):

Tabla 1. *Propietarios y lenguajes de programación de los sistemas operativos de dispositivos móviles*

Sistema Operativo	Fabricante/Propietario	Lenguaje de Programación
Android	Google	Java
iPhone OS (iOS)	Apple	Objetive – C
Windows Phone	Microsotf	C#
BlackBerry OS	RIM	Java
Symbian OS	Fundación Symbian	C++
Bada	Samsung	C

Fuente: <http://www.digimark.bg/en/mobile-os-platforms>

Por lo tanto la única manera de optimizar el funcionamiento de un videojuego en un smartphone es programar en su lenguaje nativo, pero cuando el desarrollador necesita programar para otro sistema operativo tiene que volver a desarrollar la aplicación, con lo cual duplica el trabajo, tiempo y recursos invertidos en el desarrollo del videojuego; igual o más laborioso es cuando el desarrollador pretende programar para tres o más sistemas operativos diferentes donde el trabajo y recursos se triplica o multiplica mucho más. Así lo confirma la empresa de desarrollo de aplicaciones Crew (2015), donde el costo para desarrollar una aplicación básica para Android es de \$/. 15,200.00 dólares, el costo para desarrollar una aplicación básica para iOS es de \$/. 15,200.00 dólares y el costo para desarrollar una aplicación básica para Android e iOS es de \$/. 30,400.00 dólares.

En cuanto al desarrollo de videojuegos el costo promedio de desarrollarlos va desde \$/. 10,000.00 hasta \$/. 250.000.00 dólares por plataforma, dicho costo depende en gran medida del alcance que se propone que tendrá el videojuego, es decir que un videojuego simple costará mucho menos que un videojuego complejo (Thomas, 2011; Skreech, 2014).

Esto le sucedió a la empresa de desarrollo de videojuegos Rovio Entertainment que el 2009 publicó el videojuego Angry Birds, con una inversión de \$/. 125.000.00 dólares (Skreech, 2014), exclusivamente para el sistema operativo iOS y debido al éxito que tuvo el videojuego tuvieron que volver a desarrollarlo para las los sistemas operativos Android, Windows Phone y demás sistemas operativos (nehuen, 2011).

Es por todo ello que el desarrollo de videojuegos lo más livianos posible para que puedan ejecutarse en teléfonos inteligentes de bajo rendimiento, y en varios sistemas operativos, garantiza que el entretenimiento pueda llegar a todas las personas de diversos niveles socioeconómicos, mejorándoles la calidad de vida y brindándoles momentos de relajación para combatir el estrés, este desarrollo también hace conocer a desarrolladores con empresas y desarrolladores independientes la tecnología que puede ser usada para programar eficientemente y minimizar el consumo de recursos económicos en dicho proceso.

1.2. FORMULACIÓN DEL PROBLEMA

¿Existe un videojuego de género casual que reduzca el estrés de las personas y que funcione fluidamente en los teléfonos inteligentes de gama baja con sistemas operativos Android e iOS?

1.3. OBJETIVOS

1.3.1. Objetivo General

Desarrollar un videojuego multiplataforma para teléfonos inteligentes que consuma bajos recursos de hardware y que sea capaz de reducir de los niveles de estrés de las personas.

1.3.2. Objetivos Específicos

- Determinar el videojuego apropiado de género casual.
- Diseñar el videojuego.
- Desarrollar el videojuego.
- Realizar las pruebas de funcionamiento del videojuego.
- Compilar el videojuego para que funcione en las plataformas Android e iOS de los teléfonos inteligentes.
- Lograr que el videojuego funcione fluida e íntegramente en los teléfonos inteligentes de gama baja.

- Evaluar el nivel de estrés de las personas usuarias del videojuego que tendrá la capacidad de captar la atención del usuario para brindarle momentos de relajación y distracción.

1.4. JUSTIFICACIÓN

La presente ejecución del proyecto de tesis justificó su realización por los siguientes motivos:

- En la actualidad el uso de los teléfonos inteligentes ha crecido de sobre manera, los cuales usan diferentes sistemas operativos o plataformas. Esto hace que el desarrollo de una aplicación móvil tenga que realizarse de manera particular para que pueda funcionar en un determinado sistema operativo.
- Los sistemas operativos más utilizados hoy en día son Android e iOS (véase tabla 5), por ello el desarrollo de un videojuego que funcione en dichas plataformas garantiza un alcance masivo al público usuario joven y adulto.
- Aunque se ha masificado la acogida de los teléfonos inteligentes, se puede observar que el precio de estos juega un papel muy importante en su desempeño, es decir que existe una relación directamente proporcional entre el precio y el desempeño o capacidad del teléfono inteligente ya que en muchos teléfonos inteligentes de bajo costo que pertenecen a la gama baja, algunos videojuegos no funcionan de manera eficiente y óptima.
- Es importante desarrollar un videojuego que funcione en los teléfonos inteligentes de gama baja y por consecuencia es los de gama media y superior, permitiendo así que todas las personas de diversos niveles socioeconómicos tengan acceso al videojuego para poder entretenerse.
- Usar videojuegos que funcionan en teléfonos inteligentes tiene un gran potencial para llegar a muchos usuarios, puesto que según el estudio realizado por la Harvard Business Review, el 46% de usuarios de teléfonos inteligentes usa estos dispositivos para relajarse y entretenerse (Management Society, 2015).
- Se sabe que el estrés favorece al desarrollo de todas las enfermedades porque disminuye las defensas (Lifshitz, 2012), por ello es de suma importancia combatirlos mediante medios de distracción como los videojuegos.

- Las personas del siglo XXI son las más afectadas por el estrés, y esto irá en aumento, según la OMS el estrés crónico será la segunda causa de invalidez en las personas para el 2020 (riniga, 2009), así que brindarles un medio de relajación y entretenimiento como un videojuego es muy importante para ayudar a muchas personas y mejorarles su calidad de vida.
- “La industria del entretenimiento y de los videojuegos es tres veces más grande que la del cine y dos veces más grande que la música.” (El Comercio, 2014b, Párrafo 5).
- Además la realización de este videojuego permitirá servir como fuente de información a investigadores interesados en el tema de los videojuegos.

1.5. VIABILIDAD DE LA INVESTIGACIÓN

Es importante mencionar que la disponibilidad de tiempo fue favorable ya que el planeamiento y ejecución el proyecto fue posible llevarlo a cabo en un periodo no muy extenso, el cual fue de 11 meses.

1.5.1. Viabilidad Técnica

El presente proyecto de investigación fue viable técnicamente porque la adquisición de las herramientas de hardware (véase sección 3.3.2) fue posible, excepto la laptop Mac que no pudo ser adquirida por motivos económicos, en cuanto a las herramientas software (véase sección 3.3.1), todas pudieron ser adquiridas vía web, mientras que las demás herramientas software eran de licencia de uso gratuita.

También se contó con la información y soporte necesario de la metodología a emplear para el desarrollo del videojuego.

1.5.2. Viabilidad Económica

El proceso para distribuir el producto del proyecto en las tiendas web tanto de Android (Play Store) e iOS (App Store) incluyó un costo en dólares mayor para iOS que para Android, dicho costo pudo ser asumido por quien presenta este informe.

Cabe mencionar también que ya se contaba con algunas herramientas hardware y software.

1.5.3. Viabilidad Social

Se debe resaltar que el desarrollo de este proyecto no tuvo efectos negativos para la sociedad ya que al estar dirigido a un público joven y adulto se pudo asegurar un uso responsable del producto del proyecto. Además su realización estuvo enfocada a incluir socialmente a las personas de todos los niveles socioeconómicos y brindarles momentos de sana diversión que beneficien su salud emocional.

1.6. LIMITACIÓN DEL ESTUDIO

El presente proyecto no presentó limitaciones en cuanto al tiempo, pero si en recursos económicos como se mencionó en la viabilidad económica; en cuanto al público al cual estuvo dirigido el proyecto, principalmente aquellas personas afectadas por el estrés las cuales cada día van aumentando en número no hubo limitaciones para lograr que puedan usar el videojuego. También se contó con información de la metodología que se usó.

2. MARCO TEÓRICO

2.1. ESTRÉS

2.1.1. Definición

La empresa de salud Sanitas (2014) en su artículo “Qué es el estrés”, menciona que el estrés es “un conjunto de reacciones fisiológicas que se presentan cuando una persona sufre un estado de tensión nerviosa, producto de diversas situaciones en el ámbito laboral o personal: exceso de trabajo, ansiedad, situaciones traumáticas que se hayan vivido, etc.” (Párrafo 2).

Así también el portal web de salud WebConsultas (2013a) define al estrés como:

Un proceso natural del cuerpo humano, que genera una respuesta automática ante condiciones externas que resultan amenazadoras o desafiantes, que requieren una movilización de recursos físicos, mentales y conductuales para hacerles frente, y que a veces perturban el equilibrio emocional de la persona (Párrafo 1).

De manera similar el Centro de salud física emocional PLENIUM (2015), define al estrés como:

Se entiende por estrés aquella situación en la cual las demandas externas (sociales, laborales, etc.) o las demandas internas (psicológicas, auto exigencia, etc.) superan nuestra capacidad de respuesta. Se provoca así una alarma orgánica que actúa sobre el sistema nervioso, cardiovascular, endocrino e inmunológico, produciendo un desequilibrio psicofísico y la consiguiente aparición de la enfermedad (Párrafo 1).

Puede verse que el estrés es originado cada día ya que no podemos estar distantes de actividades tan comunes como las ya mencionadas, es por eso que es necesario buscar maneras de lidiar con este problema y reducirlo lo más que sea posible.

2.1.2. Tipos de estrés

Según la Asociación Americana de Psicología, existen tres tipos de estrés detallados a continuación:

A. Agudo

Definido por Miller & Smith (2015):

El estrés agudo es la forma de estrés más común. Surge de las exigencias y presiones del pasado reciente y las exigencias y presiones anticipadas del futuro cercano. El estrés agudo es emocionante y fascinante en pequeñas dosis, pero cuando es demasiado resulta agotador. Una bajada rápida por una pendiente de esquí difícil, por ejemplo, es estimulante temprano por la mañana. La misma bajada al final del día resulta agotadora y desgastante. Esquiar más allá de sus límites puede derivar en caídas y fracturas de huesos. Del mismo modo, exagerar con el estrés a corto plazo puede derivar en agonía psicológica, dolores de cabeza tensionales, malestar estomacal y otros síntomas (Párrafo 2).

B. Agudo episódico

Definido por Miller & Smith (2015):

Aquellas personas que tienen estrés agudo con frecuencia, cuyas vidas son tan desordenadas que son estudios de caos y crisis. Siempre están apuradas, pero siempre llegan tarde. Si algo puede salir mal, les sale mal.

Asumen muchas responsabilidades, tienen demasiadas cosas entre manos y no pueden organizar la cantidad de exigencias autoimpuestas ni las presiones que reclaman su atención. Es común que las personas con reacciones de estrés agudo estén demasiado agitadas, tengan mal carácter, sean irritables, ansiosas y estén tensas. Suelen describirse como personas con "mucha energía nerviosa". Siempre apuradas, tienden a ser cortantes y a veces su irritabilidad se transmite como hostilidad. Las relaciones interpersonales se deterioran con rapidez cuando otros responden con hostilidad real. El trabajo se vuelve un lugar muy estresante para ellas (Párrafos 6-7).

C. Crónico

Definido por Miller & Smith (2015):

Si bien el estrés agudo puede ser emocionante y fascinante, el estrés crónico no lo es. Este es el estrés agotador que desgasta a las personas día tras día, año tras año. El estrés crónico destruye al cuerpo, la mente y la vida. Hace estragos mediante el desgaste a largo plazo. Es el estrés de la pobreza, las familias disfuncionales, de verse atrapados en un matrimonio infeliz o en un empleo o carrera que se detesta. El estrés crónico surge cuando una persona nunca ve una salida a una situación deprimente. Es el estrés de las exigencias y presiones implacables durante períodos aparentemente interminables. Sin esperanzas, la persona abandona la búsqueda de soluciones. Algunos tipos de estrés crónico provienen de experiencias traumáticas de la niñez que se interiorizaron y se mantienen dolorosas y presentes constantemente. Algunas experiencias afectan profundamente la personalidad. Se genera una visión del mundo, o un sistema de creencias, que provoca un estrés interminable para la persona (por ejemplo, el mundo es un lugar amenazante, las personas descubrirán que finge lo que no es, debe ser perfecto todo el tiempo) (Párrafos 13-15).

2.1.3. Factores que originan el estrés

Según los doctores Comín, De la Fuente y Gracia (2009), existen varios factores causantes de la aparición del estrés en las personas, estos se pueden agrupar en

tres grandes grupos: laborales, familiares y personales, a continuación se detalla cada uno de ellos:

- A) Factores laborales:** estos van desde como es el ambiente de trabajo, donde puede haber exceso de iluminación, mucho ruido o simplemente la temperatura sea muy alta o baja, también se incluye el propio desempeño del trabajo, que muchas veces es arduo, así como el salario percibido, la productividad exigida por los jefes e incluso las horas extra.
- B) Factores familiares:** incluyen las relaciones conyugales conflictivas, malas relaciones con los hijos, convivencia con ancianos, cuidado de familiares enfermos así como convivir con personas desempleadas.
- C) Factores personales:** los cuales dependen del patrón de conducta de la persona, existen dos tipos: la personalidad tipo A y la personalidad tipo B:
 - Patrón de conducta tipo A: se presenta en personas que “necesitan sentir que poseen el control de todas las tareas en las que están presentes, son competitivos, agresivos, muy ambiciosos e irritables y están en alerta constante” (p. 16). Las personas con este patrón de conducta son ambiciosas y les gusta ser los protagonistas. Por lo tanto estas personas tienden a estresarse más fácilmente e incluso son más propensas a sufrir problemas cardíacos en proporción de seis a uno en comparación con una persona normal.
 - Patrón de conducta tipo B: se presenta en “personas que no son competitivas, poco ambiciosas, no agresivas, prefieren el trabajo en equipo, delegan responsabilidades, no les gusta vivir pendientes de horarios y disponen a menudo de su tiempo libre” (p. 16). Personas con este patrón de conducta tienden a estresarse menos.

Como puede notarse el estrés está presente siempre en la vida de las personas, y los factores que lo originan son varios, por ello es necesario saber convivir con el estrés y así sufrir menos los problemas que causa.

2.1.4. Medios emocionales para prevenir y reducir el estrés

Los medios emocionales positivos para prevenir y reducir el estrés son aquellos que lo afectan de manera inversa, es decir que los efectos de estos medios positivos en

las personas harán que sus niveles de estrés disminuyan, estos medios positivos funcionan de manera preventiva o reguladora ante la presencia del estrés, algunos de estos medios positivos se listan y detallan a continuación:

- **Entretenimiento:** es un medio positivo cuyo efecto en las personas hace que el nivel de estrés se reduzca, ya que el entretenimiento permite a las personas relajar su sistema nervioso autónomo, así lo afirman Russoniello, O'Brien, & Parks (2009), donde encontraron que la recreación o entretenimiento estaba relacionado con la disminución de cortisol, hormona que indica la presencia del estrés. En ese mismo sentido se demostró que uno de los principales efectos del entretenimiento en la salud es la reducción del estrés (Bryant & Vorderer, 2011). Para lograr entretener a una persona se debe captar su atención (Pastor, 2010) mediante actividades divertidas y placenteras que pueden ser jugar videojuegos, ir al cine, hacer deporte, etc.
- **Distracción:** es un medio positivo cuyo efecto aumenta la distracción u olvido de los problemas (Davis, 2008), permite la reducción de los niveles de estrés ya que se libera de pensamientos sobre problemas que causan mal estar mediante envío de señales al cerebro para que genere estrés y así afrontar esos problemas; la distracción mental tiene beneficios que permiten entrar en un estado mental positivo, junto con la sensación de equilibrio y relajación emocional (Russoniello et. al., 2009).
- **Vigor:** es un medio o fuerza interna positiva cuya presencia en las personas implica que hay menos presencia de estrés, ya que si se logra estabilizar o equilibrar el nivel de cortisol se logrará equilibrar el estado de ánimo óptimo para la salud (Talbot, 2012). "Así mismo, mantener la condición y el vigor del cuerpo son habilidades vitales que nos ayudan a manejar el estrés" (Schmitz & Hipp, 2005, p. 97). Este indicador está ligado a las emociones percibidas por la persona y es una subescala del POMS (Perfil de Estados de Ánimo) (Russoniello et. al., 2009).
- **Bienestar a corto plazo:** es un medio o estado emocional positivo por el cual se puede reducir el nivel de estrés de las personas, el ocio y el empleo del tiempo libre ayudan a aumentar el bienestar (Paris y Omar, 2013), en ese sentido medios como jugar un videojuego hacen que una persona disfrute de emociones que surgen durante la resolución de los diferentes objetivos que se presentan

en el videojuego, ello hace que las personas rejuvenezcan su mente y alivien su estrés (Artigames, 2014).

- Bienestar a largo plazo: es un medio o estado emocional positivo por el cual se puede reducir el nivel de estrés de las personas; hacer alguna actividad recreativa o placentera durante el tiempo de ocio hace que se reduzca el estrés considerablemente puesto que aumenta el bienestar a largo plazo (Paris y Omar, 2013), así también lo afirma la empresa Peruana de videojuegos Artigames en su portal de noticias, donde corrobora que:

Jugar un videojuego es el método más efectivo de descargar todo el estrés que traes encima, ya sea por el trabajo o el estudio. Y esto es porque descargas mucha energía emocional que se plasma en los personajes mientras juegas. Lo que te produce al final un estado de calma y relajación (2014, Párrafo 4).

2.1.5. Manejo de síntomas para reducir el estrés

Por otro lado los síntomas negativos del estrés tienen la característica de que al ser reducidos harán que los niveles de estrés disminuyan, estos pueden ser síntomas psicológicos, fisiológicos, cognitivos y conductuales (The American Institute of Stress, 2015; López, 2013), estos síntomas negativos forman parte del POMS (Perfil de Estados de Ánimo), el cual fue diseñado por el jefe del departamento de Psicología y laboratorio clínico de Psicofarmacología de la Universidad de Boston, Douglas McNair en 1971 con la finalidad de medir los sentimientos y emociones de las personas (Brianmac, 2015); reducir estos síntomas negativos permite regular la presencia del estrés, algunos de estos síntomas negativos se listan y detallan a continuación:

- Tensión: la tensión es un síntoma negativo generado por el estrés (WebConsultas, 2013b), así que cuando más estrés hay en una persona más tensión experimenta, por ello para reducir el estrés se debe reducir la tensión mediante el relajamiento y liberación emocional (Davis, 2008) que puede incluir varias maneras entre ellas la musicoterapia (Universidad de Nevada, 2014). La tensión también es una subescala del Perfil de Estados de Ánimo.
- Depresión: la depresión es un síntoma negativo generado por el estrés (WebConsultas, 2013b; López, 2013), así que cuando más estrés hay en una

persona más depresión experimenta, por ello para reducir el estrés se debe reducir la depresión mediante la risa, la alegría y el buen sentido del humor (Mayo Clinic, 2013; Davis, 2008). La depresión también es una subescala del Perfil de Estados de Ánimo.

- Ira: la ira es un síntoma negativo generado por el estrés (López, 2013), así que cuando más estrés hay en una persona más ira experimenta (Black, 2010), por ello para reducir el estrés se debe reducir la ira mediante la tranquilidad (Spielberger y Deffenbacher, 2015), y eso se puede lograr por medio de jugar con tu teléfono celular y así tranquilizarse ante situaciones estresantes, tales como hacer cola en un banco o estar atrapado en medio del tráfico. La ira también es una subescala del Perfil de Estados de Ánimo.
- Fatiga: la fatiga es un síntoma negativo generado por el estrés (WebConsultas, 2013b; Safer Healthcare, 2012), así que cuando más estrés hay en una persona más fatiga experimenta, por ello para reducir el estrés se debe reducir la fatiga mediante la eliminación de actividades que son agotadoras o desahogándose de ellas mediante algún medio recreativo o de entretenimiento (Davis, 2008). La fatiga también es una subescala del Perfil de Estados de Ánimo.
- Confusión: la confusión es un síntoma negativo generado por el estrés (WebConsultas, 2013b), así que cuando más estrés hay en una persona más confusión experimenta, especialmente cuando el estrés se va haciendo crónico ya que el cortisol afecta al hipocampo y genera fallas en la memoria (Loizte, 2004), por ello para reducir el estrés se debe reducir la confusión para que las personas tengan una mente más lucida y clara, y así puedan resolver los problemas que les aquejan y de esta manera puedan reducir su nivel de estrés. La confusión también es una subescala del Perfil de Estados de Ánimo.

2.1.6. Tratamientos para el estrés

Existen dos maneras de tratar el estrés, entre ellas se tiene los tratamientos farmacológicos y no farmacológicos, los cuales se detallan a continuación:

A. Tratamientos farmacológicos

Estos a su vez pueden ser tratamientos especializados como Homeoterapia, que usa medicamentos homeopáticos, tanto para prevenir como aliviar trastornos

emocionales; otro tratamiento es la Fitoterapia, que usa especies de origen vegetal para la prevención y tratamiento del estrés; por último se tiene el tratamiento con Medicamentos, que usa benzodiazepinas, que son fármacos con efecto hipnótico, antidepresivos, anti convulsionantes y relajantes musculares, que generalmente contienen drogas (Boiron, 2009).

Los dos primeros tratamientos conllevan pocos efectos secundarios leves, sin embargo el tratamiento con benzodiazepinas, tiene efectos secundarios que pueden ser muy perjudiciales para la salud e incluso generar dependencia, aunque esto varía de persona en persona (Hordé, 2014).

B. Tratamientos no farmacológicos

Es la manera más segura de tratar el estrés sin riesgos que incluyen los medicamentos, se pueden listar las siguientes maneras o terapias para reducir el estrés:

- **Risoterapia:** a corto plazo aligera la carga mental, estimula a los pulmones y al corazón porque permite la entrada de oxígeno y hace que el cerebro libere endorfinas, reduce el estrés y calma la tensión; mientras que a largo plazo, mejora el sistema inmunológico porque ayuda a liberar neuropeptidos que combaten el estrés y otras enfermedades, también alivia el dolor, ayuda a tener una mentalidad que permite ver los problemas como cosas que si tienen solución y además mejora el estado de ánimo, (Mayo Clinic, 2013). Esta terapia es divertida y fácil de realizar porque la risa es contagiosa, ya que esta “demostrado que cuando observamos una cara sonriente se activa un grupo de células nerviosas llamadas neuronas espejo, que nos impulsan a sonreír” (La Nación, 2006).
- **Musicoterapia:** donde la música adecuada ayuda a relajarse, a estar más positivo y reducir el estrés, incluso médicos de la Universidad de Stanford descubrieron que la música tiene los mismos efectos que los medicamentos, ello porque se puede inducir al cerebro a diferentes estados mediante la frecuencia que posee la música (Universidad de Nevada, 2014).
- **Psicoterapia:** que básicamente consiste en visitar a un psicólogo hasta sentir que el estrés ha sido reducido lo suficiente como para sentirse bien (Lifshitz, 2014).

- Técnicas de relajación: que ayudan a liberar la tensión, relajar el corazón, respirar con tranquilidad, descansar la mente y así reducir el estrés. Incluso se puede practicar Meditación, Joga y Taichi. (Lifshitz, 2014).
- Cambio de enfoque: consiste en dejar de pensar en preocupaciones, problemas, y en lugar de ello pensar en cosas positivas, así lo manifiesta el doctor Ignacio Morgado de la Universidad de Autónoma de Barcelona, donde afirma que los pensamientos negativos hacen que el cerebro reaccione y produzca cambios fisiológicos para afrontar dichas preocupaciones, sin embargo los pensamientos positivos hacen que en cerebro se sienta tranquilo y por ende mejore el funcionamiento del sistema inmunológico y también se reduzca el estrés (LaFamilia.info, 2013).

También se tiene otras actividades como hacer cosas que a una persona le resulten entretenidas, salir a pasear, pasar tiempo con mascotas, buenos amigos, hacer ejercicio e incluso dormir adecuadamente.

2.2. COEFICIENTE ALFA DE CRONBACH

2.2.1. Definición

Según la Universidad Nacional de Colombia “el coeficiente α fue propuesto en 1951 por Cronbach como un estadístico para estimar la confiabilidad de una prueba, o de cualquier compuesto obtenido a partir de la suma de varias mediciones” (Cervantes, 2005, p.9). Además el alfa de Cronbach permite “estimar la fiabilidad de un instrumento de medida a través de un conjunto de ítems que se espera que midan el mismo constructo” (Frías, 2014, p.2).

De acuerdo con Corral (2009), el alfa de Cronbach se usa para evaluar la homogeneidad de las preguntas o ítems que forman parte de escalas tipo Likert (González y Pazmiño, 2015) con respuestas politómicas.

El alfa de Cronbach mide la consistencia interna de los ítems de escalas, cuestionarios, índices, inventarios y pruebas (Oviedo y Campo-Arias, 2005, 573), es decir el grado en que estos ítems están correlacionados y miden el mismo constructo o instrumento (Campo-Arias y Oviedo, 2008; Oviedo y Campo-Arias, 2005; González y Pazmiño, 2015). Además el alfa de Cronbach generalmente se usa para

medir escalas cuyos ítems ofrecen respuestas politómicas, es decir de varias opciones.

Los valores del alfa de Cronbach oscilan entre 0 y 1, donde 0 indica ausencia de correlación entre los ítems y 1 indica una correlación perfecta (Campo-Arias y Oviedo, 2008); una de las fórmulas para calcular el coeficiente alfa de Cronbach es:

Alfa de Cronbach $\alpha = k/(k - 1) \times [1 - (\sum oi^2 / oT^2)]$, donde:

k , numero de ítems.

oi , varianza del ítem.

oT , varianza total de la escala.

2.2.2. Características

Para realizar el cálculo del coeficiente alfa de Cronbach en una escala se deben considerar las siguientes características:

- Calcular el alfa de Cronbach para cada una de las partes o secciones en las que este dividido el test (Mendoza, 2011). De esta manera se asegura que se tiene una medición de la confiabilidad adecuada para todo el test al medir todos los subgrupos que lo componen.
- Para las escalas que agrupan ítems con características diferentes (Palella, y Martins, 2006) o que tiendan a formar subescalas dentro de una escala, es recomendable calcular el alfa de Cronbach para cada sub escala por separado, así se garantiza el correcto cálculo de la consistencia interna (Oviedo y Campo-Arias, 2005).
- El valor del alfa de Cronbach aumenta a medida que aumenta el número de ítems o preguntas que conforman el test, por ello cuando una escala posee más de 20 ítems el valor del alfa de Cronbach tiende a superar el 0.9 (Cervantes, 2005).
- La cantidad adecuada de ítems que conforman una escala debe estar entre 3 y 20, otra cantidad fuera de este rango hace que el uso del alfa de Cronbach pierda su precisión del cálculo de correlación (Oviedo y Campo-Arias, 2005).
- El número de personas o encuestar o tamaño de la muestra debe estar preferentemente en proporción 5 a 1, es decir se debe encuestar a 5 personas

por cada ítem que posea el test o prueba (Herrán y Ardilla, 2007; Campo-Arias y Oviedo, 2008).

- Cuando el valor del alfa de Cronbach se encuentra entre 0.7 y 0.9 se considera que existe una consistencia interna del test aceptable (Campo-Arias y Oviedo, 2008).
- De acuerdo con Corral (2009), valores entre 0.81 y 1.00 representan una confiabilidad *muy alta*, entre 0.61 y 0.80, representa una confiabilidad *alta*, entre 0.41 y 0.60, representa una confiabilidad *moderada*, entre 0.21 y 0.40, una confiabilidad *baja* y entre 0.01 y 0.20 una confiabilidad del instrumento *muy baja*.
- De acuerdo con la Universidad de Valencia “es muy importante que todos los ítems estén puntuados en el mismo sentido y nunca tener ítems en sentido directo e inverso en un mismo análisis de fiabilidad pues daría valores de alfa de Cronbach bajos e incluso negativos” (Frías, 2014, p.4). Para lograr ello se puede invertir las puntuaciones de las respuestas, por ejemplo cambiar el 1, 2, 3, 4 por el 4, 3, 2, 1 y realizar el cálculo del alfa de Cronbach.

2.3. VIDEOJUEGOS

2.3.1. Definición

Un videojuego o juego de video:

“Es un software creado para el entretenimiento en general y basado en la interacción de una a varias personas por medio de un controlador y un aparato electrónico que ejecuta dicho videojuego. En muchos casos, los videojuegos recrean entornos y situaciones virtuales en los que el jugador puede controlar uno o varios personajes (o cualquier otro elemento de dicho entorno) para conseguir objetivos por medio de reglas determinadas (Quispe, 2013, p. 9).

Los videojuegos son aquellos programas desarrollados con la finalidad de capturar la atención de los usuarios para brindarles entretenimiento y permitirles desarrollar diversas habilidades con el uso moderado de los mismos.

2.3.2. Videojuegos como entretenimiento

Para García S. (2015) los videojuegos:

Suponen una nueva forma de concebir el entretenimiento, en un "entorno" de gran atractivo visual donde el usuario pasa de mero espectador a protagonista de la acción. La imagen, el sonido e incluso las sensaciones pasan a ser "interactivas" y cobran vida en este nuevo "mundo digital" (Párrafo 5).

Así mismo García S. (2015) menciona que los videojuegos además de ser un medio de entretenimiento sano permiten mejorar las habilidades psicomotrices de los niños y también sus reflejos.

Por ello los videojuegos deben ser considerados como medios de entretenimiento tan iguales como ir al cine o ver televisión, todo ello siempre en un contexto controlado por los padres.

Es importante mencionar el impacto que tiene el uso de videojuegos en la vida de las personas ya que existen efectos beneficiosos y también negativos los cuales se presentan a continuación:

A) Efectos positivos de los videojuegos

La investigación realizada por Tejeiro, Pelegrina del Rio y Gómez (2009) muestra beneficios como:

- Entrenamiento y mejora de habilidades tales como mejor atención visual, mayor rapidez de búsqueda visual y mejor ejecución de tareas de atención dividida.
- Tratamiento médico y psicológico, así como en el campo de la rehabilitación fisioterapeuta, también mejorando la comunicación entre niños autistas y no autistas, siendo una de las más importantes la distracción para la reducción del dolor ocasionado por diversas lesiones.
- Método didáctico educativo, al combinar la lúdica propia de los videojuegos con la función de enseñanza pedagógica porque logran captar el interés de los niños y los mantienen motivados, así el docente logra aplicar las estrategias pedagógicas pertinentes.

Así también el portal de noticias CNNEXPANSION (2008), publicó los resultados de la investigación que se realizaron en la Universidad del Este de California (ECU)

durante seis meses donde el Doctor Carmen Russoniello afirmó que el estudio realizado será es uno de los pilares para incluir como receta médica los videojuegos.

El Doctor Carmen Russoniello dijo que los videojuegos casuales tienen la capacidad de prevenir y tratar el estrés y la conducta mejorando así la salud emocional y fisiológica, en esto concluyó Russoniello luego de ver como los videojuegos casuales como Bejeweled 2, Peggle y Bookworm Adventures que fueron usados en la investigación lograron reducir los niveles de estrés y la tensión psicológica, siendo el juego Bejeweled 2 quien redujo mayor nivel de estrés el cual fue de un 54%, mientras que Peggle y Bookworm Adeventures tuvieron más éxito en la mejora de estados de ánimo.

La investigación realizada por Russoniello et. al. (2009), involucró a 143 personas entre hombres y mujeres, estos fueron separados en un grupo de control y un grupo experimental, el grupo de control realizó búsquedas académicas en internet durante 20 minutos y el grupo experimental jugó los videojuegos durante 20 minutos también, a ambos grupos se les hizo llenar la encuesta de Perfil de Estados de Ánimo, y también se les realizó mediciones avanzadas usando el Electroencefalograma así como también se les midió la Variabilidad de Frecuencia Cardiaca. Los resultados demostraron la eficacia de los videojuegos para reducir el estrés y mejorar el estado de ánimo o conducta de las personas.

Las siguientes imágenes muestran dichos videojuegos usados en el estudio:



Figura 3. Videojuego Bejeweled 2

Fuente: www.gonzagames.eu



Figura 4. Videojuego Peggle

Fuente: www.gonzagames.eu



Figura 5. Videojuego Bookworm Adventures

Fuente: www.popcap.com/bookworm-adventures

Las publicaciones de la página web de tecnología CEINTEC (2015), demuestran que hay usos médicos de los videojuegos en los que destacan los siguientes:

- En el año 2006 el centro de investigación tecnológica para la mejora de la salud HopeLab, evaluó 375 pacientes con cáncer entre los 13 y 29 años quienes usaron un videojuego llamado Re-Mission, donde el que el protagonista tenía que recorrer el cuerpo humano eliminando células cancerígenas, el videojuego está basado en disparos en tercera persona. Con ello el 80% de jóvenes que lo usaron mostraron mejoras en su salud y afrontaron de mejor manera el tratamiento de quimioterapia.

- Otro estudio realizado en el 2009 por la Dra. Emily Holmes del departamento de Psiquiatría de la Universidad de Oxford demostró que jugar el videojuego Tetris por 10 minutos es capaz de curar progresivamente el estrés postraumático ya que inhibe los recuerdos negativos que generan ansiedad, ello se basa en que el videojuego compite por los mismos recursos, es decir los sensores perceptivos de la persona, entonces la persona no puede enfocarse en los recuerdos negativos cuando juega porque tiene que prestar atención al videojuego, así el cerebro va eliminando los recuerdos negativos.
- Entre los años 2007 y 2010, los ingenieros, médicos y programadores dirigidos por el Dr. Fernando Fernández Aranda, jefe de la unidad de trastornos de la alimentación del hospital universitario de Bellvitge, y la Dra. Susana Jiménez Murcia, jefe de la unidad de juego patológico de ese mismo hospital desarrollaron el videojuego llamado Island, dicho videojuego funciona como complemento para el tratamiento de pacientes, y les permite crear estrategias y maneras de resistir a las emociones negativas estresantes en su vida así como lograr tener autocontrol para superar situaciones de estrés, dicho tratamiento dura de 12 a 14 semanas con sesiones de 20 minutos.
- Los videojuegos se pueden usar también en la rehabilitación de víctimas de accidentes cerebro vascular, con este entrenamiento pueden llegar a recuperar el movimiento y el habla. El hospital de parapléjicos de Toledo también usa esta herramienta desde el año 2008 para potenciar la agilidad, la movilidad y la coordinación de los movimientos de tronco y brazos de pacientes ayudados por una consola Wii.

El artículo publicado por Mercer (2015) de la Universidad del Estado de Bridgewater de los Estados Unidos describe los resultados obtenidos luego de aplicar videojuegos para reducir el estrés. Este estudio se realizó con seis personas, las personas uno y dos jugaron el videojuego llamado Scavenger Hunt, el cual consistía en buscar objetos ocultos, las personas tres y cuatro jugaron el juego Tetris de manera online, y las personas cinco y seis hicieron ejercicios de relajación al aire libre. Los resultados afirman que los videojuegos reducen el estrés ya que las personas que jugaron dichos videojuegos mencionaron que se sentían desestresados, además reportaron reducción en su presión y pulso sanguíneo.

De manera similar Serrone (2012), en su tesis afirma que jugar videojuegos violentos y no violentos ayuda a reducir el estrés, la frustración así como mejorar el autocontrol y entusiasmo.

“Cuando se trata de lidiar con el estrés o la frustración, los videojuegos pueden ser herramientas eficaces para aliviar el estrés, siempre y cuando el jugador sea capaz de sentirse competente, centrado, divertido e inmerso en el juego” (Serrone, 2012, p.87).

Además “después de una experiencia negativa o estresante, una experiencia positiva en una sesión de videojuegos podría reparar el estrés, aumentar el estado de ánimo y aumentar la sensación de control o dominación” (Serrone, 2012, p.87).

Entre otros beneficios se tiene el entrenamiento para médicos, según Karam (2005) que indica que se realizó pruebas donde médicos cirujanos que jugaron videojuegos con joysticks cometían menos errores que aquellos médicos que no los jugaban, esto a la hora de realizar una intervención quirúrgica donde predomina la coordinación entre el ojo y la mano humana.

B) Efectos negativos de los videojuegos

Los efectos negativos de los videojuegos según diversos autores se listan a continuación:

Según Karam (2005) pueden volverse adictivos en los niños y púberes si no existe un control adecuado por parte de los padres o de los mismos jugadores.

Pueden ser causa de conducta agresiva en los niños, aunque ello no esté totalmente demostrado (Tejeiro et. al. 2009). Sin embargo esto está más relacionado con los juegos de tipo acción donde intervienen situaciones de guerra y matanza; mientras que los juegos casuales son totalmente opuestos a ese tipo de videojuegos agresivos.

Tejeiro et. al. (2009) en su investigación aduce que:

No existe relación estadística entre uso de videojuegos y número de amigos del mismo sexo, número de amigos de sexo opuesto, número de integrantes del círculo social, clima social, frecuencia de las actividades habituales, frecuencia de las interacciones sociales, número de parejas, retraimiento social, timidez, hipersensibilidad, soledad, popularidad, estatus social,

sociabilidad, amigabilidad, visitas a casa de los amigos, recepción de visitas de amigos en el propio domicilio, popularidad percibida en la escuela, y ajuste social; la práctica de videojuegos se asocia a una mayor frecuencia de encuentro con amigos fuera de la escuela, mayor interacción familiar, mayor número de visitas al domicilio de familiares y amigos, mayor número de amigos y mayor disposición a hablar (p. 240).

Por ello se puede ver que el aislamiento social como efecto negativo de los videojuegos queda bastante reducido.

En relación al rendimiento escolar Tejeiro et. al. (2009) menciona que una causa por la que los estudiantes que juegan videojuegos tiendan a disminuir su rendimiento es porque dichos alumnos presentan absentismo o inasistencia a clases y en lugar de ello se ponen a jugar ya sea en cabinas públicas o en su propio domicilio.

2.3.3. Géneros de videojuegos

Con el paso del tiempo han ido apareciendo videojuegos cada vez más modernos, es por ello que muchos de estos videojuegos requieren un nuevo género para clasificarlos e incluso hay algunos que pertenecen a varios géneros, como es el caso del videojuego GTA San Andreas, el cual puede ser clasificado como juego de rol, simulación e incluso disparos. Según la enciclopedia en línea cubana EcuRed (2010) y el diccionario online de términos sobre videojuegos GamerDic (2013), los géneros y subgéneros de los videojuegos más representativos son los siguientes:

- a)** Aventura: en este género de videojuegos el personaje debe avanzar por distintos escenarios 2D o 3D, interactuando con otros personajes que pueden ser aliados o enemigos así como con objetos que pueden ser puntos de vida o monedas. El objetivo es superar los niveles al vencer a un personaje jefe o enemigo que hace el rol de malo del videojuego, para llegar a destruir al jefe el personaje debe superar diversos obstáculos y situaciones que requieren usar el razonamiento.
- b)** Disparos: videojuegos de este género se caracterizan porque se desarrollan en entornos 3D, en donde el objetivo es matar a otros personajes usando armas de destrucción y así ganar las partidas. Este género se popularizó gracias al internet que permitió que se pueda jugar en tiempo real con otras personas. Algunos subgéneros son: FPS (disparos en primera persona), disparos en tercera persona, multijugador online, mata marcianos y otros.

- c)** Educativos: son videojuegos que pueden pertenecer a cualquier género, tienen el objetivo de dar conocimientos a los jugadores, ya sea sobre temas de matemáticas o comunicación, medio ambiente o historia, idiomas, etc., estos videojuegos pueden jugados incluso por niños de 2 años de edad y ya vienen siendo incorporados por instituciones educativas para complementar el aprendizaje de los estudiantes.
- d)** Estrategia: son videojuegos donde el jugador tiene que administrar recursos como personajes y objetos dentro de la temática económica y social, sin embargo la introducción de la temática de lucha hizo que este género se popularice y adquiera más jugadores, este género puede ser jugado por turnos o en tiempo real como el subgénero RTS (Estrategia en Tiempo Real), donde la capacidad de imaginación y táctica del jugador le permitirá ganar las batallas. Uno de sus principales subgéneros es el MOBA (Arena de Combate Multijugador Online) donde el jugador forma equipo con otros jugadores para enfrentarse a un equipo oponente dentro de mapas simétricos.
- e)** Lucha: conocido también como acción, el objetivo es derrotar al oponente, para ello se usan puños, patadas, cabezazos e incluso armas como cuchillos, piedras y otros. Pueden ser jugados en modo uno contra uno o todos contra todos, no incluye los videojuegos de boxeo ya que estos son de género deportivo. Los videojuegos de lucha tienen muchos seguidores en todo el mundo.
- f)** Plataformas: son videojuegos en 2D y también ya disponibles en 3D gracias al avance tecnológico, el objetivo es superar niveles, para ello se deben atravesar escenarios realizando desplazamientos a la derecha e izquierda además de saltos, durante el recorrido se deben recoger monedas y otros objetos para completar el nivel.
- g)** Rol: también conocido como RPG (Juego de Rol), en este tipo de videojuegos el personaje se centra en mejorar sus habilidades a medida que se relaciona con otros personajes y objetos, además las decisiones del jugador influirán en el futuro el personaje, se pueden jugar de manera online y por turnos. Tiene varios subgéneros como MMO (Juego Multijugador Masivo en Línea) y MMORPG (Juego de Rol Multijugador Masivo en Línea), donde concurren miles de jugadores que compiten entre ellos o realizan misiones.

- h)** Simulación: este género de videojuegos no posee un objetivo principal más allá de simular entornos como manejar un carro, un avión, un tren o simular la gestión de un zoológico o una ciudad. Estos videojuegos deben ser capaces de acercarse lo más que puedan al realismo para que el jugador se sienta atraído por el videojuego.
- i)** Deportivos: se basan en deportes populares como fútbol, vóley, básquet, golf, tenis, box, etc. Donde el objetivo es ganar las partidas, se puede jugar de manera individual o colectiva e incluso de manera online. Muchos de estos videojuegos son lanzados cuando se llevan grandes eventos deportivos a nivel mundial para aprovechar la publicidad.
- j)** Carreras: videojuego basados en concursos de velocidad donde inicialmente el objetivo era ganar la carrera usando carros, naves, lanchas, motos, etc., Sin embargo con el avance de la tecnología dichas competiciones adquieren una mayor temática donde ahora se basa incluso en hacer puntuaciones altas y competir de manera online tanto en entornos 2D y 3D.

También existen los videojuegos de género casual, la International Game Developers Association (2008), define a los videojuegos casuales como juegos de video desarrollados para la mayoría de consumidores, incluso para aquellas personas que no se consideran jugadores. Los videojuegos casuales pueden ser jugados por niños, abuelos, profesionales y amas de casa que se encuentren en cualquier lugar, pudiendo usar para ello portales web, consolas y teléfonos.

Así mismo la IGDA (2008) menciona las siguientes características de los videojuegos casuales:

- Son fáciles de conseguirlos y fáciles de aprender a jugarlos.
- Tienen controles fáciles, por ejemplo la mayoría de los videojuegos casuales de computadora solo necesitan el click izquierdo para jugarlos.
- El gameplay es adictivo, además se juegan en pequeñas sesiones de 5 a 20 minutos ya sea antes o después del almuerzo o la cena.
- No aplican penalidades inmediatas cuando el jugador comete errores.
- Diseñados para aumentar la complejidad a medida que se avanza en el juego.

- Diseñados para ser inclusivos y no exclusivos, no suelen presentar machismo, feminismo o violencia.
- También diseñados para divertir y relajar, y no generar adrenalina o emociones fuertes como lo hacen los videojuegos hardcore.
- En cuanto a los costos de producción, los juegos casuales son más baratos que los juegos hardcore, pero poco a poco esta diferencia se va recortando.

Una característica fundamental para el diseño de videojuegos de género casual definida por el Profesor de Psicología de Hungría Mihaly Csikszentmihalyi es que estos deben mantener el equilibrio entre la dificultad y habilidad del jugador, es decir el videojuego casual no debe ser muy fácil como para hacer el jugador se aburra ni muy difícil como para hacer que el jugador abandone el videojuego (Dolci, 2014).

Los videojuegos casuales pueden pertenecer a alguna de los siguientes subgéneros:

Tabla 2. *Subgéneros del videojuego casual*

Subgénero de videojuego casual	Ejemplo
Puzzles	Brain Teaser, Bejeweled series, Luxor series y Marble Popper.
Objetos ocultos	Mystery Case Files series, Mortimer Beckett series, Hidden Expedition series.
Palabras, vocabulario y diccionario	Bookworm, Bookworm Adventures series y Bonnie's Bookstore.
Arcade	Plants vs. Zombies, Peggle y Feeding Frenzy
Cartas	Slingo Quest, Lottso! Deluxe y Luxor Mahjong.

Fuente: <http://www.bigfishgames.com/blog/the-next-casual-game-genre/>

Se puede observar que existe una buena variedad de géneros de videojuegos lo cual le permite llegar a una gran cantidad de público, ya sea desde niños, jóvenes y adultos, porque cada persona en cada etapa de su vida tiene inclinación a diversas preferencias.

2.3.4. Desarrollo de videojuegos

Desarrollar un videojuego consiste en la creación de este desde su inicio, donde nace la idea del videojuego hasta su producción final, donde el juego ya está terminado, utilizando para ello diversas metodologías de desarrollo, además:

Vallejo (2014) afirmó que en la mayoría de los casos el desarrollo es llevado a cabo por estudios, que pueden ser grandes productoras o pequeños estudios independientes, algunas veces los videojuegos pueden ser desarrollados por una sola persona, un fenómeno que era muy común en los inicios de la industria por allá en los años 80. En la actualidad los videojuegos con altos presupuestos son conocidos como AAA y aquellos que son desarrollados por pequeños estudios y bajo presupuesto son conocidos como Juegos Indie (Independientes). (Párrafo 1).

Asimismo Vallejo (2014) menciona que el desarrollo de videojuegos generalmente requiere de un publiser o patrocinador que se encargara de financiar el videojuego y publicarlo. Desarrollar un videojuego puede tardar desde unos pocos meses hasta varios años y pueden ser distribuidos en diversas plataformas tales como para dispositivos móviles, computadoras y demás.

Según un artículo virtual del diario Peru21 (2013) hoy en día el desarrollo de los juegos es una gran alternativa económica ya que:

La industria del videojuego es millonaria. Según una estimación de la empresa Business Insights, las ventas de videojuegos serán de 76 mil millones de dólares para este año [2013]. Esta cifra indica que, sin duda, es un campo atractivo para aquellos que sienten que crear videojuegos es su destino profesional (Párrafo 1).

2.3.5. Recursos multimedia para el desarrollo

Para desarrollar videojuegos se requieren de dos componentes multimedia principales como son el audio o sonido digital y los gráficos o imágenes digitales.

Audio digital

El sonido o audio digital posee las siguientes características (Ministerio de Educación de España, 2008a):

Frecuencia: es el número de vibraciones por segundo que generan al sonido analógico, su unidad de medida es el Hertzio (Hz).

Frecuencia de muestreo: es la frecuencia en que se toma el audio analógico para generar el audio digital que es una secuencia de ceros y unos, por ejemplo si se toma una frecuencia de muestreo de 22050 Hz, significa que se tomarán 22050

muestras del sonido analógico para generar el sonido digital, usualmente se usan frecuencias de muestreo de 11025Hz, 22050 Hz y 44100Hz, donde a mayor frecuencia de muestreo se tiene mejor calidad.

Resolución: conocido también como resolución de bit, es el número de bits que se necesitan para almacenar una muestra analógica, por ejemplo 8 bits equivalen a tener 2 elevado a la 8, 256 niveles de amplitud, mientras mayor sea la resolución, mayor calidad tendrá el audio digital.

Velocidad de transmisión: conocido también como bitrate, es la cantidad de espacio de almacenamiento que ocupa un segundo del audio, generalmente se usa bitrates de 128kbps, mientras mayor sea el bitrate mayor calidad tendrá el audio y más espacio ocupará en memoria.

Modo: son dos modos, mono estéreo y estéreo, el primero solo usa una pista, es decir fue grabado usando solo un micrófono, mientras que el estéreo fue grabado usando dos micrófonos y por lo tanto tiene dos pistas, el modo estéreo pesa casi el doble que el modo mono estéreo.

CBR/VBR: significan bitrate constante y bitrate variable, esta característica la adquiere un audio cuando es codificado.

Códec: es un algoritmo que codifica y decodifica y que reduce en número de bytes que ocupa un archivo de audio, uno de los códec más famosos es el MP3.

Decibelio: es la medida del volumen o intensidad del sonido.

Imagen digital

Las características de la imagen o gráfico son (Ministerio de Educación de España, 2008b):

Pixel: es la mínima unidad de visualización que posee una imagen digital.

Profundidad de color: se mide en bits y significa qué cantidad de colores tendrá una imagen para ser representada, es decir con 1 bit se tiene 1 color, con 2 bits se tiene 4, con 3 bits se tiene 8 colores y así sucesivamente, estos colores se aplican pixel por pixel, mientras mayor profundidad, mayor calidad de imagen.

Modo de color: existen cuatro modos, el primero es el monocromático, que solo usa 1 bit de profundidad con los colores blanco y negro, el segundo es el modo escala de grises, que usa 256 tonos de gris entre blanco y negro, el tercero es el modo

color indexado, que utiliza una profundidad de 8 bits con 256 colores y el cuarto es el modo RGB, que utiliza 3 canales: Rojo, Verde y Azul (colores primarios), donde cada uno tiene una intensidad de 0 a 255, por lo tanto con dicha combinación se puede obtener 16.7 millones de colores, mientras se tenga más valores de intensidad mayor será la calidad de la imagen digital y tendrá más peso.

2.3.6. Desarrollo nativo vs multiplataforma

Desarrollo nativo

El desarrollo nativo de videojuegos y aplicaciones significa crear estos programas de manera exclusiva para cada plataforma, es decir programar específicamente para cada sistema operativo. Por ejemplo se debe usar el SDK de Android junto al lenguaje de programación Java para desarrollar aplicaciones para Android, se debe usar el IDE xCode junto al lenguaje Objective-C o Swift para desarrollar aplicaciones para iOS, se debe usar el IDE Visual Basic junto al XALM y el lenguaje C# para desarrollar aplicaciones para Windows Phone y así para los demás sistemas operativos móviles (CampusMPV, 2014).

Desarrollo multiplataforma

En cambio desarrollar videojuegos y aplicaciones multiplataforma significa programar una sola vez y realizar pequeñas configuraciones de interfaz y compilación para desarrollar aplicaciones para diferentes sistemas operativos móviles como Android, iOS, Windows Phone y otros. Existen muchas herramientas para este propósito, una de ellas es Xamarin, que usa el lenguaje de programación C# bajo el framework .NET de Microsoft que te permite desarrollar y compilar para Android, iOS y Windows Phone (CampusMPV, 2014).

También otra manera de desarrollar para varias plataformas, una de ellas es el desarrollo basado en HTML5, en donde se usa el framework PhoneGap y lenguajes HTML, CSS, y JavaScript, con ello se puede desarrollar aplicaciones que serán interpretadas dentro del teléfono inteligente con cualquier sistema operativo que tenga un hardware y navegador capaz de soportar HTML5 (CampusMPV, 2014).

Comparación

A continuación se lista las ventajas y desventajas del desarrollo nativo y multiplataforma:

Tabla 3: Comparación desarrollo nativo vs multiplataforma

	Ventajas	Desventajas
Desarrollo nativo	Permite que la aplicación tenga un excelente rendimiento cuando se ejecuta	Requiere aprender cada lenguaje de programación por plataforma
	La aplicación puede acceder al todo hardware	Existen paradigmas de programación propios de cada lenguaje de programación
		Mayor tiempo requerido para desarrollar y distribuir aplicaciones para diferentes plataformas
		Mayor cantidad de recursos requeridos para desarrollar y distribuir aplicaciones para diferentes plataformas
	Ventajas	Desventajas
Desarrollo multiplataforma	Ahorro de tiempo de desarrollo	Dependiendo de la herramienta y tecnología utilizada es posible que en algunos casos la aplicación no pueda acceder a algunos componentes de hardware
	Ahorro de los recursos para el desarrollo	Dependiendo de la herramienta y tecnología utilizada es posible que el rendimiento de la aplicación pueda no ser igual al de una aplicación nativa
	Aprender un solo lenguaje de programación	
	Reutilización de gran parte del código fuente	
	Realizar pequeñas configuraciones para desarrollar y distribuir para diferentes sistemas operativos	
	Permite llegar más rápido al mercado	

Fuente: <http://www.campusmv.es/recursos/post/Programacion-movil-Que-herramienta-y-lenguaje-elegir.aspx>

Por otra parte existen diferentes metodologías para desarrollar videojuegos, a continuación se verán algunas de las metodologías de desarrollo.

2.4. METODOLOGÍAS DE DESARROLLO DE VIDEOJUEGOS

2.4.1. SUM

Según la página web de SUM (2015), en la sección “Introducción a SUM” se define que:

La metodología SUM para videojuegos tiene como objetivos desarrollar videojuegos de calidad en tiempo y costo, así como la mejora continua del proceso para incrementar la eficacia y eficiencia de esta. Pretende obtener resultados predecibles, administrar eficientemente los recursos y riesgos del proyecto, y lograr una alta productividad del equipo de desarrollo (Párrafo 1).

Es importante mencionar además que:

SUM adapta para videojuegos la estructura y roles de Scrum. Se utiliza esta metodología ya que brinda flexibilidad para definir el ciclo de vida y puede ser combinada fácilmente con otras metodologías de desarrollo para adaptarse a distintas realidades. Para la adaptación se toma en cuenta la experiencia de las empresas de desarrollo de videojuegos que adaptan metodologías ágiles a nivel mundial (Párrafo 2).

Acerenza et. al. (2009) mencionan que la metodología SUM para el desarrollo de videojuegos permite mejorar la calidad, tiempo y costo así como la eficiencia y eficacia. La metodología SUM fue creada para funcionar en equipos pequeños de dos a siete personas, lo cual respalda el hecho de que esta metodología funciona mayormente para proyectos de corta duración.

Gestión de proyecto

La gestión del proyecto o plan de proyecto que integra la metodología SUM incluye la gestión de los siguientes planes y áreas (SUM, 2015):

A) Gestión del alcance

En esta área se indica qué se pretende lograr con el proyecto, es decir qué se realizará y de ser necesario qué no se realizará con la ejecución del proyecto, la especificación de esta área puede ser complementada con las fases de concepto y planificación de la metodología SUM.

B) Plan de personal

Dentro de este plan se debe indicar qué miembros de la organización serán los encargados de desarrollar el proyecto, además se debe especificar qué rol se le asignará a cada miembro así como el periodo en el que será requerida su participación.

C) Plan de recursos

Dentro de este plan se debe listar los costos externos requeridos para el desarrollo del proyecto, es decir costos como: contratistas externos, proveedores, licencias de software, licencias de distribución, etc.

Además se debe mencionar qué recursos ya fueron adquiridos o ya se tenían antes que inicie el proyecto, y en caso de que no hayan sido adquiridos, entonces deben ser especificados, ello puede incluir recursos como: computadoras, laptops, teléfonos, consolas, ambientes físicos, materiales, etc.

D) Plan presupuestal

Dentro de este plan se da un informe detallado de los recursos económicos necesarios para la realización del proyecto, lo cual es recomendable que se detalle indicando el presupuesto requerido por cada mes de desarrollo del proyecto para que de esta manera siempre se tenga a disposición el dinero y así el proyecto siga su curso normal, algunos costos relevantes que deben ser considerados son: pago del personal, adquisición de hardware, pago por licencias software, pago para contratistas externos, pago por uso de contenido con derechos de autor, publicidad, costos fijos (alquiler, viajes, cuentas), etc.

E) Cronograma e hitos

Dentro de este plan se incluye la duración de las iteraciones y tareas necesarias para desarrollar el proyecto, además se debe indicar qué hitos serán alcanzados luego de culminar estas iteraciones y tareas.

F) Seguimiento del proyecto

Dentro de esta área se detalla qué porcentaje de avance de las iteraciones y tareas que componen el desarrollo del proyecto fueron ya realizadas, así como los encargados de estas tareas e iteraciones.

De esta manera se asegura que el avance del proyecto esté dentro de lo establecido, es decir dentro del cronograma, también es útil para verificar si hay recursos que estén sobrecargados y así tomar las acciones correspondientes.

G) Riesgos

En este plan se deben describir los posibles riesgos que pueda afrontar el proyecto, también se debe describir que efectos podrían tener estos riesgos y la manera en que de presentarse deban ser mitigados, en caso de haber demasiados riesgos identificados se puede incluir desarrollar un plan de contingencia para asegurar la continuidad del desarrollo del proyecto.

Proceso de Entrega

La estructura general del proceso de entrega de la metodología SUM se presenta a continuación (SUM, 2015):

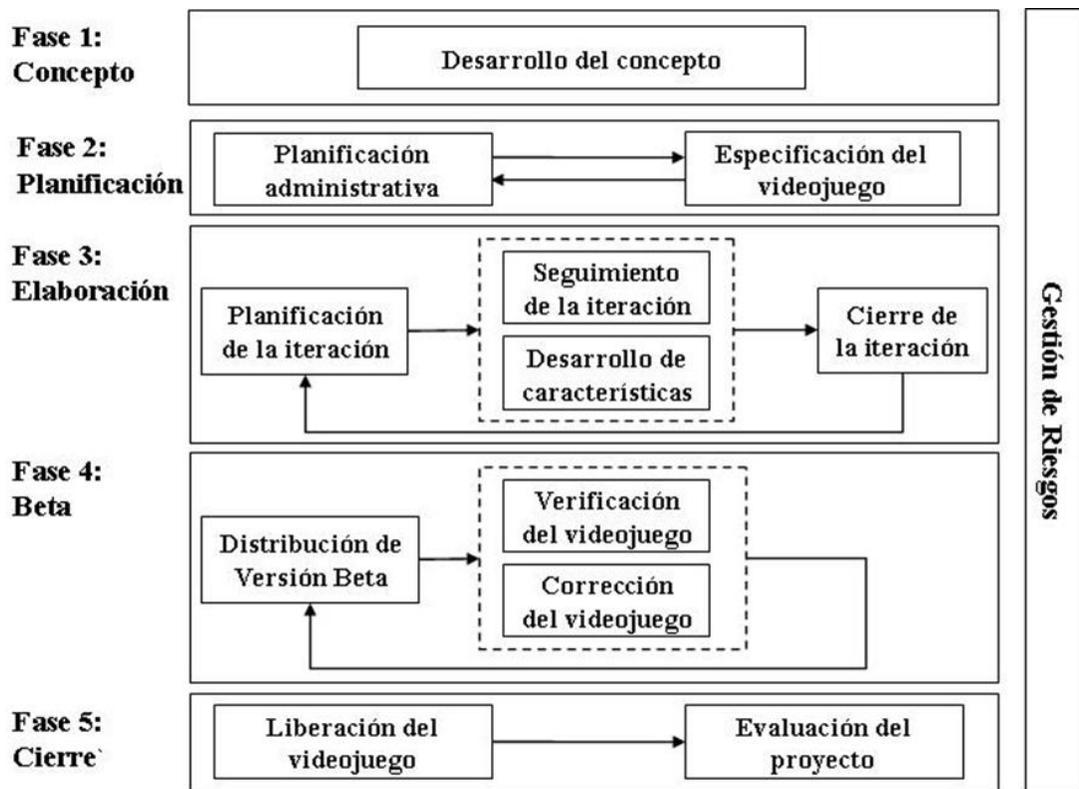


Figura 6: Fases del proceso de entrega de la metodología SUM

Fuente: <http://www.gemserk.com/sum/>

Estas cinco fases se detallan a continuación:

A) Concepto

Los encargados de esta fase son el productor interino, el cliente y el equipo de desarrollo, quienes realizan reuniones para definir el concepto del videojuego mediante propuestas que son refinadas. La fase de concepto incluye tres aspectos que son:

Negocio: donde se define el objetivo del proyecto, el público usuario y los posibles modelos de negocio.

Técnicos: donde se decide que herramientas se van a utilizar, que tecnologías y también para que plataformas se va a desarrollar el videojuego.

Videojuego: donde se define el gameplay, la historia, personajes, escenarios, y las características más importantes que tendrá el videojuego.

B) Planificación

Esta fase será aprobada cuando el cliente este de acuerdo con los planes y características establecidos. Esta fase es flexible puesto que de acuerdo con la fase de elaboración se puede modificar la fase de planificación para mostrar la situación real del proyecto.

Se realizan las actividades de planificación administrativa donde se planifica el resto de las fases del proyecto, también se realiza la actividad de especificar las características o requisitos que serán desarrollados en la fase de elaboración, la realización de estas actividades involucra actualizar el plan de proyecto, además estas actividades deben ser realizadas en paralelo para que exista mayor precisión en el desarrollo del proyecto.

C) Elaboración

Esta es la fase más extensa del proyecto, se realiza de manera iterativa e incremental, donde luego de cada iteración se obtiene una versión ejecutable del videojuego.

Trabajar de esta manera permite evaluar cuál es el avance del proyecto, y con ello se puede realizar las modificaciones necesarias para lograr cumplir los plazos establecidos en la fase de planificación.

Además trabajar de manera iterativa permite ganar experiencia y así mejorar el trabajo en las siguientes iteraciones.

D) Beta

Esta fase es realizada por una persona o equipo externo al equipo de desarrollo. El objetivo es que se pueda verificar que ciertos aspectos funcionen correctamente como se estableció en las fases de concepto y planificación, tales aspectos son: el gameplay, la diversión, curva de aprendizaje, curva de dificultad, etc. También esta fase se centra en detectar la mayor cantidad de errores que puedan existir en el videojuego.

Se realiza de manera iterativa, donde se verifica cada versión de del videojuego, la cual es revisada y se envían reportes con los errores encontrados (en caso que existan) para que sean corregidos. Adicionalmente se puede enviar otra versión beta para verificar que se realizaron las correcciones de manera correcta.

