

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**DESARROLLO DE UN JUEGO EN TERCERA DIMENSIÓN
UTILIZANDO UNA METODOLOGÍA FORMAL**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

MARCELO ALEJANDRO AGILA ACURIO

marceloalejandro@gmail.com

ANDRÉS ALEJANDRO CEVALLOS LÓPEZ

andrescevallosl@yahoo.com

DIRECTOR: ING, M.Sc CARLOS ESTALESMIT MONTENEGRO ARMAS

carlos.montengero@epn.edu.ec

Quito, Marzo 2014

DECLARACIÓN

Nosotros, Andrés Alejandro Cevallos López y Marcelo Alejandro Agila Acurio, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedemos nuestros derechos de propiedad intelectual correspondiente a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional.

Andrés Cevallos

Marcelo Agila

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Andrés Alejandro Cevallos López y Marcelo Alejandro Agila Acurio, bajo mi supervisión.

Ing. Carlos Montenegro
DIRECTOR DE PROYECTO

AGRADECIMIENTO

A la vida, por darme la oportunidad de terminar el proyecto de titulación y de rodearme de personas buenas que con su ejemplo han sabido inculcar valores.

A mis padres, que me educaron y que han sido pacientes ante cada uno de los problemas que se nos han presentado, con mucha fuerza y unión.

A mis hermanos, que han sido el motor básico y fundamental, de la familia a la que pertenezco, sin sus muestras de esfuerzo en cada una de sus vidas; esto no habría sido posible.

A mi compañero de tesis Andrés Cevallos, que fue el soñador y mentalizador del proyecto de titulación. Demostrando que los sueños de la niñez son posibles.

Al Ingeniero Carlos Montenegro, por creer en nosotros, que a pesar de que esta tesis se extendió en el tiempo, supo comprender lo difícil que fue su realización.

A toda mi familia, mis tíos(as), primos(as), me han servido de apoyo en muchas ocasiones, me siento orgulloso de ser parte de su familia.

A mis amig@s de la poli, muchas gracias por todo el apoyo, por todos los chistes incluso sarcásticos que nos motivaron más a continuar con este proyecto de titulación.

Marcelo

DEDICATORIA

Dedico este trabajo a mi familia, mis padres Hernán y Martha; a mis hermano Edison, y Jessenia. Especialmente a ti Jessy, todo tu esfuerzo ha sido un ejemplo para todos nosotros, gracias por ser mi ñaña, sé que pronto estarás mejor.

Marcelo

AGRADECIMIENTO

A mis tíos que me han acogido en su hogar y me han apoyando constantemente para terminar mi carrera.

A mi novia con la cual nos hemos desvelado y siempre supo darme ideas y su apoyo.

A mi compañero de tesis con el que compartimos el trabajo en el proyecto.

A mi amigos y familiares que siempre estuvieron ahí cuando más los necesite.

A mi madre que siempre me ha cuidado aunque ya no se encuentre conmigo.

Agradezco a demás a todas las personas que colaboraron con nuestro proyecto que nos supieron ayudar y orientar para que podamos realizarlo

Andrés

DEDICATORIA

Este trabajo es dedicado a la memoria de mi madre que supo formarme como persona para poder convertirme en un profesional, a mis tíos que sin su apoyo no hubiera podido salir adelante y a mi novia que me ha acompañado en este tiempo.

Andrés

CONTENIDO

1.	CAPÍTULO 1: PLANTEAMIENTO DEL PROBLEMA	1
	1.1 DESCRIPCIÓN DE AVANCES DE LA INDUSTRIA DE LOS VIDEOJUEGOS Y SU SITUACIÓN EN EL PAÍS	1
	1.1.1 HISTORIA DEL DESARROLLO DE LOS VIDEOJUEGOS	1
	1.1.2 PRINCIPALES AVANCES EN LOS ÚLTIMOS AÑOS	13
	1.1.3 INDUSTRIA DESARROLLADA EN EL PAÍS	19
	1.2 SELECCIÓN DE LA METODOLOGÍA PARA EL DESARROLLO DE VIDEOJUEGOS	21
	1.2.1 DESCRIPCIÓN DE LAS METODOLOGÍAS ÁGILES	21
	1.2.2 DISEÑO DE CRITERIOS DE EVALUACIÓN	32
	1.2.3 EVALUACIÓN Y SELECCIÓN DE LAS METODOLOGÍAS	33
	1.3 SELECCIÓN DE HERRAMIENTAS PARA EL DESARROLLO DE VIDEOJUEGOS	36
	1.3.1 AMBIENTE PARA EL DESARROLLO DE VIDEOJUEGOS	36
	1.3.2 EVALUACIÓN Y SELECCIÓN DE HERRAMIENTAS	49
2.	CAPÍTULO 2: FASE DE PREPRODUCCIÓN	56
	2.1 CONCEPTUALIZACIÓN DEL VIDEOJUEGO	56
	2.1.1 REQUERIMIENTOS FUNCIONALES DEL SISTEMA	56
	2.1.2 PRIMER ENTREGABLE R1 - PRIMER SPRINT	57
	2.1.3 SELECCIÓN DEL TIPO DE VIDEOJUEGO	68
	2.1.4 HISTORIA DEL VIDEOJUEGO	70
	2.1.5 CREACIÓN DEL GUIÓN DEL VIDEOJUEGO	75
	2.2 DISEÑO DE LAS ARTES GRÁFICAS	79
	2.2.1 DISEÑO DE INTERFACES	79
	2.2.2 CREACIÓN DE LOS PERSONAJES	81
	2.2.3 CREACIÓN DEL AMBIENTE DEL VIDEOJUEGO	82
	2.3 HISTORIAS DE USUARIO	82
	2.3.1 PRIMER ENTREGABLE R1-SEGUNDO SPRINT	83
	2.3.2 DIAGRAMA DE CLASES	87
	2.3.3 HISTORIAS DE USUARIO	92
3.	CAPÍTULO 3: FASE DE PRODUCCIÓN	98
	3.1 CONSTRUCCIÓN DE AMBIENTES Y PERSONAJES	98
	3.1.1 PRIMER ENTREGABLE R1-SEGUNDO SPRINT-CONTINUACIÓN	98
	3.1.2 PRIMER ENTREGABLE R1-TERCER SPRINT	102
	3.1.3 SEGUNDO ENTREGABLE R2-PRIMER SPRINT	105
	3.1.4 SEGUNDO ENTREGABLE R2-SEGUNDO SPRINT	107
	3.1.5 SEGUNDO ENTREGABLE R2-TERCER SPRINT	109
	3.1.6 TERCER ENTREGABLE R3-PRIMER SPRINT	115
	3.1.7 TERCER ENTREGABLE R3-SEGUNDO SPRINT	117
	3.1.8 TERCER ENTREGABLE R3-TERCER SPRINT	118
	3.1.9 CONSTRUCCION DE PERSONAJES	120
	3.1.10 CONSTRUCCION DE ASSETS	124
	3.1.11 CONSTRUCCION DEL TERRENO	126
	3.1.12 INCORPORACION DE SONIDOS	129
	3.1.13 ESCENAS DE CORTE	130
	3.1.14 TEXTURIZADO DE PERSONAJES Y OBJETOS	133
	3.2 DISEÑO DE INTERFACES	134
	3.2.1 CONSTRUCCIÓN DE MENUS	135

3.2.2	CONSTRUCCION INTERFAZ DE JUEGO	137
3.3	MODELADO Y ANIMACIÓN	141
3.3.1	MODELADO DE ARMATURES Y RIGGING	141
3.3.2	PROCESO DE ANIMACIÓN	144
3.4	PROGRAMACIÓN DEL JUEGO	148
3.4.1	GAMEOBJECT PERSONAJE	148
3.4.2	GAMEOBJECT CÁMARA.	149
3.4.3	GAMEOBJECT ENEMIGO	150
3.4.4	GAMEOBJECT PICKUP	151
3.4.5	GAMEOBJECT LEVELSTATUS	151
3.4.6	GAMEOBJECT PAUSE MANAGER	151
3.4.7	GAMEOBJECT MISCELANEOS	151
3.4.8	GAMEOBJECT DIALOGS	151
3.4.9	UNITYSERIALIZER	152
3.4.10	SCENE MANAGER	152
3.4.11	PRUEBAS	152
4.	CAPÍTULO 4: FASE DE POSTPRODUCCIÓN	158
4.1	DISEÑO DE CRITERIOS DE CALIDAD Y EVALUACIÓN	158
4.1.1	TERCER ENTREGABLE R3 - TERCER SPRINT CONTINUACIÓN	158
4.1.2	EVALUACIÓN	160
4.1.3	CRITERIOS DE CALIDAD	164
4.1.4	FINALIZACIÓN DEL ENTREGABLE R3 RESULTADOS DE LA METODOLOGÍA	168
4.2	DETALLE DE DISTRIBUCIÓN DEL VIDEOJUEGO	171
4.2.1	BLOG VIDEOJUEGO	171
4.2.2	DISTRIBUCIÓN	172
4.3	RECOLECCIÓN DE INFORMACIÓN DE ENCUESTAS DE JUGABILIDAD	172
4.4	ANÁLISIS DE RESULTADOS	180
5.	CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES	181
5.1	CONCLUSIONES	181
5.2	RECOMENDACIONES	183
	GLOSARIO	184
	BIBLIOGRAFÍA	187
	ANEXOS	191

ÍNDICE DE FIGURAS

FIGURA 1.1 EVOLUCIÓN DE LOS LARGOS CICLOS DE DESARROLLO EN CASCADA (A) A CICLOS ITERATIVOS MÁS CORTOS (B) Y LA MEZCLA QUE HACE XP.	24
FIGURA 1.2 CICLO DE VIDA DEL PROCESO UNIFICADO ÁGIL (AUP)	26
FIGURA 1.3 DISEÑO GENERAL DE SCRUM	27
FIGURA 1.4 EL CICLO DE UN SPRINT	28
FIGURA 1.5 FAMILIA DE CRYSTAL METHODS	29
FIGURA 1.6 CICLOS ANIDADOS DE CRYSTAL CLEAR	30
FIGURA 1.7 CICLOS DE FDD.....	32
FIGURA 1.8 CICLO DEL SPRINT-INCREMENTO DE FUNCIONALIDAD	34
FIGURA 1.9 FLUJO DE SCRUM.....	35
FIGURA 1.10 EJEMPLO DE PRODUCT BACKLOG.....	36
FIGURA 2.1 R1-PRIMER SPRINT TAREAS	57
FIGURA 2.2 R1-PRIMER SPRINT CARACTERÍSTICAS	58
FIGURA 2.3 DETALLE DE ROLES.....	58
FIGURA 2.4 TAREA CREACIÓN HISTORIA DEL VIDEOJUEGO	59
FIGURA 2.5 TAREA DISEÑO INTERFAZ PRESENTACIÓN	60
FIGURA 2.6 TAREA DISEÑO DE INTERFAZ PAUSA DE JUEGO	61
FIGURA 2.7 TAREA DISEÑO DE INTERFAZ INICIO	62
FIGURA 2.8 TAREA DISEÑO DE INTERFAZ JUEGO	63
FIGURA 2.9 TAREA CREACIÓN DEL GUIÓN	64
FIGURA 2.10 TAREA CREACIÓN DE PERSONAJES	65
FIGURA 2.11 TAREA MODELADO DE AMBIENTES.....	66
FIGURA 2.12 TAREA SELECCIONAR TIPO DE VIDEOJUEGO	67
FIGURA 2.13 BOCETOS INTERFAZ PRESENTACIÓN.....	80
FIGURA 2.14 BOCETO NAVEGACIÓN MENÚ PRINCIPAL	80
FIGURA 2.15 BOCETO INTERFAZ DE PAUSA.....	81
FIGURA 2.16 BOCETO DE LA INTERFAZ DE JUEGO.....	81
FIGURA 2.17 BOCETOS PERSONAJES VIDEOJUEGO	82
FIGURA 2.18 BOCETO DEL TERRENO	82
FIGURA 2.19 R1-SEGUNDO SPRINT TAREAS.....	83
FIGURA 2.20 TAREAS R1-SEGUNDO SPRINT CAPÍTULO 2	83
FIGURA 2.21 CARACTERÍSTICAS R1-SEGUNDO SPRINT	84
FIGURA 2.22 TAREA DISEÑO DE CLASES	84
FIGURA 2.23 TAREA DEFINIR ATRIBUTOS	85
FIGURA 2.24 TAREA DEFINIR RELACIONES ENTRE CLASES	85
FIGURA 2.25 TAREA CREACIÓN DE HISTORIAS DE USUARIO	86
FIGURA 2.26 DIAGRAMA DE PAQUETES AZRAEL AWAKENING. FUENTE: ASTAH	87
FIGURA 2.28 DIAGRAMA DE CLASES PAQUETE CAMERA. FUENTE: ASTAH	88
FIGURA 2.27 DIAGRAMA DE CLASES PAQUETE PICKUP. FUENTE: ASTAH	88
FIGURA 2.29 DIAGRAMA DE CLASES PAQUETE PLAYER FUENTE: ASTAH	89
FIGURA 2.30 DIAGRAMA DE CLASES PAQUETE MISCELANEOS. FUENTE: ASTAH	90
FIGURA 2.31 DIAGRAMA DE CLASES PAQUETE ENEMY. FUENTE: ASTAH	91
FIGURA 3.1 R1-SEGUNDO SPRINT TAREAS	99
FIGURA 3.2 TAREA MODELADO DE PERSONAJES	99

FIGURA 3.3 TAREA MODELADO DE PERSONAJES LOWPOLY	100
FIGURA 3.4 TAREA MODELADO DE ASSETS PARA ESCENARIOS	101
FIGURA 3.5 TAREA TEXTURIZADO DE ASSETS	102
FIGURA 3.6 R1-TERCER SPRINT TAREAS	102
FIGURA 3.7 R1-TERCER SPRINT CARACTERÍSTICAS	103
FIGURA 3.8 TAREA ANIMACIÓN	103
FIGURA 3.9 CONSTRUCCIÓN DEL ESCENARIO	104
FIGURA 3.10 R2-PRIMER SPRINT TAREAS	105
FIGURA 3.11 TAREA LÓGICA DE CÁMARA DEL PERSONAJE	106
FIGURA 3.12 TAREA LÓGICA DE MOVIMIENTOS PERSONAJE	106
FIGURA 3.13 R2-SEGUNDO SPRINT TAREAS	107
FIGURA 3.14 TAREA LÓGICA MOVIMIENTO ENEMIGO	108
FIGURA 3.15 TAREA LÓGICA DE ATAQUE	108
FIGURA 3.16 TAREAS RECURRENTES R2 -TERCER SPRINT	109
FIGURA 3.17 R2-TERCER SPRINT TAREAS	109
FIGURA 3.18 R2-TERCER SPRINT CARACTERÍSTICAS	110
FIGURA 3.19 TAREA CONSTRUCCIÓN MENÚ PRINCIPAL	110
FIGURA 3.20 TAREA CONSTRUCCIÓN MENÚ OPCIÓN NUEVO.....	111
FIGURA 3.21 TAREA CONSTRUCCIÓN DEL MENÚ OPCIÓN CONTINUAR	111
FIGURA 3.22 TAREA CONSTRUCCIÓN DEL MENÚ OPCIÓN CRÉDITOS.....	112
FIGURA 3.23 TAREA DISEÑAR ESQUEMA MÓDULOS VIDEOJUEGO.....	113
FIGURA 3.24 TAREA PRUEBA DE UNIDAD.....	113
FIGURA 3.25 TAREA PRUEBA DE INTEGRACIÓN.....	114
FIGURA 3.26 TAREA PRUEBA DE REGRESIÓN.....	115
FIGURA 3.27 TAREAS RECURRENTES R3 -PRIMER SPRINT	115
FIGURA 3.28 R3-PRIMER SPRINT TAREAS	116
FIGURA 3.29 TAREA CUTSCENELVL1	116
FIGURA 3.30 TAREA DEPURACIÓN ESCENARIO LVL 1	117
FIGURA 3.31 TAREAS RECURRENTES R3 -SEGUNDO SPRINT	117
FIGURA 3.32 R3-SEGUNDO SPRINT TAREAS.....	117
FIGURA 3.33 TAREAS RECURRENTES R3 -TERCER SPRINT.....	118
FIGURA 3.34 R3-TERCER SPRINT TAREAS	118
FIGURA 3.35 TAREAS R3-TERCER SPRINT CAPITULO 3	119
FIGURA 3.36 TAREA CREACIÓN DE ESCENA INICIAL.....	119
FIGURA 3.37 TAREA CREACIÓN DE ESCENA FINAL	120
FIGURA 3.38 CONSTRUCCIÓN DE PERSONAJES MODELO BASE.....	121
FIGURA 3.39 CONSTRUCCIÓN DE PERSONAJES MODELO DE ALTOS POLÍGONOS.....	121
FIGURA 3.40 CONSTRUCCIÓN DE PERSONAJES AJUSTE DE VÉRTICES	122
FIGURA 3.41 CONSTRUCCIÓN DE PERSONAJES MODELO DE BAJOS POLÍGONOS.....	122
FIGURA 3.42 CONSTRUCCIÓN DE PERSONAJES SEGMENTACIÓN MODELO	123
FIGURA 3.43 CONSTRUCCIÓN DE PERSONAJES SEGMENTACIÓN MODELO PUNTOS DE CORTE	123
FIGURA 3.44 CONSTRUCCIÓN DE PERSONAJES UNWRAPPING	124
FIGURA 3.45 MODELADO DE ASSETS.....	124
FIGURA 3.46 MODELADO DE ASSETS DETALLES ADICIONALES	125
FIGURA 3.47 UNWRAPPING DE ASSETS	125

FIGURA 3.48	TEXTURIZADO DE ASSETS	126
FIGURA 3.49	CONSTRUCCIÓN DEL TERRENO	126
FIGURA 3.50	VISTA SUPERIOR TERRENO CONSTRUIDO	127
FIGURA 3.51	AMBIENTACIÓN, USO DE SKYBOX.....	127
FIGURA 3.52	AMBIENTACIÓN, USO DE LUCES.....	128
FIGURA 3.53	AMBIENTACIÓN, USO DE EFECTOS-NEBLINA	128
FIGURA 3.54	AMBIENTACIÓN, USO DE EFECTOS-LLUVIA.....	128
FIGURA 3.55	INCORPORACIÓN DE SONIDOS	129
FIGURA 3.56	INCORPORACIÓN SONIDOS FONDO	129
FIGURA 3.57	CONSTRUCCIÓN DE DIÁLOGO ESCENA DE CORTE	131
FIGURA 3.58	REPRODUCCIÓN DE DIÁLOGO	131
FIGURA 3.59	REPRODUCCIÓN DE DIÁLOGO OBJETIVO COMPLETO	131
FIGURA 3.60	REPRODUCCIÓN DIÁLOGO AUTOPLAY.....	132
FIGURA 3.61	CAMINO DE ESCENA	132
FIGURA 3.62	CONFIGURACIÓN ITWEENEVENTS	133
FIGURA 3.63	CONFIGURACIÓN DEL PROCESO DE BAKING	133
FIGURA 3.64	IMAGEN MAPEADA DEL MODELO	134
FIGURA 3.65	IMAGEN MAPEADA CON DETALLES GEOMÉTRICOS DEL MODELO	134
FIGURA 3.66	DISEÑO DEL MENÚ PRINCIPAL	135
FIGURA 3.67	ASIGNACIÓN DEL BOTÓN NEW GAME A LA PRIMERA ESCENA DEL JUEGO.	135
FIGURA 3.68	ASIGNACIÓN DEL BOTÓN CONTINUE A LA ÚLTIMA SESIÓN GUARDADA	135
FIGURA 3.69	RESULTADO DE EJECUCIÓN DEL BOTÓN OPTIONS	136
FIGURA 3.70	ESCENA GAME OVER	136
FIGURA 3.71	INTEGRACIÓN MENÚ OPCIONES Y GAME OVER CON EL VIDEOJUEGO.....	137
FIGURA 3.72	PAUSA DEL VIDEOJUEGO	137
FIGURA 3.73	UBICACIÓN DE CÁMARA RESPECTO AL PERSONAJE	138
FIGURA 3.74	AJUSTE DE ÁNGULOS MÁXIMOS Y MÍNIMOS.....	138
FIGURA 3.75	AJUSTE DE VISUALIZACIÓN	138
FIGURA 3.76	INDICADOR DE SALUD	139
FIGURA 3.77	INTEGRACIÓN DE INDICADOR DE SALUD EN LA INTERFAZ IN-GAME	139
FIGURA 3.78	OBJETO INDICADOR DE OBJETIVO.....	140
FIGURA 3.79	INTEGRACIÓN DE OBJETO INDICADOR A LA INTERFAZ IN-GAME	140
FIGURA 3.80	AJUSTE CÁMARA MINIMAPA	141
FIGURA 3.81	INTEGRACIÓN DEL MÍNIMAPA EN LA INTERFAZ IN-GAME	141
FIGURA 3.82	ESTRUCTURA HUMAN RIGGING.....	142
FIGURA 3.83	DISEÑO DEL ARMATURE, POSICIONAMIENTO Y CORRECCIÓN DE EJES SOBRE EL MODELO	142
FIGURA 3.84	ESTRUCTURA DE HUESOS SOBRE VARIOS OBJETOS	143
FIGURA 3.85	ASIGNACIÓN DE PESOS DE LA ESTRUCTURA DE HUESOS SOBRE EL MODELO	143
FIGURA 3.86	EJEMPLO DE MOVIMIENTOS CON PESOS AUTOMÁTICOS	144
FIGURA 3.87	AJUSTE DE VÉRTICES CON WEIGHT PAINTING	145
FIGURA 3.88	HUESOS AUXILIARES CON WEIGHT PAINTING	145
FIGURA 3.89	DISEÑO DE HUESOS PARA EL CONTROL DE ARTICULACIONES, TOBILLO, RODILLA	145
FIGURA 3.90	CONSTRAINTS PARA CONTROL DE MOVIMIENTO DE LA MANO	146
FIGURA 3.91	USO DE CONSTRAINT IK PARA LA RODILLA.....	146

FIGURA 3.92 SECUENCIA DE ANIMACIÓN RECIBIRDAÑO	147
FIGURA 3.93 ANIMACIÓN CICLO SECUENCIAL	147
FIGURA 3.94 ESQUEMA DE MÓDULOS DEL VIDEOJUEGO. FUENTE: MICROSOFT OFFICE VISIO	155
FIGURA 4.1 R3-TERCER SPRINT TAREAS.....	158
FIGURA 4.2 TAREA PRUEBAS ALFA	159
FIGURA 4.3 TAREA PRUEBAS BETA	160
FIGURA 4.4 DISEÑO DE ENCUESTA.....	167
FIGURA 4.5 CARACTERÍSTICAS VIDEOJUEGO ESFUERZO EN PUNTOS ESTIMADOS.....	168
FIGURA 4.6 DIAGRAMA BURNUP DEL VIDEOJUEGO EN PUNTOS. PUNTOS VS ITERACIONES [8]	169
FIGURA 4.7 FLUJO ACUMULADO VIDEOJUEGO ITERACIONES VS NRO. DE HISTORIAS	170
FIGURA 4.8 CAPTURA BLOG VIDEOJUEGO	171
FIGURA 4.9 RESULTADOS PREGUNTAS SIMPLICIDAD. FUENTE: GOOGLE DOCS	173
FIGURA 4.10 RESULTADOS PREGUNTAS COHERENCIA. FUENTE: GOOGLE DOCS	174
FIGURA 4.11 RESULTADOS PREGUNTAS CLARIDAD. FUENTE: GOOGLE DOCS	175
FIGURA 4.12 RESULTADOS PREGUNTAS ADAPTABILIDAD. FUENTE: GOOGLE DOCS.....	176
FIGURA 4.13 RESULTADOS PREGUNTAS JUGABILIDAD. FUENTE: GOOGLE DOCS.....	177
FIGURA 4.14 RESULTADOS PREGUNTAS EFECTOS Y SONIDO. FUENTE: GOOGLE DOCS	178
FIGURA 4.15 RESULTADOS PREGUNTAS GRÁFICAS. FUENTE: GOOGLE DOCS.....	179

ÍNDICE DE TABLAS

TABLA 1.1 INICIOS DE LA HISTORIA DE LOS VIDEOJUEGOS.....	1
TABLA 1.2 HISTORIA DE LOS VIDEOJUEGOS DÉCADA DEL 70.	3
TABLA 1.3 HISTORIA DE LOS VIDEOJUEGOS DÉCADA DEL 80.	7
TABLA 1.4 HISTORIA DE LOS VIDEOJUEGOS DÉCADA DEL 90.	10
TABLA 1.5 HISTORIA DE LOS VIDEOJUEGOS ACTUALIDAD	13
TABLA 1.6 USO DE POLÍGONOS EN LOS VIDEOJUEGOS.....	16
TABLA 1.7 CONSOLAS DE SOBREMESA 1.	17
TABLA 1.8 DIFERENCIAS ENTRE METODOLOGÍAS TRADICIONALES Y ÁGILES	23
TABLA 1.9 CRITERIOS DE EVALUACIÓN PARA LAS METODOLOGÍAS ÁGILES.....	33
TABLA 1.10 . EVALUACIÓN Y SELECCIÓN DE METODOLOGÍAS	33
TABLA 1.11 CARACTERÍSTICAS DE UNREAL ENGINE 3.	38
TABLA 1.12 CARACTERÍSTICAS DE GAMEBRYO.....	39
TABLA 1.13 CARACTERÍSTICAS DE CRYENGINE 3.....	40
TABLA 1.14 CARACTERÍSTICAS DE UNITY.....	41
TABLA 1.15 CARACTERÍSTICAS DE BLITZTECH	42
TABLA 1.16 CRITERIOS DE EVALUACIÓN DE LOS MOTORES GRÁFICOS.....	50
TABLA 1.17 EVALUACIÓN Y SELECCIÓN DE LOS MOTORES GRÁFICOS	51
TABLA 1.18 CRITERIOS DE EVALUACIÓN DE LAS HERRAMIENTAS DE MODELADO Y ANIMACIÓN 3D.....	52
TABLA 1.19 EVALUACIÓN Y SELECCIÓN DE LAS HERRAMIENTAS DE MODELADO.....	52
TABLA 1.20 CRITERIOS DE EVALUACIÓN DE LAS HERRAMIENTAS DE DISEÑO 2D.	53
TABLA 1.21 EVALUACIÓN Y SELECCIÓN DE LAS DE LAS HERRAMIENTAS DE DISEÑO 2D.	54
TABLA 2.1 REQUERIMIENTOS FUNCIONALES.....	57
TABLA 2.2 DETALLE DE PRINCIPALES CARACTERÍSTICAS R1	58
TABLA 2.3 SUBTAREAS CREACIÓN HISTORIA DEL VIDEOJUEGO	59
TABLA 2.4 SUBTAREAS DISEÑO DE INTERFAZ PRESENTACIÓN	60
TABLA 2.5 SUBTAREAS DISEÑO DE INTERFAZ PAUSA DE JUEGO	61
TABLA 2.6 SUBTAREAS DISEÑO DE INTERFAZ INICIO	62
TABLA 2.7 SUBTAREAS DISEÑO DE INTERFAZ DE JUEGO	63
TABLA 2.8 SUBTAREAS CREACIÓN DEL GUIÓN	64
TABLA 2.9 SUBTAREAS CREACIÓN DE PERSONAJES	65
TABLA 2.10 SUBTAREAS MODELADO DE AMBIENTES.....	66
TABLA 2.11 . SUBTAREAS SELECCIONAR TIPO DE VIDEOJUEGO	67
TABLA 2.12 CLASIFICACIÓN DE LOS VIDEOJUEGOS	69
TABLA 2.13 CARACTERÍSTICA AÑADIDA R1-SEGUNDO SPRINT.....	84
TABLA 2.14 SUBTAREAS DISEÑO DE CLASES	84
TABLA 2.15 SUBTAREAS DEFINIR ATRIBUTOS Y OPERACIONES.....	85
TABLA 2.16 SUBTAREAS DEFINIR RELACIONES ENTRE CLASES	86
TABLA 2.17 SUBTAREAS CREACIÓN DE HISTORIAS DE USUARIO	86
TABLA 2.18 HISTORIA USUARIO1	92
TABLA 2.19 HISTORIA USUARIO2	92
TABLA 2.20 HISTORIA USUARIO3	93
TABLA 2.21 HISTORIA USUARIO4	93
TABLA 2.22 HISTORIA USUARIO5	93

TABLA 2.23 HISTORIA USUARIO6	94
TABLA 2.24 HISTORIA USUARIO7	94
TABLA 2.25 HISTORIA USUARIO8	94
TABLA 2.26 HISTORIA USUARIO9	95
TABLA 2.27 HISTORIA USUARIO10	95
TABLA 2.28 HISTORIA USUARIO11	95
TABLA 2.29 HISTORIA USUARIO12	96
TABLA 2.30 HISTORIA USUARIO13	96
TABLA 2.31 HISTORIA USUARIO14	96
TABLA 2.32 HISTORIA USUARIO15	97
TABLA 2.33 HISTORIA USUARIO16	97
TABLA 3.1 SUBTAREAS MODELADO DE PERSONAJES	99
TABLA 3.2 SUBTAREA MODELADO DE PERSONAJES LOWPOLY	100
TABLA 3.3 SUBTAREAS MODELADO DE ASSETS PARA ESCENARIOS	101
TABLA 3.4 SUBTAREA TEXTURIZADO DE ASSETS	102
TABLA 3.5 DETALLE PRINCIPALES CARACTERÍSTICAS R1-TERCER SPRINT	103
TABLA 3.6 SUBTAREAS ANIMACIÓN	104
TABLA 3.7 SUBTAREAS CONSTRUCCIÓN DEL ESCENARIO	105
TABLA 3.8 SUBTAREAS LÓGICA DE CÁMARA DEL PERSONAJE	106
TABLA 3.9 SUBTAREAS LÓGICA DE MOVIMIENTOS PERSONAJE	107
TABLA 3.10 SUBTAREAS LÓGICA MOVIMIENTO ENEMIGO	108
TABLA 3.11 SUBTAREAS LÓGICA DE ATAQUE	109
TABLA 3.12 DETALLE PRINCIPALES CARACTERÍSTICAS R2- TERCER SPRINT	110
TABLA 3.13 SUBTAREAS CONSTRUCCIÓN MENÚ PRINCIPAL	110
TABLA 3.14 SUBTAREAS CONSTRUCCIÓN MENÚ OPCIÓN NUEVO	111
TABLA 3.15 SUBTAREAS CONSTRUCCIÓN DEL MENÚ OPCIÓN CONTINUAR	112
TABLA 3.16 SUBTAREAS CONSTRUCCIÓN DEL MENÚ OPCIÓN CRÉDITOS	112
TABLA 3.17 SUBTAREAS DISEÑAR ESQUEMA MÓDULOS VIDEOJUEGO	113
TABLA 3.18 SUBTAREAS PRUEBA DE UNIDAD	114
TABLA 3.19 SUBTAREAS PRUEBA DE INTEGRACIÓN	114
TABLA 3.20 SUBTAREAS PRUEBA DE REGRESIÓN	115
TABLA 3.21 SUBTAREAS CUTSCENELVL 1	116
TABLA 3.22 SUBTAREAS DEPURACIÓN ESCENARIO LVL 1	117
TABLA 3.23 SUBTAREAS CREACIÓN DE ESCENA INICIAL	119
TABLA 3.24 SUBTAREAS CREACIÓN DE ESCENA FINAL	120
TABLA 3.25 EJEMPLO PRUEBA DE UNIDAD THIRDPERSONCONTROLLER	153
TABLA 3.26 EJEMPLO PRUEBA DE INTEGRACIÓN ENEMY.JS	156
TABLA 4.1 SUBTAREAS PRUEBAS ALFA	159
TABLA 4.2 SUBTAREAS PRUEBAS BETA	160
TABLA 4.3 PRUEBAS DE RESISTENCIA NIVELES VIDEOJUEGO	161
TABLA 4.4 PRUEBAS DE DESEMPEÑO NIVELES VIDEOJUEGO	162
TABLA 4.5 PRUEBA DE INTERFAZ MENU.JS	163
TABLA 4.6 PESOS DE CRITERIOS DE CALIDAD	166
TABLA 4.7 EVALUACIÓN RESULTADOS ENCUESTA	180

RESUMEN

En el presente proyecto de titulación se investiga y se aplica una metodología que pueda ser usada para el desarrollo de los videojuegos en tercera dimensión.

Se realiza una selección de metodologías de desarrollo de software y herramientas que puedan ser aplicadas al proyecto.

Se definen requerimientos para identificar el alcance del videojuego.

En la etapa de Pre-producción se define la historia, el diseño del personaje, diseño de escenarios y se establece la lógica del videojuego, mediante un diagrama de clases.

En la etapa de Producción, usando las herramientas seleccionadas, se construye los componentes multimedia que se integran dentro del motor gráfico. Usando clases en Javascript y C# para brindar lógica a la aplicación.

Finalmente en la etapa de Post-producción, se realiza pruebas para validar el funcionamiento del producto, debe cumplir los requerimientos planteados.

PRESENTACIÓN

En el capítulo 1 se revisa antecedentes de la industria de los videojuegos, en el mundo y en nuestro país. Se hace un estudio de las metodologías de desarrollo que puedan aplicarse para la construcción de un videojuego, seleccionando la metodología más adecuada para su desarrollo. Se continúa realizando la selección de herramientas gráficas y motores de videojuegos.

En el capítulo 2 se inicia el desarrollo en base a la metodología seleccionada. Se realiza el diseño de la aplicación, definiendo el perfil de los personajes así como un guión conteniendo la historia del videojuego. Continúa con los bocetos de los personajes y de los escenarios. Finalmente se hace un diseño de la lógica del videojuego mediante un diagrama de clases.

En el capítulo 3 se inicia la construcción del videojuego, empezando por los paquetes multimedia. Se describe el procedimiento de construcción de personajes assets, escenarios y la animación de los personajes. En este capítulo se añade la funcionalidad de las clases que conforman el videojuego.

En el capítulo 4 se definen criterios de calidad, el método de evaluación del producto sobre estos criterios; además de un resumen de las pruebas efectuadas en el proceso de desarrollo de la aplicación.

En el capítulo 5 finalmente se realiza el análisis de resultados, se tabulan las encuestas seleccionando la opción más votada y se procede a realizar promedios sobre los criterios de calidad.

1. CAPÍTULO 1: PLANTEAMIENTO DEL PROBLEMA

1.1 DESCRIPCIÓN DE AVANCES DE LA INDUSTRIA DE LOS VIDEOJUEGOS Y SU SITUACIÓN EN EL PAÍS

La industria de los videojuegos no se ha desarrollado en nuestro entorno, debido a varias causas, la principal es que en la actualidad no existe apoyo para elaborar este tipo específico de software. Esto sumado a la piratería ha puesto un freno para consolidar bases sólidas y desarrollar el crecimiento de la industria [1]. Como se mostrará con más detalle en el presente documento, existen empresas ecuatorianas que se dedican al desarrollo de videojuegos. Cabe recalcar que no hay una metodología formal que haya sido creada con este fin específico, pero existen varias metodologías ágiles que se pueden acoplar las características del videojuego a desarrollar.

1.1.1 HISTORIA DEL DESARROLLO DE LOS VIDEOJUEGOS

1.1.1.1 Sus Inicios. [2] [3] [4] [5] [6] [7]

El inicio de los videojuegos nace de la idea de un ingeniero eléctrico, para entretener a las personas que visitaban la empresa en la que trabajaba. Los primeros acercamientos hacia lo que hoy se conoce como videojuego, se obtuvieron de tesis de grado y pruebas sobre la capacidad que podían tener los computadores más sofisticados (años 40-70). En la Tabla 1.1 se detalla lo más relevante del nacimiento de la industria.

Año	Acontecimiento
1947	Thomas T. Goldsmith y Estle Ray Mann desarrollan un sistema electrónico compuesto de válvulas y una pantalla de rayos catódicos ¹ , simulaba el lanzamiento de misiles contra un objetivo.
1952	Alexander Sandy Douglas crea el primer juego con versión digital "Nought and crosses" o "Ceros y cruces"; el famoso 3 en raya, no se lo consideró como videojuego, ya que no poseía movimiento en pantalla
1958	William Higinbotham, físico diseñador de circuitos electrónicos, realizó una especie de juego basado en un programa de cálculos de trayectoria "Tennis for Two".
1961	Steve Russell junto con otros compañeros, desarrollan Spacewar. En un ordenador PDP-1 ² donado al MIT, el juego circuló a través de Arpanet. Es considerado el primer juego de computador de la historia

Tabla 1.1 Inicios de la Historia de los Videojuegos. [8]

¹ Corrientes de electrones observados en tubos de vacío que se equipan por lo menos con dos electrodos, un cátodo y un ánodo

² Primer computador de la serie PDP de la Digital Equipment, famoso por ser el computador más importante de la cultura hacker en el MIT

1.1.1.2 Década del 70. [2] [3] [4] [5] [6] [7]

En ésta década la historia de los videojuegos, pasa por un punto de inflexión, surgen varias compañías creadoras de consolas con un único propósito, producir videojuegos y hacerlos comerciales. Aparece una industria que crecerá aceleradamente en los años venideros. En la Tabla 1.2 se resume los acontecimientos más importantes de esta década.

Año	Acontecimiento
1971	Nolan Bushnell creó su propia versión de Spacewar, "Computer Space": la diferencia fundamental de que su juego se presentaba en una pantalla común en lugar de requerir de un computador, reduciendo considerablemente los costos. Más tarde su experiencia le serviría para fundar Atari ³ .
1972	Ralph Baer, Magnavox Odyssey: Ralph Baer es considerado el inventor de los videojuegos, inventó las videoconsolas. Construyó la primera consola doméstica de videojuegos "Odyssey". Contaba con pistolas que se podían disparar en pantalla. Ping Pong para dos, generaba dos puntos por pantalla, una pelota y una línea central. Pusó en el mercado su primera consola Magnavox Odyssey o la "Caja Marrón" como fue bautizada. Atari y Pong: Nolan Bushnell junto con Ted Dabney, un ingeniero de Ampex, fundan Atari. El nombre de la compañía proviene del juego de mesa Jo, similar al "jaque mate" del ajedrez. Desarrollaron un juego de ping pong denominado Pong, más real que el de la consola Magnavox Odyssey de Baer. Pong fue introducido en bares en donde tuvo un éxito jamás visto hasta ese momento.
1973	Crecimiento de Atari: Bushnell amplió la capacidad de oficinas, contrato personal sin experiencia que montará los muebles de Pong. Se rodeo de personas que compartan su visión, por este motivo Ted Dabney salió de la compañía. Colocó a gente joven y con ambición, su selecto grupo fue conocido como "El Rey, la Reina y los cinco Príncipes". Atari lanzó al mercado su segundo juego Space Race, cohetes que competían por llegar a la meta mientras esquivaban asteroides, fue clonado posteriormente por Midway. Se creó una versión Pong Doubles, para dos jugadores y Gotcha en el que un jugador con una caja trataba de atrapar a otro con un X, precedente del Pac Man; sin embargo ninguno tuvo el mismo éxito que Pong
1974	Competidores de Atari: cada vez que Atari sacaba una máquina, sus competidores la copiaban. Mientras las patentes tardaban en llegar la competencia tenía tiempo para vender las copias y ganar dinero. Entre estas compañías se encontraban: Ramtek, Meadows Games, Nutting y Midway. Bushnell astutamente crea Kee Games haciéndola parecer como si fuera la competencia, dándose mayor publicidad. Kee Games creó un juego innovador llamado Tank, que consistía en dos tanques que se enfrentaban entre sí en un laberinto, tenía dos joystick ⁴ para controlar el tanque y un botón para disparar. Representó un gran avance, fue el primer juego en contar con una memoria ROM para almacenar los gráficos. Fue el juego del año. Japón entra al juego: los arcades se estaban convirtiendo en un buen negocio también en Japón. Namco fundada por Mayasa Nakamura, se convierte en la distribuidora oficial de Atari en Japón. Se crea el Home Pong: el prototipo del primer chip que contenía el Home Pong. Harold Lee un ingeniero de Atari propuso la idea de una máquina que pueda conectarse al televisor para jugar Pong en casa, cambiando la idea anterior de Magnavox; llegaría a la mayor cantidad de gente posible.
1975	Comercialización de Home Pong: Home Pong finalmente logró ser comercializado en Sears ⁵ . Para cubrir la gran demanda de unidades, Atari tuvo que obtener un crédito.