E) Cierre

Esta fase se compone de dos actividades:

Liberación del videojuego: que consiste en obtener la versión final videojuego para el cliente.

Evaluación del proyecto: donde se registran las lecciones aprendidas durante el proyecto y se da cierre a este.

Adicionalmente se cuenta con una fase que está presente en todas las demás fases, esta es la gestión de riesgos descrita a continuación:

F) Gestión de riesgo

Esta fase se realiza durante el desarrollo del proyecto, es decir está presente en cada una de las fases ya descritas, su realización puede ser registrada en el plan del proyecto donde se debe registrar cada una de los riesgos que puedan afectar el proyecto y de ser necesario se deba crear un plan de contingencia.

Roles

La metodología SUM incluye cuatro roles que son los siguientes (SUM, 2015):

A) Cliente

Es la persona encargada de: definir la visión y concepto del videojuego, validar dicho concepto, aprobar los planes del proyecto y establecer los hitos, realizar la priorización de las características del videojuego, priorizar la corrección de errores y evaluar el videojuego final y el proyecto.

B) Productor Interno

Se encarga de realizar funciones como: definir objetivos, planificar, ejecutar y realizar el seguimiento del proyecto, además establece el vínculo para la comunicación con el cliente, se asegura de mantener al equipo de desarrollo enfocado en su trabajo, y también realiza acciones para mejorar el proyecto.

C) Equipo de desarrollo

Está conformado por diseñadores, programadores, artistas gráficos y artistas sonoros, donde pueden haber de dos a siete personas, se encargan de apoyar en la definición del concepto de videojuego, además se encargan de realizar acciones para desarrollar el videojuego, tales acciones son: estimar, asignar, realizar y verificar el cumplimiento de las tareas.

También el equipo de desarrollo aporta soluciones en caso de haber posibles problemas.

D) Verificador beta

Es la persona o equipo externo al equipo de desarrollo, cuya función es verificar que el videojuego funcione de manera correcta conforme a lo establecido, este rol puede ser asumido por un experto en realizar revisiones de software o por un simple jugador o "gamer".

2.4.2. GUP (Game Unified Process)

Esta metodología para el desarrollo de videojuegos es una adaptación de la metodología para desarrollo de software RUP (Rational Unified Process), permite terminar el desarrollo del videojuego dentro del tiempo establecido (EcuRed, 2012).

Según la enciclopedia Cubana EcuRed (2012) la metodología GUP es:

Una combinación de dos metodologías de desarrollo utilizadas comúnmente en el software tradicional. La primera es la utilización del Rational Unified Process el cual plantea un diseño estricto y una documentación rigurosa de cada paso y funcionalidad a implementar. Y la metodología Extreme Programming con algunas variaciones para que pueda ser aplicada por personas de otras disciplinas. En este proceso los ciclos cortos ayudan a mantener la comunicación fluida entre equipos, y el componente artístico no se restringe tanto como cuando se utiliza el RUP, proporcionando así mayor capacidad comunicativa (Características, Párrafo 1).

Por ello se puede afirmar que el enorme crecimiento de la industria de los videojuegos hace que se desarrollen y adapten metodologías para su desarrollo, en este caso se combinan la metodología RUP y la metodología de Programación extrema dando así lugar a esta metodología GUP.

La siguiente imagen muestra la relación entre la metodología GUP y con la metodología RUP:

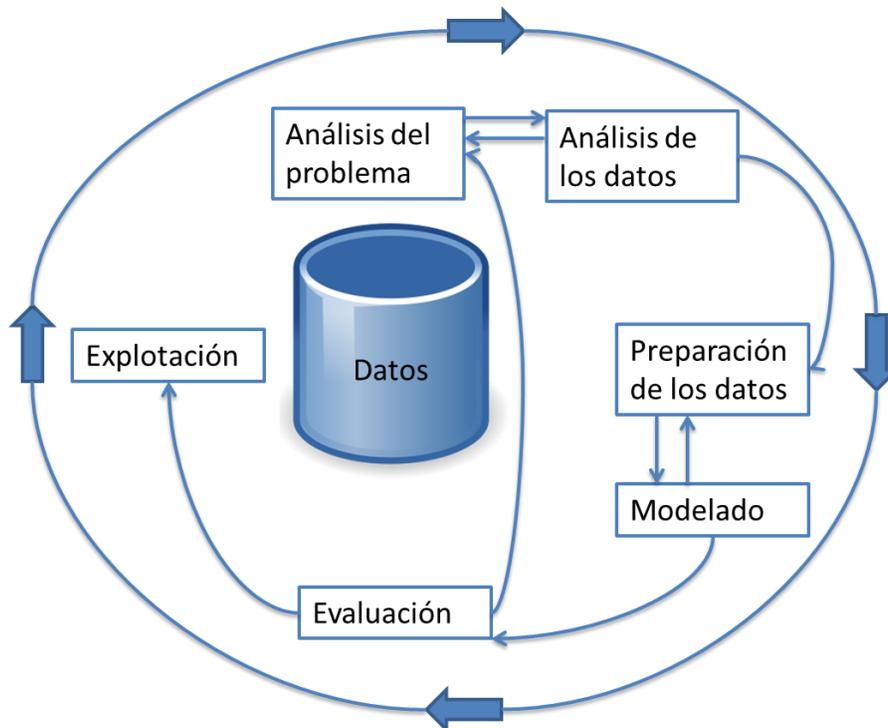


Figura 7. Metodología GUP

Fuente: http://www.ecured.cu/index.php/Game_Unified_Process

La metodología GUP consta de flujos de trabajo, fases y roles:

Flujos de trabajo

Los componentes del flujo de trabajo se encuentran en la sección “Flujos de Trabajo” de EcuRed (2012):

- **Requisitos:** se define la idea o concepción del videojuego, es decir la historia y objetivo principal, además se especifica los requisitos gráficos y sonoros, y se establece la relación de estos requisitos con el objetivo principal del videojuego.
- **Especificación de los requisitos:** se realiza una descripción detallada los requisitos anteriormente especificados en el anterior flujo de trabajo “requisitos”, para así entender mejor el funcionamiento que tendrá el videojuego.
- **Arquitectura:** se detalla la arquitectura software que está involucrada con el desarrollo del videojuego, es decir la metodología de desarrollo, las herramientas tanto de programación, multimedia, diseño, etc.
- **Especificación o detalles del diseño:** en este flujo de trabajo se define la dimensión que tendrá el videojuego, es decir 2D o 3D, adicionalmente se deben diseñar los bocetos de las interfaces o pantallas con las que contará el videojuego indicando las características que éstas tendrán.
- **Código:** en este flujo de trabajo se deberá programar con la finalidad de hacer realidad todo lo que fue diseñado en el flujo de trabajo anterior, esta parte será una de las más extensas.
- **Prueba y corrección de errores:** flujo de trabajo que debe estar presente desde la fase de inicio para garantizar la calidad del videojuego, consiste en verificar que se cumplen los requisitos que fueron especificados y detallados, también se corrigen errores detectados y en general se debe asegurar que el videojuego funcione correctamente.
- **Despliegue:** dentro de este flujo de trabajo se deberá crear y ejecutar el plan que asegure que el videojuego llegue hasta los usuarios finales.

Fases

En lo referente a las fases, estas son las mismas que las de RUP, y son descritas en la sección “Fases” de EcuRed (2012):

- **Creación o inicio:** en esta fase de deben definir los objetivos de funcionamiento del videojuego y los objetivos técnicos del proyecto, además los requisitos funcionales y no funcionales, se debe hacer hincapié en verificar que los flujos de trabajo “requisitos y especificación de requisitos” se realicen correctamente. El resultado de esta fase es la lista de casos de uso y el documento de factores de riesgo del proyecto, además en esta fase no se obtiene una versión ejecutable del videojuego.
- **Elaboración:** en esta fase de debe definir la arquitectura del sistema, y se debe generar una versión ejecutable del videojuego en base a los casos de uso definidos en la fase anterior. Se debe tener en cuenta que la definición de la arquitectura estará regida dentro los requisitos funcionales y no funcionales descritos en la fase anterior.
- **Construcción:** esta fase se desarrolla en ciclos o iteraciones, si existen cambios en los requisitos estos serán aplicados luego de terminar una iteración, trabajar con iteraciones permite tener versiones ejecutables que cada vez van satisfaciendo los casos de uso definidos, todo esto dentro del marco de los riesgos del proyecto identificados en la fase de inicio.
- **Transición:** en esta fase se genera la versión beta, la cual al final del proyecto, será la versión ejecutable final.

Roles

En cuanto a los roles o actores dentro esta metodología se tiene los siguientes en la sección “Roles” de Ecured (2012):

- **Programador:** es quien escribe, depura y actualiza el código fuente del videojuego.
- **Diseñador:** es quien realiza los diseños de los gráficos en dimensiones 2D o 3D.
- **Artistas:** pueden ser guionista o quien crea la historia del videojuego, también está el dibujante quien es el encargado de crear personajes, escenarios y demás objetos del videojuego.
- **Ingenieros de software:** deben cubrir los siguientes roles: gestor de software, líder del proyecto, analista, arquitecto, probador y revisor.

2.4.3. XGD (Extreme Game Development)

Esta es una metodología para el desarrollo de videojuegos que nace de la metodología de Programación extrema, se realizaron modificaciones para que se adapte a programadores, diseñadores, artistas, etc. En lugar de usar casos de uso esta metodología usa estructuras análogas llamadas historia de usuario (EcuRed, 2011).

Según Ecured (2011) la historia de usuario es:

Una representación de un requerimiento de software escrito en una o dos frases utilizando el lenguaje común del usuario. Las historias de usuario son utilizadas en las metodologías de desarrollo ágiles para la especificación de requerimientos (acompañadas de las discusiones con los usuarios y las pruebas de validación). Cada historia de usuario debe ser limitada, esta debería poderse escribir sobre una nota adhesiva pequeña. Las historias de usuario permiten responder rápidamente a los requerimientos cambiantes. Dentro de la definición de esta metodología se tiene que el cliente debe formar parte del equipo de desarrollo, puesto es quien redacta las historias de usuario (Párrafos 2-3).

Se evidencia la correspondencia entre la metodología XGD y la Programación extrema, debido a que los requisitos son cubiertos por los casos de uso dentro la metodología de Programación extrema y por las historias de usuario dentro de la metodología XGD.

Fases

De acuerdo con Armendáriz y Saltos (2013), las fases de la metodología XGD son:



Figura 8. Fases de la metodología XGD

Fuente: http://www.ecured.cu/index.php/Extreme_Game_Development

a) Exploración

Esta fase dura de unas semanas, se realiza para que el cliente aporte sus ideas para las historias de usuario, también en esta fase el equipo de desarrollo se familiariza con las herramientas y la tecnología que usará para desarrollar el videojuego, y además se intenta diseñar el videojuego mediante una lluvia de ideas.

b) Planificación

En esta fase el cliente prioriza las historias de usuario, y los desarrolladores estiman el esfuerzo y tiempo necesario para realizar cada historia de usuario, y en base a ello se construye el cronograma, dicho cronograma indica la duración de las iteraciones y la primera entrega correspondiente al finalizar dichas iteraciones, es importante mencionar que la entrega no puede tomar más de tres meses de desarrollo. Además esta fase puede ser realizada en base al alcance definido para el videojuego.

c) Iteraciones

En esta fase es el cliente quien decide qué historias deben ser implementadas. En general en la primera iteración se establece la idea del videojuego que debe ser desarrollado, mientras que en la última iteración ya se tendrá definido totalmente el videojuego para ser desarrollado en la siguiente fase, además cada iteración no debe durar más de tres semanas. También se realiza el plan de iteración, que está compuesto por: historias de usuario y pruebas de aceptación.

d) Producción

En esta fase las iteraciones suelen durar una semana, durante estas iteraciones se decide qué funcionalidades y características se deben desarrollar, también se debe documentar dicho desarrollo. Esta fase también incluye la realización de pruebas para verificar el funcionamiento correcto del videojuego, y una vez realizado ello ya se puede entregar el videojuego al cliente.

e) Mantenimiento

Por lo general esta fase requiere de nuevo personal, quienes darán soporte al cliente desarrollando nuevas versiones del videojuego hasta que se haya cumplido con todo lo planificado con respecto a la funcionalidad del videojuego,

además la velocidad de trabajo ya no será tan exigente como al inicio del proyecto.

f) Muerte del proyecto

Esta fase se realiza cuando todas las funcionalidades exigidas por el cliente fueron desarrolladas y por lo tanto ya no se debe seguir generando nuevas versiones, además a estas alturas del desarrollo, los objetivos como rendimiento y jugabilidad deben estar ya logrados.

Roles

De acuerdo con Armendáriz y Saltos (2013) los roles de la metodología XGD se clasifican en principales y opcionales:

Principales

a. Development Team (Equipo de desarrollo)

Es un equipo que incluye diseñadores, programadores y artistas sonoros; ellos son los encargados de desarrollar el videojuego y realizar las pruebas de funcionamiento.

b. Manager (Jefe de Proyecto)

Es la persona encargada de lograr que se tengan las condiciones adecuadas y un ambiente de trabajo saludable, además gestiona y supervisa las reuniones que se llevan a cabo.

c. Customer (Cliente)

Es el encargado de escribir las historias de usuario, y las pruebas que deben ser satisfactoriamente superadas para verificar que el videojuego este correctamente desarrollado. Gestiona las historias de usuario, es decir asigna prioridades y también define en que iteraciones serán desarrolladas. Las decisiones del cliente tienen mucha relevancia e impacto en el resto de las personas encargadas del desarrollo del videojuego.

d. Tester (Encargado de Pruebas)

Apoya al cliente en la escritura de las funciones que verifican si el videojuego fue desarrollado correctamente, además realiza las pruebas y presenta los

resultados al equipo de desarrollo, también es responsable de gestionar las herramientas para realizar las pruebas.

e. Tracker (Rastreador)

Se encarga de realizar funciones administrativas, es decir verifica las estimaciones planificadas y el tiempo real empleado, recursos reales empleados, etc.; realiza el seguimiento de las iteraciones, evalúa si los objetivos se pueden cumplir, determina qué cambios deben realizarse para lograr cumplir los objetivos, con todo ello el rastreador es capaz de brindar retroalimentación al equipo de desarrollo.

f. Coach (Entrenador)

Es el encargado de hacer que se aplique de manera correcta la metodología XGD, es decir que se cumplan las reglas y lo establecido para garantizar el éxito del proyecto.

g. Gestor (Big boss)

Cumple la función de manager o representante, ya que se encarga de contactar a los clientes, además proporciona un vínculo directo entre el cliente y el equipo de desarrollo, para que así exista una comunicación fluida.

Opcionales

a. Consultor

Es un equipo o una persona externa al proyecto que tiene la capacidad de apoyar al equipo de desarrollo en algún tema particular, el consultor resolverá las dudas y ayudará al equipo de desarrollo a lograr aquello en lo que se requiere un especialista.

b. Doomsayer (Augur de desastres)

Es el responsable de identificar riesgos durante el desarrollo del proyecto, además debe presentar un plan para mitigar dichos riesgos, la información la proporciona al equipo de desarrollo de manera objetiva, sin agrandar ni disminuir el impacto de los riesgos.

2.4.4. Comparación de las metodologías

La siguiente tabla muestra las características de las metodologías especializadas en el desarrollo de videojuegos, la información fue obtenida de este mismo documento así como del análisis respectivo:

Tabla 4. *Comparación de metodologías de desarrollo de videojuegos*

CARACTERÍSTICAS	METODOLOGÍAS		
	SUM	GUP	XGD
Integrantes del equipo	2 a 7	9 a más	5 a 9
Documentación generada	Regular	Abundante	Regular
Flexibilidad de variación requisitos	Alta	Alta	Alta
Tiempo de desarrollo	Menos de un año	Más de un año	Menos de un año
Metodología base	SCRUM	RUP y XP	XP
Adaptación con otras metodologías	Si	Si	Si
Capacidad de desarrollar software complejo	Media	Alta	Media

FUENTE: elaboración propia.

Las siguientes dos metodologías a describir (SCRUM y RUP) son necesarias para comprender mejor las metodologías de desarrollo de videojuegos ya descritas (SUM, GUP, XGD), las metodologías SCRUM y RUP se detallan a continuación:

2.4.5. SCRUM

Según la SCRUM GUIDES (2013), es una metodología o marco de trabajo para desarrollar software y productos complejos, con la cual se puede entregar avances del software a medida que se avanza el proyecto. Esta metodología es ligera, fácil de entender, pero difícil de dominarla.

SCRUM fue creada en los años 90 y no es un proceso o técnica para crear productos, sin que más bien es un marco de trabajo para usar procesos y técnicas.

En la investigación de Armendáriz y Saltos (2013) se encuentra que la metodología SCRUM es una metodología ágil para gestionar el desarrollo de software de manera iterativa e incremental, siendo esta estructura adecuada para el manejo de software complejo.

La metodología SCRUM utiliza Sprints, estos son iteraciones de dos a cuatro semanas en las que se convierten los requisitos en incrementos grandemente significativos para el proyecto, gracias a la obtención de piezas de software que pueden ser usadas.

Sus características son las siguientes:

- Centrado en la satisfacción del cliente.
- Flexible, ya que se pueden cambiar los requisitos.
- Produce en cada iteración incrementos de software funcionales.
- Diseñado para optimizar el trabajo de las personas más que para mejorar procesos.
- Auto organizado, puesto que el Equipo de Desarrollo gestiona su trabajo.

Eventos

Los eventos de esta metodología se detallan a continuación (SCRUM GUIDES, 2013):

a. Sprint

Es la parte más importante de SCRUM, dura un par de semanas, con su finalización se obtiene nuevos incrementos de software funcionales. El Sprint incluye: Sprint Planning Meeting (Reunión de Planificación del Sprint), Daily Scrums (Scrum Diario), Sprint Review (Revisión de Sprint), Sprint Retrospective (Retrospectiva del Sprint), etc.

b. Planificación del Sprint

Lo crea el Equipo Scrum y contiene lo que se debe realizar en un Sprint, es dirigido por el Scrum Master y es éste quien se asegura que la reunión de planificación se lleve dentro de un plazo no mayor de ocho horas para Sprints de un mes, además se asegura que los asistentes entiendan el plan.

c. Scrum Diario

Es una reunión donde el Equipo de Desarrollo durante 15 minutos sincroniza lo que realizará en las próximas 24 horas, esta reunión se realiza con la finalidad de que cada integrante del Equipo de Desarrollo informe sobre el trabajo que realizó y que realizará para lograr cumplir lo planificado acerca del Sprint.

d. Revisión del Sprint

Es una reunión de unas cuatro horas para revisar el Sprint de un mes, donde se verifica el estado del Sprint y su incremento, básicamente es una reunión

informal más que una reunión de seguimiento, el Scrum Master se encarga de que se cumpla el tiempo fijado para dicha reunión así como que los asistentes entiendan lo tratado en la reunión.

e. Retrospectiva de Sprint

Es donde el Equipo Scrum se examina a sí mismo para ver qué cosas pueden mejorar en ellos, se realiza luego de la Revisión del Sprint y antes de la Planificación del siguiente Sprint. De igual manera que la Revisión del Sprint, el Scrum Master se encarga de gestionar esta reunión que debe llevarse en aproximadamente tres horas para Sprints de un mes.

Artefactos

Los artefactos de esta metodología se detallan a continuación (SCRUM GUIDES, 2013):

a. Pila del producto

Conocido como Product Backlog, está a cargo del Product Owner (Dueño del Producto), quién se encarga de ver todo lo que necesita el producto y administra los cambios dentro de los requisitos.

b. Pila del sprint

Conocido como Sprint Backlog (Lista de pendientes del Sprint), incluye los elementos de la Lista del Producto pero enfocados para un determinado Sprint, permite al Equipo de Desarrollo identificar los objetivos del Sprint.

c. Incremento

Incluye los elementos de la Lista del Producto de los Sprint desarrollados hasta el momento.

Roles

Los roles de esta metodología, se incluyen dentro del Scrum Team (Equipo Scrum), que consta de un Product Owner, un Scrum Master y el Equipo de Desarrollo, el Equipo Scrum es auto organizado porque gestiona su trabajo y es multifuncional ya que no depende de externos para realizar varios tipos de tareas, los roles se detallan a continuación (SCRUM GUIDES, 2013):

a. Product Owner

Es el encargado de asignar el trabajo al Equipo de Desarrollo, así como de gestionar la Lista de Producto. Generalmente este rol es asumido por una sola persona, así que toda la organización debe respetar sus decisiones para que su trabajo pueda ser satisfactorio.

b. Scrum Master

Es el encargado de que todo lo realizado con la metodología Scrum sea entendido y realizado dentro lo establecido, principalmente en el tiempo, además se encarga de que el Equipo Scrum cumpla con las reglas de la metodología. El Scrum Master también brinda apoyo al Equipo Scrum.

c. Equipo de Desarrollo

Son profesionales que crean los incrementos para brindar luego del Sprint el producto funcional, son auto organizados puesto que ellos mismos organizan y gestionan su trabajo, también son multifuncionales ya que trabajan como equipo de varios especialistas.

2.4.6. RUP (Rational Unified Process)

Aunque la metodología RUP no haya sido diseñada para aplicarla al desarrollo de videojuegos, no se puede negar que la manera en la que gestiona el desarrollo de software es muy útil a la hora de llevarlo a cabo, por ello se puede adaptar ciertas partes para aplicarlas al desarrollo de videojuegos, a continuación se presenta una breve descripción de esta útil metodología:

RUP (2002) es un producto que resulta del proceso de ingeniería de software, que se aplica a una organización dedicada al desarrollo de software, su objetivo principal es desarrollar software de alta calidad dentro del tiempo y dinero establecido. La siguiente imagen muestra en resumen como funciona RUP:

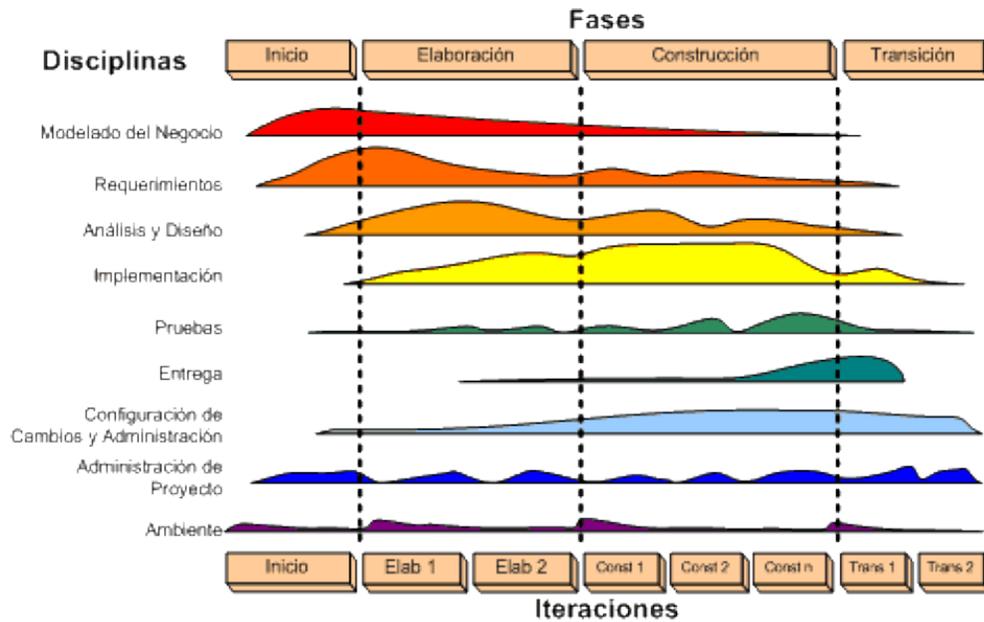


Figura 9: Fases y disciplinas de RUP

Fuente: <http://sce.uhcl.edu/helm/rationalunifiedprocess/>

El eje horizontal muestra el ciclo de vida y las iteraciones que se realizan en cada fase, las cuales culminan en hitos, este eje tiene naturaleza dinámica.

El eje vertical muestra las disciplinas agrupadas por su tipo de finalidad, estas generan artefactos y requieren de roles o encargados para ser llevadas a cabo, este eje tiene naturaleza estática.

Fases de RUP

Las fases de RUP (2002), están separadas por hitos y son realizadas a través del tiempo durante el ciclo de vida del software, estas fases son:

Fase de Inicio: en esta fase se concibe la idea o definición del proyecto y del producto, los interesados definen los requisitos generales que deben ser satisfechos, aquí se determina el tiempo, el costo y el alcance del proyecto, se define que se pretende alcanzar y que no, algunos artefactos generados en esta fase son: la visión, los casos de negocio, la lista de riesgos, el plan de desarrollo de software, el plan de iteración de la fase de Elaboración, el glosario y otros.

Fase de Elaboración: en esta fase se desarrolla la arquitectura y se planea las iteraciones para la fase de Construcción, dicha arquitectura está regida por los casos de uso críticos y la lista de riesgos. Dentro de esta fase se busca definir más formalmente el costo, tiempo y alcance del proyecto, se crean plantillas y

herramientas para apoyar a la administración de los riesgos. Se producen artefactos como: prototipos de la arquitectura, lista de riesgos actualizada, casos de desarrollo refinados, documentación de la arquitectura de software, modelo de datos, documento de visión refinado, plan de iteración para la fase de construcción, modelo de casos de uso al 80% y otros.

Fase de Construcción: en esta fase se termina de definir los requisitos y se desarrolla el sistema para luego hacerle las pruebas necesarias, dicho desarrollo se realiza optimizando el uso de recursos y asegurando la calidad, en esta fase se debe decidir si el sistema está listo para ser desplegado, los artefactos generados son: el sistema desarrollado, el plan de despliegue, el modelo de implementación, actualización del modelo de diseño, refinación de los casos de desarrollo, actualización del modelo de datos, y otros.

Fase de Transición: conocido también como fase de cierre, se realizan las pruebas y correcciones finales antes de poner el sistema a disposición de los usuarios finales, en esta fase se realizan varias iteraciones, la complejidad de una iteración de la fase de Transición puede ser alta o baja, es baja cuando se realiza corrección de errores, y es alta cuando se añade un nuevo requisito, lo cual implica volver a hacer un análisis, diseño e implementación; los artefactos generados en esta fase son: el producto terminado, notas de la versión, manuales, etc.

Disciplinas de RUP

Estas disciplinas se realizan para generar los artefactos, los cuales están relacionados con actividades y actores, las disciplinas de RUP (2002) son las siguientes:

Modelado del negocio: se realiza para entender la dinámica y funcionamiento de la organización donde se desarrollara el software o sistema, identificar los problemas y crear posibles soluciones, asegurar que los usuarios y desarrolladores puedan entenderse y entender la situación a resolver, todo ello con la finalidad de mejorar a la organización. Los actores son: Analista de proceso de negocio, Diseñador de Negocio, Revisor del modelo de negocio. Los artefactos son: glosario de negocio, visión, modelo de casos de uso del negocio, actores del negocio, entidades del negocio, etc.

Requerimientos: permite mantener el entendimiento y comunicación entre los usuarios y los interesados, indica a los desarrolladores lo que deben lograr porque se define los límites de lo que deben realizar, además permite realizar mejor el planeamiento de las iteraciones, y también estimar el tiempo y costo requerido. Los actores son: Analista de sistemas, Arquitecto de software, Especificador de requerimientos, Diseñador de la interfaz de usuario, Revisor de requerimientos. Los artefactos generados son: solicitud de los interesados, glosario y visión, modelo de casos de uso, prototipo de interfaz de usuario, actores, etc.

Análisis y Diseño: dentro de esta disciplina se transforma los requerimientos en diseño del sistema, para darle una arquitectura robusta y adaptar el diseño al entorno donde se implementará. Los actores necesarios en esta disciplina son: Arquitecto de software, Revisor de arquitectura, Diseñador de casos de uso, Revisor de casos de uso, Diseñador de Base de Datos y Diseñador de cápsula. Los artefactos generados en esta disciplina son: documento de arquitectura de software, modelo de análisis, modelo de diseño, cápsula, modelo de base de datos, diseño de paquetes, clases de análisis, diseño de subsistemas, clases de diseño, realización de casos de uso y otros.

Implementación: se desarrolla el sistema, mediante los objetos y clases que son representados en código fuente, archivos ejecutables, además se realizan las pruebas de los componentes que luego son integrados al sistema. Los actores necesarios son: Arquitecto de Software, Programador, Integrador y Revisor de código. Los artefactos generados en esta disciplina son: componentes, plan de integración, versión del sistema y modelo de implementación.

Test: conocido también como prueba, en esta disciplina se detectan y corrigen defectos y errores que afectan la calidad y desempeño del software, esta disciplina permite validar si se cumplió con lo especificado durante el desarrollo de los requerimientos y el diseño. Los actores necesarios son: Diseñador e implementador de prueba, Administrador de prueba, Analista de prueba, y el Probador o Tester. Los artefactos generados son: casos de prueba (para pruebas de caja blanca y pruebas de caja negra), datos de prueba, resultado de prueba, plan de prueba, especificación de la interfaz del test, clases de prueba, componentes de prueba, etc.

Las pruebas de caja blanca son pruebas rigurosas e internas donde se prueba todas las rutas que puede tomar el código fuente, los casos de prueba derivados por cada

ruta deben permitir evaluar expresiones verdaderas y falsas, además de bucles infinitos; una de las técnicas es realizar la prueba de “Camino básico” que permite recorrer cada línea de código al menos una vez, para ello se debe: realizar un diagrama de flujo y un grafo que represente todos los caminos posibles, después se debe calcular la complejidad ciclomática para determinar la cantidad de caminos posibles y finalmente realizar casos de prueba para cada camino.

En cuanto a las pruebas de caja negra se realizan para verificar el comportamiento del sistema sin ver la estructura interna o el código fuente, es decir es una prueba externa que se basa en la relación entre las entradas y salidas del sistema, una de las técnicas para realizar pruebas de caja negra es la “Partición equivalente” que consiste en determinar CE (Condiciones Externas) que pueden ser CEV (Clases de equivalencias válidas) o CEI (Clases de equivalencia inválidas), la primera identifica los resultados o datos que se espera que genere el sistema y la segunda identifica los resultados o datos que se espera que no genere el sistema, luego se deben derivar casos de prueba para cada tipo de clase equivalente; adicionalmente hay dos maneras de realizar casos de prueba, la primera es basada en argumentos de entrada y la segunda es basada en argumentos de salida.

Despliegue: se provee el software a los usuarios finales, además se despliega el software mediante internet si así fue planteado al inicio del desarrollo. Los actores necesarios son: Administrador de despliegue, Administrador de configuración, Programador (Implementador), Escritor técnico, Artista gráfico. Los artefactos generados son: plan de despliegue, notas de versión, el software, artefactos de instalación, el trabajo de arte y manuales de usuario.

Entorno: describe las actividades necesarias para desarrollar las directrices que dan apoyo al proyecto, además provee de procesos y herramientas que apoyan al equipo de desarrollo de software de la organización. Los actores requeridos para ello son: Ingeniero de procesos, Especialista de herramientas, Diseñador de interfaz de usuario, Analista de sistemas, Analista del proceso de negocios, Arquitecto de software, Diseñador de prueba, Escritor técnico y Administrador del sistema. Los artefactos generados son: casos de desarrollo, evaluación de la organización, directrices del modelado del negocio, pruebas e interfaz de usuario, herramientas, infraestructura de desarrollo, etc.

Administración del proyecto: esta disciplina provee de un marco para administrar proyectos intensivos de software, planeación, ejecución y monitoreo de proyectos y gestión de riesgos. Los actores necesarios para llevar a cabo esta disciplina son: Administrador de proyecto y Revisor de proyecto. Los artefactos generados por el desarrollo de esta disciplina son: plan de desarrollo de software, casos de negocio, planes de iteración, evaluación y estado de iteración, plan de administración de riesgos, lista de riesgos, plan de aceptación del producto, plan de evaluación de calidad, revisión de registros, etc.

Configuración y gestión del cambio: se utiliza esta disciplina para evitar problemas de cambio referentes a: múltiples versiones, notificación de solución de errores y actualización simultánea, esto porque durante en el desarrollo del proyecto se generan numerosos artefactos que están distribuidos en el área encargada de varios actores. Los actores requeridos para llevar a cabo esta disciplina son: Administrador de configuración, Administrador de control de cambio e Integrador. Los artefactos generados son: plan de administración de configuración, solicitud de cambios, workspace de integración, workspace de desarrollo.

2.5. TELÉFONOS INTELIGENTES

2.5.1. Definición

Alonso, Artime, Rodríguez y Baniello (2009) aducen que el teléfono inteligente o smartphone es:

Un dispositivo electrónico que funciona como un teléfono móvil con características similares a las de un ordenador personal. Es un elemento a medio camino entre un teléfono móvil clásico y una PDA [Agenda Personal de Datos] ya que permite hacer llamadas y enviar mensajes de texto como un móvil convencional pero además incluye características cercanas a las de un ordenador personal. Una característica importante de casi todos los teléfonos inteligentes es que permiten la instalación de programas para incrementar el procesamiento de datos y la conectividad. Estas aplicaciones pueden ser desarrolladas por el fabricante del dispositivo, por el operador o por un tercero. Los teléfonos inteligentes se distinguen por muchas características, entre las que destacan las pantallas táctiles, un sistema

operativo así como la conectividad a Internet y el acceso al correo electrónico. Otras aplicaciones que suelen estar presentes son las cámaras integradas, la administración de contactos, el software multimedia para reproducción de música y visualización de fotos y video-clips y algunos programas de navegación así como, ocasionalmente, la habilidad de leer documentos de negocios en variedad de formatos como PDF y Microsoft Office. Una característica común a la mayoría de “smartphones” es una lista de contactos capaz de almacenar tantos contactos como la memoria libre permita, en contraste con los teléfonos clásicos que tienen un límite para el número máximo de contactos que pueden ser almacenados. Casi todos los teléfonos inteligentes también permiten al usuario instalar programas adicionales (p. 3).

La mayoría de los teléfonos inteligentes permiten la instalación de diversas aplicaciones, también poseen la característica de ser táctiles, por ello no necesitan de un teclado físico aunque algunos modelos si lo poseen.

2.5.2. Características

A) Plataformas (Sistemas Operativos)

En el sitio de web de estadística de uso de tecnología a nivel mundial Netmarketshare (2015) se obtuvo el resultado del uso de sistemas operativos móviles para smartphones y tabletas para Noviembre del 2015, en el cual el líder es el sistema operativo de Google (Android), luego le sigue el de la empresa Apple (iOS), después el de Microsoft (Windows Phone), quedando relegados los de la compañía Nokia (Symbian), BlackBerry (BlackBerry OS), y los demás.

La siguiente tabla detalla la información descrita:

Tabla 5. *Sistemas operativos más usados*

Sistema operativo	Cuota en el mercado
Android	54.83%
iOS	37.61%
Windows Phone	3.02%
Java ME	1.75%
Symbian	1.60%
BlackBerry	1.11%
Samsung	0.03%
Kindle	0.02%
Otros	0.02%

Fuente: <http://www.netmarketshare.com/>

Se hará énfasis en la descripción de los sistemas operativos Android e iOS debido a su gran demanda.

El sistema operativo Android fue desarrollado en el 2008, está basado en el Kernel de Linux, fue desarrollado inicialmente por “Android Inc.” y luego comprada por la compañía Google, este sistema operativo funciona en los teléfonos inteligentes, tabletas y demás dispositivos modernos (Motyka, 2014).

A continuación se lista las versiones que tuvo el sistema operativo Android (2015):

- **Android 1.6:** nombrado “Donut”, fue lanzado en el 2008 antes que existiera la tienda de Google (Play Store), era compatible con pantallas de 320x480 pixeles.
- **Android 2.2:** nombrado “Froyo”, introdujo el compilador JIT Dalvik que mejoró el rendimiento del CPU, también agregó la funcionalidad de hotspot Wi-Fi, con la cual se podía hacer funcionar al teléfono inteligente similar a un enrutador. Otra característica que incluyó fue la de realizar acciones en el teléfono usando la voz en lugar de los dedos.
- **Android 2.3:** nombrado “Gingerbread”, incorporó la característica de ver qué cantidad de batería consumía cada aplicación, también incorporó una API (Interfaz de Programación de Aplicaciones), con lo cual los desarrolladores crearon videojuegos para Android.
- **Android 3.0:** nombrado “Honeycomb”, incorporó una funcionalidad para realizar ajustes rápidos como ver la fecha y hora, duración de la batería y estado de la conexión, además adecuó el sistema operativo para que también funcionara en tabletas.
- **Android 4.0:** nombrado “Ice Cream Sandwich”, los teléfonos Android ahora son capaces de compartir datos entre sí mediante NFC (Near field Communication), también incorporó una funcionalidad para administrar el uso de datos móviles y permitió personalizar la pantalla de inicio.
- **Android 4.1:** nombrado “Jelly Bean”, habilitó la funcionalidad que permitió varios usuarios en un solo dispositivo, también incorpora la notificación de

acciones concretas que permite responder inmediatamente a una notificación.

- **Android 4.4:** nombrado “Kitkat”, permite gestionar de manera inteligente los contactos con quienes se realizan comunicaciones más frecuentes, permite abrir el navegador de Google, enviar mensaje de texto o reproducir una canción al decir “Ok Google”.
- **Android 5.0:** nombrado “Lollipop”, mejora el sistema de notificaciones, las cuales ahora pueden ser accedidas sin desbloquear el teléfono, la funcionalidad de pantalla múltiple permite continuar lo que se estaba haciendo con una aplicación pero ahora en cualquier otro dispositivo de Android como el Android TV y el Android Wear, además mejora la manera de navegar en el teléfono, haciéndola más fácil.
- **Android 6.0:** nombrado “Marshmallow”, incorpora una funcionalidad que brinda ayuda a los usuarios, permite definir lo que quieres compartir con tus aplicaciones y apagar permisos cuando sea, optimiza el uso de la batería.

Por otro lado se tiene el sistema operativo “iOS” (en inglés iPhone Operating System) es un sistema operativo propietario de la empresa Apple Inc. utilizado en dispositivos como smartphones, tablets y otros dispositivos como televisiones o reproductores mp4, entre ellos los famosos iPhone, iPad y iPod” (Filgueira, 2009, Párrafo 1).

Es muy similar a Android, permitiéndonos como éste instalar aplicaciones o apps, para aumentar las funcionalidades de nuestros dispositivos, con lo que además de las funciones básicas de un teléfono tendríamos aplicaciones ofimáticas, de productividad, almacenamiento en la nube, educativas, juegos y muchas más. Para instalar aplicaciones podemos utilizar su tienda oficial App Store que ofrece actualmente [2009] cerca de un millón de apps (Filgueira, 2009, Párrafo 2).

El primer teléfono inteligente que uso el sistema operativo iOS fue el iPhone en el año 2007. Las versiones de este sistema operativo se describen a continuación por el Servicio técnico de Apple (2015):

- **iPhone OS:** fue lanzado en el 2007, fue el primer sistema operativo de Apple, no soportaba tecnología 3G, tampoco la función de copiar y pegar, tampoco aceptaba instalación de programas fabricados por terceros.
- **iPhone OS 2:** lanzado en el 2008, se instaló en el iPhone 3G a la vez que se creó la App Store (tienda virtual de Apple), ahora sí se podía instalar aplicaciones de terceros.
- **iPhone OS 3:** lanzado en el 2009, se instaló en el iPhone 3GS, ahora se podían realizar las funciones de copiar y pegar, además se agregó el servicio de mensajería multimedia, también se podían comprar aplicaciones desde iTunes. La versión 3.2 fue adaptada para ser instalada en el iPad.
- **iOS 4:** lanzado en el 2010, a partir de ahora el sistema operativo llevará el nombre de iOS, fue instalado en el iPhone 4, permitía almacenar 12 aplicaciones dentro de una carpeta, lo más trascendental fue que permitía la ejecución de aplicaciones en modo multitarea.
- **iOS 5:** lanzado en el 2011, fue instalado en el iPhone 4S, lo más resaltante fue el que el sistema operativo incorporó un asistente de voz llamado Siri que en realidad fue la voz de Susan Bennet, además se integró Twitter y la mensajería iOS.
- **iOS 6:** lanzado el 2012, en la WWDC (Conferencia Mundial de Desarrolladores) en la ciudad de San Francisco, se reemplazó a Google Maps por una aplicación similar a ésta pero desarrollado por Apple, además se actualizó la interfaz de Siri.
- **iOS 7:** lanzado el 2013, fue instalado en el iPhone 5S y 5C, su interfaz fue con un diseño aplanado, incorporó la función Centro de control, con la cual se podía acceder a los ajustes del iPhone de manera rápida, además Siri ya estaba disponible en francés, alemán y en voces de mujer y hombre.
- **iOS 8:** lanzado el 2014 en la WWCD, se instaló en el iPhone 6 e iPhone 6 Plus, incluyó aplicaciones de salud como HealthKit, posteriormente fue actualizado a iOS 8.4. También incorporó QuickType, aplicación de tipificación predictiva y mejoró el servicio de almacenamiento en iCloud.

- **iOS 9:** lanzado el 2015, incluye mejoras en la administración de la batería, Siri es capaz de realizar búsquedas profundas y se incluyen las aplicaciones Over, Dividir vista y Picture in Picture para el iPad.

B) Hardware

Se puede encontrar especificaciones generales del hardware de los teléfonos móviles en la página web de la Universidad Nacional Abierta y a Distancia (2013b):

Placa base: circuito integrado en cuyo interior se encuentra el procesador, y todas las piezas electrónicas que permiten que el teléfono funcione.

Antena: pieza que permite la recepción y envío de señales que permiten la continuación entre los teléfonos.

Antena WiFi: pieza que permite la recepción y envío de señales que pertenecen al estándar 802.11 “a”, “b”, “g” y también “n”.

Antena NFC: Near Field Communication, para la recepción y envío de datos entre teléfonos que no estén separados por grandes distancias.

Pantalla: permite la comunicación entre el usuario y el teléfono, existen pantallas de cristal líquido (LCD) y la mayoría de ellas son táctiles.

Teclado: aunque la mayoría de los teléfonos ya no posee un teclado físico, aún existen, y estos teclados ya sean físicos o virtuales suelen usar el formato de entrada QWERTY.

Micrófono: es una pieza importante ya que transforma la voz del usuario en energía eléctrica para que así pueda ser transmitida a otros teléfonos.

Altavoz: es la pieza que reproduce los sonidos generados por el teléfono o por las aplicaciones que este posea.

Batería: es una pieza muy importante ya que permite que el teléfono pueda funcionar, puesto que le brinda la energía eléctrica para que realice sus funciones.

Puerto de carga: es utilizado para permitir cargar energía a la batería del teléfono, existen cargadores en forma de enchufe y en forma de USB (Universal Serial Bus).

Memoria: existen varios tipos de memoria en el teléfono, se puede mencionar la memoria RAM, que es la que usa el procesador para ejecutar instrucciones y aplicaciones; también está la memoria ROM (Memoria de Solo Lectura), donde se encuentra instalado el sistema operativo, así como memorias externas, como pueden ser una tarjeta SD, etc.

C) Gama

Se mostrará a continuación algunos de los teléfonos inteligentes de las gamas baja, media y alta, se hará énfasis en el detalle de los smartphones con sistema operativo Android e iOS debido a que sobre ellos se ejecutó el presente proyecto. También se mostrarán algunos smartphones que usan los diferentes sistemas operativos descritos en la “Tabla 5: Sistemas operativos más usados”.

Baja

Aquí pertenecen los teléfonos móviles que poseen funciones básicas, aunque pueden ejecutar varias aplicaciones, muchas veces terminan obteniendo un rendimiento bajo y además no pueden ejecutar aplicaciones pesadas.

Para los teléfonos inteligentes con sistema operativo Android se tiene la siguiente lista (Cejas, 2012):

- Samsung Galaxy S: posee un procesador ARM Cortex A8 1GHz, memoria RAM de 512MB, almacenamiento interno de 8/16GB, cámara de 5 MP y sistema operativo 2.3 Gingerbread.
- NEXUS S: posee un procesador de un núcleo 1GHz Cortex-A8, memoria RAM de 512MB, almacenamiento interno de 16GB, cámara de 5 MP y sistema operativo Android 2.1 Eclair.
- Samsung Galaxy S Advance: posee un procesador Cortex-A9 dual-core 1GHz, memoria RAM de 768MB, almacenamiento interno de 8GB/16GB cámara de 5 MP y sistema operativo Android 2.3 Gingerbread.
- SAMSUNG GALAXY ACE: posee un procesador de un núcleo Qualcomm MSM7227 800 MHz, memoria RAM de 278MB, almacenamiento interno de 128MB, cámara de 5 MP y sistema operativo Android 2.3 Gingerbread.

- Samsung Galaxy Ace 2: posee un procesador de doble núcleo NovaThor U8500 a 800MHz, memoria RAM de 768MB, almacenamiento interno de 4GB, cámara de 5 MP y sistema operativo Android 2.3 Gingerbread.
- NEXUS ONE: posee un procesador Qualcomm Snapdragon QSD8250 1GHz, memoria RAM de 512MB, cámara de 5 MP y sistema operativo Android 2.1 Eclair, actualizable a Android 2.3 Gingerbread.
- Sony Xperia Play: posee un procesador Qualcomm MSM8255 Snapdragon 1GHz, memoria RAM de 380MB, cámara de 5 MP y sistema operativo Android 2.3 Gingerbread, actualizable a Android 4.0 Ice Cream Sandwich.

También se tiene los smartphones de la familia LG (LG, 2015), por ejemplo:

LG Optimus L3 II: posee un procesador de 1GHz de núcleo simple, 512MB de RAM, 4GB de almacenamiento interno, cámara de 3 megapíxeles, y sistema operativo Android 4.1 Jelly Bean. Es por ello uno de los terminales más básicos de la línea LG Optimus.

Mientras que por el lado de la empresa Apple que fabrica los smartphones de gama baja con sistema operativo iOS se tiene la siguiente lista proporcionada por la página web de teléfonos móviles GSMarena (2014):

- iPhone 1G: posee un procesador Samsung ARM 1176 a 412MHz, una memoria RAM de 128 MB y sistema operativo iOS 3.1.3.
- iPhone 3G: posee un procesador Samsung ARM1176JZF-S (412MHz), memoria RAM de 128 MB y sistema operativo iOS 4.2.1.
- iPhone 3GS: posee un procesador Samsung ARM Cortex A8, 600Mhz, memoria RAM de 256 MB y sistema operativo iOS 6.1.6.

También se pueden ver los siguientes ejemplos de Conde (2013) donde se muestra el nombre del teléfono móvil para la línea BlackBerry:

- BlackBerry Bold 9930.
- BlackBerry Torch 9860.
- BlackBerry Curve 9220.
- BlackBerry Bold 9780.

- BlackBerry Curve 9310.
- BlackBerry Curve 9315.
- BlackBerry Curve 9350.
- Blackberry Curve 9360 Javelin 2.
- BlackBerry Pearl 3G 9100.
- BlackBerry 9620.

Media

Los de esta categoría si pueden ejecutar la mayoría de las aplicaciones desarrolladas debido a su hardware mejorado, sin embargo el costo va aumentando significativamente.

Para los teléfonos inteligentes con sistema operativo Android se tiene la siguiente lista (Cejas, 2012):

- Samsung Galaxy S2: posee un procesador de doble núcleo Exynos Cortex A9 dual-core 1.2GHz, memoria RAM de 1GB, memoria de almacenamiento interno de 16GB/32GB, cámara de 8 MP y sistema operativo Android 2.3 Gingerbread.
- Samsung Galaxy S2 HD: posee un procesador dual-core 1.5GHz Scorpion, memoria RAM de 1GB, memoria de almacenamiento interno de 16GB, cámara de 8 MP y sistema operativo Android 2.3 Gingerbread.
- Samsung Galaxy Nexus: posee un procesador de doble núcleo TI OMAP 4460 dual-core 1.2GHz, memoria RAM de 1GB, memoria de almacenamiento interno de 16GB/32GB, cámara de 5 MP y sistema operativo Android 4.0 Ice Cream Sandwich.
- Samsung Galaxy S3 Mini: posee un procesador de doble núcleo NovaThor U8420 dual-core 1GHz, memoria RAM de 1GB, memoria de almacenamiento interno de 8/16GB, cámara de 5 MP y sistema operativo Android 4.1 Jelly Bean.

- Motorola Droid Razr Maxx: posee un procesador de doble núcleo Cortex-A9 dual-core 1.2GHz, memoria RAM de 1GB, memoria de almacenamiento interno de 16GB, cámara de 8 MP y sistema operativo Android 2.3 Gingerbread.
- HTC ONE S: posee un procesador Qualcomm MSM8260A dual-core 1.5GHz, memoria RAM de 1GB, memoria de almacenamiento interno de 16GB, cámara de 8 MP y sistema operativo Android 4.0 Ice Cream Sandwich.
- HTC SENSATION: posee un procesador Qualcomm Snapdragon MSM 8260 dual core 1.2GHz, memoria RAM de 768MB, memoria de almacenamiento interno de 1GB, cámara de 8 MP y sistema operativo 2.3 Gingerbread, actualizable a Jelly Bean.
- SONY XPERIA S: posee un procesador Qualcomm MSM8260 dual-core 1.5GHz, memoria RAM de 1GB, memoria de almacenamiento interno de 16GB/32GB, cámara de 12 MP y sistema operativo 2.3 Gingerbread, actualizable.

De la misma manera la empresa Apple también fabrica los smartphones de gama media con sistema operativo iOS cuya lista proporcionada por la página web de teléfonos móviles GSMarena (2014):

- iPhone 4: posee un procesador ARM Cortex A-8 a 800MHz, memoria RAM de 512/712/912 MB y usa sistema operativo iOS 7.1.2.
- iPhone 4S: posee un procesador ARM Cortex A5 Doble núcleo a 800MHz, memoria RAM DDR2 de 512 MB y usa sistema operativo iOS 8.1.

También se pueden ver los siguientes ejemplos de Conde (2013) donde se muestra el nombre de los teléfonos móviles que usan otros sistemas operativos:

- BlackBerry Q10.
- Nokia Lumia 822.
- Nokia Lumia 928.
- Nokia Lumia 925.

Alta

Los teléfonos móviles que pertenecen a esta categoría son los más caros pero a la vez eficientes en cuanto a su rendimiento ya que pueden ejecutar cualquier tipo de aplicación.

Para los teléfonos inteligentes con sistema operativo Android se tiene la siguiente lista (Cejas, 2012):

- NEXUS 4: posee un procesador quad-core Qualcomm Snapdragon APQ8064 1.5GHz, memoria RAM de 2GB, cámara de 8 MP y sistema operativo Android 4.2 Jelly Bean.
- LG Optimus G: posee un procesador quad-core Qualcomm Snapdragon APQ8064 1.5GHz, memoria RAM de 2GB, almacenamiento interno de 32GB, cámara de 13 MP y sistema operativo Android 4.0.1 Ice Cream Sandwich.
- LG Optimus 4X HD. posee un procesador NVIDIA Tegra 3 quad-core 1.5GHz, memoria RAM de 1GB, almacenamiento interno de 16GB, cámara de 8 MP y sistema operativo Android 4.0.1 Ice Cream Sandwich.
- Samsung Galaxy S3: posee un procesador Exynos 4 Quad quad-core 1.4 GHz, memoria RAM de 1GB, almacenamiento interno de 16/32/64GB, cámara de 8 MP y sistema operativo Android 4.1 Ice Jelly Bean.
- Samsung Galaxy Note II: posee un procesador Exynos 4412 Quad quad-core 1.6GHz, memoria RAM de 2GB, almacenamiento interno de 16/32/64GB, cámara de 8 MP y sistema operativo Android 4.1 Ice Jelly Bean.
- HTC ONE X: posee un procesador Nvidia Tegra 3 quad-core 1.5GHz, memoria RAM de 1GB, almacenamiento interno de 32GB, cámara de 8 MP y sistema operativo Android 4.0.1 Ice Cream Sandwich.

La empresa Apple también fabrica los smartphones de gama alta con sistema operativo iOS cuya lista proporcionada por la página web de teléfonos móviles GSMarena (2014):

- iPhone 5: posee un procesador Apple A6 doble núcleo a 1.3GHz, memoria RAM DDR2 de 1GB y usa sistema operativo iOS 8.

- iPhone 5S: posee un procesador PowerVR G6430 (4 Núcleos a 200 MHz), memoria RAM DDR3 de 1GB y usa sistema operativo iOS 8.
- iPhone 5C: posee un procesador PowerVR G6430 (4 Núcleos a 200 MHz), memoria RAM DDR3 de 1GB y usa sistema operativo iOS 8.
- iPhone 6: posee un procesador 1.4GHz dual-core ARMv8-A, memoria RAM DDR3 de 1GB, usa sistema operativo iOS 8 y mide 4.7 pulgadas.
- iPhone 6 Plus: posee un procesador 1.4GHz dual-core ARMv8-A, memoria RAM DDR3 de 1GB y usa sistema operativo iOS 8 y mide 5.5 pulgadas.

También se pueden ver los siguientes ejemplos de Conde (2013) donde se muestra el nombre del teléfono móvil para los smartphones que usan los demás sistemas operativos:

- BlackBerry Z10.
- Nokia Lumia 1020.
- Nokia Lumia 1520.

2.6. HERRAMIENTAS DE DESARROLLO DE VIDEOJUEGOS

2.6.1. Motores de desarrollo

Hoy en día existen diversos motores de desarrollo conocidos como “Game Engines” para el desarrollo de Videojuegos, según el artículo Gerardi (2013) un motor de juego es el que se utiliza para hacer la mayor parte de las parte de un videojuego, se le conoce también como el corazón y el cerebro de un videojuego. Es el código del motor quien se encarga de las cosas aparentemente simples que son universales para la mayoría de los videojuegos, como la interpretación de las entradas del controlador, carga y administración de los gráficos y el sonido, así como el modelado de la física. Una vez que todas esas cosas son implementadas, el motor pone en marcha del juego, así los desarrolladores son libres de crear el arte, los sonidos y las interacciones entre la apariencia y la personalidad de su videojuego, sin perder tanto tiempo en preocuparse por los fundamentos tecnológicos.

Los principales motores de desarrollo de videojuegos son:

A. Corona SDK

De acuerdo a su sitio web oficial CoronaLabs (2015b) es un motor de desarrollo de videojuegos en 2D para dispositivos móviles, y es considerado como el mejor en esta área, sin embargo también permite el desarrollo de aplicaciones de negocio, posee un emulador que permite ver al instante el resultado del código que se va escribiendo, además de un depurador en línea de comandos; se pueden usar editores como el Sublime, Bloc de Notas, Notepad y demás. Las plataformas soportadas son iOS, Android, Android TV, Windows Phone y Kindle, es decir que se puede publicar para estas plataformas usando el mismo código fuente, sin realizar cambio alguno. También soporta plataformas como Mac OS X y Windows. Corona SDK se basa en OpenAL, Box2D, Facebook, SQLite y otros; el motor gráfico y de física además del lenguaje que emplea, es decir Lua, permiten obtener resultados rápidos usando pocas líneas de código, la rapidez de desarrollo es de hasta 10 veces más que otros motores. Además se puede hacer la manipulación nativa del hardware de las diferentes plataformas. Tiene soporte para 64 bits gracias a las constantes actualizaciones y corrección de errores que viene realizando. También permite la monetización por la venta de aplicaciones en las tiendas Play Store y App Store.

Corona SDK tiene una documentación completa sobre cómo usar de manera eficiente el lenguaje Lua, sus librerías y APIs para desarrollar videojuegos e incluso otras aplicaciones, dentro de dicha documentación se incluye las secciones de (CoronaLabs, 2015a):

Guía de inicio: donde se explica los requerimientos de sistema para instalar Corona SDK en Windows y Mac OS X, además se da una pequeña introducción a Lua.

Corona básico: donde se explica cómo configurar el proyecto, realizar las depuraciones y como optimizar el desarrollo del videojuego para hacer que este consuma pocos recursos del teléfono inteligente.

Distribución y construcción: en esta sección se explica cómo realizar las configuraciones para compilar las aplicaciones tanto para Android donde se genera el .apk que es la extensión del instalador, así también compilar para iOS

donde se genera el .app que es la extensión del instalador de dicho sistema operativo, la extensión de este archivo .app puede ser convertido en extensión .ipa para ser instalado directamente en un iPhone sin necesidad de subirla a la App Store.

Animación, audio y gráficos: en esta sección se explica cómo cargar, eliminar y optimizar el uso de los recursos de audio y gráfico para el videojuego, además se explican cómo funcionan las transiciones y la animación Sprite.

Eventos y listeners: aquí se explica cómo funcionan los eventos de detección de acciones del usuario, también se detalla la manera en que estos eventos afectan el rendimiento del videojuego.

Escenas e interfaz de usuario: en esta sección se muestra las APIs (Interfaz de programación de aplicaciones) de los Widget (objetos prediseñados) con las que cuenta Corona SDK, además detalla cómo se gestionan las escenas del videojuego y también muestra la manera de personalizar el tipo de letra o fuente.

Monetización y publicidad: muestra cómo se puede programar las funciones encargadas de permitir compras dentro del videojuego y también como insertar publicidad.

Física: donde se explica la manera en la que se debe usar la física para el videojuego, además se detalla las funciones y propiedades necesarias para el correcto uso de la física.

Comunicación social: esta sección se dedica a mostrar cómo se hace para vincular el videojuego con la red social Facebook.

Cadenas archivos y datos: aquí se detalla como programar las funciones encargadas de leer y escribir datos entre el videojuego y la memoria interna del teléfono inteligente.

En cuanto a la estructura interna de Corona SDK, según su página web de documentación CoronaLabs (2015c), se define los subsistemas, librerías y clases más importantes que conforman las APIs del motor de desarrollo de videojuegos Corona SDK, los cuales se muestran a continuación:

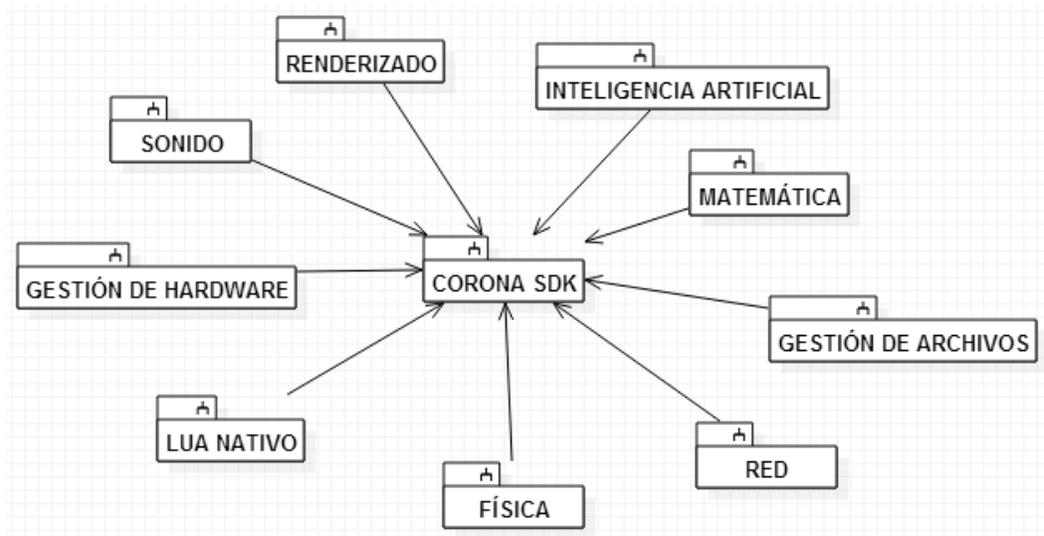


Figura 10: Subsistemas del motor de videojuegos Corona SDK

Fuente: <https://docs.coronalabs.com/api/index.html>

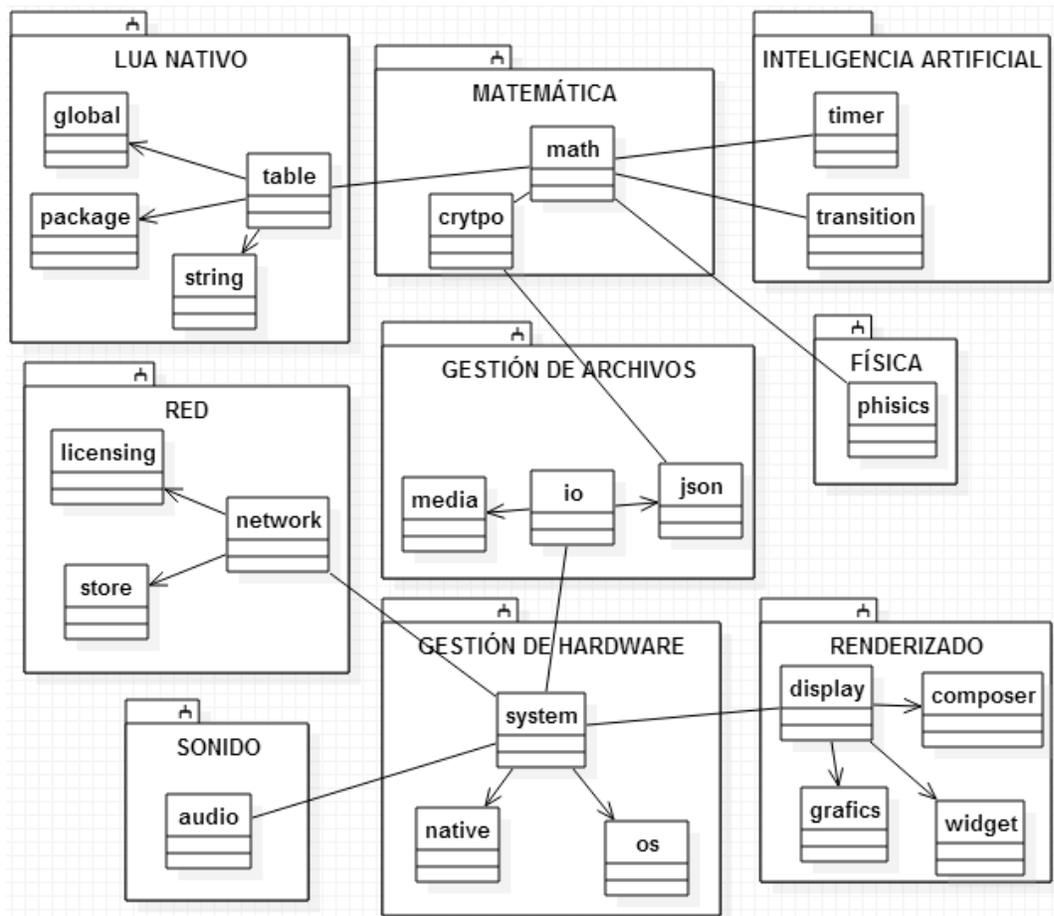


Figura 11: Diagrama de clases del motor de videojuegos Corona SDK

Fuente: <https://docs.coronalabs.com/api/index.html>

Tabla 6: Clases del motor de videojuegos Corona SDK

Clases de Corona SDK						
<ul style="list-style-type: none"> • Subsistema: Sonido • Nombre de la clase: audio • Descripción: pertenece a la librería para acceder a las características de OpenAL • Propiedades y métodos: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 2px;">audio</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;"> +freeChannels +totalChannels +unreservedFreeChannels +unreservedUsedChannels +usedChannels </td> </tr> <tr> <td style="padding: 2px;"> +dispose() +fade() +fadeOut() +findFreeChannel() +getDuration() +getMaxVolume() +getMinVolume() +getVolume() +isChannelActive() +isChannelPaused() +isChannelPlaying() +loadSound() +loadStream() +pause() +play() +reserveChannels() +resume() +rewind() +seek() +setMaxVolume() +setMinVolume() +setVolume() +stop() +stopWithDelay() </td> </tr> </tbody> </table>	audio	+freeChannels +totalChannels +unreservedFreeChannels +unreservedUsedChannels +usedChannels	+dispose() +fade() +fadeOut() +findFreeChannel() +getDuration() +getMaxVolume() +getMinVolume() +getVolume() +isChannelActive() +isChannelPaused() +isChannelPlaying() +loadSound() +loadStream() +pause() +play() +reserveChannels() +resume() +rewind() +seek() +setMaxVolume() +setMinVolume() +setVolume() +stop() +stopWithDelay()	<ul style="list-style-type: none"> • Subsistema: Red • Nombre de la clase: network • Descripción: pertenece a la librería para administrar conexiones de red • Propiedades y métodos: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 2px;">network</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;"> +canDetectNetworkStatusChanges() +setStatusListener() +request() +cancel() +download() +upload() </td> </tr> </tbody> </table>	network	+canDetectNetworkStatusChanges() +setStatusListener() +request() +cancel() +download() +upload()
audio						
+freeChannels +totalChannels +unreservedFreeChannels +unreservedUsedChannels +usedChannels						
+dispose() +fade() +fadeOut() +findFreeChannel() +getDuration() +getMaxVolume() +getMinVolume() +getVolume() +isChannelActive() +isChannelPaused() +isChannelPlaying() +loadSound() +loadStream() +pause() +play() +reserveChannels() +resume() +rewind() +seek() +setMaxVolume() +setMinVolume() +setVolume() +stop() +stopWithDelay()						
network						
+canDetectNetworkStatusChanges() +setStatusListener() +request() +cancel() +download() +upload()						

Continuación (Tabla 6: Clases del motor de videojuegos Corona SDK)

<ul style="list-style-type: none"> • Subsistema: Red • Nombre de la clase: licensing • Descripción: pertenece a la librería para verificar si la app fue comprada desde Play Store (única que soporta esta librería) • Propiedades y métodos: <table border="1" data-bbox="467 443 646 590"> <tr><th>licensing</th></tr> <tr><td>+licensing() +verify()</td></tr> </table>	licensing	+licensing() +verify()	<ul style="list-style-type: none"> • Subsistema: Red • Nombre de la clase: store • Descripción: pertenece a la librería para realizar ventas dentro de la aplicación (sólo disponible para Play Store y App Store) • Propiedades y métodos: <table border="1" data-bbox="980 474 1248 856"> <tr><th>store</th></tr> <tr><td>+availableStores +canLoadProducts +canMakePurchases +isActive +target</td></tr> <tr><td>+finishTransaction() +init() +loadProducts() +purchase() +restore()</td></tr> </table>	store	+availableStores +canLoadProducts +canMakePurchases +isActive +target	+finishTransaction() +init() +loadProducts() +purchase() +restore()
licensing						
+licensing() +verify()						
store						
+availableStores +canLoadProducts +canMakePurchases +isActive +target						
+finishTransaction() +init() +loadProducts() +purchase() +restore()						

<ul style="list-style-type: none"> • Subsistema: Lua Nativo • Nombre de la clase: global • Descripción: pertenece a la librería básica de Lua 5.1 para usarla en Corona SDK • Propiedades y métodos: <table border="1" data-bbox="467 1087 686 1770"> <tr><th>global</th></tr> <tr><td>~assert() ~collectgarbage() ~error() ~getfenv() ~getmetatable() ~ipairs() ~next() ~pairs() ~pcall() ~print() ~rawequal() ~rawget() ~rawset() ~select() ~setfenv() ~setmetatable() ~tonumber() ~tostring() +type() +unpack()</td></tr> </table>	global	~assert() ~collectgarbage() ~error() ~getfenv() ~getmetatable() ~ipairs() ~next() ~pairs() ~pcall() ~print() ~rawequal() ~rawget() ~rawset() ~select() ~setfenv() ~setmetatable() ~tonumber() ~tostring() +type() +unpack()	<ul style="list-style-type: none"> • Subsistema: Lua Nativo • Nombre de la clase: package • Descripción: pertenece a la librería de Lua 5.1 para crear y cargar librerías • Propiedades y métodos: <table border="1" data-bbox="980 1087 1159 1262"> <tr><th>package</th></tr> <tr><td>+loaded +loaders</td></tr> <tr><td>+require()</td></tr> </table>	package	+loaded +loaders	+require()
global						
~assert() ~collectgarbage() ~error() ~getfenv() ~getmetatable() ~ipairs() ~next() ~pairs() ~pcall() ~print() ~rawequal() ~rawget() ~rawset() ~select() ~setfenv() ~setmetatable() ~tonumber() ~tostring() +type() +unpack()						
package						
+loaded +loaders						
+require()						

Continuación (Tabla 6: Clases del motor de videojuegos Corona SDK)

- Subsistema: Lua Nativo
- Nombre de la clase: string
- Descripción: pertenece a la librería estándar de Lua 5.1 para manipular cadenas (extraer subcadenas, realizar comparaciones de caracteres, etc.)
- Propiedades y métodos:

string
+byte()
+char()
+find()
+format()
+gmatch()
+gsub()
+len()
+lower()
+match()
+rep()
+reverse()
+sub()
+upper()

- Subsistema: Lua Nativo
- Nombre de la clase: table
- Descripción: pertenece a la librería de Lua 5.1 que extiende la funcionalidad de los array, ya que las tablas pueden almacenar varios tipos de datos excepto el vacío o nulo, indexando de manera similar a los arrays
- Propiedades y métodos:

table
+concat()
+copy()
+indexOf()
+insert()
+maxn()
+remove()
+sort()

- Subsistema: Matemática
- Nombre de la clase: math
- Descripción: pertenece a la librería con funciones matemáticas de Lua 5.1 que llaman a la biblioteca estándar de C
- Propiedades y métodos:

- Subsistema: Matemática
- Nombre de la clase: crypto
- Descripción: pertenece a la librería para encriptar y desencriptar mensajes
- Propiedades y métodos:

crypto
+md4
+md5
+sha1
+sha224
+sha256
+sha384
+sha512
+digest()
+hmac()

Continuación (Tabla 6: Clases del motor de videojuegos Corona SDK).

<table border="1"> <thead> <tr> <th>math</th> </tr> </thead> <tbody> <tr><td>+abs()</td></tr> <tr><td>+acos()</td></tr> <tr><td>+asin()</td></tr> <tr><td>+atan()</td></tr> <tr><td>+atan2()</td></tr> <tr><td>+ceil()</td></tr> <tr><td>+cos()</td></tr> <tr><td>+cosh()</td></tr> <tr><td>+deg()</td></tr> <tr><td>+exp()</td></tr> <tr><td>+floor()</td></tr> <tr><td>+fmod()</td></tr> <tr><td>+frexp()</td></tr> <tr><td>+ldexp()</td></tr> <tr><td>+log()</td></tr> <tr><td>+log10()</td></tr> <tr><td>+max()</td></tr> <tr><td>+min()</td></tr> <tr><td>+modf()</td></tr> <tr><td>+pi()</td></tr> <tr><td>+pow()</td></tr> <tr><td>+rad()</td></tr> <tr><td>+random()</td></tr> <tr><td>+randomseed()</td></tr> <tr><td>+round()</td></tr> <tr><td>+sin()</td></tr> <tr><td>+sinh()</td></tr> <tr><td>+sqrt()</td></tr> <tr><td>+tan()</td></tr> <tr><td>+tanh()</td></tr> </tbody> </table>	math	+abs()	+acos()	+asin()	+atan()	+atan2()	+ceil()	+cos()	+cosh()	+deg()	+exp()	+floor()	+fmod()	+frexp()	+ldexp()	+log()	+log10()	+max()	+min()	+modf()	+pi()	+pow()	+rad()	+random()	+randomseed()	+round()	+sin()	+sinh()	+sqrt()	+tan()	+tanh()	
math																																
+abs()																																
+acos()																																
+asin()																																
+atan()																																
+atan2()																																
+ceil()																																
+cos()																																
+cosh()																																
+deg()																																
+exp()																																
+floor()																																
+fmod()																																
+frexp()																																
+ldexp()																																
+log()																																
+log10()																																
+max()																																
+min()																																
+modf()																																
+pi()																																
+pow()																																
+rad()																																
+random()																																
+randomseed()																																
+round()																																
+sin()																																
+sinh()																																
+sqrt()																																
+tan()																																
+tanh()																																
<ul style="list-style-type: none"> • Subsistema: Gestión de Archivos • Nombre de la clase: media • Descripción: pertenece a la librería para acceder a los archivos multimedia del dispositivo (audio, video, fotos) • Propiedades y métodos: <table border="1"> <thead> <tr> <th>media</th> </tr> </thead> <tbody> <tr><td>+RemoteSource</td></tr> <tr><td>+capturePhoto()</td></tr> <tr><td>+captureVideo()</td></tr> <tr><td>+hasSource()</td></tr> <tr><td>+newEventSound()</td></tr> <tr><td>+newRecording()</td></tr> <tr><td>+pauseSound()</td></tr> <tr><td>+playEventSound()</td></tr> <tr><td>+playSound()</td></tr> <tr><td>+playVideo()</td></tr> <tr><td>+save()</td></tr> <tr><td>+selectPhoto()</td></tr> <tr><td>+selectVideo()</td></tr> <tr><td>+setSoundVolume()</td></tr> <tr><td>+stopSound()</td></tr> </tbody> </table>	media	+RemoteSource	+capturePhoto()	+captureVideo()	+hasSource()	+newEventSound()	+newRecording()	+pauseSound()	+playEventSound()	+playSound()	+playVideo()	+save()	+selectPhoto()	+selectVideo()	+setSoundVolume()	+stopSound()	<ul style="list-style-type: none"> • Subsistema: Gestión de Archivos • Nombre de la clase: io • Descripción: pertenece a la librería para crear, escribir y leer archivos • Propiedades y métodos: <table border="1"> <thead> <tr> <th>io</th> </tr> </thead> <tbody> <tr><td>+close()</td></tr> <tr><td>+close()</td></tr> <tr><td>+input()</td></tr> <tr><td>+lines()</td></tr> <tr><td>+open()</td></tr> <tr><td>+output()</td></tr> <tr><td>+read()</td></tr> <tr><td>+tmpfile()</td></tr> <tr><td>+type()</td></tr> <tr><td>+write()</td></tr> </tbody> </table>	io	+close()	+close()	+input()	+lines()	+open()	+output()	+read()	+tmpfile()	+type()	+write()				
media																																
+RemoteSource																																
+capturePhoto()																																
+captureVideo()																																
+hasSource()																																
+newEventSound()																																
+newRecording()																																
+pauseSound()																																
+playEventSound()																																
+playSound()																																
+playVideo()																																
+save()																																
+selectPhoto()																																
+selectVideo()																																
+setSoundVolume()																																
+stopSound()																																
io																																
+close()																																
+close()																																
+input()																																
+lines()																																
+open()																																
+output()																																
+read()																																
+tmpfile()																																
+type()																																
+write()																																

Continuación (Tabla 6: Clases del motor de videojuegos Corona SDK)

- Subsistema: Gestión de Archivos
- Nombre de la clase: json
- Descripción: pertenece a la librería para serializar y deserializar tablas entre JSON y Lua
- Propiedades y métodos:

json
+decode() +encode() +prettify()

- Subsistema: Gestión de Hardware
- Nombre de la clase: system
- Descripción: pertenece a la librería para saber información del dispositivo (orientación de la pantalla, información del dispositivo), también sirve para gestionar hardware como el acelerómetro, el GPS, el multitouch de pantalla, etc.)
- Propiedades y métodos:

system
+orientation +CachesDirectory +DocumentsDirectory +ResourceDirectory +TemporaryDirectory
+activate() +cancelNotification() +deactivate() +getIdleTimer() +getInfo() +getInputDevices() +getPreference() +getTimer() +hasEventSource() +openURL() +pathForFile() +scheduleNotification() +setAccelerometerInterval() +setGyroscopeInterval() +setIdleTimer() +setLocationAccuracy() +setLocationThreshold() +setTapDelay() +vibrate()

- Subsistema: Gestión de Hardware
- Nombre de la clase: native
- Descripción: pertenece a librería que permite manipular hardware específico del dispositivo, este hardware es controlado y la renderización es realizada por el propio dispositivo, no por el motor de Corona SDK
- Propiedades y métodos:

- Subsistema: Gestión de Hardware
- Nombre de la clase: os
- Descripción: pertenece a la librería estándar de Lua 5.1 para saber funciones relacionadas al sistema operativo (hora, fecha, etc.)
- Propiedades y métodos:

os
+clock() +date() +diffTime() +execute() +exit() +remove() +rename() +time()

Continuación (Tabla 6: Clases del motor de videojuegos Corona SDK).

<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 2px;">native</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">+systemFont +systemFontBold</td> </tr> <tr> <td style="padding: 2px;">+cancelAlert() +cancelWebPopup() +canShowPopup() +getFontNames() +getProperty() +getSync() +newFont() +newMapView() +newTextBox() +newTextField() +newVideo() +newWebView() +requestExit() +setActivityIndicator() +setKeyboardFocus() +setProperty() +showAlert() +showPopup() +showWebPopup() +setSync()</td> </tr> </tbody> </table>	native	+systemFont +systemFontBold	+cancelAlert() +cancelWebPopup() +canShowPopup() +getFontNames() +getProperty() +getSync() +newFont() +newMapView() +newTextBox() +newTextField() +newVideo() +newWebView() +requestExit() +setActivityIndicator() +setKeyboardFocus() +setProperty() +showAlert() +showPopup() +showWebPopup() +setSync()		
native					
+systemFont +systemFontBold					
+cancelAlert() +cancelWebPopup() +canShowPopup() +getFontNames() +getProperty() +getSync() +newFont() +newMapView() +newTextBox() +newTextField() +newVideo() +newWebView() +requestExit() +setActivityIndicator() +setKeyboardFocus() +setProperty() +showAlert() +showPopup() +showWebPopup() +setSync()					
<ul style="list-style-type: none"> • Subsistema: Inteligencia Artificial • Nombre de la clase: timer • Descripción: pertenece a librería para llamar a las funciones escritas en el código fuente luego de un periodo determinado de tiempo • Propiedades y métodos: <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th style="text-align: center; padding: 2px;">timer</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">+cancel() +pause() +performWithDelay() +resume()</td> </tr> </tbody> </table>	timer	+cancel() +pause() +performWithDelay() +resume()	<ul style="list-style-type: none"> • Subsistema: Inteligencia Artificial • Nombre de la clase: transition • Descripción: pertenece a librería para manipular el movimiento y las propiedades (tamaño, rotación, transparencia de los objetos) • Propiedades y métodos: <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th style="text-align: center; padding: 2px;">transition</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">+cancel() +from() +pause() +resume() +to() +blink() +dissolve() +fadeIn() +fadeOut() +moveBy() +moveTo() +scaleBy() +scaleTo()</td> </tr> </tbody> </table>	transition	+cancel() +from() +pause() +resume() +to() +blink() +dissolve() +fadeIn() +fadeOut() +moveBy() +moveTo() +scaleBy() +scaleTo()
timer					
+cancel() +pause() +performWithDelay() +resume()					
transition					
+cancel() +from() +pause() +resume() +to() +blink() +dissolve() +fadeIn() +fadeOut() +moveBy() +moveTo() +scaleBy() +scaleTo()					

Continuación (Tabla 6: Clases del motor de videojuegos Corona SDK)

<ul style="list-style-type: none"> • Subsistema: Física • Nombre de la clase: physics • Descripción: pertenece a librería para administrar física que incluye simulación de gravedad, detección de colisiones, etc. • Propiedades y métodos: 	<ul style="list-style-type: none"> • Subsistema: Renderizado • Nombre de la clase: display • Descripción: pertenece a librería para crear y administrar objetos en pantalla • Propiedades y métodos: 						
<table border="1"> <thead> <tr> <th>physics</th> </tr> </thead> <tbody> <tr> <td>+engineVersion</td> </tr> <tr> <td>+addBody() +fromMKS() +getAverageCollisionPositions() +getDebugErrorsEnabled() +getGravity() +getMKS() +getReportCollisionsInContentCoordinates() +newJoint() +newParticleSystem() +pause() +queryRegion() +rayCast() +reflectRay() +removeBody() +setAverageCollisionPositions() +setContinuous() +setDebugErrorsEnabled() +setDrawMode() +setGravity() +setMKS() +setPositionIterations() +setReportCollisionsInContentCoordinates() +setScale() +setTimeStep() +setVelocityIterations() +start() +stop() +toMKS()</td> </tr> </tbody> </table>	physics	+engineVersion	+addBody() +fromMKS() +getAverageCollisionPositions() +getDebugErrorsEnabled() +getGravity() +getMKS() +getReportCollisionsInContentCoordinates() +newJoint() +newParticleSystem() +pause() +queryRegion() +rayCast() +reflectRay() +removeBody() +setAverageCollisionPositions() +setContinuous() +setDebugErrorsEnabled() +setDrawMode() +setGravity() +setMKS() +setPositionIterations() +setReportCollisionsInContentCoordinates() +setScale() +setTimeStep() +setVelocityIterations() +start() +stop() +toMKS()	<table border="1"> <thead> <tr> <th>display</th> </tr> </thead> <tbody> <tr> <td>+actualContentHeight +actualContentWidth +contentCenterX +contentCenterY +contentHeight +contentScaleX +contentScaleY +contentWidth +currentStage +fps +imageSuffix +pixelHeight +pixelWidth +screenOriginX +screenOriginY +statusBarHeight +topStatusBarContentHeight +viewableContentHeight +viewableContentWidth</td> </tr> <tr> <td>+capture() +captureBounds() +captureScreen() +colorSample() +getCurrentStage() +getDefault() +loadRemoteImage() +newCircle() +newContainer() +newEmbossedText() +newEmitter() +newGroup() +newImage() +newImageRect() +newLine() +newPolygon() +newRect() +newRoundedRect() +newSnapshot() +newSprite() +newText() +remove() +save() +setDefault() +setDrawMode() +setStatusBar()</td> </tr> </tbody> </table>	display	+actualContentHeight +actualContentWidth +contentCenterX +contentCenterY +contentHeight +contentScaleX +contentScaleY +contentWidth +currentStage +fps +imageSuffix +pixelHeight +pixelWidth +screenOriginX +screenOriginY +statusBarHeight +topStatusBarContentHeight +viewableContentHeight +viewableContentWidth	+capture() +captureBounds() +captureScreen() +colorSample() +getCurrentStage() +getDefault() +loadRemoteImage() +newCircle() +newContainer() +newEmbossedText() +newEmitter() +newGroup() +newImage() +newImageRect() +newLine() +newPolygon() +newRect() +newRoundedRect() +newSnapshot() +newSprite() +newText() +remove() +save() +setDefault() +setDrawMode() +setStatusBar()
physics							
+engineVersion							
+addBody() +fromMKS() +getAverageCollisionPositions() +getDebugErrorsEnabled() +getGravity() +getMKS() +getReportCollisionsInContentCoordinates() +newJoint() +newParticleSystem() +pause() +queryRegion() +rayCast() +reflectRay() +removeBody() +setAverageCollisionPositions() +setContinuous() +setDebugErrorsEnabled() +setDrawMode() +setGravity() +setMKS() +setPositionIterations() +setReportCollisionsInContentCoordinates() +setScale() +setTimeStep() +setVelocityIterations() +start() +stop() +toMKS()							
display							
+actualContentHeight +actualContentWidth +contentCenterX +contentCenterY +contentHeight +contentScaleX +contentScaleY +contentWidth +currentStage +fps +imageSuffix +pixelHeight +pixelWidth +screenOriginX +screenOriginY +statusBarHeight +topStatusBarContentHeight +viewableContentHeight +viewableContentWidth							
+capture() +captureBounds() +captureScreen() +colorSample() +getCurrentStage() +getDefault() +loadRemoteImage() +newCircle() +newContainer() +newEmbossedText() +newEmitter() +newGroup() +newImage() +newImageRect() +newLine() +newPolygon() +newRect() +newRoundedRect() +newSnapshot() +newSprite() +newText() +remove() +save() +setDefault() +setDrawMode() +setStatusBar()							

Continuación (Tabla 6: Clases del motor de videojuegos Corona SDK)	
<ul style="list-style-type: none"> Subsistema: Renderizado Nombre de la clase: composer Descripción: pertenece a librería para crear y administrar pantallas (interfaces) en Corona SDK Propiedades y métodos: <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <p style="text-align: center;">composer</p> <hr/> <pre> +isDebug +recycleOnLowMemory +recycleOnSceneChange +recycleOnSceneChange +getScene() +getSceneName() +getVariable() +gotoScene() +hideOverlay() +loadScene() +newScene() +removeHidden() +removeScene() +setVariable() +showOverlay() </pre> </div>	<ul style="list-style-type: none"> Subsistema: Renderizado Nombre de la clase: grafics Descripción: pertenece a librería para agregar efectos a los objetos Propiedades y métodos: <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <p style="text-align: center;">grafics</p> <hr/> <pre> +defineEffect() +newImageSheet() +newMask() +newOutline() +newTexture() +releaseTextures() </pre> </div>
<ul style="list-style-type: none"> Subsistema: Renderizado Nombre de la clase: widget Descripción: pertenece a librería para mostrar objetos prediseñados en Corona SDK Propiedades y métodos: <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <p style="text-align: center;">widget</p> <hr/> <pre> +newButton() </pre> </div>	

Fuente: <https://docs.coronalabs.com/api/index.html>

B. Unity 3D

Como puede encontrarse en su página oficial Unity (2015) es la plataforma de desarrollo de videojuegos más potente, permite desarrollar en 2D y 3D, diseñado para aficionados y profesionales de la industria, cuenta con versiones gratuitas y de paga. Permite desarrollar para plataformas móviles como: iOS, Android, Windows Phone y Tizen, para plataformas de escritorio como: Windows, Windows Store, Mac OS X y Linux, para consolas como: Play Station 3 y 4, Play Station Vita, Xbox One y 360 y Nintendo Wii, además de otros como Play Station VR, Oculus Rift, Gear VR, Microsoft Hololens, Navegadores Web y WebGL.

Maneja múltiples formatos de imagen, audio, video y texto, permite la gestión de animación, gráficos, audio, física 2D y 3D, scripting y control de versiones que incluso puede realizarse en la nube. Posee compatibilidad con arquitecturas de 64 bits y también vende gráficos 3D. Para el 2015 la versión 5 es la que se puede obtener desde su página oficial.

C. LibGDX

Framework de desarrollo de videojuegos basado en Java, permite desarrollar para varias plataformas como: Windows, Linux, Mac OS X, Android, iOS, Blackberry y Java Applet. Permite la gestión de gráficos basados en OpenGL ES 2.0, también permite el desarrollo para 2D y 3D. Puede administrar música streaming y efectos de sonido, además incluye un motor de matemática y física, también permite leer y escribir datos en formato Json y XML y posee herramientas propias para crear gráficos (LibGDX, 2013)

Es Open Source y para programar con este motor se necesita tener un conocimiento medio o alto en Java.

D. AndEngine

Framework de desarrollo de videojuegos 2D para la plataforma Android, se basa en OpenGL ES 2.0, usa Java y además posee un motor de física y funciones para detectar colisiones. Incluye librerías para programar videojuegos en red local y con conexión a internet. AndEngine fue desarrollado por una sola persona, Nicolas Gramlich, además es de código abierto y totalmente gratuito. Para aprender a manejar este framework se puede comprar los libros oficiales de AndEngine (Gramlich, 2013).

2.6.2. Lenguajes de programación

Los lenguajes de programación más usados para el desarrollo de aplicaciones móviles según la Universidad Nacional Abierta y a Distancia (2013a) son los siguientes:

- Android Java
- iOS Objective C
- HTML 5
- Windows Phone C#

- Bada C++
- webOS
- Blackberry OS Java, web.

También se tiene el lenguaje de programación Lua, desarrollado óptimamente para el desarrollo de aplicaciones para teléfonos inteligentes.

Lua es un lenguaje de programación estructurado, muy ligero, potente e integrable, fue creado en 1993 en la Pontificia Universidad Católica de Rio en Brasil, maneja una sintaxis procedimental asociada a una semántica extensible y arrays, su escritura se realiza de forma dinámica y se ejecuta mediante la interpretación de código en bytes para una máquina virtual, cuenta con una gestión automática de recolección de basura, con este lenguaje se crearon videojuegos como World of Warcraft y Angry Birds. Lua puede ejecutarse en todas las plataformas que tengan un compilador C, ya sea Windows, Mac, o incluso plataformas móviles como Android, iOS, Windows Phone, Symbian y hasta en microprocesadores embebidos que generalmente requieren ser programados en lenguaje ensamblador o C. Lua es capaz de extender programas escritos en C, C++, Java, C #, Fortran, Ada y Perl y Rubi; aunque Lua no está totalmente orientado a objetos es capaz de proporcionar la herencia y clases, sin embargo es más usado mediante la definición de funciones lo cual permite una programación rápida, además de ello Lua es gratis (Lua, 2014).

2.6.3. Herramientas multimedia

Audacity: es una aplicación para editar audio, sus principales funciones son grabar audio externo o interno al computador, editar archivos de diferentes formatos, convertir grabaciones a sonido digital, modificar características de los audios, etc. (Audacity, 2014).

CorelDRAW x7: según la página oficial de CorelDRAW (2014):

Es una aplicación informática de diseño gráfico vectorial, es decir, que usa fórmulas matemáticas en su contenido. Ésta, a su vez, es la principal aplicación de la suite de programas CorelDRAW Graphics Suite ofrecida por la corporación Corel y que está diseñada para suplir múltiples necesidades, como el dibujo, la maquetación de páginas para impresión y/o la publicación web, todas incluidas en un mismo programa.

Inkscape: es un editor de gráficos vectoriales en formato SVG (Gráfico Vectorizado Escalable), funciona en Windows, Mac OS y Linux, permite crear una gran variedad de gráficos e ilustraciones, logos, iconos y demás. Inkscape es un software libre que permite exportar gráficos en formato SVG y PNG (Inkscape, 2014).

Texture Packer: para la edición de imágenes Sprite, así como imágenes, permite reducir el peso de estas sin bajar la calidad, funciona insertando todas las imágenes en una sola que contenga todas las imágenes que se agregan, para que así se consuma menos memoria. Permite ahorrar memoria también haciendo que el procesador pueda administrar mejor las imágenes para hacer más rápido al videojuego, se pueden importar imágenes en formato: PSD (Photoshop image), SWF (Flash animations), PNG (Portable network graphics), TGA (Targa image file), JPG (Joint photographic experts group), TIFF (Tagged image file format) y BMP (Bitmap). Y permite exportar en los formatos PNG, TGA, JPG, TIFF y BMP, además de: PRV (PowerVR - iPhone image format), PKM (PKM with ETC1), PPM (Netbpm), WebP (Google image format), ATF (Adobe texture format). También recorta transparencias sobrantes de las imágenes y funciona en varios Game Engines como: Unity, Cocos2D, LibGDX, AndEngine, Corona SDK, HTML5 y otros (TexturePacker, 2012).

Tiled: es un editor de mapas para las interfaces o escenas de los videojuegos, está disponible para Windows, Mac OS X, Ubuntu y Fedora, es software libre pero es constantemente actualizado gracias a las donaciones que recibe de sus usuarios. La creación de mapas se hace usando tiles (piezas o losetas), también se pueden usar imágenes grandes y seccionarlas en tiles dentro del editor, permite manejar tiles ortogonales e isométricos. También incluye una capa donde se pueden agregar detectores de colisión a los tiles, para ello usa formas geométricas como cuadrados, círculos y polígonos. Aun no funciona para crear mapas 3D (Tiled, 2006).

Xilisoft Video Converter: como se describe en su página oficial (Xilisoft, 2015), es un programa profesional para la edición de videos y audio, disponible en tres versiones: Ultimate, Platinum y Standard. Es capaz de transformar archivos de video en los formatos: AVI (Audio Video Interleave), MPEG (Moving Pictures Experts Group), WMV (Windows Media Video), DivX, MP4, H.264/AVC, AVCHD (Advanced Video Coding High Definition), MKV (Matroska Video), RM (Real Media), MOV, XviD, 3GP, así también convertir video en audio en formato MP3, WMA, WAV (de WAVE -

Waveform Audio File Format), RA (Real Audio), M4A, AAC (Audio Advanced Coding), AC3, OGG, también permite convertir videos de 2D a 3D; lo más resaltante de este programa es que permite recalibrar propiedades de audio y video como bitrate, calidad de audio, framerate o frecuencia de fotogramas y demás. Los archivos transformados se pueden usar en dispositivos como: PSP, PS3, iPod, iPhone, Apple TV, Xbox, Zune, Archos, Wii, DS, móviles y otros.

MP3 Quality Modifier: como se detalla en su sitio web oficial InspireSoft (2015), este es un programa gratuito y portable para cambiar la calidad de los archivos MP3, permite administrar la frecuencia de muestreo, el modo estéreo y mono estéreo, y la tasa de bits de reproducción, de esta manera se puede reducir a aumentar la calidad y el tamaño de los archivos MP3, también soporta la modificación de archivos MP1 y MP2, su última versión es la 2.53.

Compress PNG: es un compresor de imágenes en formato PNG, JPEG e incluso PDF. Reduce el tamaño de las imágenes al variar la profundidad en bits y la cantidad de colores, esta última variación se puede hacer mediante la configuración avanzada de este programa, funciona de manera online y es totalmente gratuita (Compress PNG, 2015).

Wondershare Filmora Video Editor: es un editor de video, es de paga pero tiene una buena capacidad de edición, permite editar video y su audio, texto y títulos, así también permite agregar efectos y transiciones sobre capas, estabilizar el video, reproducirlo en forma inversa, entre otras funciones permite tomar screenshots de la pantalla del ordenador y permite exportar para varios formatos de video además tiene la funcionalidad para quitar marcas de agua (Filmora, 2010).

VirtualBox: es un potente virtualizador que funciona en arquitecturas de 32 y 64 bits de la familia de procesadores Intel y AMD, está disponible de manera gratuita e incluso con código abierto. Puede ser instalado en los sistemas operativos Windows, Linux, Macintosh y Solaris y soporta sistemas operativos hospedados como: Windows, Linux, Macintosh, Solaris y otros. Incluso ahora permite la virtualización de sistemas operativos como Android (VirtualBox, 2015).

Sublime Text: es un potente editor de texto que tiene características que permiten realizar una codificación más eficiente, posee las funcionalidades de buscar línea y buscar caracteres, edición múltiple y otras. Además permite configurar la sintaxis a casi cualquier lenguaje de programación y está disponible para Mac OS X, Windows

y Linux, viene disponible en una versión de paga y una versión gratuita (Sublime Text, 2015).

Microsoft Office: potente herramienta que incluye productos como Word, Power Point, Excel, MS Project, Microsoft Visio y demás, donde las últimas versiones son de paga; está disponible para arquitecturas de 32 y 64 bits. Permite la instalación en alemán, árabe, chino, coreano, francés, inglés y demás idiomas (Microsoft, 2014).

Java: como se describe en su sitio oficial Java (2015), es una tecnología para programar, compilar y ejecutar programas hechos con el lenguaje de programación Java, a su vez incluye:

- JDK (Java Development Kit): que permite desarrollar programas y applets basados en Java.
- JRE (Java Run Environment): que permite ejecutar programas y applets basados en Java.
- JVM (Java Virtual Machine): que sirve para hacer que programas y applets basados en Java puedan funcionar en hardware creado por diferentes fabricantes.

xCode: es el IDE (Integrated Development Environment) de Apple, que incluye herramientas para desarrollar programas y aplicaciones para Mac, iPhone, iPad, Apple TV, y Apple Watch. Posee una herramienta para desarrollar con facilidad interfaces gráficas y es capaz de compilar lenguajes de programación como: C, C++, Objective-C, Objective-C++, AppleScript y Java. Posee aplicaciones como: el Simulador de iOS, el Cargador de Aplicaciones y otros, muy útiles a la hora de desarrollar y distribuir lo construido con xCode (2012).

App Monitor Performance Tool: es una herramienta de monitoreo disponible para teléfonos inteligentes con sistema operativo Android, con la cual se puede ver los recursos que usan estas aplicaciones, tales como memoria RAM, porcentaje de CPU, consumo de batería, consumo de datos, espacio de almacenamiento, incluso se puede ver qué aplicaciones funcionan en segundo plano y qué aplicaciones requieren demasiados permisos, con App Monitor se puede saber qué aplicaciones desinstalar para mantener optimo el funcionamiento del teléfono (App Monitor, 2015).

2.7. CALIDAD DEL SOFTWARE

En esta sección se describe la ISO 25000 (2011a), conocida como SQuaRE (System and Software Quality Requirements and Evaluation), puesto que es “el resultado de la evolución de otras normas anteriores, especialmente de las normas ISO/IEC 9126, que describe las particularidades de un modelo de calidad del producto software, e ISO/IEC 14598, que abordaba el proceso de evaluación de productos software” (Párrafo 2), la ISO 25000 está compuesta por cinco grupos (ISO 25000, 2011a):

A. ISO/IEC 2500n – División de Gestión de Calidad

Conformada por las normas que describen los modelos y términos referenciados por las demás normas de la familia de la ISO/IEC 25000; está compuesta por:

- ISO/IEC 25000 - Guía de SQuaRE: describe la arquitectura de SQuaRE, los términos, usuarios involucrados y modelos de referencia.
- ISO/IEC 25001 – Planeamiento y administración: establece los planes para evaluar y gestionar los requisitos del software o sistema.

B. ISO/IEC 2501n – División de Modelo de Calidad

Conformada por las normas que describen las características internas, externas y de uso del software; esta división está compuesta por:

- ISO/IEC 25010 – Modelos de calidad del software y sistema: esta norma describe el modelo de calidad del software que incluye las características y subcaracterísticas, también describe el modelo de calidad de uso del software.
- ISO/IEC 25012 – Modelo de calidad de datos: brinda información sobre el modelo para evaluar datos estructurados dentro de un Sistema de Información.

C. ISO/IEC 2502n – División de Medición de Calidad

Describe el modelo para medir la calidad interna, externa y de uso del software, así como la guía para su aplicación; esta división está compuesta por:

- ISO/IEC 25020 – Modelo de referencia de medición y guía: muestra un modelo de referencia de medición de calidad y una guía para que se pueda seleccionar y usar las medidas brindadas por la ISO 25000.

- ISO/IEC 25021 – Elementos de medida de la calidad: describe las métricas básicas para aplicarlas en el ciclo de vida del software.
- ISO/IEC 25022 - Medición de la calidad en el uso: describe las métricas para aplicarlas a la calidad de uso.
- ISO/IEC 25023 - Medición de sistema y la calidad del producto software: describe las métricas para aplicarlas a la calidad del software.
- ISO/IEC 25024 – Medición de la calidad de datos: describe las métricas para aplicarlas a la calidad de datos.

D. ISO/IEC 2503n – División de Requisitos de Calidad

Especifica los requisitos de calidad para obtener los requisitos de calidad del software a desarrollar; está compuesta por la norma:

- ISO/IEC 25030 – Calidad de requerimientos: brinda consejos para especificar los requisitos de calidad del software a desarrollar.

E. ISO/IEC 2504n – División de Evaluación de Calidad

Esta división contiene las normas para evaluar el software; está compuesto por las siguientes normas:

- ISO/IEC 25040 - Modelo de referencia de la evaluación y la guía: describe las entradas, restricciones y recursos para generar las salidas correctas.
- ISO/IEC 25041 - Guía de evaluación para los desarrolladores, compradores y evaluadores independientes: muestra desde el punto de vista de los desarrolladores, clientes y verificadores los requerimientos para evaluar el software.
- ISO/IEC 25042 - Módulos de evaluación: describe qué es un módulo de evaluación, su documentación y contenido.
- ISO/IEC 25045 - Módulo de evaluación de recuperabilidad: describe un módulo para la subcaracterística capacidad de recuperación (recuperabilidad) de la característica Fiabilidad.

Se debe especificar con más detalle la norma “ISO/IEC 25010”, de la ISO/IEC 2501n – División de Modelo de Calidad que describe el modelo para el producto software y

la calidad de uso, expresado en ocho características o categorías que a su vez contienen subcaracterísticas o métricas descritas por la ISO 25000 (2011b):

A. Adecuación Funcional

Representa la capacidad del software para brindar las funciones que satisfacen los requisitos planteados. Posee las siguientes subcaracterísticas:

- Completitud funcional: medida en la que el software posee todas las funcionalidades especificadas en los requisitos.
- Corrección funcional: medida en la que el software genera resultados correctos.
- Pertinencia funcional: medida en la que el software brinda la funcionalidad que satisface los objetivos especificados del usuario.

B. Eficiencia de desempeño

Representa la cantidad en el consumo de recursos del software en condiciones establecidas. Incluye las subcaracterísticas de:

- Comportamiento temporal: mide los tiempos de respuesta y procesamiento del software bajo condiciones establecidas.
- Utilización de recursos: mide la cantidad de recursos usados por el software en condiciones determinadas.
- Capacidad: mide el grado en que un parámetro del software cumple con los requisitos definidos.

C. Compatibilidad

Representa la capacidad de dos sistemas para compartir el mismo entorno de hardware o software e intercambiar información. Incluye las subcaracterísticas de:

- Coexistencia: mide la capacidad del software para funcionar junto con otro software compartiendo los mismos recursos.
- Interoperabilidad: mide la capacidad de dos sistemas para intercambiar información y hacer uso de ella.

D. Usabilidad

Representa la capacidad del software para que los usuarios puedan comprenderlo, usarlo y también para que el software sea atractivo o amigable. Incluye las subcaracterísticas de:

- Capacidad para reconocer su adecuación: mide el grado en que el software le permite reconocer al usuario si es útil para sus necesidades.
- Capacidad de aprendizaje: mide el grado en que el software puede ser aprendido de manera fácil para ser usado por los usuarios.
- Capacidad para ser usado: mide el grado con que el usuario puede usar el software de manera fácil.
- Protección contra errores de usuario: mide el grado en que el software permite que los usuarios no cometan errores.
- Estética de la interfaz de usuario: mide el grado en que la interfaz de software resulta amigable y cómoda al usuario.
- Accesibilidad: capacidad del software para brindar facilidades de uso a personas con discapacidad.

E. Fiabilidad

Representa la capacidad de un sistema para realizar las funciones correctas bajo el tiempo y condiciones establecidas. Incluye las subcaracterísticas de:

- Madurez: es la capacidad del sistema para funcionar fiablemente.
- Disponibilidad: es la capacidad del sistema para estar operativo cuando se lo necesite.
- Tolerancia a fallos: es la capacidad del sistema para seguir funcionando aun cuando existan fallos.
- Capacidad de recuperación: es la capacidad del sistema para recuperar los datos y estados luego de un fallo en hardware o software.

F. Seguridad

Representa la capacidad del sistema para proteger contra accesos no autorizados y modificaciones a los datos e información. Incluye las subcaracterísticas de:

- Confidencialidad: mide la capacidad de proteger contra accesos no autorizados.
- Integridad: mide la capacidad de proteger contra accesos y modificaciones de datos y programas no autorizados.
- No repudio: mide la capacidad de grabar todas las acciones realizadas en el sistema a modo de vigilancia de acciones.
- Responsabilidad: mide la capacidad de realizar el seguimiento de las acciones dentro del sistema de alguna entidad o usuario.
- Autenticidad: mide la capacidad de mostrar la identidad de una entidad o usuario.

G. Mantenibilidad

Representa la capacidad de modificar el software o sistema de manera efectiva, eficiente y fácil. Incluye las subcaracterísticas de:

- Modularidad: es el grado en el que sistema compuesto por módulos discretos pueda ser modificado sin implicar grandes impactos en los demás módulos.
- Reusabilidad: es el grado en el que los componentes del sistema puedan ser usados en otros sistemas o en el desarrollo de otros componentes.
- Analizabilidad: es el grado en el que se puede evaluar el impacto de un cambio sobre el resto del sistema.
- Capacidad para ser modificado: es el grado en el que sistema puede ser modificado de manera fácil y efectiva sin introducir fallos.
- Capacidad para ser probado: es el grado en el que sistema o sus componentes pueden ser fácilmente probados para verificar si satisfacen los requisitos planteados.

H. Portabilidad

Representa la capacidad de un software o sistema para ser transferido de un entorno hardware o software a otro de manera efectiva y eficiente. Incluye las subcaracterísticas de:

- Adaptabilidad: es la capacidad del software para ser migrado de un entorno hardware o software eficientemente.
- Capacidad para ser instalado: es la capacidad del software que permite que pueda ser instalado y desinstalado de manera fácil.
- Capacidad para ser reemplazado: es la capacidad del software para reemplazar a otro que fue diseñado con el mismo propósito y dentro del mismo entorno.

2.8. ESTADO DEL ARTE

Se han realizado contadas investigaciones sobre el desarrollo de un videojuego multiplataforma para dispositivos móviles que consuma bajos recursos de hardware para la reducción de los niveles de estrés de las personas, tanto en nuestro país como en el mundo. Este apartado se focaliza fundamentalmente en los estudios y experiencias nacionales, latinoamericanas, e internacionales debido a la importancia que se le adjudica en esta área de investigación tecnológica de los dispositivos móviles.

2.8.1. Experiencias e investigaciones a nivel mundial

Metodología ontológica para el desarrollo de videojuegos.

Esta tesis propone la metodología ontología como nueva alternativa para el desarrollo de videojuegos con la finalidad de tener una opción más, adicionalmente de las metodologías ágiles que tradicionalmente se usan. La metodología ontológica se basa en especificar el dominio del juego usando herramientas visuales para generar contenido semántico formal, de esta manera la comunicación entre los miembros del equipo de desarrollo es más fluida, cabe resaltar también que la creación de las especificaciones del juego son más fáciles usando esta metodología. Las técnicas que usa esta metodología son: Identificación automática de

componentes de software, Generación Procedimental de Contenido y también Chequeo de inconsistencias.

Como resultado se obtuvo la herramienta visual Rossete, la cual permite apoyar el desarrollo del software en las fases de pre-producción, producción, post-producción y mantenimiento necesarias garantizando siempre un contenido semántico claro y preciso (Llansó, 2014).

En relación al presente informe de tesis se puede ver que esta metodología descrita es bastante útil cuando se trabaja en equipo, sin embargo si solo se tiene una persona que hace los papeles de analista, diseñador, programador entonces no tendría mucho sentido utilizarla.

Videojuego multiplataforma para dispositivos móviles.

En esta tesis se desarrolla un videojuego multiplataforma para teléfonos inteligentes con sistemas operativos Android, iOS, Windows Phone y Blackberry, esto se realizó usando el motor de desarrollo Marmalade SDK que está orientado al desarrollo de aplicaciones en C++, junto con el framework de Cocos2d-x que permite el desarrollo de videojuegos multiplataforma también en C++. Las librerías utilizadas fueron CJON, s3ePushwoosh e lwHTTP y la metodología empleada fue RUP, la cual se adaptó al desarrollo de este videojuego online.

Como resultado se obtuvo un videojuego llamado "Risk" que es un videojuego de estrategia para dos a seis jugadores, la lógica de "Risk2 se basa en la complejidad estadística y el azar, además incorpora un sistema de notificaciones (León, 2014).

Esta tesis muestra la posibilidad de realizar un videojuego multiplataforma para teléfonos inteligentes, haciendo uso de los motores de desarrollo que en los últimos años han empezado a surgir. Además muestra una idea de cómo se puede aplicar la metodología RUP para dicho desarrollo, que cabe resaltar posee artefactos que pueden ser usados en el ámbito de desarrollo de los videojuegos.

Es importante señalar que en esta tesis no se hace mención sobre técnicas para mejorar la adaptabilidad del juego a diversos tipos de dispositivos, incluyendo principalmente a los dispositivos de gama baja que poseen pocas funcionalidades y no son capaces de soportar la ejecución de videojuegos que requieren altos niveles de hardware.

2.8.2. Experiencias e investigaciones en Latinoamérica

Juego de Ajedrez sobre dispositivos móviles con sistema operativo Android para personas con limitación visual.

La tesis desarrollada por Montoya, Suárez y Ortiz (2014) presenta el desarrollo de un juego de para personas con problemas visuales, dicho juego es el conocido Ajedrez.

La metodología de desarrollo del juego fue la XP (Programación extrema), la cual es una metodología ágil con conceptos iterativos e incrementales, además se hizo uso de una de sus características que es la adaptabilidad a cambios en los requisitos del usuario.

Se utilizó la arquitectura cliente - servidor para el desarrollo, ya que tiene la ventaja de aligerar la carga de procesamiento de información gracias a la implementación adicional de multiusuarios que trabajan de manera independiente físicamente pero juntos lógicamente. Cabe resaltar que la herramienta para los diseños usada fue Enterprise Architect, la cual es una herramienta para diseñar y modelar casos de uso. También se usó el lenguaje de programación Java y el SDK de Android.

En relación al trabajo de investigación se puede resaltar que el uso del SDK de Android para la programación consume muchos recursos de hardware, es por ello se debe buscar otras herramientas que no sean tan costosas en el tema de hardware. También se puede hacer énfasis en el uso de la metodología XP e indicar que tuvo un buen uso para el desarrollo del videojuego.

Videojuegos: Más allá de la mente.

Este artículo escrito por Betancourth (2007) describe la influencia que tienen los videojuegos en la mente y la vida de las personas, menciona que para los programadores, un videojuego lleva su mente hacia la creación de procesos creativos y que además le exige que se reúna con un equipo para ser capaces de abstraer la realidad del mundo y plasmarla en videojuegos, lo cual incluye también la simulación, para que por medio del pensamiento colectivo se obtenga un producto que llame la atención de los usuarios.

Se habla también sobre los efectos de los videojuegos tales como la liberación de estrés, y la reducción del dolor al llevar la atención del paciente a los objetivos del videojuego que esté jugando.

Sin embargo para los niños se menciona que se debe tener un control especial tanto en horas, tipo y lugares de juego, ya que de no ser así se producirían efectos negativos en su personalidad y su vida académica.

En relación a la investigación realizada se puede comprobar los efectos beneficiosos en la salud psicología de aquellas personas que juegan videojuegos, además hace tener presente la responsabilidad con los niños y el control adecuado que se les debe dar ante el uso de los videojuegos.

2.8.3. Experiencias e investigaciones en Perú

Desarrollo de la mecánica y dinámica de un videojuego serio 3D en tercera persona.

En esta tesis de Fernández (2014) se desarrolla un videojuego cuya dinámica se basa en aportar información a todos los usuarios, especialmente a aquellas personas jóvenes que no conocen la historia de la época del terrorismo en el Perú, mediante un juego en tercera persona donde el jugador puede moverse por las calles diseñadas en el videojuego y ser testigo de cómo personajes dotados con inteligencia artificial comenten actos violentos contra otros personajes (la población), además el usuario que observa todo esto no puede hacer mucho por ayudar a la población más que completar unas pequeñas misiones.

Para su realización se hizo la comparación de la metodología RUP y XP, y se optó por usar la metodología RUP debido a que el tesista contaba con mayor conocimiento de dicha metodología, también se usó el motor de desarrollo Unity 3D, y las herramientas Visual Paradigm UML para los diseños, así como el IDE MonoDevelop. Mientras que para el control de los personajes del juego se usó el algoritmo A* que es un algoritmo de búsqueda.

Finalmente se logró obtener el juego que funciona vía web donde todas las personas, especialmente los jóvenes pueden acceder a dicho juego.

Esta tesis permite conocer una vez más la potencialidad que posee la metodología RUP para el desarrollo de software lo cual incluye también los videojuegos, además muestra nuevas herramientas de diseño y modelado que se pueden utilizar, también nos permiten conocer como poder adicionar inteligencia artificial al videojuego así como usar algoritmos que sean más eficientes y puedan hacer que el videojuego sea más liviano e interesante.

Diseño e implementación de una plataforma interactiva móvil orientada al aprendizaje temas de física.

En la presente tesis realizada por KU (2013) se desarrolla un videojuego con fines educativos bajo la plataforma Android, este videojuego se basa en la temática de la física, se pretende mejorar el aprendizaje de los estudiantes mediante el uso de este videojuego como complemento para la parte teórica impartida por el profesor de física y así mejorar la estrategia pedagógica de enseñanza de esta materia tan importante. Para ello se usó el motor de desarrollo de videojuegos Unity 3D, así como el SDK de Android y la librería Orthello 2D mientras que las herramientas fueron Adobe Photoshop, Adobe Illustrator, Blender y OpenGameArt para los diseños. En relación con la investigación realizada se puede afirmar que al no mostrarse la metodología que se aplicó para el desarrollo del juego de la tesis presentada, se puede asumir que se utilizó una metodología llamada codificar y corregir la cual consiste en ponerse a programar directamente e ir obteniendo resultados inmediatos, es posible utilizar esta metodología pero solo es recomendable para proyectos pequeños.

A partir de las investigaciones y experiencias expuestas en este apartado queda en evidencia la necesidad de ahondar en la mejora del rendimiento de las aplicaciones, especialmente los videojuegos para que puedan funcionar en dispositivos móviles inteligentes de gama baja, así como garantizar su funcionamiento en plataformas como Android e iOS.

3. INGENIERÍA DEL PROYECTO

La ingeniería del proyecto para el desarrollo de un videojuego multiplataforma para teléfonos inteligentes que consume bajos recursos de hardware para reducir los niveles de estrés, consta de la gestión del proyecto, la metodología para el desarrollo, las herramientas utilizadas, la gestión de la calidad, la optimización de uso de recursos, las distribución multiplataforma y la metodología de reducción de estrés, las cuales se detallan a continuación:

3.1. GESTIÓN DEL PROYECTO

Se usó la metodología de gestión de proyecto integrada en la metodología SUM, la cual incluye los siguientes planes y áreas:

3.1.1. Gestión del alcance

Aquí se detalló lo que la ejecución del proyecto buscó alcanzar, ello se menciona a continuación:

Construcción un videojuego de género casual, que pueda ser instalado en teléfonos inteligentes de gama baja con sistema operativo Android e iOS a partir del mismo código fuente.

Dicho videojuego será en dos dimensiones y constará de 20 niveles, además tendrá interfaces como: bienvenida, menú principal, menú de selección de niveles, interfaz de juego, interfaz de pausa e interfaz que se muestra cuando un nivel es resuelto. Además el videojuego tendrá la capacidad de reducir el estrés de la personas. Más detalle fue incluido durante la fase de concepto y planificación.

3.1.2. Plan de personal

Para la realización del presente proyecto se requirió de cinco personas, el tesista, quien desarrolló cada una de las fases de la metodología a lo largo de la ejecución del proyecto, el asesor, quien supervisó y apoyó durante la planificación y ejecución del proyecto de tesis, y el equipo de Psicólogos, quienes apoyaron en la medición de reducción de estrés, la siguiente tabla detalla esta información:

Tabla 7: *Plan de personal para el proyecto*

Personal	Nombre	Rol	Dedicación
Tesista	Iván Marcial Palomino Taype	Productor interino	Durante cada mes del proyecto
Tesista	Iván Marcial Palomino Taype	Diseñador del juego	Mes: 2, 7 y 8
Tesista	Iván Marcial Palomino Taype	Programador	Mes: del 3 al 8
Tesista	Iván Marcial Palomino Taype	Artista gráfico	Mes: 2, 3, 7 y 8
Tesista	Iván Marcial Palomino Taype	Artista sonoro	Mes: 7 y 8
Tesista	Iván Marcial Palomino Taype	Verificador Beta	Mes: 7
Ingeniero de sistemas	Juan José Oré Cerrón	Asesor	Durante cada mes del proyecto
Equipo de Psicólogos	Carmen R. Salas Zúñiga Mirella Morales del Campo Efraín Altamirano Moran	Psicólogos	Mes: 9

Fuente: elaboración propia

3.1.3. Plan de recursos

Se listan los costos incurridos en la ejecución del proyecto, así como el momento en que fueron requeridos, dichos costos abarcan costo de equipos y software. También se listan aquellos equipos que ya se tenían adquiridos, lo cual incluye, una laptop, teléfonos inteligentes y demás, los siguientes cuadros muestran dicha planificación:

Tabla 8: Plan de recursos, recursos por adquirir

RECURSOS POR ADQUIRIR		
	Costo	Mes de adquisición
Herramientas hardware		
Smartphone con SO iOS	S/. 550.00	3er mes
Herramientas software		
Licencia Corona SDK v 2014.2511	S/. 257.00	1er mes
Wondershare video editor v 3.1.1	S/. 98.00	8vo mes
Microsoft Office 2013	S/. 326.00	1er mes
Licencia Xilisoft Video Converter v 7.8	S/. 147.00	3er mes
CorelDRAW Graphics Suite X7	S/. 1,834.00	2do mes
Licencia para distribución en Play Store	S/. 82.00	8vo mes
Licencia para distribución en App Store	S/. 323.00	3er mes
Servicios		
Internet	S/. 350.00	Durante cada mes del proyecto
Impresión y anillado	S/. 253.00	1er, 8vo y 9no mes
Movilidad	S/. 200.00	Durante cada mes del proyecto
Otros		
Imprevistos	S/. 2,000.00	Durante cada mes del proyecto
Viáticos	S/. 1,000.00	Durante cada mes del proyecto
Sub total	S/. 7,420.00	

Fuente: Elaboración propia.

Tabla 9: Plan de recursos, recursos ya adquiridos

RECURSOS YA ADQUIRIDOS		
	Costo	Mes de adquisición
Herramientas hardware		
Computadora Lenovo	S/. 1,800.00	1er mes
Smartphone con SO Android	S/. 400.00	3er mes
Recurso humano		
Asesor	S/. 2,500.00	Durante cada mes del proyecto
Diseñador	S/. 770.00	2do, 7mo y 8vo mes
Programador	S/. 8,460.00	del 3er al 8vo mes
Artista sonoro	S/. 810.00	7mo y 8vo mes
Artista gráfico	S/. 980.00	2do, 3er, 7mo y 8vo mes
Verificador beta	S/. 750.00	7mo mes
Sub total	S/. 16,470.00	

Fuente: Elaboración propia.

3.1.4. Plan presupuestal

Se planeó el gasto que fue requerido para el desarrollo del proyecto por mes, estos costos incluyeron: costos de adquisición de herramientas hardware, costo de adquisición de herramientas software como licencia de uso y distribución, servicios de terceros y otros, la siguiente tabla muestra dicha planificación:

Tabla 10: Plan presupuestal por mes

RECURSOS POR ADQUIRIR	MESES									
	1	2	3	4	5	6	7	8	9	10 y 11
Herramientas hardware										
Smartphone con SO iOS			S/. 550.00							
Herramientas software										
Licencia Corona SDK	S/. 257.00									
Wondershare video editor								S/. 98.00		
Microsoft Office 2013	S/. 326.00									
Licencia Xilisoft Video Converter			S/. 147.00							
CorelDRAW Graphics Suite X7		S/. 1,834.00								
Licencia para distribución en Play Store								S/. 82.00		
Licencia para distribución en App Store			S/. 323.00							
Servicios										
Internet	S/. 35.00	S/. 35.00	S/. 35.00	S/. 35.00	S/. 35.00	S/. 35.00	S/. 35.00	S/. 35.00	S/. 35.00	S/. 35.00
Impresión y anillado	S/. 53.00								S/. 100.00	S/. 100.00
Movilidad	S/. 20.00	S/. 20.00	S/. 20.00	S/. 20.00	S/. 20.00	S/. 20.00	S/. 20.00	S/. 20.00	S/. 20.00	S/. 20.00
Otros										
Imprevistos	S/. 200.00	S/. 200.00	S/. 200.00	S/. 200.00	S/. 200.00	S/. 200.00	S/. 200.00	S/. 200.00	S/. 200.00	S/. 200.00
Viáticos	S/. 100.00	S/. 100.00	S/. 100.00	S/. 100.00	S/. 100.00	S/. 100.00	S/. 100.00	S/. 100.00	S/. 100.00	S/. 100.00
Inversión mensual	S/. 991.00	S/. 2,189.00	S/. 1,375.00	S/. 355.00	S/. 355.00	S/. 355.00	S/. 355.00	S/. 535.00	S/. 455.00	S/. 455.00
Sub total	S/. 7,420.00									

Fuente: Elaboración propia.