³ Empresa que desarrolla, publica y distribuye videojuegos para consolas

⁴ Proviene del inglés joy, alegría y stick, palo. Dispositivo de control de dos o tres ejes que se usa desde una computadora o videoconsola

⁵ Cadena estadounidense de tiendas departamentales fundada por Richard Warren Sears en el siglo XIX

	<p>Nuevos juegos: Atari lanza Indi 800 primer juego en color. Steeple Chase juego en donde varios jugadores corrían a caballo en un recorrido de obstáculos. Maneater creado por Project Support Engineers, intentó aprovechar el tirón de la película Tiburón; en el juego los personajes debían recoger monedas del fondo del océano mientras esquivaban tiburones. Midway tenía su propio Gran Track 10 Racer, Wheels y Wheels II. Western Gun de Taito, presentaba a un personaje en el juego por tanto tenía una historia, Dave Nutting mejoró los gráficos del juego y utilizó un microprocesador siendo el primer videojuego en utilizarlo.</p> <p>Aparición de Nintendo: Yamahuchi, propietario de Nintendo, tras discutir sobre los nuevos avances tecnológicos en el campo del entretenimiento e investigar el uso que la nueva tecnología en EEUU, negoció una licencia para vender el sistema de videojuegos de Magnavox en Japón. Ya que Nintendo no tenía conocimiento sobre el uso de microprocesadores se alió con Mitsubishi Electric.</p>
1976	<p>Atari Breakout: Atari contrató a Steve Jobs, que comenzó a diseñar Breakout. Concepto jugable a partir de Pong, en un extremo había una pala y en el otro una serie de bloques que había que destruir con la pelota que rebotaba entre ellos. Breakout acabo siendo el mejor juego del año y el primer clon de Pong que conseguía ser más adictivo que el original.</p> <p>Fairchild y Channel F: Fairchild crea la consola Channel F, en la que se podían jugar diferentes juegos guardados cada uno de ellos en cartuchos, que se podían intercambiar. Los juegos se podían adquirir por separado y venían en unos cartuchos de plástico. No tuvo demasiado éxito pero cambió la industria.</p> <p>Atari en venta: Bushnell sabía que el nuevo negocio estaba en los juegos de cartucho, necesitaba recuperar su posición, decidió pedir ayuda a Grass Valley para producir un microprocesador que no fuera tan caro como el de la Channel F. Se creó el Video Computer System (VCS), más poderoso y barato de fabricar. Sin embargo al tener varios problemas económicos, Bushnell decide que la mejor forma de obtener capital económico era poner en venta a Atari. Warner Communications compró finalmente Atari en 28 millones de dólares</p>
1977	<p>Cinematronic: primer juego con Gráficos Vectoriales, remake de Space War</p> <p>Atari lanza su VCS: contenía 9 juegos, con sensores para juegos tipo pong y un dispositivo especial llamado joystick (una palanca con varios botones). Incluía un procesador de 8 bits, 2 joystick, un conector de antena, un transformador y el juego Combat. El negocio ya no estaba sólo en el hardware, sino también en los cartuchos de juegos.</p> <p>Nintendo saca a la venta Nintendo Tv Game 6, consola a color: permitía jugar diferentes versiones del juego Ligth Tennis, negociaron sus derechos con Magnavox. Shigeru Miyamoto entra a trabajar como diseñador para los arcades, sería de gran importancia en el futuro.</p>
1978	<p>Bushnell deja Atari: Bushnell anuncia su salida de Atari por discrepancias con la cúpula directiva.</p> <p>Taito lanza Space Invaders: en este juego el jugador debía vencer 4 filas de marcianos que se aproximaban hacia el más y más rápido. Su innovación fue el Hi Score.</p> <p>Nintendo lanza el Tv Game 15: esta consola presenta dos controles con cables, lo que representó mayor facilidad para jugar.</p> <p>Magnavox Odyssey 2: la segunda generación de consolas Odyssey, similar a la de su competidora Atari 2600, incorporaba un teclado en la consola de 8 bits.</p> <p>Bally Astrocade: contaba con un procesador de 3.5 Mhz con 8k de ROM y 4K de RAM ampliable a 64K. Pero debido a retrasos en la producción su costo y la competencia, desapareció.</p>
1979	<p>Atari lanza Asteroids: una nave disparaba y podía girar sobre su propio eje. El objetivo del juego era disparar y destruir los asteroides sin que ningún fragmento se llegase a estrellar contra la nave que se controla. Vendió 70.000 unidades inicialmente destronando a Space Invaders de Taito.</p> <p>Namco lanza Pac Man: Pac Man desarrollado por Namco y diseñado por Toru Iwantani, supuso una revolución total de los videojuegos, su éxito fue tal que se creó una serie de dibujos animados y una canción.</p> <p>Capcom y Activision: se fundan durante este año. Capcom bajo el nombre de Japan Capsule Computers. Por otro lado algunos programadores abandonan Atari fruto del malestar interno y fundan Activision.</p>

Tabla 1.2 Historia de los videojuegos década del 70. [22] [61]

1.1.1.3 Década del 80. [2] [3] [4] [5] [6] [7]

En los años 80 existe mayores avances, pero la industria de reciente nacimiento y cuyo crecimiento se aceleraba sufre por primera vez una crisis de la que se recuperaría poco a poco. Se destaca en este período la aparición de microprocesadores, consolas portátiles y nuevos tipos de consola, revolucionando la industria. Ya no dependerá tan sólo del hardware, sino también de la creación de videojuegos específicos que se pueden correr en distintos tipos de consolas gracias a los cartuchos. Se detalla los acontecimientos importantes en la Tabla 1.3.

Año	Acontecimiento
1980	Atari licencia el Space Invaders: su consola que incluía 112 modos de juego, al licenciarla Atari ganó unos 100 millones de dólares barriendo el mercado.
	Intellivision de Mattel: fue la primera en sustituir los joystick por mandos con botones, compitió seriamente con Atari. Vendió unas 175.000 unidades.
	Nintendo distribuye su Game & Watch: la compañía presentó Nintendo of América Inc., empezaron a distribuir Game & Watch. Se podía jugar en una pequeña pantalla LCD, era reloj y alarma. Estas consolas fueron las responsables de la invención del Pad. Su éxito fue tan grande que la compañía en menos de 10 años lanzó cerca de 60 dispositivos. Los juegos más conocidos sobre este dispositivo fueron: Donkey Kong, The Legend of Zelda, Mario Bros, Mickey Mouse, Ballon Fighth.
1981	Donkey Kong: diseñado por Shigeru Miyamoto, en el juego, un fontanero llamado Jumpman-luego se llamaría Mario-, debe rescatar a su novia de las garras de un gorila llamado Donkey Kong. Este juego fue un gran éxito.
	Haunted House: fue el precursor del género Survival Horror ⁶ . En este juego para la Atari 2600, el jugador era representado con un par de ojos y debía recorrer una mansión recolectando piezas de una urna evitando las criaturas de la habitación, de esta forma se trataba de recrear el miedo.
1982	Commodore 64: el proyecto fue creado en Japón por Yashi Terakura, sus ventas fueron fabulosas en todo el mundo. Utilizaba casete y disquetera, existiendo expansiones para cartucho, permitía usar el lenguaje BASIC, paleta de colores y procesador.
	Emerson y su Arcadia 2011: se creó al mismo tiempo que la Atari 5200 y la Colecovision, era de menor calidad pero por su tamaño se la intentó vender como una consola portátil.
	Colecovision: lanzada al mercado por Coleco, era superior a todas las consolas de su época, además de que tenía un sistema con un adaptador en el que se podía jugar los juegos de la competencia. Se vendió con Donkey Kong y llegó a vender 5 millones de unidades.
	Atari 5200 y rediseño de la 2600: en la 2600 se cambió el juego incluido por Pac-Man, vendió 10 millones de unidades. Atari 5200 ofrecía mejores gráficos, controles con teclado numérico y botones a cada lado, una palanca sensible a presión e incluía por primera vez en las consolas un botón de pausa. Pero no tuvo mucho éxito debido a su incompatibilidad con la 2600.
	Vectrex: se diferenciaba de las demás, al basar sus juegos en vectores, por esta razón era capaz de recrear entornos 3D. Debido a varios factores y la competencia se dejó de fabricar al igual que otras consolas.

⁶ En español Terror de Supervivencia, es un subgénero perteneciente al género de los videojuegos conocido como aventuras de acción

	<p>Electronic Arts y Ocean: Electronic Arts es fundada por Willian M. Hawkin III, bajo el nombre de Amazin Software, acabo facturando 5 millones de dólares el primer año y 11 millones el segundo. Ocean Software fue fundada en Manchester, sería la compañía de este género más importante en Europa por mucho tiempo.</p>
	<p>Año de Crisis: Atari sufre una crisis institucional pese a liderar el mercado. En este año las ventas de videojuegos se triplicaron, pero en las navidades de este mismo año se desataría una crisis, con serios problemas en ventas de los videojuegos.</p>
	<p>Nintendo lanza Famicom: creada por Hisoshi Yamauchi y Massayi Ueremaya. Ya que el CPU no podía realizar todo el trabajo de la consola esta incluía una PPU, (Picture Processing Unit). La Famicom en el resto del mundo se llamo NES (Nintendo Entertainment System), después de solucionar algunos errores que contenía fue un éxito en Japón.</p>
	<p>Sega y su primera consola: SG-100 fue la primera consola de SEGA, utilizaba cartuchos o tarjetas. Más tarde sacarían el SG- 3000, que era una versión de la SG-1000, con teclado incorporado.</p>
	<p>Nace Mario Bros: anteriormente visto en Donkey Kong, pero ahora tenía su propio juego. Ya no se llamaba Jumpman sino Mario, al mismo tiempo aparece su hermano Luigi. El objetivo inicial era limpiar las tuberías de las diferentes plagas que había en el juego, por debajo de ellas.</p>
	<p>Bomberman: creado por Hudson Soft, sin argumento alguno, el juego consistía en ir colocando estratégicamente bombas en cada nivel, con el objetivo de eliminar obstáculos y enemigos.</p>
1983	<p>Intellivision II: debido al éxito de su antecesora Mattel decide sacar una segunda versión. Incluía joysticks desmontables, un teclado musical opcional y un sistema que permitía cargar juegos de la Atari 2600.</p>
	<p>MSX: MicroSoft sXtended. Primer sistema que no dependía de una sola marca. Poseía cartuchos de memoria para cargar juegos y programas además de sus periféricos.</p>
	<p>Crisis: para las navidades de este año había una gran cantidad de videojuegos, pero los compradores no podían distinguir cual era bueno, malo o aburrido. Atari ya tenía mucha competencia con consolas como Bally Astrocade, Colecovision, Emerson Arcadia 2001, Magnavox Odyssey II, Mattel Intellivision, entre otras. La sobresaturación afecto al comercio, gran cantidad de juegos no se pudieron vender, los comercios empiezan a rebajar el precio de venta para deshacerse del stock; aparte de esto el boom de los ordenadores personales que ofrecen mucho más que las consolas a un precio competitivo, provocó que la industria de ser la de mayor crecimiento pase a estar en crisis. Mattel perdió 195 millones, Coleco 258 millones; esta última con Magnavox abandonarían el negocio y Atari quedo cerca de la banca rota con 356 millones en pérdidas.</p>
	<p>Se crea Squaresoft: a pesar de la crisis se funda la que sería una de las compañías importantes en la historia de los videojuegos. Al cabo de 4 años entra en banca rota y estuvo a punto de desaparecer, pero crearon Final Fantasy gracias a Hironobu Sakaguchi, levantando así la empresa.</p>
1984	<p>La Crisis Continúa: muchas compañías tuvieron que retirarse, Atari cae en banca rota es vendida por Warner a Commodore por 240 millones.</p>
	<p>King's Quest I: considerada como la primera aventura gráfica de la historia. Desarrollado por Sierra Online fue una revolución llegando a producirse 8 partes de este juego y a cada una de ellas con algunos remakes⁷.</p>
	<p>La industria comienza a recuperarse: Nintendo lanza Famicom en América, alcanzando gran éxito. Se lanza Pitfall 2 de Sega, Indiana Jones de Atari, Comando Ghosts'N Goblins de Capcom.</p>
1985	<p>Nintendo y Super Mario Bros: creado para la Famicom y diseñado por Shigeru Miyamoto, este juego fue el que lanzo al estrellato a Mario convirtiéndolo en la mascota principal de la compañía. Podía comer zetas y agrandar su tamaño, además de lanzar bolas de fuego al comerse una flor. En este videojuego aparece por primera vez la princesa Peach Toad y Bowser. Es considerado el juego más vendido de todos los tiempos, al crear un antes y un después en la historia de los videojuegos.</p>
	<p>Tetris: de los más importantes de la historia, fue inventado por Alexey Pazhintov. Se trataba de siete formas construidas a partir de bloques cuadrados, se hizo famoso cuando se incorporo a IBM PC.</p>
	<p>Sega - Master System: lanzada en Japón. Esta consola tenía una mascota llamada Alex Kidd.</p>

⁷ Se refiere a producciones que reproducen los detalles de un trabajo anterior

1986	Sega- Master System en EEUU: era la SG-1000 Mark III rediseñada para venderse en el mercado americano. Más avanzada que la NES, vendió 125.000 consolas en los 4 primeros meses y la NES 2.000.000
	Atari lanza la 7800: era una mejora de la 2600, en un intento por recuperar el mercado perdido.
	Nace Legend of Zelda: creado por Shigeru Miyamoto, la historia se ambienta en la tierra de Hyrule donde un joven llamado Link tiene que rescatar a la princesa Zelda de las garras de Ganon (Rey del mal). Contaba con innovaciones técnicas como grabar el juego a pesar de que se apagara la máquina. Llegó a vender 6 millones de copias y se convirtió en una de las mejores franquicias de Nintendo.
	Metroid para NES: fue producido por Gunpei Yokoi. Incluía por primera vez una protagonista femenina y no tenía un desarrollo lineal. Al igual que Zelda acabaría convirtiéndose en una de las insignias de Nintendo.
	Taito Lanza Arkanoid: versión renovada de Beakout de Atari. La pala se sustituye por una nave llamada Vaus, se mejoró varias cosas a tal punto que se convirtió a varios sistemas de la época.
	Castlevania: desarrollado por Konami, salió en Japón bajo el nombre de Demon Castle Dracula para Famicom Disk System. En EEUU no se conocería hasta el siguiente año, cuando fue programado para correr en la NES. Esta saga perdura hasta la actualidad.
1987	La Nec PC Engine /Turbograf 16: desarrollada entre Hudson Soft y NEC. En Japón tuvo mucha aceptación, llegó a venderse más que la Famicom en los primeros meses de su lanzamiento.
	Aparece el primer Final Fantasy: la compañía Squaresoft estaba pasando por momentos muy difíciles, Hironobu Sakaguchi empezó el proyecto de un RPG ambicioso, Final Fantasy. Se dice que se llamó así porque iba a ser el último juego de Squaresoft. Tuvo tal éxito que Sakaguchi decidió desarrollar nuevos Final Fantasy. Final Fantasy VII fue el que dio a conocer el RPG, cuando mucha gente no sabía de qué se trataba este género.
	Maniac Mansion: juego de Lucasfilm diseñado por Ron Gilbert. Gilbert diseñó un motor específico llamado SCUMM (Script Creation Unity for Maniac Mansion); supuso una revolución en las interfaces de juego de la época. Permitía interactividad entre el usuario y el juego a partir de verbos combinados con los objetos formaban frases que definían acciones. Se podían seleccionar varios personajes y tener distintos finales. Salieron versiones del juego para varias consolas.
	Nace Megaman: la idea de Keiji Inafune, inicialmente para salones recreativos, se lo desarrolló para Famicom. Se ambienta en el año 20XX, un niño robot que es creado por el Dr. Light, tenía que enfrentarse a 6 robots jefes finales de cada fase. Arrasó en EEUU se agotó al tercer día de salir a la venta. Fue una de las plataformas más fabulosas de la época.
	Primer Metal Gear: diseñado por Hideo Kojima en un principio para la MSX, poco después Konami publicó una versión para la NES. Solid Snake se encarga de infiltrarse en Outer Heaven, para recuperar a Gery Fox que se había infiltrado anteriormente, pero perdió comunicación con él. La acción se muestra en una perspectiva de área.
1988	Sega lanza Mega Drive en Japón: al ver que Master System no triunfaba en EEUU, Sega decide sacar una consola de 16 bits, tan potente como el Atari ST y el Commodore Amiga. Llamada MK-1601, más tarde se llamaría Sega Mega Drive en Japón y como Sega Genesis en EEUU. Con esta consola Sega fue recuperando poco a poco terreno y se convertiría en la consola más vendida hasta la aparición de Super Nintendo. Se destacan juegos como Sonic, Golden Axe, Light Crusader, Street Fighter II, entre otros.
	Super Mario Bros 3: el último juego de Mario para Famicom. Se trataba de recuperar las varitas robadas de los siete reyes del Mushroom World por parte de los Koopalings y salvar a la princesa del malvado Koopa. Contaba con más niveles, movimientos y habilidades. Un éxito en ventas.
1989	Atari Lynx: desarrollada por Epyx, con el nombre de Handy y comercializada por Atari. Apareció al mismo tiempo que el Game Boy, pero no había desarrolladores suficientes que programaran para la máquina. Con la aparición del Game Gear de Sega terminó fuera del mercado de las portátiles.
	Nintendo lanza la Game Boy: diseñada y fabricada por una división de Intelligent System, dirigida por Gunpei Yokoi. Su objetivo era fabricar una videoconsola que fuera pequeña, ligera, barata y con muchos videojuegos. Permitía insertar cartuchos intercambiables. Contó con más de 100 millones de unidades vendidas.

	Aparece Sim City: diseñado por Will Wright, era un simulador de construcción de ciudades en el que ejerciendo el papel de alcalde se crean industrias, zonas residenciales y servicios. Además se debía tener en cuenta factores como la seguridad, diversión, felicidad, entre otros. Era muy versátil que supuso la creación de un subgénero dentro de la estrategia.
	Capcom lanza Pang: conocido en EEUU como Buster Bros. Se empieza con globos de distintos tamaños que al tocar al personaje le quitan vida, se tenía que disparar un arma para dividir los globos hasta que desaparezcán.

Tabla 1.3 Historia de los videojuegos década del 80. [8]

1.1.1.4 Década del 90. [2] [3] [4] [5] [6] [7]

Esta década se la considera como la época de oro de los videojuegos, con los adelantos tecnológicos, se crea mejor hardware para las consolas. El ascenso de la tecnología provoca mucha rivalidad entre compañías. Finalmente Sega se terminará despidiendo del mercado y lanzando su última consola, DreamCast. Nace PlayStation y Nintendo 64, que dominarían el mercado. Se crean los primeros juegos con perspectiva 3D y se revolucionan géneros como los Survival Horror y los RPG's⁸, se resume en la Tabla 1.4.

Año	Acontecimiento
1990	Super Famicom: es una de las consolas que más ha triunfado en la historia de Nintendo la Super Famicom o Super NES, gracias en parte a sus juegos como Super Mario World, Súper Mario kart, The Legend of Zelda entre otros. En poco tiempo comenzó su rivalidad con Mega Drive de SEGA, duraría años.
	Game Gear: Sega decide sacar su alternativa del Game Boy portátil, la Game Gear. Era una adaptación de la Master System, tuvo todo el catálogo de juegos de esta consola, pero su batería no tenía mucha duración por la pantalla a color y era considerablemente más grande que el Game Boy; a pesar de estos problemas alcanzo buenas ventas.
	Sega y su Master System II: era mucho más ligera, carecía de algunas cosas de la consola anterior. Incluía Alex Kidd in Miracle World y Sonic The Hedgehog, en un esfuerzo de luchar contra el Súper Mario Bros de Nintendo.
	SNK⁹ pone a la venta Neo Geo: la tecnología de esta consola era muy superior a la de otros dispositivos de la época. Tenía un precio elevado 649.99 \$, poseía características técnicas que combinadas eran capaces de funcionar 50 % más rápido que el Mega Drive.
	Commodore comercializa el C64 Game System: Commodore sacó a la venta C64, eliminando el puerto serial, con el bus de usuario y puerto de cassette ocultos en el interior de la carcasa. Cosecharon un gran fracaso no se llegó al 25 % de unidades vendidas.
	Amstrad y el GX 4000: Amstrad intenta entrar al mercado de las videoconsolas. El GX 4000, tenía buenos gráficos pero no muchos juegos, fue un fracaso comercial, existía consolas comerciales con procesadores superiores.

⁸ Role Playing Game o Juego de Rol, en el que los jugadores asumen el rol de los personajes a medida que se desarrolla una historia.

⁹ Shin Nihon Kikaku empresa japonesa dedicada al desarrollo de videojuegos.

	<p>Super Mario World: considerado uno de los mejores juegos de la historia, fue de los primeros que compensaba al jugador al completarlo al 100% o descubrir todos los secretos. Hay nuevas habilidades para Mario, aparece Yoshi y la capa de Mario.</p> <p>The secret of the Monkey Island: desarrollado por Ron Gilbert y Steve Purcell, usaba el motor SCUMM¹⁰. Contaba la historia de un joven que llegaba a una isla con la intención de convertirse en un fiero pirata. Revoluciona las aventuras gráficas, con la novedad de las batallas de insultos.</p>	
1991	<p>Sega Mega CD: Sega intenta aprovechar las capacidades del CD ROM, es decir juegos largos con secuencias digitales, en su nueva consola. Pero al no contar con muchos juegos para el sistema termino desapareciendo.</p> <p>Philips CD -i: diseñado por Philips y Sony, reproducía discos CD's de Audio, video, etc., pero no contaba con muchos videojuegos.</p> <p>Sonic llega para hacer frente a Mario: Sega se propuso la meta de crear un videojuego que sustituya a Alex Kidd, como mascota de la compañía. Naoto Oshim creó el personaje, un erizo azul del color del logotipo de Sega. Sonic debía salvar animales secuestrado por el doctor Robotnik, que quería utilizarlos como fuente de energía para su ejército de robots; para conseguir este objetivo Sonic tiene que recolectar 7 esmeraldas del caos. Se diferenciaba de Super Mario Bros por su velocidad y dinamismo. Fue un gran éxito mundial y llegaría a las 4 millones de copias vendidas.</p> <p>The Legend of Zelda - A Link to the past: el primer juego de Zelda en tener dos mundos paralelos y el único de la saga en salir para la NES. Se convirtió en uno de los juegos más vendidos de la súper Nintendo y considerado uno de los mejores juegos de la saga.</p> <p>Street Fighter II: Este contaba ya con 8 personajes, 4 jefes finales y un final diferente para cada personaje. Cada luchador tenía distintas características, técnicas y movimientos especiales; tuvo tanto éxito que se convirtió a casi todos los sistemas domésticos de videojuegos.</p> <p>Sega Meganet: era un sistema que permitía jugar online, contra otros jugadores, varios juegos de Mega Drive. Funcionaba a través de un modem externo, fue el precursor de los futuros sistemas online, pero fracaso debido a que era un servicio muy lento.</p>	
	<p>La Segunda parte de Sonic: aparece un nuevo personaje llamado Tails, un zorro con dos colas que tenía la habilidad de volar, se jugaba con ambos a la vez, el juego era más rápido, había más enemigos y se observa por primera vez a Super Sonic cuando captura las siete esmeraldas del caos.</p> <p>Sale Mortal Kombat: luego de Street Fighter; el mercado se lleno de videojuegos de lucha. Midway creó Mortal Kombat, constaba de 7 luchadores y la misión era ser el campeón de Mortal Kombat. Para los personajes se usaron actores reales. Por su violencia se llego a prohibir, lo que hizo que aumentara su popularidad y ventas. Se lo uso en PCs y casi todas las consolas.</p> <p>Wolfenstein 3D: no se puede decir que fue el primer juego en probar la perspectiva tridimensional, pero si fue el primero que la aprovecho para obtener éxito. Se trata de un policía que es atrapado por los nazis y encerrado en un castillo, su misión escapar. El motor gráfico carecía de suelo, escaleras, etc., pero fue una revolución.</p> <p>Llega el Survival Horror: con la llegada de Alone in the Dark este género toma forma. Fue de los primeros en usar entorno 3D y gráficamente una maravilla para la época.</p>	
	<p>La última consola de Atari, Mega Drive II: era una versión mayor de placa que su antecesora, cambio su aspecto exterior y sus adaptadores.</p> <p>Doom: viendo el éxito de Wolfenstein, Id Software empieza a desarrollar Doom. Tuvo mucho más éxito y sirvió de base para todos los FPS (First Person Shooter) que salieron a la venta posteriormente. Originalmente para PC, debido su éxito tuvo conversiones a casi todas las consolas.</p> <p>El primer FIFA: creado por EA -Electronic Arts- con gráficas normales, pero con gran cantidad de animaciones, un sistema de repeticiones y buena inteligencia artificial. Fue lanzado para todo tipo de consolas.</p> <p>Atari y la Jaguar: Contaba con especificaciones impresionantes para la época, pero no tuvo mucho éxito.</p>	
	1993	

¹⁰ Script Creation Utility for Maniac Mansion, creado como motor para el videojuego Maniac Mansion.

	<p>El 3DO: diseñado por Panasonic, Sanyo y GoldStar, su precio era excesivo, podía reproducir CD's de audio, Photo CD's y video CD's. No llego a triunfar.</p> <p>Pionner Laser Active: los juegos se vendían por separado, add-on para compatibilidad con Mega Drive y la TurboGrafx-16, la consola costaba 970\$, los add-on 600\$ y el karaoke 300\$ convirtiéndose en la consola más cara de la historia.</p>
1994	<p>Sega 32X: llamado en un principio Proyecto Marte, su objetivo ser similar a la mega Drive pero con 32 bits. Permitía mostrar 32.768 colores, gráficos 3D y 25.000 polígonos por segundo, pero no llego a triunfar, ya que Super Nintendo empezó a incorporar cartuchos que le permitían mejorar sus juegos de manera notable.</p> <p>Sega Lanza Saturn en Japón: La llegada de la consola de Sony la PlayStation era inminente y para no verse superado por la competencia Sega acelera la construcción de esta consola. Con la salida de la consola de Sony, la Saturn fue perdiendo terreno y con la llegada del Nintendo 64, quedó relegada al 3er lugar.</p> <p>La PlayStation llega a Japón: con Ken Kutaragi a la cabeza, es lanzada. En su catálogo había sagas como Metal Gear Solid, Resident Evil, Tomb Raider, Tekken, Final Fantasy, entre otros. Fue exitosa barriendo con toda su competencia a su paso y convirtiéndose en una de las más vendidas de la historia.</p> <p>SNK y su Neo Geo CD: SNK decide hacer su sistema de juegos accesible a todo el mundo, ya que su consola era cara, abaratando costos aumentando la memoria RAM y haciendo adecuaciones para CD. No llego a tener muchas ventas.</p> <p>Playdia de Bandai llega a Japón: basada en el uso del video CD, con un costo de 250\$. Fue la primera consola en incorporar mandos inalámbricos de manera estándar. No tuvo gran acogida en el mercado.</p> <p>Blizzard y Warcraft: supuso un punto de inflexión en los juegos de estrategia en tiempo real, teniendo superventas. Estaba inspirado en Warhammer, había razas de orcos y humanos. Su punto fuerte era la inteligencia artificial que poseía.</p>
	<p>Nace el E3: la asociación de Software Digital Interactivo, inaugura la primera Exposición de Entretenimiento Electrónico, conocida como E3. Reúne a las principales compañías de videojuegos de todo el mundo para que demuestren sus futuros lanzamientos y consolas.</p> <p>Virtual Boy: diseñado por Gunpei Yokoi, con la intención de tomar ventaja en el interés que había por la realidad virtual. Constaba de un "array" para cada ojo y un sistema de espejos que recreaban la resolución total de pantalla. Se colocaba un sistema de visualización en la cabeza, por esta razón no se podía ver el mando de control. Por su escaso éxito fue retirada del mercado.</p> <p>Sega Nomad: era una versión portátil de la Mega Drive, permitía conexión a televisión mediante el cable de video de Mega Drive 2. Fracaso principalmente porque tenía muy poca autonomía, entre 1-2 horas</p> <p>Apple y su Pippin: una mezcla de consola y computador, corría el sistema operativo MacOS. Paso bastante desapercibida, el mercado estaba dominado por la PlayStation, la Saturn y el Nintendo 64.</p>
	<p>Nintendo 64 llega al mercado: Nintendo tras años de acuerdos con empresas fabricantes de hardware inicia un proyecto, el Ultra 64. Durante la primera semana vendió 500.000 unidades. El mando de control, poseía los botones dispuestos en forma de cruz y tenía vibración con el Rumble Pack¹¹. En su catálogo constaban juegos como Super Mario 64, The Legend of Zelda, Super Smash Bros, entre otros.</p> <p>La Game Boy Pocket: era una versión más pequeña de la Game Boy original. Con la pantalla más grande que su antecesora, redujo el consumo de energía usando pilas AAA.</p> <p>Mario 3D: Nintendo presenta su primera plataforma en 3D, Super Mario 64. Llego a vender 12 millones de copias y es considerado como uno de los mejores juegos. Existe una versión para la Nintendo DS.</p> <p>El primer Resident Evil: el género Survival Horror se reinvento con este juego. Llego de las manos de Capcom con la consola PlayStation, aunque más tarde fue portado a varias plataformas. Una de las sagas más conocidas y que más éxito han tenido en el mundo, con 30 millones de unidades vendidas, llegando a sacar secuelas, películas de cine y novelas.</p>

¹¹ Dispositivo creado por Nintendo, diseñado para el mando de la Nintendo 64 proporcionaba vibración en la acción del videojuego.

1997	Final Fantasy VII para Play Station: al principio estaba previsto lanzarlo para Nintendo 64, pero esta consola usaba cartuchos y la obra de SquareSoft no contaba con ese soporte. Sería la primera vez que se vería un juego de esta saga en la consola de Sony, acabo ocupando 3 CD's. Un gran éxito en juegos RPG's, supero los 10 millones de unidades. Personajes en 3D, secuencias de video, es considerado como uno de los mejores videojuegos de la historia.
	Grand Theft Auto: tuvo éxito gracias a la libertad de movimientos y acciones a lo largo de las grandes ciudades, conduciendo distintos vehículos y siempre al margen de la ley.
	Llega Age of Empires: basado en estrategia en tiempo real, desarrollado por Ensemble Studios y publicado por Microsoft. Utilizaba una vista isométrica ¹² para representar los escenarios. Es de las sagas más famosas. La primera Age of Empires. The Rise of Rome, Age of Empires II, The Age of Kings y finalmente Age of Empires III.
	Snakes y el teléfono móvil: Snakes hace su aparición en los teléfonos móviles, concretamente en los Nokia, siendo de los juegos más expandidos en la telefonía móvil.
1998	Sega presenta su última consola, DreamCast: fue la primera consola de 128 bits que salió al mercado, contaba con soporte para GD-ROM ¹³ . Fracaso en gran medida por el fracaso de su antecesora Saturn y por la competencia con PlayStation 2, cuya antecesora había tenido un éxito arrollador. Aún así es considerada como una de las mejores consolas que han existido, en su catálogo destacan: Marvel vs Capcom 2, Sonic Adventure, Street Fighter 3, entre otros.
	Nintendo lanza Game Boy a color: con mejor hardware y velocidad, tuvo éxito por ser compatible con todos los juegos de la Game Boy original, además Nintendo actualizó los juegos para que tuvieran mejor aspecto para esta consola. En 4 años se lanzaron 230 juegos para este sistema.
	Neo Geo Pocket: la consola portátil de SNK, no tuvo éxito a pesar de tener bastantes juegos buenos. Destacan SNK vs Capcom, Sonic the Hedgehog Pocket Adventure.
	Legend of Zelda-Ocarina of Time: el primero en 3D para la saga de Zelda, diseñado por Shigeru Miyamoto. Es considerado por muchos el mejor juego de la historia, revistas reconocidas como Famitsu o Edge le dieron puntuación perfecta.
	Metal Gear Solid para PlayStation: desarrollado por Konami, diseñado por Hideo Kojima es una de las sagas más importantes en el mundo de los videojuegos.
	El primer Gran Turismo: creado por Polyphony digital, estaba basado en la simulación de la conducción, uno de los más vendidos de PlayStation, tiene numerosas secuelas.
1999	Bandai y la WonderSwan: diseñada por Bandai y Gunpei Yokoi, creador de la Game Boy original. No tuvo éxito pero llego a controlar el 8 % del mercado japonés.

Tabla 1.4 Historia de los videojuegos década del 90. [8]

1.1.1.5 Año 2000 hasta la actualidad. [2] [3] [4] [5] [6] [7]

Gracias al crecimiento acelerado de la tecnología, a partir del año 2000, se crean mejores consolas, existe un gran avance tanto en las consolas portátiles como en las consolas de sobremesa. De esta manera nacen PlayStation 2, Nintendo Game Cube, Xbox de Microsoft; y la séptima generación de consolas: PlayStation 3, Nintendo DS, Wii, PSP. Los juegos cada vez poseen mejores gráficos y ambientación, en gran medida gracias al

¹² Esta vista es usada para crear la sensación de profundidad en los videojuegos 3D

¹³ Giga Disk Read-Only Memory, es un formato de discos óptico desarrollado por Yamaha para la compañía de videojuegos SEGA

hardware que se actualiza. Hoy en día incluso se tiene consolas 3D, los acontecimientos son resumidos en la Tabla 1.5.

Año	Acontecimiento
2000	Sony rediseña la PlayStation I: con el nombre de Psone, era más estilizada y con periféricos extra. Contaba con protección contra todos los mod-chips que había hasta entonces.
	PS2: contaba con un lector de DVD para juegos y películas. Era la primera que llevaba puertos USB. Barrió con todos sus rivales DreamCast, GameCube y XBOX. Llegó a superar en ventas a su predecesora.
	Shenmue llega a la DreamCast: creado por Yu Suzuki. Su desarrollo se remonta años atrás pero debido al fracaso de las anteriores consolas fue traspasado a la DreamCast. Se convirtió en el juego más caro desarrollado hasta ese momento con 20 millones de dólares. Se incluyó el concepto de Full Recreative Eyes Entertainment (Free), que da una mayor interacción.
2001	La nueva consola de Nintendo-Game Cube: la consola no usaría DVD. Los programadores tenían facilidad de desarrollo. Debido a la ventaja que llevaba PS2, quedó relegada al segundo lugar en ventas junto con la XBOX. En su catálogo se encuentran Eternal Darkness, Starfox Adventures, Resident Evil, Zelda: The Wind Waker, Metroid, entre otros.
	Microsoft se une-Lanzamiento de Xbox: la primera consola en la historia que venía equipada con un disco duro. Tuvo bastante éxito en EEUU y en distintos mercados se sitúa en segundo lugar de las más vendidas, nunca pudo superar a PlayStation y sus números espectaculares en ventas.
	Nintendo y su Game Boy Advance: la sucesora de la Game Boy y la Game Boy Color, era capaz de ejecutar casi todos los juegos de estas dos consolas. Gracias a su gran catálogo terminó triunfando en ventas, entre sus juegos destacan: Mario Kart, Super Circuit, Castlevania, Final Fantasy V y VI, entre otros.
2002	GP32: creada por Game Park, con el objetivo de crear una consola portátil coreana para competir con el monopolio japonés, la portátil más avanzada de su época. Estaba orientada al software libre y por ese motivo contaba con numerosas aplicaciones, pero sólo fue comercializada en Korea, España e Italia.
	Llega Halo a Xbox: el FPS ¹⁴ más famoso de la historia de Microsoft, vendió sobre los 5 millones de unidades y tuvo dos secuelas más.
	Konami y el primer Pro Evolution Soccer: para PlayStation y PlayStation2, en Japón se llamó Winning Eleven 5. A pesar de que no convencía a los usuarios al principio, ya que carecía de licencias para usar nombres de los jugadores y estadio, terminó por convertirse en el mejor simulador del año.
2003	Nintendo presenta GBA SP: la Game Boy Advance SP, es la primera portátil de Nintendo que sustituye las pilas por una batería de Ion Litio, similar a su antecesora con pequeños cambios.
	La NCage de Nokia: Nokia lanza su consola-teléfono móvil, con reproductor mp3 y radio FM integrada. Pese a sus altas expectativas fracasa debido a su precio.
	PSX de Sony: se trataba de un aparato multifuncional, que traía consigo un sintonizador de TV, una grabadora DVD, Disco Duro y la consola PlayStation 2. Sólo fue comercializada en Japón.
2004	Nintendo DS: una consola de dos pantallas Dual Screen, una es táctil, cuenta con un micrófono con reconocimiento de voz y una conexión wifi para jugar partidas online. No es sucesora del Game Boy Advance, ha sido la consola más vendida junto con la Nintendo DS Lite (su rediseño)
	La PSP¹⁵ de Sony: Sony crea una consola con características impresionantes, tratando de incursionar en el campo de Nintendo; cuenta con un formato UMD (Universal Media Disk), que permite ver películas, convirtiendo así a la PlayStation Portable en un centro multimedia, se puede ver fotos y escuchar música; además incorpora un stick analógico, siendo la primera portátil en incluirlo. Se vendió bastante bien pero con la llegada de Nintendo DS Lite quedó relegada al segundo lugar.

¹⁴ First Person Shooter, es un género de videojuegos que centra a un jugador en un arma de combate, estos videojuegos son conocidos como shooter games

¹⁵ Play Station Portable, videoconsola portable de la multinacional Sony Computer Entertainment

	Sony y el rediseño de la PS2: es un rediseño más pequeño, delgado y silencioso.
2005	Llega la Xbox 360. Salió a la venta de forma masiva para Japón, Estados Unidos y Europa.
	Nintendo y su Game Boy Micro: rediseño de la Game Boy Advance, mucho más pequeño, compatible con los juegos del Game Boy Advance.
	GP2X: continuando la línea de la GP32, orientada al software libre reproduce gran cantidad de formatos multimedia.
2006	Sony y su PlayStation 3: es la tercera consola de Sony, presentada al público en el E3. Incluye un puerto HDMI y un Ethernet, un lector Blue-ray.
	Llega la revolución con Wii: es la primera consola de Nintendo que no lleva su nombre. Como principal característica, cuenta con un control inalámbrico incorporado llamado Wiimote, capaz de detectar el movimiento y rotación en un espacio de tres dimensiones, incluye vibración y altavoz.
	DS Lite: rediseño de Nintendo DS, tiene una forma mucho más delgada es más ligera que su antecesora.
2007	Microsoft da a conocer la Elite: Se trataba de la Xbox 360, con disco duro y un conector HDMI ¹⁶ para rivalizar con la PlayStation3.
	Sony rediseña la PSP: su batería es de más capacidad y 33 % más ligera que la anterior. Se la puede conectar a la televisión, a través de una salida de video y se redujo su tiempo de carga de juegos.
2008	DSi nueva versión de la DS: se hizo más ligera, fina y aumentó el tamaño de las dos pantallas, aumentó su memoria interna e incluyó cámaras digitales en la parte externa e interna de la consola. Cuenta además con un navegador de internet, reproduce archivos de música, MP3 ¹⁷ y AAC ¹⁸ .
2009	PSP GO: la versión de la portátil de Sony. Se eliminan el formato de juegos físico, siendo las descargas digitales la única opción a la hora de instalar videojuegos.
	DSi XL: el tamaño de la pantalla se agranda, pasa a ser de 4.2 pulgadas en esto se diferencia de la DSi.
2010	El 3D estereoscópico¹⁹ llega a PS3: PlayStation 3 se actualiza y permite visualizar juegos en formato estereoscópico 3D, se requiere de un televisor compatible y unas gafas específicas.
	Nintendo 3DS: Nintendo anuncia la creación de una nueva consola, con la capacidad de mostrar juego en 3D sin la necesidad de usar gafas.
	Onlive: plataforma potente de juegos que se sirve de internet, no requiriendo un equipo potente para funcionar. El juego es computado, renderizado y almacenado en línea, se lo alquila unos días o incluso años, hasta 3. Es capaz de transmitir los juegos a la MicroConsole ²⁰ , conectada a un televisor, reproduce cualquier tipo de juego que se computa sobre el servidor.
2011	Kinect de Microsoft: permite a los usuarios de la Xbox 360, interactuar con la consola sin necesidad de tener contacto físico, mediante una interfaz de usuario que reconoce comandos de voz. Compite contra sistemas como Wiimote, Wii MotionPlus y PlayStation Move, que controlan el movimiento de las consolas de Wii y PlayStation3.
	Nintendo y su Wii MotionPlus: permite capturar movimientos complejos, con más precisión que lo que hacia Wii Remote para la consola Wii.
	PlayStation Move: sistema creado para el control de movimiento de la consola PlayStation3. Posee un PlayStation Eye que reconoce el color del control y detecta sus movimientos.
2012	Nintnedo y su Wii U: sucesora de la Wii, es la primera consola en producir gráficos de alta resolución 1080p, cuenta con un nuevo mando con una pantalla táctil incorporada llamado Wii U Game Pad. Soporta los juegos de Wii.