3.1.5. Cronograma e hitos

Se desarrolló el cronograma que incluyó la fecha de realización de las tareas e iteraciones que componen cada fase de la metodología, también se detalló la duración, inicio y fin así como los hitos generados al término de la realización de dichas tareas e iteraciones, ello es mostrado a continuación:

Nombre de tarea	Duración	Comienzo	Fin	Hito
DESARROLLO DEL PROYECTO	247 días	lun 04/05/15	mar 12/04/16	
Elaboración y aprobación del Plan de Tesis	20 días	lun 04/05/15	vie 29/05/15	Plan de tesis aprobado
DESARROLLO DEL VIDEOJUEGO	135 días	lun 01/06/15	vie 04/12/15	
Fase de concepto	5 días	lun 01/06/15	vie 05/06/15	
Definición del concepto del videojuego	5 días	lun 01/06/15	vie 05/06/15	Documento de concepto del videojuego
Fase de planificación	10 días	lun 08/06/15	vie 19/06/15	
Realización de la planificación de la administración	3 días	lun 08/06/15	mié 10/06/15	Documento de la planificación de la administración
Definición de requisitos funcionales y no funcionales	5 días	jue 11/06/15	mié 17/06/15	Documento de especificación requisitos funcionales y no funcionales
Evaluación y priorización de requisitos funcionales y no funcionales	2 días	jue 18/06/15	vie 19/06/15	Documento de priorización de requisitos funcionales y no funcionales
Fase de diseño	10 días	lun 22/06/15	vie 03/07/15	
Diseño de las interfaces	7 días	lun 22/06/15	mar 30/06/15	Documento de diseño de la interfaces
Diseño de los paquetes	3 días	mié 01/07/15	vie 03/07/15	Diseño de paquetes
Fase de elaboración	90 días	lun 06/07/15	vie 06/11/15	
Iteración 1	10 días	lun 06/07/15	vie 17/07/15	Interfaz primer splash, segundo splash, menú principal y menú de niveles
Iteración 2	10 días	lun 20/07/15	vie 31/07/15	Interfaz over relay de menú principal, pausa y nivel
Iteración 3	10 días	lun 03/08/15	vie 14/08/15	Interfaz de primer y segundo tutorial e interfaz de créditos
Iteración 4	10 días	lun 17/08/15	vie 28/08/15	Niveles 1, 2 y 3
Iteración 5	10 días	lun 31/08/15	vie 11/09/15	Niveles 4, 5, 6, 7 y 8
Iteración 6	10 días	lun 14/09/15	vie 25/09/15	Niveles 9, 10, 11 y 12
Iteración 7	10 días	lun 28/09/15	vie 09/10/15	Niveles 13, 14 y 15
Iteración 8	10 días	lun 12/10/15	vie 23/10/15	Niveles 16, 17 y 18
Iteración 9	10 días	lun 26/10/15	vie 06/11/15	Niveles 19 y 20

Figura 12: Cronograma e hitos, parte 1

Fuente: elaboración propia

Nombre de tarea	Duración	Comienzo	Fin	Hito
▲ Fase beta	15 días	lun 09/11/15	vie 27/11/15	
Iteración 1	2 días	lun 09/11/15	mar 10/11/15	Plan de revisión del videojuego
Iteración 2	10 días	mié 11/11/15	mar 24/11/15	Casos de prueba de las interfaces
Iteración 3	3 días	mié 25/11/15	vie 27/11/15	Lista de errores corregidos
▲ Cierre del proyecto	5 días	lun 30/11/15	vie 04/12/15	
Evaluación del proyecto	3 días	lun 30/11/15	mié 02/12/15	Lista de lecciones aprendidas
Liberación del videojuego	1 día	jue 03/12/15	jue 03/12/15	Videojuego desarrollado
Despliegue del videojuego	1 día	vie 04/12/15	vie 04/12/15	Modelo de despliegue del videojuego
▲ DESARROLLO CON OPTIMIZACIÓN EN EL CONSUMO DE RECURSOS	10 días	lun 07/12/15	vie 18/12/15	
Optimización de recursos de audio	5 días	lun 07/12/15	vie 11/12/15	Videojuego con audio optimizado
Optimización de recursos gráficos	5 días	lun 14/12/15	vie 18/12/15	Videojuego con gráficos optimizados
▲ DISTRIBUCIÓN MULTIPLATAFORMA	15 días	lun 21/12/15	vie 08/01/16	
Distribución para Android	3 días	lun 21/12/15	mié 23/12/15	Instalador del videojuego para Android, videojuego subido a la Play Store
Distribución para iOS	12 días	jue 24/12/15	vie 08/01/16	Instalador del videojuego para iOS, videojuego subido a la App Store
▲ METODOLOGÍA DE REDUCCIÓN DE ESTRÉS	17 días	lun 11/01/16	mar 02/02/16	
Planificación de la medición de la reducción de Estrés	3 días	lun 11/01/16	mié 13/01/16	Documento del plan de medición de reducción de Estrés
Elaboración y validación del Test de Reducción de Estrés	4 días	jue 14/01/16	mar 19/01/16	Test de medición de la reducción del Estrés validado
Aplicación del Test de medición de reducción de Estrés	10 días	mié 20/01/16	mar 02/02/16	Resultados de la aplicación del Test de reducción de Estrés
Elaboración del informe de tesis	29 días	mié 03/02/16	lun 14/03/16	Informe de Tesis y anexos
Revisión del informe y levantamiento de observaciones	18 días	mar 15/03/16	jue 07/04/16	Informe de Tesis revisado
Sustentación y defensa de la tesis	1 día	mar 12/04/16	mar 12/04/16	Aprobación de la Tesis

Figura 13: Cronograma e hitos, parte 2

Fuente: Elaboración propia

3.1.6. Seguimiento del proyecto

Se elaboró el documento de seguimiento y cumplimiento de las tareas e iteraciones del proyecto así como el encargado de realizarla, ello permitió mostrar qué porcentaje del proyecto realizó, de esta manera se evitó retrasos, lo cual se muestra a continuación en las siguientes dos imágenes:

Nombre de tarea	Encargado	% completado
DESARROLLO DEL PROYECTO		100%
Elaboración y aprobación del Plan de Tesis	Tesista y Jurados	100%
DESARROLLO DEL VIDEOJUEGO		100%
 Fase de concepto		100%
Definición del concepto del videojuego	Tesista (Cliente)	100%
 Fase de planificación		100%
Realización del planificación de la administración	Tesista (Productor Interino)	100%
Definición de requisitos funcionales y no funcionales	Tesista (Diseñador)	100%
Evaluación y priorización de requisitos funcionales y no funcionales	Tesista (Diseñador)	100%
 Fase de diseño		100%
Diseño de las interfaces	Tesista (Artista gráfico)	100%
Diseño de los paquetes	Tesista (Artista gráfico)	100%
 Fase de elaboración		100%
Iteración 1	Tesista (Programador)	100%
Iteración 2	Tesista (Programador)	100%
Iteración 3	Tesista (Programador)	100%
Iteración 4	Tesista (Programador)	100%
Iteración 5	Tesista (Programador)	100%
Iteración 6	Tesista (Programador)	100%
Iteración 7	Tesista (Programador)	100%
Iteración 8	Tesista (Programador)	100%
Iteración 9	Tesista (Programador)	100%
 Fase beta		100%

Figura 14: Seguimiento del proyecto, parte 1

Fuente: Elaboración propia

Nombre de tarea	Encargado	% completado
Iteración 1	Tesista (Verificador beta)	100%
Iteración 2	Tesista (Verificador beta)	100%
Iteración 3	Tesista (Verificador beta)	100%
▸ Cierre del proyecto		100%
Evaluación del proyecto	Tesista (Equipo de desarrollo)	100%
Liberación del videojuego	Tesista (Equipo de desarrollo)	100%
Despliegue del videojuego	Tesista (Productor Interino)	100%
▸ DESARROLLO CON OPTIMIZACIÓN EN EL CONSUMO DE RECURSOS		100%
Optimización de recursos de audio	Tesista (Artista sonoro)	100%
Optimización de recursos gráficos	Tesista (Artista grafico)	100%
▸ DISTRIBUCIÓN MULTIPLATAFORMA		100%
Distribución para Android	Tesista	100%
Distribución para iOS	Tesista	100%
▸ METODOLOGÍA DE REDUCCIÓN DE ESTRÉS		100%
Planificación de la medición de la reducción de Estrés	Tesista	100%
Elaboración y validación del Test de Reducción de Estrés	Tesista y Equipo de Psicólogos	100%
Aplicación del Test de medición de reducción de Estrés	Tesista	100%
Elaboración del informe de tesis	Tesista	100%
Revisión del informe y levantamiento de observaciones	Tesista y Jurados	100%
Sustentación y defensa de la tesis	Tesista y Jurados	100%

Figura 15: Seguimiento del proyecto, parte 2

Fuente: elaboración propia

3.1.7. Gestión de riesgos

Se listaron los riesgos que pudieron afectar el desarrollo del proyecto y se creó un plan de contingencia que de ser necesario se aplique al proyecto, de esta manera se garantizó el éxito del desarrollo normal del proyecto.

A continuación se muestra la lista de riesgos:

- Cambios en la moneda: considerado como riesgo ALTO porque la adquisición de licencias para distribución del videojuego y el motor de desarrollo tiene costo en moneda extranjera (dólares).
- Inflación: considerado como riesgo ALTO porque afecta seriamente la adquisición de los equipos de cómputo y smartphones.

- Cambios: considerado como riesgo BAJO ya que el diseño del videojuego puede estar sujeto a cambios necesarios para poder ser optimizado.
- Fallos en equipos: considerado como riesgo BAJO porque si los equipos fallan, retrasan al proyecto, ello afecta el tiempo y el costo.
- Requisitos de permisos: considerado como riesgo BAJO en el caso que en el Perú se tome la decisión de no permitir el normal funcionamiento de las tiendas virtuales Play Store y App Store.
- Obsolescencia: considerado como riesgo MEDIO, ya que los equipos tecnológicos cada vez tienden a ser remplazados con equipos de mayor capacidad y en periodos de tiempo cada vez más cortos.

La siguiente tabla muestra la probabilidad, impacto y riesgo, se tomó en cuenta la escala (1 = BAJO, 2 = MEDIO y 3 = ALTO):

Tabla 11: *Evaluación de la lista de riesgos*

Riesgo	Probabilidad	Impacto	Riesgo
Cambios en la moneda	3	3	6
Inflación	3	3	6
Cambios	1	1	2
Fallos en equipos	1	1	2
Requisitos permisos	1	1	2
Obsolescencia	2	2	4

Fuente: Elaboración propia.

Por los resultados obtenidos de riesgo: 6, 6, 2, 2, 2 y 4. Se considera riesgo bajo para el proyecto.

También se muestra el plan de contingencia que contiene las acciones necesarias para mitigar los riesgos posibles que ya fueron detallados:

Tabla 12: *Plan de contingencia para mitigar riesgos*

Riesgo	Acción
Cambios en la moneda	Usar el presupuesto para imprevistos
Inflación	Usar el presupuesto para imprevistos
Cambios	Ampliar el tiempo de ejecución del proyecto
Fallos en equipos	Usar el presupuesto para imprevistos y ampliar el tiempo de ejecución del proyecto
Requisitos de permisos	Realizar el despliegue del videojuego de manera local
Obsolescencia	Modificar los requisitos del diseño del videojuego

Fuente: Elaboración propia.

3.2. METODOLOGÍA DE DESARROLLO

La metodología que se empleó fue SUM, la cual es una adaptación de la metodología SCRUM para los videojuegos, se combinó con algunas partes de la metodología GUP y RUP con la finalidad de lograr un mejor entendimiento y desarrollo, las razones primordiales para optar por esta metodología se listan a continuación:

- Permite trabajar con equipos de desarrollo pequeños, incluso de 2 integrantes.
- Permite desarrollar proyectos en tiempos menores a un año.
- Es altamente adaptable a diversas características de un videojuego, tales como videojuegos casuales, de acción, de rol y demás.
- Requiere documentación en cantidad moderada.
- Permite combinarse con otras metodologías para mejorar su capacidad y adaptación.

Las fases realizadas son las siguientes:

3.2.1. Concepto

En esta fase el rol de equipo, cliente, verificador beta y productor interino fue asumido por una sola persona, es decir quien ejecutó este proyecto de tesis. El flujo de las tareas realizadas se muestra a continuación:

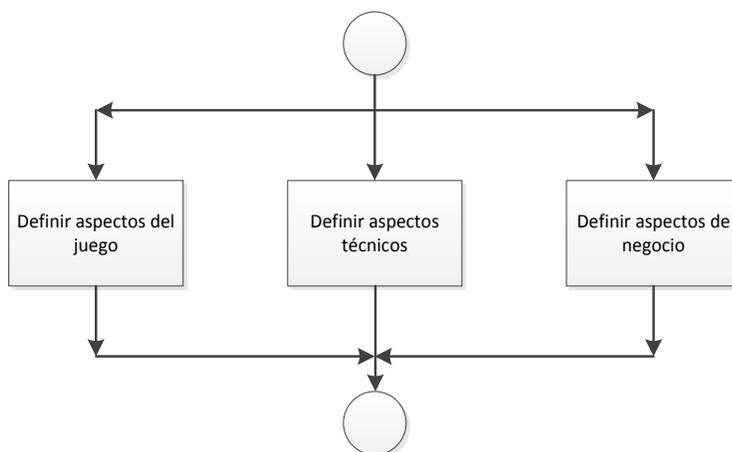


Figura 16. Flujo del desarrollo del concepto

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de estas tareas se muestra a continuación:

A. Definición de aspectos del juego

- **Genero del videojuego:** casual.
- **Subcategoría del videojuego:** puzzle.
- **Lógica del videojuego:** el nombre del videojuego es Casual Labyrinth (Laberinto Casual en español), este videojuego se desarrolló en un laberinto en dos dimensiones (2D), el cual posee física del mundo real, lo que le permite simular un ambiente natural, donde el usuario debe ubicar bloques que se le indiquen en las posiciones establecidas dentro del laberinto, para lograr esto el usuario debe ser capaz de cambiar la dirección de la física en cuatro direcciones (arriba, abajo, derecha e izquierda) mediante el uso de cuatro botones que corresponden a cada dirección de la física.

El videojuego consta de 20 niveles en donde a medida que se vayan superando los niveles, la dificultad para resolver el puzzle irá aumentando. Cabe mencionar que solo se puede jugar un nivel si el anterior fue solucionado, exceptuando el nivel 1. El videojuego está disponible en los idiomas Español e Inglés.

- **Puntos clave vinculados a la reducción de estrés:** para maximizar los efectos reductores del estrés se realizaron los siguientes puntos:
 - Se evitó el uso de contadores de tiempo que terminan con el videojuego haciendo perder al usuario.
 - El videojuego siempre premia al usuario aun cuando éste no juegue óptimamente.
 - El videojuego no cuenta con situaciones en las que el personaje muere o es destruido.
 - El videojuego ayuda al usuario cuando no puede resolver un nivel con alta dificultad.
 - El videojuego incita al usuario a realizar cálculos mentales y de proyección para solucionar cada nivel, enfocando su atención al videojuego y no a otras cosas como problemas o preocupaciones.

- El videojuego cuenta con el equilibrio necesario entre dificultad y habilidad requerida para resolver cada nivel, así el videojuego no es muy difícil ni muy fácil.
- El personaje del videojuego es una carita feliz con la finalidad de proyectar alegría y felicidad al usuario, además no cuenta con imágenes sprite con la finalidad de focalizar la atención en la carita feliz.
- La música de fondo es instrumental y de carácter alegre y relajante.
- El videojuego cuenta con la combinación de varios colores para proyectar visualmente vida, ánimo y vigor.
- El videojuego posee opciones para pausar y reiniciar el videojuego y así hacer que el usuario tenga varias opciones cuando juega.
- El videojuego cuenta con instrucciones que permiten aprender a jugarlo rápidamente y así evitar confusión en el usuario.
- **Interfaces del videojuego:** se cuenta con las siguientes interfaces:
 - Primer Splash o presentación del equipo de desarrollo del videojuego.
 - Segundo Splash o presentación del nombre del videojuego
 - Menú principal.
 - Menú de niveles.
 - Over de Menu principal (sobre el menú principal).
 - Over de pausa (sobre el escenario del videojuego).
 - Over de nivel (sobre el escenario del videojuego).
 - Primer tutorial.
 - Segundo tutorial.
 - Escenario del videojuego.
 - Créditos de desarrollo del videojuego.
- **Objetivos del videojuego:** el objetivo de la lógica del videojuego, es captar la atención del usuario para así “desconectarlo” del mundo y brindarle un momento de distracción y relaxo. El objetivo técnico del videojuego es que este consuma

pocos recursos de los teléfonos inteligentes (memoria RAM, procesamiento de CPU, consumo de batería y espacio de almacenamiento de instalación) y que funcione en los sistemas operativos Android e iOS.

B. Definición de aspectos técnicos

- **Plataformas del videojuego:** se puede jugar en teléfonos inteligentes con sistema operativo Android desde la versión 2.3.3 en adelante y en teléfonos inteligentes con sistema operativo iOS desde la versión 6.0.
- **Tecnologías y herramientas:** se usaron las siguientes:
 - Motor de desarrollo de videojuegos Corona SDK v 2014.2511.
 - Lenguaje de programación Lua.
 - Editor de imagen CorelDRAW x7.
 - Compresor de imágenes online Compress PNG.
 - Editor de audio Audacity v 2.1.
 - Editor de audio Xilisoft Video Converter v 7.8.12.
 - Compresor de audio MP3QualityModifier v 2.53.
 - Editor de video Wondershare video editor v 3.1.1.
 - Virtualizador de máquinas Virtuales VirtualBox v 4.3.26.
 - Herramienta de programación para iOS xCode v 6.4.
 - Editor de texto Sublime v 2.
 - Paquete Microsoft Office 2013 (Word, Excel, PowerPoint, Project).
 - Java JDK (Java Development Kit) v 1.8.45.
 - Editor de mapas Tiled v 0.12.
 - App Monitor Performance Tool v 2.1.4.

C. Definición de aspectos del negocio

- **Público objetivo:** el videojuego está dirigido para personas tanto hombres como mujeres a partir de los 12 años, esto porque los menores a esta edad no poseen aún muchos problemas de estrés o fatiga laboral, sin embargo también es apto si ellos deciden jugar a este tipo de videojuegos casuales tipo puzzle.

- **Alcance del videojuego:** es importante mencionar que el videojuego está disponible para ser descargado en las tiendas principales de Android (Play Store) e iOS (App Store), esto permite que cualquier persona, incluso de otros países tenga acceso al beneficio ofrecido por el videojuego.

3.2.2. Planificación

Dentro de la fase de planificación se desarrolló el plan del proyecto lo cual incluyó realizar actividades de la planificación de la administración y la especificación del videojuego.

A. Planificación de la administración

Se realizaron las tareas de definir el cronograma y el presupuesto para la ejecución de todas las fases, y el equipo de desarrollo del proyecto, dichas tareas se muestran a continuación:

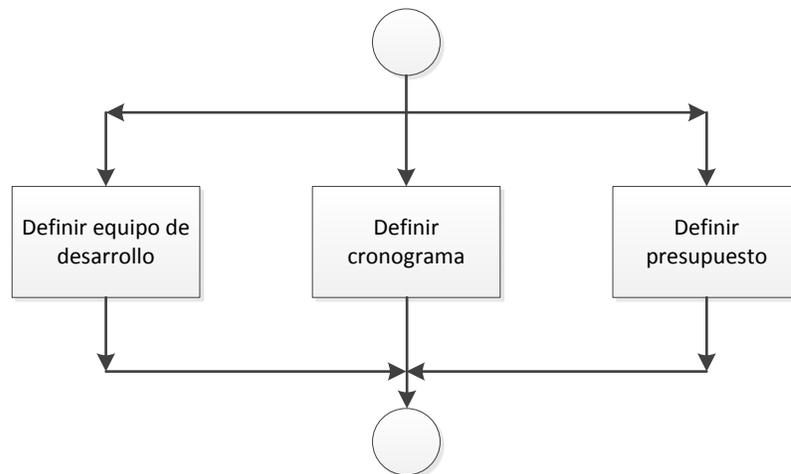


Figura 17. Flujo de tareas de la planificación de la administración

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de estas tareas se muestra a continuación:

- **Definición del equipo de desarrollo**

Para el desarrollo del videojuego planteado se requirió de los siguientes especialistas, divididos en dos grupos:

Grupo de administración: conformado por:

- Productor interno: quien manejó correctamente todo el proceso de desarrollo del videojuego, así como la gestión con el revisor.
- Verificador beta: quien de revisó adecuadamente el videojuego desarrollado.

Grupo de desarrollo: conformado por:

- Diseñador del videojuego: quien creó el gameplay o lógica del videojuego, género, niveles y funcionalidad de este.
- Programador: quien manejó el lenguaje “Lua” con el editor “Sublime” y el motor “Corona SDK”, para desarrollar el videojuego diseñado.
- Artista sonoro: quien generó la música adecuada para el videojuego de género casual.
- Artista gráfico: quien creó las imágenes multimedia específicas para el videojuego diseñado.

- **Definición del cronograma**

Que incluye el cronograma de las fases de concepto, planificación, diseño, elaboración, beta y cierre:

Cronograma de concepto

Tabla 13: *Cronograma de la fase de concepto*

Nombre	Fecha de inicio	Fecha fin	Duración	Hito
Concepto				
Definición del concepto del videojuego	lun 01/06/15	vie 05/06/15	5 días	Documento de concepto del videojuego (véase sección 3.2.1. Concepto)

Fuente: elaboración propia

Cronograma de planificación

Tabla 14: *Cronograma de la fase de planificación*

Nombre	Fecha de inicio	Fecha fin	Duración	Hito
Planificación				
Realización del planificación de la administración	lun 08/06/15	mié 10/06/15	3 días	Documento de la planificación de la administración (véase sección 3.2.2 Planificación, apartado A)

Continuación (Tabla 14: Cronograma de la fase de planificación)				
Nombre	Fecha de inicio	Fecha fin	Duración	Hito
Definición de requisitos funcionales y no funcionales	jue 11/06/15	mié 17/06/15	5 días	Documento de especificación requisitos funcionales y no funcionales (véase sección 3.2.2. Planificación, apartado B)
Evaluación y priorización de requisitos funcionales y no funcionales	jue 18/06/15	vie 19/06/15	2 días	Documento de priorización de requisitos funcionales y no funcionales (véase sección 3.2.2. Planificación, apartado B)

Fuente: elaboración propia

Cronograma de diseño

Tabla 15: Cronograma de la fase de diseño

Nombre	Fecha de inicio	Fecha fin	Duración	Hito
Diseño				
Diseño de las interfaces	lun 22/06/15	mar 30/06/15	7 días	Documento de diseño de las interfaces del videojuego (véase sección 3.2.3. Diseño, apartado A)
Diseño de los paquetes	mié 01/07/15	vie 03/07/15	3 días	Documento de diseño de los paquetes del videojuego (véase sección 3.2.3. Diseño, apartado B)

Fuente: elaboración propia

Cronograma de elaboración

Tabla 16: *Cronograma de la fase de elaboración*

Nombre	Fecha de inicio	Fecha fin	Duración	Hito
Elaboración				
Iteración 1	lun 06/07/15	vie 17/07/15	10 días	Interfaz primer splash, segundo splash, menú principal y menú de niveles.
Iteración 2	lun 20/07/15	vie 31/07/15	10 días	Interfaz over relay de menú principal, pausa y nivel
Iteración 3	lun 03/08/15	vie 14/08/15	10 días	Interfaz de primer y segundo tutorial e interfaz de créditos
Iteración 4	lun 17/08/15	vie 28/08/15	10 días	Niveles 1, 2 y 3
Iteración 5	lun 31/08/15	vie 11/09/15	10 días	Niveles 4, 5, 6, 7 y 8
Iteración 6	lun 14/09/15	vie 25/09/15	10 días	Niveles 9, 10, 11 y 12
Iteración 7	lun 28/09/15	vie 09/10/15	10 días	Niveles 13, 14 y 15
Iteración 8	lun 12/10/15	vie 23/10/15	10 días	Niveles 16, 17 y 18
Iteración 9	lun 26/10/15	vie 06/11/15	10 días	Niveles 19 y 20

Fuente: elaboración propia

Cronograma beta

Tabla 17: *Cronograma de la fase beta*

Nombre	Fecha de inicio	Fecha fin	Duración	Hito
Beta				
Iteración 1	lun 09/11/15	mar 10/11/15	2 días	Plan de revisión del videojuego
Iteración 2	mié 11/11/15	mar 24/11/15	10 días	Casos de prueba de las interfaces
Iteración 3	mié 25/11/15	vie 27/11/15	3 días	Lista de errores corregidos

Fuente: elaboración propia

Cierre del proyecto

Tabla 18: Cronograma de la fase de cierre

Nombre	Fecha de inicio	Fecha fin	Duración	Hito
Cierre				
Evaluación del proyecto	lun 30/11/15	mié 02/12/15	3 días	Documento de lecciones aprendidas
Liberación del videojuego	jue 03/12/15	jue 03/12/15	1 día	Videojuego desarrollado y liberado
Despliegue del videojuego	vie 04/12/15	vie 04/12/15	1 día	Distribución del videojuego a usuarios finales

Fuente: elaboración propia

- **Definición del presupuesto**

Donde se detalla el presupuesto que se necesitó para la ejecución del proyecto, y la fuente de financiamiento para ello:

Presupuesto para el desarrollo

Tabla 19: Presupuesto detallado del proyecto

Rubro	Descripción	Monto (Nuevos Soles)
Equipos		S/. 2,750.00
	Laptop Lenovo	S/. 1,800.00
	Teléfono inteligente con sistema operativo Android	S/. 400.00
	Teléfono inteligente con sistema operativo iOS	S/. 550.00
Licencias de Software		S/. 2,662.00
	Corona SDK (Versión Básica)	S/. 257.00
	Wondershare video editor	S/. 98.00
	Microsoft Office 2013	S/. 326.00
	Xilisoft Video Converter	S/. 147.00
	CorelDRAW Graphics Suite X7	S/. 1,834.00
Licencias para distribución		S/. 405.00
	Licencia para distribución en Play Store (Android)	S/. 82.00
	Licencia para distribución en App Store (iOS)	S/. 323.00

Continuación (Tabla 19: Presupuesto detallado del proyecto)		
Rubro	Descripción	Monto (Nuevos Soles)
Desarrollo del videojuego		S/. 11,770.00
	Diseñador	S/. 770.00
	Programador	S/. 8,460.00
	Artista sonoro	S/. 810.00
	Artista gráfico	S/. 980.00
	Verificador beta	S/. 750.00
Asesor		S/. 2,500.00
	Asesor	S/. 2,500.00
Servicios		S/. 803.00
	Internet	S/. 350.00
	Impresión y anillado	S/. 253.00
	Movilidad	S/. 200.00
Otros		S/. 3,000.00
	Imprevistos	S/. 2,000.00
	Viáticos	S/. 1,000.00
	TOTAL	S/. 23,890.00

Fuente: elaboración propia

Financiamiento

Autofinanciado	S/. 23,890.00
Total	S/. 23,890.00

B. Especificación del videojuego

Se realizaron las tareas de definir las características del videojuego, es decir los requisitos funcionales y no funcionales, luego se evaluó cuanto tiempo necesitó cada requisito para ser desarrollado y posteriormente se priorizó estos requisitos para desarrollar primero las características más relevantes que permitieron maximizar el avance del proyecto, dichas tareas se muestran a continuación:

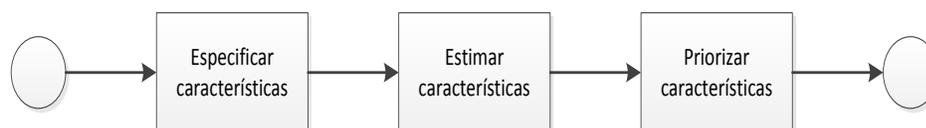


Figura 18. Flujo de tareas de la especificación del videojuego

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de estas tareas se muestra a continuación:

- **Especificación de los requisitos**

A continuación se listan los requisitos (características) referentes al desarrollo del videojuego, los cuales incluyen aspectos relacionados con la lógica del videojuego (gameplay), las interfaces y también el funcionamiento de éste. Por otra parte también se listan los requisitos más relevantes, vinculados con la parte no funcional del videojuego, tales como seguridad, calidad, rendimiento, etc.

Requisitos funcionales

- Área de interfaz

Requisito 1: el videojuego debe contar con una interfaz donde se muestre el nombre del equipo de desarrollo, a modo de presentación (primer splash).

Requisito 2: el videojuego debe contar con una interfaz donde se muestre el nombre del videojuego, a modo de presentación (segundo splash).

Requisito 3: una interfaz (menú principal) donde se muestren las opciones principales del videojuego, es decir opciones de jugar, salir, ver créditos, opciones de configuración, etc. Esta interfaz debe tener una música de fondo.

Requisito 4: una interfaz (menú de niveles) donde se muestren los niveles, además de una opción para regresar a la interfaz anterior. Esta interfaz debe tener una música de fondo.

Requisito 5: una interfaz para cada nivel (interfaz de nivel), donde se resolverá el puzzle, con las opciones de pausar y reiniciar del juego. Esta interfaz debe tener una música de fondo.

Requisito 6: una interfaz tipo pop-up, emergente (over relay de pausa), que será mostrada cuando el usuario presione el botón de pausa en la interfaz de nivel, la cual tendrá las opciones para continuar con el juego y regresar al menú de niveles.

Requisito 7: una interfaz tipo pop-up, emergente (over relay de nivel), que será mostrada cuando el usuario resuelva el nivel correspondiente, la cual tendrá las opciones para volver a jugar, pasar al siguiente nivel y regresar al menú de niveles.

Requisito 8: una interfaz tipo pop-up, emergente (over relay de menú), que será mostrada cuando el usuario presione el botón opciones del menú principal, esta interfaz tendrá las opciones de reiniciar el progreso del videojuego, cambiar idioma y ajustar volumen, además de una opción para volver a mostrar el menú principal.

Requisito 9: una interfaz (créditos) donde se muestre los datos del equipo desarrollo del videojuego, contando además con una opción para regresar a la interfaz anterior. Esta interfaz debe tener una música de fondo.

Requisito 10: una interfaz (primer tutorial) que explicará cómo se debe jugar los primeros 10 niveles del videojuego. Esta interfaz debe tener una música de fondo.

Requisito 11: una interfaz (segundo tutorial) que explicará cómo se debe jugar los 10 niveles restantes del videojuego. Esta interfaz debe tener una música de fondo.

- Área de gameplay

Requisito 12: el botón jugar del menú principal permitirá ingresar al menú de niveles, el botón salir permitirá salir del videojuego, el botón opciones mostrará una interfaz emergente con las opciones de configuración del videojuego.

Requisito 13: al mostrarse la interfaz de over relay del menú principal, se tendrá un botón para reiniciar, esta reiniciará el videojuego y bloqueará todos los niveles, excepto el nivel 1, tal y como se tenía el videojuego luego del acceso por primera vez.

Requisito 14: al mostrarse la interfaz de over relay del menú principal, se tendrá dos botones para cambiar el idioma, de español a inglés y viceversa, dicho cambio deberá efectuarse sin tener que reiniciar el videojuego.

Requisito 15: al mostrarse la interfaz de over relay del menú principal, se tendrá dos botones para cambiar el volumen, dicho cambio deberá ser capaz de subir y bajar el volumen.

Requisito 16: al ingresar al videojuego por primera vez el menú de niveles solo habilitará el acceso al nivel 1, para acceder al siguiente nivel primero se

debe desbloquear (resolver) el nivel anterior, una vez desbloqueado un nivel, se puede acceder a este en cualquier momento.

Requisito 17: en la interfaz de nivel se mostrará el videojuego, el cual presentará:

- Un laberinto, por donde se desplazará al *personaje*.
- Para los 10 primeros niveles un *personaje* (imagen a desplazar), que será ubicado en una posición inicial y al cual se debe llevar a través del laberinto a una posición final (meta).
- Para los 10 niveles restantes dos *personajes* (imágenes a desplazar), que serán ubicados en una posición inicial y a los cuales se debe llevar a través del laberinto a una posición final hasta su respectiva posición final (meta).
- Controles de dirección, que permitirán mover al *personaje* o *personajes* a la derecha, izquierda, arriba y abajo a través del laberinto.
- Para los 10 primeros niveles un *control especial* para habilitar y deshabilitar el desplazamiento del *personaje* en el laberinto.
- Para los 10 niveles restantes dos *controles especiales* para habilitar y deshabilitar el desplazamiento de los *personajes* en el laberinto, con esto solo se podrá desplazar un *personaje* si su respectivo *control especial* está habilitado.
- Un contador de tiempo, que indicará cuanto tiempo se lleva jugando.
- Un contador de movimientos que indicará cuantos movimientos se realizan.
- Flechas que indiquen el último control de dirección presionado.

Requisito 18: la interfaz de nivel incluirá una opción para pausar el juego, durante dicha pausa los demás controles serán bloqueados, también se contará con la opción para reiniciar el juego, con lo cual se reiniciará la posición inicial del *personaje(es)*, el contador de tiempo y el contador de movimientos.

Requisito 19: cuando se pause el juego, el contador de tiempo se detendrá, se mostrará la interfaz de over relay de pausa, la cual contendrá dos opciones, una para continuar con el juego y otra para regresar al menú de niveles, al seleccionar la segunda opción, no se guardará el progreso correspondiente a dicho nivel desde donde se pausó el juego y se regresará al menú de niveles, al presionar la opción continuar el juego será reanudado así como el contador de tiempo.

Requisito 20: en la interfaz de nivel (si se está en un nivel donde hay dos *personajes*), solo se podrá desplazar un *personaje* por vez, para ello previamente se deberá activar la imagen (*control especial*) del *personaje*, en caso contrario el *personaje* no podrá ser movido, además solo se empezará a contar el tiempo cuando un *personaje* este activado mediante su *control especial*.

Requisito 21: mientras un *personaje* no sea activado mediante su *control especial*, no se iniciará el contador de tiempo, además se mostrará un mensaje en pantalla indicando que se presione el *control especial* del *personaje* para poder desplazarlo por el laberinto, dicho mensaje desaparecerá una vez que se presione el *control especial* de un *personaje*.

Requisito 22: en la interfaz de nivel al presionar un control de dirección (derecha, izquierda, arriba, abajo) el *personaje* será desplazado en dicha dirección, mientras ocurre esto, no se podrá cambiar la dirección del desplazamiento hasta que el *personaje* este detenido por completo.

Requisito 23: en la interfaz de nivel, al activar la opción pausa, se bloquearán los controles de dirección, *el control especial* que permite activar al *personaje* y la opción reiniciar, pero no se pausará la música de fondo, cuando se a presione la opción continuar del over relay de pausa el juego continuará con su curso normal.

Requisito 24: mientras el *personaje* este en movimiento no se podrá poner pausa, tampoco cambiar de dirección, pero si se podrá reiniciar el nivel.

Requisito 25: la puntuación obtenida por el usuario dependerá de cuantas veces usa los controles para mover al *personaje* y llegar al punto final (meta) y el tiempo empleado, donde lo más importante para obtener una puntuación

alta es la cantidad de movimientos empleados, ya que el tiempo empleado solo funcionará como un bono de descuento en la cantidad de movimientos total.

Requisito 26: se resolverá el nivel cuando el *personaje* sea llevado hasta la meta o cuando los dos *personajes* sean llevados a la meta, para este último caso una vez que uno de los dos *personajes* haya sido llevado hasta su meta no podrá ser siendo movido, con lo cual lo único que restará para resolver el nivel será llevar el otro *personaje* hasta su meta.

Requisito 27: para los 10 últimos niveles en caso de resolver inadecuadamente el primer *personaje*, se mostrará en pantalla un mensaje dando una pista para resolver correctamente dicho nivel.

Requisito 28: una vez que se haya solucionado en nivel, se mostrará la interfaz de over relay de nivel, en la cual se mostrarán estrellas, las cuales representan el puntaje obtenido por el usuario, mientras menor sea el uso de los controles de dirección, mayor será la cantidad de estrellas obtenidas por resolver el nivel.

Requisito 29: en la interfaz de over relay de nivel la opción ir al menú de niveles permitirá ir automáticamente a esta interfaz, la opción volver a jugar permitirá reiniciar el mismo nivel para volver a jugarlo y la opción siguiente nivel cargará el siguiente nivel correspondiente.

Requisito 30: la dificultad de la resolución de los niveles irá aumentando conforme se vayan resolviendo los niveles inferiores.

Requisito 31: en la interfaz de menú de niveles, cada vez que se ingrese a esta, se mostrarán los niveles resueltos con sus respectivos puntajes representados por estrellas, así como aquellos niveles aun no resueltos.

Requisito 32: cada vez que el usuario salga del videojuego y vuelva a ingresar, se mostrará el progreso que dejó la última vez que uso el videojuego, es decir los datos de niveles y sus respectivos puntajes serán guardados en un archivo externo, así como el idioma que se tenía antes de salir del videojuego.

Requisito 33: al ingresar al nivel 1 desde la interfaz de niveles, primero se mostrará la interfaz de primer tutorial donde se indicará como jugar el

videojuego, luego se pasará a la interfaz del nivel 1 cuando el usuario presione el botón OK, de manera similar ocurrirá con el nivel 11 mostrándose el segundo tutorial. Adicionalmente solo en el nivel 1 se mostrará además del mensaje que indica presionar el *control especial* del *personaje* para moverlo, un mensaje adicional indicando que se deben presionar los controles de dirección para desplazar al *personaje*.

Requisitos no funcionales

Requisito 34: el videojuego funcionará en el sistema operativo Android, desde la versión 2.3.3 Gingerbread.

Requisito 35: el videojuego funcionará en el sistema operativo iOS, desde la versión 6.0.

Requisito 36: el videojuego usará el mismo código fuente para generar los instaladores para los sistemas operativos Android e iOS.

Requisito 37: el videojuego mantendrá la resolución óptima en ambos sistemas operativos.

Requisito 38: el videojuego presentará compatibilidad en la reproducción de audio para ambos sistemas operativos.

Requisito 39: el videojuego funcionará fluidamente (sin lag) en ambos sistemas operativos.

Requisito 40: la navegación por las interfaces del videojuego será fácil, es decir el videojuego será amigable e intuitivo.

Requisito 41: el videojuego deberá consumir la menor cantidad de memoria RAM al momento de ser usado, igualmente deberá utilizar la menor cantidad de recursos del procesador, así como también la batería y espacio ocupado luego de la instalación.

Requisito 42: el videojuego no presentará errores al momento de liberarlo.

- Estimación y priorización de requisitos

Estimación y priorización de los requisitos						
Prioridad (1-5)	Realizado	Id	Requisito	Responsable	Duracion (días)	Notas
3	100%	1	Requisito 1	Artista gráfico		2 Nota 1
4	100%	2	Requisito 2	Artista gráfico		2 Nota 1
5	100%	3	Requisito 3	Artisa gráfico, Artista Sonoro		2 Nota 1
5	100%	4	Requisito 4	Artisa gráfico, Artista Sonoro		2 Nota 1
5	100%	5	Requisito 5	Artista gráfico, Artista Sonoro		2 Nota 1
4	100%	6	Requisito 6	Artista gráfico, Programador		2 Nota 1
5	100%	7	Requisito 7	Artista gráfico, Programador		2 Nota 1
4	100%	8	Requisito 8	Artista gráfico, Programador		2 Nota 1
4	100%	9	Requisito 9	Artisa gráfico, Artista Sonoro		2 Nota 1
5	100%	10	Requisito 10	Artisa gráfico, Artista Sonoro		2 Nota 1
5	100%	11	Requisito 11	Artisa gráfico, Artista Sonoro		2 Nota 1
5	100%	12	Requisito 12	Programador		2 Nota 1
4	100%	13	Requisito 13	Diseñador del juego, Artista gráfico, Programador		2 Nota 1
5	100%	14	Requisito 14	Artista gráfico, Programador		2 Nota 1
4	100%	15	Requisito 15	Artista gráfico, Programador		2 Nota 1
4	100%	16	Requisito 16	Diseñador del juego, Artista gráfico, Programador		2 Nota 1
5	100%	17	Requisito 17	Diseñador del juego, Artista gráfico, Programador		10 Nota 1
4	100%	18	Requisito 18	Diseñador del juego, Programador		2 Nota 1
4	100%	19	Requisito 19	Diseñador del juego, Programador		3 Nota 1
5	100%	20	Requisito 20	Diseñador del juego, Programador		5 Nota 1
5	100%	21	Requisito 21	Diseñador del juego, Artista gráfico, Programador		3 Nota 1
4	100%	22	Requisito 22	Diseñador del juego, Programador		3 Nota 1
4	100%	23	Requisito 23	Diseñador del juego, Programador		3 Nota 1
5	100%	24	Requisito 24	Diseñador del juego, Programador		3 Nota 1
5	100%	25	Requisito 25	Diseñador del juego, Programador		3 Nota 1

Figura 19: Estimación y priorización de los requisitos parte 1

Fuente: elaboración propia

Estimación y priorización de requisitos (continuación)

Estimación y priorización de los requisitos (continuación)							
Prioridad (1-5)	Realizado	Id	Requisito	Responsable	Duración (días)	Notas	
5	100%	26	Requisito 26	Diseñador del juego, Programador		2 Nota 1	
4	100%	27	Requisito 27	Diseñador del juego, Artista gráfico, Programador		3 Nota 1	
4	100%	28	Requisito 28	Diseñador del juego, Artista gráfico, Programador		2 Nota 1	
4	100%	29	Requisito 29	Diseñador del juego, Programador		3 Nota 1	
5	100%	30	Requisito 30	Diseñador del juego, Programador		5 Nota 1	
4	100%	31	Requisito 31	Diseñador del juego, Artista gráfico, Programador		2 Nota 1	
5	100%	32	Requisito 32	Programador		2 Nota 1	
4	100%	33	Requisito 33	Diseñador del juego, Artista gráfico, Programador		2 Nota 1	
5	100%	34	Requisito 34	Programador		3 Nota 1	
5	100%	35	Requisito 35	Programador		12 Nota 1	
5	100%	36	Requisito 36	Programador		3 Nota 1	
5	100%	37	Requisito 37	Artista gráfico, Programador		3 Nota 1	
5	100%	38	Requisito 38	Artista Sonoro, Programador		3 Nota 1	
5	100%	39	Requisito 39	Programador		4 Nota 1	
5	100%	40	Requisito 40	Diseñador del juego, Artista gráfico, Artista sonoro		4 Nota 1	
5	100%	41	Requisito 41	Diseñador del juego, Artista gráfico, Artista sonoro		10 Nota 1	
5	100%	42	Requisito 42	Verificador Beta		15 Nota 1	
	Nota 1:		Este requisito requiere la supervisión del Producto interino				

Figura 20: Estimación y priorización de requisitos parte 2

Fuente: elaboración propia

3.2.3. Diseño

Se implementó una fase adicional, la fase de diseño, donde se especificó la manera en cómo se representa gráficamente cada interfaz del videojuego.

Al no ser una fase propia de la metodología SUM, esta fase de diseño siguió una similitud a la parte de la metodología GUP para diseñar las interfaces donde se aplicó el flujo de trabajo Especificación o detalles del diseño, así como la metodología RUP para explicar el flujo de eventos de estas interfaces, haciendo referencia a la Especificación de casos de uso.

También se diseñaron los paquetes que agrupan los ficheros que son parte del código fuente del videojuego con la finalidad de comprender mejor la manera en la que estos ficheros se relacionan para hacer funcionar al videojuego, el diseño de estos paquetes se hizo tomando referencia al artefacto Diseño de paquetes de la metodología RUP, el flujo de las tareas realizadas se muestran a continuación:



Figura 21. Flujo de tareas del diseño de las interfaces y paquetes del videojuego

Fuente: elaboración propia

El desarrollo de estas tareas se muestra a continuación:

A. Diseño de las interfaces

- **Representación del curso normal y alterno del videojuego**

- a. **Interfaz de primer splash:** la cual muestra el nombre del equipo de desarrollo.

Tabla 20: Flujo de eventos de la interfaz primer splash

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario inicia el videojuego.	2. El videojuego carga los archivos multimedia.
3. El usuario espera unos segundos.	4. El videojuego muestra la primera interfaz de bienvenida con el nombre del equipo de desarrollo.

Fuente: elaboración propia

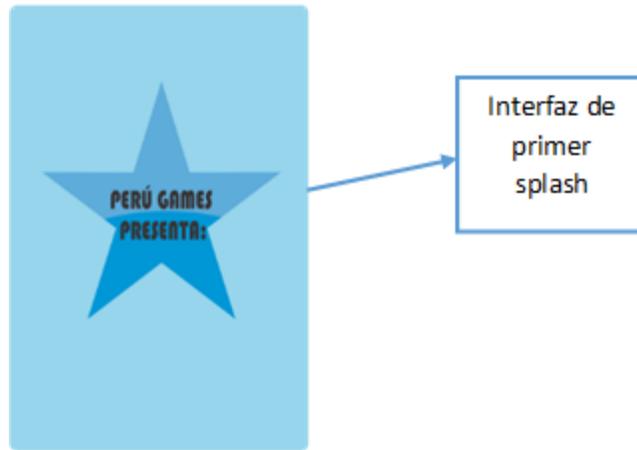


Figura 22: Diseño de la interfaz primer splash

Fuente: elaboración propia

- b. **Interfaz de segundo splash:** la cual muestra el nombre del videojuego.

Tabla 21: Flujo de eventos de la interfaz de segundo splash

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario espera unos segundos.	2. El videojuego muestra la segunda interfaz de bienvenida con el nombre del videojuego.

Fuente: elaboración propia

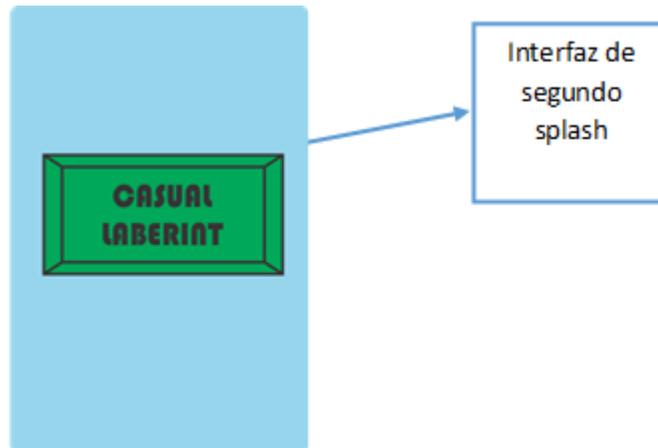


Figura 23: Diseño de la interfaz segundo splash

Fuente: elaboración propia

- c. **Interfaz de menú principal:** representada por la segunda figura, con las opciones de JUGAR, OPCIONES, VER CRÉDITOS Y SALIR DEL VIDEOJUEGO.

Tabla 22: Flujo de eventos de la interfaz menú principal

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario espera unos segundos luego que aparece la segunda interfaz de bienvenida.	2. El videojuego muestra la interfaz de menú principal.
FLUJO ALTERNO	
1. Si el usuario presiona el botón OPCIONES, se mostrará la interfaz de over relay de configuración del menú principal.	
2. Si el usuario presiona el botón CRÉDITOS, se mostrará la interfaz con los datos del equipo desarrollador del videojuego.	
3. Si el usuario presiona el botón SALIR, se cerrará el videojuego.	

Fuente: elaboración propia

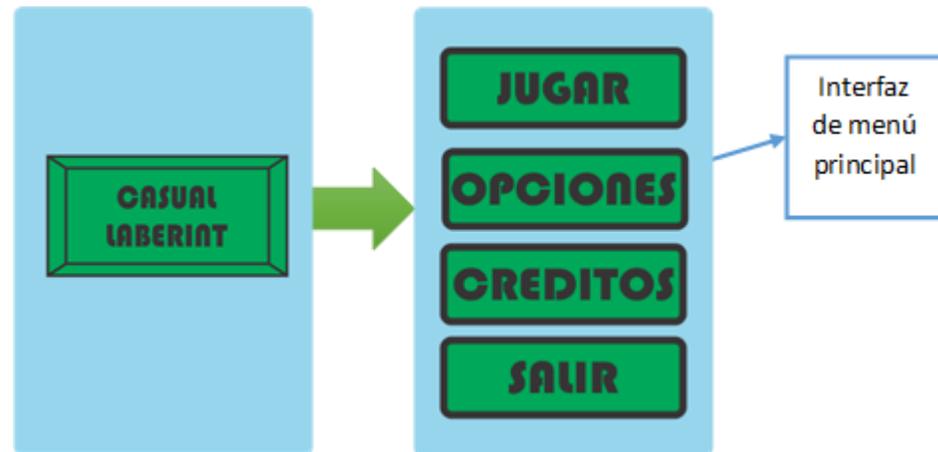


Figura 24: Diseño de la interfaz menú principal

Fuente: elaboración propia

- d. **Interfaz menú de niveles:** como se muestra en la segunda imagen, donde están los 20 niveles que podrán ser resueltos.

Tabla 23: Flujo de eventos de la interfaz menú de niveles

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario presionó el botón JUGAR.	2. El videojuego muestra la interfaz menú de niveles.
3. El usuario presiona el botón que representa el nivel 1.	4. El videojuego muestra la siguiente interfaz (interfaz de primer tutorial).
FLUJO ALTERNO	
4. Si el usuario presiona algún botón que no posee un número, no podrá ingresar a dicho nivel, para ello primero deberá jugar y resolver el nivel anterior.	
5. Si el usuario presiona la imagen que representa el botón ATRAZ, se regresará a la interfaz de menú principal.	

Fuente: elaboración propia

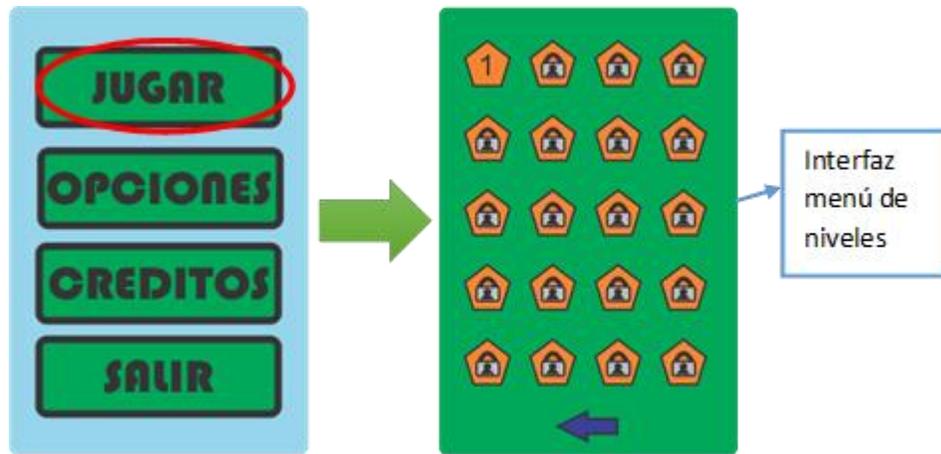


Figura 25: Diseño de la interfaz menú de niveles

Fuente: elaboración propia

- e. **Interfaz de primer tutorial:** la cual brinda información de cómo jugar los 10 primeros niveles del videojuego.

Tabla 24: Flujo de eventos de la interfaz primer tutorial

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario presiona el botón que representa el nivel 1.	2. El videojuego muestra la interfaz de tutorial.
3. El usuario presiona el botón OK	4. El videojuego muestra la siguiente interfaz.

Fuente: elaboración propia

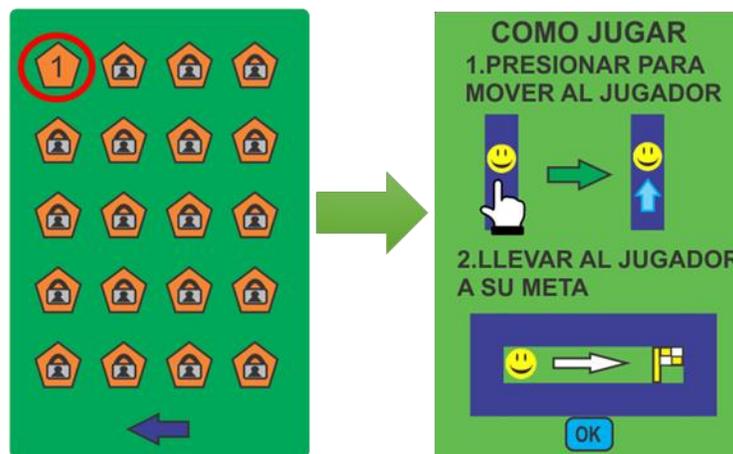


Figura 26: Diseño de la interfaz primer tutorial

Fuente: elaboración propia

- f. **Interfaz de nivel (10 primeros):** como se muestra en la segunda imagen, donde se tendrá que resolver el puzzle, es decir llevar al personaje desde la posición inicial (punto de partida) hasta la posición final (llegada en la bandera) usando los controles de dirección (flechas que están en la parte inferior de la interfaz de nivel).

Tabla 25: Flujo de eventos de la interfaz 10 primeros niveles

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario presionó el botón OK de la interfaz de primer tutorial.	2. El videojuego muestra la interfaz de nivel.
3. El usuario presiona la imagen que representa el control especial que habilita el movimiento del personaje.	4. Desaparece el mensaje que indica que se presione el control especial, se muestra una flecha debajo de la imagen que activa el personaje, comienza a correr el reloj.
5. El usuario presiona el control de dirección que representa la dirección derecha.	6. La imagen que representa al personaje se desplaza a la derecha y aparecerán flechas que indicarán la dirección del último control de dirección que fue presionado, se aumenta en uno la cantidad de movimientos.
7. El usuario presiona el control de dirección que representa la dirección izquierda.	8. La imagen que representa al personaje se desplaza a la izquierda y aparecerán flechas que indicarán la dirección del último control de dirección que fue presionado, se aumenta en uno la cantidad de movimientos.
9. El usuario presiona el control de dirección que representa la dirección arriba.	10. La imagen que representa al personaje se desplaza hacia arriba y aparecerán flechas que indicarán la dirección del último control de dirección que fue presionado, se aumenta en uno la cantidad de movimientos.
11. El usuario presiona el control de dirección que representa la dirección abajo.	12. La imagen que representa al personaje se desplaza hacia abajo y aparecerán flechas que indicarán la dirección del último control de dirección que fue presionado, se aumenta en uno la cantidad de movimientos.
13. El personaje llega a la imagen que representa la meta.	14. Se muestra la interfaz que indica que se resolvió el puzzle.
FLUJO ALTERNO	
6. Si el usuario no presiona la imagen que representa el control especial que activa al personaje y en lugar de ello presiona cualquiera de los controles de dirección, entonces el personaje no se moverá, el tiempo no comenzará a correr y no desaparecerá el mensaje en pantalla indicando que se debe presionar el control especial para activar al personaje y poder desplazarlo.	
7. Si el usuario presiona la imagen que representa al botón pausa, se mostrará la interfaz de over relay de pausa.	

FLUJO ALTERNO (Continuación Tabla 25: <i>Flujo de eventos de la interfaz 10 primeros niveles</i>)
8. Si el usuario presiona la imagen que representa al botón reiniciar, la imagen que representa al personaje será reubicada a la posición en la que estaba al inicio del juego y las flechas que representan la última flecha que fue presionada desaparecerán, también se reiniciará el contador del tiempo y los movimientos.

Fuente: elaboración propia

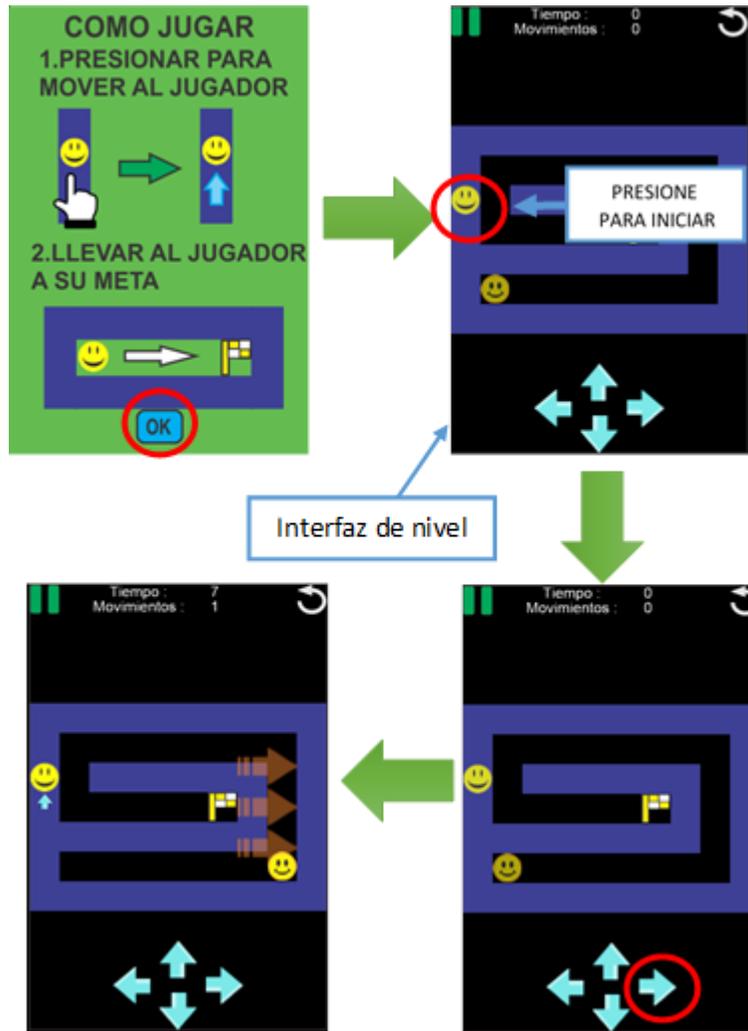


Figura 27: Diseño de la interfaz 10 primeros niveles

Fuente: elaboración propia

- g. Interfaz de segundo tutorial:** la cual brinda información de cómo jugar los 10 niveles restantes del videojuego.

Tabla 26: Flujo de eventos de la interfaz de segundo tutorial

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario presiona el botón que representa el nivel 11.	2. El videojuego muestra la interfaz de segundo tutorial.
3. El usuario presiona el botón OK.	4. El videojuego muestra la siguiente interfaz.

Fuente: elaboración propia

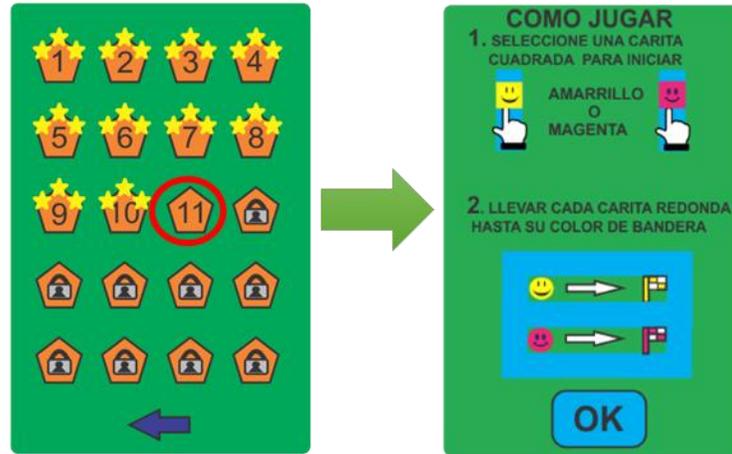


Figura 28: Diseño de la interfaz segundo tutorial

Fuente: elaboración propia

- h. **Interfaz de nivel (10 últimos):** como se muestra en la segunda figura, donde se tendrá que resolver el puzzle, es decir llevar a los personajes desde la posición inicial (punto de partida) hasta la posición final (llegada en la bandera) usando los controles de dirección (flechas que están en la parte inferior de la interfaz de nivel).

Tabla 27: Flujo de eventos de la interfaz 10 últimos niveles

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario presionó el botón OK de la interfaz de segundo tutorial.	2. El videojuego muestra la interfaz de nivel.
3. El usuario presiona una de las imágenes que representa el control especial que habilita el movimiento del personaje.	4. Desaparece el mensaje que indica que se presione el control especial, se muestra una flecha debajo de la imagen que activa el personaje, comienza a correr el reloj.
5. El usuario presiona el control de dirección que representa la dirección derecha.	6. La imagen que representa al personaje se desplaza a la derecha y aparecerán flechas que indicarán la dirección del último control de dirección que fue presionado, se aumenta en uno la cantidad de movimientos.

CURSO NORMAL DE LOS EVENTOS (Continuación Tabla 27: <i>Flujo de eventos de la interfaz 10 últimos niveles</i>)	
7. El usuario presiona el control de dirección que representa la dirección izquierda.	8. La imagen que representa al personaje se desplaza a la izquierda y aparecerán flechas que indicarán la dirección del último control de dirección que fue presionado, se aumenta en uno la cantidad de movimientos.
9. El usuario presiona el control de dirección que representa la dirección arriba.	10. La imagen que representa al personaje se desplaza hacia arriba y aparecerán flechas que indicarán la dirección del último control de dirección que fue presionado, se aumenta en uno la cantidad de movimientos.
11. El usuario presiona el control de dirección que representa la dirección abajo.	12. La imagen que representa al personaje se desplaza hacia abajo y aparecerán flechas que indicarán la dirección del último control de dirección que fue presionado, se aumenta en uno la cantidad de movimientos.
13. El primer personaje llega a la imagen que representa su color de meta.	14. Ahora este personaje ya no podrá ser movido.
15. El segundo personaje llega a la imagen que representa su color de meta.	16. Se muestra la interfaz que indica que se resolvió el puzzle.
FLUJO ALTERNO	
Los mismos de los 10 primeros niveles	

Fuente: elaboración propia

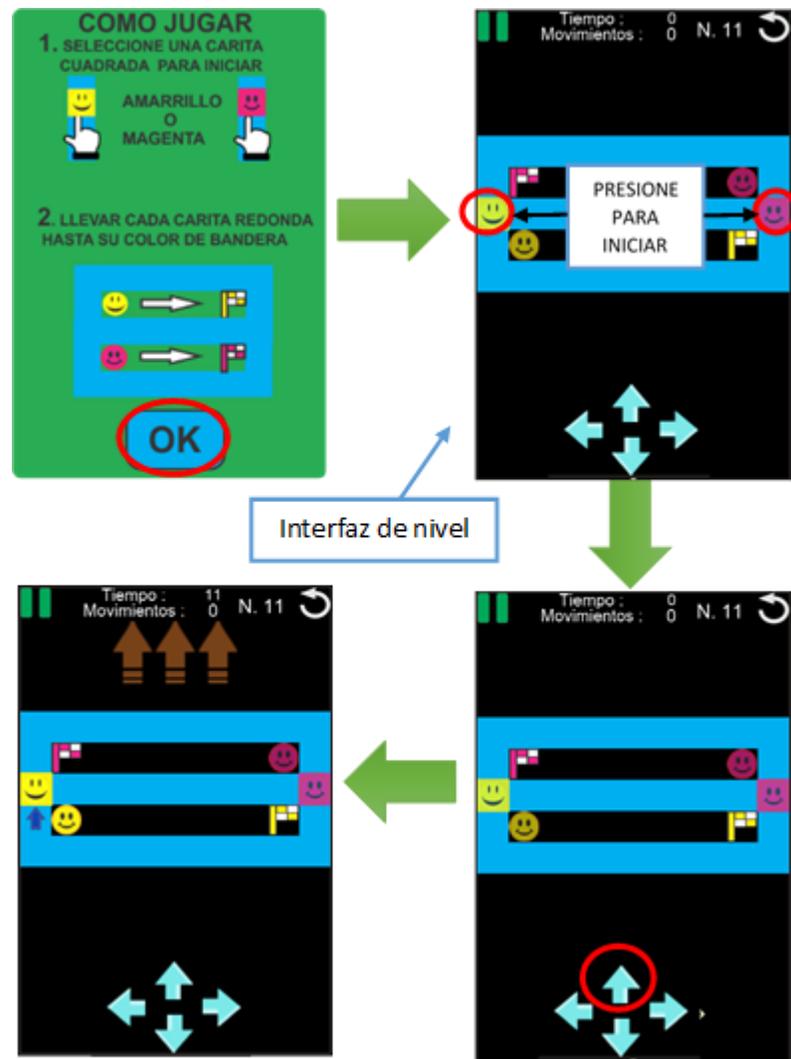


Figura 29: Diseño de la interfaz 10 últimos niveles

Fuente: elaboración propia

i. Interfaz de over relay de nivel

Tabla 28: Flujo de eventos de la interfaz over relay de nivel

CURSO NORMAL DE LOS EVENTOS	
ACCIÓN DEL USUARIO	ACCIÓN DEL VIDEOJUEGO
1. El usuario resolvió el puzzle.	2. El videojuego muestra la interfaz over relay de nivel, donde se incluyen las estrellas que representan el puntaje, donde además ya se desbloquea el siguiente nivel y ya se guarda el progreso (puntaje y nivel desbloqueado), además se muestran las estadísticas del nivel resuelto y como se puede obtener un mejor puntaje.

CURSO NORMAL DE LOS EVENTOS (Continuación Tabla 28: <i>Flujo de eventos de la interfaz over relay de nivel</i>)	
3. El usuario presiona el botón que representa el siguiente nivel.	4. El videojuego muestra la interfaz del siguiente nivel.
FLUJO ALTERNO	
9. Si el usuario presiona la imagen que representa al botón reiniciar, se volverá a cargar dicho nivel.	
10. Si el usuario presiona la imagen que representa el botón menú, se mostrará la interfaz menú de niveles y se guardará el progreso.	