¹⁶ High Definition Multimedia Interface, es una norma de audio y video digital cifrado sin compresión.

¹⁷ MPEG-2 Audio Layer III, formato de compresión de audio digital

¹⁸ Advanced Audio Coding, formato informático de señal digital de audio basado en un algoritmo de compresión por pérdida, obteniendo mayor grado de compresión.

¹⁹ Es una técnica capaz de recoger información visual y crear una ilusión de profundidad en una fotografía o película, se crea creando una imagen diferente para cada ojo. Fue inventada por Sir Charles Wheatstone en 1840.

²⁰ Es un dispositivo conectado al televisor que permite la transmisión de los datos desde internet de los videojuegos almacenados en la nube de Onlive

	<p>The Walking dead: desarrollado por Telltale Games, se basa en la historia de los sobrevivientes de un apocalipsis zombie del comic del mismo nombre. Con una excelente trama es de los juegos más galardonados de este año.</p> <p>Diablo III: desarrollado por Blizzard, la continuación de la saga brinda el crecimiento de los distintos personajes que se pueden escoger, es exclusivo para PC.</p> <p>Super Mario Bros.U: creado por Shigeru Miyamoto, es exclusivo para la consola de Nintendo Wii U, combina la jugabilidad clásica de Mario en dos dimensiones con Figuras espectaculares</p>
2013	<p>Ouya y su código abierto: videoconsola de código abierto, funciona con Android originalmente pensada para dispositivos móviles. Julie Uhrman, encontró financiamiento para su proyecto a través de Kickstarter²¹. Cada consola puede ser usada como kit de desarrollo, permitiendo a los propietarios desarrollar sobre la consola.</p> <p>Nvidia entra en el terreno de las consolas: Nvidia Shield es la consola creada por Nvidia, trabaja con Android. Cuenta con un mando similar al del Xbox360 con una pantalla integrada, permite realizar descargas de juegos en línea a través de Google Play²², además conectarse con una PC que cuente con una tarjeta GeForce GTX a través de wifi y ejecutar los juegos alojados en el computador.</p> <p>PlayStation 4: la cuarta videoconsola de Sony, directa competidora de la Wii U de Nintendo. Se cambió la arquitectura del procesador para atraer a los desarrolladores, cuenta con nuevas aplicaciones y servicios. Permite a las tablets y teléfonos inteligentes convertirse en una segunda pantalla, además Sony se alió con Gankai, permitiendo alojamiento de contenidos y descargas en línea.</p> <p>Xbox One: tercera videoconsola producida por Microsoft, sucesora de la Xbox 360. 10 veces más potente y con un Kinect actualizado permite conectarse a Skype, utilizar comandos de voz y gestos mejorando su funcionalidad. Puede grabar video y transmitir juegos en línea a través de Xbox Live.</p>
2014	<p>Nuevas perspectivas: se espera que la escasez de videojuegos de las nuevas consolas sea solventado, en este año se tendrá una visión del impacto en el mercado causado por la nueva generación de videoconsolas.</p>

Tabla 1.5 Historia de los videojuegos actualidad [8]

1.1.2 PRINCIPALES AVANCES EN LOS ÚLTIMOS AÑOS

Se ha dado un breve vistazo a los acontecimientos históricos y como ha ido evolucionando la industria. Ahora se procede a revisar las técnicas que son aplicadas para la creación de los videojuegos, para visualizar un panorama general de los últimos avances. Los videojuegos de PC tienen mayor mercado que los de las consolas, al estar disponibles para más personas; existen comunidades dedicadas a su distribución, se los anuncia por internet y con un solo clic se los puede descargar. Hay videojuegos online que tienen mucha popularidad, existiendo un sinnúmero de tiendas digitales para comercializarlos y darlos a conocer.

1.1.2.1 Videoconsolas aproximándose a su fin.

En la actualidad, se puede afirmar que las consolas desaparecerán tal como

²¹ Sitio web de financiamiento masivo para proyectos creativos, se presentan proyectos y metas a cumplir .

²² Tienda de software en línea desarrollada por Google.

se las conoce [9]. El día de hoy se cuenta con numerosas tiendas online, en las que desarrolladores de todo tipo comparten sus creaciones; existe una gran cantidad de juegos gratis, que sirven usualmente para promocionar al desarrollador. Los videojuegos se los baja legalmente, con ayuda de las tiendas digitales. Onlive²³ ha revolucionado el mundo con su sistema, no es necesario tener una máquina poderosa para acceder a sus juegos, son totalmente dependientes del servidor en el que se alojen; esto junto con la llegada de los juegos sociales, Smartphones y últimamente tablets están dando mucha más competencia a la industria. Próximamente se verá una gran generación de videojuegos, que se puedan jugar en línea y en las redes sociales [10].

1.1.2.1.1 Next-Gen de Videoconsolas.

Si se da un breve vistazo a foros de internet, muchos coinciden en que la próxima generación de consolas podría ser la última [9], Xbox 720, Wii U y PlayStation4. Debido principalmente a que las tablets están ganando mucho mercado; algunas como el iPad3 están equipadas con tecnología que en algunos puntos es superior incluso a consolas como la PlayStation3. Esto permite esbozar sin mucha dificultad que Microsoft y Sony terminarán por acoplarse a las nuevas plataformas existentes [9]. Pero el futuro lo dirá, compañías como Ubisoft han descartado totalmente esta posibilidad, argumentan que el hardware tendrá que cambiar, pero la experiencia de jugar en casa no tendrá reemplazo [9]; lo único cierto es que con la revolución de adelantos tecnológicos y la tecnología móvil, las nuevas consolas tendrán que reinventarse para no dejar que se les acorte su mercado.

1.1.2.1.2 Streaming duro rival para las Videoconsolas.

Tecnologías como Onlive o Gankai han entrado en escena para el mundo de los videogamers²⁴, estas plataformas ofrecen la posibilidad de disfrutar

²³ Sistema de distribución de videojuegos bajo demanda, con videojuegos almacenados computados y renderizados en línea.

²⁴ Jugador de videojuegos que juega o se informa con gran dedicación e interés

de juegos vía streaming²⁵. Los videojuegos se pueden alquilar, o sólo probarlos gratis por 30 minutos [9]. No se necesita descargar, sólo se debe contar con una cuenta y el cliente. En un futuro no muy distante, cuando las velocidades de internet se incrementen se podrán contar con juegos de última generación totalmente funcionales vía streaming. Esto es a lo que apunta el futuro, todo en la actualidad se está orientando a que las aplicaciones estén disponibles en la nube.

1.1.2.2 Evolución de los Gráficos.

Desde la aparición de la Saturn de Sega, los videojuegos cambiaron su concepto, antes se programaban con píxeles y en 2D; desde entonces los videojuegos son programados con gráficos poligonales²⁶ texturizados²⁷ y en 3D [11] [12]. A partir de ese momento, la cantidad de polígonos que manejan las consolas se ha multiplicado, con el paso del tiempo, principalmente por el hardware cada vez más potente. La PlayStation o la PSone manejaban algunos centenares de polígonos. La PlayStation 2 hasta 75.000 polígonos sin texturizar y 20.000 texturizados; en la actualidad las consolas de la Next-Gen procesan más de medio millón de polígonos [11] [12]. El número de polígonos es importante, aporta calidad a los gráficos, pero dependen mucho del procesamiento del GPU²⁸ que tenga cada consola; así que para evitar costo de procesamiento se puede simular la presencia de polígonos texturizando los gráficos o añadiendo sombras²⁹. Este punto es muy importante, se debe observar el número de polígonos para que el juego pueda procesarse de manera correcta, en la mayoría de equipos. Al día de hoy existen videojuegos, por ejemplo para la

²⁵ Distribución multimedia a través de una red de computadoras, el usuario consume el producto ,al mismo tiempo que lo descarga de manera continua sin interrupción con ayuda de un buffer de datos que almacena datos y los muestra al usuario.

²⁶ Son Figuras que la computadora almacena con posición de puntos, líneas y caras, construyendo un polígono, en un espacio de 3 dimensiones

²⁷ Se refiere a las imágenes colocadas sobre una o más caras de un polígono

²⁸ Graphics Processing Unit o Unidad de Procesamiento de Figuras, es un coprocesador dedicado al procesamiento de Figuras u operaciones de coma flotante para aligerar la carga de trabajo del procesador central.

²⁹ Shading o sombreado implica la simulación de computadora del comportamiento de las caras de un polígono cuando son iluminadas por una fuente de luz virtual.

PlayStation3, que contienen 20.000 polígonos. La Tabla 1.6, contiene el número de polígonos que se ha usado para algunos videojuegos.

Año	Consolas	Juegos	Objeto- Personaje	Número de Polígonos
1995	Psone	Tekken	Personajes	300
1996	Psone	Tekken II y III	Personajes	400 a 500
1998	PC	Half Life	Zombie	844
2000	DreamCast	Half Life	Zombie	1.949
2002	Xbox	Mortal Kombat: Deadly Alliance	Personajes	7000 a 10000
2003	Game Cube	The Legend of Zelda: The Wind Waker	Link	2.800
2004	PC	Half Life II	Alyx Vance	8.323
2005	PS2	Metal Gear Solid 3	Snake	4.000
2005	Game Cube	Resident Evil 4	Leon	10.000
2006	Xbox 360	Project Gotham Racin 3	Coches	80.000 a 100.000
2007	Xbox 360	Lost Planet: Extreme Condition	Wayne	17.765
2008	Xbox 360	Gears of War 2	Marcus	15.000
2008	PS3	Metal Gear Solid 4	Snake	60.000 a 80.000
2010	PS3	God of War III	Kratos	20.000
2011	PS3	Uncharted 3	Nathan	30.000
2012				
2013	PS4	Kill Zone Shadow Fall	Helghast	40.000

Tabla 1.6 Uso de polígonos en los videojuegos [8] [13] [14]

La Tabla 1.7, contiene las características de las videoconsolas (no portátiles) que han sido creadas a partir de la quinta generación.

Consolas							
Compañía	Saturn	PSX	DreamCast	PS2	Xbox	Game Cube	
Generación	Sega Quinta 1994	Sony Quinta 1994	Sega Sexta 1999	Sony Sexta 2000	Microsoft Sexta 2001	Nintendo Sexta 2001	
Año de lanzamiento	1994	1994	1999	2000	2001	2001	
CPU	2x32 bit RISC SH2 28.6 y 20 Mhz	32 bit RISC a 33.8 Mhz	32 bit Hitachi SH4 RISC a 200 Mhz	128 bits RISC MIPS-IV R5900	Intel Coppermite Core 32 bit a 733 Mhz	PowerPC Gekko a 485 Mhz 32 y 64 bits	
GPU	2 procesadores VDP1 y VDP2	No disponible	128 bit PowerVR2DC a 100 Mhz	Dos unidades que realizan operaciones vectoriales	NV2A MPCX 256 bit a 233-250 mhz	ATI Technologies a 162 Mhz	
Memoria RAM	2 MB	2 MB	16 MB	32 MB	64 MB	40 MB	
Memoria Video	1.5 MB	1 MB	8 MB	4 MB	Opcional dentro de los 64 MB	24 MB	
Memoria Sonido	0.5 MB	0.5 MB	2 MB	4 MB	Opcional dentro de los 64 MB	16 MB	
Polígonos/Segundo	500.000	1.5 millones	7 millones	2.4 millones	125.000		
Polígonos Textura Mapeada/S	250.000	360.000	3.5 millones	1.2 millones	4 millones	12.millones	
Formato Juego	CD-ROM	CD-ROM	GD-ROM	DVD-Sony	DVD	Mini-DVD Panasonic	

Tabla 1.7 Consolas de Sobremesa 1. [8] [11] [12]

Consolas	Xbox 360		Wii		PS3	
Compañía	Microsoft		Nintendo		Sony	
Generación	Séptima		Séptima		Séptima	
Año de lanzamiento	2005		2006		2007	
CPU	PowerPC Tri-Core a 3.2 Ghz IBM Xenon		IBM Broadway a 729 Mhz		Cell BroadbandEngine a 3,2 Ghz con 1PPE y 7 SPEs(Synergistic Processing Elements)	
GPU	ATI Xenos a 500 Mhz		ATI Hollywood a 243 Mhz		NVIDIA/SCEI RSX a 550 Mhz 128 bit	
Memoria RAM	512 MB de RAM GDDR3		24 MB 1T-SRAM		256 MB XDR	
Memoria Video	Incluye 10 MB dentro del GPU EDRAM		1 MB		256MB GDDR3	
Mandos Inalámbricos	Kinect		Wiimote		PlayStation Move	
Polígonos/Segundo Mapeada/S	500 millones		1.500 millones		7.000 millones	
Formato Juego	DVD y DVD doble capa		Wii Optical Disc, Nintendo GameCube Game Disc		Blu-ray Disc	

Tabla 1.7 (Continuación) Consolas de Sobre mesa 2. [8] [11] [12]

Con estos datos y en base a la capacidad que tiene la mayoría de equipos actualmente, se ha estimado que los personajes del videojuego estarán en un rango de 3.000 a 4.000 polígonos, con una resolución en texturas de 1024, medida que es usual en juegos independientes (no cuentan con financiamiento) y en aplicaciones móviles.

1.1.3 INDUSTRIA DESARROLLADA EN EL PAÍS

Según un artículo, tomado del diario El Comercio el 8 de enero del año 2011 titulado “El desarrollo de los videojuegos en el país tiene un freno por la piratería” [1], los videojuegos no han tenido mayor crecimiento en nuestro país por una falta de inversión, promoción y principalmente la piratería. La industria de los videojuegos no tiene un mercado sólido para desarrollarse, aún así existen compañías ecuatorianas que han logrado vender sus productos en el exterior, se verá una de estas compañías y como ha logrado sobrevivir.

1.1.3.1 Blue Lizard Games. [18]

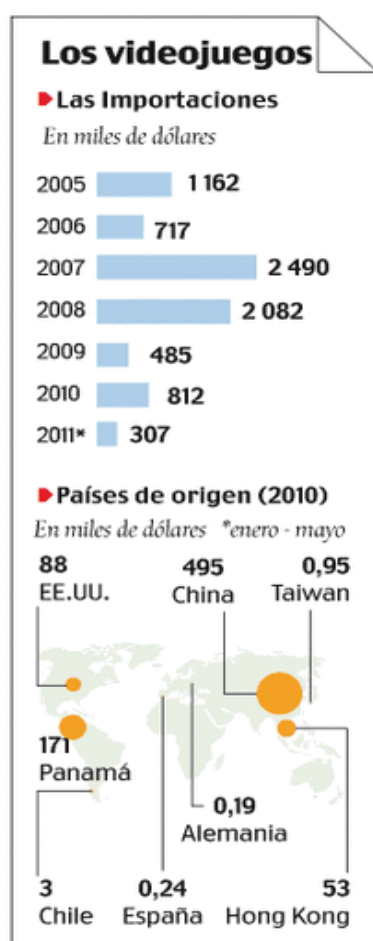
Esta empresa fue creada en el 2008, Blue Lizard Games o BLG, por Alfredo Chaves y Nathalie Gauthier; apostándole a los videojuegos ya tienen a su haber 35, su juego insignia es Nanoland³⁰, que ya cuenta con más de 140.000 usuarios a escala mundial en la red social Facebook. Los juegos que desarrollan son bajo pedido, mayoritariamente desde Europa y Estados Unidos y representan el 80% de los ingresos de la compañía. BLG es el único desarrollador aprobado por Nintendo en Ecuador y uno de los pocos en Latinoamérica [15]. Actualmente BLG ofrece arte y diseño en videojuegos para PC, MAC, Smartphones, tablets, redes sociales y las videoconsolas DS y Wii. El equipo humano está compuesto por 40 personas, los videojuegos pasan por distintas etapas: preproducción, producción y mantenimiento, con tiempos que dependen del cliente de acuerdo al tamaño del videojuego. Están trabajando con tecnologías de reconocimiento de movimientos para pantallas interactivas y en el

³⁰ Videojuego MMO(Masive Multiplayer Online) desarrollado por Blue Lizard Games de Ecuador para Santillana Ecuador, para niños de 6 a 14 años

lanzamiento de Nanoland para Estado Unidos. Según la gerente general y fundadora de BLG Nathalie Gauthier, en la entrevista realizada para el diario Hoy, en Ecuador “falta el conocimiento de los procesos generales para que el producto tenga éxito. A esto se suma el poco interés de empresas y agencias de publicidad para hacer conocer, para difundir sus servicios a través de estas plataformas que tienen gran alcance” [15] .

1.1.3.2 Análisis de la Industria en el Ecuador

En Ecuador, en el área de desarrollo de software, se destacan sistemas de tipo comercial, la mayoría dirigidos a mejorar procesos en entidades bancarias y públicas [16]; pero en el desarrollo de videojuegos no se ha logrado plasmar una verdadera industria. De los pocos desarrolladores que



Fuente: Banco Central del Ecuador

existen la mayoría lo hace simplemente por pasatiempo. Nuestro país no cuenta con ofertas académicas, los que realmente persiguen esta meta, se ven obligados a buscar este tipo de oferta académica en otros países como Argentina y España. Sin embargo en el área multimedia, el país goza de grandes proyectos que se los puede considerar como pequeños desarrollos.

No existe difusión acerca de la rentabilidad que puede llegar a tener esta industria, alrededor del mundo es muy buena. Esto queda demostrado, tan sólo al ver las estadísticas de las importaciones de videojuegos, en el reportaje del diario El Comercio hasta principios del año 2011. Actualmente en países hermanos como Colombia se brinda apoyo a esta industria, a través del Ministerio de Comunicaciones [17], con el objetivo de consolidar este tipo de

empresas nacionalmente primero y luego expandirlas por el continente [1].

Los ingresos de ventas por el software para las consolas fueron de 8,9 mil millones de dólares, con 189 millones de unidades vendidas [18]; son cifras astronómicas que demuestran la rentabilidad de este tipo de industria que debería ser apoyada en el Ecuador. La principal carencia que se observa es la falta de conocimiento de las empresas locales acerca de los beneficios que puede conllevar esta industria; con ventajas como la difusión de los videojuegos y su alcance a nivel global, se podrían implementar estrategias de mercadeo para dar a conocer ciudades, fomentar el turismo, el conocimiento, etc. Hoy en día la piratería, la falta de ofertas académicas, el desconocimiento de la rentabilidad de esta industria, ha mermado el crecimiento en el país de empresas que se dediquen específicamente a la creación de videojuegos [1].

1.2 SELECCIÓN DE LA METODOLOGÍA PARA EL DESARROLLO DE VIDEOJUEGOS

Existen numerosas metodologías³¹ que inciden en distintas dimensiones en el proceso de desarrollo. Las Metodologías tradicionales, se centran en el control del proceso, estableciendo rigurosamente actividades que se deben cumplir, los artefactos³² a producirse, las herramientas y notaciones que se usarán. A pesar de ser efectivas, en muchos casos han demostrado deficiencias; por otro lado están las metodologías ágiles, que se centran específicamente en el factor humano y no en el producto de software para el proceso de desarrollo [19]. Al dar mayor valor al individuo, se explota su habilidad de creación no limitándola; las metodologías ágiles han revolucionado en cierta medida la forma de producir el software.

1.2.1 DESCRIPCIÓN DE LAS METODOLOGÍAS ÁGILES

Las metodologías ágiles surgieron a principios de los 90, estando en contra de la creencia de que con procesos totalmente definidos se lograba un software a tiempo, con el costo y la calidad requerida. El primer enfoque de estas

³¹ En ingeniería de software es el marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información.

³² Es un producto tangible resultante del proceso de desarrollo de software por ejemplo: casos de uso, diagramas de clase, etc.

metodologías, se dio a conocer por Martin, en la comunidad de Ingeniería de Software con el nombre de RAD o Rapid Application Development; establecía grupos pequeños de programadores que generaban código de forma automática, tomando como entradas sintaxis de alto nivel; pero no se las apreciaría hasta que Kent Beck crea XP o eXtreme Programming, lo que dio inicio al movimiento de metodologías ágiles [19]. El término ágil surge después, antes la mayoría de desarrolladores las consideraban meramente intuitivas; de esta manera nace la alternativa para los procesos tradicionales de desarrollo de software [20]. El resultado de esta tendencia fue el Manifiesto Ágil³³ y sus principales ideas son:

- Los individuos y las interacciones entre ellos son más importantes que las herramientas y los procesos empleados.
- Es más importante crear un producto software que funcione, a escribir documentación exhaustiva.
- La colaboración con el cliente debe prevalecer, sobre la negociación de contratos.
- La capacidad de respuesta ante un cambio es más importante que el seguimiento estricto de un plan.

La Tabla 1.8 describe las características de las metodologías ágiles, contrastándolas con las características de las metodologías tradicionales; con esto se puede apreciar las diferencias existentes entre los dos tipos de metodologías.

³³ Extracto de http://es.wikipedia.org/wiki/Manifiesto_ágil

Metodologías	Tradicionales	Ágiles
Características	Se basan en normas provenientes de estándares seguidos por el entorno de desarrollo	Se basan en heurísticas provenientes de prácticas de producción de código
	Resistencia a cambios	Preparadas para cambios en el proyecto
	Procesos controlados con numerosas políticas y normas.	Procesos menos controlados con pocos principios
	Contrato no flexible	Contrato flexible
	El cliente interacciona con el equipo de desarrollo mediante reuniones	El cliente es parte del equipo de desarrollo
	Grupos grandes y posiblemente distribuidos	Grupos pequeños y trabajando en el mismo sitio
	Más artefactos y roles	Pocos artefactos y roles
	La arquitectura de software es esencial y se expresa mediante modelos	Menos énfasis en la arquitectura de software

Tabla 1.8 Diferencias entre Metodologías Tradicionales y Ágiles [20]

La principal diferencia en el proceso de desarrollo, es la agilidad en los procesos de documentación, al ser más livianos se ahorra tiempo y recursos; lo que es conveniente para la elaboración de un videojuego, en el que se cuenta con un número limitado de personas; además la corrección de errores implicaría un costo alto. Con esta aclaración se hace una breve descripción de algunas metodologías ágiles y cómo funcionan sus procesos de desarrollo de software. Se ha seleccionado las siguientes metodologías, para su análisis:

- ✓ XP.
- ✓ AUP.
- ✓ SCRUM.
- ✓ Crystal Methodologies.
- ✓ FDD.

A continuación, se presenta un resumen de cada metodología enfocada al modo de desarrollo de software y se establece una tabla de criterios de selección. Con el resultado obtenido se procede al desarrollo del videojuego sobre la metodología seleccionada.

1.2.1.1 XP- eXtreme Programming [20]

Es la primera metodología ágil, que de la mano de Kent Beck le dio conciencia al movimiento actual de metodologías ágiles. Esta metodología

se diferencia de las tradicionales, principalmente al dar más énfasis en la adaptabilidad que en la previsibilidad, considerando que los cambios de requisitos sobre la marcha son un aspecto natural.

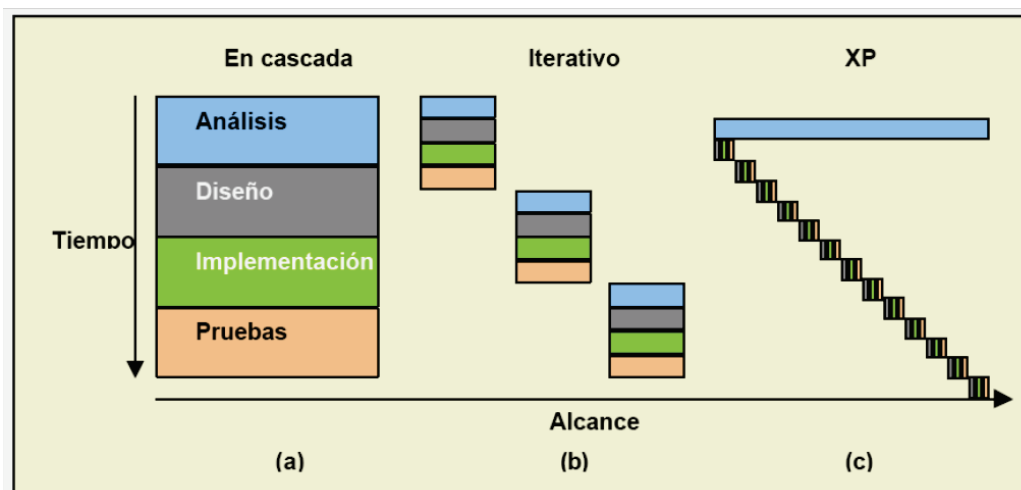


Figura 1.1 Evolución de los largos ciclos de desarrollo en cascada (a) a ciclos iterativos más cortos (b) y la mezcla que hace XP. [20]

El ser capaz de adaptarse a los cambios de requisitos en cualquier etapa del ciclo de vida de un producto, es una aproximación más realista que la que se intenta definir con todos los requisitos al inicio de un proyecto.

XP consta de 5 valores:

Simplicidad, necesaria para refactorizar el código, de esta manera se asegura que si el proyecto crece, el código se mantenga simple y todo el equipo conozca el sistema completo.

Comunicación, va de la mano con la simplicidad, para que la comunicación sea fluida, se establece la programación por parejas, el cliente forma parte del equipo de trabajo.

Retroalimentación (feedback), con el cliente formando parte del equipo de desarrollo, su opinión sobre el estado del proyecto se conoce en tiempo real; los ciclos cortos ayudan a obtener un mejor feedback; además de las pruebas unitarias que son realizadas, se hacen revisiones del sistema existente para evitar cambios futuros.

Respeto, se debe respetar el trabajo de todos los miembros del equipo, de ellos en conjunto depende la calidad del producto que se desarrolle; con

esto se crea una mejor autoestima y un elevado ritmo de producción.

Sus características principales son:

- Desarrollo iterativo e incremental, se realizan pequeñas mejoras una tras otra.
- Pruebas unitarias continuas, incluyen pruebas de regresión, se hacen de manera repetida y automatizada.
- Esta dirigido por historias de usuario.
- Programación por parejas, las tareas de desarrollo son llevadas a cabo por dos personas en un mismo puesto. De esta manera el código es revisado y discutido mientras se lo escribe.
- Frecuente interacción, que existe entre el equipo de programación y el cliente, ya que forma parte del equipo de desarrollo.
- Corrección de errores, se lo hace antes de añadir una nueva funcionalidad; es decir que existen entregas frecuentes.
- Propiedad del código compartida, todo el personal puede corregir y extender cualquier parte del proyecto.
- Simplicidad de código.
- Tolerancia a cambios.
- Equipo de desarrollo de 2 a 10 personas.

1.2.1.2 AUP – Agil Unified Process [21]

Esta metodología es una versión dinámica del Proceso Unificado de Desarrollo (RUP) de IBM³⁴, basada en disciplinas y entregables incrementadas con el tiempo. Su primera versión fue lanzada en el año 2005. Lo primero que se distingue es el cambio de disciplina, este modelo abarca:

- Modelado de Negocio.
- Implementación.
- Prueba.
- Despliegue.

³⁴ International Business Machines (IBM) (NYSE: IBM), empresa multinacional de tecnología y consultoría fundada en 1911

- Administración de la configuración.
- Administración o gerencia del Proyecto.
- Entorno.

Se puede apreciar las fases del AUP en la Figura 1.2.

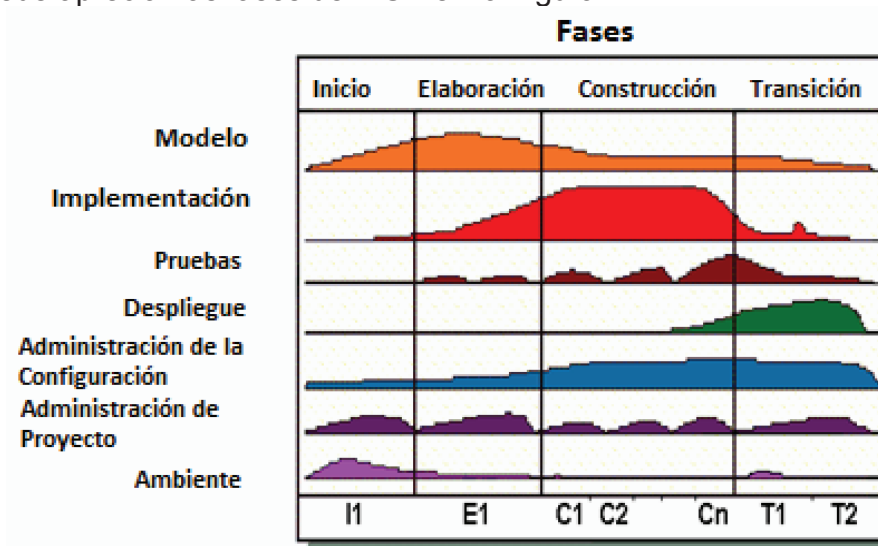


Figura 1.2 Ciclo de vida del Proceso Unificado Ágil (AUP) [34], traducido por [61]

Las fases en las que se desarrollan las iteraciones son 4:

Inception o Principio, su meta es identificar el alcance inicial del proyecto, su arquitectura potencial y obtener la financiación y aceptación de todas las partes involucradas.

Elaboration o Elaboración, su meta es proveer la arquitectura del sistema.

Construction o Construcción, su meta es la construcción del software de manera gradual y que este responda a las necesidades del cliente.

Transition o Transición, su meta es validar e implementar el sistema en un entorno de producción.

Sus características principales son:

- Esta dirigido por casos de uso.
- Es iterativo e Incremental.
- Puede ser ampliamente escalable.
- Se puede tener releases en cada iteración.
- Trabaja con diseño UML para los casos de uso.

- Los artefactos se presentan de acuerdo al tipo de proyecto.
- Es tolerante a cambios.
- Equipo de desarrollo de 4 a 25 personas.

1.2.1.3 Scrum [22] [23] [24]

Esta metodología se realiza de manera iterativa e incremental, en donde cada ciclo o iteración también conocido como “Sprint”, termina con una pieza de software ejecutable con cierta funcionalidad; que cada iteración o “Sprint” proporciona un resultado completo que se lo entrega con un mínimo esfuerzo. Cada iteración dura de 2 a 4 semanas y utiliza como marco otras prácticas como RUP o XP.

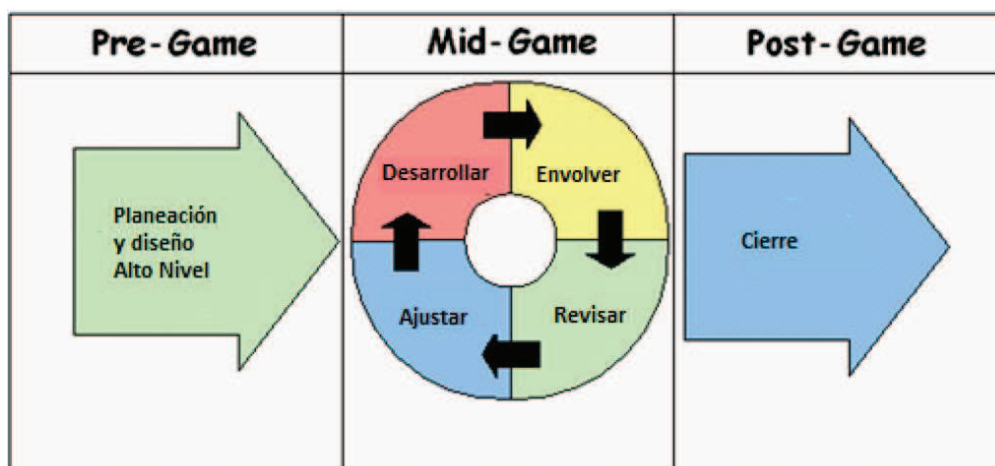


Figura 1.3 Diseño General de SCRUM ³⁵ [22], Traducido por [8]

El proceso de desarrollo de SCRUM consta de 3 fases como se puede apreciar en el Figura 3, que son:

- 1) Pre-Game: Planificación y Montaje.
- 2) The Game: Ciclo del “Sprint” y
- 3) Post-Game: Cierre.

Las principales características del proceso de desarrollo en SCRUM son:

- Es iterativo e incremental, con esto podemos mitigar el riesgo.
- El producto avanza en una serie de “Sprints”

³⁵ Figura tomado de <http://www.codeproject.com/Articles/4798/What-is-SCRUM>

- Los requisitos son capturados como elementos de una lista de “Product Backlog”
- El tiempo de cada “Sprint“, depende del número de elementos del “Product Backlog”.
- No hay prácticas de ingeniería prescritas para esta metodología.
- El proceso de desarrollo es flexible y adaptable.
- Es fácil de aprender y requiere poco esfuerzo para empezar a utilizarlo.
- Los equipos que se forman en SCRUM son auto-organizados, impulsan la comunicación verbal entre todos los miembros y disciplinas involucrados en el proyecto.
- Tolerancia a cambios.

Cada Sprint consta del siguiente ciclo:



Figura 1.4 El ciclo de un Sprint ³⁶ [25]

Cada ciclo empieza con reuniones para planificar el “Sprint”, lo que se debe realizar en cada “Sprint”. Termina con una reunión de revisión del “Sprint” en donde se lo revisa y es ajustado al proyecto. El ciclo es repetido hasta que el producto este completo, se desarrolla, envuelve, revisa y ajusta.

³⁶ Figura tomado de <http://www.codeproject.com/Articles/4798/What-is-SCRUM>

1.2.1.4 Crystal Methodologies [26]

Crystal Clear es la metodología ágil de la que más documentación se posee. Propone más énfasis en la comunicación, manejando iteraciones cortas con feedbacks frecuentes por parte de los usuarios/clientes; minimizando de esta forma la necesidad de productos intermedios. La familia Crystal dispone de un código de color, para marcar la complejidad de una metodología, cuánto más oscuro, más pesado es el método. La Figura 1.5 esquematiza la criticidad del sistema de acuerdo al tamaño del proyecto, ambos directamente proporcionales.

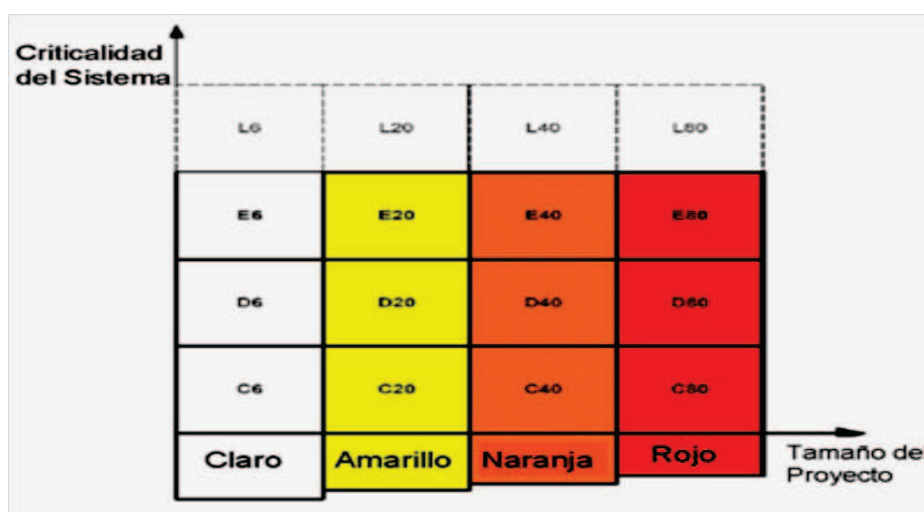


Figura 1.5 Familia de Crystal Methods [26]

Los métodos son llamados Crystal, evocando las facetas de una gema, existen 4 variantes de metodologías: Crystal Clear para equipos de 8 o menos integrantes; Amarillo para 8 a 20 integrantes; Naranja para 20 a 50 integrantes; Rojo para 50 a 100 integrantes, siguen con los colores marrón, Azul y Violeta. Crystal Clear posee 7 valores:

- Entrega frecuente, el software a los clientes es entregado de forma frecuente; puede ser diaria, semanal o mensualmente.
- Comunicación osmótica, todos en un mismo cuarto discuten los diferentes puntos el producto.
- Mejora Reflexiva, tiempo en el que se toman notas se reflexiona y discute sobre lo que se debe hacer.

- Seguridad Personal, los miembros del grupo poseen total libertad de opinar sobre lo que creen que está mal o se puede mejorar.
- Foco, saber lo que se hace y tener la tranquilidad y tiempo para hacerlo.
- Fácil acceso a usuarios expertos, el contacto con los expertos del desarrollo del proyecto es importante, el equipo regularmente cuenta con un experto en negocios.
- Ambiente Técnico con prueba automatizada, administración de configuración e integración frecuente.

Crystal Clear se basa en una explotación refinada de los inconvenientes de los modelos clásicos, enfatiza el proceso como un conjunto de ciclos anidados en la mayoría de Proyectos, se establecen 7 ciclos de la siguiente forma:

- (1) Del proyecto.
- (2) De entrega de la unidad.
- (3) De la iteración.
- (4) De la semana laboral.
- (5) Del periodo de integración (de 30 minutos a 3 días).
- (6) Del día de trabajo.
- (7) Del día de trabajo. Del episodio de desarrollo de una sección de código, de pocos minutos a pocas horas

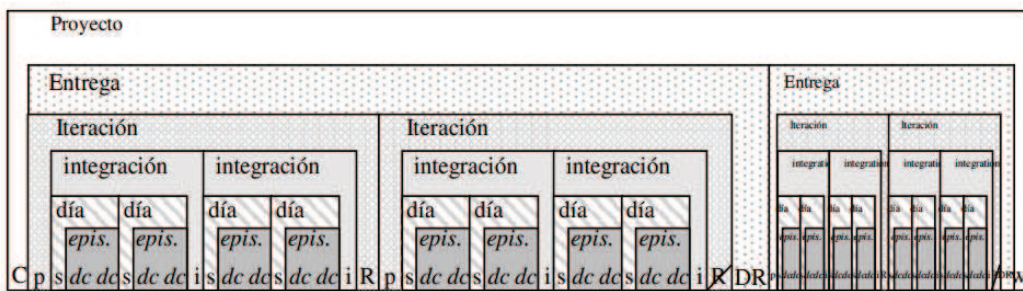


Figura 1.6 Ciclos anidados de Crystal Clear [26]

Las letras denotan Chartering, planeamiento de la iteración, reunión diaria de pie (standup), desarrollo, check-in, integración, taller de Reflexión, Entrega (Devilery), y empaquetado del proyecto (Wrapup).

Sus características principales son las siguientes:

- Equipo de trabajo depende de la dificultad del proyecto.
- Las metodologías cristal se adaptan según el proyecto de desarrollo.
- Es de desarrollo iterativo – incremental.
- Tolerancia a cambios.
- Genera documentación aceptable.