Fuente: elaboración propia



Figura 30: Diseño de la interfaz over relay de nivel

Fuente: elaboración propia

- **Representación específica de flujos alternos**

1. Si el usuario presiona el botón OPCIONES, se mostrará la interfaz de over relay del menú principal:

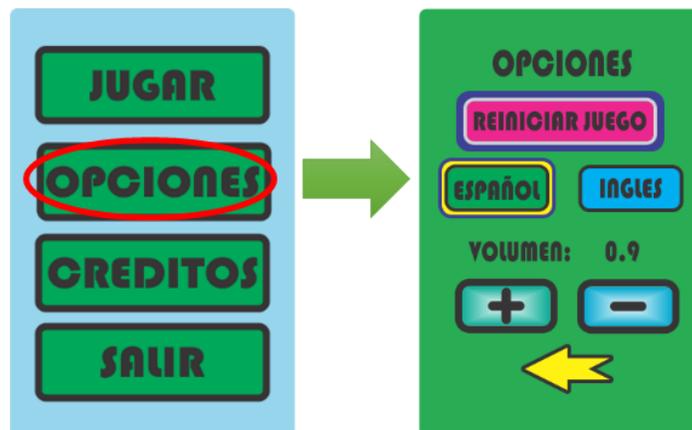


Figura 31: Flujo alterno OPCIONES

Fuente: elaboración propia

- Si el usuario presiona el botón Reiniciar, se borrará todo el progreso:

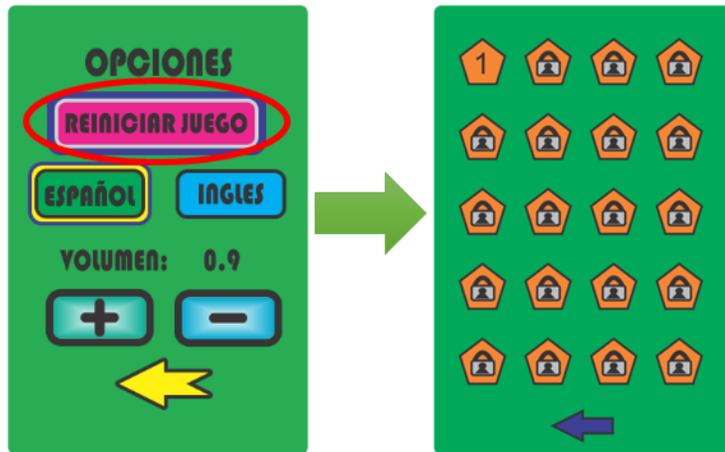


Figura 32: Flujo alternativo REINICIAR

Fuente: elaboración propia

- Si el usuario presiona el botón Español, no pasará nada porque el idioma ya está en Español, si el usuario presiona el botón Inglés entonces el idioma del juego cambiará a dicho idioma, el cambio de idioma se realizará al salir de la interfaz over relay de menú principal:

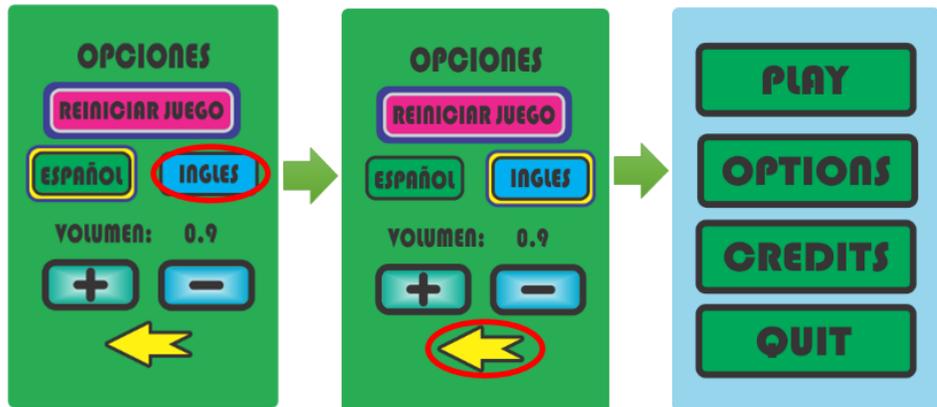


Figura 33: Flujo alternativo cambiar idioma a Inglés

Fuente: elaboración propia

- Si el usuario presiona el botón Inglés, no pasará nada porque el idioma ya está en Inglés, si el usuario presiona el botón Spanish entonces el idioma del juego cambiará a dicho idioma, el cambio de idioma se realizará al salir de la interfaz over relay de menú principal:

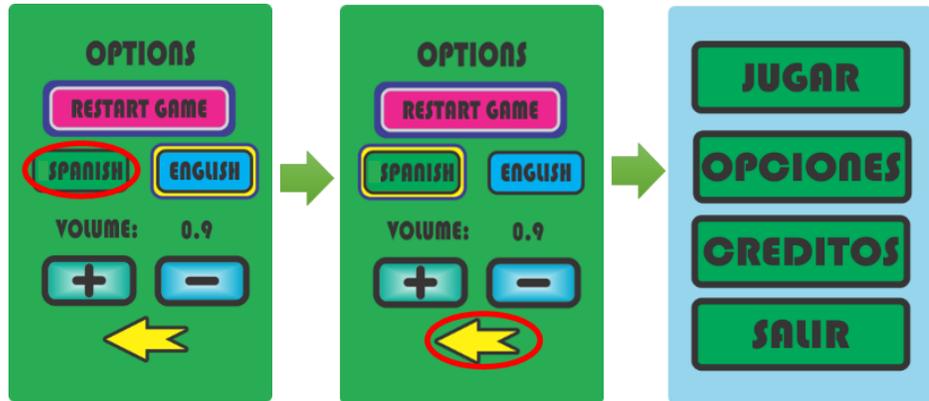


Figura 34: Flujo alterno cambiar idioma a Español

Fuente: elaboración propia

- Si el usuario presiona el botón “+” subirá el volumen, y si presiona el botón “-”, bajará el volumen:

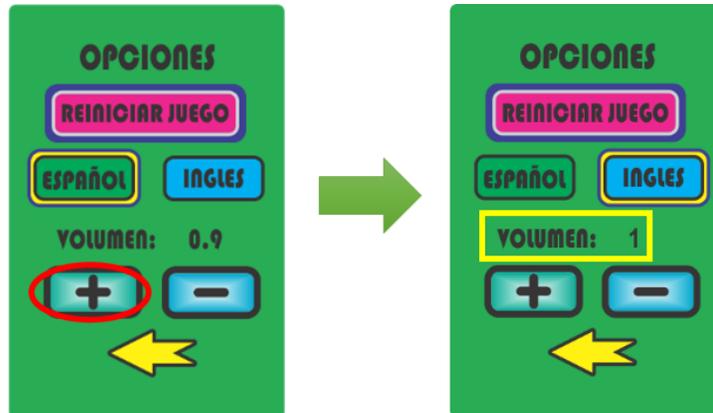


Figura 35: Flujo alterno subir volumen

Fuente: elaboración propia

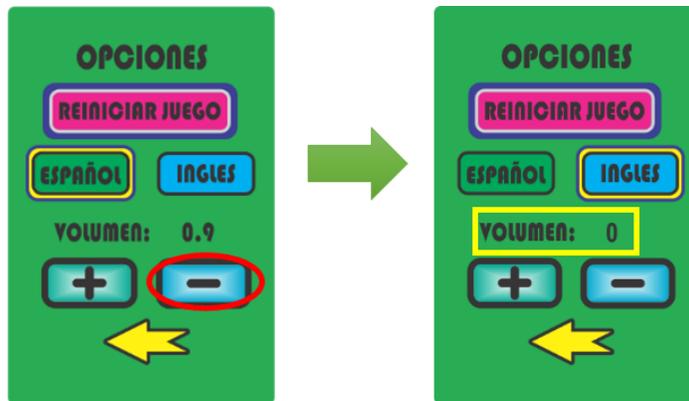


Figura 36: Flujo alterno bajar volumen

Fuente: elaboración propia

2. Si el usuario presiona el botón CRÉDITOS, se mostrarán los datos del equipo creador del videojuego:



Figura 37: Flujo alterno CRÉDITOS

Fuente: elaboración propia

3. Si el usuario presiona el botón SALIR, se cerrará el videojuego:



Figura 38: Flujo alternativo SALIR

Fuente: elaboración propia

4. Si el usuario presiona algún botón que no posee un número, no podrá ingresar a dicho nivel, para ello primero deberá jugar y resolver el nivel anterior:

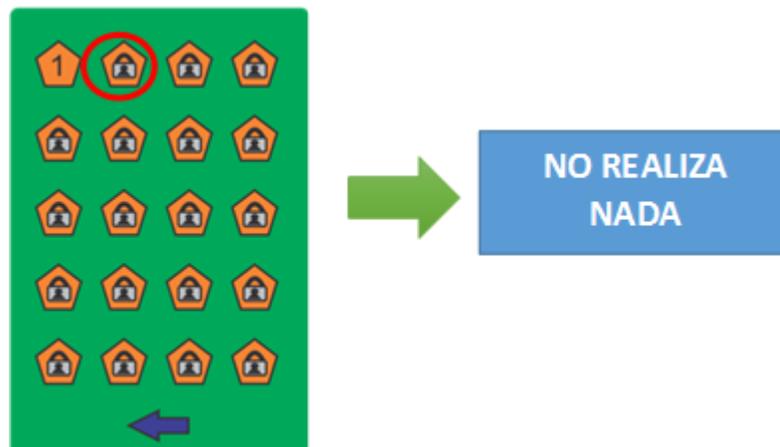


Figura 39: Flujo alternativo intentar acceder a nivel bloqueado

Fuente: elaboración propia

5. Si el usuario presiona la imagen que representa el botón ATRAZ, se mostrará la interfaz de menú principal:

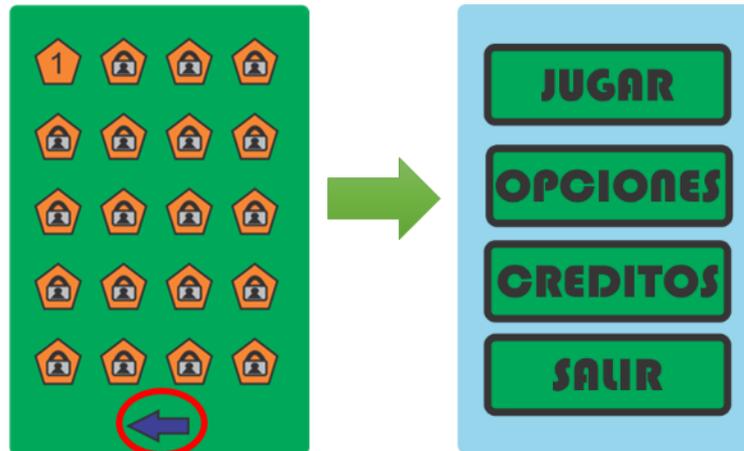


Figura 40: Flujo alterno salir del menú de niveles

Fuente: elaboración propia

6. Si el usuario no presiona la imagen que representa el activador del personaje y en lugar de ello presiona cualquiera de las flechas de dirección, entonces el personaje no se moverá, no comenzará a correr el tiempo y no desaparecerá el mensaje en pantalla indicando que se debe presionar el control especial para activar al personaje y poder desplazarlo:

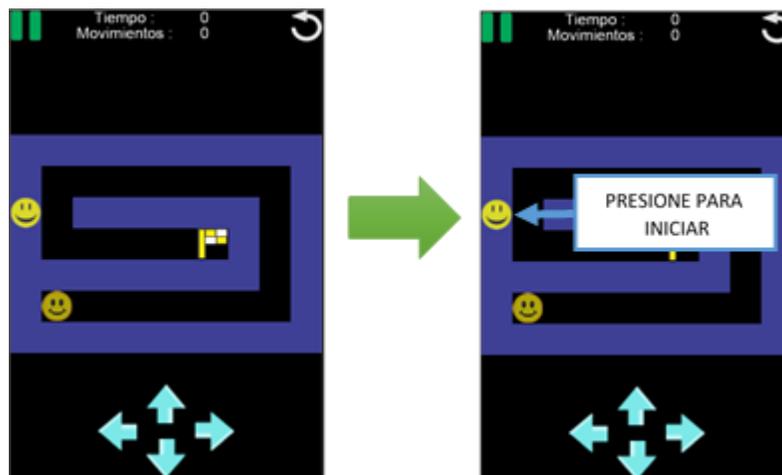


Figura 41: Flujo alterno no activar movimiento del personaje

Fuente: elaboración propia

7. Si el usuario presiona la imagen que representa al botón pausa, se mostrará la interfaz de over relay de pausa:

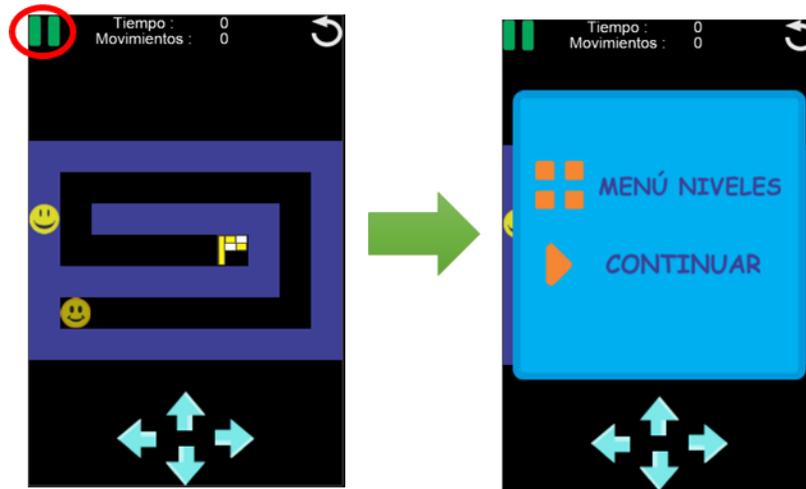


Figura 42: Flujo alternativo mostrar over relay de pausa

Fuente: elaboración propia

- Si el usuario presiona el botón menú niveles, se regresará a la interfaz de niveles y no se guardará ningún progreso.

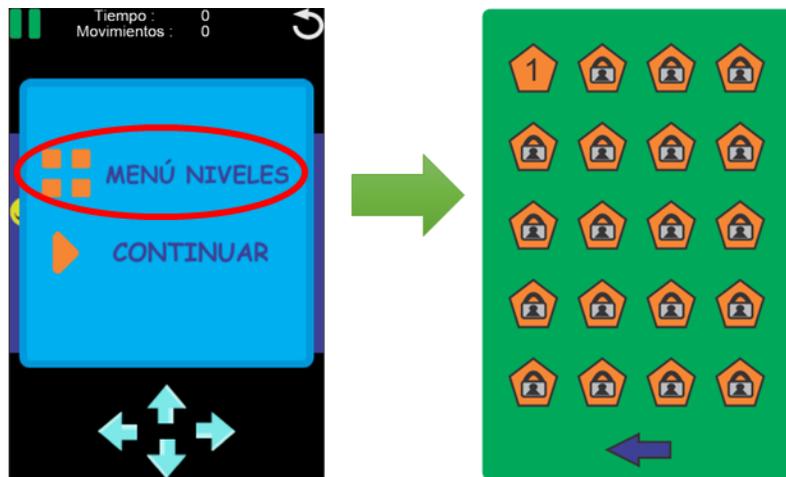


Figura 43: Flujo alternativo regresar al menú de niveles desde el over de pausa

Fuente: elaboración propia

- Si el usuario presiona el botón continuar, se reanuda el juego.

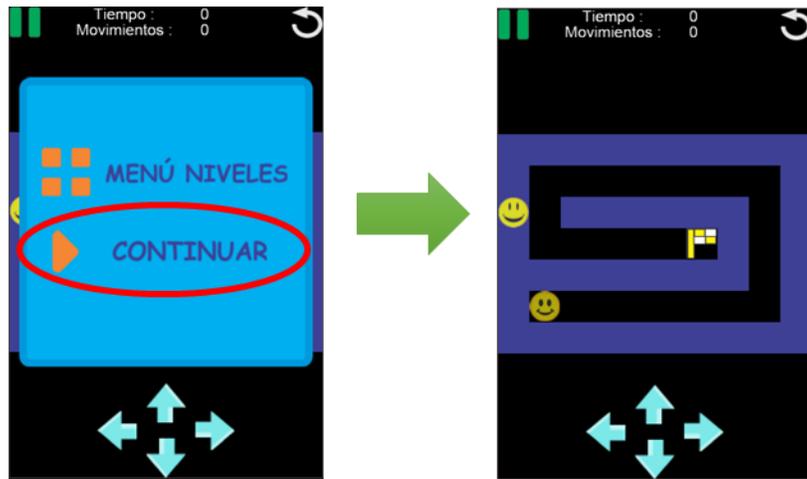


Figura 44: Flujo alternativo reanudar juego

Fuente: elaboración propia

8. Si el usuario presiona la imagen que representa al botón reiniciar, la imagen que representa al personaje será reubicada a la posición en la que estaba al inicio del juego y las flechas que representan la última flecha que fue presionada desaparecerán, también se reiniciará el contador del tiempo y los movimientos:

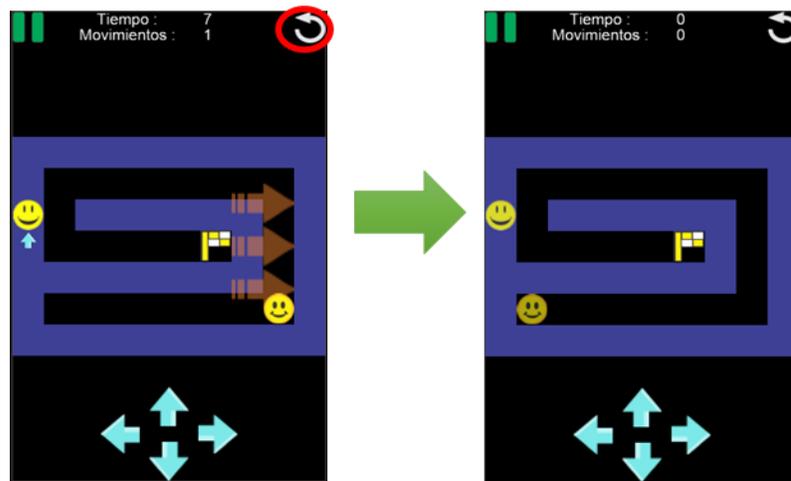


Figura 45: Flujo alternativo reiniciar juego desde interfaz de nivel

Fuente: elaboración propia

9. Si el usuario presiona la imagen que representa al botón reiniciar, se volverá a cargar dicho nivel:



Figura 46: Flujo alterno reiniciar juego desde interfaz de over de nivel

Fuente: elaboración propia

10. Si el usuario presiona la imagen que representa el botón MENÚ se mostrará la interfaz menú de niveles y se guardará el progreso:



Figura 47: Flujo alterno regresar al menú de niveles desde el over de nivel

Fuente: elaboración propia

B. Diseño de los paquetes

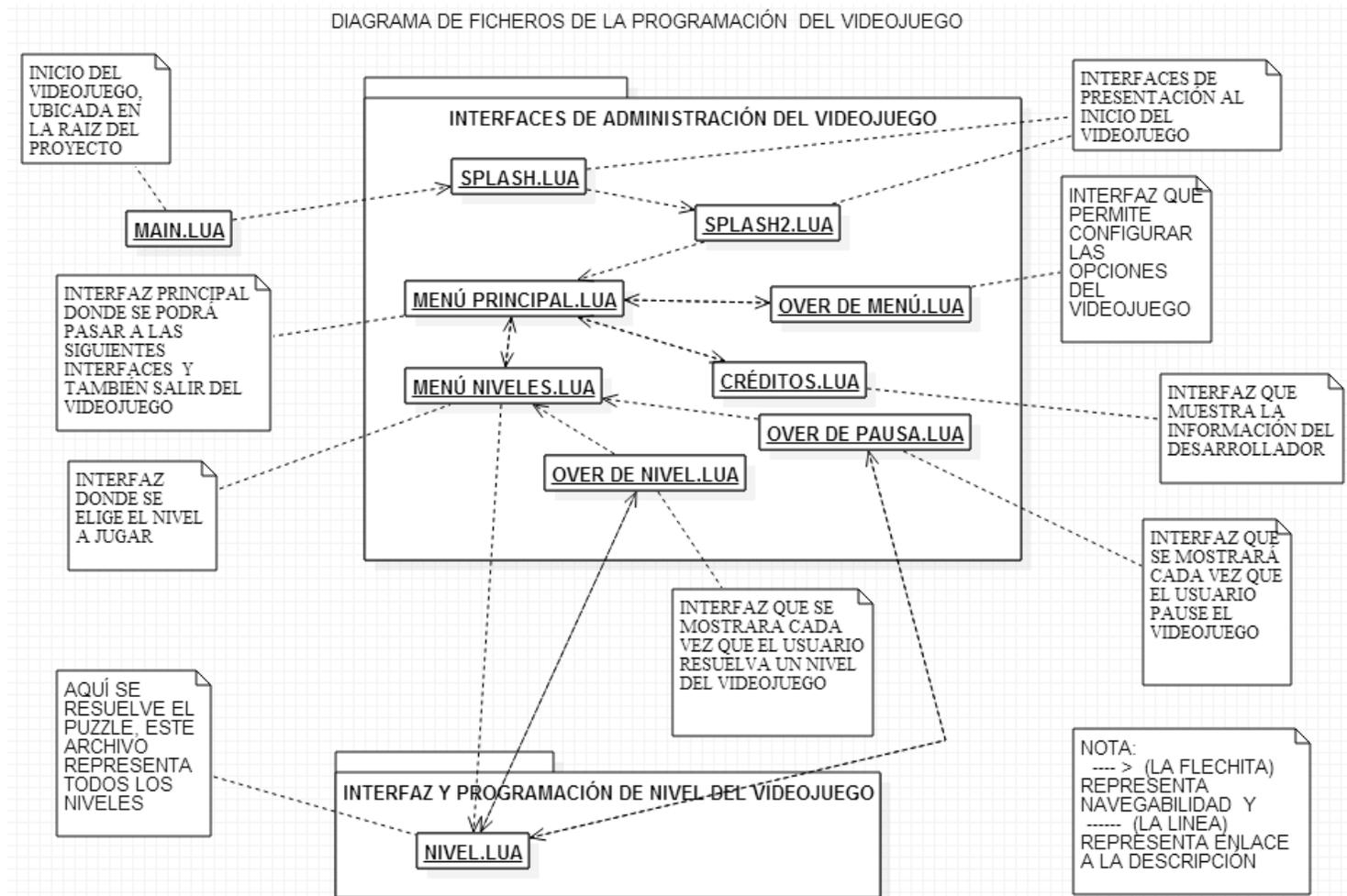


Figura 48: Diseño de paquetes

Fuente: elaboración propia

3.2.4. Elaboración

En esta fase se realizaron nueve iteraciones para obtener el producto definido en la fase de planificación.

Además se realizaron 2 actividades: la planificación de la iteración, y el cierre de la iteración y las tareas de desarrollo de las características y seguimiento de la iteración.

A. Planificación de la iteración

Se creó el plan para cada iteración, lo cual incluyó las tareas de definir objetivos de la iteración, seleccionar requisitos a desarrollar y refinarlos para obtener componentes más pequeños a ser desarrollados, el flujo de las tareas realizadas se muestra a continuación:



Figura 49. Flujo de tareas de la planificación de la iteración de la elaboración

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de estas tareas se muestra a continuación:

Definición de objetivos, selección de características y refinación de características

Se realizaron nueve iteraciones, cada iteración contempla los requisitos funcionales a ser realizados, la manera en la que se realizaron y las interfaces necesarias para satisfacer dichos requisitos; la siguiente tabla muestra dicha información:

Tabla 29: *Relación entre las iteraciones, requisitos e interfaces*

Iteración	Requisitos a desarrollar	Refinamiento de requisitos	Interfaces
Iteración 1	Requisitos 1, 2, 3, 4, 12, 16, 31 y 32.	Desarrollar cada requisito de manera secuencial.	Interfaz primer splash, segundo splash, menú principal y menú de niveles.

Continuación (Tabla 29: <i>Relación entre las iteraciones, requisitos e interfaces</i>)			
Iteración	Requisitos a desarrollar	Refinamiento de requisitos	Interfaces
Iteración 2	Requisitos 6, 7, 8, 13, 14, 15, 19, 23, 28 y 29.	Desarrollar cada requisito de manera secuencial, en cuanto a los requisitos repetidos desarrollar la parte correspondiente a la interfaz de esta iteración.	Interfaz over relay de menú principal, pausa y nivel
Iteración 3	Requisitos 9, 10, 11 y 33 .	Desarrollar cada requisito de manera secuencial	Interfaz de primer y segundo tutorial e interfaz de créditos
Iteración 4	Requisitos 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 28, 30 y 33 .	Desarrollar cada requisito de manera secuencial, en cuanto a los requisitos repetidos desarrollar la parte correspondiente a la interfaz de esta iteración.	Niveles 1, 2 y 3
Iteración 5	Requisitos 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 28, 30, y 33 .	Desarrollar cada requisito de manera secuencial, en cuanto a los requisitos repetidos desarrollar la parte correspondiente a la interfaz de esta iteración.	Niveles 4, 5, 6, 7 y 8
Iteración 6	Requisitos 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27 (excepto nivel 9 y 10), 28, 30 y 33 .	Desarrollar cada requisito de manera secuencial, en cuanto a los requisitos repetidos desarrollar la parte correspondiente a la interfaz de esta iteración.	Niveles 9, 10, 11 y 12
Iteración 7	Requisitos 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27 (excepto nivel 9 y 10), 28, 30 y 33 .	Desarrollar cada requisito de manera secuencial, en cuanto a los requisitos repetidos desarrollar la parte correspondiente a la interfaz de esta iteración.	Niveles 13, 14 y 15
Iteración 8	Requisitos 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27 (excepto nivel 9 y 10), 28, 30 y 33 .	Desarrollar cada requisito de manera secuencial, en cuanto a los requisitos repetidos desarrollar la parte correspondiente a la interfaz de esta iteración.	Niveles 16, 17 y 18
Iteración 9	Requisitos 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27 (excepto nivel 9 y 10), 28, 30 y 33 .	Desarrollar cada requisito de manera secuencial, en cuanto a los requisitos repetidos desarrollar la parte correspondiente a la interfaz de esta iteración.	Niveles 19 y 20

Fuente: elaboración propia

B. Desarrollo de las características

En esta tarea se programó o desarrolló las características ya refinadas. Esta fue la actividad más extensa en cada iteración.

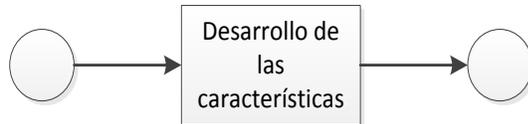


Figura 50. Flujo de la tarea desarrollo de las características

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de esta tarea se puede ver en el **anexo nro. 1 “CÓDIGO FUENTE DEL VIDEOJUEGO” (digital)**, que contiene el código fuente del videojuego con la versión final.

C. Seguimiento de la iteración

En esta tarea se controló que las características desarrolladas estén dentro del marco de los objetivos y requisitos planteados y se aseguró que no haya problemas técnicos durante el desarrollo. El flujo de esta tarea se muestra a continuación:

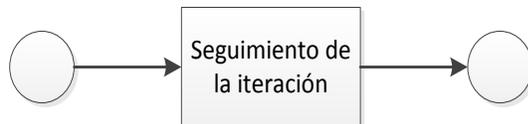


Figura 51. Flujo de la tarea seguimiento de la iteración

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de esta tarea se muestra a continuación:

Tabla 30: Seguimiento de las iteraciones

Iteración	Problemas técnicos durante el desarrollo	Realizado conforme a los requisitos
Iteración 1	Ninguno	Si
Iteración 2	Ninguno	Si
Iteración 3	Ninguno	Si
Iteración 4	Ninguno	Si
Iteración 5	Ninguno	Si
Iteración 6	Ninguno	Si
Iteración 7	Ninguno	Si
Iteración 8	Ninguno	Si
Iteración 9	Ninguno	Si

Fuente: elaboración propia

D. Cierre de la iteración

En esta actividad se realizaron tres tareas que fueron: evaluar el estado del videojuego donde ya se tuvo una versión del videojuego, también se evaluó toda la iteración realizada para verificar que se iba trabajando conforme a lo planeado y se actualizó el plan del proyecto con los nuevos resultados obtenidos, el flujo de estas tareas se muestra a continuación:

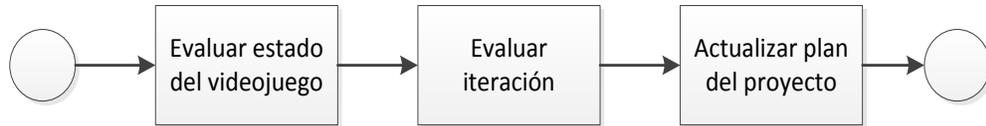


Figura 52. Flujo de tareas del cierre de la iteración

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de estas tareas se muestra a continuación:

Evaluación del estado del videojuego, evaluación de iteración, actualización del plan del proyecto

El videojuego contó con 10 versiones, cada una de las cuales pudo ser ejecutada de manera correcta, agregándose cada vez más las características o requisitos restantes en cada iteración, luego de la cual fue actualizado el plan del proyecto, la siguiente tabla detalla la información de las versiones:

Tabla 31: *Versiones del videojuego*

Versión	Características	Porcentaje de cumplimiento de los requisitos	Iteraciones realizadas	Versión jugable
1.0	Desarrollo de la interfaz primer splash, segundo splash, menú principal y menú de niveles y corrección de errores encontrados.	10%	1	Si
2.0	Desarrollo de interfaz over relay de menú principal, pausa y nivel y corrección de errores encontrados.	20%	2	Si
3.0	Desarrollo de la interfaz de primer y segundo tutorial e interfaz de créditos y corrección de errores encontrados.	30%	3	Si
4.0	Desarrollo de los niveles 1, 2 y 3 y corrección de errores encontrados.	40%	4	Si

Continuación (Tabla 31: <i>Versiones del videojuego</i>)				
Versión	Características	Porcentaje de cumplimiento de los requisitos	Iteraciones realizadas	Versión jugable
5.0	Desarrollo de los niveles 4, 5, 6, 7 y 8 y corrección de errores encontrados.	50%	5	Si
6.0	Desarrollo de los niveles 9, 10, 11 y 12 y corrección de errores encontrados.	60%	6	Si
7.0	Desarrollo de los niveles 13, 14 y 15 y corrección de errores encontrados.	70%	7	Si
8.0	Desarrollo de los niveles 16, 17 y 18 y corrección de errores encontrados.	80%	8	Si
9.0	Desarrollo de los niveles 19 y 20 y corrección de errores encontrados.	90%	9	Si
10.0	Corrección de errores del videojuego completamente desarrollado.	100%	9	Si

Fuente: elaboración propia

3.2.5. Beta

En esta fase se realizaron tres iteraciones para verificar que el videojuego planteado este correctamente implementado, ello incluyó la realización de las actividades de planificación de la iteración y corrección del videojuego, así como la tarea de verificación del videojuego, cabe mencionar que debido a que la fase de diseño se basó en la metodología RUP, entonces con la finalidad de guardar concordancia y asegurar una correcta revisión, la verificación del funcionamiento del videojuego se basó en el artefacto Casos de prueba (los de Caja negra) de la metodología RUP.

A. Planificación de la iteración

Se realizaron las tareas de planificar la iteración que consistió en definir qué requisitos funcionales y no funcionales se debían revisar, así también se definió a los verificadores beta, luego se realizó la tarea de distribuir la versión beta que fue verificada, el flujo de dichas tareas se muestra a continuación:

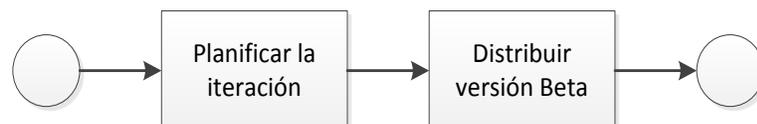


Figura 53: Flujo de tareas de la planificación de la iteración de la fase beta

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de estas tareas se muestra a continuación:

Planificación de la iteración, distribución de la versión beta

Se revisión se llevó a cabo para cada requisito funcional, el verificador beta revisó el cumplimiento de cada uno de estos requisitos y los reportó de acuerdo a los Casos de prueba de Caja negra de la metodología RUP, para este caso el verificador beta fue quien ejecutó el presente proyecto.

B. Verificación del videojuego

Se realizaron los Casos de prueba de Caja negra más relevantes para las especificaciones de diseño de la interfaces de la fase de diseño, las cuales fueron diseñados a partir de los requisitos especificados, el flujo de esta tarea es el siguiente:

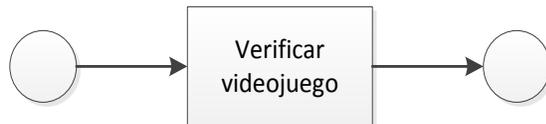


Figura 54. Flujo de la tarea verificación del videojuego

Fuente: <http://www.gemserk.com/sum/>

PRUEBAS DE CAJA NEGRA

Identificación de Condiciones externas, Clases de equivalencia válidas y Clases de equivalencia inválidas

Las condiciones externas basadas en los resultados que se espera que genere y que no genere el videojuego se detallan en la siguiente tabla:

Tabla 32: Identificación de CE, CEV, CEI

Nro.	Interfaz probada	CE (Condiciones externas)	CEV (Clases de equivalencia válidas)	CEI (Clases de equivalencia inválidas)
1	Primer splash	Mostrar la interfaz "primer splash"	Se muestre la interfaz de "primer splash"	No se muestre la interfaz de "primer splash"
2	Segundo splash	Mostrar la interfaz "segundo splash"	Se muestre la interfaz de "segundo splash"	No se muestre la interfaz de "segundo splash"
3	Menú principal	Mostrar la interfaz "menú principal"	Se muestre la interfaz de "menú principal"	No se muestre la interfaz de "menú principal"
4	Menú principal	Mostrar la interfaz "over relay de menú"	Se muestre la interfaz de "over relay de menú"	No se muestre la interfaz de "over relay de menú"

Continuación (Tabla 32: <i>Identificación de CE, CEV, CEI</i>)				
Nro.	Interfaz probada	CE (Condiciones externas)	CEV (Clases de equivalencia validas)	CEI (Clases de equivalencia invalidas)
5	Menú principal	Reiniciar el progreso del juego	Se reinicie el progreso del juego	No se reinicie el progreso del juego
6	Menú principal	Cambiar el idioma de Español a Inglés	Se cambie el idioma de Español a Inglés	No se cambie el idioma de Español a Inglés
7	Menú principal	Cambiar el idioma de Inglés a Español	Se cambie el idioma de Inglés a Español	No se cambie el idioma de Inglés a Español
8	Menú principal	Ajustar el volumen	Se ajuste el volumen	No se ajuste el volumen
9	Menú principal	Mostrar la "interfaz créditos"	Se muestre la "interfaz créditos"	No se muestre la "interfaz créditos"
10	Menú principal	Cerrar el videojuego	Se cierre el videojuego	No se cierre el videojuego
11	Menú de niveles	Mostrar "la interfaz menú de niveles"	Se muestre la interfaz "menú de niveles"	No se muestre la interfaz "menú de niveles"
12	Menú de niveles	Mostrar la "interfaz de tutorial"	Se muestre la "interfaz de tutorial"	No se muestre la "interfaz de tutorial"
13	Menú de niveles	No mostrar la "interfaz de nivel"	No se muestre la "interfaz de nivel"	Se muestre la "interfaz de nivel"
14	Menú de niveles	Retornar a la interfaz anterior (menú principal)	Se retorne a la interfaz anterior (menú principal)	No se retorne a la interfaz anterior (menú principal)
15	Menú de niveles	Mantener el progreso del juego	Se mantenga el progreso del juego	No se mantenga el progreso del juego
16	Primer tutorial	Mostrar la interfaz del nivel 1	Se muestre la interfaz del nivel 1	No se muestre la interfaz del nivel 1
17	Interfaz de nivel (10 primeros)	Mover al personaje	Se pueda mover al personaje	No se pueda mover al personaje
18	Interfaz de nivel (10 primeros)	Hacer desaparecer el mensaje de consejo en pantalla y empezar a hacer correr el tiempo	Desaparezca el mensaje de consejo en pantalla y el tiempo empiece a correr	No desaparezca el mensaje de consejo en pantalla y el tiempo no empiece a correr
19	Interfaz de nivel (10 primeros)	Mover el personaje hacia la derecha	Se mueva el personaje hacia la derecha	No se mueva el personaje hacia la derecha
20	Interfaz de nivel (10 primeros)	Mover el personaje hacia la izquierda	Se mueva el personaje hacia la izquierda	No se mueva el personaje hacia la izquierda
21	Interfaz de nivel (10 primeros)	Mover el personaje hacia arriba	Se mueva el personaje hacia arriba	No se mueva el personaje hacia arriba
22	Interfaz de nivel (10 primeros)	Mover el personaje hacia abajo	Se mueva el personaje hacia abajo	No se mueva el personaje hacia abajo

Continuación (Tabla 32: <i>Identificación de CE, CEV, CEI</i>)				
Nro.	Interfaz probada	CE (Condiciones externas)	CEV (Clases de equivalencia validas)	CEI (Clases de equivalencia invalidas)
23	Interfaz de nivel (10 primeros)	Mostrar la interfaz "over de nivel"	Se muestre la interfaz "over de nivel"	No se muestre la interfaz "over de nivel"
24	Interfaz de nivel (10 primeros)	No cambiar la dirección de desplazamiento del personaje ni pausar el juego	No se cambie la dirección de desplazamiento del personaje ni se pause el juego	Se cambie la dirección de desplazamiento del personaje y se pause el juego
25	Interfaz de nivel (10 primeros)	No hacer desaparecer el mensaje en pantalla	No desaparezca el mensaje en pantalla	Desaparezca el mensaje en pantalla
26	Interfaz de nivel (10 primeros)	Mostrar la interfaz de "over de pausa"	Se muestre la interfaz de "over de pausa"	No se muestre la interfaz de "over de pausa"
27	Interfaz de nivel (10 primeros)	Continuar con el juego	Se continúe con el juego	No se continúe con el juego
28	Interfaz de nivel (10 primeros)	Regresar al menú de niveles	Se regrese al menú de niveles	No se regrese al menú de niveles
29	Interfaz de nivel (10 primeros)	Reiniciar el juego	Se reinicie el juego	No se reinicie el juego
30	Interfaz segundo tutorial	Mostrar la interfaz del nivel 11	Se muestre la interfaz del nivel 11	No se muestre la interfaz del nivel 11
31	Interfaz de nivel (10 últimos)	Mover al personaje	Se pueda mover al personaje	No se pueda mover al personaje
32	Interfaz de nivel (10 últimos)	Hacer desaparecer el mensaje de consejo en pantalla y empezar a hacer correr el tiempo	Desaparezca el mensaje de consejo en pantalla y el tiempo empiece a correr	No desaparezca el mensaje de consejo en pantalla y el tiempo no empiece a correr
33	Interfaz de nivel (10 últimos)	Mover el personaje hacia la derecha	Se mueva el personaje hacia la derecha	No se mueva el personaje hacia la derecha
34	Interfaz de nivel (10 últimos)	Mover el personaje hacia la izquierda	Se mueva el personaje hacia la izquierda	No se mueva el personaje hacia la izquierda
35	Interfaz de nivel (10 últimos)	Mover el personaje hacia arriba	Se mueva el personaje hacia arriba	No se mueva el personaje hacia arriba
36	Interfaz de nivel (10 últimos)	Mover el personaje hacia abajo	Se mueva el personaje hacia abajo	No se mueva el personaje hacia abajo
37	Interfaz de nivel (10 últimos)	Ya no mover al primer personaje que llego a su meta	Ya no se mueva al primer personaje que llego a su meta	Aún se mueva al primer personaje que llego a su meta

Continuación (Tabla 32: <i>Identificación de CE, CEV, CEI</i>)				
Nro.	Interfaz probada	CE (Condiciones externas)	CEV (Clases de equivalencia validas)	CEI (Clases de equivalencia invalidas)
38	Interfaz de nivel (10 últimos)	Mostrar la interfaz "over de nivel"	Se muestre la interfaz "over de nivel"	No se muestre la interfaz "over de nivel"
39	Interfaz de nivel (10 últimos)	No cambiar la dirección de desplazamiento del personaje ni pausar el juego	No se cambie la dirección de desplazamiento del personaje ni se pause el juego	Se cambie la dirección de desplazamiento del personaje y se pause el juego
40	Interfaz de nivel (10 últimos)	No hacer desaparecer el mensaje en pantalla	No desaparezca el mensaje en pantalla	Desaparezca el mensaje en pantalla
41	Interfaz de nivel (10 últimos)	Mostrar la interfaz de "over de pausa"	Se muestre la interfaz de "over de pausa"	No se muestre la interfaz de "over de pausa"
42	Interfaz de nivel (10 últimos)	Continuar con el juego	Se continúe con el juego	No se continúe con el juego
43	Interfaz de nivel (10 últimos)	Regresar al menú de niveles	Se regrese al menú de niveles	No se regrese al menú de niveles
44	Interfaz de nivel (10 últimos)	Reiniciar el juego	Se reinicie el juego	No se reinicie el juego
45	Interfaz over de nivel	Guardar los datos del nivel resuelto, mostrar el puntaje representado con estrellas y desbloquear el siguiente nivel	Se guarden los datos del nivel resuelto, se muestre el puntaje representado con estrellas y se desbloquee el siguiente nivel	No se guarden los datos del nivel resuelto, ni se muestre el puntaje representado con estrellas y tampoco se desbloquee el siguiente nivel
46	Interfaz over de nivel	Mostrar el siguiente nivel	Se muestre el siguiente nivel	No se muestre el siguiente nivel
47	Interfaz over de nivel	Reiniciar el juego	Se reinicie el juego	No se reinicie el juego
48	Interfaz over de nivel	Regresar al menú de niveles	Se regrese al menú de niveles	No se regrese al menú de niveles

Fuente: elaboración propia

Casos de prueba para las clases de equivalencia válidas

Los Casos de prueba de Caja negra para las clases de equivalencia válidas son:

Caso de prueba 1

Tabla 33: Caso de prueba de la interfaz primer splash (clase válida)

Escenario probado		INTERFAZ DE PRIMER SPLASH			
Fecha:		11/11/2015	Avance %	100%	
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de primer splash del videojuego, el cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T1	Se comprobará que el videojuego muestre la interfaz de primer splash	Ejecutar el videojuego	Se muestre la interfaz de primer splash	Sucedió	Pass

Fuente: elaboración propia

Caso de prueba 2

Tabla 34: Caso de prueba de la interfaz de segundo splash (clase válida)

Escenario probado		INTERFAZ DE SEGUNDO SPLASH			
Fecha:		11/11/2015	Avance %	100%	
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de segundo splash del videojuego, el cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T2	Se comprobará que el videojuego muestre la interfaz de segundo splash	-	Se muestre la interfaz de segundo splash	Sucedió	Pass

Fuente: elaboración propia

Caso de prueba 3

Tabla 35: Caso de prueba de la interfaz menú principal (clase válida)

Escenario probado	INTERFAZ DE MENÚ PRINCIPAL				
Fecha:	12/11/2015	Avance %	100%		
Tester:	Iván Marcial Palomino Taype				
Descripción de la Prueba:	Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de menú principal del videojuego, el cual fue detallado en los requisitos funcionales.				
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T3	Se comprobará que el videojuego muestre la interfaz menú principal	-	Se muestre la interfaz menú principal	Sucedió	Pass
T4	Se comprobará que el videojuego muestre la interfaz ove relay de menú	Toque en el botón "Opciones"	Se muestre la interfaz over relay de menú	Sucedió	Pass
T4.1	Se comprobará que se reinicia el progreso del juego	Toque en el botón "Reiniciar Juego"	Se reinicie el progreso del juego	Sucedió	Pass
T4.2	Se comprobará que se pueda cambiar el idioma de Español a Inglés	<ul style="list-style-type: none"> • Toque en el botón "Inglés" • Tocar el botón para salir de la interfaz 	Se cambie el idioma de Español a Inglés	Sucedió	Pass
T4.3	Se comprobará que se pueda cambiar el idioma de Inglés a Español	<ul style="list-style-type: none"> • Toque en el botón "Spanish" • Tocar el botón para salir de la interfaz 	Se cambie el idioma de Inglés a Español	Sucedió	Pass

Continuación (Tabla 35: Caso de prueba de la interfaz menú principal (clase válida))					
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T4.4	Se comprobará que se puede ajustar el volumen	Toque en botón "+", toque en botón "-"	Se ajuste el volumen	Sucedió	Pass
T5	Se comprobará que el videojuego muestre la interfaz créditos	Toque en el botón "Créditos"	Se muestre la interfaz créditos	Sucedió	Pass
T6	Se comprobará que el videojuego puede cerrar	Toque en el botón "Salir"	Se cierre el videojuego	Sucedió	Pass

Fuente: elaboración propia

Caso de prueba 4

Tabla 36: Caso de prueba de la interfaz menú de niveles (clase válida)

Escenario probado	INTERFAZ MENÚ DE NIVELES				
Fecha:	13/11/2015	Avance %	100%		
Tester:	Iván Marcial Palomino Taype				
Descripción de la Prueba:	Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz menú de niveles del videojuego, el cual fue detallado en los requisitos funcionales.				
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T7	Se comprobará que el videojuego muestre la interfaz menú de niveles	Toque en el botón "Jugar" del menú anterior	Se muestre la interfaz menú de niveles	Sucedió	Pass
T8	Se comprobará que el videojuego muestre la interfaz de primer tutorial	Toque en el botón que representa al nivel 1	Se muestre la interfaz de primer tutorial	Sucedió	Pass
T9	Se comprobará que el videojuego no permita a acceso a niveles no desbloqueados	Toque en el botón que representa un nivel bloqueado	No se muestre la interfaz de nivel	Sucedió	Pass

Continuación (Tabla 36: Caso de prueba de la interfaz menú de niveles (clase válida))					
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T10	Se comprobará que se puede regresar a la interfaz anterior	Toque en el botón que representa una flecha con dirección a la izquierda	Se retorne a la interfaz anterior (menú principal)	Sucedió	Pass
T11	Se probará que se guarda el progreso del usuario aun después de cerrar el videojuego	<ul style="list-style-type: none"> • Toque en el botón salir de menú principal • Ejecutar el juego • Toque en botón "Jugar" 	Se mantenga el progreso del juego	Sucedió	Pass

Fuente: elaboración propia

Caso de prueba 5

Tabla 37: Caso de prueba de la interfaz primer tutorial (clase válida)

Escenario probado	INTERFAZ DE PRIMER TUTORIAL				
Fecha:	16/11/2015	Avance %	100%		
Tester:	Iván Marcial Palomino Taype				
Descripción de la Prueba:	Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de primer tutorial del videojuego, el cual fue detallado en los requisitos funcionales.				
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T12	Se comprobará que el videojuego muestre la interfaz del nivel 1	Toque en el botón OK	Se muestre la interfaz del nivel 1	Sucedió	Pass

Fuente: elaboración propia

Caso de prueba 6

Tabla 38: Caso de prueba de la interfaz 10 primeros niveles (clase válida)

Escenario probado		INTERFAZ DE NIVEL (10 primeros)			
Fecha:		17/11/2015 - 18/11/2015	Avance %	100%	
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de los primeros 10 niveles del videojuego, lo cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T13	Se comprobará que solo se puede mover al personaje cuando se presiona la imagen que representa el control especial que activa al personaje	Toque en la imagen que representa el control especial que activa al personaje	Se pueda mover al personaje	Sucedió	Pass
T14	Se comprobará que desaparece el mensaje en pantalla y el tiempo empieza a correr solo cuando se presiona la imagen que representa el control especial que activa al personaje	Toque en la imagen que representa el control especial que activa al personaje	Desaparezca el mensaje de consejo en pantalla y el tiempo empiece a correr	Sucedió	Pass
T15	Se comprobará que se puede mover al personaje a la derecha	Toque en la imagen que representa el control de dirección derecha	Se mueva el personaje hacia la derecha	Sucedió	Pass
T16	Se comprobará que se puede mover al personaje a la izquierda	Toque en la imagen que representa el control de dirección izquierda	Se mueva el personaje hacia la izquierda	Sucedió	Pass
T17	Se comprobará que se puede mover al personaje hacia arriba	Toque en la imagen que representa el control de dirección arriba	Se mueva el personaje hacia arriba	Sucedió	Pass

Continuación (Tabla 38: <i>Caso de prueba de la interfaz 10 primeros niveles (clase válida)</i>)					
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T18	Se comprobará que se puede mover al personaje hacia abajo	Toque en la imagen que representa el control de dirección abajo	Se mueva el personaje hacia abajo	Sucedió	Pass
T19	Se comprobará que se muestra la interfaz de over relay de nivel cuando se soluciona el nivel	Solucionar el nivel	Se muestre la interfaz over de nivel	Sucedió	Pass
T20	Se comprobará que mientras el personaje se desplaza no puede cambiar de dirección ni pausar el juego	-	No se cambie la dirección de desplazamiento del personaje ni se pause el juego	Sucedió	Pass
T21	Se probará que al no activar al personaje no desaparece el mensaje indicando que se debe activarlo	Toque en cualquiera de las flechas de dirección	No desaparezca el mensaje en pantalla	Sucedió	Pass
T22	Se probará que se muestra el over de pausa	Toque en la imagen que representa la pausa	Se muestre la interfaz de over de pausa	Sucedió	Pass
T22.1	Se probará que se puede reanudar el juego desde el over relay de pausa	Toque en botón Continuar del over relay de pausa	Se continúe con el juego	Sucedió	Pass
T22.2	Se probará que se puede salir del nivel y regresar al menú de niveles desde el over relay de pausa	Toque en el botón Menú de niveles del over relay de pausa	Se regrese al menú de niveles	Sucedió	Pass
T23	Se probará que se reinicia el juego desde la interfaz de nivel	Toque en la imagen que representa reiniciar el juego	Se reinicie el juego	Sucedió	Pass

Fuente: elaboración propia

Caso de prueba 7

Tabla 39: Caso de prueba de la interfaz de segundo tutorial (clase válida)

Escenario probado		INTERFAZ DE SEGUNDO TUTORIAL			
Fecha:		19/11/2015	Avance %		100%
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de segundo tutorial del videojuego, el cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T24	Se comprobará que el videojuego muestre la interfaz del nivel 11	Toque en el botón OK	Se muestre la interfaz del nivel 11	Sucedió	Pass

Fuente: elaboración propia

Caso de prueba 8

Tabla 40: Caso de prueba de la interfaz 10 últimos niveles (clase válida)

Escenario probado		INTERFAZ DE NIVEL (10 últimos)			
Fecha:		20/11/2015 - 23/11/2015	Avance %		100%
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de los 10 últimos niveles del videojuego, lo cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T25	Se comprobará que solo se puede mover al personaje cuando se presiona la imagen que representa el control especial que activa al personaje	Toque en la imagen que representa el control especial que activa al personaje	Se pueda mover al personaje	Sucedió	Pass
T26	Se comprobará que desaparece el mensaje en pantalla y el tiempo empieza a correr solo cuando se presiona la imagen que representa el control especial que activa al personaje	Toque en la imagen que representa el control especial que activa al personaje	Desaparezca el mensaje de consejo en pantalla y el tiempo empiece a correr	Sucedió	Pass

Continuación (Tabla 40: *Caso de prueba de la interfaz 10 últimos niveles (clase válida)*)

Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T27	Se comprobará que se puede mover al personaje a la derecha	Toque en la imagen que representa el control de dirección derecha	Se mueva el personaje hacia la derecha	Sucedió	Pass
T28	Se comprobará que se puede mover al personaje a la izquierda	Toque en la imagen que representa el control de dirección izquierda	Se mueva el personaje hacia la izquierda	Sucedió	Pass
T29	Se comprobará que se puede mover al personaje hacia arriba	Toque en la imagen que representa el control de dirección arriba	Se mueva el personaje hacia arriba	Sucedió	Pass
T30	Se comprobará que se puede mover al personaje hacia abajo	Toque en la imagen que representa el control de dirección abajo	Se mueva el personaje hacia abajo	Sucedió	Pass
T31	Se comprobará que el primer personaje en llegar a la meta ya no pueda ser movido	Llevar el primer personaje a su meta	Ya no se mueva al primer personaje que llego a su meta	Sucedió	Pass
T32	Se comprobará que se muestra la interfaz de over relay de nivel cuando se soluciona el nivel	Llevar el segundo personaje a su meta	Se muestre la interfaz over de nivel	Sucedió	Pass
T33	Se comprobará que mientras el personaje se desplaza no puede cambiar de dirección ni pausar el juego	-	No se cambie la dirección de desplazamiento del personaje ni se pause el juego	Sucedió	Pass

Continuación (Tabla 40: Caso de prueba de la interfaz 10 últimos niveles (clase válida))					
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T34	Se probará que al no activar al personaje no desaparece el mensaje indicando que se debe activarlo	Toque en cualquiera de las flechas de dirección	No desaparece el mensaje en pantalla	Sucedió	Pass
T35	Se probará que se muestra el over de pausa	Toque en la imagen que representa la pausa	Se muestre la interfaz de over de pausa	Sucedió	Pass
T35.1	Se probará que se puede reanudar el juego desde el over relay de pausa	Toque en botón Continuar del over relay de pausa	Se continúe con el juego	Sucedió	Pass
T35.2	Se probará que se puede salir del nivel y regresar al menú de niveles desde el over relay de pausa	Toque en el botón Menú de niveles del over relay de pausa	Se regrese al menú de niveles	Sucedió	Pass
T36	Se probará que se reinicia el juego desde la interfaz de nivel	Toque en la imagen que representa reiniciar el juego	Se reinicie el juego	Sucedió	Pass

Fuente: elaboración propia

Caso de prueba 9

Tabla 41: Caso de prueba de la interfaz over relay de nivel (clase válida)

Escenario probado		INTERFAZ DE OVER DE NIVEL			
Fecha:		24/11/2015	Avance %		100%
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de over relay de nivel del videojuego, el cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T37	Se comprobará que se guardan los datos del nivel resuelto, se muestra el puntaje representado con estrellas y se desbloquea el siguiente nivel	Resolver el nivel	Se guarden los datos del nivel resuelto, se muestre el puntaje representado con estrellas y se desbloquee el siguiente nivel	Sucedió	Pass
T38	Se comprobará que se puede pasar al siguiente nivel	Toque en la imagen que representa siguiente nivel	Se muestre el siguiente nivel	Sucedió	Pass
T39	Se probará que se reinicia el juego desde la interfaz de over relay de nivel	Toque en la imagen que representa reiniciar el juego	Se reinicie el juego	Sucedió	Pass
T40	Se probará que se puede salir del nivel y regresar al menú de niveles	Toque en el botón menú de niveles	Se regrese al menú de niveles	Sucedió	Pass

Fuente: elaboración propia

Casos de prueba para las clases de equivalencia inválidas

Los Casos de prueba de Caja negra para las clases de equivalencia inválidas son los siguientes:

Caso de prueba 1

Tabla 42: Caso de prueba de la interfaz primer splash (clase inválida)

Escenario probado		INTERFAZ DE PRIMER SPLASH			
Fecha:		11/11/2015	Avance %	100%	
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de primer splash del videojuego, el cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T41	Se comprobará que el videojuego no muestre la interfaz de primer splash	Ejecutar el videojuego	No se muestre la interfaz de primer splash	No sucedió	Pass

Fuente: elaboración propia

Caso de prueba 2

Tabla 43: Caso de prueba de la interfaz de segundo splash (clase inválida)

Escenario probado		INTERFAZ DE SEGUNDO SPLASH			
Fecha:		11/11/2015	Avance %	100%	
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de segundo splash del videojuego, el cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T42	Se comprobará que el videojuego no muestre la interfaz de segundo splash	-	No se muestre la interfaz de segundo splash	No sucedió	Pass

Fuente: elaboración propia

Caso de prueba 3

Tabla 44: Caso de prueba de la interfaz menú principal (clase inválida)

Escenario probado		INTERFAZ DE MENÚ PRINCIPAL			
Fecha:		12/11/2015	Avance %		100%
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de menú principal del videojuego, el cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T43	Se comprobará que el videojuego no muestre la interfaz menú principal	-	No se muestre la interfaz menú principal	No sucedió	Pass
T44	Se comprobará que el videojuego no muestre la interfaz ove relay de menú	Toque en el botón "Opciones"	No se muestre la interfaz over relay de menú	No sucedió	Pass
T44.1	Se comprobará que no se reinicia el progreso del juego	Toque en el botón "Reiniciar Juego"	No se reinicie el progreso del juego	No sucedió	Pass
T44.2	Se comprobará que no se pueda cambiar el idioma de Español a Inglés	<ul style="list-style-type: none"> • Toque en el botón "Inglés" • Tocar el botón para salir de la interfaz 	No se cambie el idioma de Español a Inglés	No sucedió	Pass
T44.3	Se comprobará que no se pueda cambiar el idioma de Inglés a Español	<ul style="list-style-type: none"> • Toque en el botón "Spanish" • Tocar el botón para salir de la interfaz 	No se cambie el idioma de Inglés a Español	No sucedió	Pass
T44.4	Se comprobará que no se puede ajustar el volumen	Toque en botón "+", toque en botón "-"	No se ajuste el volumen	No sucedió	Pass
T45	Se comprobará que el videojuego no muestre la interfaz créditos	Toque en el botón "Créditos"	No se muestre la interfaz créditos	No sucedió	Pass

Continuación (Tabla 44: Caso de prueba de la interfaz menú principal (clase inválida))					
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T46	Se comprobará que el videojuego no se puede cerrar	Toque en el botón "Salir"	No se cierre el videojuego	No sucedió	Pass

Fuente: elaboración propia

Caso de prueba 4

Tabla 45: Caso de prueba de la interfaz menú de niveles (clase inválida)

Escenario probado	INTERFAZ MENÚ DE NIVELES				
Fecha:	13/11/2015	Avance %	100%		
Tester:	Iván Marcial Palomino Taype				
Descripción de la Prueba:	Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz menú de niveles del videojuego, el cual fue detallado en los requisitos funcionales.				
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T47	Se comprobará que el videojuego no muestre la interfaz menú de niveles	Toque en el botón "Jugar" del menú anterior	No se muestre la interfaz menú de niveles	No sucedió	Pass
T48	Se comprobará que el videojuego no muestre la interfaz de primer tutorial	Toque en el botón que representa al nivel 1	No se muestre la interfaz de primer tutorial	No sucedió	Pass
T49	Se comprobará que el videojuego permita el acceso a niveles no desbloqueados	Toque en el botón que representa un nivel bloqueado	Se muestre la interfaz de nivel	No sucedió	Pass
T50	Se comprobará que no se puede regresar a la interfaz anterior	Toque en el botón que representa una flecha con dirección a la izquierda	No se retorne a la interfaz anterior (menú principal)	No sucedió	Pass
T51	Se probará que no se guarda el progreso del usuario después de cerrar el videojuego	<ul style="list-style-type: none"> • Toque en el botón salir de menú principal • Ejecutar el juego • Toque en botón "Jugar" 	No se mantenga el progreso del juego	No sucedió	Pass

Fuente: elaboración propia

Caso de prueba 5

Tabla 46: Caso de prueba de la interfaz primer tutorial (clase inválida)

Escenario probado	INTERFAZ DE PRIMER TUTORIAL				
Fecha:	16/11/2015	Avance %	100%		
Tester:	Iván Marcial Palomino Taype				
Descripción de la Prueba:	Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de primer tutorial del videojuego, el cual fue detallado en los requisitos funcionales.				
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T52	Se comprobará que el videojuego no muestre la interfaz del nivel 1	Toque en el botón OK	No se muestre la interfaz del nivel 1	No sucedió	Pass

Fuente: elaboración propia

Caso de prueba 6

Tabla 47: Caso de prueba de la interfaz 10 primeros niveles (clase inválida)

Escenario probado	INTERFAZ DE NIVEL (10 primeros)				
Fecha:	17/11/2015 - 18/11/2015	Avance %	100%		
Tester:	Iván Marcial Palomino Taype				
Descripción de la Prueba:	Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de los primeros 10 niveles del videojuego, lo cual fue detallado en los requisitos funcionales.				
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T53	Se comprobará que se puede mover al personaje cuando se presionan las flechas de dirección, pausa o reiniciar	Toque en las flechas de dirección, pausa o reiniciar	No se pueda mover al personaje	Sucedió	Pass
T54	Se comprobará que desaparece el mensaje en pantalla y el tiempo empieza a correr cuando se presiona las flechas de control, pausa o reiniciar	Toque en la imagen que representa las flechas de dirección, pausa o reiniciar	No desaparezca el mensaje de consejo en pantalla y el tiempo no empiece a correr	Sucedió	Pass

Continuación (Tabla 47: Caso de prueba de la interfaz 10 primeros niveles (clase inválida))

Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T55	Se comprobará que no se puede mover al personaje a la derecha	Toque en la imagen que representa el control de dirección derecha	No se mueva el personaje hacia la derecha	No sucedió	Pass
T56	Se comprobará que no se puede mover al personaje a la izquierda	Toque en la imagen que representa el control de dirección izquierda	No se mueva el personaje hacia la izquierda	No sucedió	Pass
T57	Se comprobará que no se puede mover al personaje hacia arriba	Toque en la imagen que representa el control de dirección arriba	No se mueva el personaje hacia arriba	No sucedió	Pass
T58	Se comprobará que no se puede mover al personaje hacia abajo	Toque en la imagen que representa el control de dirección abajo	No se mueva el personaje hacia abajo	No sucedió	Pass
T59	Se comprobará que no se muestra la interfaz de over relay de nivel cuando se soluciona el nivel	Solucionar el nivel	No se muestre la interfaz over de nivel	No sucedió	Pass
T60	Se comprobará que mientras el personaje se desplaza no puede cambiar de dirección ni pausar el juego	Presionar las flechas de dirección o pausa	Se cambie la dirección de desplazamiento del personaje o se pause el juego	No sucedió	Pass
T61	Se probará que desaparece el mensaje en pantalla cuando se presiona las flechas de dirección	Toque en cualquiera de las flechas de dirección	Desaparezca el mensaje en pantalla	No sucedió	Pass