1.2.1.5 FDD- Feature Driven Development [26] [27]

Esta metodología está basada en FOP (Feature Oriented Programming), técnica de programación guiada por rasgos y centrada en el usuario. FDD es un método ágil, iterativo y adaptativo que no cubre todo el ciclo de vida, sino las fases de diseño y construcción; es adecuada para proyectos grandes con dimensión crítica. Se lo puede considerar como una mezcla intermedia entre RUP y XP. Se basa en un proceso iterativo, con iteraciones cortas.

Sus principios son:

- Se requiere un sistema para construir sistemas, si se pretende escalar a proyectos grandes.
- Un proceso simple y bien definido trabaja mejor.
- Los pasos de un proceso deben ser lógicos y su mérito obvio para cada miembro del equipo.
- Vanagloriarse del proceso, puede impedir el trabajo real.
- Los buenos procesos van hasta el fondo del asunto, de modo que los miembros del equipo se puedan concentrar en los resultados.
- Los ciclos cortos, iterativos y orientados por rasgos son mejores.

FDD consta de 5 procesos secuenciales, durante los cuales se construye y se diseña el sistema, cada fase tiene un criterio de entrada, tareas, pruebas y un criterio de salida; las iteraciones de un rasgo duran de 1 a 3 semanas.

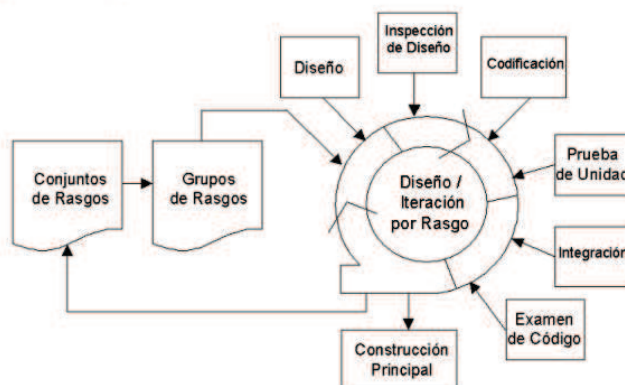


Figura 1.7 Ciclos de FDD [26]

Las fases de FDD son:

- Desarrollo de un modelo general.
- Construcción de una lista de rasgos o funcionalidades.
- Planeamiento de releases por funcionalidades.
- Diseño en base a los rasgos.
- Construcción por rasgo.

Y sus características principales son las siguientes:

- Es de desarrollo iterativo e incremental.
- Posee un desarrollo orientado a objetos.
- Es de carga ligera, para el equipo de desarrollo.
- Se lo puede modelar con UML.
- Tolerancia a cambios.
- El código fuente que es generado tiene propietario.
- Los equipos de trabajo varían, en función de los rasgos a implementar.
- El conocimiento sobre el producto desarrollado se reparte a través del trabajo, en equipo y revisiones.
- Genera documentación aceptable.

1.2.2 DISEÑO DE CRITERIOS DE EVALUACIÓN [8]

Para el diseño de los criterios de evaluación de las metodologías ágiles, se ha tomado en cuenta el conocimiento adquirido de cada una de ellas.

Los criterios de evaluación están dados en gran medida por este factor. Los

criterios que se han seleccionado en relación con el grupo de desarrollo con el que se cuenta (2 personas) son los siguientes:

Criterio de Evaluación		Peso(%)
1	Grado de Conocimiento	20
2	Soporte orientado a objetos	10
3	Adaptable a cambios	15
4	Basada en historias de usuario o casos de uso	10
5	Documentación adecuada	10
6	Facilita integración de las etapas de desarrollo	10
7	Relación con UML	10
8	Permite el desarrollo de software sobre cualquier tecnología	15
Total		100

Tabla 1.9 Criterios de Evaluación para las Metodologías Ágiles [61]

Como se observa en la Tabla 1.9, el Grado de Conocimiento de la metodología ocupa el mayor porcentaje, junto con la Adaptabilidad a cambios y Permitir el desarrollo de software sobre cualquier tecnología. El Grado de conocimiento es importante ya que enfatiza la experiencia o dominio sobre determinada metodología. La Adaptabilidad a cambios, permite mitigar riesgos y soportar los continuos cambios de requisitos que se presenten. Finalmente Permitir el desarrollo sobre cualquier tecnología, da la pauta para saber si la metodología se puede adaptar al videojuego en un proceso de desarrollo. Los demás criterios son una especie de características comunes, que la mayoría de metodologías poseen. El porcentaje total llega al 100 %, sumando el peso de cada criterio de selección.

1.2.3 EVALUACIÓN Y SELECCIÓN DE LAS METODOLOGÍAS

	Peso (%)	20	10	15	10	10	10	10	15	100
Metodologías	Criterios	1	2	3	4	5	6	7	8	Total
<i>XP</i>		15	10	15	10	6	7	6	10	79
<i>AUP</i>		15	10	8	10	10	10	10	10	83
<i>SCRUM</i>		19	8	15	10	10	8	8	15	93
<i>Crystal Methodologies</i>		10	10	12	8	10	10	10	12	82
<i>FDD</i>		5	10	10	8	10	8	8	12	71

Tabla 1.10 . Evaluación y Selección de Metodologías [61]

Notación:

- (1) Grado de Conocimiento
- (2) Soporte orientado a objetos
- (3) Adaptable a cambios
- (4) Basada en historias de usuario o casos de uso
- (5) Documentación adecuada
- (6) Facilita integración de las etapas de desarrollo
- (7) Relación con UML
- (8) Permite el desarrollo de software sobre cualquier tecnología

El resultado de la selección favorece a SCRUM, es la mejor metodología adaptada a los criterios de selección propuestos. Se hace un análisis más detallado a continuación.

1.2.3.1 Scrum [22] [23] [24]

Antes de iniciar cada iteración, se debe revisar las tareas pendientes y seleccionar la parte que se entregará como un incremento de funcionalidad en cada iteración o sprint.



Figura 1.8 Ciclo del Sprint-Incremento de funcionalidad [24]

Las inspecciones se las realizan cada 24 horas, teniendo como referencia el Product Backlog o pila de requerimientos; con esto se planifica cada iteración. La metodología SCRUM cuenta con tres roles, también conocidos como comprometidos³⁷: Product Owner, Scrum Master y Scrum Team.

Product Owner, representa a los interesados en el producto final, marca las prioridades del producto, lleva el control de estimaciones y el Retorno de Inversión (ROI).

³⁷ Hace referencia a la analogía de SCRUM gallinas y cerdos, en la cual la gallina propone abrir un restaurante al cerdo, con los productos, huevos y jamón, pero el cerdo no acepta ya que el está directamente comprometido, mientras la gallina sólo se encuentra implicada.

Scrum Master, responsable del proceso de SCRUM, incorpora SCRUM en la cultura de la organización, asegura el cumplimiento de roles y responsabilidades, provee la formación y el entrenamiento en el proceso.

Scrum Team, se encarga de transformar las tareas del Sprint Backlog en un incremento de funcionalidad de software, desarrolla el producto con calidad (auto-organizado, auto-gestionado, multifuncional). La Figura 1.9 muestra el flujo que sigue SCRUM.

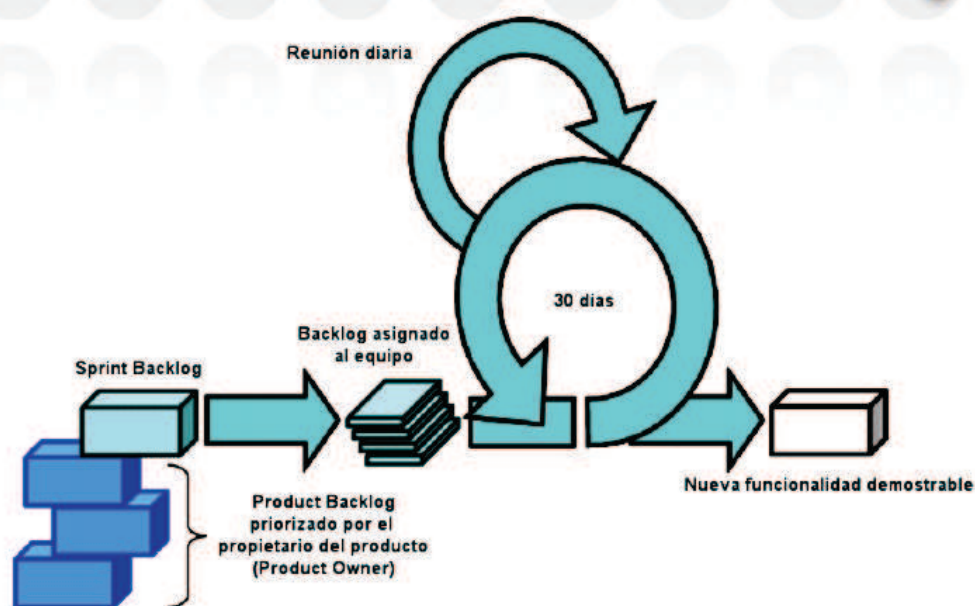


Figura 1.9 Flujo de Scrum [37]

Sprint es el tiempo en el cual se desarrolla un incremento de funcionalidad, tiene una duración máxima de 30 días. Durante este tiempo el Sprint Backlog no puede modificarse. Se puede desechar un sprint cuando la tecnología seleccionada no funciona, cuando se cambia el negocio o cuando se tiene varias interferencias; esta es tarea de Scrum Master. Los artefactos que se producen en este ciclo son: Sprint Backlog, Product Backlog y la Gráfica del proceso.

Product Backlog, es un listado con los requisitos del sistema; es mantenido y priorizado por el Product Owner durante todo el ciclo de vida, es un documento dinámico que incorpora constantemente las necesidades del sistema.

La Figura 1.10 muestra un ejemplo de Product Backlog:

Product Backlog			Trabajo pendiente			
ID	Elemento	Estimación inicial Completado Estim. ajustada	Sprint			
			1	2	3	4
1	Nuevo formulario para peticiones de clientes	2 0,2 2,4	2,4	0	0	0
2	Configuración de respuestas automáticas	3 0,2 3,6	3,6	0	0	0
3	Envío automático de respuestas	1 0,2 1,2	1,2	0	0	0
4	Consulta para los clientes de peticiones enviadas	1 0,2 1,2	1,2	0	0	0
5	Modificación del cliente de sus peticiones enviadas	2 0,2 2,4	2,4	0	0	0
6	Acceso a peticiones sólo para clientes del portal jurídico	5 0,2 6	6	0	6	0
7	Consulta de peticiones por parte del staff	1 0,2 1,2	1,2	0	0	0
SPRINT 1		15	18	18	0	0
8	Inserción de comentarios y reasignación a peticiones (staff)	2 0,2 1,2	1,2	1,2	0	0
9	Consultas por clientes, fechas y temas	3 0,2 3,6	3,6	3,6	0	0

Figura 1.10 Ejemplo de Product Backlog [37]

Sprint Backlog, es el listado de tareas realistas, extraídas del Product Backlog, que serán convertidas en un incremento de funcionalidad. Cada tarea debe tener una duración entre 4 y 16 hrs, si son de mayor duración se deben descomponer en subtareas.

Grafica de Progreso, sirve para ver el desarrollo de cada sprint en función del tiempo, horas vs días del sprint.

1.3 SELECCIÓN DE HERRAMIENTAS PARA EL DESARROLLO DE VIDEOJUEGOS

La selección de las herramientas para el desarrollo del videojuego es fundamental, de estas herramientas dependen las características que tendrá el videojuego de acuerdo a su funcionalidad, como son: video, sonido, jugabilidad, entre otras. Para este fin se debe realizar una evaluación del tipo de juego que se desea crear y la plataforma para la que va a ser creado.

1.3.1 AMBIENTE PARA EL DESARROLLO DE VIDEOJUEGOS

El ambiente para el desarrollo de videojuegos, se refiere al conjunto de herramientas y requisitos necesarios para su correcta ejecución en un ordenador. De acuerdo a lo acotado en la introducción del subtema se evalúa las siguientes herramientas:

- ✓ Motor Gráfico.
- ✓ Herramientas de Modelado y Animación 3D.

- ✓ Herramientas de Diseño 2D.
- ✓ Herramientas adicionales al modelado.

1.3.1.1 Motor Gráfico [28]

El Motor Gráfico provee las rutinas³⁸ necesarias para representar cada uno de los objetos con su lógica; controla, gestiona y actualiza los modelos 3D en tiempo real. Contiene una colección de librerías con elementos que entre otras cosas permiten: renderizar³⁹ imágenes, realizar movimientos, incorporar animaciones, manejar colisiones⁴⁰, añadir sonidos, añadir efectos visuales, inteligencia artificial, entre otros, es decir en el Motor Gráfico se implementa la funcionalidad del videojuego.

Consideraciones Técnicas:

- **Physics:** Se encarga del manejo de las colisiones entre objetos dentro del motor gráfico, maneja la gravedad de los cuerpos, las estructuras de los terrenos.
- **Shaders:** Son elementos que permiten iluminar ciertas partes de los objetos tridimensionales, por ejemplo: espadas, armaduras, ojos, etc.
- **Iluminación:** Se diferencia de los shaders por ser aplicada a nivel global, modifica el ambiente del videojuego.
- **Manejo de Terrenos:** El motor debe proporcionarnos esta opción, los terrenos son creados desde el propio motor o incorporados como objetos. Los terrenos o terrains forman la ambientación y detalle del videojuego.
- **Uso de IA:** El motor gráfico debe permitir incorporar IA (inteligencia Artificial), realizando acciones que se puedan activar dependiendo de variables dadas.
- **Soporte para distintas consolas:** Debe tener portabilidad, es decir que se pueda correr el videojuego en distintas plataformas o consolas.

³⁸ Son series de instrucciones que permiten al computador llevar a cabo una tarea

³⁹ Render en inglés, se refiere al proceso de generar una imagen desde un modelo, interpreta una escena en 3 dimensiones y la plasma en una imagen bidimensional

⁴⁰ Se refiere a la simulación del choque entre dos objetos a los que se les da propiedades como por ejemplo la gravedad,

- **Manejo de Sonido Stereo:** El sonido es parte importante del videojuego, el motor debe tener la capacidad para incluir sonido.

1.3.1.2 Descripción de Motores Gráficos más usados [29]

Según la revista develop online, www.develop-online.net estos son los Motores Gráficos que se encuentran dentro del top 10:

1. Unreal Engine 3. [29] [30]

Creado por la compañía Epic Games, inicialmente implementado en el shooter en primera persona Unreal, ha sido la base de juegos como Gears of War, BioShock, entre otros. Está escrito en C++ y es compatible con varias plataformas: Microsoft Windows, GNU/Linux, Mac OS; y las consolas: Wii, Xbox, Xbox360, PS2 y PS3. La tercera generación de este motor aparece en el 2006, diseñado para PC con soporte DirectX⁴¹ 9/10, Xbox 360 y PlayStation3. Soporta tecnologías como HDRR (High Dynamic Range Rendering), creando imágenes mucho más realistas; normal mapping (técnica 3D que permite dar iluminación y relieve detallado a la superficie de un objeto) y sombras dinámicas. Cuenta con el UDK (Unreal Development kit⁴²), que permite a grupos de desarrolladores amateur realizar juegos.


Motor	Características	
	Desarrollado por:	Epic Games
	Plataformas Soportadas	Xbox 360, PS3
	Lenguajes que usa	C++
	Juegos Desarrollados	Gears of War, BioShock
	Costo	UDK gratis para desarrolladores novatos, Unreal Engine costo bajo pedido.
	Tecnología soportada	HDRR, Normal Mapping
	Licencia	Propietaria, Libre.
	Sistemas Operativos	Mac OS, GNU/Linux, Windows

Tabla 1.11 Características de Unreal Engine 3. [29] [30] [8]

⁴¹ Es una colección de API desarrolladas para facilitar las tareas relacionadas con multimedia, especialmente programación de juegos y video sobre Microsoft.

⁴² Versión sin costo del Motor Unreal Engine para promover el desarrollo de novatos sobre este motor.

2. Gamebryo 4.0 [29] [31]

Creado por la compañía Emergent Games Technologies, Gamebryo motores 3D y LigthSpeed son propiedad actualmente de Gamebase Co. Está basado en un conjunto de módulos de bibliotecas en C++ que soportan un proceso de desarrollo iterativo incremental. Actualmente soporta una amplia variedad de plataformas: DirectX9, 10, 11, OpenGL⁴³/Linux, MAC y las consolas: PS2, PS3 Xbox360 y Wii. Este motor es ampliamente robusto, se complementa con el diseño de paisajes del motor LigthSpeed de la misma compañía. Cuenta con varias herramientas como plugins⁴⁴ para optimización de gráficos, editores de escenas, niveles, etc.


Motor	Características	
	Desarrollado por:	Gamebase EEUU & Gamebase Ltd
	Plataformas Soportadas	Xbox 360, PS3, Wii, PC
	Lenguajes que usa	C++
	Juegos Desarrollados	Fallout 3, Civilization Revolution, Dragonica
	Costo	Costo bajo respuesta
	Tecnología soportada	Optimizador de escenas y mallas.
	Licencia	Propietaria
	Sistemas Operativos	Windows, GNU/Linux, Mac OS

Tabla 1.12 Características de Gamebryo [29] [31] [8]

3. CryEngine 3 [29] [32]

Desarrollado por Crytek para Microsoft, PlayStation 3, Xbox 360. Obtuvo el premio en el 2009 a la mejor simulación en tiempo real; este motor de videojuego cuenta con una versión completa gratuita para fines no comerciales. Fue utilizado por primera vez en el juego para PC Far Cry como una demostración de Nvidia⁴⁵. Ubisoft adquiere todos sus derechos de propiedad en el 2006; actualmente está disponible para diversas

⁴³ Open Graphics Library, define una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan Figuras en 2D y 3D

⁴⁴ O complemento es una aplicación que se relaciona con otra para agregarle nueva funcionalidad.

⁴⁵ Empresa Multinacional especializada en el desarrollo de unidades de procesamiento Figura.

plataformas con uso de DirectX 9, 10 y 11, iOS, Android⁴⁶. Este motor permite a varios desarrolladores trabajar sobre una misma capa sin que haya conflictos en su trabajo, la vegetación que se rige por reglas y parámetros naturales como altura y densidad; además de usar streaming para agrupar datos y así obtener un acceso rápido en cada nivel del videojuego. Es sin duda uno de los mejores motores, pero tiene un gran costo computacional.


Motor	Características	
	Desarrollado por:	Crytek
	Plataformas Soportadas	Xbox 360, PS3, Wii, PC, MMO and next-gen ready
	Lenguajes que usa	C++
	Juegos Desarrollados	Crysis, Aion, Crysis Warhead
	Costo	Gratis sin fines comerciales, bajo respuesta
	Tecnología soportada	Optimizador de escenas y mallas.
	Licencia	Propietaria, libre
	Sistemas Operativos	Windows, iOS, Android.

Tabla 1.13 Características de CryEngine 3 [29] [32] [8]

4. Unity 3D [29] [33]

Desarrollado por Unity Technologies, disponible para plataformas como DirectX10, 11, Direct3D, iOS, Android y consolas: Xbox 360, PlayStation 3, Wii. Cuenta con un browser game⁴⁷, que es usado para el despliegue de widgets⁴⁸ como en la Mac, explotando las capacidades 3D de Adobe Flash pero también con sus limitaciones. Cuenta con un ambiente integrado primario de desarrollo, similar a otros engines como Blender para modelar objetos en 3 dimensiones. Ganó el Premio a la Innovación tecnológica del Wall Street Journal 2010. Soporta integración con otras herramientas como 3ds, Maya o Blender, de una manera sencilla se puede importar objetos desde este motor.

⁴⁶ Sistema operativo móvil basado en Linux

⁴⁷ Juego de ordenador que puede ser reproducido a través de internet usando un navegador web

⁴⁸ Es una pequeña aplicación o programa, usualmente usados para dar fácil acceso a funciones, presentado en archivos o ficheros que son ejecutados por un motor de widgets o Widget Engine


Motor	Características	
	Desarrollado por:	Unity Technologies
	Plataformas Soportadas	Xbox 360, PS3, Wii
	Lenguajes que usa	C++, C#
	Juegos Desarrollados	Tiger Woods,
	Costo	Gratis sin fines comerciales Unity, Unity Pro licencia comercial bajo respuesta
	Tecnología soportada	bump mapping ⁴⁹ , reflection mapping ⁵⁰ , parallax mapping ⁵¹
	Licencia	Propietaria, Libre
	Sistemas Operativos	Windows, Mac OS, iOS, Android

Tabla 1.14 Características de Unity [27] [31] [8]

5. *BlitzTech* [29] [34]

Desarrollado por Blitz Game Studios, de Reino Unido, alcanzando los best seller con 3 millones de ventas en esa región gracias a su serie Dizzy. La empresa tiene numerosas divisiones, que siguen y elaboran líneas distintas, videojuegos, arcades, programas de formación relacionados con la medicina, programas de extensión educativa; cuenta con el programa Blitz1UP que ayuda a desarrolladores independientes a llevar sus videojuegos al mercado, proporciona ayuda y asesoramiento gratuito en aspectos de producción de videojuegos. Este motor de videojuego ha ganado premios como Nickelodeon Kids Choice 2007- Bob esponja, ITC Excellence in Skills, Develop Industry Excellence Award Bussines Development, entre otros. Compatible con las consolas: PS3, Xbox 360, Wii, PSP, PSVita⁵²; y para las plataformas: Mac OSX, iOS, Android.

⁴⁹ Técnica en computación gráfica para simular golpes y arrugas en la superficie de un objeto

⁵⁰ En computación gráfica es una eficiente técnica de iluminación basada en imagen, para aproximar la apariencia de una superficie reflectante por medio de una imagen de textura pre-calculada.

⁵¹ También llamado mapeo de offset, o mapeo de desplazamiento virtual, es una mejora del bump mapping, las texturas tienen profundidad mucho más evidente lo que les da más realismo

⁵² Sucesora de la PSP, Play Station Portable, presentada el 27 de enero del 2011


Motor	Características	
	Desarrollado por:	Blitz Game Studios
	Plataformas Soportadas	Xbox 360, PS3, Wii, PSP Vita
	Lenguajes que usa	
	Juegos Desarrollados	House of the Dead: Overkill, Power Up Forever, Dead to Rights Retribution
	Costo	Gratis sin fines comerciales Blitz1UP, Licencia comercial bajo respuesta
	Tecnología soportada	3DTV, Normal, Parallax, Cube ⁵³ y Refraction Mapping. HDR, Motions Blur ⁵⁴
	Licencia	Propietaria, Libre
	Sistemas Operativos	Windows, Mac OS, iOS, Android

Tabla 1.15 Características de BlitzTech [29] [34] [8]

1.3.1.3 Herramientas de Modelado y Animación 3D [35]

Las herramientas de modelado y animación 3D, son las responsables de la creación de los objetos o elementos visuales de un videojuego. Estas herramientas permiten crear abstracciones tridimensionales de los objetos, que pueden ser usados o incorporados posteriormente en el motor gráfico.

Consideraciones Técnicas:

- Uso de **multitextura** en los objetos tridimensionales, como su nombre lo indica la multitextura es la capacidad de dar diferentes tipos de textura o múltiples texturas a la superficie de un objeto tridimensional (personajes, suelo, muebles, entre otros.), la herramienta debe contar con esta capacidad.
- **Compatibilidad con el motor gráfico** seleccionado, la herramienta debe ser compatible con el motor gráfico para que los objetos modelados puedan interactuar sobre la plataforma.
- **Armatures y animaciones**; el armature es el objeto básico de una animación, es como el esqueleto de una criatura viva pero para el objeto o personaje; está formado por un conjunto de huesos que pueden ser

⁵³ Método de mapeado que usa un cubo de 6 caras sobre las que se pueden almacenar diferentes texturas.

⁵⁴ O desenfoque de movimiento, desenfoque que se produce en la imagen por el movimiento a gran velocidad

independientes o no de otros huesos y son los que determinan como serán los movimientos del personaje. Las animaciones se las obtiene con el movimiento de los huesos; se pueden visualizar secuencias de movimientos en líneas de tiempo, programar movimientos como resultado de una acción del usuario, entre otros. La herramienta de animación y modelado debe contar con esta capacidad.

1.3.1.4 Descripción de Herramientas de Modelado y Animación 3D más usadas

Herramientas de modelado y animación 3D, es el software que nos permite realizar la creación tridimensional de piezas, objetos o estructuras, que según el ámbito van presentando diferentes características, por ejemplo para arquitectura, diseño multimedia, diseño industrial, cine, medicina, entre otros; en este caso el modelado de objetos para la creación de los personajes que conformarán el videojuego. A continuación se presenta una breve descripción de las herramientas de Modelado y Animación 3D en la actualidad las más usadas.

Autodesk 3ds Max [36]



Desarrollado por Autodesk, salió a la venta por primera vez en 1990 para DOS⁵⁵. A partir de ese año se han producido varias versiones hasta llegar a la actual, siendo ésta la primera versión que puede instalarse en sistemas operativos de 32 o 64 bits, con una alta gama de elementos para el modelado tridimensional. Posee variaciones para la creación de objetos, contando con la posibilidad de escoger entre polígonos, splines⁵⁶ y curvas NURBs⁵⁷. Su amplia gama de herramientas la han hecho la herramienta favorita de muchos diseñadores, por su realismo; un ejemplo de esto lo podemos ver en la película Transformers. En lo referente a

⁵⁵ Familia de sistemas operativos para PC, disk operating system, creado originalmente para computadoras IBM

⁵⁶ Es una curva diferenciable definida en porciones mediante polinomios

⁵⁷ En inglés de Non-Uniform Rational B-spline, es un modelo matemático utilizado en computación gráfica para generar y representar curvas de superficies.

videojuegos Autodesk 3ds Max posee características que hacen mucho más fácil la creación de objetos, el texturizado y la animación; es tan popular que varios motores de videojuegos poseen plugins para conectarse directamente con 3D Max. El costo de la licencia es elevado.

Autodesk Maya [37]



Herramienta desarrollada por Autodesk, dedicado al desarrollo de gráficos 3D, efectos especiales y animación. Surgió inicialmente como Power Animator de la fusión de las empresas Alias y Wavefront, pero finalmente fue absorbido por Autodesk con el nombre de Maya en enero del 2006. Posee MEL (Maya Embedded Language) que es el código que forma el núcleo de la herramienta, se pueden crear scripts y personalizar paquetes. Esta herramienta en particular ha causado un gran impacto en la industria cinematográfica siendo único software de 3D acreditado con un Oscar⁵⁸, por su capacidad de efectos visuales, se pueden observar sus efectos en películas como Jurassic Park, Terminator 2, entre otras. Es altamente personalizable gracias al kit de desarrollo que posee, se encuentra abierto para modificaciones permitiendo la integración y creación de plugins escritos en C++ o Python. En cuanto al modelado de objetos, trabaja con NURBs, polígonos, subdivisión de superficies. Ayuda a la realización de videojuegos permitiendo la pre visualización de los objetos creados y la construcción de objetos con un número bajo de polígonos necesarios para bajar el nivel de procesamiento.

Cinema 4D [38]



Desarrollado originalmente para Commodore Amiga por la compañía alemana Maxon. Cuenta con soporte para varias plataformas: Windows, Linux y Macintosh.

⁵⁸ Se refiere a los premios cinematográficos otorgados anualmente cada mes de febrero o marzo por la Academia de las Artes y de las Ciencias Cinematográficas en los Ángeles California

Permite modelar con primitivas, splines y polígonos; gracias a las herramientas con las que cuenta para el modelado de personajes se puede acoplar a cualquier proyecto de animación con facilidad, lo que es muy necesario en el desarrollo de un videojuego, por esta razón es ampliamente recomendable usarla. Se puede personalizar su interfaz, gracias a esto el aprendizaje de la herramienta es sencillo. Es de licencia propietaria y cuenta con varios módulos para añadir iluminación, gravedad y efectos físicos, armatures de personajes, cabello, clonación de objetos entre otros.

Blender [39]



Es una herramienta de código abierto creada para realizar animaciones 3D, desarrollado originalmente por el estudio holandés NeoGeo; al caer en banca rota se la ofreció como un producto de código abierto de licencia GNU⁵⁹ GPL⁶⁰, con esto se creó la Fundación Blender. Cuenta con soporte para varias plataformas: Windows, Mac OSX, Linux, Solaris, FreeBSD⁶¹ e IRIX. Su interfaz gráfica es ampliamente intuitiva, con ciertas ventajas al presentar distintos tipos de ventanas para cada vista, lo que le da al modelador una mejor perspectiva del modelo que está siendo creado; en el modelado los objetos son representados por superficies, polígonos, curvas NURBs, metaballs con la posibilidad de añadir distintos tipos de materiales sobre estos objetos. Cuenta además con detección de colisiones, efectos de iluminación y textura, sistemas de partículas para la simulación de cabellos y pelajes y un sinnúmero de propiedades en los shaders para lograr texturas más realistas. Por todo esto y por contar con un código abierto, es una muy buena opción para los desarrolladores independientes de videojuegos.

⁵⁹ Proyecto iniciado por Richard Stallman con el objetivo de crear un sistema operativo completamente libre

⁶⁰ Licencia Pública General de GNU, creada por la Free Software Foundation, orientada a proteger la libre distribución, modificación y uso de software

⁶¹ Sistema operativo para arquitecturas x86, derivado de BSD versión de Linux desarrollada por la Universidad de California

ZBrush [40]



Esta herramienta ha sido desarrollada por Pixologic ZBrush, para modelado 3D, textura y pintura digital. Empezó siendo un programa que permitía insertar pinturas digitales e insertar objetos 3D. Fue usado para realizar la fase beta de los personajes del señor de los anillos alcanzando gran popularidad. Los modelos se esculpen pintando sobre un objeto 3D, de esta manera se realiza el modelado, su licencia es propietaria pero se puede descargar sin costo registrándose en la página de Pixologic. Cuenta con una gran comunidad ZbrushCentral, que desarrolla plugins para la herramienta haciéndola mucho más versátil y poderosa. Está disponible para las plataformas Windows y Macintosh.

1.3.1.5 Herramientas de Diseño 2D [41]

En un videojuego 3D no sólo existen objetos tridimensionales, también los hay en dos dimensiones; se pueden crear objetos sencillos con sprites⁶². Son usados para los menús, diseño de estados de los personajes como por ejemplo para representar la vida del personaje, el puntaje, es decir complementan las imágenes del videojuego 3D.

Consideraciones Técnicas:

- **Manejo de capas**, las capas sirven para superponer gráficos añadiendo información, por ejemplo, es importante que se pueda superponer fotogramas en el videojuego para la elaboración de texturas.
- **Nivel de detalle de resolución**, que se cuente con una resolución aceptable y con gráficos vectoriales es imprescindible dentro de una herramienta de diseño 2D.

⁶² En ingles duendecillos, son un tipo de mapa de bits dibujados en la pantalla del computador por un hardware Figura, generalmente usados en videojuegos para producir una animación de un personaje corriendo o una expresión facial por su facilidad para cambiar colores y eliminar fondos

- Incorporación de **brochas prediseñadas**, permiten pintar con la forma de pinceles, lápices, entre otros, usando distintos tipos de texturas para la elaboración de los bocetos.

1.3.1.6 Descripción de Herramientas de diseño 2D más usadas

Estas herramientas son las encargadas de la representación de imágenes, tanto en el diseño gráfico, como en el diseño publicitario; las figuras tienen tratamiento plano, alto y ancho; los colores son planos y existe una textura visual, sin perspectivas ni volumen. Son importantes para crear objetos como partes de paisajes, barras de acción, las texturas de los personajes, que carecen de perspectiva y son incorporados en los videojuegos. Es necesario contar con herramientas que permitan diseñar estos objetos, a continuación se hace una breve descripción de algunas de las herramientas de diseño 2D más usadas.

InkScape [42]



Es un editor de gráficos vectoriales⁶³ de código abierto y multiplataforma: Linux, Windows, MAC OS X, usa el estándar W3C⁶⁴ con el formato de archivos Scalable Vector Graphics (SVG). Soporta formas, trazos, texto, marcadores, clones, mezclas de canales, transformaciones de gradientes, entre otros. Con capacidad para importar en distintos formatos Postscript⁶⁵, EPS⁶⁶, JPEG, PNG y TIFF y exportar en PNG. Se inició en el 2003 como una bifurcación del proyecto Sodipodi⁶⁷, es totalmente compatible con los estándares XML(Extensible Markup Language), SVG(Scalable Vector Graphics) y

⁶³ Imagen digital formada por objetos geométricos independientes (segmentos, polígonos, arcos, etc.) definidos por distintos atributos matemáticos de forma, posición, color, etc. Con esto se puede ampliar el tamaño de la imagen sin perder calidad

⁶⁴ The World Wide Web Consortium W3C, comunidad internacional que desarrolla estándares que aseguran el crecimiento de la web a largo plazo.

⁶⁵ Es un lenguaje de descripción de páginas usado de manera usual como formato de transporte de archivos Figuras.

⁶⁶ PostScript Encapsulado, formato de archivo Figura.

⁶⁷ Editor de Figuras vectoriales libre con licencia GNU GPL, discontinuado su última versión salió en el 2

CSS(Cascade Style Sheets), cuenta con herramientas para mover, rotar, escalar y estirar. Se pueden ajustar ángulos, tramas, guías y objetos de nodos; los objetos también pueden agruparse, desagruparse y duplicarse.

Adobe Photoshop [43]



Trabaja en forma de taller de pintura y fotografía sobre un lienzo destinado para la edición y retoque fotográfico. Su nombre literalmente significa tienda de fotos y es el programa de edición de imágenes más famoso del mundo. Se lo usa en diseño web, composición de bitmaps, edición de video y en cualquier actividad que requiera el tratamiento de imágenes digitales; trabaja con capas. Soporta formatos: PSD, PDD, PostScript, EPS, BMP, GIF, PNG, JPEG, TIFF (Tagged Image File Format), PDF, entre otros. Para gráficos vectoriales es aconsejable usar Adobe Illustrator. Es de licencia propietaria Adobe CLUF, sólo funciona sobre Mac y Windows.

GIMP [44]



GNU Image Manipulation Program, es un programa de edición de imágenes digitales de mapa de bits. Libre y gratuito. Forma parte del proyecto GNU y está disponible bajo licencia GPL. Es multiplataforma disponible para sistemas operativos Unix, GNU/Linux, FreeBSD, Solaris, Windows y Mac OS X. Se inicio en 1995 por Spencer Kimball y Peter Mattis, procesa gráficos y fotografías digitales; se usa para la creación de logos, modificación de fotografías, conversión de formatos; trabaja con capas y se lo usa incluso para edición de video. Soporta los formatos JPG, GIF, PNG, PCX, TIFF y posee su propio formato XCF, además importa imágenes vectoriales SVG. Cuenta con herramientas de selección, tijeras, modificadores de escala, filtros para manipulación de colores, menú con catálogos y efectos de imágenes.

CorelDraw [45]



Es una aplicación informática de manejo vectorial altamente intuitiva, usada para dibujo, maquetación de páginas para impresión o publicación web. Fue oficialmente lanzado en el año de 1989 diseñado por Michel Bouillon y Pat Beirne de Corel Corporation. Soporta tanto vectores como pixeles y textos, para diseñar logotipos, afiches, volantes, letreros, etc. Permite crear botones para aplicaciones de diseño web, gracias a plugins que incluyen software de otras marcas. Soporta una gran cantidad de formatos AI(formato de Adobe Illustrator), BMP, CAL, CDR, SVG, RTF, PPT, JPG, etc., es software propietario disponible para la plataforma Windows.

Autodesk AutoCAD [46]



Programa de Diseño Asistido por Computadora (CAD) para dibujo en 2 y 3 dimensiones, comercializado por la empresa Autodesk. Su primera aparición fue en 1982, convirtiéndose hoy en día en uno de los programas más usados por arquitectos, ingenieros y diseñadores industriales. Procesa imágenes de tipo vectorial, pero también admite incorporar archivos de tipo fotográfico o mapa de bits. Organiza los objetos mediante capas o estratos, ordenando el dibujo en partes independientes con diferente color y grafismo. Este programa está orientado principalmente, a la producción de planos, pero también se puede modelar objetos; está disponible para plataformas Windows y Mac OSX con licencia de tipo propietario.

1.3.2 EVALUACIÓN Y SELECCIÓN DE HERRAMIENTAS [8]

Consideraciones de Uso para las Herramientas:

El presente proyecto de titulación tiene la finalidad de promover el desarrollo de videojuegos, fortaleciendo la industria naciente en futuro cercano, por

esta razón las herramientas seleccionadas deben ser baratas, con abundante información disponible sobre su uso y fáciles de manejar; se ha tomado en cuenta las siguientes características para su selección:

- ***Precio de licenciamiento.***
- ***Información Disponible.***
- ***Usabilidad.***

1.3.2.1 Motores Gráficos

Para el diseño de los criterios de evaluación de los Motores Gráficos, además de las consideraciones de uso para las herramientas acotadas, se toma en cuenta otros criterios. El criterio principal es el *Precio de licenciamiento de la herramienta*, estas herramientas son bastante costosas por lo general. Los criterios de selección establecidos se presentan en la Tabla 1.16.

Criterio de Evaluación		Peso(%)
1	Precio de licenciamiento	20
2	Información Disponible	15
3	Usabilidad	15
4	Efectos	10
5	Perspectivas de Diseño	10
6	Importación de objetos de otras herramientas	10
7	Bajo costo computacional	20
Total		100

Tabla 1.16 Criterios de Evaluación de los Motores Gráficos. [8]

Con mayor peso se encuentra el *Precio de licenciamiento*, y el *Bajo costo computacional*. La *Información Disponible* tiene un peso de 15, éste criterio permite conocer de mejor forma la herramienta para darle un mejor uso. La *Usabilidad* un peso de 15, la presentación de la interfaz de la herramienta debe ser muy intuitiva para un fácil manejo. Los *Efectos* aportan mayor elegancia al juego y le dan versatilidad, este criterio tiene un peso de 10; la mayoría de motores incorporan los mismos efectos como manejos de colisiones, iluminación, mapeado y los espectaculares como las vistas

estereoscópicas en 3D están fuera de alcance por su elevado precio para usuarios y desarrolladores. Las Perspectivas de Diseño, ayudan a visualizar de mejor manera la acción al permitir observar los objetos desde distintos ángulos en una misma pantalla, tiene un peso de 10. La importación de objetos de otras herramientas, permite trabajar con programas modeladores y de imágenes externos e insertar objetos para usarlos en el motor, tiene un peso de 10. El Bajo costo computacional es importante, se busca que el juego pueda ejecutarse en la mayoría de equipos y ser desarrollado en ordenadores accesibles, por esta razón tiene un peso de 20.

1.3.2.2 Selección de Motores Gráficos

	Peso (%)	20	15	15	10	10	10	20	100
Herramientas	Criterios	1	2	3	4	5	6	7	Total
Unreal Engine		19	12	12	10	8	9	5	75
Gamebryo		10	10	11	9	9	9	14	72
CryENGINE3		19	8	13	10	9	9	5	73
Unity 3D		20	14	13	8	8	9	15	87
BlitzTech		20	10	12	8	9	9	14	82

Tabla 1.17 Evaluación y Selección de los Motores Gráficos [8]

Notación:

- (1) Precio de licenciamiento
- (2) Información Disponible
- (3) Usabilidad
- (4) Efectos
- (5) Perspectiva de diseño
- (6) Importación de Objetos de otras herramientas
- (7) Bajo costo computacional

El resultado como se aprecia en la Tabla 1.17 favorece a Unity 3D, como el motor gráfico que mejor se adapta a los criterios de evaluación.

1.3.2.3 Herramientas de Modelado y Diseño 3D

Para el diseño de los criterios de evaluación de las Herramientas de

Modelado y Diseño 3D, se ha tomado en cuenta las consideraciones de uso, la portabilidad y la creación de objetos y armatures, en la Tabla 1.18 se presentan los criterios de evaluación.