Continuación (Tabla 47: <i>Caso de prueba de la interfaz 10 primeros niveles (clase inválida)</i>)					
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T62	Se probará que se no muestra el over relay de pausa	Toque en la imagen que representa la pausa	No se muestre la interfaz de over relay de pausa	No sucedió	Pass
T62.1	Se probará que no se puede reanudar el juego desde el over relay de pausa	Toque en botón Continuar del over relay de pausa	No se continúe con el juego	No sucedió	Pass
T62.2	Se probará que no se puede salir del nivel y regresar al menú de niveles desde el over relay de pausa	Toque en el botón Menú de niveles del over relay de pausa	No se regrese al menú de niveles	No sucedió	Pass
T63	Se probará que no se reinicia el juego desde la interfaz de nivel	Toque en la imagen que representa reiniciar el juego	No se reinicie el juego	No sucedió	Pass

Fuente: elaboración propia

Caso de prueba 7

Tabla 48: *Caso de prueba de la interfaz de segundo tutorial (clase inválida)*

Escenario probado	INTERFAZ DE SEGUNDO TUTORIAL				
Fecha:	19/11/2015	Avance %	100%		
Tester:	Iván Marcial Palomino Taype				
Descripción de la Prueba:	Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de segundo tutorial del videojuego, el cual fue detallado en los requisitos funcionales.				
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T64	Se comprobará que el videojuego no muestre la interfaz del nivel 11	Toque en el botón OK	No se muestre la interfaz del nivel 11	No sucedió	Pass

Fuente: elaboración propia

Caso de prueba 8

Tabla 49: Caso de prueba de la interfaz 10 últimos niveles (clase inválida)

Escenario probado		INTERFAZ DE NIVEL (10 últimos)			
Fecha:		20/11/2015 - 23/11/2015	Avance %	100%	
Tester:		Iván Marcial Palomino Taype			
Descripción de la Prueba:		Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de los 10 últimos niveles del videojuego, lo cual fue detallado en los requisitos funcionales.			
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T65	Se comprobará que se puede mover al personaje cuando se presionan las flechas de dirección, pausa o reiniciar	Toque en las flechas de dirección, pausa o reiniciar	No se pueda mover al personaje	Sucedió	Pass
T66	Se comprobará que desaparece el mensaje en pantalla y el tiempo empieza a correr cuando se presiona las flechas de control de dirección, pausa o reiniciar	Toque en la imagen que representa las flechas de dirección, pausa o reiniciar	No desaparezca el mensaje de consejo en pantalla y el tiempo empiece a correr	Sucedió	Pass
T67	Se comprobará que no se puede mover al personaje a la derecha	Toque en la imagen que representa el control de dirección derecha	No se mueva el personaje hacia la derecha	No sucedió	Pass
T68	Se comprobará que no se puede mover al personaje a la izquierda	Toque en la imagen que representa el control de dirección izquierda	No se mueva el personaje hacia la izquierda	No sucedió	Pass
T69	Se comprobará que no se puede mover al personaje hacia arriba	Toque en la imagen que representa el control de dirección arriba	No se mueva el personaje hacia arriba	No sucedió	Pass

Continuación (Tabla 49: *Caso de prueba de la interfaz 10 últimos niveles (clase inválida)*)

Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T70	Se comprobará que no se puede mover al personaje hacia abajo	Toque en la imagen que representa el control de dirección abajo	No se mueva el personaje hacia abajo	No sucedió	Pass
T71	Se comprobará que el primer personaje en llegar a la meta aún pueda ser movido	Llevar el primer personaje a su meta	Aún se mueva al primer personaje que llego a su meta	No sucedió	Pass
T72	Se comprobará que no se muestra la interfaz de over relay de nivel cuando se soluciona el nivel	Llevar el segundo personaje a su meta	No se muestre la interfaz over de nivel	No sucedió	Pass
T73	Se comprobará que mientras el personaje se desplaza puede cambiar de dirección o pausar el juego	Presionar las flechas de dirección o pausa	Se cambie la dirección de desplazamiento del personaje o se pause el juego	No sucedió	Pass
T74	Se probará que desaparece el mensaje en pantalla cuando se presiona las flechas de dirección	Toque en cualquiera de las flechas de dirección	Desaparezca el mensaje en pantalla	No sucedió	Pass
T75	Se probará que no se muestra el over de pausa	Toque en la imagen que representa la pausa	No se muestre la interfaz de over de pausa	No sucedió	Pass
T75.1	Se probará que no se puede reanudar el juego desde el over relay de pausa	Toque en botón Continuar del over relay de pausa	No se continúe con el juego	No sucedió	Pass

Continuación (Tabla 49: Caso de prueba de la interfaz 10 últimos niveles (clase inválida))

Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T75.2	Se probará que no se puede salir del nivel y regresar al menú de niveles desde el over relay de pausa	Toque en el botón Menú de niveles del over relay de pausa	No se regrese al menú de niveles	No sucedió	Pass
T76	Se probará que no se reinicia el juego desde la interfaz de nivel	Toque en la imagen que representa reiniciar el juego	No se reinicie el juego	No sucedió	Pass

Fuente: elaboración propia

Caso de prueba 9

Tabla 50: Caso de prueba de la interfaz over relay de nivel (clase inválida)

Escenario probado	INTERFAZ DE OVER DE NIVEL				
Fecha:	24/11/2015			Avance %	100%
Tester:	Iván Marcial Palomino Taype				
Descripción de la Prueba:	Esta prueba se realizó para verificar el correcto funcionamiento en la interfaz de over relay de nivel del videojuego, el cual fue detallado en los requisitos funcionales.				
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T77	Se comprobará que no se guardan los datos del nivel resuelto, ni se muestra el puntaje representado con estrellas y tampoco se desbloquea el siguiente nivel	Resolver el nivel	No se guarden los datos del nivel resuelto, ni se muestre el puntaje representado con estrellas y tampoco se desbloquee el siguiente nivel	No sucedió	Pass
T78	Se comprobará que no se puede pasar al siguiente nivel	Toque en la imagen que representa siguiente nivel	No se muestre el siguiente nivel	No sucedió	Pass
T79	Se probará que no se reinicia el juego desde la interfaz de over relay de nivel	Toque en la imagen que representa reiniciar el juego	No se reinicie el juego	No sucedió	Pass

Continuación (Tabla 50: <i>Caso de prueba de la interfaz over relay de nivel (clase inválida)</i>)					
Test ID	Descripción	Pasos	Resultado Esperado	Resultado actual	Status
T80	Se probará que no se puede salir del nivel y regresar al menú de niveles	Toque en el botón menú de niveles	No se regrese al menú de niveles	No sucedió	Pass

Fuente: elaboración propia

C. Corrección del videojuego

En esta actividad se realizaron las tareas de solucionar los errores detectados de acuerdo a su prioridad e impacto en el videojuego, el flujo de estas tareas se muestra a continuación:

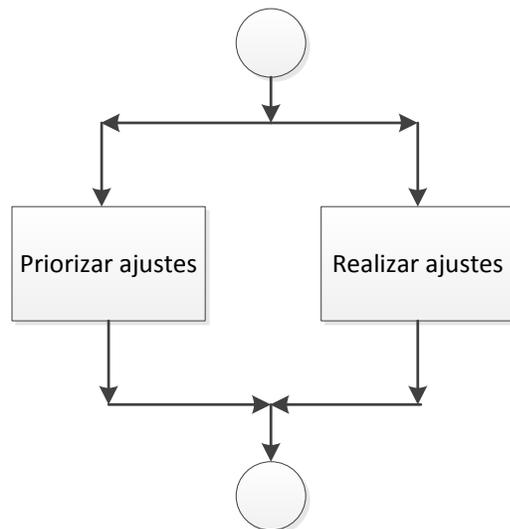


Figura 55. Flujo de las tareas de corrección del videojuego

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de estas tareas se muestra a continuación:

Priorización de ajustes, realización de ajustes

Tabla 51: *Lista de errores detectados y solucionados*

Nro.	Error	Prioridad (Alta, media, baja)	Solucionado
1	La interfaz de primer splash no carga	Alta	Si
2	La interfaz de segundo splash no carga	Alta	Si
3	La interfaz menú principal no carga	Alta	Si
4	La interfaz over relay de menú principal no carga	Media	Si
5	La interfaz de créditos no carga	Baja	Si

Continuación (Tabla 51: <i>Lista de errores detectados y solucionados</i>)			
Nro.	Error	Prioridad (Alta, media, baja)	Solucionado
6	La funcionalidad del botón cerrar de la interfaz de menú principal no se ejecuta	Media	Si
7	La funcionalidad reiniciar el juego de la interfaz over relay de menú no se ejecuta	Media	Si
8	La funcionalidad cambiar idioma de la interfaz over relay de menú no se ejecuta	Media	Si
9	La funcionalidad ajustar volumen no se ejecuta	Media	Si
10	La interfaz de menú de niveles no carga	Alta	Si
11	La funcionalidad del botón regresar del menú de niveles no funciona	Media	Si
12	La funcionalidad de bloqueo de niveles del menú de niveles no se ejecuta	Alta	Si
13	La funcionalidad de graficar los niveles con sus respectivas estrellas que representan puntaje no se ejecuta	Alta	Si
14	La interfaz de nivel no carga	Alta	Si
15	La funcionalidad del botón pausa de la interfaz de nivel no se ejecuta	Media	Si
16	La funcionalidad del botón reiniciar de la interfaz de nivel no se ejecuta	Media	Si
17	La funcionalidad del botón de control especial para habilitar el desplazamiento del personaje no se ejecuta	Alta	Si
18	La funcionalidad de los controles de dirección no se ejecuta	Alta	Si
19	La funcionalidad de los contadores de tiempo y movimiento no se ejecuta	Alta	Si
20	La funcionalidad del guardado de datos cuando se resuelve el nivel no se ejecuta	Alta	Si
21	La interfaz de over relay de nivel no carga	Alta	Si
22	La gráfica de estrellas del over de relay de nivel y del menú de niveles no coincide	Alta	Si
23	La interfaz de primer tutorial no carga	Alta	Si
24	La interfaz de segundo tutorial no carga	Alta	Si
25	La funcionalidad regresar al menú de niveles de la interfaz de over relay de pausa no se realiza	Media	Si
26	La funcionalidad regresar al menú de niveles de la interfaz de over relay de nivel no se realiza	Media	Si
27	El audio de fondo no carga	Media	Si

Fuente: elaboración propia

3.2.6. Cierre

La fase de cierre incluyó tres tareas a realizar, la evaluación del proyecto, liberación del videojuego y el despliegue, dicho despliegue se basa en la metodología RUP, la finalidad de esta fase fue terminar con todo el proyecto.

A. Evaluación del proyecto

En esta tarea se registró los acontecimientos más importantes y significativos, tales como las lecciones aprendidas, algunas recomendaciones así como dificultades y logros realizados, el flujo de trabajo de esta tarea se muestra a continuación:



Figura 56. Flujo de la tarea evaluación del proyecto

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de esta tarea se muestra a continuación:

Lecciones aprendidas, dificultades, logros y recomendaciones

- Hacer un cronograma para la realización del proyecto no siempre podrá ser cumplido como se estableció debido a muchos factores externos e internos.
- Hacer un presupuesto es solo una referencia de los gastos que se necesitarán, puesto que factores como el cambio de la moneda harán que este presupuesto varíe, también factores como requerir nuevas herramientas con la finalidad de lograr los objetivos planteados en el proyecto.
- Desarrollar un videojuego realmente requiere de especialistas para cada área, especialmente para el área de arte gráfico y sonoro.
- El proceso de programación presenta más errores de los que se calcula, es por ello que debe ser realizado tomándose el tiempo necesario para realizarlo de la manera correcta.
- Realizar pruebas de funcionamiento del videojuego en cada versión que se obtiene es vital para así evitar errores que serían muy perjudiciales en el futuro.
- Es muy importante realizar pruebas desde el inicio del desarrollo del videojuego en dispositivos reales donde funcionará el videojuego, para así realizar los ajustes necesarios mucho antes de tener la versión final del videojuego.

- Se recomienda enormemente aprender el idioma inglés a nivel intermedio como mínimo, puesto que la mayoría de la referencia y manuales de programación están en inglés.
- La distribución para el sistema operativo Android no presenta muchos inconvenientes a la hora de realizarse.
- La distribución para el sistema operativo iOS presenta muchos inconvenientes a la hora de realizarse, especialmente para poder instalar el videojuego en un iPhone, para lo cual se requiere realizar antes una extensa configuración y actualización de diversos programas.
- Luego de realizar la optimización del consumo de recursos del videojuego se llegó a la conclusión de que lo que más consume los recursos del teléfono inteligente es el audio, en segundo lugar las imágenes, y por último el código fuente escrito.
- Con la finalidad de lograr un mejor desarrollo del proyecto se recomienda combinar la metodología SUM con alguna otra (GUP y RUP para este caso) para complementar dicho desarrollo y hacerlo más entendible y más fácil de realizar.

B. Liberación del videojuego

En esta tarea se terminó con la versión final del videojuego. El flujo de esta tarea se muestra a continuación:



Figura 57. Flujo de la tarea liberación del videojuego

Fuente: <http://www.gemserk.com/sum/>

Los instaladores del videojuego para los sistemas operativos Android e iOS se pueden encontrar en el **anexo nro. 2 “INSTALADORES DEL VIDEOJUEGO” (digital)**.

C. Despliegue del videojuego

En esta tarea se realizó el diagrama de despliegue que incluyó definir el soporte físico que usa el videojuego para su correcto funcionamiento en teléfonos

inteligentes con sistema operativo Android e iOS. Por razones de dar un mejor entendimiento se hizo referencia al artefacto Modelo de despliegue de RUP. El flujo esta tarea se muestra a continuación:

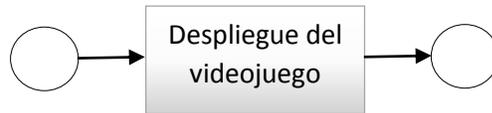


Figura 58. Flujo de la tarea despliegue del videojuego

Fuente: <http://www.gemserk.com/sum/>

El desarrollo de esta tarea se muestra a continuación:

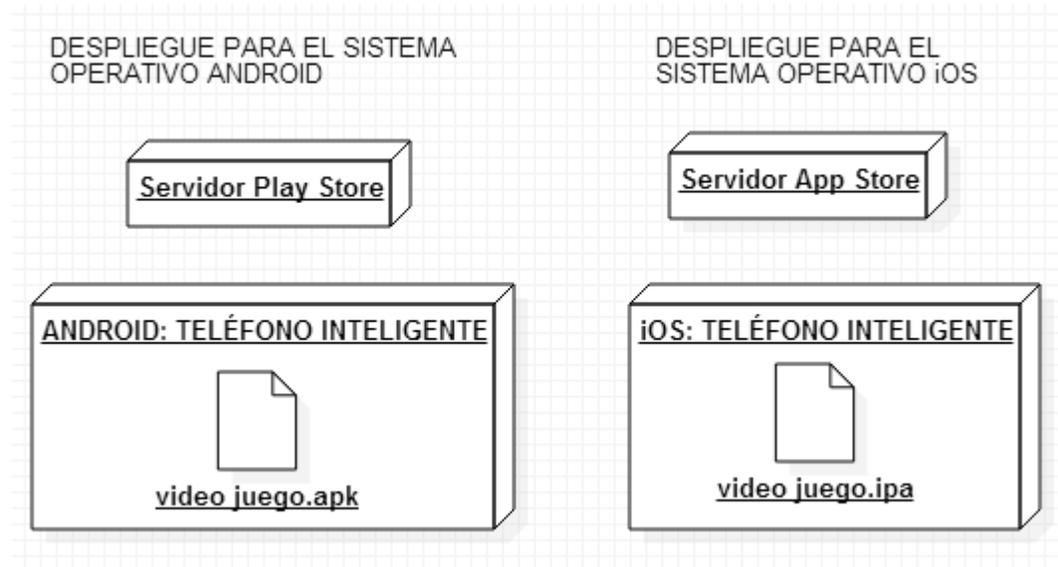


Figura 59: Modelo del despliegue del videojuego

Fuente: elaboración propia

3.3. HERRAMIENTAS

Para el desarrollo del videojuego se usaron herramientas software de distribución gratuita así como de paga; mientras que las herramientas hardware fueron adquiridas necesariamente por medio de pago.

3.3.1. Software

Las herramientas software utilizado se listan y detallan a continuación:

- Corona SDK v2014.2511: motor con el que se desarrolló el videojuego.
- Lenguaje Lua: lenguaje de programación que se usó.

- Audacity v 2.1: para la edición de sonidos.
- CorelDRAW x7: para la creación y edición de imágenes.
- Xilisoft Video Converter v 7.8.12: para la edición de audio y video.
- Compress PNG: para optimizar y reducir el peso de los archivos gráficos.
- Wondershare Filmora Video Editor v 3.1.1: para editar los videos promocionales del videojuego.
- VirtualBox v 4.3.26: para instalar el sistema operativo Mac OS 10.10.1 Yosemite.
- MP3 Quality Modifier v 2.53: para optimizar y reducir el peso de los archivos de audio.
- Sublime Text v 2: como editor para escribir el código fuente del videojuego.
- Microsoft Office 2013: para la realización del documento de proyecto e informe de tesis, cronograma, presupuesto y gráficos.
- Java v 1.8.45: para compilar el videojuego para el sistema operativo Android.
- xCode v 6.4: para compilar el videojuego para el sistema operativo iOS.
- App Monitor Performance Tool v 2.1.4: para ver la cantidad de recursos que consume el videojuego.
- Tiled Map Editor: para diseñar cada uno de los 20 niveles del videojuego.

3.3.2. Hardware

Las herramientas hardware necesarias para el desarrollo del videojuego fueron:

- Computadora de escritorio o personal con las siguientes especificaciones:
 - Sistema Operativo Windows 8.1 Pro.
 - Disco Duro: 1 TB.
 - Memoria RAM: 6GB.
 - Procesador: Intel Core i3 de 2.5GHz.
 - Tarjeta Gráfica: Intel HD Graphics 4000 de 1792 MB.
- Teléfono inteligente de gama baja con sistema operativo Android v 4.1.
- Teléfono inteligente de gama baja con sistema operativo iOS v 6.1.

3.4. GESTIÓN DE LA CALIDAD DEL VIDEOJUEGO

Se aplicaron las características (categorías) y subcaracterísticas (métricas) de la división ISO/IEC 25010, las cuales se detallan en las siguientes tablas:

Categoría 1: Adecuación funcional

Tabla 52: Medición de la métrica “Complejidad funcional”

Nombre:	Complejidad de implementación funcional
Propósito:	Qué tan completa está la implementación funcional del videojuego
Método de aplicación:	Contar las funciones faltantes detectadas en la evaluación y comparar con el número de funciones descritas en la especificación de requisitos
Fórmula:	$X = 1 - A/B$ A = número de funciones faltantes B = número de funciones descritas en la especificación de requisitos
Medición:	A = 0 B = 42 $X = 1 - 0/42 = 1$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, más completa la implementación funcional
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad / cantidad A = cantidad B = cantidad
Fuente de medición:	Especificación de requisitos Diseño de la interfaces, paquetes Código fuente
Audiencia:	Diseñador Artista gráfico Artista sonoro Desarrollador

Fuente: elaboración propia

Tabla 53: Medición de la métrica “Corrección funcional”

Nombre:	Capacidad de obtención de resultados acordes
Propósito:	Qué cantidad de resultados acordes se obtiene
Método de aplicación:	Contar la cantidad de pruebas de caja negra que no producen resultados acordes y la totalidad de pruebas de caja negra realizadas
Fórmula:	$X = 1 - A/B$ A = número de pruebas no exitosas B = número de total de pruebas de caja negra
Medición:	A = 0 B = 80 $X = 1 - 0/80 = 1$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, más capacidad de obtener resultados acordes a la especificación de requisitos
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad / cantidad A = cantidad B = cantidad

Continuación (Tabla 53: <i>Medición de la métrica "Corrección funcional"</i>)	
Nombre:	Capacidad de obtención de resultados acordes
Fuente de medición:	Especificación de requisitos Diseño Código fuente Informe de verificación beta
Audiencia:	Diseñador Artista gráfico Artista sonoro Desarrollador Verificador Beta

Fuente: elaboración propia

Tabla 54: *Medición de la métrica "Pertinencia funcional"*

Nombre:	Cumplimiento de objetivos especificados
Propósito:	Que cantidad de funciones son apropiadas para cumplir los objetivos especificados por el usuario
Método de aplicación:	Contar la cantidad de funciones que no cumplen los objetivos especificados por el usuario y comparar con la cantidad de requisitos especificados
Fórmula:	$X = 1 - A/B$ A = número de funciones que no cumplen con los objetivos especificados por el usuario B = número total de funciones descritas en la especificación de requisitos
Medición:	A = 0 B = 42 $X = 1 - 0/42 = 1$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor cumplimiento de los objetivos especificados
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad/ cantidad A = cantidad B = cantidad
Fuente de medición:	Especificación de requisitos Informe de verificación beta
Audiencia:	Diseñador Verificador Beta

Fuente: elaboración propia

Categoría 2: Eficiencia de desempeño

Tabla 55: *Medición de la métrica "Comportamiento temporal"*

Nombre:	Determinación de los tiempos de respuesta
Propósito:	Cuáles son los tiempos de respuesta del videojuego en condiciones normales
Método de aplicación:	Contar el tiempo de respuesta del videojuego ante las acciones del usuario
Fórmula:	A = tiempo requerido para mostrar la primera interfaz del videojuego B = tiempo requerido para cerrar el videojuego C = tiempo requerido para cambiar de una interfaz a otra

Continuación (Tabla 55: <i>Medición de la métrica "Comportamiento temporal"</i>)	
Nombre:	Determinación de los tiempos de respuesta
Medición:	A = 1.6 B = 0.8 C = 1.2
Interpretación:	Entre más corto, mejor.
Tipo de escala:	Proporción
Tipo de medida:	A = tiempo en segundos B = tiempo en segundos C = tiempo en segundos
Fuente de medición:	Código fuente
Audiencia:	Desarrollador

Fuente: elaboración propia

Tabla 56: *Medición de la métrica "Utilización de recursos"*

Nombre:	Determinación de los recursos usados
Propósito:	Cuáles son los recursos usados por el videojuego en un teléfono inteligente de gama baja
Método de aplicación:	Medir la cantidad de recursos usados por el videojuego
Fórmula:	A = consumo de batería B = uso de memoria RAM C = uso de CPU D = uso de espacio de almacenamiento
Medición:	A = 438 B = 24 C = 18 D = 13
Interpretación:	Mientras sea menor, mejor.
Tipo de escala:	Proporción
Tipo de medida:	A = energía en mW (mili Watts) B = memoria en MB (Mega Bytes) C = porcentaje CPU mono núcleo de 1 GHz D = memoria en MB
Fuente de medición:	Informe de Tesis (sección Resultados)
Audiencia:	Artista gráfico Artista sonoro Desarrollador

Fuente: elaboración propia

Tabla 57: *Medición de la métrica "Capacidad"*

Nombre:	Grado de cumplimiento de los recursos consumidos
Propósito:	Verificar que los recursos máximos usados sean menores a los disponibles en un teléfono inteligente de gama baja
Método de aplicación:	Medir la cantidad de recursos usados por el videojuego con respecto a otros videojuegos y al teléfono inteligente de gama baja
Fórmula:	A = consumo de batería B = uso de memoria RAM C = uso de CPU D = uso de espacio de almacenamiento

Continuación (Tabla 57: Medición de la métrica "Capacidad")			
Nombre:	Grado de cumplimiento de los recursos consumidos		
Medición:	Videojuego desarrollado	Promedio otros videojuegos	Capacidad máxima teléfono inteligente
A =	438	591.8	7700
B =	24	39	512
C =	18	55.4	100
D =	13	26.8	4000
Interpretación:	El videojuego desarrollado consume menor cantidad de recursos tal y como se solicitó en los requisitos, esto dentro de un teléfono inteligente de gama baja		
Tipo de escala:	Proporción		
Tipo de medida:	A = energía en mW (mili Watts) B = memoria en MB (Mega Bytes) C = porcentaje CPU mono núcleo de 1 GHz D = memoria en MB		
Fuente de medición:	Informe de Tesis (sección Resultados y Marco teórico)		
Audiencia:	Artista gráfico Artista sonoro Desarrollador		

Fuente: elaboración propia

Categoría 3: Compatibilidad

Tabla 58: Medición de la métrica "Coexistencia"

Nombre:	Determinación de coexistencia con otros videojuegos	
Propósito:	Verificar que el videojuego desarrollado pueda ejecutarse junto con otros videojuegos, dentro del mismo teléfono inteligente	
Método de aplicación:	Medir la cantidad de recursos usados por el videojuego desarrollado y por otro videojuego que compite por los mismos recursos	
Fórmula:	A = consumo de batería B = uso de memoria RAM C = uso de CPU D = uso de espacio de almacenamiento	
Medición:	Videojuego desarrollado	Videojuego Bejeweled 2
A =	438	518
B =	24	53
C =	18	36
D =	13	40
Interpretación:	El videojuego desarrollado puede ejecutarse de manera normal incluso cuando el videojuego Bejeweled 2 se encuentra ejecutándose en segundo plano haciendo uso de los recursos del teléfono inteligente	
Tipo de escala:	Proporción	
Tipo de medida:	A = energía en mW (mili Watts) B = memoria en MB (Mega Bytes) C = porcentaje CPU mono núcleo de 1 GHz D = memoria en MB	
Fuente de medición:	Informe de Tesis (sección Resultados)	

Continuación (Tabla 58: <i>Medición de la métrica “Coexistencia”</i>)	
Nombre:	Determinación de coexistencia con otros videojuegos
Audiencia:	Artista gráfico Artista sonoro Desarrollador

Fuente: elaboración propia

Categoría 4: Usabilidad

Tabla 59: *Medición de la métrica “Capacidad de aprendizaje”*

Nombre:	Facilidad para aprender a jugar el videojuego
Propósito:	Verificar si es fácil aprender a jugar el videojuego
Método de aplicación:	Comparar la cantidad de usuarios que mencionan que no es fácil de aprender y la cantidad de usuarios total
Fórmula:	$X = 1 - A/B$ A = número de usuarios que se les dificultó aprender a jugar B = número total de usuarios.
Medición:	A = 2 B = 25 $X = 1 - 2/25 = 0.92$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, más fácil de aprender a jugar
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad / cantidad A = cantidad B = cantidad
Fuente de medición:	Sección opinión sobre el videojuego del anexo “Fichas de medición de reducción de estrés llenadas”
Audiencia:	Diseñador Artista gráfico Artista sonoro Desarrollador

Fuente: elaboración propia

Tabla 60: *Medición de la métrica “Capacidad para ser usado”*

Nombre:	Facilidad para usar el videojuego
Propósito:	Verificar si es fácil usar el videojuego (si presta la interfaz adecuada para jugar, ello no implica que el videojuego sea fácil)
Método de aplicación:	Comparar la cantidad de usuarios que mencionan que no es fácil de usar el videojuego y la cantidad de usuarios total
Fórmula:	$X = 1 - A/B$ A = número de usuarios que se les dificultó usar el videojuego B = número total de usuarios
Medición:	A = 2 B = 25 $X = 1 - 2/25 = 0.92$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, más fácil de usar el videojuego
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad / cantidad A = cantidad B = cantidad

Continuación (Tabla 60: <i>Medición de la métrica "Capacidad para ser usado"</i>)	
Nombre:	Facilidad para usar el videojuego
Fuente de medición:	Sección opinión sobre el videojuego del anexo "Fichas de medición de reducción de estrés llenadas"
Audiencia:	Diseñador Artista gráfico Artista sonoro Desarrollador

Fuente: elaboración propia

Tabla 61: *Medición de la métrica "Estética de la interfaz de usuario"*

Nombre:	Aceptación de la estética del videojuego
Propósito:	Verificar si la estética de las interfaces del videojuego es agradable para los usuarios
Método de aplicación:	Comparar la cantidad de usuarios que mencionan que no son agradables las interfaces el videojuego y la cantidad de usuarios total
Fórmula:	$X = 1 - A/B$ A = número de usuarios que mencionan que la interfaz del videojuego no es agradable B = número total de usuarios
Medición:	A = 6 B = 25 $X = 1 - 6/25 = 0.76$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor estética e interfaz agradable al usuario
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad / cantidad A = cantidad B = cantidad
Fuente de medición:	Sección opinión sobre el videojuego del anexo "Fichas de medición de reducción de estrés llenadas"
Audiencia:	Diseñador Artista gráfico Artista sonoro Desarrollador

Fuente: elaboración propia

Categoría 5: Fiabilidad

Tabla 62: *Medición de la métrica "Madurez"*

Nombre:	Capacidad de desempeñar las funciones especificadas
Propósito:	Verificar si el videojuego cumple con las funciones especificadas en los requisitos funcionales y no funcionales
Método de aplicación:	Comparar la cantidad de funciones que no se ejecutan correctamente con la cantidad total de funciones especificadas en los requisitos
Fórmula:	$X = 1 - A/B$ A = número de funciones que no se desempeñan como fue especificado B = número total de funciones especificadas en los requisitos

Continuación (Tabla 62: Medición de la métrica "Madurez")	
Nombre:	Capacidad de desempeñar las funciones especificadas
Medición:	A = 0 B = 42 $X = 1 - 0/42 = 1$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor capacidad de desempeñar las funciones especificadas en los requisitos
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad / cantidad A = cantidad B = cantidad
Fuente de medición:	Especificación de requisitos Código fuente Verificación Beta
Audiencia:	Diseñador Desarrollador Verificador Beta

Fuente: elaboración propia

Tabla 63: Medición de la métrica "Disponibilidad"

Nombre:	Capacidad del videojuego de estar disponible y operativo
Propósito:	Verificar que el videojuego esté disponible las 24 horas del día para cualquier usuario de un teléfono con sistema operativo Android o iOS
Método de aplicación:	Comparar la cantidad de horas por día en las que no se puede descargar el videojuego desde las tiendas virtuales Play Store y App Store
Fórmula:	$X = 1 - A/B$ A = número de horas no disponibles para descargar el videojuego B = número total de horas por día
Medición:	A = 0 B = 24 $X = 1 - 0/24 = 1$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor disponibilidad para el usuario
Tipo de escala:	Absoluta
Tipo de medida:	X = tiempo / tiempo A = tiempo B = tiempo
Fuente de medición:	Sitio web donde está subido el videojuego para las tiendas virtuales: Play Store y App Store
Audiencia:	Productor interino

Fuente: elaboración propia

Categoría 6: Seguridad

Tabla 64: Medición de la métrica "Integridad"

Nombre:	Capacidad de prevenir alteración de datos
Propósito:	Verificar que el videojuego guarde íntegramente los datos generados durante la ejecución del videojuego y luego de cerrar y volver a abrir el videojuego

Continuación (Tabla 64: Medición de la métrica "Integridad")	
Nombre:	Capacidad de prevenir alteración de datos
Método de aplicación:	Comparar la cantidad de niveles del videojuego que una vez resueltos no almacenan ni mantienen el puntaje obtenido luego de cerrar y volver a abrir el videojuego
Fórmula:	$X = 1 - A/B$ A = número niveles que no almacenan ni mantienen el puntaje que se obtiene en dicho nivel B = número total de niveles del videojuego
Medición:	A = 0 B = 20 $X = 1 - 0/20 = 1$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor capacidad de prevenir la alteración de datos
Tipo de escala:	Absoluta
Tipo de medida:	$X = \text{cantidad} / \text{cantidad}$ A = cantidad B = cantidad
Fuente de medición:	Código fuente Verificación Beta
Audiencia:	Desarrollador Verificador Beta

Fuente: elaboración propia

Categoría 7: Mantenibilidad

Tabla 65: Medición de la métrica "Modularidad"

Nombre:	Capacidad de realizar cambios con impacto mínimo
Propósito:	Verificar que el videojuego pueda ser modificado (cada interfaz que lo compone) y que dicha modificación tenga un impacto mínimo o nulo para las demás interfaces
Método de aplicación:	Comparar la cantidad de interfaces que no pueden ser modificadas sin implicar impactos medianos o grandes para las demás interfaces
Fórmula:	$X = 1 - A/B$ A = número de interfaces que no pueden ser modificadas sin implicar impacto a otras interfaces B = número total de interfaces que componen en videojuego
Medición:	A = 1 B = 11 $X = 1 - 1/11 = 0.91$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor capacidad de realizar cambios sin impacto crítico
Tipo de escala:	Absoluta
Tipo de medida:	$X = \text{cantidad} / \text{cantidad}$ A = cantidad B = cantidad
Fuente de medición:	Código fuente
Audiencia:	Desarrollador

Fuente: elaboración propia

Tabla 66: *Medición de la métrica "Reusabilidad"*

Nombre:	Capacidad de reutilizar módulos
Propósito:	Verificar si los módulos (interfaces, ya que contienen código fuente) pueden ser reutilizados total o parcialmente
Método de aplicación:	Comparar la cantidad de interfaces que no pueden ser reutilizadas total ni parcialmente para crear otras interfaces
Fórmula:	$X = 1 - A/B$ A = número de interfaces que no pueden ser reutilizadas total ni parcialmente B = número total de interfaces que componen en videojuego
Medición:	A = 1 B = 11 $X = 1 - 1/11 = 0.91$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor capacidad de reutilizar módulos (interfaces)
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad / cantidad A = cantidad B = cantidad
Fuente de medición:	Diseño Código fuente
Audiencia:	Artista gráfico Artista sonoro Desarrollador

Fuente: elaboración propia

Tabla 67: *Medición de la métrica "Capacidad para ser modificado"*

Nombre:	Capacidad de realizar cambios de manera fácil
Propósito:	Verificar si los módulos (interfaces, ya que contienen código fuente) pueden ser modificados de manera fácil, efectiva y eficiente.
Método de aplicación:	Comparar la cantidad de interfaces que no pueden ser modificadas de manera fácil, efectiva o eficiente.
Fórmula:	$X = 1 - A/B$ A = número de interfaces que no pueden modificadas de manera fácil, efectiva o eficiente B = número total de interfaces que componen en videojuego
Medición:	A = 2 B = 11 $X = 1 - 2/11 = 0.82$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor capacidad de modificar módulos de manera fácil, efectiva y eficiente.
Tipo de escala:	Absoluta
Tipo de medida:	X = cantidad / cantidad A = cantidad B = cantidad
Fuente de medición:	Diseño Código fuente
Audiencia:	Artista gráfico Desarrollador

Fuente: elaboración propia

Categoría 8: Portabilidad

Tabla 68: *Medición de la métrica "Adaptabilidad"*

Nombre:	Capacidad para adaptarse a otros entornos
Propósito:	Verificar si el código fuente del videojuego puede ser compilado para otras plataformas (sistemas operativos móviles)
Método de aplicación:	Comparar la cantidad de sistemas operativos a los que no puede ser desplegado el código fuente del videojuego y la cantidad total de sistemas operativos móviles con mayor demanda al 2015
Fórmula:	$X = 1 - A/B$ A = número de sistemas operativos móviles a los que no puede ser desplegado el código fuente del videojuego B = número total de sistemas operativos móviles
Medición:	A = 3 B = 6 $X = 1 - 3/6 = 0.5$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor capacidad de desplegar el código fuente del videojuego a otros sistemas operativos móviles
Tipo de escala:	Absoluta
Tipo de medida:	$X = \text{cantidad} / \text{cantidad}$ A = cantidad B = cantidad
Fuente de medición:	Código fuente Tabla de sistemas operativos (sección Marco teórico)
Audiencia:	Artista sonoro Desarrollador

Fuente: elaboración propia

Tabla 69: *Medición de la métrica "Capacidad para ser instalado"*

Nombre:	Capacidad para instalar y desinstalar el videojuego
Propósito:	Verificar la facilidad de la instalación y desinstalación del ejecutable del videojuego (.apk, .ipa) en teléfonos inteligentes con sistema operativo Android e iOS
Método de aplicación:	Instalar y desinstalar el videojuego 10 veces, jugando 2 niveles antes de desinstalar
Fórmula:	$X = 1 - A/B$ A = número de veces que no se pudo instalar o desinstalar el videojuego B = número total de instalaciones y desinstalaciones
Medición:	A = 0 B = 40 $X = 1 - 0/40 = 1$
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mayor facilidad de instalar y desinstalar el videojuego
Tipo de escala:	Absoluta
Tipo de medida:	$X = \text{cantidad} / \text{cantidad}$ A = cantidad B = cantidad
Fuente de medición:	Instalador .apk, .ipa para Android e iOS
Audiencia:	Desarrollador

Fuente: elaboración propia

3.5. DESARROLLO CON OPTIMIZACIÓN EN EL CONSUMO DE RECURSOS

Para lograr que el videojuego consuma la menor cantidad de recursos posibles: Memoria RAM, Procesador, espacio de instalación y consumo de batería, se realizó el siguiente proceso de optimización en el consumo de recursos:

3.5.1. Optimización de recursos de audio

Debido a que éste es uno de los componentes del videojuego que consume más recursos se utilizó el programa MP3QualityModifier para hacer que su inclusión en el videojuego no cause un mayor consumo de recursos, dentro de este programa se realizaron las siguientes configuraciones:

- A. Disminución de Bitrate (variable de 65 a 105 kbps).
- B. Cambio de Estéreo a Mono estéreo.
- C. Cambio de frecuencia de muestreo a 22050 Hz.

Esto aplicado a los archivos de audio en formato .mp3 hizo que se redujera considerablemente su tamaño a coste de disminuir la calidad del sonido, pero estas configuraciones garantizaron que no se pierda mucha calidad de sonido. La siguiente tabla detalla esta información:

Tabla 70: *Optimización de recursos de audio*

CARACTERÍSTICAS	ANTES	DESPUÉS
Nombre del archivo de Audio	4gg02.mp3	4gg02.mp3
Velocidad (kbits)	128	40
Modo	Estéreo	Mono estéreo
Frecuencia de muestreo	44100 Hz	22050 Hz
Duración (segundos)	30	30
Tamaño	470 KB	149 KB

Fuente: elaboración propia

3.5.2. Optimización de recursos gráficos

Las pruebas demostraron que este componente del videojuego era quien ocupaba el segundo lugar en cuanto a consumo de recursos, para ello se utilizó el compresor online de imágenes en formato .png llamado Compress PNG, donde se realizaron las siguientes configuraciones:

- A. Reducción de profundidad de bit (de 32 a 8).
- B. Reducción de cantidad de colores (de 256 a 213).

Con estas configuraciones se redujo el tamaño de los archivos. png, sin mermar la calidad gráfica de manera considerable. La siguiente tabla detalla esta información:

Tabla 71: *Optimización de recursos gráficos*

CARACTERÍSTICAS	ANTES	DESPUÉS
Archivo gráfico		
Pixeles	32x32	32x32
Profundidad en bits	32	8
Cantidad de colores	256	213
Tamaño	4.12KB	0.75KB

Fuente: elaboración propia

3.5.3. Optimización de código fuente

Para optimizar el consumo de recursos del código fuente se realizaron las siguientes optimizaciones, consejos, y buenas prácticas de codificación basadas en la documentación oficial del motor de desarrollo de videojuegos Corona SDK:

- A. Se utilizó el lenguaje que usa Corona SDK, es decir el lenguaje Lua, este lenguaje está orientado a la programación por procesos, además está diseñado para ser súper ligero y poderoso.
- B. Se utilizó en gran mayoría variables y funciones locales, las cuales pueden ser administradas por el recolector de basura propio de Lua, liberando así memoria RAM para reducir su consumo.
- C. Se redujo el tamaño de los archivos gráficos y de sonido, reduciendo así el espacio de almacenamiento del videojuego instalado y reduciendo la cantidad necesaria de procesamiento para cargar dichos archivos a la hora de ejecutar el videojuego. También se reutilizó los archivos de sonido y especialmente los gráficos en la medida de lo posible.
- D. Se realizó la carga de archivos gráficos en el momento en que se los necesita, debido a que cargar con anticipación estos archivos antes de que se los necesite afecta seriamente a la memoria RAM que puede ser saturada. Lo contrario se aplicó a los archivos de sonido, estos fueron cargados con anterioridad, más específicamente al inicio del videojuego en las interfaces de segundo splash, menú principal y créditos donde no se requiere de mucha capacidad de procesamiento, así se evitó la sobrecarga del procesador.

- E.** Se realizó la referenciación a nulo (nil en Lua) de los objetos (gráficos) que se muestran en la pantalla del teléfono inteligente, permitiendo así que el Garbage Collector (recolector de basura) propio de Lua, libere la memoria que estos objetos ocupaban al momento de ser creados, de esta manera se optimizó el consumo de memoria RAM.
- F.** No se utilizó conexión de red local ni a servidores, tampoco GPS ni acelerómetro puesto que esto consume enormemente la batería del teléfono inteligente, además la lectura y escritura a la memoria interna del teléfono se realizó usando tablas y no elementos individuales de tablas, puesto que transmitir datos de manera individual y repetidamente consume más batería que transmitirlos en paquetes grandes como las tablas de Lua.
- G.** No se utilizó imágenes sprite debido a que el consumo de procesador que realizan es mucho mayor al de una imagen.
- H.** Se realizó la descarga de memoria de archivos gráficos cuando no fueron más requeridos para la escena actual del videojuego, de esta manera pudieron ser administrados por el recolector de basura optimizando el uso de memoria RAM, sin embargo el gráfico de fondo que fue necesario en todas las escenas no se descargó y se estableció como variable global ya que era necesario, de esta manera se optimizó la carga y descarga para dicho gráfico optimizando el uso del procesador.
- I.** Para la optimización más específica del lenguaje Lua se realizó lo siguiente:
- Se utilizó variables y funciones locales, aunque se utilizó al mínimo las variables globales porque realmente fue necesario, esto permitió que el código fuente se ejecute de manera más rápida y se evite el lag (ralentización gráfica) en tiempo de ejecución crítica.
 - También se utilizó las librerías de Lua de manera local, librerías como la de física y manejo de escenas.
 - Se evitó el uso de la función “table.insert()” para la carga de datos en una tabla, esto debido a que no está optimizada, en lugar de ello se usó la función recomendada por Lua.

- Se evitó el uso de la función “unpack()” para el acceso a datos en una tabla, esto debido a que no está optimizada, en lugar de ello se usó la función recomendada por Lua.
 - Se optimizó el procesamiento matemático de Lua, utilizando consejos como: utilizar multiplicación con decimales en lugar de usar división, esto hace que el procesamiento sea menos pesado y más rápido.
- J.** En la administración de sonido además de la reducción de tamaño realizada, en lo que a codificación se refiere, los sonidos de fondo necesarios para las escenas fueron declarados como variables globales, para evitar cargarlos y descargarlos en cada escena, con esto se redujo el poder de procesamiento necesario para su administración, sin embargo se referencio a nil el canal donde eran reproducidos para evitar sobrecargar los canales disponibles. Solo se usó carga y descarga en cada escena para el sonido que se reproduce cuando se soluciona un nivel, esto porque dicho sonido no es muy extenso y puede ser cargado en tiempo de ejecución crítica, además se referenció a nil el canal donde se reproduce.
- K.** Se utilizó audio en formato .mp3 en lugar de .wav porque luego de realizar pruebas de desempeño se concluyó que el sonido en formato .mp3 era más ligero que el formato .wav, ya que requería menos capacidad de procesamiento, memoria RAM, consumo de batería y además tenía menor peso. Además se usó la función “audio.loadSound()”, en lugar de la función “audio.loadStream()”, debido a que la primera era menos costosa de procesar.
- L.** También en el archivo de configuración “config.lua”, se estableció la variable “audioPlayFrequency = 22050”, porque los archivos de audio fueron transformados a 22050 Hz de frecuencia de muestreo, con esto se redujo la capacidad de procesamiento que se requiere para su reproducción.
- M.** En cuanto a los “Listeners” o registradores de eventos, se usó en su mayoría Listener locales y solo un Listener global con la finalidad de detectar varias colisiones entre los diferentes objetos, además se borró todos los Listener cuando ya no fueron más necesitados.
- N.** En cuanto a la administración de escenas se usó la librería “Storyboard”, cuya documentación estuvo disponible al inicio de la formulación de este proyecto en

el año 2015, sin embargo debido a las actualizaciones de Corona SDK, dicha librería se cambió por otra llamada “Composer”, la cual tiene la misma funcionalidad pero con algunas modificaciones. El uso de las funciones de la librería “Storyboard” permitieron administrar de mejor manera la codificación y las escenas del videojuego, especialmente las funciones de:

- CreateScene: que permitió cargar los archivos gráficos y de sonido y donde también se inicializó variables para evitar bugs o errores, así como la inicialización de la física.
 - EnterScene: donde se asignó los Listener locales y globales para optimizar el rendimiento.
 - ExitScene: donde se borró los Listener locales y globales y se detuvieron los “Timers” o relojes para que estos no sigan consumiendo el CPU, también se descargó el sonido que se cargó para cuando se resolvía el nivel y se liberó los canales de reproducción de sonido. Además se detuvo la física.
 - DestroyScene: donde liberó la memoria RAM que los objetos usaban.
 - Adicionalmente se usaron las funciones Over relay: “overlayBegan” y “overlayEnded”, que permitieron realizar acciones necesarias para el videojuego.
- O.** Se limitó el uso de la propiedad “isSleepingAllowed” para los personajes (caritas felices) para que puedan detectar las colisiones en todo momento, con esta limitación se redujo el CPU requerido.
- P.** Se dejó el Frame Rate (Velocidad de Fotogramas) por defecto a 30 en el archivo de configuración, significa que en cada segundo se redibuja la pantalla 30 veces, al no cambiarlo a 60 se redujo el consumo de CPU.

3.6. DISTRIBUCIÓN MULTIPLATAFORMA

3.6.1. Distribución para la plataforma Android

Se generó el instalador (.apk) del videojuego para instalación en teléfonos inteligentes con sistema operativo Android y también para subirlo a su tienda oficial Play Store.

3.6.1.1. Generación de .apk para teléfonos con SO Android

Para generar el instalador para teléfonos inteligentes con sistema operativo Android se realizaron los siguientes pasos:

- A. Se instaló el motor de desarrollo de videojuegos Corona SDK v.2014.2511.
- B. Se instaló el editor de texto Sublime v.2.
- C. Se instaló Java JDK (Java Development Kit) v.1.8.45.
- D. Se conectó a internet y una vez terminado el videojuego se procedió a compilarlo (generar el .apk), para ello se fue al menú "file", luego "Build for Android", la siguiente imagen muestra dicha ruta:

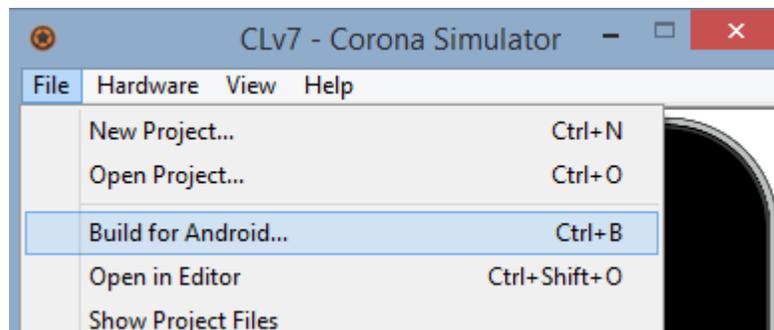


Figura 60: Ruta para compilación para teléfonos Android

Fuente: elaboración propia

- E. Se especificaron los datos como: Nombre de la Aplicación, Versión de Código, Versión del Nombre, Paquete, versión del SDK de Android, el Keystore (en modo debug), el alias del Keystore y la ruta donde se almacenó el instalador, la siguiente imagen muestra estos datos:

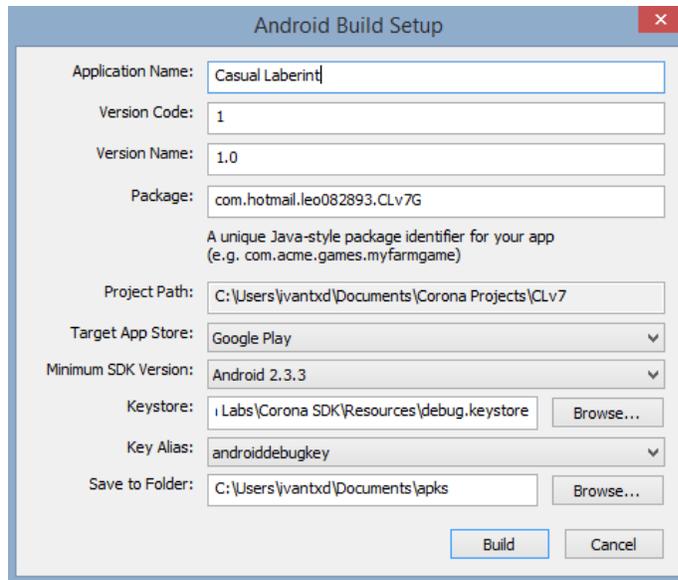


Figura 61: Configuración para compilación para teléfonos para Android

Fuente: elaboración propia

- F.** Para instalar el videojuego al teléfono inteligente con sistema operativo Android se copió el .apk a la memoria interna del teléfono y se procedió a instalarlo normalmente.

3.6.1.2. Generación de .apk para Play Store

Para generar el instalador para subirlo a la Play Store se realizaron los siguientes pasos:

- A.** Se instaló el motor de desarrollo de videojuegos Corona SDK v.2014.2511.
- B.** Se instaló el editor de texto Sublime v.2.
- C.** Se instaló Java JDK (Java Development Kit) v.1.8.45.
- D.** Se generó el Keystore (en modo release) que permitió firmar la aplicación para subirla a la Play Store, para ello fue necesario:
 - Abrir la ruta de JDK de Java instalado y ejecutar el programa Keytool.exe
 - Poner la siguiente línea de comandos: “keytool -genkey -v -keystore PGRelease.keystore -alias aliasPG -keyalg RSA -keysize 2048 -validity 10000”
 - Luego completar los datos solicitados: “pass keystore, first name, organization until, organization, city, state, two letter” y luego escribir “s”

- Volver a poner el mismo pass keystore y listo.

- E. Se conectó a internet y una vez terminado el videojuego se procedió a compilarlo para la Play Store, para ello se fue al menú “file”, luego “Build for Android”, la siguiente imagen muestra dicha ruta:

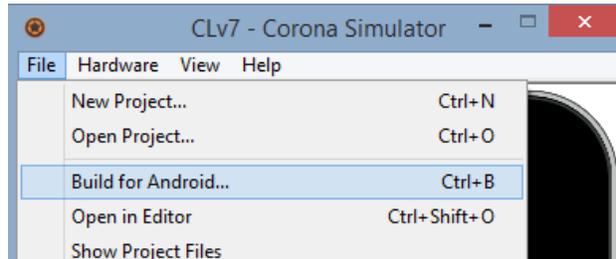


Figura 62: Ruta para compilación para Play Store

Fuente: elaboración propia

- F. Se especificaron los datos como: Nombre de la Aplicación, Versión de Código, Versión del Nombre, Paquete, versión del SDK de Android, el Keystore (en modo release), el alias del Keystore, que ahora deben ser seleccionados de la ruta del programa Keytool porque fue allí donde se generó, finalmente y la ruta donde se almacenó el instalador, la siguiente imagen muestra estos datos:

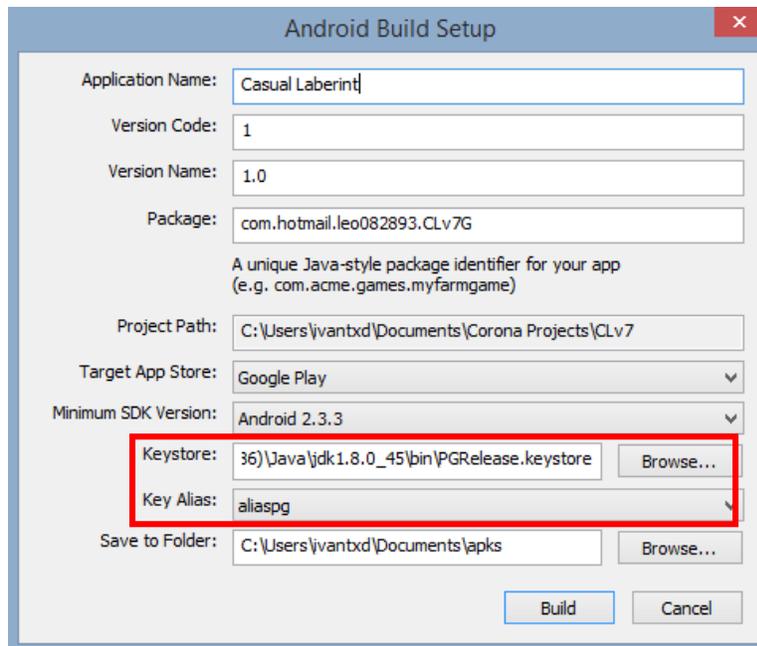


Figura 63: Configuración para compilar para Play Store

Fuente: elaboración propia

3.6.1.3. Subir a la Play Store

Para subir el videojuego a la tienda de google se realizaron los siguientes pasos:

- A. Se accedió a la consola de desarrolladores de Google Play, en <https://play.google.com/apps/publish/>
- B. Se realizó el logueo con una cuenta de Gmail.
- C. Se realizó el pago de 25 dólares con tarjeta.
- D. Una vez realizado el pago volver se volvió a acceder a <https://play.google.com/apps/publish/> , y se inició con el proceso para subir la aplicación:
- E. Se dio click en “Añadir nueva Aplicación” y se puso el nombre de la aplicación:

AÑADIR NUEVA APLICACIÓN

Idioma predeterminado *

Español (Latinoamérica) – es-419 ▼

Nombre *

Casual Laberint 1.0|

19 de 30 caracteres

¿Cómo te gustaría empezar?

Subir APK Preparar ficha de Play Store Cancelar

Figura 64: Subir videojuego a Play Store

Fuente: elaboración propia

- F. Luego se completó los requisitos que pide Google para que se apruebe el videojuego y pueda ser publicado en la Play Store.

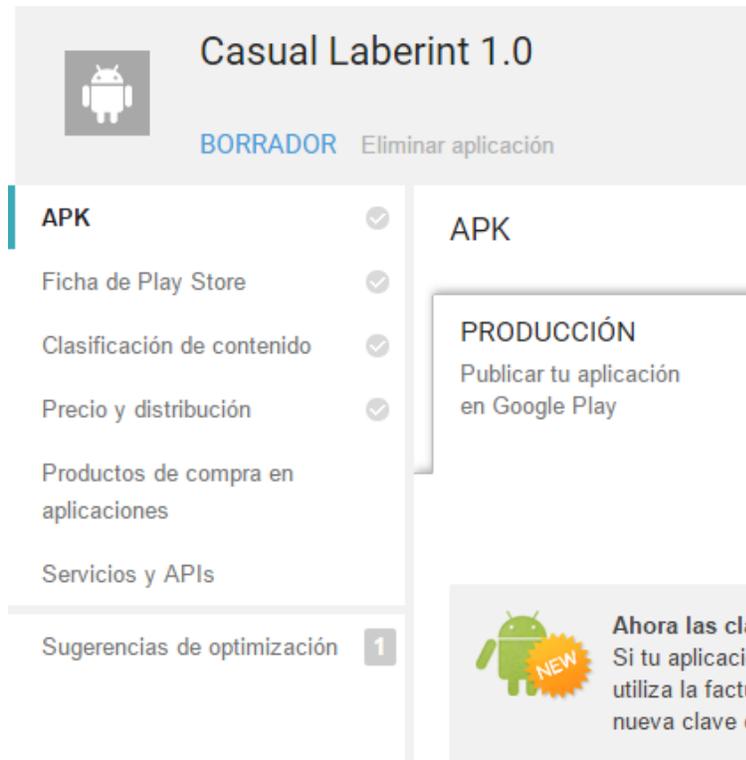


Figura 65: Requisitos para subir el videojuego a Play Store

Fuente: elaboración propia

- G. En la sección “APK”, se debe subió el .apk generado con la keystore (modo release) y se seleccionó los teléfonos inteligentes que son compatibles con la versión del videojuego, en este caso el videojuego es compatible para los sistemas operativos Android desde 2.3.3 en adelante.
- H. En la sección “FICHA DEL PLAY STORE”, se puso información del videojuego, el ícono, las imágenes y el video promocional que previamente fue subido a YouTube, también se puso los datos de contacto y la política de privacidad.
- I. En la sección “CLASIFICACIÓN DE CONTENIDO”, se realizó un cuestionario que verificó e indicó que el contenido del videojuego era apto para todas las edades.
- J. Y en la sección “PRECIO Y DISTRIBUCIÓN”, se indicó que el videojuego es gratuito, además se seleccionó los países en los que se va a distribuir, finalmente se aceptaron los contratos de política de servicio.

- K. Se esperó a que Google Play revise el videojuego que estuvo conforme a las reglas y fue publicado luego de 24 horas.

3.6.2. Distribución para la plataforma iOS

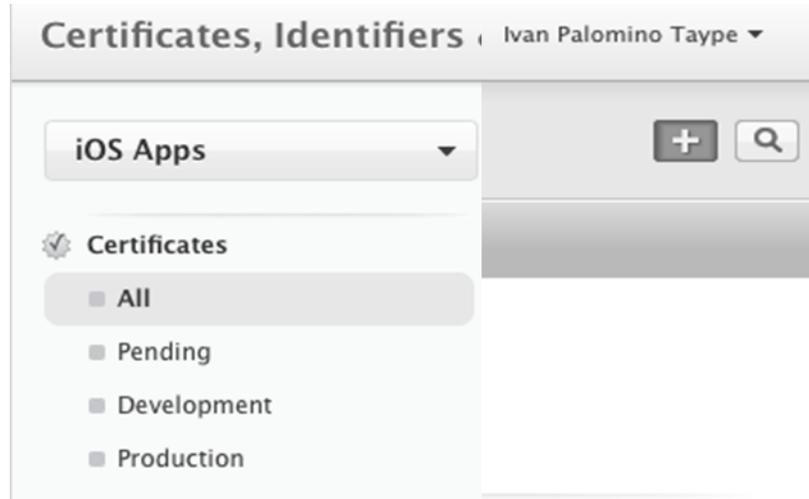
Se generó el instalador (.app) del videojuego para instalación en teléfonos inteligentes con sistema operativo iOS y también para subirlo a su tienda oficial App Store. Cabe mencionar que para ello es necesario utilizar una computadora Mac OS X, pero debido a motivos económicos no pudo ser adquirida, y en su reemplazo se utilizó una máquina virtual con sistema operativo Mac OS X Yosemite 10.10.1 de 64 bits que fue virtualizada usando el VirtualBox v.4.3.26 sobre una computadora Lenovo con sistema operativo Windows 8.1 Pro de 64 bits.

3.6.2.1. Generación del .app para la teléfonos iPhone

Para generar el instalador para teléfonos inteligentes con sistema operativo iOS se realizaron los siguientes pasos:

- A. Se creó una cuenta de Apple, en su página web <https://appleid.apple.com/> , la cual fue necesaria para instalar el xCode desde el iTunes para poder compilar para teléfonos con sistema operativo iOS.
- B. Una vez se tuvo el ID de Apple se procedió a instalar el xCode desde el iTunes, se usó la versión 6.4.
- C. Lo siguiente fue crear una cuenta de desarrollador de Apple desde su página web <https://developer.apple.com/> , para ello primero se pagó 99 dólares con tarjeta y se esperó en promedio dos días hasta que se active la cuenta.
- D. Con la cuenta de desarrollador de Apple activada se procedió a crear el Perfil de aprovisionamiento, que contiene a su vez el certificado con el que se firmó el .app generado, el ID de aplicación (que es único para cada aplicación) y también el ID de dispositivo en el que se instaló el videojuego, este proceso se detalla a continuación:

- En primer lugar se creó el certificado con el que se firmó el videojuego, para ello en la sección de “Certificates” se dio click al botón “+” y luego se dio click en “Worldwide Developer Relations Certificate Authority”, se descargó la solicitud del certificado y se instaló dando click en el archivo descargado.



 [Worldwide Developer Relations Certificate Authority](#)

Figura 66: Creación de certificado para iOS

Fuente: elaboración propia

- Después se accedió al “Acceso a Llaveros” del Mac y se fue a “Asistente para Certificados” y se dio click en “Solicitar un certificado de una autoridad de certificación”, y se pusieron los datos de la cuenta de desarrollador de Apple, después se guardó en el escritorio el archivo de solicitud generado.



Figura 67: Solicitar un certificado a una autoridad de certificación

Fuente: elaboración propia

- Luego de vuelta en el sitio de desarrolladores de Apple en la sección de “Producción”, se seleccionó “App Store and Ad Hoc” y se dio click en “Continue”, ahí se cargó el archivo de solicitud generado anteriormente y se dio click a “Generate”, para luego descargar el certificado con el que se firmó el videojuego.

Production

App Store and Ad Hoc

Sign your iOS app for submission to the App Store or for Ad Hoc distribution.

Apple Push Notification service SSL (Sandbox & Production)

Establish connectivity between your notification server, the Apple Push Notification sandbox, and production environments to deliver remote notifications to you utilizing HTTP/2, the same certificate can be used to deliver app notifications complication data, and alert background VoIP apps of incoming activity. A sep is required for each app you distribute.

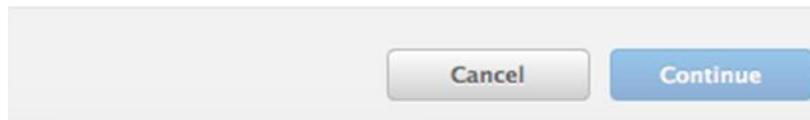


Figura 68: Generación del certificado para iOS

Fuente: elaboración propia

- En segundo lugar se creó el identificador de aplicación, para ello se dio click en la sección “Identifiers” y luego click en el botón “+”.

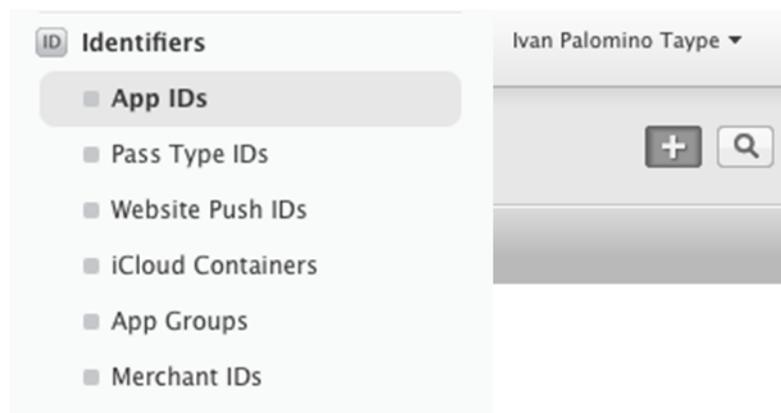


Figura 69: Identificador de aplicación para iOS

Fuente: elaboración propia

- Después se puso el nombre del identificador de aplicaciones (se puso CasualL), y el Bundle ID, se usó (com.perugames.casual), luego se dio click en continuar y ya se tuvo registrado el identificador de aplicación el cual es único para cada aplicación.

App ID Description

Name:
You cannot use special characters such as @, &, *, ', "

App ID Suffix

• Explicit App ID

If you plan to incorporate app services such as Game Center, In-App Purchase, Data Protection, and iCloud, or want a provisioning profile unique to a single app, you must register an explicit App ID for your app.

To create an explicit App ID, enter a unique string in the Bundle ID field. This string should match the Bundle ID of your app.

Bundle ID:

We recommend using a reverse-domain name style string (i.e., com.domainname.appname). It cannot contain an asterisk (*).

Cancel

Continue

Figura 70: Configuración del ID de aplicación para iOS

Fuente: elaboración propia

- En tercer lugar se registró el teléfono iPhone donde se instaló el videojuego, para ello en la sección “Devices” se dio click en el botón “+”.

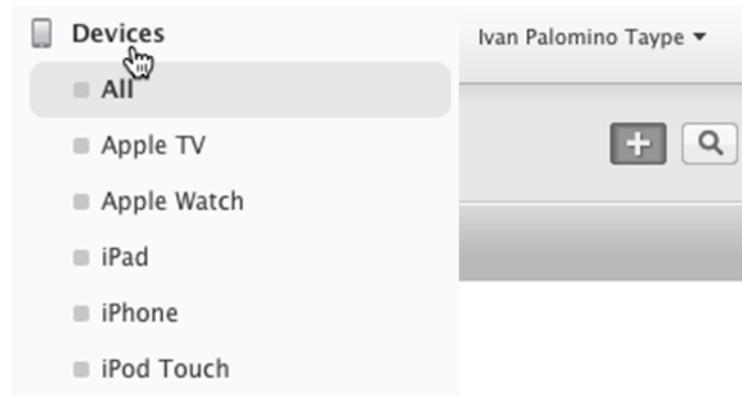


Figura 71: Dispositivo (iPhone) donde se instala el videojuego

Fuente: elaboración propia

- Luego se puso el nombre del dispositivo (se puso Casuald para este caso), y se pone el UDID que es el identificador que tienen los teléfonos iPhone, este es único por cada teléfono iPhone, luego se dio click en “Continue” y ya se tuvo registrado el iPhone donde se instaló el videojuego.

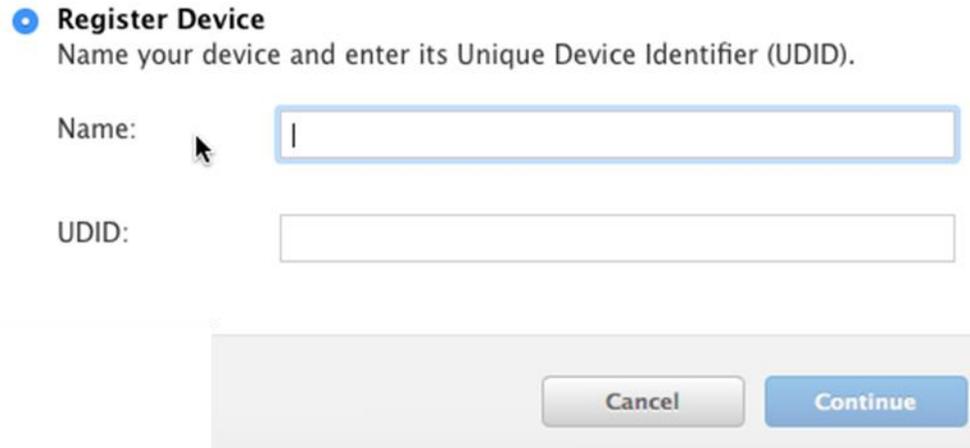


Figura 72: Configuración del dispositivo (iPhone) donde se instala el videojuego

Fuente: elaboración propia

- Como paso final se generó el “Perfil de aprovisionamiento” para instalar el videojuego al teléfono iPhone, para ello en la sección “Provisioning Profiles” se dio click en el botón “+”

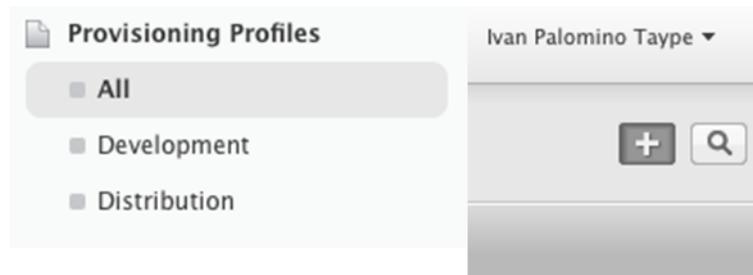


Figura 73: Perfil de aprovisionamiento para compilar para dispositivos iPhone

Fuente: elaboración propia

- Después se seleccionó “Ad Hoc” para generar el Perfil de Aprovisionamiento para generar el .app que se instaló en el teléfono iPhone, y se dio click en “Continue”.

Distribution

- App Store**
Create a distribution provisioning profile to submit your app to the App Store.
- Apple TV App Store**
Create a distribution provisioning profile to submit your tvOS app to the Apple TV App Store.
- Ad Hoc**
Create a distribution provisioning profile to install your app on devices.
- tvOS Ad Hoc**
Create a distribution provisioning profile to install your app on tvOS devices.



Figura 74: Selección del tipo de certificado para el Perfil de aprovisionamiento para dispositivos iPhone

Fuente: elaboración propia

- Después se seleccionó el ID de aplicación que fue creado, y se dio click en “Continue”.



Figura 75: Selección del ID de aplicación para el Perfil de aprovisionamiento para dispositivos iPhone

Fuente: elaboración propia

- Luego se seleccionó el certificado que fue generado para firmar el videojuego, y se dio click en “Continue”.

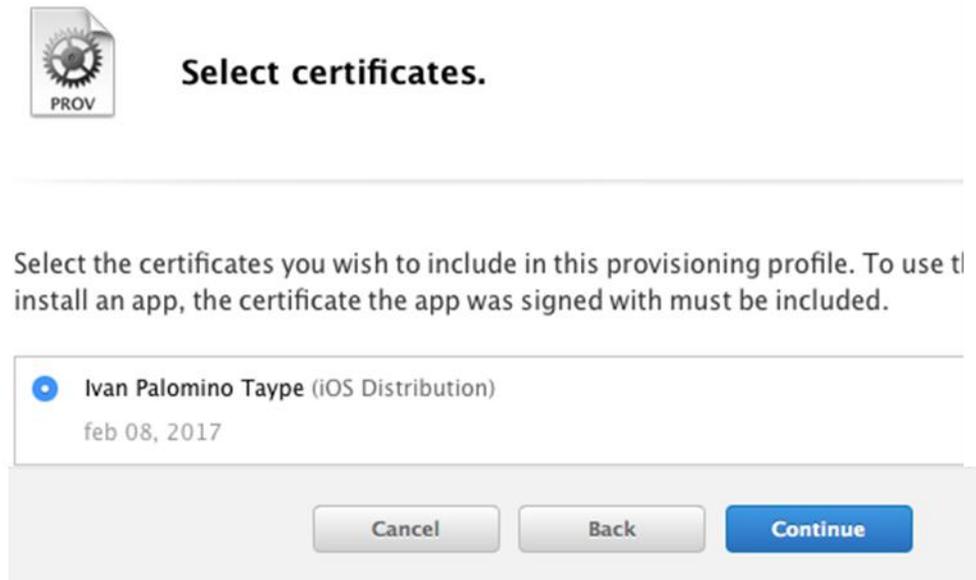


Figura 76: Selección del certificado para el Perfil de aprovisionamiento para compilar para dispositivos iPhone

Fuente: elaboración propia

- Después se seleccionó el dispositivo registrado donde se instaló el videojuego y se dio click en “Continue”.

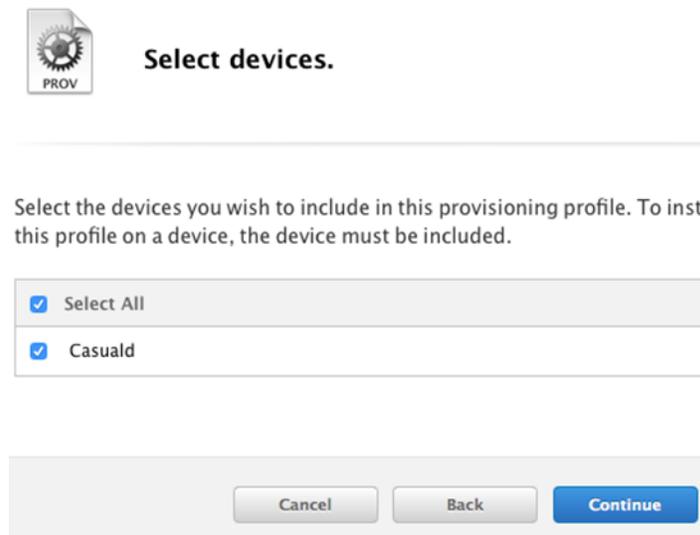


Figura 77: Selección del dispositivo iPhone donde se instala el videojuego

Fuente: elaboración propia

- Finalmente se puso el nombre del Perfil de Aprovisionamiento (Casual Ad Hoc para este caso) y luego se dio click en “Generatate” y por último se descargó el Perfil de Aprovisionamiento y se instaló en el “Acceso a Llaveros”

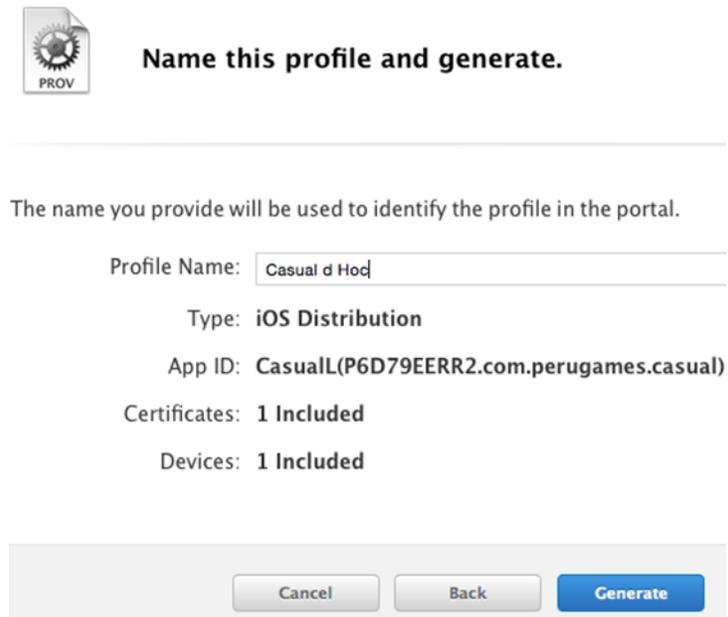


Figura 78: Generación del Perfil de aprovisionamiento para compilar para dispositivos iPhone

Fuente: elaboración propia

- E. Una vez que se tuvo el Perfil de Aprovisionamiento se procedió a generar el .app para instalar el videojuego en el teléfono inteligente iPhone registrado, para ello se instaló el Corona SDK v.2015.2545 que tiene soporte para 64 bits, después se fue a la ruta “File”, “Build”, “iOS”.

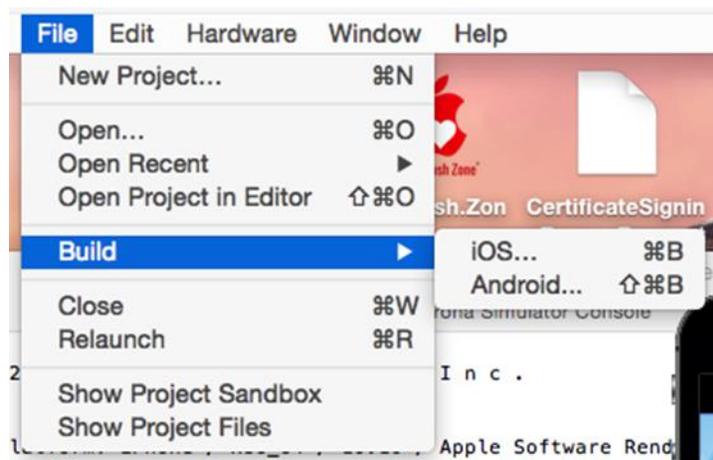


Figura 79: Ruta para compilar para dispositivos iOS

Fuente: elaboración propia

- F. Después se puso el nombre del Videojuego, la versión y se seleccionó el Perfil de aprovisionamiento con el nombre “Casual Ad Hoc” y se puso la ruta donde se generó al .app.

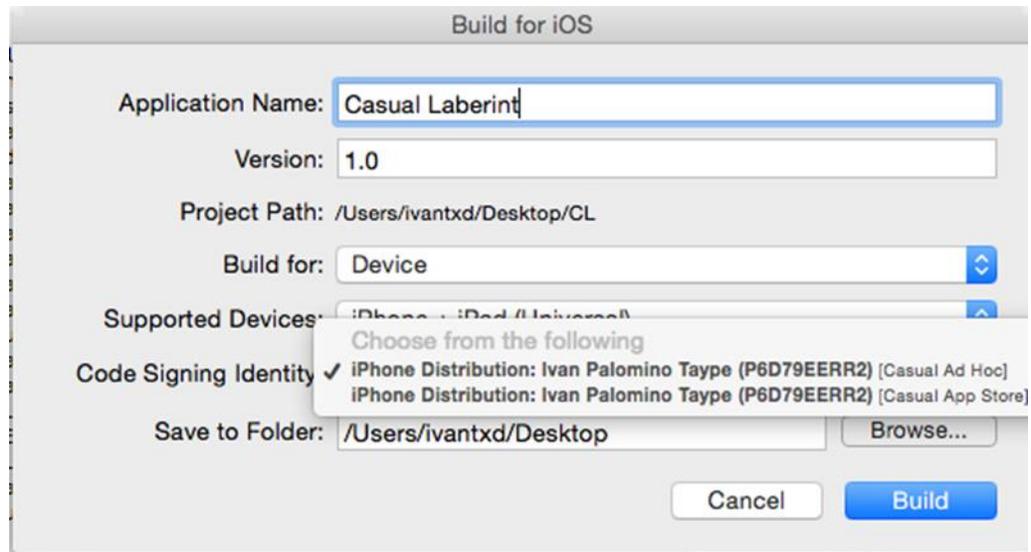


Figura 80: Configuración para compilar para dispositivos iOS

Fuente: elaboración propia

- G. Finalmente para instalar el videojuego en el teléfono iPhone se creó una carpeta con el nombre “Payload”, se puso el .app dentro de esa carpeta, luego se comprimió dicha carpeta en un archivo .zip con el nombre “CL” y se cambió la extensión del .zip por .ipa obteniéndose el instalador “CL.ipa” el cual se instaló al iPhone por medio del programa iFunBox.



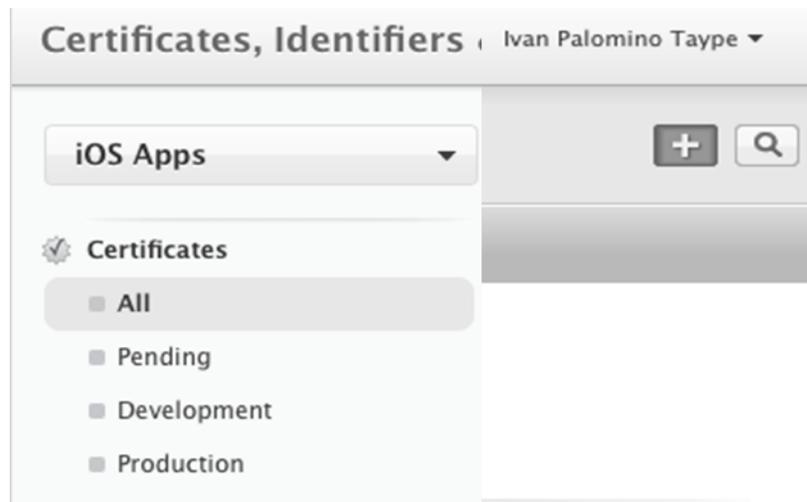
Figura 81: Instalación del .ipa en iPhone 3GS

Fuente: elaboración propia

3.6.2.2. Generación del .zip para la App Store

Para generar el instalador para subir a la App Store, se realizaron los siguientes pasos:

- A. Se creó una cuenta de Apple, en su página web <https://appleid.apple.com/> , la cual fue necesaria para instalar el xCode desde el iTunes para poder compilar para teléfonos inteligentes con sistema operativo iOS.
- B. Una vez se tuvo el ID de Apple se procedió a instalar el xCode desde el iTunes, (se usó la versión 6.4 para este caso).
- C. Lo siguiente fue crear una cuenta de desarrollador de Apple desde su página web <https://developer.apple.com/> , para ello primero se pagó 99 dólares con tarjeta y se esperó en promedio dos días hasta que se active la cuenta.
- D. Con la cuenta de desarrollador de Apple activada se procedió a crear el Perfil de aprovisionamiento para subir a la App Store, que contiene a su vez el certificado con el que se firmó el .app generado y el ID de aplicación (que es único para cada aplicación), este proceso se detalla a continuación:
 - En primer lugar se creó el certificado con el que se firmó el videojuego, para ello en la sección de “Certificates” se dio click al botón “+” y luego se dio click en “Worldwide Developer Relations Certificate Authority”, se descargó la solicitud del certificado y se instaló dando click en el archivo descargado.

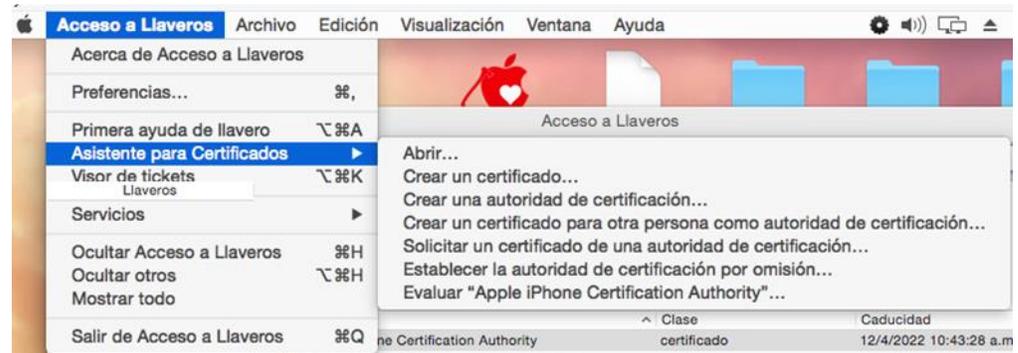


 [Worldwide Developer Relations Certificate Authority](#)

Nota: es la misma figura que la “Figura 66: Creación de certificado para iOS”

- Después se accedió al “Acceso a Llaveros” del Mac, se fue a “Asistente para Certificados” y se dio click en “Solicitar un certificado de una autoridad de

certificación”, después se pusieron los datos de la cuenta de desarrollador de Apple, y se guardó en el escritorio el archivo de solicitud generado.



Nota: es la misma figura que la “Figura 67: Solicitar un certificado a una autoridad de certificación”

- Luego de vuelta en el sitio de desarrolladores de Apple en la sección de “Producción”, se seleccionó “App Store and Ad Hoc” y se dio click en continuar, ahí se cargó el archivo de solicitud generado anteriormente y se dio click a “Generate”, para luego descargar el certificado con el que se firmó el videojuego.

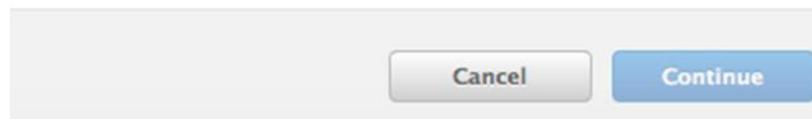
Production

App Store and Ad Hoc

Sign your iOS app for submission to the App Store or for Ad Hoc distribution.

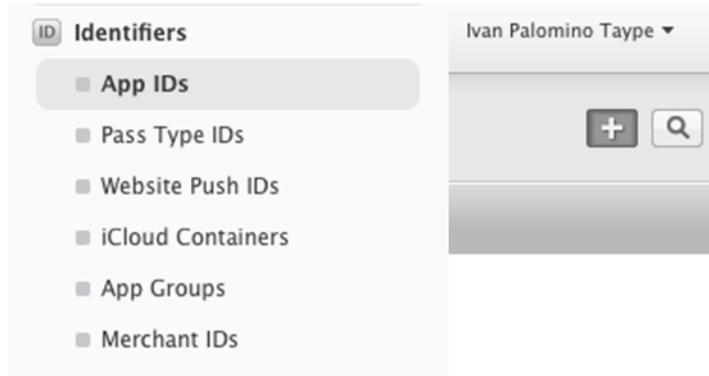
Apple Push Notification service SSL (Sandbox & Production)

Establish connectivity between your notification server, the Apple Push Notification service sandbox, and production environments to deliver remote notifications to you utilizing HTTP/2, the same certificate can be used to deliver app notifications, app complication data, and alert background VoIP apps of incoming activity. A separate certificate is required for each app you distribute.



Nota: es la misma figura que la “Figura 68: Generación del certificado para iOS”

- En segundo lugar se creó el identificador de aplicación, para ello se dio click en la sección “Identifiers” y luego click en el botón “+”.



Nota: es la misma que la “Figura 69: Identificador de aplicación para iOS”

- Después se puso el nombre del identificador de aplicaciones (se puso Casual L), y el Bundle ID, se usó (com.perugames.casual), luego se dio click en “Continuar” y ya se tuvo registrado el identificador de aplicación el cual es único para cada aplicación.

App ID Description

Name:

You cannot use special characters such as @, &, *, ', "

App ID Suffix

• Explicit App ID

If you plan to incorporate app services such as Game Center, In-App Purchase, Data Protection, and iCloud, or want a provisioning profile unique to a single app, you must register an explicit App ID for your app.

To create an explicit App ID, enter a unique string in the Bundle ID field. This string should match the Bundle ID of your app.

Bundle ID:

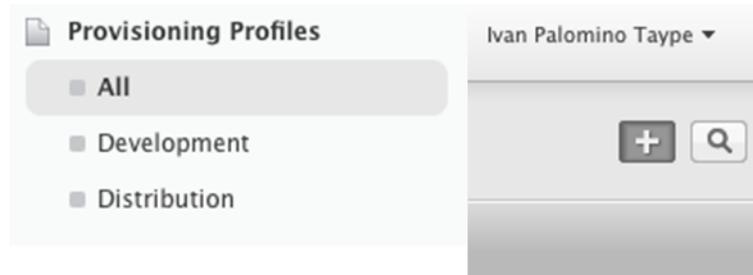
We recommend using a reverse-domain name style string (i.e., com.domainname.appname). It cannot contain an asterisk (*).

Cancel

Continue

Nota: es la misma que la “Figura 70: Configuración del ID de aplicación para iOS”

- Como paso final se generó el “Perfil de aprovisionamiento” para subir el videojuego a la App Store, para ello en la sección “Provisioning Profiles” se dio click en el botón “+”



Nota: es la misma figura que la “Figura 73: Perfil de aprovisionamiento para compilar para dispositivos iPhone”

- Después se seleccionó “App Store” para generar el Perfil de Aprovisionamiento para generar el .app que se subió a la App Store, y se dio click en “Continue”.

Distribution

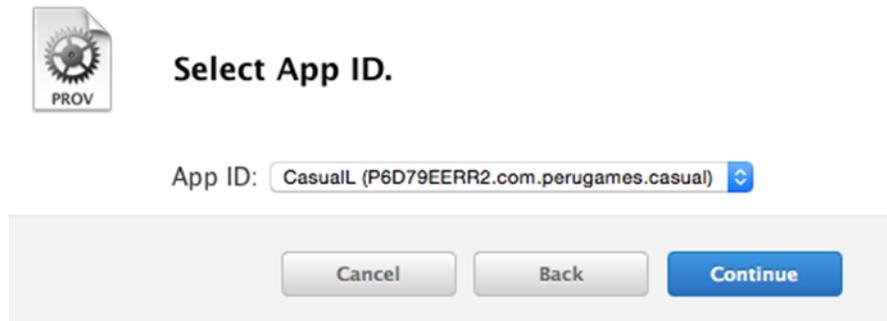
- App Store**
Create a distribution provisioning profile to submit your app t
- Apple TV App Store**
Create a distribution provisioning profile to submit your tvOS
- Ad Hoc**
Create a distribution provisioning profile to install your app on devices.
- tvOS Ad Hoc**
Create a distribution provisioning profile to install your app on tvOS devices.



Figura 82: Selección del tipo de certificado para el Perfil de aprovisionamiento para subir a App Store

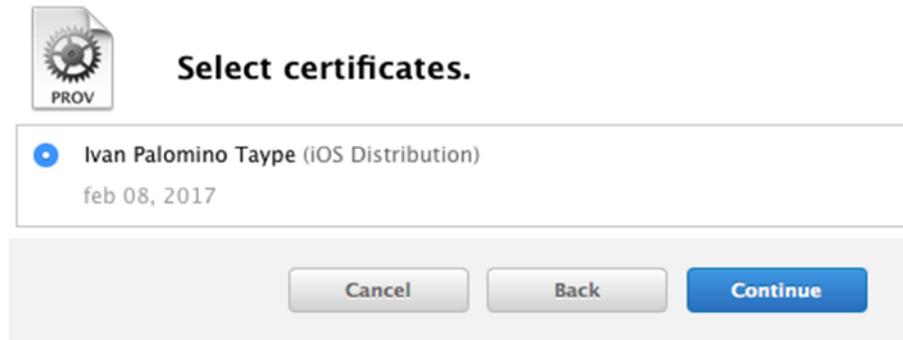
Fuente: elaboración propia

- Después se seleccionó el ID de aplicación que fue creado, y se dio click en “Continue”.



Nota: es la misma figura que la “Figura 75: Selección del ID de aplicación para el Perfil de aprovisionamiento para dispositivos iPhone”

- Luego se seleccionó el certificado que fue generado para firmar el videojuego y se dio click en “Continue”.



Nota: es la misma figura que la “Figura 76: Selección del certificado para el Perfil de aprovisionamiento para compilar para dispositivos iPhone”

- Finalmente se puso el nombre del Perfil de Aprovisionamiento (Casual App Store para este caso) y luego se dio click en “Generatate” y por último se descargó el Perfil de Aprovisionamiento y se instaló en el “Acceso a Llaveros”

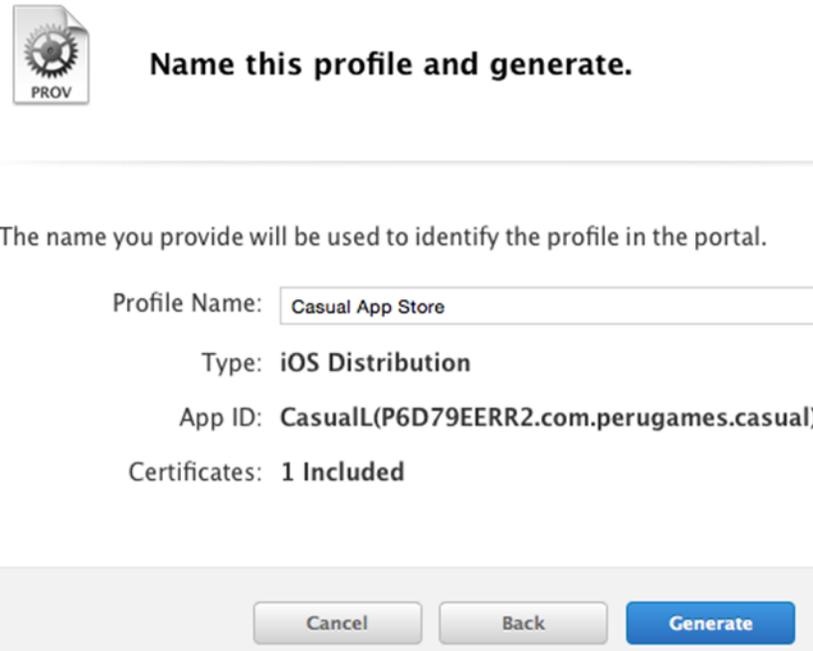


Figura 83: Generación del Perfil de aprovisionamiento para subir a App Store

Fuente: elaboración propia

- E. Una vez que se tuvo el Perfil de Aprovisionamiento se procedió a generar el .app para subir el videojuego a la App Store, para ello se instaló el Corona SDK v.2015.2545 que tiene soporte para 64 bits, después se fue a la ruta “File”, “Build”, “iOS”.

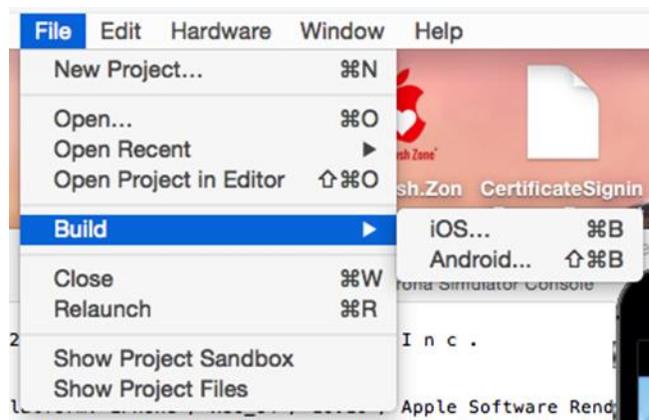


Figura 84: Ruta para compilar para subir a App Store

Fuente: elaboración propia

- F. Después se puso el nombre del Videojuego, la versión y se seleccionó el Perfil de aprovisionamiento con el nombre “Casual App Store” y se puso la ruta donde se generó al .app y el .zip.

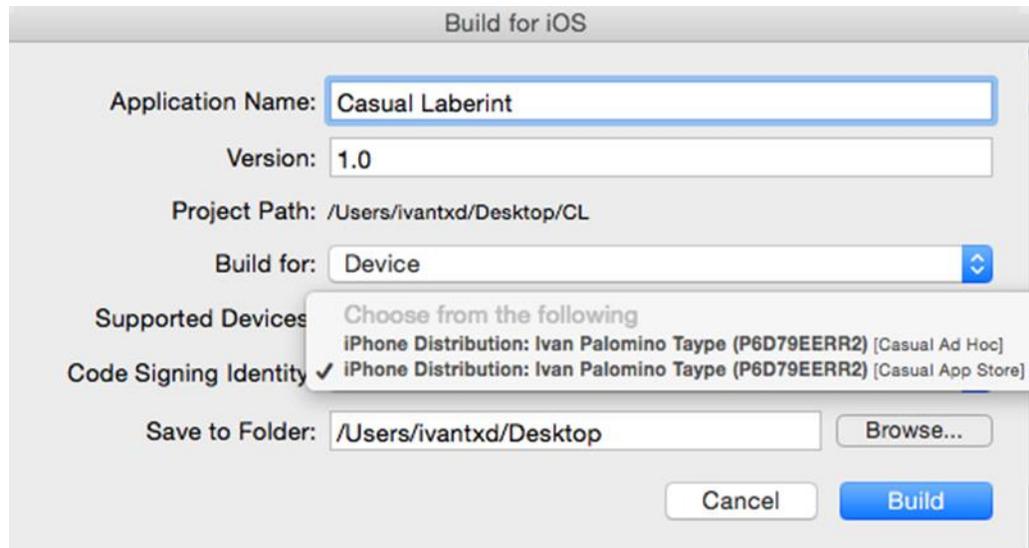


Figura 85: Configuración para compilar para subir a la App Store
Fuente: elaboración propia

3.6.2.3. Subir a la App Store

Para subir el videojuego a la App Store se realizaron los siguientes pasos:

- A. Se accedió al iTunes Connect en <https://itunesconnect.apple.com/>, y en la sección de “Mis apps” se dio click en “+” y luego en “Nueva app” y después se puso el nombre de la aplicación, en este caso “Casual Labyrinth”, además se puso el idioma, el Bundle ID y la versión.

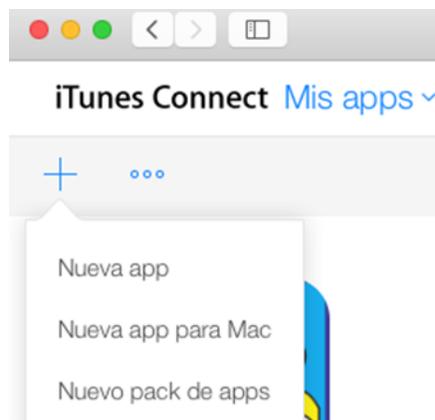


Figura 86: Subir videojuego a App Store
Fuente: elaboración propia

- B. Luego se abrió el Application Loader, se dio click en “Entregar aplicación” y se subió el .zip generado con el Perfil de aprovisionamiento “Casual App Store”.



Figura 87: Enviar .zip del videojuego a la App Store

Fuente: elaboración propia

- C. De regreso en iTunes Connect se procedió a llenar toda la información del videojuego en la secciones “Información de la app”, “Precio y disponibilidad” y “Preparar para la presentación”.

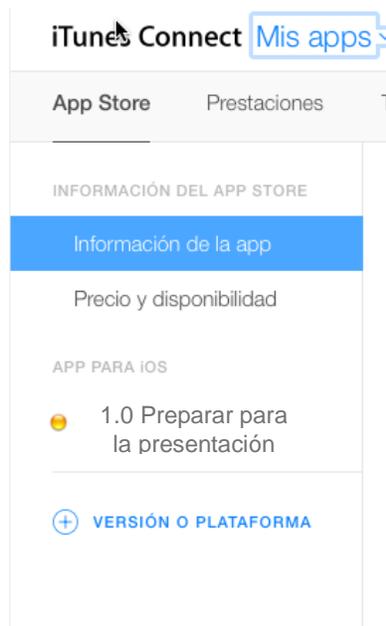


Figura 88: Secciones a configurar para subir videojuego a App Store

Fuente: elaboración propia

- D. En la sección de “Información de la App”, se completó la información referente a la categoría del videojuego, y la clasificación de contenido.

Información que se puede traducir

 **Nombre** ?

Casual Labyrinth

Información general

ID de pack ?

com.perugames.casual

Idioma principal ?

Español (España)

SKU ?

C001

 **Categoría** ?

Juegos

Puzzle

Subcategoría (opcional)

Entretenimiento

Contrato de licencia [Editar](#)

[Contrato de licencia estándar de Apple](#)

Clasificación ?

4+ años [Clasificaciones adicionales](#)

Figura 89: Configuración de la sección Información de la app

Fuente: elaboración propia

- E. En la sección de “Precio y disponibilidad”, se estableció el precio del videojuego, en este caso fue gratis, también se especificó la disponibilidad territorial, es decir desde que países podrá ser descargado el videojuego (en este caso disponible en todos los países); también se especificó la disponibilidad temporal, es decir hasta que fecha estará disponible el videojuego (en este caso sin fecha limite).

Planificación de precios [Todos los precios](#)

Precio ?	Fecha de inicio ?	Fecha final ?
USD 0 (Gratis) v	Otras divisas 26 feb. 2016	Sin fecha final

[Planificar un cambio de precio](#)

Disponibilidad

- Disponible en todos los territorios [Editar](#)
- Retirar de la venta

Figura 90: Configuración de la sección Precio y disponibilidad

Fuente: elaboración propia

- F. Y en la sección de “Preparar para la presentación”, se puso las imágenes y el video promocional para los diferentes tamaños de los teléfonos iPhone donde funcionará el videojuego, el ícono, una descripción del videojuego, las palabras clave para que los motores de búsqueda puedan ubicar el videojuego, la url de soporte, la información de contacto del representante comercial y la información de quien revisó el videojuego, la siguiente imagen muestra lo descrito:

Información de la versión

Vídeo de la vista previa de la app y capturas de pantalla de la app ?

4,7 pulgadas 5,5 pulgadas 4 pulgadas 3,5 pulgadas

Descripción ?

Entertaining casual game, where you have to use your head to solve each level. The aim is to bring the smiley face to the flag in order to solve each level and become the champion of Casual Labyrinth.

Technical specifications:
Requires iOS 6.0 or later.
Available for iPhone.

Videojuego casual de entretenimiento, en donde tendrás que usar la cabeza para resolver cada nivel. El objetivo es llevar a la

Palabras clave ?

Casual entertainment game supports iPhone 3GS. Videojuego casual entretenimiento iPhone 3GS

URL de soporte ?

<https://www.facebook.com/ivan.palominotayp>

URL de marketing ?

<http://example.com> (opcional)

Información general de la app

Icono de la app ?



Copyright ?

2015 Ivan Palomino Taype

Información de contacto del representante comercial ?

Muestra la información de contacto del representante comercial del App Store de Corea.

Ivan Palomino Taype

Ivan Palomino Taype

Palomino Taype

Información de Revisión de apps

Datos de contacto ?

Ivan Palomino Taype

+51 983010789 leo082893@hotmail.c

Notas ?

Figura 91: Configuración de la sección Preparar para la presentación
Fuente: elaboración propia

- G. Luego de llenar estas secciones se dio click en “Submit for Review”, y se esperó de 4 a 5 días hasta que los trabajadores de Apple revisaran el videojuego y lo aprobaran para ser publicado en la App Store.

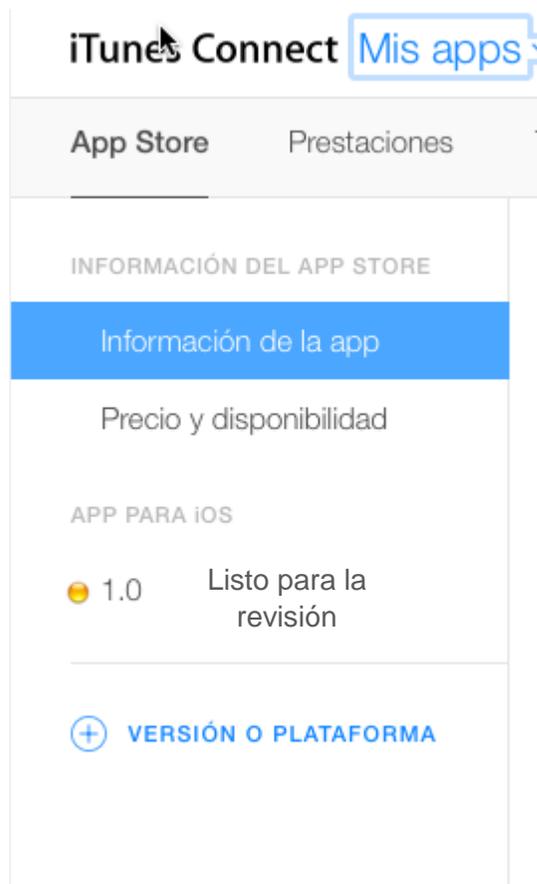


Figura 92: Videojuego listo para ser revisado por Apple

Fuente: elaboración propia

3.7. METODOLOGÍA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Esta metodología para medir la reducción de estrés está basada en la teoría para reducir el estrés, lo cual está detallado en las secciones (2.1.4 y 2.1.5) del presente informe, así como las características y definición del alfa de Cronbach detallados en la sección (2.2), además esta metodología se apoya principalmente en el Perfil de Estados de Ánimo elaborado por el jefe del departamento de Psicología y laboratorio clínico de Psicofarmacología de la Universidad de Boston, Douglas McNair y en el estudio “Eficacia de los videojuegos casuales para mejorar el buen humor y reducir el

estrés” realizado por el Doctor Carmen Russoniello de la Universidad del Este de California.

Este test y el cálculo de reducción de estrés fue validado por los doctores: Carmen R. Salas Zúñiga, Mirella Morales del Campo y Efraín Altamirano Morán, especialistas en Psicología humana, dicha validación puede verse en el **anexo nro. 3 “DOCUMENTOS DE VALIDACIÓN DE TEST DE REDUCCIÓN DE ESTRÉS”**, adicionalmente para medir la fiabilidad se aplicó el coeficiente alfa de Cronbach, el cual dio un valor de 0.82 para los medios positivos y un valor de 0.89 para los síntomas negativos con lo cual se obtuvo un coeficiente de alfa de Cronbach promedio de 0.86 para el test, lo cual puede verse en el **anexo nro. 5. “CÁLCULO DEL COEFICIENTE ALFA DE CRONBACH”**.

3.7.1. Estrategia de la recolección de datos

El test fue aplicado a 25 personas por el siguiente motivo: se formularon 5 preguntas en base a los medios positivos, entonces de acuerdo a la teoría se le aplicó a 5 personas por pregunta, ello da como resultado una cantidad de 25 personas; de manera similar se formularon 5 preguntas en base a los síntomas negativos y también se aplicó a 5 personas por pregunta con lo que da 25 personas. Entonces se aplicó el primer subgrupo de preguntas basadas en los medios positivos a cada persona y también se aplicó el segundo subgrupo de preguntas basadas en los síntomas negativos a cada persona, es decir que se aplicó los dos subgrupos a cada persona. Es así que en lugar de aplicar el primer subgrupo de preguntas a un grupo exclusivo de 25 personas y el segundo subgrupo de preguntas a otro grupo diferente de 25 personas, se aplicó el primer y segundo subgrupo a las mismas 25 personas, por ello se juntó estas preguntas en un solo cuestionario y se puso estas preguntas en un orden al azar para evitar inducir al encuestado a contestar de la misma manera cada pregunta. Este test se aplicó a personas entre hombres y mujeres de los 21 a 38 años de edad y se realizaron los siguientes pasos:

Paso 1: se hizo hacer jugar el videojuego por 5 minutos como mínimo a la persona a aplicar el test de medición de reducción de estrés.

Paso 2: se hizo llenar el test de medición de reducción de estrés a la persona.

3.7.2. Ficha de medición de reducción de estrés en las personas

El test de medición de reducción de estrés realizado por las personas luego de jugar el videojuego tuvo el siguiente formato, donde el orden se puso indistintamente para cada ítem de cada subgrupo con la finalidad de evitar inducir al encuestado a responder de la misma manera cada pregunta:

Evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa “**nada**” y **10** representa “**fuertemente/muchísimo**”.

1. Cuanto captó su atención el videojuego mientras lo jugaba (**mide aumento del entretenimiento a corto plazo**).

Respuesta (del 0 al 10): _____

2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades) (**mide aumento de distracción a corto plazo**).

Respuesta (del 0 al 10): _____

3. Cuanto lo relajó el videojuego (**mide reducción tensión a largo plazo**).

Respuesta (del 0 al 10): _____

4. Cuan alegre se siente ahora (**mide reducción de depresión a largo plazo**).

Respuesta (del 0 al 10): _____

5. Cuan tranquilo se siente (**mide reducción de ira a largo plazo**).

Respuesta (del 0 al 10): _____

6. El videojuego elevo su estado de ánimo (**mide aumento de vigor a largo plazo**).

Respuesta (del 0 al 10): _____

7. Cuan desahogado se siente (liberado, descansado) (**mide reducción de fatiga a largo plazo**).

Respuesta (del 0 al 10): _____

8. Cuan lucido se siente (cuanto despejo su mente el videojuego) (**mide reducción de confusión a largo plazo**).

Respuesta (del 0 al 10): _____

9. Cuan bien se sintió mientras jugaba el videojuego (**mide aumento de bienestar a corto plazo**).

Respuesta (del 0 al 10): _____

10. Se siente bien ahora (**mide aumento de bienestar a largo plazo**).

Respuesta (del 0 al 10): _____

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

3.7.3. Cálculo de la reducción del estrés

Los siguientes puntos detallan como se calculó el porcentaje de reducción de estrés.

- A. Se estableció como valor nulo (nada) al número 0, y como valor máximo (fuertemente/muchísimo) al número 10.
- B. Se realizó la equivalencia porcentual (0 = 0%) y (10 = 100%).
- C. Se tomó en cuenta los medios y síntomas del estrés, los medios que forman parte del primer subgrupo (positivos) que deben ser aumentados, para reducir el nivel de estrés; y los síntomas que forman parte del segundo subgrupo (negativos) que deben ser reducidos, para reducir el nivel de estrés.
- D. Los medios positivos a ser aumentados y que forman parte del primer subgrupo son:
 - **Entretenimiento:** el cual se logra con el aumento de la **captación de la atención** del usuario del videojuego.
 - **Distracción:** que se logra con el aumento de la **distracción** u olvido de los problemas del usuario del videojuego.
 - **Vigor:** que se logra con el aumento del estado de **ánimo** del usuario del videojuego.
 - **Bienestar a corto plazo:** que se logra con el aumento del **bienestar** general del usuario **mientras juega el videojuego**.

- **Bienestar a largo plazo:** que se logra con el aumento del **bienestar** general del usuario **después de haber jugado el videojuego**.
- E. Los síntomas negativos a ser reducidos y que forman parte del segundo subgrupo son:
- **Tensión:** que se logra con el aumento del nivel de **relajamiento**.
 - **Depresión:** que se logra con el aumento del nivel de **alegría**.
 - **Ira:** que se logra con el aumento del nivel de **tranquilidad**.
 - **Fatiga:** que se logra con el aumento del **desahogo**.
 - **Confusión:** que se logra con el aumento de la **lucidez mental**.
- F. Para los medios positivos a ser aumentados se consideró que el valor obtenido “n” represente un porcentaje con relación al intervalo de 0 a 10, el cual es obtenido por la regla de tres simple “(n x 100%)/10”, por lo tanto el valor porcentual obtenido se consideró como porcentaje de reducción de estrés obtenido.
- G. Para los síntomas negativos a ser reducidos se consideró que el valor obtenido “n” represente un porcentaje con relación al intervalo de 0 a 10, el cual es obtenido por la regla de tres simple “(n x 100%)/10”, por lo tanto el valor porcentual obtenido se consideró como porcentaje de reducción de estrés obtenido.
- H. La siguiente tabla explica como la variación de medios y síntomas reduce el estrés.

Tabla 72: *Relación entre la variación de los medios y síntomas del estrés y su reducción*

+		+		-		+
Pregunta al encuestado	Valor obtenido (0 al 10)	Medio positivo	Porcentaje de aumento	Síntoma negativo	Porcentaje de reducción	Porcentaje Reducción de Estrés
Captar la atención	8	Entretenimiento	80%			80%
Distracción (olvido de problemas)	9	Distracción	90%			90%
Relajamiento	7			Tensión	70%	70%
Alegría	6			Depresión	60%	60%

Fuente: elaboración propia

NOTA: los valores numéricos y porcentuales mostrados en tabla anterior fueron reemplazados por valores reales luego de la aplicación del test de reducción de estrés para cada medio positivo y síntoma negativo.

De la tabla se afirma que:

- El aumento de un medio positivo reduce el estrés.
- La reducción de un síntoma negativo reduce el estrés.

También se afirmar que:

- Para aumentar un medio positivo se debe maximizarlo.
- Para reducir un síntoma negativo se debe minimizarlo.

4. RESULTADOS

Basados en el objetivo general **“Desarrollar un videojuego multiplataforma para dispositivos móviles que consuma bajos recursos de hardware y que sea capaz de reducir de los niveles de estrés de las personas”**, el cual fue logrado al cumplir con los objetivos específicos:

- Determinar el videojuego apropiado de género casual.
- Diseñar el videojuego.
- Desarrollar el videojuego.
- Realizar las pruebas de funcionamiento del videojuego.
- Compilar el videojuego para que funcione en las plataformas Android e iOS de los teléfonos inteligentes.
- Lograr que el videojuego funcione fluida e íntegramente en los teléfonos inteligentes de gama baja.

- Evaluar el nivel de estrés de las personas usuarias del videojuego que tendrá la capacidad de captar la atención del usuario para brindarle momentos de relajación y distracción.

Se detalla a continuación los resultados obtenidos luego de la implementación y ejecución del proyecto de tesis:

Objetivo específico 1: Determinar el videojuego apropiado de género casual.

Este objetivo fue logrado al inicio de la ejecución del proyecto de tesis, y la documentación que valida esto puede verse en:

Sección 3.2.1. Concepto: donde se definió el Gameplay general del videojuego y los aspectos técnicos, de negocio y aspectos relacionados al estrés.

Sección 3.2.2. Planificación (Sección B): donde se detalló de manera específica el Gameplay del videojuego mediante los requisitos descritos.

Objetivo específico 2: Diseñar el videojuego.

Este objetivo fue logrado en la ejecución del proyecto de tesis, y la documentación que valida esto puede verse en:

Sección 3.2.3. Diseño: donde se diseñó cada una de las interfaces que tiene el videojuego basado en el flujo de trabajo Especificación o detalles del diseño de la metodología GUP y descripción de flujos de eventos que es parte de la Especificación de Casos de uso de RUP, también se hizo una representación gráfica de cómo se relacionan estas interfaces en el funcionamiento del videojuego, esto basado en el Diseño de paquetes de la metodología RUP.

Objetivo específico 3: Desarrollar el videojuego.

Este objetivo fue logrado en la ejecución del proyecto de tesis, y la documentación que valida esto puede verse en:

Sección 3.2.4. Elaboración: donde se muestra los requisitos desarrollados por cada una de las nueve iteraciones y la relación entre estos requisitos y las interfaces que los satisfacen, también se hace referencia al anexo donde se tiene el código fuente del videojuego.

Objetivo específico 4: Realizar las pruebas de funcionamiento del videojuego.

Este objetivo fue logrado en la ejecución del proyecto de tesis, y la documentación que valida esto puede verse en:

Sección 3.2.5. Beta: donde se planificó la manera en la que se realizaron las pruebas basado en el artefacto Casos de prueba de Caja Negra de la metodología RUP, se muestra el resultado de estas pruebas aplicado a cada una de las interfaces descritas en la fase de diseño mediante la técnica de partición equivalente, y también se describe la lista de errores corregidos.

Objetivo específico 5: Compilar el videojuego para que funcione en las plataformas Android e iOS de los teléfonos inteligentes.

A continuación se muestra una tabla con las fotos de las interfaces del videojuego funcionando en un teléfono inteligente LG L3 II E245g con sistema operativo Android v 4.1.2 (en la columna izquierda) y en un teléfono inteligente iPhone 3GS con sistema operativo iOS v 6.1.6 (en la columna derecha).

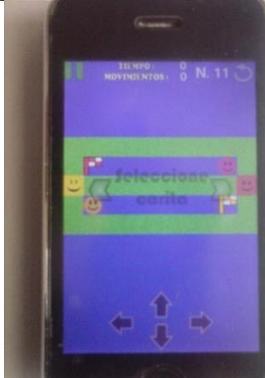
Tabla 73: *Imágenes de las interfaces del videojuego funcionando sobre los sistemas operativos Android e iOS de teléfonos inteligentes*

Interfaces	Sistema operativo	
	Android	iOS
Interfaz de primer splash		
Interfaz de segundo splash		

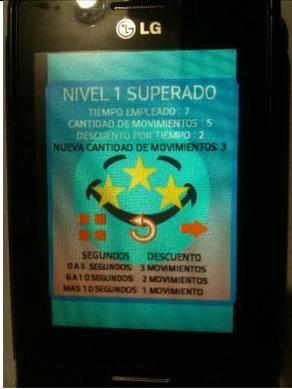
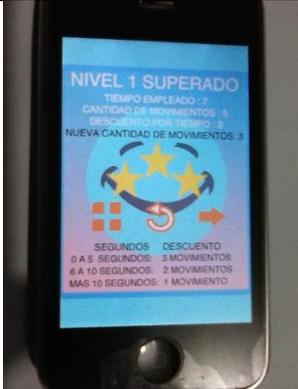
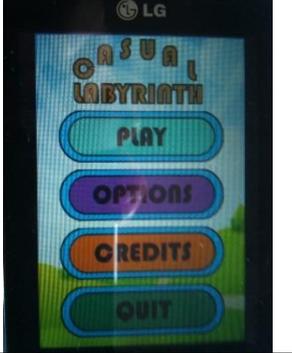
Continuación (Tabla 73: Imágenes de las interfaces del videojuego funcionando sobre los sistemas operativos Android e iOS de teléfonos inteligentes)

Interfaces	Sistema operativo	
	Android	iOS
Interfaz de menú principal		
Interfaz de over relay de menú principal		
Interfaz de menú de niveles		
Interfaz de primer tutorial		

Continuación (Tabla 73: *Imágenes de las interfaces del videojuego funcionando sobre los sistemas operativos Android e iOS de teléfonos inteligentes*)

Interfaces	Sistema operativo	
	Android	iOS
Interfaz de nivel (10 primeros)		
Interfaz de segundo tutorial		
Interfaz de nivel (10 últimos)		
Interfaz de over relay de pausa		

Continuación (Tabla 73: Imágenes de las interfaces del videojuego funcionando sobre los sistemas operativos Android e iOS de teléfonos inteligentes)

Interfaces	Sistema operativo	
	Android	iOS
Interfaz de over relay de nivel		
Interfaz de créditos		
Interfaz de menú principal en inglés		
Interfaz de over relay de menú principal en inglés		

Fuente: elaboración propia

También se muestra el videojuego subido a la tienda Play Store de Android, disponible en: <https://play.google.com/store/apps/details?id=com.gmail.alfil1520.CL>

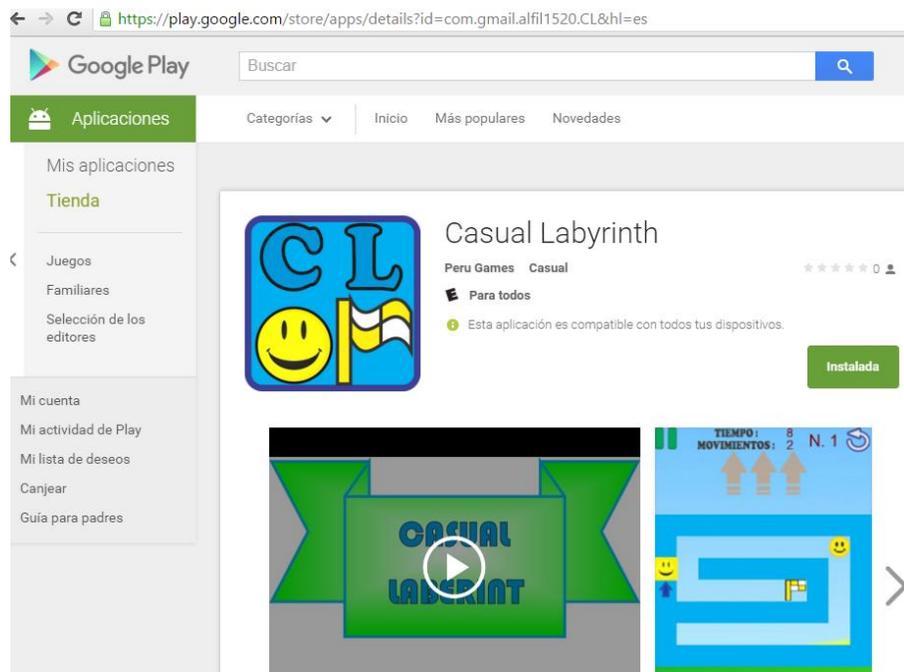


Figura 93: Videojuego subido a Play Store
Fuente: elaboración propia

Así como también se muestra el videojuego subido a la tienda App Store de iOS, disponible en: <https://itunes.apple.com/us/app/casual-labyrinth/id1083974867?mt=8>

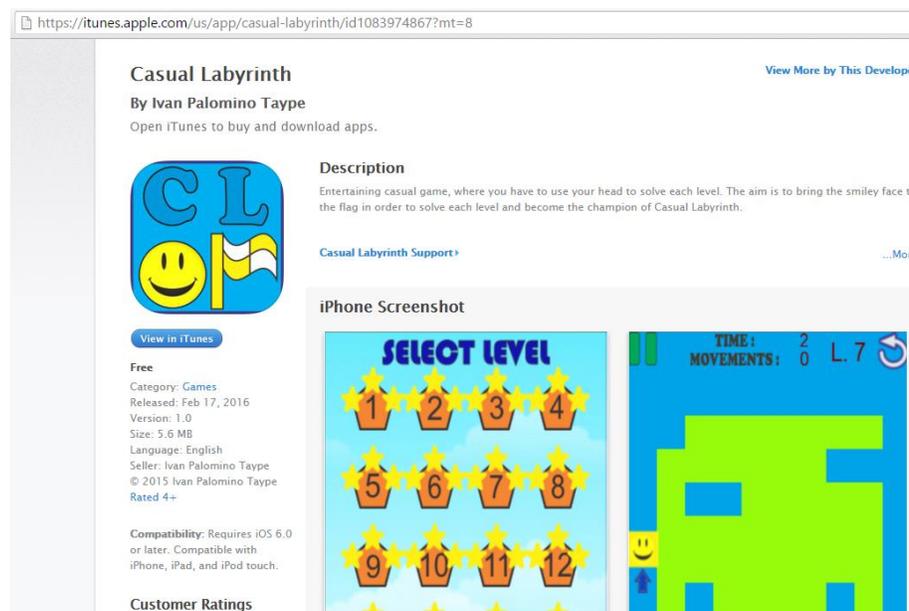


Figura 94: Videojuego subido a App Store
Fuente: elaboración propia

Objetivo específico 6: Lograr que el videojuego funcione fluida e íntegramente en los teléfonos inteligentes de gama baja.

Este objetivo fue logrado al lograr que el videojuego desarrollado consuma menos recursos que otros videojuegos, la prueba se realizó usando la aplicación de monitoreo de recursos usados M2AppMonitor v 2.1.4 dentro de un teléfono de gama baja LG L3 II con 512 MB de memoria RAM, 1GHz de procesador mono núcleo, batería Standard, Li-Ion 1540 mAh (7700 mWh) y 4GB de almacenamiento interno, la siguiente tabla detalla información la consumo de recursos por cada videojuego, cuyos datos pueden corroborarse en el **anexo nro. 6. “IMÁGENES DEL CONSUMO DE RECURSOS”**:

Tabla 74: Recursos de hardware consumidos por los videojuegos

Recursos	Videojuegos					
	Casual Labyrinth	Bejeweled 2	Magne Tank	Maze King	Pegland Deluxe	WordWorm Adventures
Batería (mW)	438	518	646	469	721	605
Memoria RAM (MB)	24	53	19	37	11	75
CPU (%)	18	36	100	23	100	18
Espacio de almacenamiento (MB)	13	40	13	19	38	24

Fuente: elaboración propia

De la tabla mostrada se puede ver que el videojuego desarrollado consume menos batería que los demás videojuegos, necesita menos memoria RAM que los demás videojuegos excepto Magane Tank y Pegland, usa menos porcentaje de CPU que los demás pero igual que WordWorm y necesita menos espacio de almacenamiento pero igual que Magne Tank. La siguiente tabla muestra el porcentaje de reducción de consumo por cada recurso principal que posee un teléfono inteligente:

Tabla 75: Comparación de recursos de hardware consumidos por los otros videojuegos y el videojuego desarrollado

Recursos	Videojuego		
	Promedio otros videojuegos	Casual Labyrinth	Porcentaje de consumo reducido
Batería (mW)	591.8	438	26.0%
Memoria RAM (MB)	39	24	38.5%
CPU (%)	55.4	18	67.5%
Espacio de almacenamiento	26.8	13	51.5%

Fuente: elaboración propia

De la tabla mostrada se deduce que en general el videojuego desarrollado consume menos recursos tanto en batería con un 26.0% menos, memoria RAM con un 38.5% menos, CPU con un 67.5% menos y espacio de almacenamiento con un 51.5% menos.

Objetivo específico 7: Evaluar el nivel de estrés de las personas usuarias del videojuego que tendrá la capacidad de captar la atención del usuario para brindarle momentos de relajación y distracción.

Este objetivo fue logrado luego de obtener y procesar los datos del efecto del videojuego, para lo cual se aplicó el test de reducción de estrés a 25 personas, quienes antes de realizar el test de reducción de estrés jugaron el videojuego por aproximadamente 5 minutos, las encuestas realizadas pueden verse en el **anexo nro. 4 “FICHAS DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS LLENADAS”** los resultados se muestran a continuación:

A) Reducción de estrés por medio y síntoma

Tabla 76: *Estrés reducido por medio y síntoma*

N.	AUMENTO DE:	IMPLICA REDUCCIÓN DE:	IMPLICA AUMENTO DE:	VALOR REDUCIDO /AUMENTADO	VALOR PORCENTUAL (%)
1	Captación de la atención		Entretenimiento	7.80	78.0%
2	Distracción(olvido de problemas)		Distracción	8.4	84.0%
3	Relajo	Tensión		7.68	76.8%
4	Alegría	Depresión		7.64	76.4%
5	Tranquilidad	Ira		7.52	75.2%
6	Estado de ánimo		Vigor	7.88	78.8%
7	Desahogo	Fatiga		7.8	78.0%
8	Lucidez	Confusión		7.48	74.8%
9	Bienestar durante el juego		Bienestar a corto plazo	7.64	76.4%
10	Bienestar después del juego		Bienestar a largo plazo	8.08	80.8%
				MEDIA ARITMÉTICA GLOBAL	77.9%

Fuente: elaboración propia

De la tabla anterior se puede concluir lo siguiente:

En los medios positivos:

- Al aumentar el entretenimiento en un 78.0%, el entretenimiento redujo el estrés en un 78.0%.
- Al aumentar la distracción en un 84.0%, la distracción redujo el estrés en un 84.0%.
- Al aumentar el vigor en un 78.8%, el vigor redujo el estrés en un 78.8%.
- Al aumentar el bienestar a corto plazo en un 76.4%, el bienestar a corto plazo redujo el estrés en un 76.4%.
- Al aumentar el bienestar a largo plazo en un 80.8%, el bienestar a largo plazo redujo el estrés en un 80.8%.