Criterio de Evaluación		Peso(%)
1	Precio de licenciamiento	20
2	Información Disponible	15
3	Usabilidad	15
4	Portabilidad	20
5	Creación de objetos y Armatures	20
6	Incorporación de multitextura	10
Total		100

Tabla 1.18 Criterios de Evaluación de las herramientas de Modelado y Animación 3D. [8]

El mayor peso esta dado por el Precio de licenciamiento de la herramienta, la Portabilidad y Creación de objetos y armatures con un peso de 20; la portabilidad por ser necesaria para exportar objetos y usarlos en el motor gráfico; las armatures por permitir animar a los objetos. La Información Disponible junto con la Usabilidad tienen un peso de 15. La incorporación de multitextura tiene un peso de 10, sirve para añadir colores y arrugas a los objetos y darles más realismo.

1.3.2.4 Selección de Herramientas de Modelado y Diseño 3D

Herramientas	Criterios	Peso (%)						Total
		1	2	3	4	5	6	
Autodesk 3ds Max		10	11	12	7	8	10	58
Autodesk Maya		10	13	11	8	9	8	59
Cinema 4D		10	9	13	7	9	7	55
Blender		20	14	13	9	10	9	75
ZBrush		15	10	12	8	5	8	58

Tabla 1.19 Evaluación y Selección de las herramientas de Modelado y Animación 3D. [8]

Notación:

- (1) Precio de licenciamiento
- (2) Información Disponible
- (3) Usabilidad
- (4) Portabilidad
- (5) Creación de objetos y armatures
- (6) Incorporación de Multitextura

El resultado como se aprecia en la Tabla 1.19 favorece a Blender, como la herramienta de modelado 3d que mejor se adapta a los criterios de evaluación.

1.3.2.5 Herramientas de Diseño 2D

Para el diseño de los criterios de evaluación de las Herramientas de Diseño 2D, se ha tomado en cuenta las consideraciones de uso, la portabilidad y su información disponible

Criterio de Evaluación		Peso(%)
1	Precio de licenciamiento	20
2	Información Disponible	10
3	Manejo de capas	25
4	Nivel de detalle de resolución	20
5	Incorporación de brochas prediseñadas	15
6	Compatibilidad con otro software	10
Total		100

Tabla 1.20 Criterios de Evaluación de las Herramientas de Diseño 2D. [8]

El mayor peso esta dado por el Manejo de Capas con 25, que permite crear gráficos superponiéndolos ganando de esta manera mayor realismo. El Nivel de detalle de resolución junto con el Precio de Licenciamiento, tienen un peso de 20. La incorporación de brochas prediseñadas cuenta con un peso de 25, muy importantes para la definición de los gráficos creados. La información disponible y compatibilidad con otro software toman un peso de 20, permiten tener una mejor visión de la herramienta y que esta cuente con varios tipos de formatos que pueden ser transportados y usarlos con otro software.

1.3.2.6 Selección de Herramientas de Diseño 2D

		Peso (%)						100
Herramientas	Criterios	1	2	3	4	5	6	Total
<i>InkScape</i>		20	8	20	18	13	10	89
<i>Adobe Photoshop</i>		10	9	22	20	14	9	85
<i>GIMP</i>		20	8	21	18	13	10	90
<i>Corel Draw</i>		10	9	20	18	13	9	79
<i>Autodesk AutoCAD</i>		10	10	20	20	14	8	86

Tabla 1.21 Evaluación y Selección de las de las Herramientas de Diseño 2D. [8]

Notación:

- (1) Precio de licenciamiento
- (2) Información Disponible
- (3) Manejo de capas
- (4) Nivel de detalle de resolución
- (5) Incorporación de brochas prediseñadas
- (6) Compatibilidad con otro software

El resultado como se aprecia en la Tabla 1.21 favorece a GIMP, es la herramienta de diseño 2D que mejor se adapta a los criterios de evaluación.

1.3.2.7 Justificación del uso de Herramientas de Apoyo [8]

Para la creación de un videojuego además de las herramientas seleccionadas, es necesario usar herramientas adicionales, de esta manera se logra obtener mejores resultados; tanto para el modelado como para la creación de imágenes en 2d. Las herramientas adicionales a usarse son:

Sculptris [47]: esta herramienta permite realizar esculturas digitales. Se añade un objeto y literalmente se lo puede esculpir, gracias a la gran variedad de brochas creadas con este objetivo. Es útil al permitir crear posiciones básicas de los personajes de manera más sencilla y de esta forma estructurar fácilmente un objeto base (modelado con Blender), añadiendo detalles a los objetos modelados y realizando mezclas de color.

Se obtiene una mejor visualización de la estructura de los personajes mientras son creados. Además gracias a la posibilidad de pintar mapas UV⁶⁸ sobre los objetos, se mejora la pintura sobre el modelo.

Inkscape [42]: es una herramienta muy buena para la creación de iconos. Se la usará tanto como para realizar logos como para mejorar la calidad de los menús del videojuego. Al ser un editor de gráficos SVG libre que incorpora gran cantidad de formatos, es una buena opción para apoyarse con la generación de gráficos 2D para el desarrollo del videojuego.

IceSCRUM [48]: esta herramienta de software libre es una aplicación web, ayudará a manejar la metodología SCRUM durante todo el desarrollo del proyecto. Puede ser usada por varios usuarios al mismo tiempo, lo que es muy conveniente especialmente para la gestión de proyectos.

⁶⁸ Se refiere al proceso de representar una imagen 2D sobre un modelo 3D, las letras U, V se refieren a los ejes de la textura 2D

2. CAPÍTULO 2: FASE DE PREPRODUCCIÓN

2.1 CONCEPTUALIZACIÓN DEL VIDEOJUEGO

En la conceptualización del videojuego se define la idea general, las reglas, los objetivos y características funcionales. Se define un plan de desarrollo establecido con ayuda de la metodología seleccionada SCRUM. Esta etapa consta de dos diseños: el inicial y el detallado; estructurados por diferentes puntos que ayudan a la creación del videojuego. Se usa la herramienta de software libre iceSCRUM, que esta apegada a la metodología de desarrollo.

2.1.1 REQUERIMIENTOS FUNCIONALES DEL SISTEMA [8]

Se ha definido los requerimientos funcionales del sistema, basándose principalmente en la jugabilidad; es decir todo lo que el usuario es capaz de hacer con su personaje, todo con lo que el videojuego debe contar para que la jugabilidad se desarrolle de forma correcta. Teniendo en cuenta estos aspectos fundamentales se presenta los requerimientos funcionales del videojuego en la Tabla 2.1.

Nro	Requerimiento	Descripción	Prioridad
Re1	<i>Manejo del personaje</i>	Se refiere al movimiento que tendrá el personaje principal dentro del videojuego, controlado por el usuario.	Alta
Re2	<i>Jugabilidad</i>	La jugabilidad establece el modo del videojuego, es decir desde la cámara con la que se enfoca el personaje, su movimiento, el modo de interacción con el ambiente propuesto, hasta los movimientos de cada uno de los NPCs ⁶⁹	Alta
Re3	<i>Incorporar barras de estado del personaje</i>	Las barras de estado definen como se está desarrollando la acción del personaje principal dentro del videojuego, estas barras representan estados del personaje como la vida, la experiencia, etc.	Baja
Re4	<i>Incorporar recolección de objetos</i>	El personaje debe estar en la capacidad de recolectar los distintos objetos que se hayan puesto en el juego específicamente con este fin.	Media
Re5	<i>Manejo de sonidos e iluminación</i>	Son parte importante del desarrollo del videojuego, la iluminación permite a los objetos mostrarse de manera correcta ante la vista de los usuarios y los sonidos involucran el ambiente que se quiere recrear.	Media

⁶⁹ Non-player character, controlado por el juego. Generalmente en los juegos de rol, son los que pueblan el mundo, desde el vendedor de fideos hasta el enemigo para el player.

Re6	<i>Secuencia del juego</i>	La secuencia está definida con la historia, debe estar correctamente establecida teniendo temporalidad y espacialidad	Media
Re7	<i>Manejo de menús</i>	Los menús representan las opciones disponibles del videojuego, forman parte de las interfaces de usuario conteniendo las acciones importantes como empezar juego nuevo o modificar el video o sonido.	Baja
Re8	<i>Incorporación de NPCs</i>	Los NPCs son los personajes incorporados en el videojuego y realizan acciones por si solos, estas acciones representan en ocasiones grado de dificultad para los usuarios	Media

Tabla 2.1 Requerimientos Funcionales. [8]

Para establecer la metodología iterativa incremental con SCRUM se crea varios entregables, cada uno con cierta funcionalidad definida; de esta forma se inicia con el desarrollo. A partir de este punto se detalla cada unos de los entregables, 3 en total (R1, R2 y R3), los sprints con los que cuenta cada entregable, las tareas y subtareas con las que cuenta cada sprint.

2.1.2 PRIMER ENTREGABLE R1 - PRIMER SPRINT

Los requerimientos se han establecido para la correcta jugabilidad del videojuego en la Tabla 2.1, basándose en estos requerimientos globales se proceden a armar el primer Sprint del entregable R1. El primer Sprint recoge los aspectos iniciales, las tareas definidas para este Sprint se encuentran detalladas en la Figura 2.1.



Figura 2.1 R1-Primer Sprint Tareas [48]

Como se aprecia el Figura 2.1, el primer Sprint del entregable R1 consta de 9 tareas, cada una con subtareas; existen diferentes colores de acuerdo a las características que cumple cada tarea dentro del videojuego.

2.1.2.1 Principales características de R1 - Primer Sprint

Las características principales tomadas en cuenta para el R1- Primer Sprint están definidas en el Figura 2.2.



Figura 2.2 R1-Primer Sprint Características [48]

Estas tres características engloban aspectos generales en la construcción del videojuego, gracias a la herramienta se puede establecer colores que ayudan a distinguir cada una de las tareas, a que característica pertenecen; además se establece una relación con el cumplimiento de los requerimientos funcionales de la Tabla 2.2.

Nombre	Descripción	Requerimientos
Diseño de Interfaces	Las interfaces son muy importantes, deben ser muy intuitivas para que el usuario se sienta cómodo, mucho más si se trata de un videojuego. Proporcionan el entorno visual con el cual el usuario se comunica con la aplicación; en este caso un videojuego.	Re2 y Re7
Arte Conceptual	Es el conjunto de características que definen el aspecto del videojuego.	Re6
Construcción de personajes	Todo lo relacionado con la construcción del personaje desde los bocetos hasta el objeto 3D final.	Re8

Tabla 2.2 Detalle de Principales Características R1 [8] [48]

2.1.2.2 Roles

Se ha establecido todos los roles de Scrum, adaptados al equipo de desarrollo con el que se cuenta para elaborar la tesis, los roles se los aprecia en la Figura 2.3.

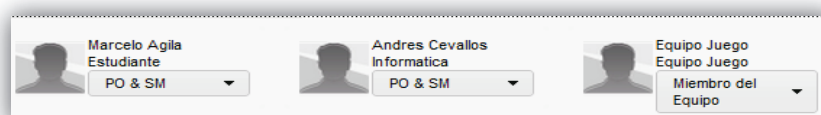


Figura 2.3 Detalle de Roles [48]

2.1.2.3 Detalle de las tareas en el Entregable R1-Primer Sprint

A continuación se detalla cada una de las tareas que están definidas en el primer Sprint del entregable R1. La herramienta IceSCRUM permite generar parte de la de documentación de la metodología, con historias de usuario. En cada sprint se detallan las tareas (ver Figura 2.2) definidas como historias, y las subtareas que la conforman (ver Tabla 2.1). Esto se repetirá en el desarrollo del presente documento, para cada sprint que forme parte de un entregable.

The screenshot shows a task card with the following details:

- Nombre:** Creación de la Historia del Videojuego
- Tipo:** Historia de usuario
- Clasificación:** 1
- Descripción:** Se define la trama de la historia, sobre la cual será realizado el videojuego. La historia debe contener las características principales de la trama en la que se desenvolverá el personaje principal.
- Notas:**
- Fecha de envío:** 04/07/2012 20:44:10
- Fecha de aceptación:** 05/07/2012 00:02:29
- Fecha de estimación:** 05/07/2012 00:08:21
- Fecha de planificación:** 05/07/2012 00:14:25
- En progreso desde:** 05/07/2012 21:45:55
- Fecha de terminación:** 05/07/2012 22:13:19
- Creador:** Marcelo Agila
- Característica:** Arte Conceptual

In the top right corner, there is a yellow box with the text: "1 Creación de la Hi... 1 Terminada".

Figura 2.4 Tarea Creación Historia del Videojuego [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Creación de la historia del videojuego	Reunión de Asociación	Terminada	La reunión se establece para la creación de la historia. Se obtiene ideas principales acerca de lo que puede servir y se seleccionan las que mejor se adapten al criterio común	Equipo Juego	06/07/2012	06/07/2012
	Redacción de la Historia	Terminada	Se redacta la historia resumiendo detalles, que se ampliarán en el guión	Equipo Juego	06/07/2012	06/07/2012

Tabla 2.3 Subtareas Creación Historia del Videojuego [8] [48]

Nombre : Diseño de Interfaz de Presentación	5
Tipo : Historia de usuario	Diseño de Interfa...
Clasificación : 2	1 Terminada
Descripción : la interfaz de inicio contiene los datos que corresponden al videojuego, como sus creadores, las herramientas con las que se desarrollo, el nombre del videojuego, etc. Se activa al iniciar el videojuego y dura pocos segundos	
Notas :	
Fecha de envío : 04/07/2012 20:54:19	
Fecha de aceptación : 05/07/2012 00:02:29	
Fecha de estimación : 05/07/2012 00:07:07	
Fecha de planificación : 05/07/2012 00:15:48	
En progreso desde : 05/07/2012 21:43:18	
Fecha de terminación : 06/07/2012 22:31:44	
Creador : Marcelo Agila	
Característica : Diseño de Interfaces	

Figura 2.5 Tarea Diseño Interfaz Presentación [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Diseño de Interfaz de Presentación	Escoger Fondo	Terminada	Se debe seleccionar colores que se puedan combinar	Marcelo Agila	07/07/2012	07/07/2012
	Reunión de asociación	Terminada	Se realiza la reunión para definir la forma que tendrá la interfaz	Marcelo Agila	06/07/2012	06/07/2012
	Redactar leyenda	Terminada	Se debe redactar lo que se va a presentar y el orden en que van a salir las imágenes	Equipo Juego	07/07/2012	07/07/2012
	Esbozar Diseño	Terminada	Se toman en cuenta factores como colores, ubicación de las leyendas, etc.	Equipo Juego	07/07/2012	07/07/2012

Tabla 2.4 Subtareas Diseño de Interfaz Presentación [8] [48]

8
Diseño de Interfa...
2 Terminada

Nombre : Diseño de Interfaz Pausa de Juego

Tipo : Historia de usuario

Clasificación : 3

Descripción : El modo de pausa se activa con un botón, en este modo el juego se detiene y la interfaz cambia a una pantalla que presenta varias opciones, como Continuar con el juego, cambiar resolución de video, bajar o subir el volumen.

Notas :

Fecha de envío : 04/07/2012 21:04:26

Fecha de aceptación : 05/07/2012 00:02:29

Fecha de estimación : 05/07/2012 00:07:18

Fecha de planificación : 05/07/2012 00:15:57

En progreso desde : 05/07/2012 21:43:18

Fecha de terminación : 06/07/2012 22:32:01

Creador : Marcelo Agila

Característica : Diseño de Interfaces

Figura 2.6 Tarea Diseño de Interfaz Pausa de Juego [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Diseño Interfaz Pausa de Juego	Reunión de asociación	Terminada	Se realiza la reunión para definir la forma que tendrá la interfaz	Marcelo Agila	06/07/2012	06/07/2012
	Esbozar Diseño	Terminada	Diseñar la interfaz de pausa. Esta se accede presionando un botón y congela las acciones del juego hasta que se pulse nuevamente	Equipo Juego	07/07/2012	07/07/2012
	Escoger controles	Terminada	Se refiere a los botones que los que el usuario puede acceder mientras el juego esta pausado.	Equipo Juego	07/07/2012	07/07/2012

Tabla 2.5 Subtareas Diseño de Interfaz Pausa de Juego [8] [48]

Nombre : Diseño de Interfaz Inicio

Tipo : Historia de usuario

Clasificación : 4

Descripción : La interfaz de inicio se activa inmediatamente después de la interfaz de presentación. Contiene el menú principal, con botones que en los que se puede seleccionar el inicio del juego, sonido, volumen, y setear el nombre del personaje.

Notas :

Fecha de envío : 04/07/2012 20:57:35

Fecha de aceptación : 05/07/2012 00:02:29

Fecha de estimación : 05/07/2012 00:07:14

Fecha de planificación : 05/07/2012 00:15:51

En progreso desde : 05/07/2012 21:43:18

Fecha de terminación : 06/07/2012 22:32:32

Creador : Marcelo Agila

Característica : Diseño de Interfaces

6
Diseño de Interfa...
2 Terminada

Figura 2.7 Tarea Diseño de Interfaz Inicio [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Diseño Interfaz Inicio	Reunión de asociación	Terminada	Se realiza la reunión para definir la forma que tendrá la interfaz	Marcelo Agila	06/07/2012	06/07/2012
	Esbozar Diseño	Terminada	Se toma en cuenta los controles que han sido seleccionados	Equipo Juego	07/07/2012	07/07/2012
	Escoger Controles	Terminada	Se refiere a los botones que los que el usuario puede acceder cuando se inicia el juego.	Equipo Juego	07/07/2012	07/07/2012

Tabla 2.6 Subtarear Diseño de Interfaz Inicio [8] [48]

Nombre : Diseño de la Interfaz Juego

Tipo : Historia de usuario

Clasificación : 5

Descripción : Se presenta al usuario cuando se encuentra jugando, contiene el personaje que es controlado por el usuario, el ambiente en el que se desarrolla la acción del player, una foto del personaje con una barra de vida.

Notas :

Fecha de envío : 04/07/2012 21:02:44

Fecha de aceptación : 05/07/2012 00:02:29

Fecha de estimación : 05/07/2012 00:07:12

Fecha de planificación : 05/07/2012 00:15:55

En progreso desde : 05/07/2012 21:43:18

Fecha de terminación : 06/07/2012 22:32:41

Creador : Marcelo Agila

Característica : Diseño de Interfaces

7

Diseño de la Inte...

2 Terminada

Figura 2.8 Tarea Diseño de Interfaz Juego [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Diseño Interfaz de Juego	Reunión de asociación	Terminada	Se realiza la reunión para definir la forma que tendrá la interfaz	Marcelo Agila	06/07/2012	06/07/2012
	Esbozar Diseño	Terminada	Se debe colocar los aspectos importantes, como ubicación del personaje, ubicación de barras de vida, una barra de acción, etc. Definiendo las posiciones que esta información ocupara dentro de la pantalla	Equipo Juego	07/07/2012	07/07/2012

Tabla 2.7 Subtareas Diseño de Interfaz de Juego [8] [48]

Nombre : Creación del Guión
Tipo : Historia de usuario
Clasificación : 6
Descripción : El guión se define en concordancia con la historia, en este se desarrolla la interacción de los personajes, las característica de cada uno, el ambiente en el que se desenvuelven etc.
Notas :
Fecha de envío : 04/07/2012 20:45:45
Fecha de aceptación : 05/07/2012 00:02:29
Fecha de estimación : 05/07/2012 00:06:58
Fecha de planificación : 05/07/2012 00:14:31
En progreso desde : 05/07/2012 21:43:18
Fecha de terminación : 14/07/2012 23:03:10
Creador : Marcelo Agila
Característica : Arte Conceptual

2
 Creación del Guió...
 2 Terminada

Figura 2.9 Tarea Creación del guión [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Creación del guión	Revisar la Historia	Terminada	Para corregir posibles errores sintácticos y semánticos	Marcelo Agila	05/07/2012	10/07/2012
	Redactar Guión	Terminada	Se redacta de acuerdo a la historia y los personajes	Equipo Juego	10/07/2012	15/07/2012
	Crear personajes (Descripción)	Terminada	Se define los rasgos y características físicas de los personajes así como su personalidad	Andrés Cevallos	05/07/2012	15/07/2012

Tabla 2.8 Subtareas Creación del guión [8] [48]

Nombre : Creación de los Personajes
Tipo : Historia de usuario
Clasificación : 8

Descripción : Los personajes son creados a partir del guión, aquí definimos sus características físicas creando los primeros bocetos.

Notas :
Fecha de envío : 04/07/2012 20:48:03
Fecha de aceptación : 05/07/2012 00:02:29
Fecha de estimación : 05/07/2012 00:13:13
Fecha de planificación : 05/07/2012 00:16:48
En progreso desde : 05/07/2012 21:43:18
Fecha de terminación : 24/07/2012 01:15:56
Creador : Marcelo Agila
Característica : Construcción de personajes

3
 Creación de los P...
 3 Terminada

Figura 2.10 Tarea Creación de Personajes [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Creación de los personajes	Asociar personajes con el guión	Terminada	Revisar el guión	Andrés Cevallos	20/07/2012	20/07/2012
	Diseñar Bocetos	Terminada	Diseñar los bocetos iniciales de cada personaje	Andrés Cevallos	20/07/2012	20/07/2012
	Descripción de Personajes	Terminada	Establecer una breve descripción del personaje de acuerdo al guión y la historia	Marcelo Agila	24/07/2012	24/07/2012

Tabla 2.9 Subtareas Creación de Personajes [8] [48]

Nombre : Modelado de Ambientes	11 Modelado de Ambie...
Tipo : Historia de usuario	8 Terminada
Clasificación : 9	
Descripción : El ambiente es el panorama que se presenta al jugador en el que se desenvuelve la historia del videojuego, por esta razón es importante que se adapte al guion.	
Notas :	
Fecha de envío : 04/07/2012 21:36:50	
Fecha de aceptación : 05/07/2012 00:02:29	
Fecha de estimación : 05/07/2012 00:07:48	
Fecha de planificación : 05/07/2012 00:17:12	
En progreso desde : 05/07/2012 21:43:18	
Fecha de terminación : 24/07/2012 01:58:54	
Creador : Marcelo Agila	
Característica : Arte Conceptual	

Figura 2.11 Tarea Modelado de Ambientes [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Modelado de Ambientes	Esbozar Ambiente	Terminada	Crear un boceto del mapa en general, que formará los distintos niveles. Esto sirve de guía para el modelado	Andrés Cevallos	24/07/2012	24/07/2012
	Modelado del Terreno	Terminada	Se debe crear un terreno para ubicar cada uno de los assets. Se debe tomar en cuenta los diseños para que este quede de manera adecuada.	Equipo Juego	24/07/2012	24/07/2012
	Construcción de Activos(Asset)	Terminada	Se deben construir distintos modelos para darle la ambientación adecuada a un escenario. Objetos, arboles, césped, casas.	Equipo Juego	24/07/2012	24/07/2012

Tabla 2.10 Subtareas Modelado de Ambientes [8] [48]

Nombre : Seleccionar tipo Videojuego
Tipo : Historia de usuario
Clasificación : 7
Descripción : Esta tarea la realizamos describiendo los distintos tipos de videojuegos que existen en la industria, y seleccionando uno de acuerdo a nuestro criterio
Notas :
Fecha de envío : 17/07/2012 13:39:04
Fecha de aceptación : 17/07/2012 13:39:12
Fecha de estimación : 17/07/2012 13:51:36
Fecha de planificación : 17/07/2012 13:51:51
En progreso desde : 17/07/2012 13:51:51
Fecha de terminación : 17/07/2012 13:55:45
Creador : Marcelo Agila
Característica : Arte Conceptual

14 Seleccionar tipo ...
 1 Terminada

Figura 2.12 Tarea Seleccionar tipo de Videojuego [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Selección del tipo de videojuego	Investigar tipos de videojuegos	Terminada	Investigación de la clasificación que se les da a los videojuegos	Equipo Juego	17/07/2012	17/07/2012
	Describir tipos de videojuegos	Terminada	Hacer una breve descripción de lo que se trata cada tipo	Marcelo Agila	17/07/2012	17/07/2012
	Escoger tipo Videojuego	Terminada	De acuerdo a la descripción definida se debe escoger un tipo de videojuego	Equipo Juego	17/07/2012	17/07/2012

Tabla 2.11 . Subtareas Seleccionar tipo de Videojuego [8] [48]

Una vez que se ha definido y detallado las tareas del primer sprint del entregable R1, se procede con la documentación del trabajo realizado en cada una de las tareas en el presente documento.

2.1.3 SELECCIÓN DEL TIPO DE VIDEOJUEGO [49] [50]

Los videojuegos al igual que la tecnología fueron evolucionando, es por esta razón que surgió la necesidad de agruparlos bajo ciertas características comunes que comparten entre sí, surgiendo los géneros de los videojuegos emulando a la música o al cine; entre los distintos se resumen en la Tabla 2.12.

Genero	Características	Subgénero	Características	Algunos ejemplos
Lucha	Se basa en los combates cuerpo a cuerpo, generalmente se ejecutan acciones repetitivas usando combinaciones de botones	Beat'em up	Son videojuegos de lucha en progresión, es decir el jugador o jugadores deben combatir con un gran número de enemigos mientras avanzan niveles	God Of War, Devil MayCry
		Lucha Uno contra Uno	Combates entre enemigos generalmente vistos desde una perspectiva lateral, los personajes son controlados por los jugadores o por IA. Generalmente son de artes marciales, armas cortas, armas largas, combates mágicos, etc.	Street Fighter, Tekken
		Free for All	Estos videojuegos tienen un escenario cerrado llamado arena con dos o más jugadores, y consiste en eliminarse entre si.	Super Smash Bros, Jump Ultimate Stars
Aventura	Juegos con argumento extenso, el personaje principal debe atravesar niveles interactuando con varios NPCs y objetos, generalmente son de larga duración.	Aventura Conversacional	Muestran una historia semejante a la de un libro, acompañada de imágenes estáticas con la posibilidad de intervenir en la historia. Tuvieron gran popularidad en los 80.	Zork, Phoenix Wright: Ace Attorney
		Aventura Gráfica	Se puede interactuar con el escenario de una manera gráfica combinando acciones con los objetos y elementos del escenario.	Indiana Jones and the Fate of Atlantis, Maniac Mansion
		Aventura de Rol	El personaje evoluciona a medida que avanza la historia, el jugador conoce nuevos personajes, juntando armas, experiencia, aliados, etc. Existen dos tipos los RPG(Role-Playing Game), y en la actualidad los MMORPG(Masive Multiplayer Online RPG) en el que cada jugador tiene un personaje creado a su gusto, y se conecta con miles de jugadores en un mundo virtual.	Final Fantasy, World of Warcraft
Estrategia	Se manipula a un gran número de personajes, unidades, con el fin de lograr objetivos. Existen videojuegos de Gestión económica o social, y videojuegos bélicos	TBS (Estrategia por Turnos)	En este tipo de videojuegos los jugadores interactúan por turnos, es decir que se realiza una acción de cada jugador a la vez.	Civilization, Total War
		RTS (Estrategia en tiempo real)	Difieren de los TBS por ejecutar las acciones simultáneamente entre jugadores o con AIS	Warcraft, Age of Empires

Survival Horror	Videojuegos de aventuras de terror, el personaje debe atravesar por situaciones generalmente sobrenaturales, con zombies, fantasmas, etc., generalmente resolviendo puzzles y dañando a los enemigos con armas u objetos contundentes. El terror psicológico junto con la ambientación y el sonido son característicos de este tipo de videojuegos			Alone in the Dark, Resident Evil, Silent Hill
Plataformas	El personaje debe recorrer, saltar o escalar una serie de plataformas y acantilados con enemigos, generalmente usando desplazamientos a la izquierda o derecha, dependiendo del tipo de videojuego. De este tipo de videojuegos han surgido leyendas como Mario Bross o Sonic.			Super Mario Bros, Sonic the Hedgehog, Megaman, Donkey Kong
Musicales	Videojuegos desarrollados en torno a la música, la mayoría de estos videojuegos requieren de accesorios especiales de acuerdo a las consolas para los que sea desarrollado	Baile	Se debe seguir un conjunto de pasos de acuerdo al ritmo de la música, literalmente bailando, los controles son accionados por los pies del jugador que permanece de pie frente a una pantalla que le marca que posición o control debe presionar.	Dance Dance Revolution, Pop 'n' Music
		Creación Musical	Difiere del anterior ya que estos permiten componer música emulando distintos instrumentos musicales	MTV Music Generator, Techno Motor
		Karaoke	Se debe interpretar la canción con la música de fondo, con la letra de la canción en la pantalla.	Singstar, Dreamkara
		Precisión	Los jugadores pueden tomar el papel de integrante de la banda tocando un instrumento, siendo DJ, o formando una banda en conjunto con otros jugadores.	Guitar Hero, Rock Band, DJ Hero
Simulación	Son videojuegos caracterizados por tener aspectos de la vida real, las situaciones cotidianas son enmarcadas para que el usuario escogiendo distintas opciones llegue a un final distinto.	Combate	Caracterizado por el realismo de todos los aspectos, por ejemplo el comportamiento de los personajes en el que un disparo significa la muerte del personaje.	Armed Assault
		Construcción	Se proporcionan herramientas para la elaboración de un proyecto apegado a la realidad, como ser alcaldes de ciudades y construirlas o evolucionar civilizaciones y comandarlas en una guerra	SimCity, Age of Empires
		Vida	Se controla personajes con emociones humanas, controlando además los aspectos de su vida como la alimentación, vivienda, trabajo. Generalmente en línea de tiempo pueden comer, dormir, evolucionar e incluso morir.	The Sims, Spore
		Deporte	Simulan juegos de deporte, como fútbol, tenis básquet, hockey, etc. con el mismo propósito que el juego real.	FIFA, NBA, Wii Sports
		Carreras	Simulan a las competiciones de carreras legales que basan su juego en ganar la partida o ilegales basadas en juegos de persecución.	Need For Speed, Mario Kart

Tabla 2.12 Clasificación de los Videojuegos [8] [49] [50]

El género que se ha seleccionado es RPG, cuenta con una historia, un guión sobre un personaje específico dentro de un mundo virtual, interacción entre personajes, entre otros. Estos elementos hacen que se pueda contar la historia a medida que avanza el videojuego.

2.1.4 HISTORIA DEL VIDEOJUEGO [8]

Para tener una idea general de lo que se quiere en la historia del videojuego, se recomienda escribir pequeñas frases, con el contenido de las ideas que se vayan presentando; esto con el fin de enlazarlas posteriormente para formar un argumento más extenso que contenga la historia del videojuego. Además de las ideas sobre la historia, se debe definir el ambiente en el que se va a desarrollar, es decir las características del lugar donde se realizarán las acciones; posteriormente se hace una breve descripción de los personajes principales al mismo tiempo que se va construyendo la historia.

2.1.4.1 Ideas Principales:

- Seres malignos comandados por un orco esclavizan a las personas de un pequeño pueblo, haciéndolas trabajar en ruinas antiguas que guardan un antiguo poder oculto.
- Una valiente guerrera decide deshacerse de la maldad que asola al pueblo y marcha en contra de estos seres malignos.
- Un poder aún mayor que el de orco se encuentra escondido tras las ruinas, es maligno.
- La guerrera derrota al orco, al hacerlo libera un poder aún mayor; pero ella se revela al mismo tiempo como el arcángel de la muerte Azrael.

2.1.4.2 Ambiente

- La tierra está poblada por criaturas mágicas y míticas que conviven junto a los humanos.
- La historia se desarrolla en un escenario pequeño, una aldea es el centro de atención de la historia. No está ambientado en un escenario

normal, por los personajes fantásticos que toman parte dentro de la historia.

2.1.4.3 Personajes Principales

- **Azrael:** es una chica fuerte, luchadora, su fuerza es sobrehumana y misteriosa; gracias a esta fuerza se puede enfrentar a criaturas demoníacas. Es distante de la mayoría de los habitantes de la villa, oculta un gran secreto. Su personalidad es firme y fuerte de carácter; le encanta disfrutar de los placeres simples de la vida como caminar entre la naturaleza, admirando a los seres y su perfecta armonía con el entorno. Lleva sobre su espalda una espada legendaria forjada con orichalcum⁷⁰, un metal sumamente resistente; su espada junto a su fuerza son capaces de invocar hechizos efectivos contra demonios. Siempre está acompañada de su fiel mascota Wylla, un lobo descendiente del gran Fenrir⁷¹. Ayuda a los débiles combatiendo las injusticias que se comenten contra ellos. Ha recorrido el mundo desde el principio de los tiempos y sabe que la maldad es un mal difícil de apaciguar. Se revela como el Arcángel de la Muerte, para liberar su verdadero poder y luchar contra el señor del mal BoomDraen que invadió Villanaciente.
- **Wylla:** es un lobo de gran tamaño, fiel como toda mascota. Descendiente del gran Fenrir, el lobo que desoló al mundo hace mucho tiempo. Quedó huérfano a temprana edad, gracias a eso su carácter es apacible, siendo criado por una manada de lobos hasta que Azrael lo encuentra y Wylla se encariñó con ella. Es compañero leal y sirve de montura a su dueña. Posee ciertas habilidades heredadas de sus antepasados, que han ayudado a vencer a seres poderosos y malignos en tiempos antiguos. A pesar que tiene varios años es joven para su

⁷⁰ Metal legendario mencionado en escritos antiguos siendo los más significativos los escritos de Platón sobre la Atlántida

⁷¹ En la mitología nórdica es un lobo monstruoso, padre de los lobos Sköll y Hati. Hijo de Loki que debe matar al dios Odin durante el Ragnarök pero es asesinado por el hijo de Odin Viðarr

especie y puede llegar a tener tamaños descomunales. Le encanta perseguir insectos y disfrutar del agua de ríos y riachuelos.

- **Samuro:** un orco con poderes mágicos, invocado hace milenios por un hechicero que poseído por la locura abrió una puerta hacia un mundo desconocido; al volverse loco en un mundo que no era el suyo, Samuro destruyó el portal y mato al hechicero. Usualmente viajaba sólo, atacando pequeñas aldeas y saqueando lo que podía para sobrevivir. En una época fue reconocido por todas las razas del mundo por ser un gran guerrero. Pero una fuerza desconocida que empezaba a crecer en el mundo invadió la mente de Samuro y lo transformó en un siervo fiel; era el poder de BoomDraen. Samuro es grande y gordo, está provisto tan sólo de una trusa, usa dos espadas para magnificar su fuerza en contra de sus adversarios. Se dice que murió en la batalla contra el Arcángel de la Muerte, pero nunca encontraron su cuerpo.
- **BoomDraen:** es un antiguo humanoide dragón. Fue condenado en la primera guerra de los arcángeles a pasar la eternidad con su forma de cascarón, a causa de su interés en las artes oscuras, para sumir al mundo en un completo caos que sólo él fuera capaz de gobernar. Apenas se notan sus brazos y se puede observar parte de su rostro. Fue encerrado en un gran lago capaz de contener la fuerza mágica, según escrituras antiguas, pero debido a alteraciones producidas por los habitantes de Villanaciente, empezó a liberarse tomando poco a poco más poder y más poder. Con su mente controla seres malignos, encargados de su completa liberación; sus esbirros obligan a los habitantes de la villa a cavar rompiendo los sellos que lo mantienen cautivo del mundo.
- **Shanko:** es un pequeño niño audaz, vive en Villanaciente. Logró escapar del pueblo cuando los esbirros de BoomDraen comandados por Samuro, atacaban a la pequeña aldea. Su curiosidad lo llevó a descubrir que la mujer misteriosa que habitaba a las afueras de la villa era una valiente guerrera; tras escapar de la Villa, es asediado por

criaturas maléficas, finalmente Azrael acude a su ayuda. Shanko solicita a Azrael que ayude a su pueblo a ser liberado del ataque de los demonios.

- **Demonios o Moustros:** comandados por Samuro, son los esbirros de BoomDraen.
- **Soldados:** guerreros del Rey, que batallan para defender el pueblo.

2.1.4.4 Detalle de la Historia

Historias míticas, han formado parte de las leyendas de los pueblos del mundo por siglos. Esta es una de esas leyendas. Cuenta la historia que existió una noble guerrera, solitaria, nacida desde el inicio de los tiempos. Participó en una antigua guerra entre ángeles y demonios. Después de la guerra se dedicó a viajar por el mundo, tratando de apaciguar la maldad que acompaña a los seres humanos. Viajó durante siglos hasta que sintiendo la maldad liberada se establece en un pequeño pueblo llamado Villanaciente. Un acontecimiento que marcaría a los habitantes del pequeño pueblo, estaba a punto de acontecer; los habitantes sin saber trabajaban extrayendo minerales en lo que antes fue un gran lago, parte de la prisión del Draco Demonio BoomDraen; por esta razón la maldad se pudo desatar y el demonio débil consigue reunir un ejército que se encargaría de liberarlo por completo. Un día un grupo de malvadas criaturas guiadas por Samuro, un orco poseído por el poder de BoomDraen, marchan sobre el pueblo y esclavizan a los habitantes, obligándolos a trabajar en las ruinas en las que se encontraba sellado el antiguo demonio. BoomDraen con su poder aún débil pero en aumento había logrado manipular a diversas criaturas y esbirros que ahora trataban de liberarlo completamente. El pueblo es atacado, pero sus defensas son inútiles contra esas criaturas; un joven valiente y astuto fue capaz de escapar del yugo que estaban imponiendo los demonios sobre los habitantes del pueblo; se trataba de Shanko, que anteriormente había

escuchado relatos sobre la valiente guerrera en pueblos por los que viajaba con su padre un mercader orfebre. Shanko escapa en busca de la guerrera, que habitaba en el bosque, pero es atacado lejos de la villa. La guerrera, se da cuenta de que alguien está en problemas, acude a su rescate. Sorprendida por la rapidez con la que actúa la legión acude a detener a los monstruos y consuela al chico. Su viaje estará lleno de peligros, al llegar a la aldea presencia una batalla entre las fuerzas del Rey y la Legión; pero el panorama es poco alentador y Azrael decide intervenir. Derrotan inicialmente a los demonios en la villa y esta queda en protección de los soldados del Rey. Azrael viaja a las ruinas acompañada de Shanko y Wylla. Se enfrenta a varios enemigos hasta llegar a la cima de la villa para enfrentarse al orco Samuro; luchan intensamente y finalmente el orco es vencido, pero este hace un último sacrificio para liberar su mente del dominio de BoomDraen incendiando su cuerpo, al hacerlo inconscientemente permite que BoomDraen se libere. Finalmente la guerrera hace frente a BoomDraen liberado pero sin sus poderes al 100% y en el pueblo se libra la batalla final. El demonio es muy poderoso. La guerrera viendo que sus fuerzas decaen, libera todo su poder revelándose como el Arcángel de la Muerte Azrael; cuya misión es recibir las almas de los muertos y conducirlos para ser juzgados. El demonio es vencido y juzgado, finalmente la paz se restaura en Villanaciente aunque unas sombras se apoderan del pueblo presagiando un futuro apocalipsis. El caos se desato sin importar que BoomDraen haya sido vencido, pero esto es parte de otra historia.

2.1.4.5 Nombre del Videojuego

- Con estos elementos se puede escoger el Título del Videojuego, un nombre relevante a la historia escrita, *Azrael Awakening*. Define la idea principal de la historia - la liberación del poder oculto en una guerrera- que resulta ser uno de los 7 arcángeles del apocalipsis, según la

tradición musulmana; el Arcángel de la Muerte encargado de juzgar a las almas.

2.1.5 CREACIÓN DEL GUIÓN DEL VIDEOJUEGO [61]

El guión define las acciones que el usuario observará mientras el videojuego se esté ejecutando, por eso es importante que los diálogos vayan de acuerdo a los personajes y siguiendo la línea de la historia ya definida.

Primer Acto:

Es un día caluroso de verano, Azrael está descansando cerca de un lago con su mascota Wylla. De repente un niño clama por ayuda espantado del miedo, se encuentra huyendo de unos demonios que le persiguen; la protagonista al verlo ayuda al niño enfrentándose rodeado de demonios. (En esta escena se especifican los controles del personaje).

Diálogos:

Kid: Heeeeelpppp, pleaseeee heeeeelp.

Azrael: oh, Wylla look at that!!

That poor kid is being attacked by those freaking monsters. Let's go on and help!!

Wylla: Woooooof

Después de la pelea.

Azrael:(Sujetando al chico). Kiddo are you all right???

Kid: Please help (se desmaya)

Azrael: Better give him some first aid.

Wylla: Woof.

El chico explica lo sucedido.

Azrael: Hi kid so finally you wake up, you were pretty bad.

Kid: Whooooaaa, where am I?, where those monsters go? (gritando).

Azrael: Easy boy I beat those monsters that were attacking you.

Kid: Oh really, thank you, no one in my village can fight to those creatures.

Azrael: Please tell me what happen.

Kid: Well, My name it's Shanko, some days ago a powerful ogre attacked our village summoning those weird creatures. They are forcing us to work on some antique ruins to get something like crystals. I manage to escape days ago to get some help, the king told me that he will sent his army, I returned to the village to tell the good news, but when I returned the things were worst. I try to fight to protect my family them but the only thing I can do is run again (Llorando).

Azrael: Don't cry kid we'll help you.

Shanko: Thanks, but tell me who are you? Are you the Legend Warrior?

Azrael: Well, Lets say that I'm a justice lover.

Inmediatamente después de esta escena, empieza el viaje hacia la aldea.

Segundo Acto:

La guerrera llega a Villanaciente, en la aldea se encuentra desarrollando una batalla. Se trata de las fuerzas del rey tratando de contener el poder de los demonios.

Shanko: Wow the army of the king is fighting

Azrael: Yes kid but the things aren't going very well

Azrael: Don't worry kid i will help. Willa Lets Go.

(empieza la batalla en la aldea)

???: So you pathetic beings, decide to reveal against me , the mighty ogre Samuro.

Azrael: Samuro..... ??? You are the one that is forcing to work to these people.

Samuro: What if I am.

Azrael: I will beat you

(empieza la pelea).

Samuro: You are thoughter than i thought. (Cansado)

I let you win this time. But i will you a little surprise (escapa)

Azrael: Ohh geez. What tha heck it's this.

Monster: Rooooaaaar

Después de Derrotar al monstruo, reunidos en la villa los soldados del rey:

Soldier: We win, the enemy is retreating.

Captain: Well, now we are going to pursuit them into their base.

Azrael: You mustn't do it.

Captain: What, who's telling these things.

Azrael: Im telling this

Captain: Why you tell these things??

Azrael: Your army can barely stand to those creatures, and your men are really injured.

Captain. Well..

Soldier: But if we don't do nothing the kingdom...

Captain: Silence, we can't let civilians know about it

Azrael: Je je je, don't worry, I almost know it. The army of king defending a little village its very suspicious.

Captain: Ha, what are you saying is our duty to protect villagers

Azrael: You can stop that, the ogre that attacked the village was the legendary Samuro master of the "Sky Sword". He supposed passed away many years ago.

Captain: How you know that.

Azrael: Well Tell me the truth.

Captain: We didn't know too much, but a powerful demon was sealed in the ruins north of town. Many years ago Samuro warn us of his resurrection, but the king doesn't listen him. He tied to stop alone the ritual but it loops that the demon beat him and now its using him as his general for the plans of resurrection.

Azrael: So that's it. If we don't act quickly the demon will resurrect and cause many trouble.

Captain: That's we must attack

Azrael: That won't work, your army will be defeated. But don't worry I will stop the resurrection of that being.

Captain: I look you fight and I know that you are strong, but..

Azrael: Don't worry, if I don't return until night you must attack, at least I will weak the enemy defenses and you can rest for the battle.

Azrael se prepara con Wylla.

Captain: Looks like we don't have another choices.

Tercer Acto:

La guerrera parte hacia las legendarias ruinas para evitar que se desarrolle el ritual de invocación del demonio BoomDraen.

Shanko: I hear all about your plan.

Azrael: Don't worry I'll be fine

Shanko: I will go with you

Azrael: No way, you will be hurt

Shanko: If you let me go I can show you the way to the ruins.

Azrael: Ok come on Kid but you should be careful, let's go!!

(Llegan a las ruinas)

Samuro: Stop!! I can't let you pass. These ruins will be your finish.

Azrael: I'll fight with you Samuro and I'll win.

(La batalla con el orco empieza)

(Samuro viendo su final cerca finalmente se libera de la posesión demoniaca de BoomDraen, incendia su cuerpo para evitar que el hechizo de invocación culmine y Boom se pueda liberar)

Samuro: I'll die. Sorry I can't anything by myself. I hope Boom never can resurrect. Ohh I can see my family. Ahrrrr!! I'll stay with you (Mirando a su familia en su mente)

(BoomDraen es liberado rompiendo el sello en piedra, mientras Azrael y Shanko impresionados permanecen inmóviles)

BoomDraen: Finally I'm here to lead the world. All of you will be under my control. The chaos is coming!! And I'll fill the world with darkness.

Azrael: We're here!!

2.2 DISEÑO DE LAS ARTES GRÁFICAS

Las artes gráficas representan el diseño artístico del videojuego, parte de este diseño está representado en las interfaces, a continuación se detalla el diseño de las interfaces para Azrael Awakening.

2.2.1 DISEÑO DE INTERFACES [8]

Las interfaces presentadas a continuación son simples bocetos de los diseños reales que se deben realizar más adelante, pueden ser modificados en la fase de construcción.

2.2.1.1 Interfaz de Presentación:

La interfaz de presentación está compuesta por una secuencia de imágenes que se activan cuando se inicia el videojuego. Las imágenes contienen información sobre el videojuego como las herramientas que se usaron, nombres de los creadores y obviamente el nombre del videojuego, la Figura 2.13 muestra el posible diseño.



Figura 2.13 Bocetos Interfaz Presentación [8] [44]

2.2.1.2 Interfaz de Inicio:

La interfaz de Inicio aparece inmediatamente después de la secuencia de imágenes de la interfaz de presentación, esta interfaz contiene varias opciones, se detalla la posible navegabilidad en la Figura 2.14.

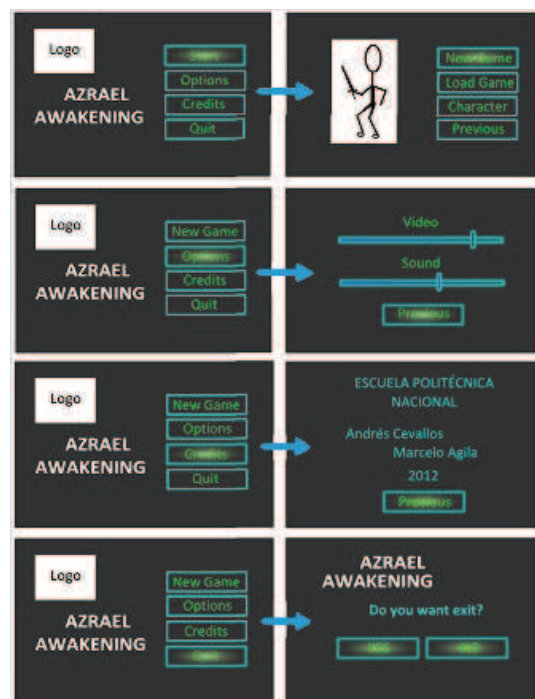


Figura 2.14 Boceto Navegación Menú Principal [8] [44]

2.2.1.3 Interfaz de Pausa:

La interfaz de Pausa se activa con una tecla, el juego que está en acción es detenido y se presenta una pantalla con algunas opciones para el usuario, estas opciones están detalladas en la Figura 2.15.

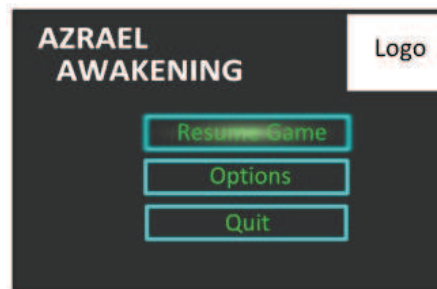


Figura 2.15 Boceto Interfaz de Pausa [8] [44]

La interfaz permite detener el juego y escoger opciones como cambiar la resolución de sonido, video, retornar al juego, además de una opción para salir.

2.2.1.4 Interfaz de Juego:

La interfaz de juego contará con varios elementos; una flecha para direccionar el objetivo a seguir, un mini-mapa en el que se visualizan los objetos, una barra de estado que representa las vidas del personaje.

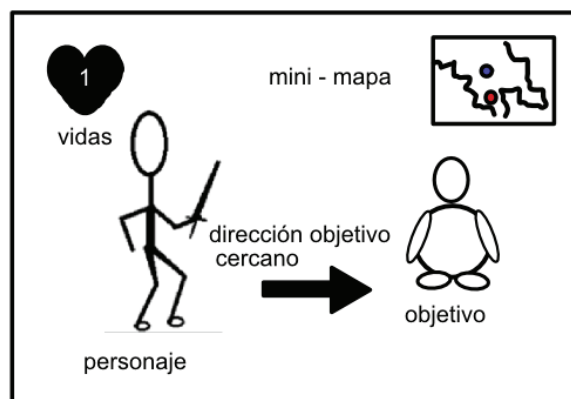


Figura 2.16 Boceto de la Interfaz de Juego [2] [61]

2.2.2 CREACIÓN DE LOS PERSONAJES [8]

Como parte del Primer Sprint del entregable R1, se debe crear los personajes a partir de la construcción de ciertos bocetos; los bocetos son creados en dos vistas, una frontal y una lateral. De esta manera se asegura que la construcción del personaje sea más versátil, la construcción se verá en capítulos posteriores. A continuación se presenta el diseño de los personajes en la Figura 2.17.



Figura 2.17 Bocetos Personajes Videojuego [8] [44]

La altura de las perspectivas deben coincidir en pies, cabeza, brazos, es muy importante para el proceso de construcción del modelo 3D de cada personaje.

2.2.3 CREACIÓN DEL AMBIENTE DEL VIDEOJUEGO [8]

La creación del ambiente al igual que la de los personajes se basa en un boceto principal, se establece primeramente el mapa en miniatura del terreno del videojuego, luego se le pone algunos detalles que son los que permiten modelarlo en 3D.

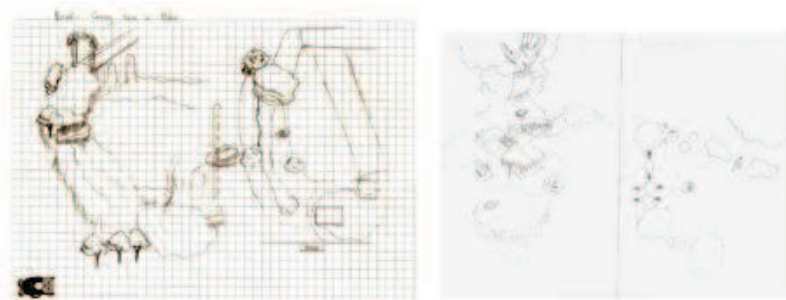


Figura 2.18 Boceto del terreno [8]

2.3 HISTORIAS DE USUARIO

Se debe notar que el Primer Sprint R1 ha finalizado, las historias de usuario

del videojuego estrictamente de la parte funcional se detallan en el R1-Segundo Sprint; de esta manera se inicia con la definición de cada una de las tareas que conforman este Sprint.

2.3.1 PRIMER ENTREGABLE R1-SEGUNDO SPRINT

El segundo Sprint del entregable R1 consta de tareas que ayudan a iniciar con la funcionalidad del videojuego, se define totalmente las características de los personajes, además de establecer la lógica del videojuego. Hay que mencionar que este capítulo se termina con algunas tareas que forman parte de este sprint, no con su totalidad. Se presentan las tareas que forman parte del segundo Sprint en la Figura 2.19.

Tareas urgentes	
Diseño de clases - Terminada	100%
Diseño de relaciones entre clases - Terminada	100%
Definir atributos y operaciones - Terminada	100%
Creación de Historias de Usuario - Terminada	100%
Modelado de Personajes - Terminada	100%
Modelado de Personajes Lowpoly - Terminada	100%
Texturizado - Terminada	100%
Modelamiento de Assets para escenarios - 3 - Terminada	100%
Texturizado de Assets - 2 - Terminada	100%

Figura 2.19 R1-Segundo Sprint Tareas [48]

Se ha separado las tareas que forman parte de este capítulo, las restantes se encuentran detalladas en el capítulo 3.

Tareas urgentes	
Diseño de clases - Terminada	100%
Diseño de relaciones entre clases - Terminada	100%
Definir atributos y operaciones - Terminada	100%
Creación de Historias de Usuario - Terminada	100%

Figura 2.20 Tareas R1-Segundo Sprint Capítulo 2 [48]

2.3.1.1 Principales características del Entregable R1-Segundo Sprint

Para la realización del segundo sprint se ha creado una característica llamada lógica del videojuego que se añade al resto de características.

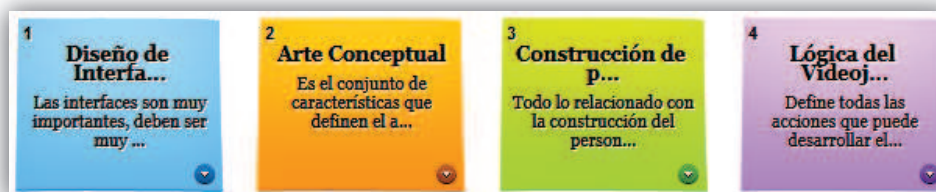


Figura 2.21 Características R1-Segundo Sprint [48]

Nombre	Descripción	Requerimientos
<i>Lógica del Videojuego</i>	Define todas las acciones que puede desarrollar el personaje y los NPCs en el videojuego, es decir todas las acciones dependan o no del jugador.	Re1, Re2, Re3, Re7 (Tabla 23)

Tabla 2.13 Característica añadida R1-Segundo Sprint [8]

2.3.1.2 Detalle de las tareas en el segundo Sprint

A continuación se detalla las tareas de este capítulo que son parte del segundo Sprint.



Figura 2.22 Tarea Diseño de Clases [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Diseño de Clases	Ubicar los objetos	Terminada	Se debe abstraer todos los objetos posibles que serán parte del videojuego, las clases representarán a cada uno de estos objetos interactuando entre sí.	Marcelo Agila	08/09/2012	14/09/2012
	Agrupar las clases	Terminada	Varias clases formarán de paquetes, estas clases se las representa en un diagrama general estableciendo relaciones entre paquetes	Marcelo Agila	14/09/2012	14/09/2012

Tabla 2.14 Subtareas Diseño de Clases [8] [48]



Figura 2.23 Tarea Definir Atributos [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Definir Atributos y Operaciones	Definir Atributos	Terminada	Cada clase posee atributos, es importante definirlos ya que sobre estos se establece las operaciones. Cada atributo representa las características que posee la clase.	Marcelo Agila	08/09/2012	16/09/2012
	Definir Operaciones	Terminada	Establecer las operaciones entre cada unas de las clases es fundamental para estructurar la lógica del videojuego.	Marcelo Agila	08/09/2012	16/09/2012

Tabla 2.15 Subtareas Definir Atributos y Operaciones [8] [48]



Figura 2.24 Tarea Definir Relaciones Entre Clases [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Diseño de relaciones entre clases	Establecer relaciones	Terminada	Las relaciones se establecen entre las clases, se complementan con cada operación que se realice.	Marcelo Agila	16/09/2012	16/09/2012

Tabla 2.16 Subtareas Definir Relaciones Entre Clases [8] [48]



Figura 2.25 Tarea Creación de Historias de Usuario [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Creación de Historias de Usuario	Elaborar Plantilla de Historia	Terminada	La creación de la plantilla para las historias de usuario tendrá similitud a las generadas por ICESCRUM, pero serán agrupadas dentro de esta tarea general como parte del proceso de desarrollo en Scrum.	Marcelo Agila	05/10/2012	05/10/2012
	Elaborar Historias de Usuario	Terminada	Las historias de usuario deben estar basadas en las características funcionales del videojuego, por esta razón está dentro de la característica, Lógica del Videojuego.	Marcelo Agila	05/10/2012	05/10/2012

Tabla 2.17 Subtareas Creación de Historias de Usuario [8] [48]

Una vez que se ha definido y detallado las tareas del primer sprint del entregable R1 en este segundo capítulo, se procede con la documentación del trabajo realizado en cada una de las tareas en el presente documento.

2.3.2 DIAGRAMA DE CLASES

La creación del diagrama de clases, ha sido planificada siguiendo las características de los personajes y estructurando la arquitectura del videojuego en relación con el motor gráfico. La Figura 2.26, muestra el diagrama de paquetes general.

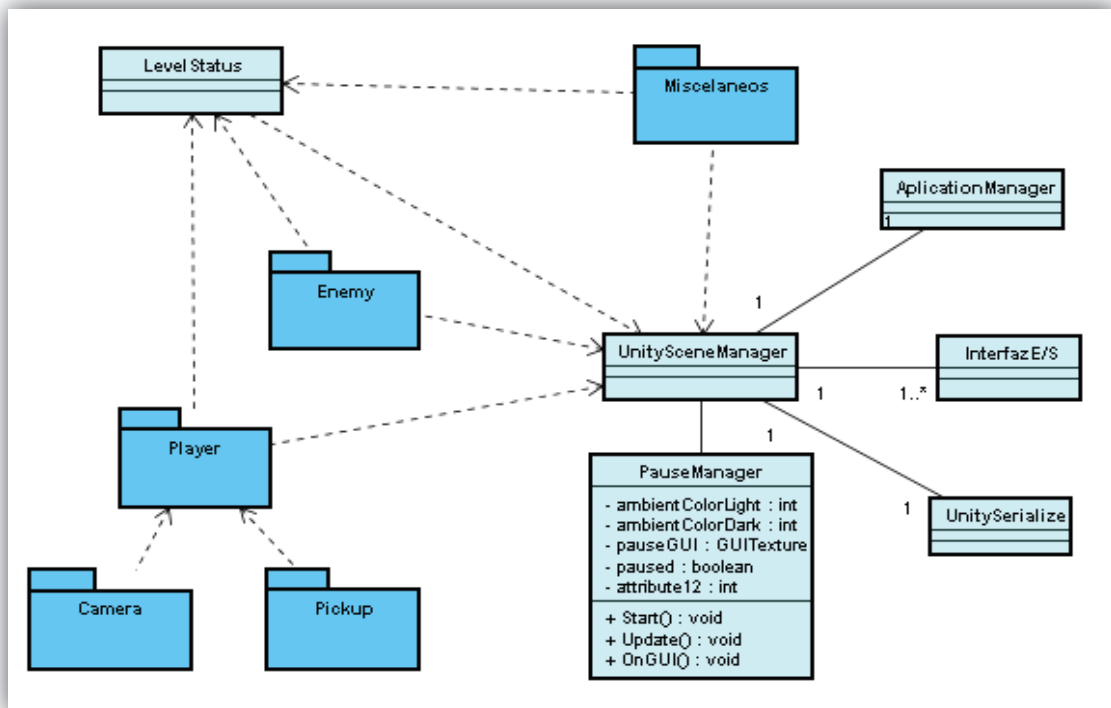


Figura 2.26 Diagrama de Paquetes Azrael Awakening. Fuente: Astah⁷² [8]

⁷² Programa para modelado UML

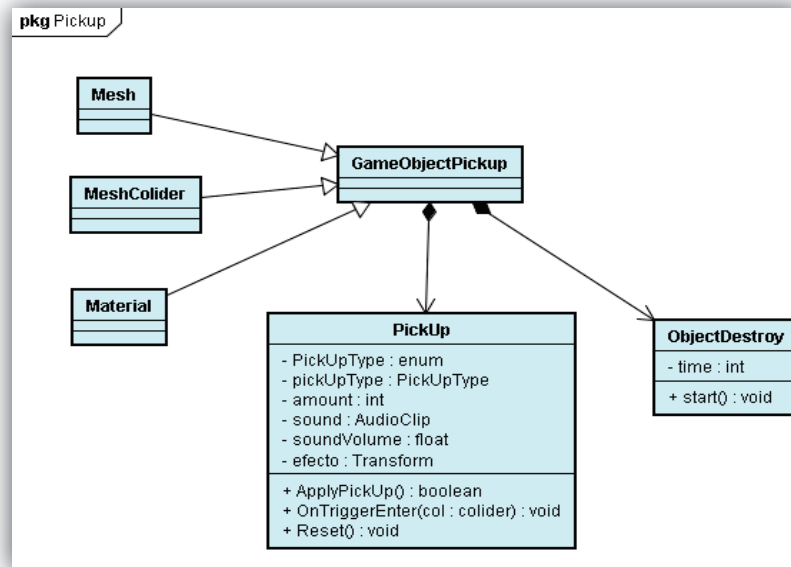


Figura 2.27 Diagrama de Clases Paquete Pickup. Fuente: Astah [8]

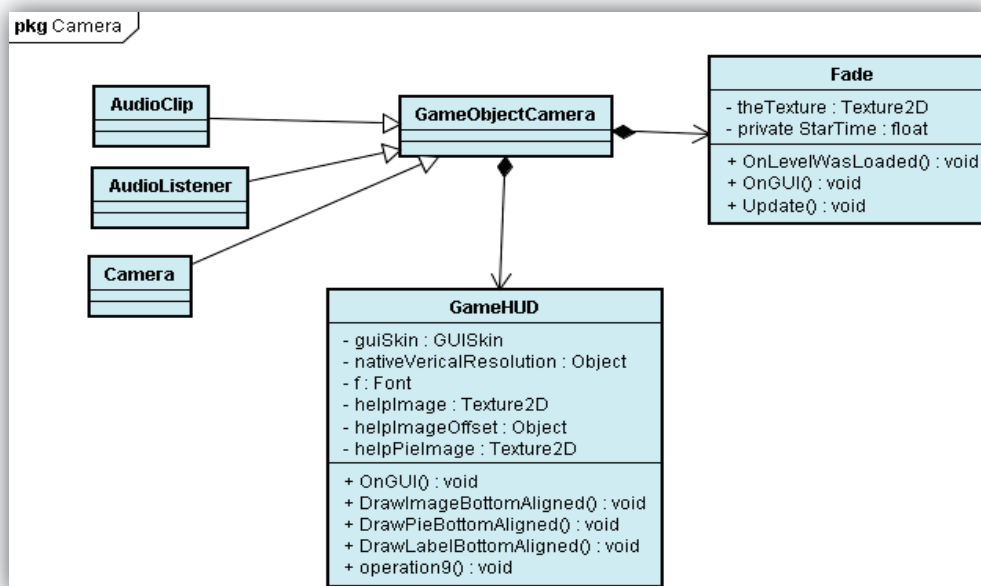


Figura 2.28 Diagrama de Clases Paquete Camera. Fuente: Astah [8]

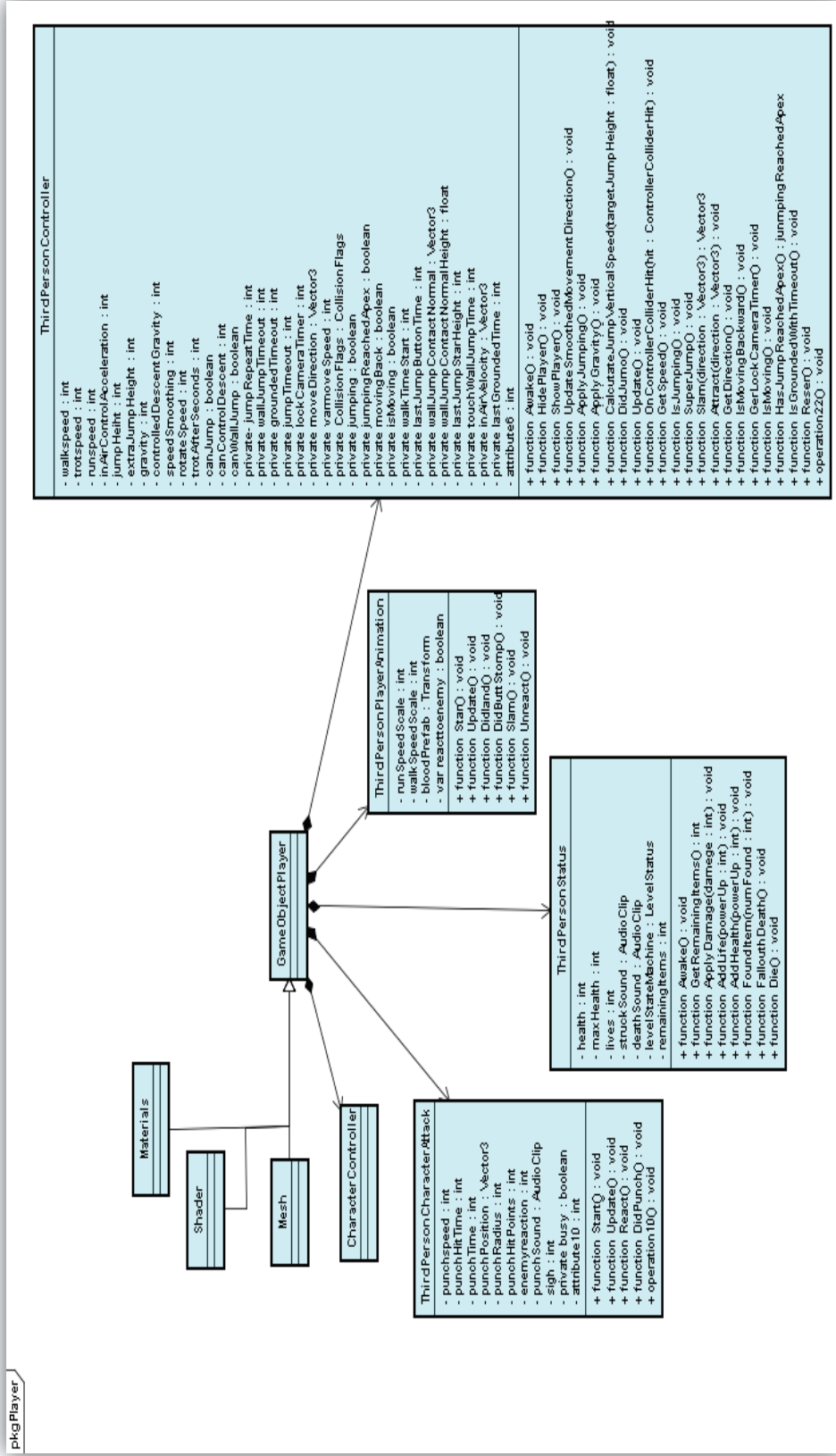


Figura 2.29 Diagrama de Clases Paquete Player Fuente: Astah [8]

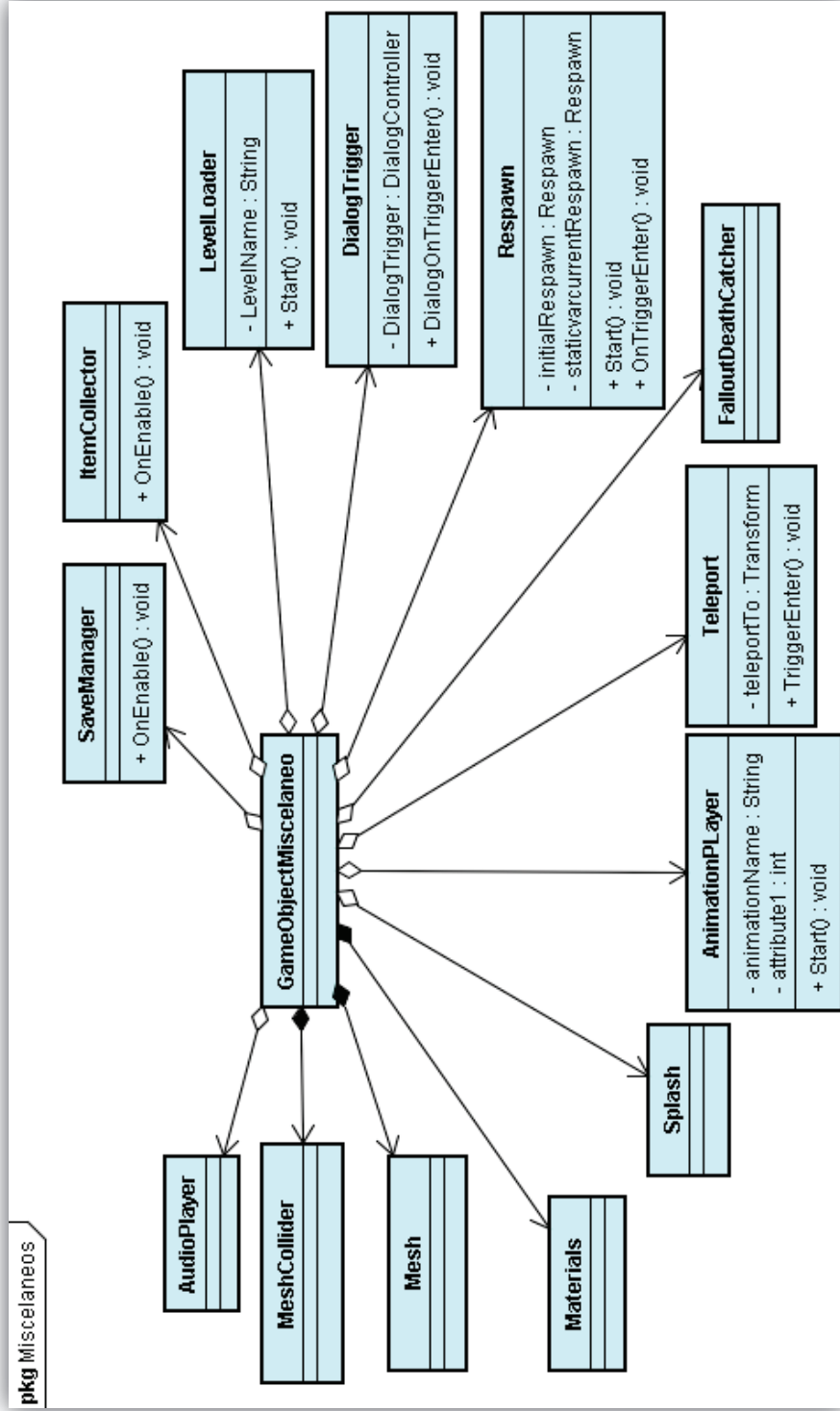


Figura 2.30 Diagrama de Clases Paquete Miscelaneos. Fuente: Astah [8]

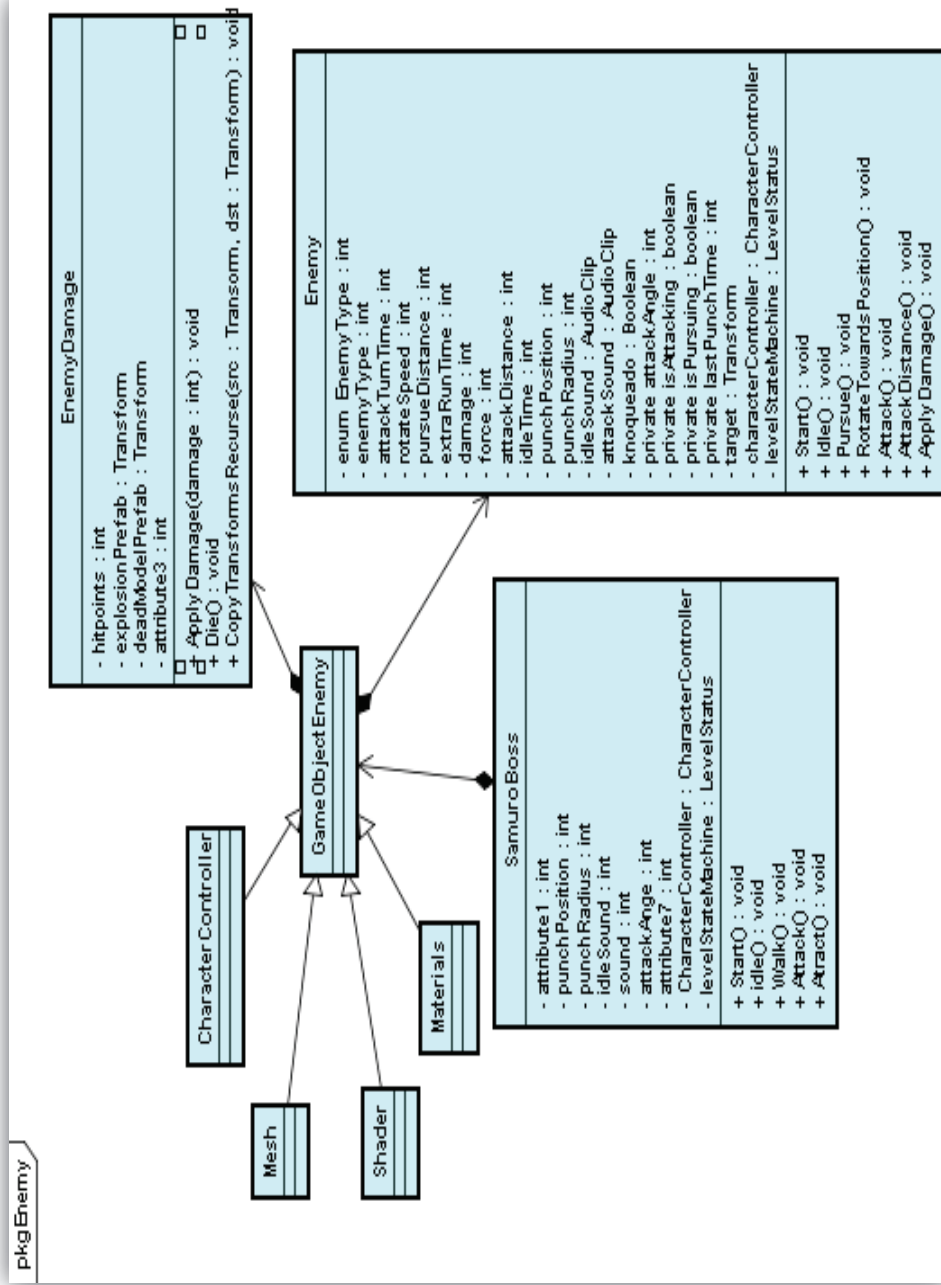


Figura 2.31 Diagrama de Clases Paquete Enemy. Fuente: Astah [8]

2.3.3 HISTORIAS DE USUARIO [8]

Las historias de usuario están definidas sólo para la funcionalidad dentro del videojuego; todo el proceso se ha manejado con historias de usuario a través de ICESCRUM, estas historias de usuario poseen la particularidad de ser customizadas para obtener de mejor manera información que servirá después en el proceso de desarrollo.

Notación:

- (1) **Usuario**, define el usuario que para todos los casos es player o jugador.
- (2) **Prioridad en el negocio**, es baja, media o alta dependiendo del nivel de celeridad necesario en el desarrollo del proyecto.
- (3) **Riesgo en desarrollo**, es bajo, medio o alto dependiendo de la importancia que se le da para la correcta ejecución del producto final
- (4) **Puntos estimados**, corresponden al esfuerzo, en este caso medido en días.

Historia de Usuario				1
Usuario:		Jugador		
Nombre historia:		Diseño Interfaz Presentación		
Prioridad en negocio:		Baja	Riesgo en desarrollo:	Medio
Puntos estimados:		1	Iteración asignada:	2
Programador responsable:		Marcelo Agila		
Descripción: Se debe crear una secuencia de animación, la cual represente el gameplay del videojuego.				
Observaciones: La presentación de inicio es importante en todo videojuego.				
Precondición: Poseer aplicación para desarrollo				

Tabla 2.18 Historia Usuario1 [8]

Historia de Usuario				2
Usuario:		Jugador		
Nombre historia:		Diseño Interfaz Inicio		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	Bajo
Puntos estimados:		4	Iteración asignada:	2
Programador responsable:		Andrés Cevallos		
Descripción: La interfaz de Inicio contiene el menú Inicial, con lo cual se puede iniciar el juego o continuarlo				
Observaciones: Las interfaces se crean dentro del juego mediante la función OnGUI.				
Precondición: Poseer aplicación para desarrollo				

Tabla 2.19 Historia Usuario2 [8]

Historia de Usuario				3
Usuario:		Jugador		
Nombre historia:		Diseño Interfaz de Juego		
Prioridad en negocio:	Alta	Riesgo en desarrollo:	Alto	
Puntos estimados:	1	Iteración asignada:	2	
Programador responsable:		Marcelo Agila		
Descripción: La interfaz de juego.				
Observaciones: La interfaz de juego es muy importante, debe ser versátil y amigable al usuario manteniendo el concepto de jugabilidad.				
Precondición: Poseer aplicación para desarrollo				

Tabla 2.20 Historia Usuario3 [8]

Historia de Usuario				4
Usuario:		Jugador		
Nombre historia:		Diseño Interfaz de Pausa		
Prioridad en negocio:	Media	Riesgo en desarrollo:	Bajo	
Puntos estimados:	1	Iteración asignada:	2	
Programador responsable:		Andrés Cevallos		
Descripción: La interfaz de pausa permite al jugador detener el juego y presenta algunas opciones de configuración.				
Observaciones: Ninguna				
Precondición: Poseer aplicación para desarrollo				

Tabla 2.21 Historia Usuario4 [8]

Historia de Usuario				5
Usuario:		Jugador		
Nombre historia:		Diseño Interfaz Presentación		
Prioridad en negocio:	Alta	Riesgo en desarrollo:	Bajo	
Puntos estimados:	2	Iteración asignada:	2	
Programador responsable:		Marcelo Agila		
Descripción: Presenta la secuencia multimedia de inicio del videojuego funcionando como presentación				
Observaciones: La presentación de inicio es importante en todo videojuego.				
Precondición: Poseer aplicación para desarrollo				

Tabla 2.22 Historia Usuario5 [8]

Historia de Usuario				6
Usuario:		Jugador		
Nombre historia:		Diseño de Personajes		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	Bajo
Puntos estimados:		6	Iteración asignada:	2
Programador responsable:		Marcelo Agila		
Descripción: Los personajes adaptados a la historia representan la esencia del videojuego en general.				
Observaciones: Ninguna				
Precondición: No Posee precondición				

Tabla 2.23 Historia Usuario6 [8]

Historia de Usuario				7
Usuario:		Jugador		
Nombre historia:		Construcción de Personajes 3D		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	Bajo
Puntos estimados:		12	Iteración asignada:	2
Programador responsable:		Andrés Cevallos		
Descripción: Los Objetos 3D son creados en un editor 3D. El proceso consiste en crear un objeto de malla				
Observaciones: Ninguna				
Precondición: Diseños de Personaje				

Tabla 2.24 Historia Usuario7 [8]

Historia de Usuario				8
Usuario:		Jugador		
Nombre historia:		Animación de Personajes		
Prioridad en negocio:		Media	Riesgo en desarrollo:	Alto
Puntos estimados:		80	Iteración asignada:	2
Programador responsable:		Marcelo Agila		
Descripción: La animación de personajes es un proceso en el cual se crearan movimientos para cada personaje.				
Observaciones: Las animaciones se pueden seguir efectuando mientras se continúa con el resto de la aplicación.				
Precondición: Construcción de Personajes				

Tabla 2.25 Historia Usuario8 [8]

Historia de Usuario				9
Usuario:		Jugador		
Nombre historia:		Construcción Escenas del Juego		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	
			Bajo	
Puntos estimados:		12	Iteración asignada:	
			3	
Programador responsable:		Andrés Cevallos		
Descripción: Las escenas de juego son construidas a base de los assets. Se construyen en base del diseño.				
Observaciones: Ninguna				
Precondición: Diseño de Niveles				

Tabla 2.26 Historia Usuario9 [8]

Historia de Usuario				10
Usuario:		Jugador		
Nombre historia:		Diseño Escenas del Juego		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	
			Bajo	
Puntos estimados:		12	Iteración asignada:	
			2	
Programador responsable:		Andrés Cevallos		
Descripción: El diseño de las escenas es el diseño estético y la forma que va a tener cada nivel. Se toman consideraciones visuales para generar un nivel atractivo y consideraciones de jugabilidad para presentar un reto para el jugador				
Observaciones: Ninguna				
Precondición: Niveles				

Tabla 2.27 Historia Usuario10 [8]

Historia de Usuario				11
Usuario:		Jugador		
Nombre historia:		Ambientes y Iluminación		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	
			Bajo	
Puntos estimados:		12	Iteración asignada:	
			3	
Programador responsable:		Andrés Cevallos		
Descripción: La ambientación e iluminación se agregan después de la construcción del nivel. Se debe considerar que el usuario pueda ver y las figuras sean claras.				
Observaciones: Ninguna				
Precondición: Construcción de Niveles				

Tabla 2.28 Historia Usuario11 [8]

Historia de Usuario				12
Usuario:		Jugador		
Nombre historia:		Clase ThirdPersonController		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	Bajo
Puntos estimados:		12	Iteración asignada:	3
Programador responsable:		Andrés Cevallos		
Descripción: Esta clase permitirá el movimiento de los personajes dentro de los niveles. Se debe adaptar de tal manera.				
Observaciones: Ninguna				
Precondición: Construcción de Niveles				

Tabla 2.29 Historia Usuario12 [8]

Historia de Usuario				13
Usuario:		Jugador		
Nombre historia:		Cámara		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	Bajo
Puntos estimados:		12	Iteración asignada:	3
Programador responsable:		Andrés Cevallos		
Descripción: Se implementa la cámara para que siga al personaje. Con ello se comprueba la que el controller funcione correctamente.				
Observaciones: Ninguna				
Precondición: Construcción de Niveles				

Tabla 2.30 Historia Usuario13 [8]

Historia de Usuario				14
Usuario:		Jugador		
Nombre historia:		Clase ThirdPersonStatus		
Prioridad en negocio:		Alta	Riesgo en desarrollo:	Bajo
Puntos estimados:		12	Iteración asignada:	3
Programador responsable:		Andrés Cevallos		
Descripción: Se construye la clase que manejara el estado de un personaje.				
Observaciones:				
Precondición: Clase ThirdPersonController				

Tabla 2.31 Historia Usuario14 [8]

Historia de Usuario				15
Usuario:		Jugador		
Nombre historia:		Clase ThirdPersonAnimation		
Prioridad en negocio:		Alta	Riesgo en desarrollo: Bajo	
Puntos estimados:		12	Iteración asignada: 3	
Programador responsable:		Andrés Cevallos		
Descripción: La clase maneja las animaciones del personaje.				
Observaciones: La clase se implementa cuando se tenga completamente las animaciones del personaje				
Precondición: Clase ThirdPersonController, Animación de Personaje				

Tabla 2.32 Historia Usuario15 [8]

Historia de Usuario				16
Usuario:		Jugador		
Nombre historia:		Clase Enemy		
Prioridad en negocio:		Alta	Riesgo en desarrollo: Bajo	
Puntos estimados:		12	Iteración asignada: 3	
Programador responsable:		Andrés Cevallos		
Descripción: Construcción de la clase Enemy para el Gameobject Enemy, permite tener enemigos que ataquen al personaje.				
Observaciones:				
Precondición: Clase ThirdPersonController				

Tabla 2.33 Historia Usuario16 [8]

3. CAPÍTULO 3: FASE DE PRODUCCIÓN

3.1 CONSTRUCCIÓN DE AMBIENTES Y PERSONAJES

La construcción de ambientes abarca a los assets y personajes del videojuego. Se sigue un proceso para obtener una calidad gráfica apropiada. Antes de profundizar el tema se debe tener en cuenta algunos conceptos previos.