En los síntomas negativos:

- Al reducir la tensión en un 76.8%, se redujo el estrés vinculado a la tensión en un 76.8%.
- Al reducir la depresión en un 76.4%, se redujo el estrés vinculado a la depresión en un 76.4%.
- Al reducir la ira en un 75.2%, se redujo el estrés vinculado a la ira en un 75.2%.
- Al reducir la fatiga en un 78.0%, se redujo el estrés vinculado a la fatiga en un 78.0%.
- Al reducir la confusión en un 74.8%, se redujo el estrés vinculado a la confusión en un 74.8%.

Por lo tanto al hacer variar adecuadamente los medios y síntomas del estrés se logró reducirlo en un 77.9%.

B) Reducción de estrés por persona

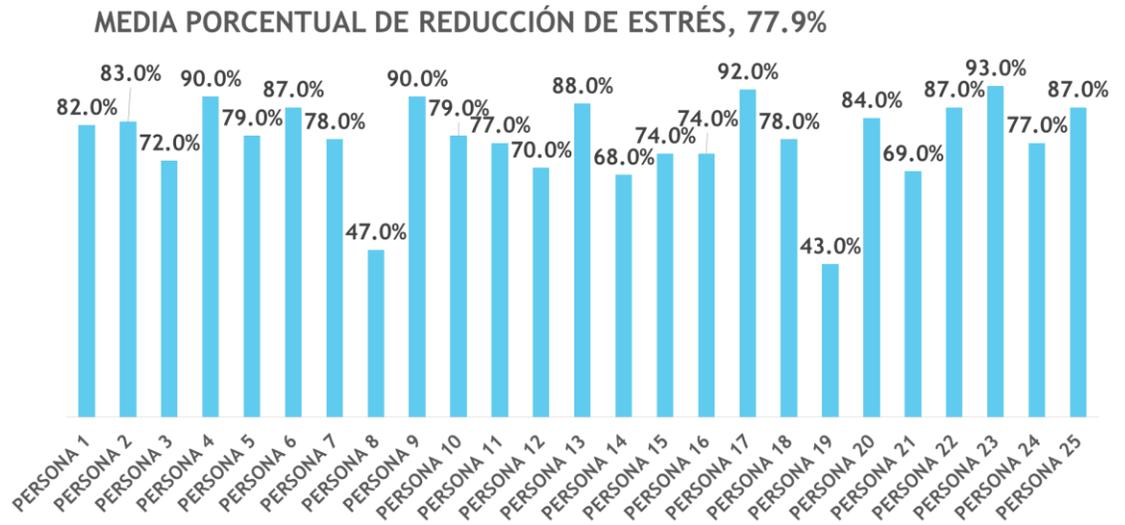


Figura 95: Estrés reducido por persona

Fuente: elaboración propia.

De la tabla anterior se puede concluir lo siguiente:

- Se logró reducir el estrés de la persona 1 en un 82.0 %.
- Se logró reducir el estrés de la persona 2 en un 83.0 %.
- Se logró reducir el estrés de la persona 3 en un 72.0 %.
- Se logró reducir el estrés de la persona 4 en un 90.0 %.
- Se logró reducir el estrés de la persona 5 en un 79.0 %.
- Se logró reducir el estrés de la persona 6 en un 87.0 %.
- Se logró reducir el estrés de la persona 7 en un 78.0 %.
- Se logró reducir el estrés de la persona 8 en un 47.0 %.
- Se logró reducir el estrés de la persona 9 en un 90.0 %.
- Se logró reducir el estrés de la persona 10 en un 79.0 %.
- Se logró reducir el estrés de la persona 11 en un 77.0 %.
- Se logró reducir el estrés de la persona 12 en un 70.0 %.
- Se logró reducir el estrés de la persona 13 en un 88.0 %.
- Se logró reducir el estrés de la persona 14 en un 68.0 %.
- Se logró reducir el estrés de la persona 15 en un 74.0 %.
- Se logró reducir el estrés de la persona 16 en un 74.0 %.

- Se logró reducir el estrés de la persona 17 en un 92.0 %.
- Se logró reducir el estrés de la persona 18 en un 78.0 %.
- Se logró reducir el estrés de la persona 19 en un 43.0 %.
- Se logró reducir el estrés de la persona 20 en un 84.0 %.
- Se logró reducir el estrés de la persona 21 en un 69.0 %.
- Se logró reducir el estrés de la persona 22 en un 87.0 %.
- Se logró reducir el estrés de la persona 23 en un 93.0 %.
- Se logró reducir el estrés de la persona 24 en un 77.0 %.
- Se logró reducir el estrés de la persona 25 en un 87.0%

En consecuencia el promedio de reducción de estrés por persona logrado fue de 77.9%.

CONCLUSIONES

Luego de ejecutar el presente proyecto de investigación tecnológica se concluye que:

- La metodología SUM permitió determinar, diseñar, programar y probar el videojuego requerido, dicho videojuego fue llamado “Casual Labyrinth”, el cual pertenece al género casual. La naturaleza de la metodología SUM permitió lograr esto ya que se combinó con otras dos metodologías; en mayor porcentaje se usó la metodología SUM, aplicado para las fases de concepto, planificación, elaboración, beta y cierre, también se usó la metodología GUP, aplicado a la fase de diseño y de manera similar la metodología RUP, aplicado a las fases de diseño, beta y cierre. De esta manera se pudo obtener el videojuego acorde a lo planificado.
- Es posible generar los instaladores del videojuego desarrollado a partir del mismo código fuente y también realizar su despliegue para que pueda ser instalado en los teléfonos inteligentes con sistema operativo Android e iOS, tanto a través de sus tiendas virtuales Play Store y App Store, así como de manera local, es decir directamente en dichos teléfonos inteligentes.
- Es posible lograr que el videojuego desarrollado funcione fluidamente en ambos sistemas operativos de teléfonos inteligentes de gama baja y consume menos recursos que otros videojuegos similares que también pertenecen al género casual y son de dimensión 2D.
- Usar videojuegos casuales para prevenir y reducir el estrés es una manera eficaz de hacer frente a los efectos generados por el estrés que afectan a la salud de las personas, ello se corrobora con la evidencia científica descubierta y con el videojuego desarrollado con el cual se logró reducir el nivel de estrés de las personas en un 77.9% en promedio.

RECOMENDACIONES

Con la finalidad de realizar trabajos de investigación similares al detallado en este documento, así como trabajos futuros, se recomienda lo siguiente:

- Desarrollar videojuegos requiere de un equipo especialista en cada área, para garantizar que el trabajo obtenido sea de alta calidad. Además se debe contar con un promotor publicitario para hacer conocer el videojuego a la mayor cantidad de personas posible.
- Combinar dos o tres metodologías adicionales a la metodología SUM, con la finalidad de realizar un proceso de desarrollo más sólido y entendible.
- Realizar pruebas desde el inicio del desarrollo para prevenir errores futuros que perjudiquen al videojuego.
- Realizar el despliegue primero para el sistema operativo Android ya que es mucho más fácil que realizar el despliegue para el sistema operativo iOS.
- Comprimir lo más que se pueda los archivos de audio y los gráficos, así como optimizar los algoritmos necesarios para el videojuego y de esta manera evitar que éste consuma demasiados recursos de hardware de los teléfonos inteligentes de gama baja.
- Usar el motor de desarrollo de videojuegos Corona SDK ya que es el mejor en cuanto a tiempo y facilidad para desarrollo de videojuegos en 2D.
- Realizar proyectos en el menor tiempo posible ya que el avance de la tecnología hace que en poco tiempo los dispositivos electrónicos se desfasen y no guarden concordancia con los temas planteados en un proyecto de investigación tecnológica.
- Aprender el idioma Inglés, debido a que la documentación más actualizada, tutoriales, manuales de programación, procesos de compra de licencia de distribución y similares están en dicho idioma.

REFERENCIAS BIBLIOGRÁFICAS

- Acerenza, N., Coppes, A., Mesa, G., Viera, A., Lorenzo, T. y Vallespir, D. (2009). Una Metodología para Desarrollo de Videojuegos. 38º JAIIO - Simposio Argentino de Ingeniería de Software (ASSE 2009), Argentina. Recuperado de: http://www.fing.edu.uy/inco/grupos/gris/wiki/uploads/Proceedings/ASSE_2009_16.pdf
- Alonso, A. B., Artime, I. F., Rodríguez, M. Á., y Baniello, R. G. (2009, 26 de Octubre). Dispositivos móviles, EPSIG Ing. Telecomunicación Universidad de Oviedo. Recuperado de: http://isa.uniovi.es/docencia/SIGC/pdf/telefonía_movil.pdf
- Amaro, J. (2015, 10 de Junio). Uso de smartphones en Perú es uno de los más crecientes en Latinoamérica. *La Republica*. Recuperado de: <http://larepublica.pe/marketing/4540-uso-de-smartphones-en-peru-es-uno-de-los-mas-crecientes-en-latinoamerica>
- Android. (2015). The Android Story [Sitio web oficial]. Recuperado de: <https://www.android.com/history>
- App Monitor. (2015). Improving The App Experience, Together [Sitio web oficial]. Recuperado de: <http://m2appmonitor.com/>
- Armendáriz Barreno, G. A. y Saltos Guaraca M. G. (2013). *Adaptación De Las Metodologías Ágiles Scrum Y Extreme Game Development en una metodología para el desarrollo de videojuegos en Android. Caso Práctico: Desarrollo de un videojuego*. (Tesis). Escuela Superior Técnica de Chimborazo. Facultad de Informática y Electrónica, Ecuador. Recuperado de <http://dspace.espace.edu.ec/bitstream/123456789/27111/1/18T00530.pdf>
- Artigames. (2014, 29 de Octubre). ¿Sabías que jugar videojuegos tiene sus beneficios?. Recuperado de: <http://videojuegos-peru.com/2014/10/29/sabias-que-jugar-videojuegos-tiene-sus-beneficios/>
- Audacity. (2014). Acerca de Audacity [Sitio web oficial]. Recuperado de <http://audacity.sourceforge.net/about/>
- AVG. (2015, Marzo). Android App Performance Report. Recuperado de: <http://www.slideshare.net/AVGShare/avg-android-app-performance-report-q1-2015>

- Betancourth-Alvarenga, J. (2007). Videojuegos: Más allá de la Mente. *BUN Synapsis*, 2(4), 15-20. Recuperado de [http://65.182.2.244/Honduras/SUN.THEPIXIE.NET/files/BUN_SYN_2007_2\(4\)D.pdf](http://65.182.2.244/Honduras/SUN.THEPIXIE.NET/files/BUN_SYN_2007_2(4)D.pdf)
- Bienestar180. (2015, 3 de Agosto). Top 5 de países más estresados. Recuperado de: <http://bienestar.salud180.com/salud-dia-dia/top-5-de-paises-mas-estresados>
- Black, B. (2010, 15 de Marzo). The Link Between Anger and Stress. Recuperado de: <https://www.mentalhelp.net/blogs/the-link-between-anger-and-stress/>
- Boiron. (2009, 12 de Noviembre). Tratamientos para el Estrés. Recuperado de: <http://www.boiron.es/tratamiento-estres-ansiedad#>
- Brianmac. (2015, 08 de Junio). Profile of Mood States (POMS). Recuperado de: <http://www.brianmac.co.uk/poms.htm#ref>
- Bryant, J., & Vorderer, P. (2011). *Psychology of Entertainment*. New York, EEUU: Routledge. Recuperado de: https://books.google.com.pe/books?id=AVnhAQAAQBAJ&pg=PA399&lpg=PA399&dq=entertainment+to+reduce+stress&source=bl&ots=cbwIVAuF44&sig=ADBunnAlFbKJHyffR2wMeyvHf38&hl=es&sa=X&ved=0ahUKEwicj_uet-TLAhXI4SYKHaE3ABQ4ChDoAQgpMAI#v=onepage&q=entertainment%20to%20reduce%20stress&f=false
- Campo-Arias, A. y Oviedo, H. (2008). Propiedades Psicométricas de una Escala: la Consistencia Interna. *Revista de Salud Pública*, 10 (5), 831-839. doi: S0124-00642008000500015
- CampusMPV. (2014, 17 de Junio). Programación móvil: Qué herramienta y lenguaje elegir. Recuperado de: <http://www.campusmpv.es/recursos/post/Programacion-movil-Que-herramienta-y-lenguaje-elegir.aspx>
- CEINTEC. (2015, 8 de Agosto). Los videojuegos cómo terapia. [Página web de tecnología]. Recuperado de <http://www.ceintec.com/articulos/los-videojuegos-como-terapia.-4145.html>
- Cejas, E. (2012, 21 de Noviembre). Mejores Teléfonos Android (2012) [Blog]. Recuperado de <http://androidzone.org/2012/11/mejores-telefonos-android-2012/>

- Cervantes, V. H. (2005). Interpretaciones del coeficiente alpha de Cronbach. *Avances en Medición*, (3), 9-28. Recuperado de: http://www.humanas.unal.edu.co/psicometria/files/1113/8574/8604/Articulo_1_Alfa_d_e_Cronbach_9-28_2.pdf
- CNNEXPANSION. (2008, 28 de Abril). Videojuegos reducen el estrés [Portal de Noticias]. Recuperado de <http://www.cnnexpansion.com/economia-insolita/comprobado-videojuegos-reducen-estres?q=2532749150>
- Comín Anadon, E.; De la Fuente, I. y Gracia Galve, A. (2009, 12 de Octubre). *El estrés y el riesgo para la salud*. España: MAZ. Recuperado de: <http://www.uma.es/publicadores/prevencion/wwwuma/estres.pdf>
- Compress PNG. (2015). With the force of an elephant [Sitio web oficial]. Recuperado de: <http://compresspng.com/>
- Conde R. (2013). Lista de los teléfonos celulares clasificados por gamas: Desde los poderosos smartphones de alta gama plus hasta los básicos. Recuperado de <http://celulares.about.com/od/Smartphones/tp/Lista-De-Los-Telefonos-Celulares-Clasificados-Por-Gamas.htm>
- CorelDRAW. (2014, 8 de Abril). CorelDRAW Graphics Suite x6. CorelDRAW Corporation. Recuperado de www.corel.com/content/vpk/cdgsx6/cdgsx6_reviewers_guide_es.pdf
- CoronaLabs. (2015a). Guides. Recuperado de: <https://docs.coronalabs.com/guide/index.html>
- CoronaLabs. (2015b). Develop Cross Platform Mobile Apps and Games. Recuperado de: <https://coronalabs.com/products/corona-sdk/>
- CoronaLabs. (2015c). SDK API Reference. Recuperado de: <https://docs.coronalabs.com/api/index.html>
- Corral, Y. (2009). Validez y confiabilidad de los instrumentos de investigación para la recolección de datos. *Revista ciencias de la educación*, 19(33), 228-247. Recuperado de: <http://servicio.bc.uc.edu.ve/educacion/revista/n33/art12.pdf>
- Crew. (2015, 04 de Marzo). How much to make an app [Empresa de desarrollo de aplicaciones móviles]. Recuperado de: <http://howmuchtomakeanapp.com/>

- Davis, C. R. (2008). "From uptight to all right!" The management of stress. Recuperado de: http://www.mindwise.com.au/articles/From_uptight_to_alright-stress_management.pdf
- Digimark. (2012). Mobile OS Platforms We Build Apps For. Recuperado de: <http://www.digimark.bg/en/mobile-os-platforms>
- Dolci, M. (2014, 23 de Enero). 8 consejos útiles para diseñar un videojuego casual: Capítulo 2. Recuperado de: <http://www.neomobile-blog.com/es/como-crear-video-juegos-casuales-2/>
- EcuRed. (2010, 24 de Junio). Anexo: Videojuego (Géneros). Recuperado de: [http://www.ecured.cu/Anexo:Videojuego_\(G%C3%A9neros\)](http://www.ecured.cu/Anexo:Videojuego_(G%C3%A9neros))
- EcuRed. (2011, 11 de Noviembre). Extreme Game Development. Recuperado de http://www.ecured.cu/Extreme_Game_Development
- EcuRed. (2012, 25 de Junio). Game Unified Process. Recuperado de http://www.ecured.cu/index.php/Game_Unified_Process
- El Comercio. (2014a, 18 de Febrero). Latinoamérica lideró el aumento de smartphones en el mundo. *El Comercio*. Recuperado de: <http://elcomercio.pe/paginas/smartphones-tablets/latinoamerica-comando-aumento-smartphones-mundo-noticia-1792109>
- El Comercio. (2014b, 10 de diciembre). La industria de videojuegos es tres veces la del cine. *El Comercio*. Recuperado de: <http://elcomercio.pe/economia/negocios/industria-videojuegos-tres-veces-cine-noticia-1777239>
- Fernández, M. (2015, 4 de Febrero). Los móviles más usados según la zona del planeta. Recuperado de: http://elpais.com/eventos/2015/01/29/mwc/1422537682_294661.html
- Fernández Martínez, L. C. (2014). *Desarrollo de la mecánica y dinámica de un videojuego serio 3D en tercera persona*. (Tesis). Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería, Perú. Recuperado de <http://tesis.pucp.edu.pe/repositorio/handle/123456789/5155>
- Filgueira, J. M. (2009, 3 de Diciembre). ¿Qué es iOS? | Gabit [Consultoría de TIC]. Recuperado de: <http://www.gabit.org/gabit/?q=es/que-es-ios>
- Filmora. (2010). Wondershare Filmora Video Editor [Sitio web oficial]. Recuperado de: <http://filmora.wondershare.es/editor-de-video/>

- Frías, D. (2014). Apuntes de SPSS. Recuperado de: <http://www.uv.es/friasnav/ApuntesSPSS.pdf>
- GamerDic. (2013, 16 de Marzo). Diccionario online de términos sobre videojuegos y cultura gamer. Recuperado de: <http://www.gamerdic.es/tema/generos>
- García Sánchez-Colomer, M. (2015). Videojuegos: entretenimiento sano, instrumento educativo... o lo contrario. Recuperado de <http://es.catholic.net/op/articulos/22145/videojuegos-entretenimiento-sano-instrumento-educativo-o-lo-contrario.html>
- Gerardi, M. (2013, 11 de Junio). ¿Qué es un motor de juego, y ¿por qué EA quieren que usted se preocupe por él?. Recuperado de <http://gameological.com/2013/06/ea-e3-2013-frostbite-plants-vs-zombies/>
- González, J. A. y Pazmiño, M. (2015). Cálculo e interpretación del Alfa de Cronbach para el caso de validación de la consistencia interna de un cuestionario, con dos posibles escalas tipo Likert. *Revista Publicando*, 2(1), 62-77. Recuperado de: <http://www.rmlconsultores.com/revista/index.php/crv/article/view/22>
- Gramlich, N. (2013, 22 de Enero). New AndEngine book – AndEngine for Android Game Development Cookbook. Recuperado de: <http://www.andengine.org/blog/>
- GSMarena. (2014). Apple phones [Sitio web de telefonía móvil]. Recuperado de <http://www.gsmarena.com/apple-phones-48.php>
- Herrán, O. F. y Ardilla, M. F. (2007). Validez factorial del cuestionario de expectativas hacia el consumo de alcohol (AEQ-III), en adultos de Bucaramanga, Colombia. *Revista médica de Chile*, 135(4), 488-495. doi: 10.4067/S0034-98872007000400011
- Hordé, P. (2014, Diciembre). Benzodiazepines - Effets secondaires et risques. Recuperado de: <http://sante-medecine.journaldesfemmes.com/faq/7748-benzodiazepines-effets-secondaires-et-risques>
- Idóneos (2013, 20 de Noviembre). CAPITULO II: EL ESTRÉS. Recuperado de <http://evaluacioninstitucional.idoneos.com/345577/>
- Inkscape. (2014). What is Inkscape? [Sitio web oficial]. Recuperado de <https://inkscape.org/es/acerca-de/>
- InspireSoft. (2015). MP3 Quality Modifier [Sitio web oficial]. Recuperado de: <http://www.inspire-soft.net/software/mp3-quality-modifier>

- International Game Developers Association. (2008). *2008-2009 Casual Games White Paper*. Estados Unidos.: Author. Recuperado de: https://c.ymcdn.com/sites/www.igda.org/resource/collection/BCB11E9B-13E6-40D0-B390-952B5E11D35A/IGDA_Casual_Games_White_Paper_2008.pdf
- ISO 25000. (2011a). La familia de normas ISO/IEC 25000. Recuperado de: <http://iso25000.com/index.php/normas-iso-25000>
- ISO 25000. (2011b). ISO/IEC 25010. Recuperado de: <http://iso25000.com/index.php/normas-iso-25000/iso-25010>
- Java. (2015). Java y tú, descargar hoy [Sitio web oficial]. Recuperado de: <https://www.java.com/es/>
- Karam J. M. (2005, 28 de enero). Psicología y videojuegos ¿adicción o arte? Recuperado de <http://www.mundogamers.com/principal/articulo/8-2/psicologia-y-videojuegos-adiccion-o-arte.html>
- Ku Lam, C. A. (2014). *Diseño e implementación de una plataforma interactiva móvil orientada al aprendizaje temas de física*. (Tesis). Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería, Perú. Recuperado de <http://tesis.pucp.edu.pe/repositorio/handle/123456789/5427>
- LaFamilia.info. (2013, 04 de Junio). ¿Pueden las emociones afectar la salud?. Recuperado de: <http://www.lafamilia.info/index.php/salud/149-salud/salud-y-bienestar/1143-pueden-las-emociones-afectar-la-salud>
- La Nación. (2006, 30 de Diciembre). Científicos explican por qué la risa es contagiosa. Recuperado de: <http://www.lanacion.com.ar/871486-cientificos-explican-por-que-la-risa-es-contagiosa>
- León León A. (2014). *Videojuego multiplataforma para dispositivos móviles*. (Tesis). Universidad de Castilla-La Mancha, Escuela Superior de Informática, España. Recuperado de <https://ruidera.uclm.es/xmlui/handle/10578/3995>
- LG. (2015). LG Optimus L3 II Móviles - LG Electronics. Recuperado de: <http://www.lg.com/es/telefonos-moviles/lg-Optimus-L3-II-E430>
- LibGDX. (2013). Goals and Features [Sitio web oficial]. Recuperado de: <https://libgdx.badlogicgames.com/features.html>

- Lifshitz, A. A. (2012, 25 de Mayo). ¿De qué forma nos enferma el estrés?. Recuperado de <http://www.vidaysalud.com/diario/estres-y-salud-mental/de-que-forma-nos-enferma-el-estres/>
- Lifshitz, A. A. (2014, 23 de Mayo). Estrategias para combatir el estrés. Recuperado de: <http://www.vidaysalud.com/diario/estres-y-salud-mental/estrategias-para-combatir-el-estres/>
- Llansó García, D. (2014). *Metodología ontológica para el desarrollo de videojuegos*. (Tesis doctoral). Universidad Complutense de Madrid, Facultad de Informática, Madrid, España. Recuperado de <http://eprints.ucm.es/25323/>
- Loizte, J. A. (2004). Emigrar en situación extrema: el Síndrome del inmigrante con estrés crónico y múltiple (Síndrome de Ulises). *Norte de Salud mental*, 5(21), 39-52. Recuperado de: <http://www.revistanorte.es/index.php/revista/article/download/348/344>
- López Jauffred, E. A., (2013, 22 de Febrero). Cómo lidiar con la tensión emocional y el estrés. Recuperado de: <http://www.altonivel.com.mx/como-lidiar-con-la-tension-emocional-y-el-estres.html>
- Lua. (2014, 12 de Octubre). ¿Qué es Lua? [Sitio web oficial]. Recuperado de <http://www.lua.org/about.html>
- Management Society. (2015). Los smartphones y el consumidor móvil. Recuperado de: <http://managementsociety.net/marketing/135-los-smartphones-y-el-consumidor-movil.html>
- Mayo Clinic. (2013, 23 de Julio). El alivio de tensión de la risa? No es un chiste. Recuperado de: <http://www.mayoclinic.org/healthy-lifestyle/stress-management/in-depth/stress-relief/art-20044456>
- Mendoza, M. (2011). Elaboración y validación del cuestionario: “Desempeños profesionales de directivos y profesores en pro de una educación creativa: Evaluación y autoevaluación desde la perspectiva docente”. *Revista Docencia e Investigación*, (21), 51-70. Recuperado de: <https://dialnet.unirioja.es/servlet/articulo?codigo=4009750>

- Mercer, N. (2015, 12 de Mayo). Stress Relieving Video Games: Creating a Game for the Purpose of Stress Relief and Analyzing Its Effectiveness. Honors program theses and projects, Item 108. Recuperado de: http://vc.bridgew.edu/honors_proj/108
- Microsoft. (2014). TechNet Evaluation Center Recuperado de: <https://www.microsoft.com/es-es/evalcenter/evaluate-office-professional-plus-2013>
- Miller, A. & Smith, A. D. (2015, 28 de Mayo). Los distintos tipos de estrés. Recuperado de: <http://www.apa.org/centrodeapoyo/tipos.aspx>
- Ministerio de Educación de España. (2008a). Diseño de materiales multimedia. Recuperado de: <http://www.ite.educacion.es/formacion/materiales/107/cd/audio/audio01.html>
- Ministerio de Educación de España. (2008b). Diseño de materiales multimedia. Recuperado de: <http://www.ite.educacion.es/formacion/materiales/107/cd/imagen/imagen01.html>
- Montoya Álvarez, D., Suárez Quintero, F. J. y Ortíz Varón, L. F. (2014). *Juego de ajedrez sobre dispositivos móviles con sistema operativo Android para personas con limitación visual*. (Tesis). Universidad Tecnológica de Pereira, Facultad de Ingenierías, Electrica, Electrónico, Físico y Ciencias de la Computación, Colombia. Recuperado de: <http://repositorio.utp.edu.co/dspace/handle/11059/4192>
- Motyka, J. (2014, 20 de Octubre). Android, el sistema operativo cuya historia se resume en seis años. Recuperado de <http://www.tuexpertomovil.com/2014/10/20/android-el-sistema-operativo-cuya-historia-se-resume-en-seis-anos/>
- nehuen. (2011, 2 de Noviembre). Angry Birds celebra las 500 millones de descargas. Recuperado de: <http://hipertextual.com/archivo/2011/11/angry-birds-celebra-las-500-millones-de-descargas/>
- Netdreams. (2015, 17 de Agosto). Dispositivos Móviles Perú Sell-In Primer Semestre 2015 (half year): Más de 3.3 millones de smartphones y tablets llegan al Perú. Recuperado de: <http://netdreams.pe/blog/mundo-movil/dispositivos-moviles-peru-sell-in-primer-semestre-2015-half-year-mas-de-3-3-millones-de-smartphones-y-tablets-llegan-al-peru/>
- Netmarketshare. (2015, Diciembre). Mobile/Tablet Operating System Market Share. Recuperado de: <http://www.netmarketshare.com/>
- Oviedo, H y Campo-Arias, A. (2005). Aproximación al uso del coeficiente alfa de Cronbach. *Revista Colombiana de Psiquiatría*, 34(4), 572-580. doi: S0034-74502005000400009

- Palella, S. y Martins, F. (2006). *Metodología de la investigación cuantitativa*. (2da ed.). Caracas, Venezuela: FEDUPEL. Recuperado de: <https://es.scribd.com/doc/178265729/Metodologia-de-la-investigacion-cuantitativa-Palella1-pdf>
- Pastor Pérez, L. (2010). FUniversity: *Los Medios de Comunicación cambian la Universidad*. Barcelona, España: UOC. Recuperado de: <https://books.google.com.pe/books?id=8gHQCwAAQBAJ&pg=PA45&lpg=PA45&dq=captar+la+atencion+para+entretener&source=bl&ots=4LsujBdKPf&sig=iCDJE8Hbix9pZfe8AiTqG7XMJYo&hl=es&sa=X&ved=0ahUKEwjpV9HH3eTLAhUMGB4KHRXnCQ04ChDoAQg8MAg#v=onepage&q=captar%20la%20atencion%20para%20entretener&f=false>
- Paris, L., y Omar, A. (2013). Estrategias de afrontamiento del estrés como potenciadoras de bienestar. *Psicología y Salud*, 19(2), 167-175. Recuperado de: <http://revistas.uv.mx/index.php/psicysalud/article/view/626>
- Peru21. (2013, 6 de Agosto). ¿Interesado en el desarrollo de videojuegos?. *Perú21*. Recuperado de <http://peru21.pe/vida21/interesado-desarrollo-videojuegos-2143421>
- PLENIUM. (2015, 09 de Octubre). Estrés Sevilla - Centro Plenium Salud Física y Emocional. Recuperado de: <http://centroplenium.es/salud-emocional/estres/>
- Quispe Gonzales, L. A. (2013). *Análisis, diseño e implementación de un videojuego en 2d orientado a la ejercitación de la memoria y el desarrollo de la inteligencia espacial*. (Tesis) Pontificia Universidad Católica del Perú, Ingeniería Informática, Perú. Recuperado de <http://tesis.pucp.edu.pe/repositorio/handle/123456789/4892>
- riniga. (2009, 19 de Octubre). El estrés, la enfermedad del siglo XXI. Recuperado de: https://www.youtube.com/watch?v=qXLkeE_cWQk
- RPP. (2015, 30 de enero). El 58% de los peruanos sufre de estrés. Recuperado de: <http://rpp.pe/peru/actualidad/el-58-de-los-peruanos-sufre-de-estres-noticia-764777>
- RUP. (2002). Rational Unified Process. Recuperado de: <http://sce.uhcl.edu/helm/rationalunifiedprocess/>

- Russoniello, C. V., O'Brien, K. & Parks, J. M. (2009). The effectiveness of casual video games in improving mood and decreasing stress. *Journal of CyberTherapy & Rehabilitation*. 2(1), 53-66. Recuperado de: <http://www.ecu.edu/cs-hhp/biofeedback/upload/The-Effectiveness-of-Casual-Video-Games-in-Improving-Mood-and-Decreasing-Stress.pdf>
- Safer Healthcare. (2012). Stress and Fatigue Management. Recuperado de: <http://www.saferhealthcare.com/high-reliability-topics/stress-and-fatigue/>
- Sanitas. (2014). Qué es el estrés. Recuperado de http://www.sanitas.es/sanitas/seguros/es/particulares/biblioteca-de-salud/psicologia/prepro_080026.html
- Schmitz, C. C. & Hipp, E. (2005). *A Leader's Guide to Fighting Tigers. A Stress Management Guide for Teens*. Ciudad de México, México. : Pax México. Recuperado de: <https://books.google.com.pe/books?id=JMv5F8bv11MC&lpq=PA97&dq=el%20vigor%20y%20el%20estrés&hl=es&pg=PA97#v=onepage&q=el%20vigor%20y%20el%20estrés&f=false>
- SCRUM GUIDES. (2013). The Scrum Guide. Recuperado de: <http://www.scrumguides.org/scrum-guide.html#team-po>
- Serrone, C. A. (2012). *Mood Management and Video-Game Engagement: The Importance of User-Experience and Gender in Assessing the Psychological Effects of Video-Game Play*. (Tesis de maestría). San José Estate University, San José, California, Estados Unidos. Recuperado de: http://scholarworks.sjsu.edu/etd_theses/4211
- Servicio técnico de Apple. (2015, 10 de Noviembre). Conoce esta breve historia sobre los cambios que ha experimentado el sistema iOS de Apple [Blog de servicio técnico de Apple]. Recuperado de: <http://servicio-tecnico-apple.es/blog/conoce-esta-breve-historia-sobre-los-cambios-que-ha-experimentado-el-sistema-ios-de-apple/>
- Skreech. (2014, 12 de Junio). How Much Does It Cost To Make an App?. Recuperado de: <https://www.infinitemonkeys.mobi/blog/how-much-does-it-cost-to-make-an-app/>
- Spielberger, C. y Deffenbacher, J. (2015). Estrategias para controlar su enojo. Recuperado de: <http://www.apa.org/centrodeapoyo/estrategias.aspx>

- Stavroula, L., Griffiths A. & Cox, T. (2004). *La organización del trabajo y el estrés*. Nottingham, Reino Unido: Author. Recuperado de: http://www.who.int/occupational_health/publications/stress/es/
- Sublime Text. (2015). Sublime Text [Sitio web oficial]. Recuperado de: <https://www.sublimetext.com/>
- SUM. (2015, 16 de Abril). SUM para Desarrollo de Videojuegos. Recuperado de <http://www.gemserk.com/sum>
- Talbott, S. M. (2012). Vigor, Stress Hormones & Exhaustion. Recuperado de: <http://myemail.constantcontact.com/Vigor--Stress-Hormones-and-Exhaustion.html?soid=1101943418597&aid=tPdAshviDUU>
- Tejeiro Salguero, R., Pelegrina del Rio, M. y Gómez Vallecillo, J. L. (2009). Efectos psicosociales de los videojuegos. *Comunicación*, 1(7), 235-250. Recuperado de http://www.revistacomunicacion.org/pdf/n7/articulos/a16_Efectos_psicosociales_de_los_videojuegos.pdf
- TexturePacker. (2012). TexturePacker - Create Sprite Sheets for your game! [Sitio web oficial]. Recuperado de <https://www.codeandweb.com/texturepacker>
- The American Institute of Stress. (2015). 50 common signs and symptoms of stress. Recuperado de: <http://www.stress.org/stress-effects/>
- Thomas, C. (2011, 06 de Marzo). How much does it cost to develop an app?. Recuperado de: <http://www.bluecloudsolutions.com/blog/cost-develop-app/#>
- Tiled. (2006). Your free, easy to use and flexible tile map editor [Sitio web oficial]. Recuperado de: <http://doc.mapeditor.org/manual/introduction/>
- Unity. (2015). Create and connect with Unity 5 [Sitio web oficial]. Recuperado de: <https://unity3d.com/es>
- Universidad de Nevada. (2014, 02 de Julio). La liberación de Estrés a través del poder de la música. Recuperado de: <http://www.unr.edu/counseling/virtual-relaxation-room/releasing-stress-through-the-power-of-music>
- Universidad Nacional Abierta y a Distancia. (2013a). Lección 3: Tipos de Aplicaciones. Recuperado de http://datateca.unad.edu.co/contenidos/233016/EXE_SAM/leccin_3_tipos_de_aplicaciones.html

- Universidad Nacional Abierta y a Distancia. (2013b). Lección 4: Hardware de Dispositivos Móviles. Recuperado de http://datateca.unad.edu.co/contenidos/233016/EXE_SAM/leccin_4_hardware_de_dispositivos_mviles.html
- Vallejo, J. P. (2014, 10 de Junio). ¿Qué es Game Development o Desarrollo de Videojuegos? [Blog de desarrollo de videojuegos]. Recuperado de: <http://www.pivotstudio.co/blog/que-es-game-development/>
- VirtualBox. (2015). Welcome to VirtualBox.org! [Sitio web oficial]. Recuperado de: <https://www.virtualbox.org/>
- WebConsultas. (2013a, 27 de Enero). Controlar el estrés. Recuperado de: <http://www.webconsultas.com/mente-y-emociones/emociones-y-autoayuda/controlar-el-estres-772>
- WebConsultas. (2013b, 27 de Noviembre). Controlar el estrés. Recuperado de: <http://www.webconsultas.com/mente-y-emociones/emociones-y-autoayuda/sintomas-y-diagnostico-del-estres-3652>
- xCode. (2012). What's new in Xcode 7 [Sitio web oficial]. Recuperado de: <https://developer.apple.com/xcode/>
- Xilisoft. (2015). Xilisoft Video Convertidor [Sitio web oficial]. Recuperado de: <http://www.xilisoft.es/video-converter.html#>

GLOSARIO DE TÉRMINOS

Alfa de Cronbach: coeficiente que mide la fiabilidad o confiabilidad de un test.

Android: sistema operativo perteneciente a Google, funciona sobre diversos tipos de dispositivos.

API (Application Programming Interface - Interfaz de programación de aplicaciones): código fuente ya desarrollado para facilitar la comunicación entre aplicaciones o programas.

Aplicación: programa desarrollado principalmente para funcionar en un dispositivo móvil.

App Store: tienda virtual donde se distribuye aplicaciones y videojuegos desarrollados para dispositivos que usan el sistema operativo iOS.

Benzodiazepinas: fármacos con efecto relajante y antidepresivo.

Bitrate: cantidad de espacio de almacenamiento que ocupa un segundo de audio digital.

Caja blanca: pruebas aplicadas a un software para verificar una correcta estructura interna.

Caja negra: pruebas aplicadas a un software para verificar su correcto funcionamiento.

Códec: algoritmo que codifica y decodifica archivos de audio.

Corona SDK: motor de desarrollo de videojuegos 2D.

CPU (Central Processing Unit - Unidad central de proceso): es la parte de una computadora, tablet o smartphone que se encarga del procesamiento de datos.

Desarrollo multiplataforma: es el desarrollo de una aplicación para que funcione en dos o más sistemas operativos diferentes.

Desarrollo nativo: es el desarrollo particular de una aplicación para un sistema operativo específico.

Escala Likert: escala cuyas preguntas ofrecen respuestas que se pueden cuantificar numéricamente de mayor a menor o viceversa.

Estrés: Reacción biológica natural del ser humano ante una situación real o aparente que pone en riesgo su integridad física o emocional.

FPS (First Person Shooter – Disparo en primera persona): Videojuego donde el jugador por lo general solo ve el arma con la que dispara.

Gama: puede ser alta, media o baja; implica qué capacidad de procesamiento, almacenamiento y calidad de captura de fotos y video tiene un teléfono inteligente.

Game Engine: motor de desarrollo de videojuegos.

Gamer: persona que juega videojuegos de manera frecuente.

Globalización: difusión mundial de tendencias y costumbres vinculadas con las tecnologías de comunicación.

GUP (Game Unified Process - Proceso unificado de videojuego): metodología basada en RUP para el desarrollo de videojuegos.

iOS: sistema operativo desarrollado por Apple para dispositivos móviles.

iPad: tablet fabricada exclusivamente por Apple.

iPhone: teléfono inteligente fabricado exclusivamente por Apple.

iPod: reproductor digital fabricado exclusivamente por Apple.

ISO (International Organization for Standardization - Organismo internacional de estandarización): organismo para crear estándares aceptados internacionalmente.

iTunes: tienda virtual de Apple donde se distribuye contenido digital.

Lua: lenguaje de programación de alto nivel, muy ligero y orientado a procesos.

MMO (Massive Multiplayer Online Game - Juego multijugador masivo en línea): videojuego donde participan varias personas que juegan en tiempo real.

MMORPG (Massive Multiplayer Online Rol Playing Game - Juego de rol multijugador masivo en línea): videojuego específicamente del género de Rol donde juegan varias personas en tiempo real.

MOBA (Multiplayer Online Battle Arena - Arena de combate multijugador online): videojuego donde los jugadores forman equipos para luchar entre sí.

Neuronas espejo: son neuronas que se activan cuando una persona ve un rostro que sonríe, produciendo una reacción que hace sonreír a la persona.

OMS (Organización Mundial de la Salud): organismo encargado de promover la salud a nivel mundial.

Play Store: Conocida también como Google Play, es la tienda virtual donde se distribuye contenido digital para dispositivos que usan el sistema operativo Android.

POMS (Profile of Mood States - Perfil de estados de ánimo): test usado para medir las emociones de la persona.

RAM (Random Access Memory - Memoria de acceso aleatorio): memoria necesaria para la ejecución de un programa.

RPG (Role Playing Game - Juego de rol): videojuego donde el jugador se centra en mejorar sus habilidades para superar los objetivos planteados por el videojuego.

RTS (Real Time Strategy - Estrategia en tiempo real): videojuegos que son jugados en tiempo real con varios usuarios que interactúan entre sí.

RUP (Rational Unified Process – Proceso unificado de desarrollo): metodología para el desarrollo de software.

SCRUM: marco de trabajo para el desarrollo de software complejo.

Smartphone (Teléfono inteligente): dispositivo electrónico más pequeño que una tablet, capaz de ejecutar varias aplicaciones al mismo tiempo y funcionar como dispositivo multimedia.

SO (Operating System - Sistema operativo): es un programa grande que hace que una computadora pueda ejecutar los programas y las tareas que un usuario requiere.

SUM: metodología basada en la metodología Scrum para el desarrollo del videojuegos.

Tablet: computadora portátil y táctil más grande que un smartphone, sin teclado físico.

Unity 3D: motor de desarrollo de videojuegos 2D y 3D.

Videojuego 2D: videojuegos que se ejecutan en entornos de 2 dimensiones.

Videojuego 3D: videojuegos que se ejecutan en entornos de 3 dimensiones.

Videojuego AAA: videojuegos desarrollados por empresas cuyo presupuesto de desarrollo supera los miles de soles o dólares.

Videojuego Casual: videojuego diseñado para todas las personas, fácil de aprender a jugarlo y con la característica de entretener y captar la atención del jugador sin generar emociones fuertes o intensas.

Videojuego Indie: videojuegos desarrollados por una persona, donde no se cuenta con mucho presupuesto.

Videojuego Hardcore: videojuegos cuya lógica y gráficos son complejos y de alta calidad, generalmente son videojuegos 3D.

XGD (Extreme Game Development - Desarrollo de videojuegos extremo): metodología para el desarrollo de videojuegos, está basada en la Programación extrema.

ANEXOS

Anexo nro. 1 “CÓDIGO FUENTE DEL VIDEOJUEGO” (digital), ver el CD.

Anexo nro. 2 “INSTALADORES DEL VIDEOJUEGO” (digital), ver el CD.

**Anexo nro. 3 “DOCUMENTOS DE VALIDACIÓN DEL TEST DE REDUCCIÓN DE
ESTRÉS”**

DOCUMENTO DE VALIDACIÓN DE TEST DE REDUCCIÓN DE ESTRÉS

Yo, Carmen R. Salas Zúñiga, especialista en Psicología Humana, con número de colegiatura C.Ps.C. 8954, doy fe de la validez y cálculo de resultados del test titulado "Ficha de medición de reducción de estrés en las personas", el cual es parte de la Metodología de medición de reducción de estrés en las personas, elaborado por el Bachiller en Ingeniería de Sistemas Iván Marcial Palomino Taype, el cual está basado la prueba psicológica estándar llamada "Perfil de estados de ánimos" (POMS), elaborada por el Doctor en investigación y medicina Douglas McNair (1971), así como en el artículo científico titulado "La eficacia de videojuegos casuales en la mejora del humor y disminución del estrés", elaborado por el Doctor en investigación Carmen Vincent Rusoniello (2009). Por lo descrito paso a firmar y sellar el presente documento:

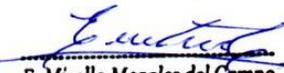
**GOBIERNO REGIONAL APURÍMAC**
HOSPITAL DE ANDAHUAYLAS
.....
Carmen R. Salas Zúñiga
PSICÓLOGA C.P.S.C. 8954
SERVIDOR DEL SERVICIO PSICOLÓGICO

Firma y sello.

Andahuaylas, 24 de Diciembre del 2015.

DOCUMENTO DE VALIDACIÓN DE TEST DE REDUCCIÓN DE ESTRÉS

Yo, Mirella Morales del Campo, especialista en Psicología Humana, con número de colegiatura C.Ps.C. 26573, doy fe de la validez y cálculo de resultados del test titulado "Ficha de medición de reducción de estrés en las personas", el cual es parte de la Metodología de medición de reducción de estrés en las personas, elaborado por el Bachiller en Ingeniería de Sistemas Iván Marcial Palomino Taype, el cual está basado la prueba psicológica estándar llamada "Perfil de estados de ánimos" (POMS), elaborada por el Doctor en investigación y medicina Douglas McNair (1971), así como en el artículo científico titulado "La eficacia de videojuegos casuales en la mejora del humor y disminución del estrés", elaborado por el Doctor en investigación Carmen Vincent Rusoniello (2009). Por lo descrito paso a firmar y sellar el presente documento:


E. Mirella Morales del Campo
Psicóloga
C.Ps.P. 26573
Firma y sello.

Andahuaylas, 30 de Marzo del 2016.

DOCUMENTO DE VALIDACIÓN DE TEST DE REDUCCIÓN DE ESTRÉS

Yo, Efraín Altamirano Morán, especialista en Psicología Humana, con número de colegiatura C.Ps.C. 2523, doy fe de la validez y cálculo de resultados del test titulado "Ficha de medición de reducción de estrés en las personas", el cual es parte de la Metodología de medición de reducción de estrés en las personas, elaborado por el Bachiller en Ingeniería de Sistemas Iván Marcial Palomino Taype, el cual está basado la prueba psicológica estándar llamada "Perfil de estados de ánimos" (POMS), elaborada por el Doctor en investigación y medicina Douglas McNair (1971), así como en el artículo científico titulado "La eficacia de videojuegos casuales en la mejora del humor y disminución del estrés", elaborado por el Doctor en investigación Carmen Vincent Rusoniello (2009). Por lo descrito paso a firmar y sellar el presente documento:

 MINISTERIO DE SALUD

Ps. EFRAIN A. ALTAMIRANO MORAN
PSICOLOGO
C.Ps.P. 2523

Firma y sello.

Andahuaylas, 30 de Marzo del 2016.

Anexo nro. 4 “FICHAS DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS LLENADAS”

Lista de personas que llenaron la “FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS”

Nro.	Nombres y Apellidos
1	Glicerio Pocco Quispe
2	Víctor Alberto Lima Román
3	Ing. Jorge Cotrado Mamani
4	Fredy Román Huachohuilla
5	Florentino Huayllas Alhuay
6	Bequer Brayan Orosco Pahuara
7	Aurelio Vilches Rojas
8	Saturnino Roldan Andrade
9	Vidal Farfán Mitma
10	Reynaldo Taipe Casas
11	Heinz Zúñiga Muños
12	Efraín Huamán Vicente
13	Sandro Peralta Moscoso
14	Pompeyo Edison Rojas Mescoco
15	Jafett Rojas Hurtado
16	Juan Gabriel Condori Torres
17	Bladimir Lenin Palomino Chávez
18	Henry Isaac Gonzales Cruz
19	Hans Doamed Leitson Delgado Ortega
20	Yuver Huamán Ancco
21	Jhancarlo Florencio Silva Ochoa
22	Miguel Ángel Reynaga Calderón
23	Liz Elizabeth Soto de la Cruz
24	Rut Huarhuachi Cóndor
25	Pisc. Carmen Salas Zúñiga

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 10
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 10
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 8
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 8
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 7
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 7
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 7
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

Especificación el manual de juego

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 9
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 9
6. El videojuego elevó su estado de ánimo.
Respuesta (del 0 al 10): 8
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 8
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

Al momento de Iniciar el juego tendria que empezar
directamente ya el juego con las Flechas, y tendria que
haber un boton que diga iniciar juego

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 5
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 8
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 8
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 7
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 7
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 5
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

RANKING DE USUARIOS

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 9
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 10
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 8
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 9
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 10
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 9
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 10

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

Aumenta La Emoción y la capacidad Mental, cada ves
Que Pasas de Nivel.

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

NRO.

5

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 7
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 7
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 9
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 7
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 9
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 7
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 9
11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)
en referencia al juego mi apreciación en lo que se
deberia mejorar es la interfaz del juego algo mas
llamativo, combinar colores y cambio de fondo musical(juego)

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 9
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 8
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 9
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 7
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 10
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 9
10. Se siente bien ahora.
Respuesta (del 0 al 10): 10

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

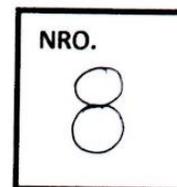
Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 8
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 7
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 8
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 8
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 7
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8
11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS



Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 4
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 5
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 4
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 4
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 6
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 5
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 3
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 4
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 6
10. Se siente bien ahora.
Respuesta (del 0 al 10): 6
11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)
más obstaculos

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 9
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 10
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 9
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 9
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 8
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 10
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 9
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 9
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 9
11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)
El videojuego esta muy bien, solo deria que el fondo
del video juego que sea otro color, sea un color
que le llame atención a la gente

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 7
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 8
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 8
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 7
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 8
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

NRO. / /

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 10
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 10
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 9
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 10
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 5
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 6
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 10
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 6
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 5
10. Se siente bien ahora.
Respuesta (del 0 al 10): 6

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

Pause cuando inicia que no llegue donde esta la pared
sino pararlo antes

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 7
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 8
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 7
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 6
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 7
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 6
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 7
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 7
10. Se siente bien ahora.
Respuesta (del 0 al 10): 7

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

En Algunos niveles te hace pensar para
llegar al objetivo

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 9
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 10
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 9
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 10
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 9
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 8
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 9
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 5
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 7
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 5
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 7
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 6
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 6
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 7
10. Se siente bien ahora.
Respuesta (del 0 al 10): 7

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

La interfaz del juego no es muy amigable, se recomienda
contratar a algun diseñador para mejorarlo.

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.

Respuesta (del 0 al 10): 6

2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).

Respuesta (del 0 al 10): 9

3. Cuanto lo relajó el videojuego.

Respuesta (del 0 al 10): 7

4. Cuan alegre se siente ahora.

Respuesta (del 0 al 10): 7

5. Cuan tranquilo se siente.

Respuesta (del 0 al 10): 5

6. El videojuego elevo su estado de ánimo.

Respuesta (del 0 al 10): 7

7. Cuan desahogado se siente (liberado, descansado).

Respuesta (del 0 al 10): 10

8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).

Respuesta (del 0 al 10): 8

9. Cuan bien se sintió mientras jugaba el videojuego.

Respuesta (del 0 al 10): 7

10. Se siente bien ahora.

Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

La musica es relajante, bien adecuado.

Tuve problemas para iniciar porq' se me hizo confuso las indicaciones.

Trata de mejorar solo la interfaz para q' sea mas llamativo (interfaz al momento q' se juega)

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 6
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 8
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 7
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 7
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 8
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 6
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 9
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 7
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 10
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 10
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 10
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 9
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 9
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 9
10. Se siente bien ahora.
Respuesta (del 0 al 10): 10

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

Proyecto Excelente que necesita salir a un proyecto
mucho mas grande Comunidades de "StarUP"

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 9
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 10
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 7
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 7
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 8
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 7
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 8
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 7
10. Se siente bien ahora.
Respuesta (del 0 al 10): 7

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

→ ME INTERESARIA VER UNA VERSION PARA NIÑOS DE EDUCACION

INICIAL

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 6
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 7
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 4
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 3
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 3
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 4
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 4
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 4
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 3
10. Se siente bien ahora.
Respuesta (del 0 al 10): 5

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 2
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 8
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 10
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 9
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 10
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 6
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 9
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

La carita feliz y el color amarillo me recuerda
a PACMAN

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 7
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 9
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 7
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 5
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 5
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 8
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 7
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 6
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 7

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

Deberia haber mas personajes segun el score acumulado,
la música es buena se relaja, los colores de los
niveles deberia variar, pero si es entretenido.

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 8
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 9
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 7
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 8
6. El videojuego elevó su estado de ánimo.
Respuesta (del 0 al 10): 10
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 9
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 9
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 9
10. Se siente bien ahora.
Respuesta (del 0 al 10): 10

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

MUY BUENO

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 10
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 9
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 10
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 10
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 9
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 9
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 10
10. Se siente bien ahora.
Respuesta (del 0 al 10): 10

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

- Mejorar la musica en cada nivel

- Mejorar los graficos

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 7
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 8
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 7
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 7
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 8
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 8
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 8
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 8
10. Se siente bien ahora.
Respuesta (del 0 al 10): 8

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

o Interfaz buena, Musica Relajante, Sugiriria autoayuda.

FICHA DE MEDICIÓN DE REDUCCIÓN DE ESTRÉS EN LAS PERSONAS

Luego de haber jugado el videojuego, por favor evalúe los siguientes enunciados con un número de la escala del 0 al 10, donde:

0 representa "nada" y

10 representa "fuertemente/muchísimo".

1. Cuanto captó su atención el videojuego mientras lo jugaba.
Respuesta (del 0 al 10): 8
2. Cuanto lo distrajo el videojuego mientras lo jugaba (le hizo olvidar de otros pensamientos, actividades).
Respuesta (del 0 al 10): 8
3. Cuanto lo relajó el videojuego.
Respuesta (del 0 al 10): 8
4. Cuan alegre se siente ahora.
Respuesta (del 0 al 10): 9
5. Cuan tranquilo se siente.
Respuesta (del 0 al 10): 9
6. El videojuego elevo su estado de ánimo.
Respuesta (del 0 al 10): 9
7. Cuan desahogado se siente (liberado, descansado).
Respuesta (del 0 al 10): 9
8. Cuan lúcido se siente (cuanto despejó su mente el videojuego).
Respuesta (del 0 al 10): 9
9. Cuan bien se sintió mientras jugaba el videojuego.
Respuesta (del 0 al 10): 9
10. Se siente bien ahora.
Respuesta (del 0 al 10): 9

11. (OPCIONAL), brinde alguna apreciación del videojuego (que le gusto o que puede mejorar)

Me gusta y puedo recomendar que la música podría ser instrumental con mayor sonido relajante por ejemplo Sanfir, naturaleza, etc.

Anexo nro. 5. "CÁLCULO DEL COEFICIENTE ALFA DE CRONBACH"

La siguiente tabla muestra cómo se calculó el coeficiente alfa de Cronbach para el subgrupo de medios positivos, para el subgrupo de síntomas negativos y para el test de Medición de Reducción de Estrés en general:

PERSONAS	REPUESTAS A LAS PREGUNTA										Suma Medios Positivos	Suma Síntomas Negativos
	1	2	3	4	5	6	7	8	9	10		
1	10	9	0	2	2	7	2	3	7	8	41	9
2	8	9	2	1	1	8	2	2	8	8	41	8
3	8	5	2	2	2	7	3	5	8	8	36	14
4	9	9	0	2	1	10	2	1	8	10	46	6
5	8	7	3	1	3	9	2	3	8	9	41	12
6	8	9	1	2	1	7	2	0	9	10	43	6
7	8	8	2	3	2	8	2	3	8	8	40	12
8	4	5	6	6	4	5	7	6	6	6	26	29
9	9	10	1	1	2	10	1	1	8	9	46	6
10	8	9	2	3	2	8	3	2	8	8	41	12
11	10	10	1	0	5	6	0	4	5	6	37	10
12	7	8	3	4	3	6	2	3	7	7	35	15
13	8	9	1	0	1	10	1	2	8	8	43	5
14	9	9	5	3	5	7	4	4	7	7	39	21
15	6	9	3	3	5	7	0	2	7	8	37	13
16	6	8	3	3	2	6	1	3	8	8	36	12
17	8	9	2	0	0	10	1	1	9	10	46	4
18	9	10	3	3	2	7	2	2	7	7	40	12
19	6	7	6	7	7	4	6	6	3	5	25	32
20	8	8	2	0	1	10	4	1	8	8	42	8

4. Las respuestas de las columnas de color verde fueron cambiadas (se restó a 10 el valor de la respuesta), se hizo ello para que la pregunta refleje lo que se quiere medir puesto que en el test se había cambiado la dirección de la pregunta.
5. En la columna celeste “Suma Medios Positivos”, se sumó todas las respuestas de las columnas azules para cada uno de los 25 encuestados.
6. En la columna verde “Suma síntomas Negativos”, se sumó todas las respuestas de las columnas verdes para cada uno de los 25 encuestados.
7. En la fila “Varianza”, se calculó la varianza para cada una de las 10 columnas de respuestas y también para las columnas “Suma Medios Positivos” y “Suma Síntomas Negativos”.
8. En la intersección de la fila “Suma de Varianzas” y la columna “Medio Positivo”, se encuentra la suma de las varianzas calculadas en el punto 7 pero solo de los resultados celestes (es decir el de las respuestas 1, 2, 6, 9 y 10).
9. En la intersección de la fila “Suma de Varianzas” y la columna “Síntoma Negativo”, se encuentra la suma de las varianzas calculadas en el punto 7 pero solo de los resultados verdes (es decir el de las respuestas 3, 4, 5, 7 y 8).
10. En la intersección de la fila “Factor 1” y la columna “Medio Positivo”, se tiene el resultado de aplicar la formula $k/(k - 1)$, donde $k = 5$ (por las preguntas 1, 2, 6, 9 y 10), con lo que se obtiene 1.25
11. En la intersección de la fila “Factor 1” y la columna “Síntoma Negativo”, se tiene el resultado de aplicar la formula $k/(k - 1)$, donde $k = 5$ (por las preguntas 3, 4, 5, 7 y 8), con lo que se obtiene 1.25
12. En la intersección de la fila “Factor 2” y la columna “Medio Positivo”, se tiene el resultado de aplicar la formula $1 - (\sum oi^2 / oT^2)$, donde $\sum oi^2 = 10.34$ y $\sum oT^2 = 30.41$, con lo que se obtiene 0.65

13. En la intersección de la fila “Factor 2” y la columna “Síntoma Negativo”, se tiene el resultado de aplicar la formula $1 - (\sum oi^2 / oT^2)$, donde $\sum oi^2 = 14.23$ y $\sum oT^2 = 50.61$, con lo que se obtiene 0.71
14. En la intersección de la fila “Alfa de Cronbach” y la columna “Medio Positivo”, se tiene el resultado de multiplicar el resultado del punto 10 con el punto 12, con lo que se obtiene 0.82, el cuál es el valor del alfa de Cronbach para los Medios Positivos.
15. En la intersección de la fila “Alfa de Cronbach” y la columna “Síntoma Negativo”, se tiene el resultado de multiplicar el resultado del punto 11 con el punto 13, con lo que se obtiene 0.89, el cuál es el valor del alfa de Cronbach para los Síntomas Negativos.
16. Finalmente para tener una referencia del valor aproximado de todo el test se sacó la media aritmética de del coeficiente alfa de Cronbach de los Medios Positivos y Síntomas Negativos con lo que se obtuvo un coeficiente de 0.86 dicho resultado se encuentra en las celdas de color amarillo

Anexo nro. 6. "IMÁGENES DEL CONSUMO DE RECURSOS"

Los recursos de hardware del teléfono inteligente que consumen o usan los videojuegos que forman parte del estudio realizado se muestran en la siguiente tabla, dichos datos fueron obtenidos usando la aplicación M2 App Monitor v 2.1.4:

VIDEOJUEGO	RECURSOS: BATERÍA, RAM Y CPU	RECURSOS: ESPACIO DE ALMACENAMIENTO
<p>CASUAL LABYRINTH</p>	 <p>CL Sin clasificar Alertas</p> <p>Bateria 438mW</p> <p>Uso de Datos 0KB/min</p> <p>Memoria 24MB</p> <p>CPU 18%</p>	 <p>CL Sin clasificar Alertas</p> <p>Fallas 0 fallas</p> <p>Permisos 0 marcado</p> <p>Espacio de Almacenamiento 13MB</p>
<p>BEJEWELED 2</p>	 <p>Bejeweled 2 Juegos Alertas</p> <p>Bateria 518mW</p> <p>Uso de Datos 0KB/min</p> <p>Memoria 53MB</p> <p>CPU 36%</p>	 <p>Bejeweled 2 Juegos Alertas</p> <p>Fallas 0 fallas</p> <p>Permisos 0 marcado</p> <p>Espacio de Almacenamiento 40MB</p>

Continuación del anexo "IMÁGENES DEL CONSUMO DE RECURSOS"

VIDEOJUEGO	RECURSOS: BATERÍA, RAM Y CPU	RECURSOS: ESPACIO DE ALMACENAMIENTO
MAGNETANK	 <p>MagneTank Juegos Alertas</p> <p>Administrar</p> <p>Bateria 646mW</p> <p>Uso de Datos 0KB/min</p> <p>Memoria 19MB</p> <p>CPU 100% LG QuickMemo</p>	 <p>MagneTank Juegos Alertas</p> <p>Administrar</p> <p>Fallas 0 fallas</p> <p>Permisos 0 marcado</p> <p>Espacio de Almacenamiento 13MB</p> <p>Acceso a Notificaciones LG QuickMemo</p>
MAZE KING	 <p>Maze King Juegos Alertas</p> <p>Administrar</p> <p>Bateria 469mW</p> <p>Uso de Datos 0KB/min</p> <p>Memoria 37MB</p> <p>CPU 23% LG QuickMemo</p>	 <p>Maze King Juegos Alertas</p> <p>Administrar</p> <p>Fallas 0 fallas</p> <p>Permisos 0 marcado</p> <p>Espacio de Almacenamiento 19MB</p> <p>Acceso a Notificaciones LG QuickMemo</p>

Continuación del anexo "IMÁGENES DEL CONSUMO DE RECURSOS"

VIDEOJUEGO	RECURSOS: BATERÍA, RAM Y CPU	RECURSOS: ESPACIO DE ALMACENAMIENTO
<p>PEGLAND DELUXE</p>	 <p>Pegland Deluxe Juegos Alertas</p> <p>Administrar</p> <p>Batería 721mW</p> <p>Uso de Datos 2KB/min</p> <p>Memoria 11MB</p> <p>CPU 100%</p> <p>LG QuickMemo</p>	 <p>Pegland Deluxe Juegos Alertas</p> <p>Administrar</p> <p>Fallas 0 fallas</p> <p>Permisos 0 marcado</p> <p>Espacio de Almacenamiento 38MB</p> <p>Acceso a Notificaciones</p> <p>LG QuickMemo</p>
<p>WORDWORM ADVENTURES</p>	 <p>WordWorm Adventure Sin clasificar Alertas</p> <p>Administrar</p> <p>Batería 605mW</p> <p>Uso de Datos 0KB/min</p> <p>Memoria 75MB</p> <p>CPU 18%</p> <p>LG QuickMemo</p>	 <p>WordWorm Adventure Sin clasificar Alertas</p> <p>Administrar</p> <p>Fallas 0 fallas</p> <p>Permisos 0 marcado</p> <p>Espacio de Almacenamiento 24MB</p> <p>Acceso a Notificaciones</p> <p>LG QuickMemo</p>