Modelado línea por línea [51]: consiste en aproximar la forma del objeto mediante el uso de líneas, las cuales, forman caras y dan la forma del objeto. Se usa la extrusión y la unión de vértices para construir la forma del objeto a realizarse.

Modelado en caja [51]: consiste en aproximar una forma mediante el uso de objetos de malla. Se construye una forma aproximada mediante extrusión⁷³. Luego se empieza a cortar mediante loops⁷⁴ y a detallar las formas que pueda tener el objeto a modelarse.

Escultura Digital [52]: este método consiste en crear un personaje con técnicas parecidas a la de la escultura tradicional del arte. Se crea una forma básica con modelado en caja o línea por línea, luego, se procede a detallar el modelo incrementando el número de polígonos que tiene la figura, brindándole más detalle. Es ampliamente usado para el modelado orgánico⁷⁵.

Escultura Dinámica [52]: este método es similar al de la escultura digital, pero, permite detallar partes exclusivas de un modelo sin la necesidad de incrementar el número de polígonos.

Se debe mencionar que las tareas de construcción y animación de personajes por ser actividades extendidas en el tiempo se repiten en varios sprints como tareas recurrentes.

3.1.1 PRIMER ENTREGABLE R1-SEGUNDO SPRINT-CONTINUACIÓN

En este capítulo se retoman las tareas definidas en el segundo sprint del entregable R1 que no son parte del segundo capítulo.

⁷³ Alargar una cara o vértices de un objeto 3D

⁷⁴ En loop es una línea consecutiva que une varios vértices.

⁷⁵ Es el modelado de criaturas orgánicas con formas más suavizadas como son los humanos y animales



Figura 3.1 R1-Segundo Sprint Tareas [8] [48]

3.1.1.1 Detalle de las tareas Entregable R1-Segundo Sprint

A continuación se detalla las tareas de este capítulo que son parte del entregable R1- segundo Sprint.



Figura 3.2 Tarea Modelado de Personajes [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Modelado de personajes	Selección de Bocetos	Terminada	Se debe seleccionar los distintos bocetos e importarlos a la herramienta modeladora para proceder a crear el personaje 3d.	Marcelo Agila	08/09/2012	08/09/2012
	Construcción de un Modelo Base	Terminada	Con la selección del boceto se procede a construir un modelo base, luego se podrá añadir detalles sobre el modelo.	Andrés Cevallos	10/09/2012	26/09/2012
	Escultura Digital	Terminada	Se esculpen los detalles en el modelo base	Andrés Cevallos	10/09/2012	26/09/2012
	Modelado de Accesorios y Ropa	Terminada	Se modela la ropa y accesorios que llevara el personaje.	Andrés Cevallos	10/09/2012	05/10/2012

Tabla 3.1 Subtareas Modelado de Personajes [8] [48]

Nombre : Modelado de Personajes Lowpoly

Tipo : Historia de usuario

Clasificación : 7

Descripción : Se debe convertir a los personajes highpoly en lowpoly, para que puedan ser renderizados en tiempo real

Notas :

Fecha de envío : 03/09/2012 23:37:53

Fecha de aceptación : 03/09/2012 23:38:54

Fecha de estimación : 03/09/2012 23:39:30

Fecha de planificación : 03/09/2012 23:41:03

En progreso desde : 31/10/2013 10:57:11

Fecha de terminación : 01/10/2012 11:26:16

Creador : Andres Cevallos

Característica : Construcción de personajes

18

Modelado de Perso...

21 Terminada

Figura 3.3 Tarea Modelado de Personajes LowPoly [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Modelado de personajes lowpoly	Unwrapping	Terminada	Se debe preparar al objeto para poder mapearlo en las dos dimensiones	Andrés Cevallos	05/10/2012	05/10/2012
	Retopología	Terminada	Se debe retopologizar ⁷⁶ el objeto teniendo en cuenta los principios de topología para que se pueda facilitar la animación.	Andrés Cevallos	05/10/2012	05/10/2012
	Retopología de Accesorios	Terminada	Accesorios que no vayan a ser animados (objetos sólidos) se los retopologiza siguiendo los principios de decimación ⁷⁷ .	Andrés Cevallos	05/10/2012	05/10/2012

Tabla 3.2 Subtarea Modelado de Personajes LowPoly [8] [48]

⁷⁶ Retopología es un proceso en el cual se reducen los polígonos de un objeto y se le brinda una topología adecuada.

⁷⁷ Proceso en el cual se mantiene la silueta del personaje pero se reducen los polígonos sin tomar en cuenta la topología. No es recomendable para objetos que deban animarse



Figura 3.4 Tarea Modelado de Assets para Escenarios [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Modelado de Assets para escenarios	Baking de Texturas	Terminada	Usando los modelos Highpoly ⁷⁸ y lowpoly obtenemos la mayor cantidad de detalles posibles realizando un baking de las texturas	Andrés Cevallos	05/10/2012	05/10/2012
	Pintar Texturas	Terminada	Debemos añadir la capa de color de las texturas	Andrés Cevallos	05/10/2012	05/10/2012
	Agregar Material	Terminada	Agregamos un material a los objetos 3d, añadimos nodos para que se vea mejor al momento de renderizar	Andrés Cevallos	05/10/2012	05/10/2012
	Corrección y Limpieza	Terminada	Se corrigen y se vuelve a realizar un bake de las texturas para que estas no tengan imperfecciones	Andrés Cevallos	05/10/2012	05/10/2012

Tabla 3.3 Subtareas Modelado de Assets para Escenarios [8] [48]

⁷⁸ Los modelos Highpoly o de altos polígonos son productos de la escultura digital que tienen a superar un millón de polígonos.



Figura 3.5 Tarea Texturizado de Assets [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Texturizado de assets	Establecer los objetos a modelar	Terminada	Usando los modelos Highpoly y lowpoly obtenemos la mayor cantidad de detalles posibles realizando un baking ⁷⁹ de las texturas	Marcelo Agila	05/10/2012	05/10/2012
	Establecer los objetos a modelar	Terminada	Debemos añadir la capa de color de las texturas	Marcelo Agila	05/10/2012	05/10/2012

Tabla 3.4 Subtarea Texturizado de Assets [8] [48]

3.1.2 PRIMER ENTREGABLE R1-TERCER SPRINT

En el tercer sprint del entregable R1 se repiten las tareas de modelado, a este sprint se agregan las tareas de animación y construcción del escenario.

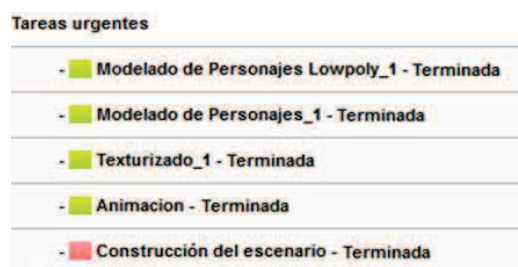


Figura 3.6 R1-Tercer Sprint Tareas [48]

⁷⁹ Baking es un proceso en el cual trasladamos la información geométrica de un modelo hacia las texturas de un modelo de bajos polígonos

3.1.2.1 Principales características del Entregable R1 - Tercer Sprint

Para realizar el tercer sprint se ha creado una característica llamada Construcción de Ambientes que se añade al resto de características.

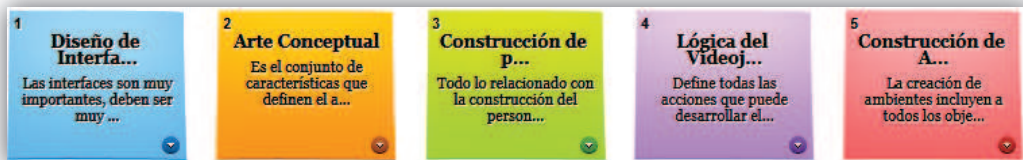


Figura 3.7 R1-Tercer Sprint Características [48]

Nombre	Descripción	Requerimientos
Construcción de ambientes y personajes	La creación de ambientes incluye a todos los objetos necesarios para crear un nivel, es decir los objetos que intervienen para la creación del terreno, la cámara, luces, entre otros.	Re4, Re5, Re7 y Re8 (Tabla 23)

Tabla 3.5 Detalle Principales Características R1-Tercer Sprint [8] [48]

3.1.2.2 Detalle de las tareas Entregable R1 - Tercer Sprint

A continuación se detalla las tareas nuevas que son parte de R1- Tercer Sprint, las que se repiten no se las detalla ya que el proceso es el mismo.



Figura 3.8 Tarea Animación [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Animación	Diseño Armature	Terminada	La estructura de huesos o armature debe adaptarse al modelo, se usa una estructura base de huesos a la que se añade o quita huesos dependiendo de la complejidad del modelo.	Marcelo Agila	04/11/2012	04/11/2012
	Weight Painting	Terminada	Asigna pesos por colores a la capacidad de movimiento que tenga el hueso sobre determinada zona del modelo. Se debe corregir para lograr movimientos no forzados	Marcelo Agila	04/11/2012	04/11/2012
	Rigging	Terminada	Cuando la estructura de huesos esta completa se añade el rigging, con esta propiedad los huesos se adaptan a la parte del modelo más cercana, una vez que se unen se puede dar movimiento	Marcelo Agila	04/11/2012	04/11/2012
	Constrains	Terminada	Los constrains son restricciones o propiedades que se añaden a determinados huesos para regular el movimiento o hacer que los huesos se muevan de manera secuencial	Marcelo Agila	04/11/2012	04/11/2012
	Rediseño Armature	Terminada	Se añaden huesos que no están dentro del weight painting, para controlar articulaciones a su vez ayudados por los constrains; esto permite la mayor fluidez de los movimientos al crear la animación.	Marcelo Agila	04/11/2012	04/11/2012
	Ciclo de Animación	Terminada	Al finalizar los constrains y los controles con huesos auxiliares se inician con la animación, consiste en mover las posiciones de los huesos del objeto a través de una línea de tiempo. Se relaciona con el weight painting ya que se realizan correcciones	Marcelo Agila	04/11/2012	04/11/2012

Tabla 3.6 Subtareas Animación [8] [48]

Nombre : Construcción del escenario

Tipo : Historia de usuario

Clasificación : 5

Descripción : En la construcción del escenario se integran los assets u objetos para la construcción del escenario

Notas :

Fecha de envío : 09/10/2012 11:51:42

Fecha de aceptación : 09/10/2012 11:52:03

Fecha de estimación : 09/10/2012 11:52:22

Fecha de planificación : 09/10/2012 11:52:37

En progreso desde : 09/10/2012 12:46:08

Fecha de terminación : 04/11/2012 12:51:00

Creador : Marcelo Agila

Característica : Construcción de Ambientes

37
Construcción del ...
21 Terminada

Figura 3.9 Construcción del Escenario [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Construcción de Escenario	Integración de Assets	Terminada	Los assets una vez modelados se los añade a un terreno, para su construcción.	Marcelo Agila	05/10/2012	05/10/2012
	Ambientación	Terminada	Debemos añadir la capa de color de las texturas, de objetos, etc.	Andrés Cevallos	05/10/2012	05/10/2012
	Iluminación	Terminada	Se añaden las luces de ambiente	Andrés Cevallos	05/10/2012	05/10/2012

Tabla 3.7 Subtareas Construcción del Escenario [8] [48]

Con el primer entregable terminado se puede visualizar el terreno, tres personajes modelados con texturas y dos de ellos animados. El siguiente entregable R2 contiene modelado de personajes, animación, incorporando la lógica dentro del motor gráfico además de la creación de la creación de menús y realización de pruebas.

3.1.3 SEGUNDO ENTREGABLE R2-PRIMER SPRINT

En el segundo entregable R2-Primer Sprint, se repiten las tareas de modelado y animación. A este sprint se agregan las tareas lógicas para usar al personaje principal dentro del motor gráfico.

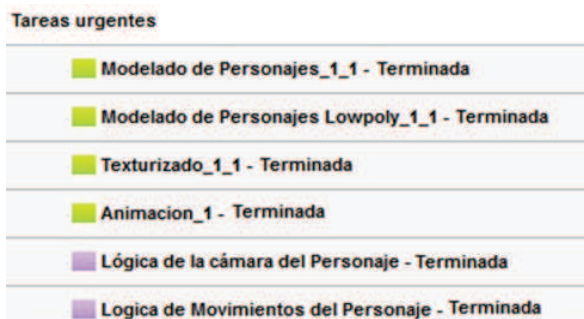


Figura 3.10 R2-Primer Sprint Tareas [8] [48]

3.1.3.1 Detalle de las tareas Entregable R2 - Primer Sprint

A continuación se detalla las tareas nuevas que son parte de R2- Primer Sprint, las tareas repetidas no se las detalla.

Nombre : Lógica de la cámara del Personaje
Tipo : Historia de usuario
Clasificación : 5
Descripción : Se implementan las clases para que funcione la cámara que segura al personaje en el transcurso del videojuego.
Notas :
Fecha de envío : 04/11/2012 13:13:06
Fecha de aceptación : 04/11/2012 13:16:12
Fecha de estimación : 04/11/2012 13:45:14
Fecha de planificación : 04/11/2012 13:20:19
En progreso desde : 04/11/2012 13:43:40
Fecha de terminación : 12/12/2012 13:53:35
Creador : Marcelo Agila
Característica : Lógica del Videojuego

41
 Lógica de la cámara...
21 Terminada

Figura 3.11 Tarea Lógica de Cámara del Personaje [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Lógica de la Cámara del Personaje	Implementación OrbitCamera	Terminada	Implementación y Construcción de la clase	Marcelo Agila	12/12/2012	12/12/2012
	Implementación GameHUD	Terminada	Implementación y construcción de la clase	Andrés Cevallos	12/12/2012	12/12/2012

Tabla 3.8 Subtareas Lógica de Cámara del Personaje [8] [48]

Nombre : Logica de Movimientos del Personaje
Tipo : Historia de usuario
Clasificación : 6
Descripción : Se construyen las clases necesarias para poder manejar al personaje con VO del usuario.
Notas :
Fecha de envío : 04/11/2012 13:14:35
Fecha de aceptación : 04/11/2012 13:16:14
Fecha de estimación : 04/11/2012 13:45:17
Fecha de planificación : 04/11/2012 13:20:21
En progreso desde : 04/11/2012 13:43:40
Fecha de terminación : 12/12/2012 13:53:39
Creador : Andres Cevallos
Característica : Lógica del Videojuego

42
 Logica de Movime...
21 Terminada

Figura 3.12 Tarea Lógica de Movimientos Personaje [8] [48]

<i>Tarea</i>	<i>Subtarea</i>	<i>Estado</i>	<i>Descripción</i>	<i>Responsable</i>	<i>Inicio</i>	<i>Fin</i>
Lógica de Movimientos de Personaje	Implementación Third Person Character Controller	Terminada	Implementación y Construcción de la clase	Marcelo Agila	12/12/2012	12/12/2012
	Implementación Third Person Character Animation	Terminada	Implementación y construcción de la clase	Andrés Cevallos	12/12/2012	12/12/2012

Tabla 3.9 Subtareas Lógica de Movimientos Personaje [8] [48]

3.1.4 SEGUNDO ENTREGABLE R2-SEGUNDO SPRINT

En el segundo entregable R2-segundo sprint, se repiten las tareas de modelado y animación. A este sprint se agrega la lógica para los movimientos de los enemigos y el ataque.



Figura 3.13 R2-Segundo Sprint Tareas [8] [48]

3.1.4.1 Detalle de las tareas Entregable R2 - Segundo Sprint

A continuación se detalla las tareas nuevas que son parte de R2- Segundo Sprint, las tareas repetidas no se las detalla.

43 Logica de Movimi...
21 Terminada

Nombre : Logica de Movimiento de Enemigos

Tipo : Historia de usuario

Clasificación : 5

Descripción : Se construye la logica necesaria para que los GameObjects enemies puedan moverse en el terreno.

Notas :

Fecha de envio : 04/11/2012 13:15:59

Fecha de aceptación : 04/11/2012 13:16:08

Fecha de estimación : 12/12/2012 14:01:27

Fecha de planificación : 19/11/2012 14:55:04

En progreso desde : 19/11/2012 15:10:36

Fecha de terminación : 18/01/2013 15:16:35

Creador : Andres Cevallos

Característica : Lógica del Videjuego

Figura 3.14 Tarea Lógica Movimiento Enemigo [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Lógica de Movimientos de Enemigo	Enemy	Terminada	Implementación de la clase, sirve para el movimiento y ataque del objeto Enemy	Andrés Cevallos	18/01/2013	18/01/2013

Tabla 3.10 Subtareas Lógica Movimiento Enemigo [8] [48]

44 Logica de Ataque
21 Terminada

Nombre : Logica de Ataque

Tipo : Historia de usuario

Clasificación : 6

Descripción : Se construyen las clases necesarias para que el personaje y los enemigos puedan hacerse daño

Notas :

Fecha de envio : 12/12/2012 14:01:12

Fecha de aceptación : 12/12/2012 14:01:17

Fecha de estimación : 12/12/2012 14:01:24

Fecha de planificación : 19/11/2012 14:55:01

En progreso desde : 19/11/2012 15:10:36

Fecha de terminación : 18/01/2013 15:16:40

Creador : Marcelo Agila

Característica : Lógica del Videjuego

Figura 3.15 Tarea Lógica de Ataque [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Lógica de Ataque	Implementación ThirdPerson Attack	Terminada	Implementación y Construcción de la clase de ataque	Andrés Cevallos	18/01/2013	18/01/2013
	Implementación clase Enemy Damage	Terminada	Implementación y construcción de la clase	Andrés Cevallos	18/01/2013	18/01/2013
	Enemy	Terminada	Implementación y construcción de la clase	Andrés Cevallos	18/01/2013	18/01/2013

Tabla 3.11 Subtareas Lógica de Ataque [8] [48]

3.1.5 SEGUNDO ENTREGABLE R2-TERCER SPRINT

En el segundo entregable R2-tercer sprint, se repiten las tareas de modelado y animación. A este sprint se agrega la lógica para los movimientos de los enemigos y el ataque; además de las tareas de prueba, se usa una tarea recurrente de corrección sobre las pruebas.

Tareas recurrentes			
Corrección	Terminada	0.0	Corrección sobre las pruebas

Figura 3.16 Tareas Recurrentes R2 -Tercer Sprint [4]

Tareas urgentes	
Diseñar el esquema Modulos Videojuego - Terminada	
Construcción MenuPrincipal - Terminada	
Costrucción Menú Opcion Nuevo - Terminada	
Construcción Menú Opcion Continuar - Terminada	
Construcción Menu Opción Créditos - Terminada	
Diseño de la Interfaz Status del Personaje - Terminada	
Animacion_1_1_1 - Terminada	
Pruebas de Unidad - Terminada	
Pruebas de Integración - Terminada	
Pruebas de Regresión - Terminada	

Figura 3.17 R2-Tercer Sprint Tareas [8] [48]

3.1.5.1 Principales características del Entregable R2 - Tercer Sprint

Para la realización del tercer sprint se ha creado una característica llamada Pruebas que se añade al resto de características.

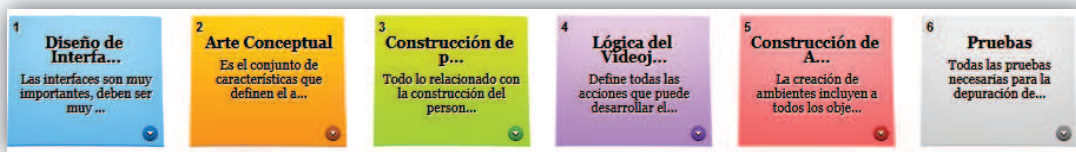


Figura 3.18 R2-Tercer Sprint Características [8] [48]

Nombre	Descripción	Requerimientos
Pruebas	Todas las pruebas necesarias para la depuración del videojuego en la etapa de desarrollo, pruebas alfa y beta	Re1, Re2, Re3, Re4, Re5, Re6, Re7, Re8 (Tabla 23)

Tabla 3.12 Detalle Principales Características R2- Tercer Sprint [8] [48]

3.1.5.2 Detalle de las tareas Entregable R2 - Tercer Sprint

A continuación se detalla las tareas nuevas que son parte de R2- Tercer Sprint, las que se repiten no se las detalla ya que el proceso es el mismo.



Figura 3.19 Tarea Construcción Menú Principal [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Construcción del Menú Principal	Construcción de la clase MenuPrincipal	Terminada	Se construye la clase que define el menú principal estableciendo cada una de sus opciones	Andrés Cevallos	14/02/2013	14/02/2013
	Integración con el Menú Principal	Terminada	El menú principal se integra con las escenas en el videojuego	Andrés Cevallos	14/02/2013	14/02/2013

Tabla 3.13 Subtareas Construcción Menú Principal [8] [48]



Figura 3.20 Tarea Construcción Menú Opción Nuevo [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Construcción del Menú Opción Nuevo	Implementar la opción Nuevo	Terminada	Se implementa la opción Nuevo en el Menú Principal	Andrés Cevallos	20/02/2013	20/02/2013

Tabla 3.14 Subtareas Construcción Menú Opción Nuevo [8] [48]



Figura 3.21 Tarea Construcción del Menú Opción Continuar [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Construcción del Menú Opción Continuar	Implementar la opción Continuar	Terminada	Se implementa la opción Continuar en el Menú Principal	Andrés Cevallos	20/02/2013	20/02/2013

Tabla 3.15 Subtareas Construcción del Menú Opción Continuar [8] [48]



Figura 3.22 Tarea Construcción del Menú Opción Créditos [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Construcción del Menú Opción Créditos	Implementar la opción Créditos	Terminada	Se implementa la opción Créditos en el Menú Principal	Andrés Cevallos	20/02/2013	20/02/2013

Tabla 3.16 Subtareas Construcción del Menú Opción Créditos [8] [48]



Figura 3.23 Tarea Diseñar Esquema Módulos Videojuego [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Diseñar Esquema Módulos Videojuego	Establecer relación de objetos	Terminada	Se establece la comunicación entre las clases de cada game object, su interacción dentro del videojuego.	Marcelo Agila	11/02/2013	11/02/2013
	Dibujar el esquema	Terminada	Se dibuja el esquema, que servirá para establecer las pruebas de regresión.	Marcelo Agila	11/02/2013	11/02/2013

Tabla 3.17 Subtareas Diseñar Esquema Módulos Videojuego [8] [48]



Figura 3.24 Tarea Prueba de Unidad [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Prueba de Unidad	Diseño de caso de pruebas	Terminada	Se diseña una tabla con las opciones pertinentes para el desarrollo de las pruebas unitarias	Marcelo Agila	27/03/2013	27/03/2013
	Evaluación	Terminada	Cada vez que se establece una funcionalidad se evalúa el caso de pruebas	Marcelo Agila	27/03/2013	27/03/2013

Tabla 3.18 Subtareas Prueba de Unidad [8] [48]



Figura 3.25 Tarea Prueba de Integración [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Pruebas de Integración	Revisar Diagrama Integración	Terminada	Cada funcionalidad añadida debe funcionar con todo el software no solo unitariamente, se comprueba desde el módulo añadido en forma ascendente, es decir a partir de la prueba unitaria.	Marcelo Agila	29/03/2013	29/03/2013

Tabla 3.19 Subtareas Prueba de Integración [8] [48]



Figura 3.26 Tarea Prueba de Regresión [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Pruebas de Integración	Establecer secuencia	Terminada	Se revisa la secuencia de todos los módulos cada vez que una nueva funcionalidad sea añadida.	Marcelo Agila	29/03/2013	29/03/2013

Tabla 3.20 Subtareas Prueba de Regresión [8] [48]

3.1.6 TERCER ENTREGABLE R3-PRIMER SPRINT

En el primer sprint del entregable R3 se repiten las tareas de y animación y las pruebas, a este sprint se agregan las tareas para la elaboración de CutScenes y la depuración del escenario de nivel 1.

Tareas recurrentes			
Corrección	Terminada	0.0	Corrección sobre las pruebas

Figura 3.27 Tareas Recurrentes R3 -Primer Sprint [8] [48]



Figura 3.28 R3-Primer Sprint Tareas [8] [48]

3.1.6.1 Detalle de las tareas Entregable R3 - Primer Sprint

A continuación se detalla las tareas nuevas que son parte de R3-Primer Sprint, las tareas repetidas no se las detalla.



Figura 3.29 Tarea CutScenelvl1 [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
CutSceneslv1	Motion Path	Terminada	Mediante la Tecnología iTween se construye el movimiento de las escenas de corte	Andrés Cevallos	31/05/2013	31/05/2013
	Creación de Diálogos	Terminada	Se debe construir los diálogos de lv1	Andrés Cevallos	31/05/2013	31/05/2013
	Lanzador de Diálogos	Terminada	Se construye los lanzadores para los diálogos	Andrés Cevallos	31/05/2013	31/05/2013

Tabla 3.21 Subtareas CutScenelvl1 [8] [48]

Nombre : Depuración Escenario1
 Tipo : Historia de usuario
 Clasificación : 3
 Descripción : Se depuran los efectos añadidos
 Notas :
 Fecha de envío : 02/04/2013 17:19:52
 Fecha de aceptación : 02/04/2013 17:22:30
 Fecha de estimación : 02/04/2013 17:22:44
 Fecha de planificación : 02/04/2013 17:23:25
 En progreso desde : 02/04/2013 17:23:25
 Fecha de terminación : 19/06/2013 17:58:25
 Creador : Marcelo Agila
 Característica : Construcción de Ambientes

Figura 3.30 Tarea Depuración Escenario lvl 1 [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Depuración Escenario lvl1	Añadir Efectos Visuales	Terminada	Se añade efectos como sangre, fuego, lluvia	Andrés Cevallos	19/06/2013	19/06/2013
	Añadir Sonidos	Terminada	Se añade sonidos ambientales, de los objetos, del personaje,	Andrés Cevallos	19/06/2013	19/06/2013

Tabla 3.22 Subtareas Depuración Escenario lvl 1 [8] [48]

3.1.7 TERCER ENTREGABLE R3-SEGUNDO SPRINT

En el segundo sprint del entregable R3 se repiten las tareas de pruebas, de creación de CutScenes y Depuración de escenario del nivel 2.

Tareas recurrentes

Corrección	Terminada	0.0	Corrección sobre las pruebas
------------	-----------	-----	------------------------------

Figura 3.31 Tareas Recurrentes R3 -Segundo Sprint [8] [48]

Tareas urgentes

CutScenas lvl2 - Terminada
Depuración Escenario2 - Terminada
Pruebas de Regresión lvl2 - Terminada
Pruebas de Integración lvl2 - Terminada
Pruebas de Unidad lvl2 - Terminada

Figura 3.32 R3-Segundo Sprint Tareas [8] [48]

3.1.7.1 Detalle de las tareas Entregable R3 - Segundo Sprint

Las tareas para este sprint se las consideran recurrentes, es decir son iguales a las del anterior sprint, pero actúan sobre un nivel diferente, por este motivo no se las detalla. El detalle total de las tareas se encuentra como anexo, en la herramienta ICESCRUM.

3.1.8 TERCER ENTREGABLE R3-TERCER SPRINT

En el tercer sprint del entregable R3, se repiten las tareas de pruebas, de creación de CutScenes y Depuración de escenario. Se agregan las tareas para la creación del escenario inicial y final, las pruebas alfa y beta.

Tareas recurrentes			
Corrección	Terminada	0.0	Corrección sobre las pruebas

Figura 3.33 Tareas Recurrentes R3 -Tercer Sprint [8] [48]

Tareas urgentes	
■	CutScenas lv13 - Terminada
■	Depuración Escenario3 - Terminada
■	Creación de Escena Inicial - Terminada
■	Creación de escena final - Terminada
■	Pruebas de Unidad lv13 - Terminada
■	Pruebas de Integración lv13 - Terminada
■	Pruebas de Regresión lv13 - Terminada
■	Pruebas Alfa - 8 - Terminada
■	Pruebas Beta - 3 - Terminada

Figura 3.34 R3-Tercer Sprint Tareas [8] [48]

Este capítulo no termina con la totalidad de las tareas del sprint por eso se ha separado las tareas que forman parte de este capítulo.



Figura 3.35 Tareas R3-Tercer Sprint Capitulo 3 [8] [48]

3.1.8.1 Detalle de las tareas Entregable R3 - Tercer Sprint

A continuación se detalla las tareas nuevas que son parte de R3- Tercer Sprint, las que se repiten no se las detalla.



Figura 3.36 Tarea Creación de Escena Inicial [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Creación de Escena Inicial	Integración de Assets	Terminada	Se Integran los diferentes objetos que compondrán la escena final.	Andrés Cevallos	27/09/2013	27/09/2013
	Integración de Animaciones	Terminada	Se unen las animaciones de los objetos añadidos dentro de la escena.	Andrés Cevallos	27/09/2013	27/09/2013

Tabla 3.23 Subtareas Creación de Escena Inicial [8] [48]



Figura 3.37 Tarea Creación de Escena Final [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Creación de Escena Final	Integración de Assets	Terminada	Se Integran los diferentes objetos que compondrán la escena final	Andrés Cevallos	10/10/2013	10/10/2013
	Integración de Animaciones	Terminada	Se unen las animaciones de los objetos añadidos dentro de la escena	Andrés Cevallos	10/10/2013	10/10/2013

Tabla 3.24 Subtareas Creación de Escena Final [8] [48]

Una vez que se ha definido y detallado las tareas de los entregables R2, y parte del R3, se procede con la documentación del trabajo realizado en cada una de las tareas en el presente documento.

3.1.9 CONSTRUCCION DE PERSONAJES [52]

3.1.9.1 Definición de Detalles

Con los conceptos previos se puede detallar de mejor forma la construcción del personaje, para construir un personaje primero se debe realizar un modelo base. El modelo base se construye mediante escultura dinámica, la cual permite de un simple cubo realizar formas complejas. La base obtenida se presenta en la Figura 3.38.



Figura 3.38 Construcción de Personajes Modelo Base [8] [39]

3.1.9.2 Definición de Detalles

Una vez que el modelo base ha sido obtenido se da detalles al objeto o personaje. Se usa diferentes técnicas como el modelado de caja y escultura digital. Para este personaje se construyó el cabello usando escultura dinámica y para la ropa se usó el modelado de caja. Al aplicar los detalles se obtiene el modelo de altos polígonos.



Figura 3.39 Construcción de Personajes Modelo de Altos Polígonos [8] [39]

Ahora los vértices del plano son ajustados a los del modelo de altos polígonos.

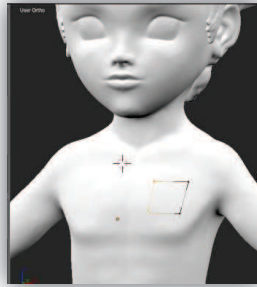


Figura 3.40 Construcción de Personajes ajuste de Vértices [8] [39]

Usando un identificador de espejo se obtiene simetría en el modelo. Con la ayuda de la extrusión se cubre el modelo de altos polígonos con el plano añadido; de esta manera se obtiene un modelo de bajos polígonos.

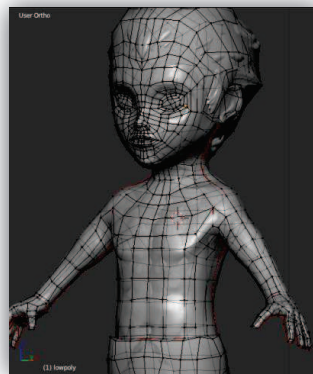


Figura 3.41 Construcción de Personajes Modelo de Bajos Polígonos [8] [39]

3.1.9.3 Unwrapping

Una vez finalizada la construcción del modelo de bajos polígonos se inicia la etapa de texturizado. Para ello, se debe mapear el objeto 3d en una imagen tridimensional. Para que el mapeado sea correcto se usó un proceso conocido como unwrapping, el cual permite segmentar partes de la figura tridimensional en una imagen 2d. Se marca segmentos de corte para que puedan ser trasladados correctamente a una imagen bidimensional. Se selecciona los vértices los cuales se convertirán en el punto de corte.

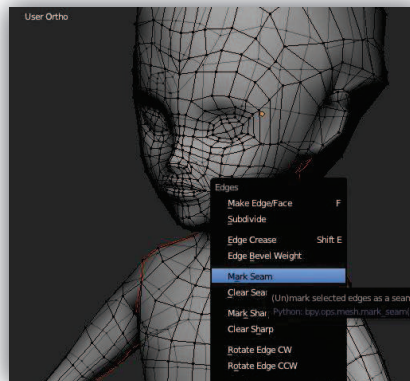


Figura 3.42 Construcción de Personajes Segmentación Modelo [8] [39]

Se debe poner especial atención en realizar puntos de corte en los brazos el pecho las piernas y la cabeza.

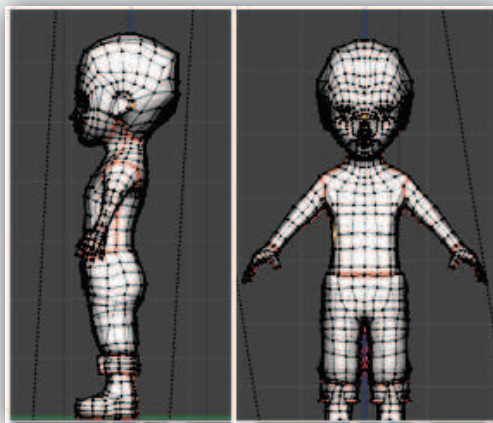


Figura 3.43 Construcción de Personajes Segmentación Modelo Puntos de Corte [8] [39]

Se realiza el Unwrapping, seleccionando todos los vértices del modelo. El resultado se aprecia al abrir el editor de imágenes.

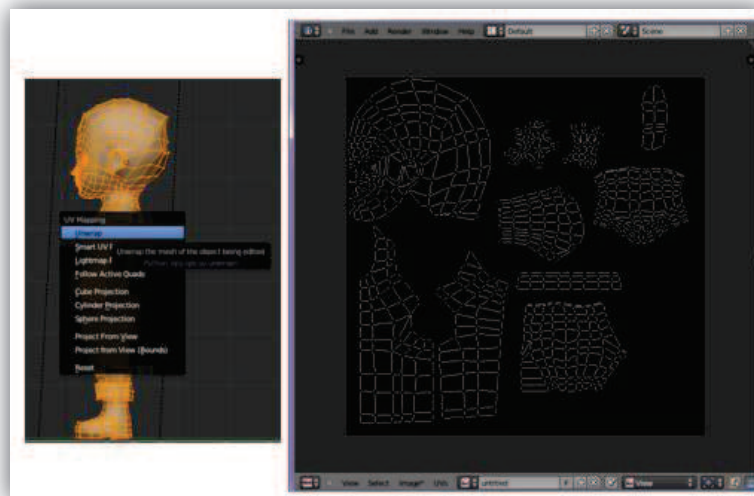


Figura 3.44 Construcción de Personajes Unwrapping [8] [39]

3.1.10 CONSTRUCCION DE ASSETS

3.1.10.1 Modelado de Assets

Los Assets o activos son objetos tridimensionales que se encuentran dentro del ambiente como plantas, lámparas, puentes. Se debe considerar el número de vértices que se empleará en cada objeto, para disminuir el uso de recursos. Se los construye usando el modelo en caja. Se realiza forma básica del objeto, en este caso una casa.

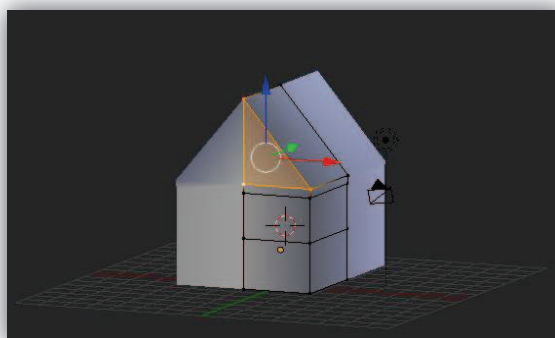


Figura 3.45 Modelado de Assets [8] [39]

Se añade loops adicionales y se extrude para agregar los detalles; con la ayuda de objetos adicionales el detalle de los assets aumenta.

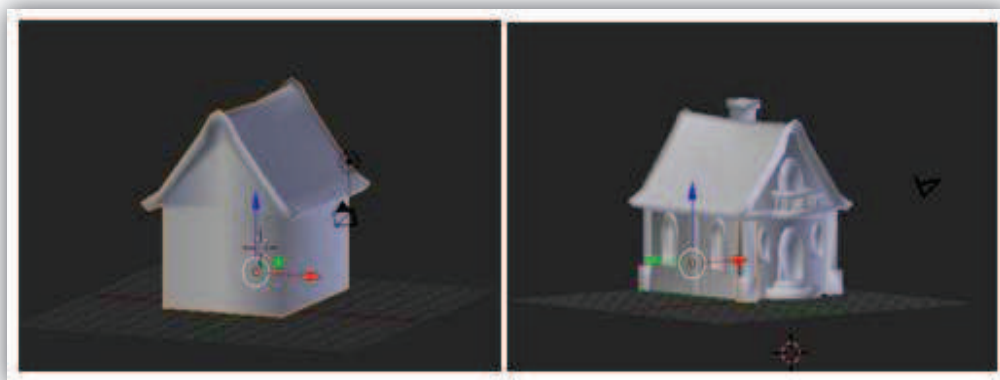


Figura 3.46 Modelado de Assets Detalles Adicionales [8] [39]

3.1.10.2 Unwrapping y Texturizado

El unwrapping es similar al realizado en los personajes, se lo realiza al objeto general y a los objetos usados para detalles añadidos.

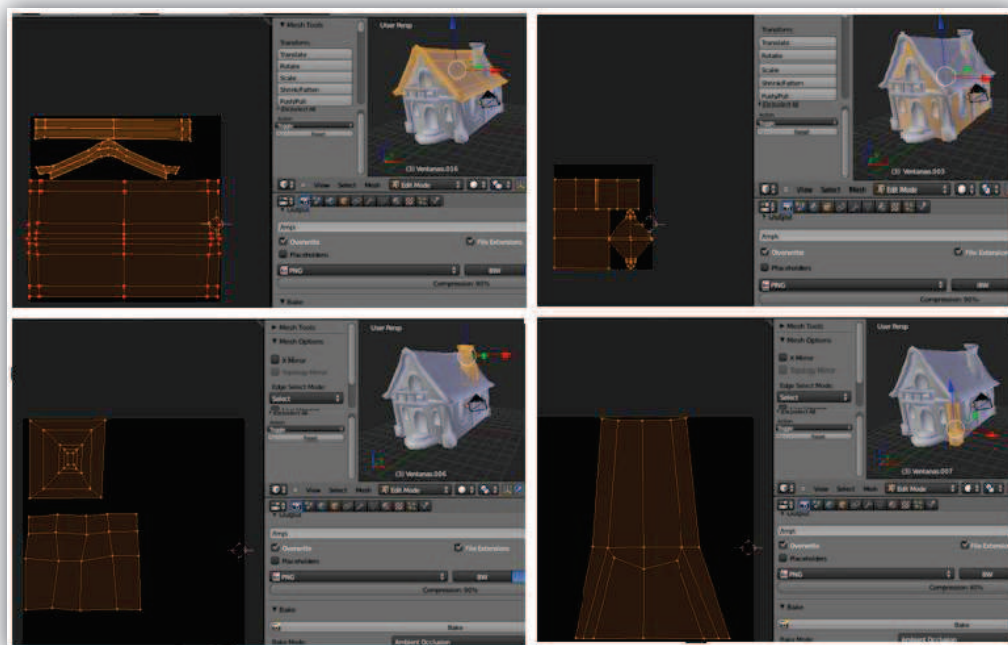


Figura 3.47 Unwrapping de Assets [8] [39]

Para el texturizado se exportan las imágenes obtenidas con el unwrapping a un editor de imágenes como Gimp, se procede a realizar pintura de capas

sobre la imagen exportada para obtener las texturas.



Figura 3.48 Texturizado de Assets [8] [39]

3.1.11 CONSTRUCCION DEL TERRENO

La construcción del terreno se la realiza mediante la integración de los assets a la escena. Se añade además un mesh collider⁸⁰ a cada uno de los objetos.

3.1.11.1 Construcción del terreno.

Para la construcción del terreno se importan los assets creados al motor de juegos. Posteriormente, se ubican de acuerdo a la forma que se le quiera dar al terreno y se construye el nivel.

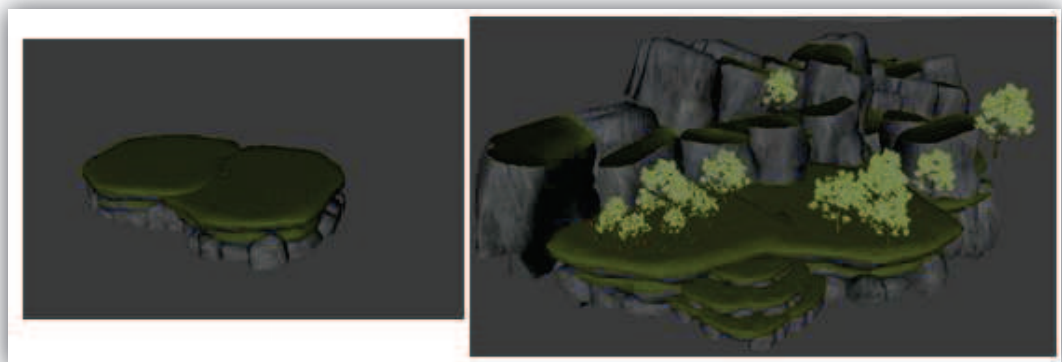


Figura 3.49 Construcción del terreno [8] [33]

Con el proceso terminado se visualiza el diseño total del terreno.

⁸⁰ Un mesh collider sirve para medir las colisiones tomando en cuenta la forma del objeto tridimensional.

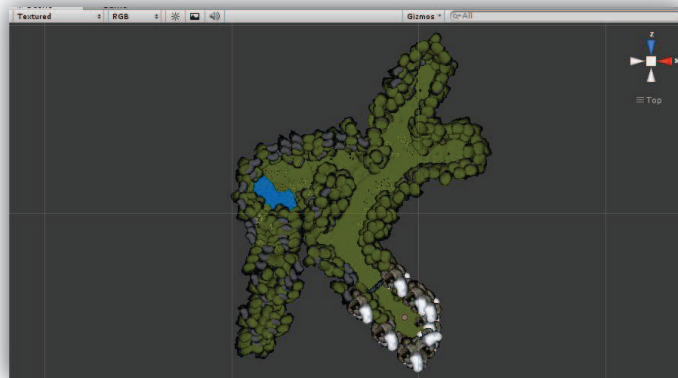


Figura 3.50 Vista superior terreno construido [8] [33]

3.1.11.2 Ambientación e iluminación.

Para la ambientación se usan materiales renderizados propios del motor gráfico, esto ayuda a mejorar la calidad gráfica de los niveles. El cielo se agrega insertando un material de tipo skybox en las opciones de renderizado.



Figura 3.51 Ambientación, uso de SkyBox [8] [33]

Se usa la opción de renderizado Ambient Light para añadir una luz blanca que permite visualizar los objetos de mejor forma.



Figura 3.52 Ambientación, uso de luces [8] [33]

Se usa la opción para crear neblina y optimizar el número de polígonos que son renderizados con texturas, con esto se optimiza el desempeño de de la aplicación.

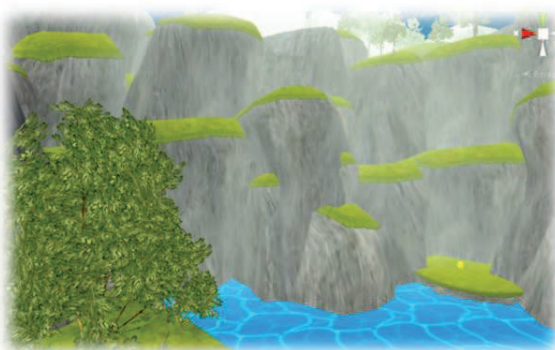


Figura 3.53 Ambientación, uso de efectos-Neblina [8] [33]

Efectos como fuego, agua, entre otros., se incorporan para mejorar el ambiente.



Figura 3.54 Ambientación, uso de efectos-Lluvia [8] [33]

3.1.12 INCORPORACION DE SONIDOS

El sonido dentro de los videojuegos es fundamental. Se incorpora con las acciones y los efectos para dar realismo al ambiente creado, mejorando su calidad. Se incorporan sonidos ambientales y de fondo.

3.1.12.1 Sonidos de Fondo

Los sonidos de fondo se usan para manejar el estado anímico de una escena. Generalmente se incorpora música que denote emociones acordes al evento que suceda. Se usan canciones gratuitas bajo licencia Creative Commons obtenidas principalmente de dos páginas <http://incompetech.com/> [53] y bybycopyright canal de youtube. Una vez descargado el sonido se añade a una de las cámaras mediante el ComponenteAudio Source en la opción AudioClip.

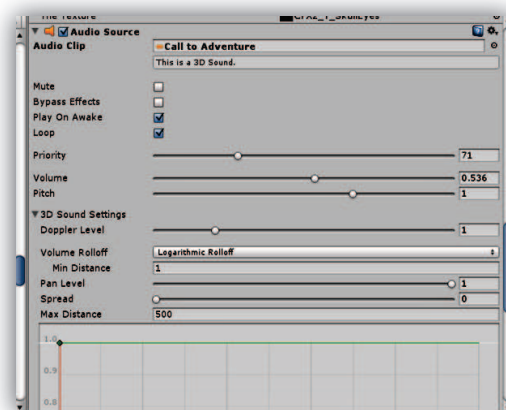


Figura 3.55 Incorporación de Sonidos [8] [33]

Hay varias opciones para la edición del sonido. Se marcan las opciones loop y play on awak, de esta forma se puede iniciar y repetir un ciclo durante la escena.

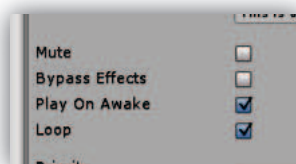


Figura 3.56 Incorporación Sonidos Fondo [8] [33]

3.1.12.2 Sonidos Ambientales

Los sonidos ambientales se incluyen en una escena para brindarle mayor realismo. Se puede encontrar varios sonidos ambientales gratuitos en internet; de no encontrarlos se pueden grabar algunos. Los sonidos ambientales se incluyen en los GameObject añadiendo el componente AudioSource.

Se deben tomar en cuenta las características del sonido y usar las opciones de AudioSource para priorizar los sonidos.

3.1.12.3 Sonidos de Personajes y NPC

Los sonidos de los personajes y NPC se reproducen cuando ejecutan una acción, por ejemplo cuando ataca el personaje, cuando los enemigos mueren, entre otras acciones. Se necesita añadir un componente AudioSource a cada GameObject que produzca un sonido, pero en este caso el sonido se añade mediante código en funciones dentro de los scripts que controlan al personaje, enemigos, entre otros.

3.1.13 ESCENAS DE CORTE

3.1.13.1.1 Diálogos.

Construir las escenas de corte es el proceso final para completar un nivel. Primero se construye los diálogos usando la extensión LocalizedDialogs and Cutscenes⁸¹ que se encuentra en asset store. Los diálogos se construyen en base al guion.

⁸¹LocalizedDialogs and Cutscenes es una extensión de Editor de Unity construida por Mel Georgio.

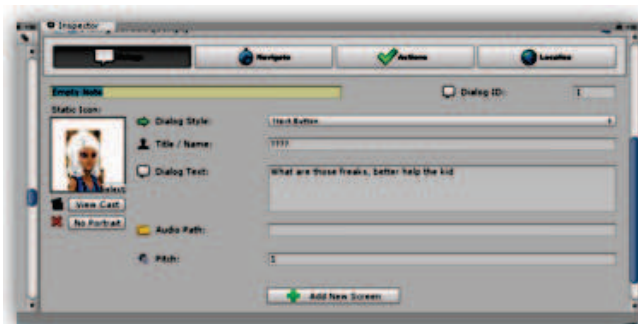


Figura 3.57 Construcción de Diálogo Escena de Corte [8] [33]

Una vez que los diálogos estén contruidos, se ingresa la funcionalidad para reproducirlos. Los diálogos se llaman a través de varias clases, DialogTrigger.js activa el dialogo cuando se cambia de lugar o se ha llegado a un objetivo.



Figura 3.58 Reproducción de Diálogo [8] [33]

La clase LevelStatus.js llama a un diálogo cuando se cumple un objetivo (se ha recolectado un número de items, ha venciendo a un numero de enemigos, entre otros.).



Figura 3.59 Reproducción de Diálogo Objetivo Completo [8] [33]

Existen diálogos que al llamar a la escena se reproducen automáticamente usando la función de los dialogos `AutoPlay()`.

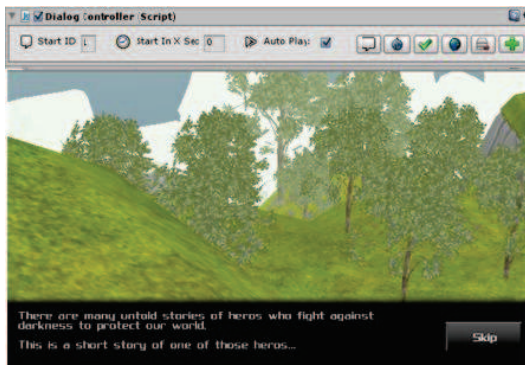


Figura 3.60 Reproducción Diálogo `AutoPlay` [8] [33]

3.1.13.1.2 *Movimiento de Personajes y Cámaras.*

Muchas Escenas de corte además de incorporar diálogos, incorporan movimiento de cámaras y movimiento de los personajes. Se usa la librería de clases `iTween` y los editores de camino y eventos de `iTween` disponibles en asset store. Primero se construye un `mothion path`, se incorpora un `EmptyObject` a la escena, añadiéndole el componente `iTweenPath`. Esto permite de manera grafica encontrar los puntos por los cuales pasaran los objetos.

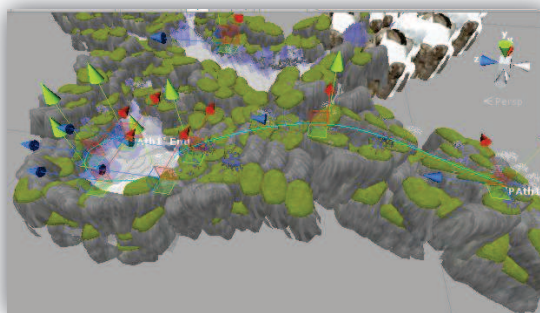


Figura 3.61 Camino de Escena [8] [33]

Para que un `GameObject` se mueva de acuerdo al path es necesario añadir un componente al `GameObject`, este se moverá a través del terreno. El componente es la clase `iTweenEvent.js`. Se configura el componente para que siga el camino deseado a una velocidad

deseada.

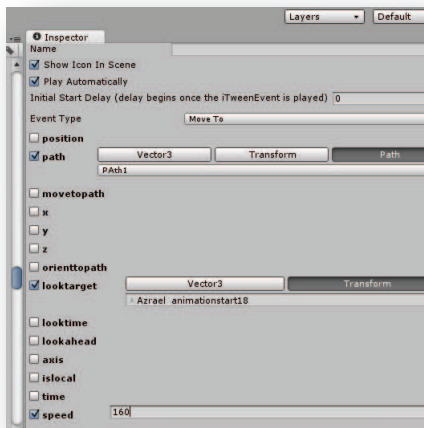


Figura 3.62 Configuración iTweenEvent.js [8] [33]

3.1.14 TEXTURIZADO DE PERSONAJES Y OBJETOS

Para finalizar la construcción de los personajes y objetos se usa el proceso de baking; este proceso permite mapear los detalles del modelo de altos polígonos en imágenes bidimensionales.

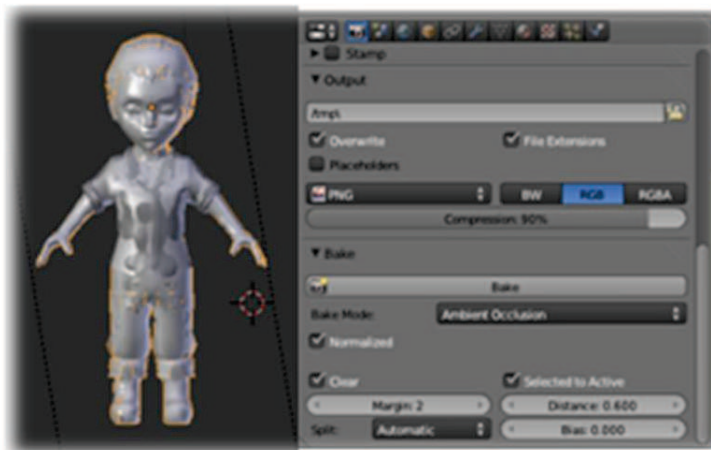


Figura 3.63 Configuración del proceso de Baking [8] [39]

Se realiza primero el baking de oclusión ambiental. El resultado de este proceso es una imagen 2d, en la cual se obtienen los detalles de iluminación del modelo de altos polígonos.



Figura 3.64 Imagen Mapeada del Modelo [8] [39]

En el proceso de baking de texturas normales se obtiene una imagen en la cual se encuentra mapeados los detalles geométricos del modelo de altos polígonos.



Figura 3.65 Imagen Mapeada con detalles Geométricos del Modelo [8] [39]

Debido a la dificultad que presenta crear un gran número de objetos 3d, se ha obtenido algunos objetos de páginas como: Unity Asset Store [54], 3dOcean [55], BlenderSwap [56], OpenGameArt.org [57].

3.2 DISEÑO DE INTERFACES

La construcción de las interfaces incluye los menús, la interfaz de juego (in-game), el diseño de la cámara y el movimiento que debe tener al seguir al personaje dentro del escenario. Los detalles de la construcción de interfaces se presentan a continuación.

3.2.1 CONSTRUCCIÓN DE MENUS

Los menús se construyen mediante código usando la función OnGUI propia de Unity, esta función permite enviar a la pantalla botones e imágenes que son generalmente usadas para creación de menús.

3.2.1.1 Construcción del Menú Principal.

El menú principal tendrá las opciones: Nuevo Juego, Continuar, Opciones y créditos, la Figura 3.66 presenta el diseño.

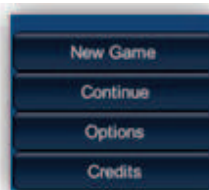


Figura 3.66 Diseño del Menú Principal [8] [33]

El botón Nuevo Juego instanciará la primera escena de la aplicación, de esta manera se puede iniciar el juego.

```
if(GUI.Button(Rect(Screen.width-150,Screen.height-120,150,30),"New Game"))
{
    print("starting new game..");
    Application.LoadLevel('Beggining');
}
```

Figura 3.67 Asignación del Botón New Game a la primera escena del juego. [8] [33]

El botón Continuar llamará a la última sesión guardada. De no existir esta sesión, el botón no se desplegará.

```
if(GUI.Button(Rect(Screen.width-150,Screen.height-90,150,30),"Continue"))
{
    print("Continue..");
    LevelSerializer.Resume();
}
```

Figura 3.68 Asignación del Botón Continue a la última sesión guardada [8] [33]

El botón Opciones despliega un menú en el que se puede manejar el volumen y el brillo de la pantalla.

El botón Créditos ocultara el menú y llamará a la animación que despliega los créditos del juego.

3.2.1.2 Construcción del Menú Opciones

El menú de opciones se puede acceder dentro del juego. Crea una opción de pausa y abre el menú de opciones. Este menú contiene la opción de subir y bajar el volumen o salir.

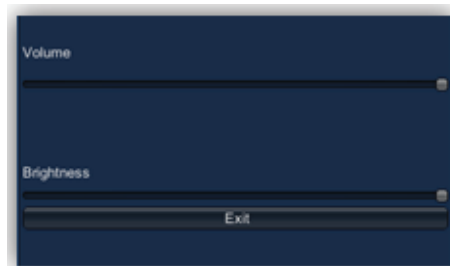


Figura 3.69 Resultado de ejecución del Botón Options [8] [33]

3.2.1.3 Construcción del Menú Game Over

El menú de Game Over aparece en la escena GameOver. La escena Game Over se activa cuando el personaje ha perdido todos los puntos de vida. Permite volver al menú principal o volver al último punto de guardado del videojuego.

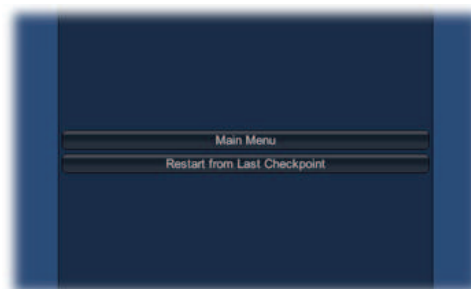


Figura 3.70 Escena Game Over [8] [33]

3.2.1.4 Integración con el videojuego.

Una vez que los menús han sido contruidos se procede a incluirlos en

las escenas. En el caso de los menús Main Menu y Game Over se los integra directamente en la cámara como componente.



Figura 3.71 Integración Menú Opciones y Game Over con el Videojuego [8] [33]

El Menú de Pausa se lo integra a la clase Pause.JS esta clase detiene el juego y muestra el menú opciones.



Figura 3.72 Pausa del Videojuego [8] [33]

3.2.2 CONSTRUCCION INTERFAZ DE JUEGO

La interfaz de juego (In-Game) se presenta al usuario mientras juega. Se debe considerar la posición de la cámara respecto al GameObjectPlayer o jugador, la distancia entre estos objetos, los ángulos de inclinación, etc.

A esta interfaz se agrega el indicador de salud, el minimapa y la flecha hacia el objetivo. Se consideran los bocetos para la construcción.

3.2.2.1 Cámara

La cámara se encuentra observando la espalda del personaje, éste se ubica en medio de la pantalla, define parte del gameplay.



Figura 3.73 Ubicación de Cámara respecto al personaje [8] [33]

Se ajusta las opciones de la cámara acercándola al personaje mejorando la visualización de los objetos. Se ajustan los ángulos máximos y mínimos en los que gira la cámara sobre el eje Y.

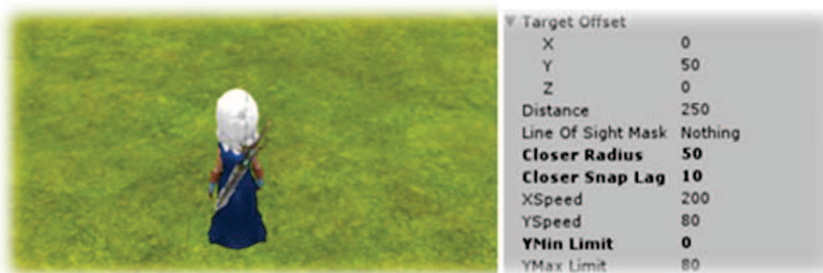


Figura 3.74 Ajuste de ángulos máximos y mínimos [8] [33]

Se ajusta la distancia en la que la cámara renderiza los objetos tridimensionales. Es importante dar una distancia prudencial para proveer una buena visión del terreno, además, optimizar el desempeño mediante un número correcto de triángulos.

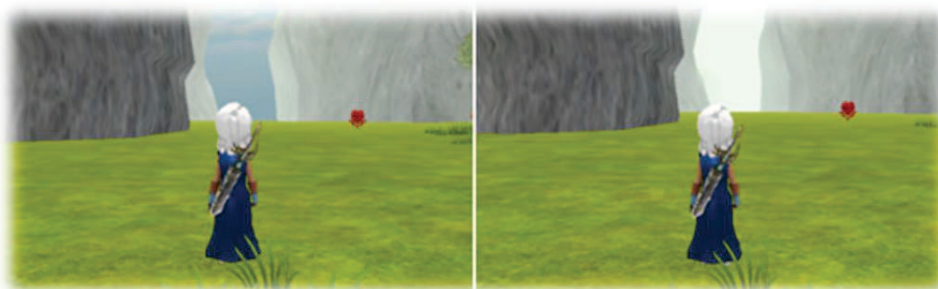


Figura 3.75 Ajuste de visualización [8] [33]

3.2.2.2 Indicador de Salud

El indicador de salud permite visualizar la vida del personaje, es decir, cuánto falta para que el personaje muera.

Para realizar el indicador de salud se debe construir un vector de imágenes que indiquen la vida que se ha perdido, en este caso hay un corazón que irá perdiendo su color.



Figura 3.76 Indicador de Salud [8] [33]

El objeto se ubica dentro de la escena usando la función `OnGUI()`. Los corazones son localizados en la esquina superior izquierda de la pantalla. Además se usa la misma función para escribir números sobre los corazones representando las vidas restantes y vidas máximas.



Figura 3.77 Integración de Indicador de Salud en la Interfaz In-Game [8] [33]

3.2.2.3 Indicador de Objetivo.

El indicador de objetivo señala al objeto o lugar, al cual el personaje debe dirigirse. Se usa un objeto 3D en forma de flecha, para darle una mejor visualización.



Figura 3.78 Objeto Indicador de Objetivo [8] [33]

A la flecha se la empata con el `GameObjectAzrael` y se añade el componente `DirectionalArrow.js` el cual rotará a la flecha de acuerdo a la posición del objetivo.



Figura 3.79 Integración de objeto indicador a la interfaz In-Game [8] [33]

La flecha se oculta dependiendo si existe o no un objetivo.

3.2.2.4 Minimapa.

El minimapa provee una vista superior que ayuda a identificar el camino a seguir. Aunque la flecha señala la ubicación hacia el objetivo, el minimapa da una perspectiva vertical del escenario, mejorando la visualización de la dirección del objetivo. El minimapa es una cámara extra que se encuentra emparentada con el personaje y muestra una vista perpendicular del mapa. Se crea un `GameObjectCamera` emparentando con el personaje. Se ajustan las opciones a `orthographic` para establecer el nivel de profundidad superior al de la cámara principal.

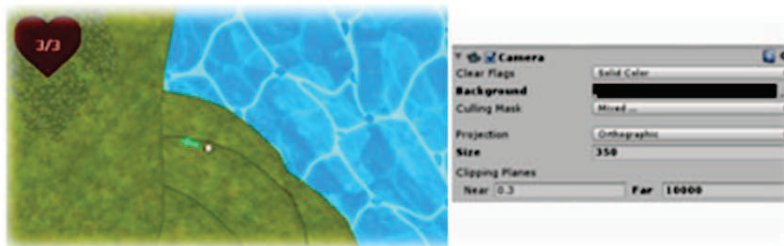


Figura 3.80 Ajuste Cámara Minimapa [8] [33]

Se debe ajustar el tamaño del minimapa, ubicarlo en la posición deseada, en este caso será la esquina superior derecha de la pantalla. De esta manera se visualizan las dos cámaras.



Figura 3.81 Integración del minimapa en la Interfaz In-Game [8] [33]

3.3 MODELADO Y ANIMACIÓN

Para establecer la animación de los personajes, se debe contar con una estructura base llamada armature. Esta armature está atada a la malla del personaje fungiendo como los huesos lo hacen en los seres vivos. Para la realización de las animaciones, se debe conocer la esencia del movimiento, empleando creatividad.

3.3.1 MODELADO DE ARMATURES Y RIGGING [58]

Las armatures deben coincidir con los polígonos del modelo creado, por esta razón, se hace un diseño sobre el modelo. El proceso de rigging, emplea la posición de los huesos sobre la malla para asignar movimiento a cada parte del modelo de bajos polígonos.

3.3.1.1 Construcción de Armatures

La construcción del armature inicia al añadir un objeto propio de Blender. En este caso, se usa una de las extensiones del programa modelador para obtener una estructura de huesos similar a la del ser humano, llamada *Human*.



Figura 3.82 Estructura Human Rigging [8] [39]

Cada uno de los huesos se deben adaptar al modelo, esto se logra gracias a las propiedades de movimiento y extrusión de los huesos. Además se corrigen los ejes para darle una mejor posición respecto al modelo [58].

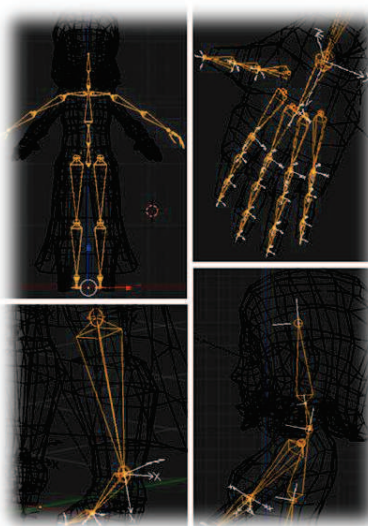


Figura 3.83 Diseño del Armature, posicionamiento y corrección de ejes sobre el modelo [8] [39]

Si el modelo no coincide con las estructura de los huesos, se elimina o añade huesos, dependiendo de lo que se quiera obtener. Los huesos deben estar emparentados con la estructura principal. Este proceso se usa principalmente cuando se importa objetos secundarios, como espadas y se los añade a la estructura esqueleto principal para darle movimiento.

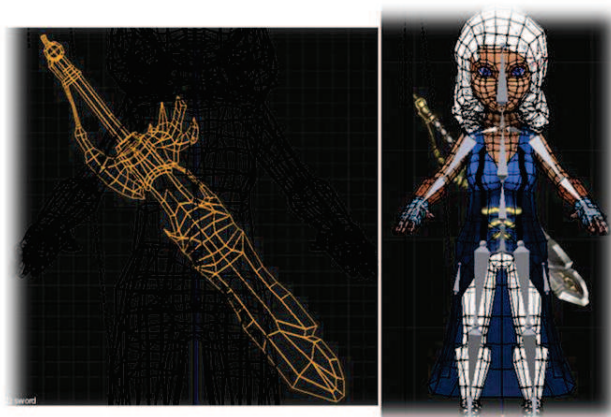


Figura 3.84 Estructura de huesos sobre varios objetos [8] [39]

3.3.1.2 Rigging

Este proceso asocia cada hueso de la estructura a una parte específica del modelo. Es necesario para darle movimiento en la animación. El rigging inicia seleccionando el modelo base, junto con la estructura de esqueleto, se asigna como padre de la estructura de huesos al modelo; dándole pesos automáticos como propiedad. Estos pesos definen a que porción de la malla del modelo afecta determinado hueso.



Figura 3.85 Asignación de pesos de la estructura de huesos sobre el modelo [8] [39]

Cuando concluye la asignación de pesos automática, el modelo es capaz de lograr movimiento, en una fase inicial.

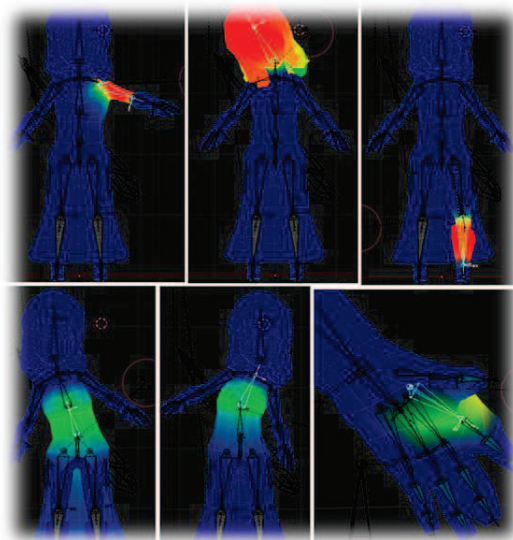


Figura 3.86 Ejemplo de Movimientos con pesos automáticos [8] [39]

Para perfeccionar los movimientos que pueda tener el modelo, se realiza otro proceso llamado weight painting. Este proceso modifica las regiones asignadas por los pesos automáticos.

3.3.2 PROCESO DE ANIMACIÓN

Para realizar la animación del modelo se debe tener en cuenta que los movimientos no sean bruscos o forzados, que todos los vértices del modelo estén afectados por los huesos, además añadir todos los objetos que se emplearán en la animación.

3.3.2.1 Weight Painting y Rediseño del Armature [58]

Weight Painting asigna pesos por pintura a las regiones que afectan los huesos. Colores más fuertes indican que el peso es mayor, por tanto el movimiento sobre esa región será más evidente que la de una región con un color más suave. Se lo realiza de manera paralela a la animación, de esta forma se corrigen errores mientras se anima el modelo. Para su realización es necesario seleccionar el hueso, verificar el área afectada y proceder a añadir o quitar vértices.

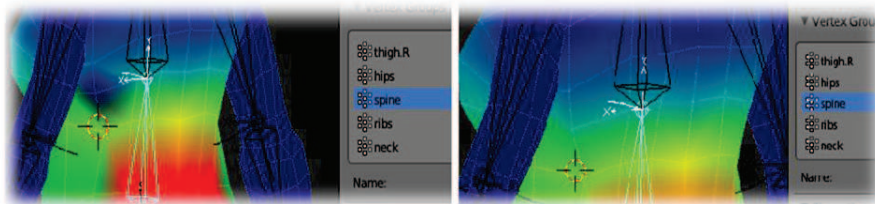


Figura 3.87 Ajuste de vértices con Weight Painting [8] [39]

El Rediseño del Armature se lo realiza para controlar el movimiento. Si se tiene distintas mallas en un mismo modelo, es difícil que todo sea controlado por los huesos de la estructura base; por este motivo se añade huesos auxiliares. En este caso los huesos auxiliares controlan la falda del modelo.

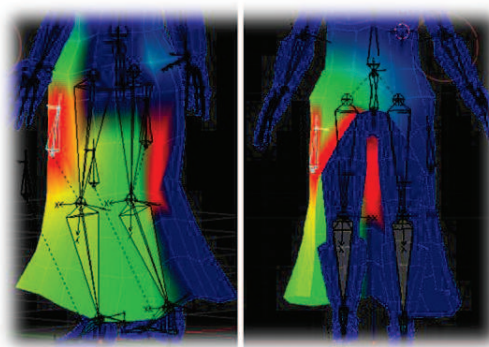


Figura 3.88 Huesos Auxiliares con Weight Painting [8] [39]

El rediseño además sirve para controlar el movimiento de articulaciones como la pierna y el tobillo.

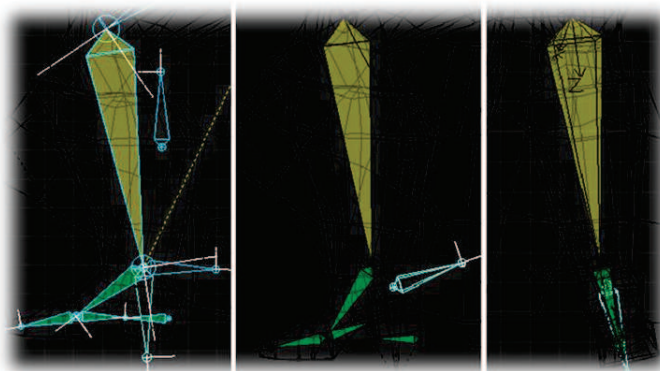


Figura 3.89 Diseño de Huesos para el control de articulaciones, tobillo, rodilla [1] [61]

3.3.2.2 Implementación de Constrains.

Los constrains añaden restricciones al movimiento de los huesos, se puede limitar el movimiento, trasladar el movimiento de un hueso a otro o establecer secuencias para que los huesos se muevan simultáneamente.

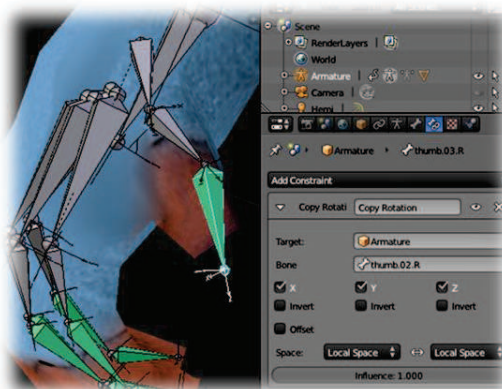


Figura 3.90 Constrains para control de movimiento de la mano [1] [61]

Es importante diseñar los constrains antes de realizar la animación. Los constrains establecen el movimiento no forzoso del objeto, se anclan a la estructura de huesos base, mejorando la naturalidad de los movimientos que el objeto pueda alcanzar. En el caso de este objeto se realizaron constrains para limitar el movimiento de los pies, se estableció un IK⁸² para las piernas y el movimiento de las rodillas.

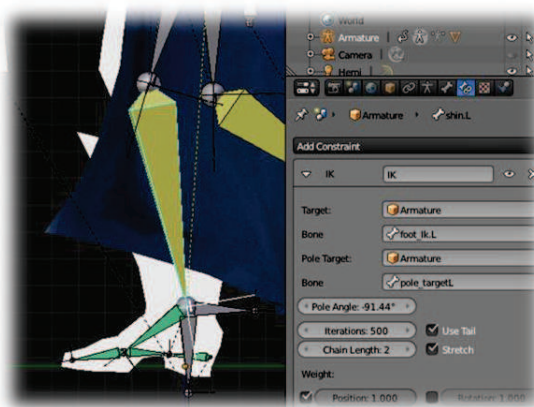


Figura 3.91 Uso de constrain IK para la rodilla [1] [61]

⁸² Inverse Kinematics, define ecuaciones para cadenas de movimiento, en el caso de la animación estas cadenas conectan a los huesos seleccionados.

Cuando se finaliza el proceso de constrains, se cuenta con toda la estructura necesaria para el proceso de animación

3.3.2.3 Ciclo de animación

La animación establece secuencias de movimientos de la estructura base de huesos anclados al objeto a través de una línea tiempo. De esta manera se crea el efecto de movimiento añadiendo frames con posiciones distintas.

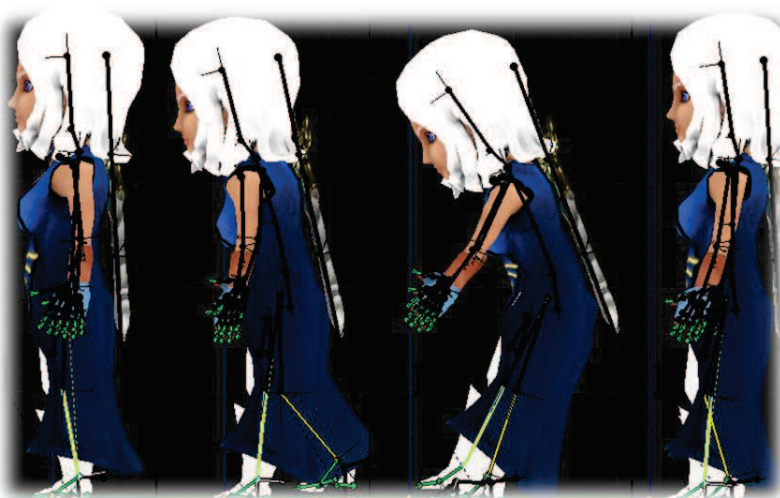


Figura 3.92 Secuencia de animación RecibirDaño [8] [39]

Cuando una animación debe repetirse indefinidamente en el videojuego como por ejemplo caminar o correr, se establece un ciclo secuencial con las posiciones del objeto.

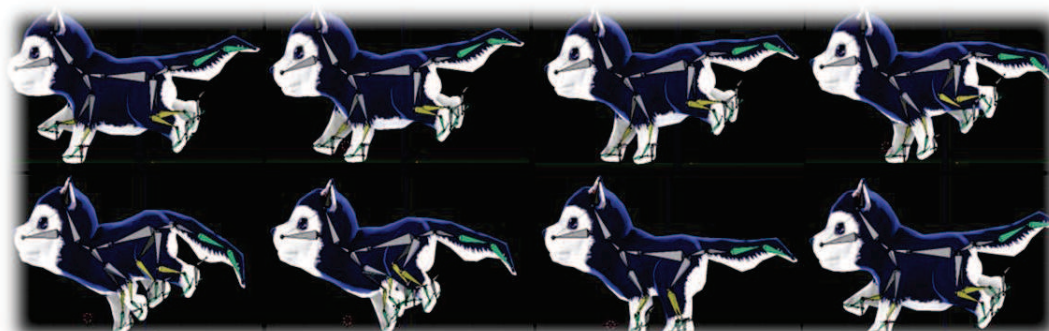


Figura 3.93 Animación Ciclo Secuencial [8] [39]

En este ciclo, es importante que la posición final sea exactamente igual a

la inicial, para mantener la secuencia de manera natural al repetirse el ciclo. Cada animación es almacenada junto con el objeto, al igual que las texturas. Una vez que el proceso de animación concluye se exporta el objeto que será usado en el Motor Gráfico.

3.4 PROGRAMACIÓN DEL JUEGO [59]

La programación se implementa a partir de tutoriales de Unity: 3d Platformer Tutorial [60], Penelope [61] y Unity3 Hotshot [62], cuyas licencias permiten modificarlo y usado por la persona que lo descargue. La programación de cada Game Object, asigna la lógica necesaria para que el objeto se maneje dentro del entorno del motor gráfico. A cada Game Object le corresponderá ciertas clases asignando de esta forma un "gameplay" funcional dentro de la aplicación.

A continuación se realiza una breve descripción de las clases implementadas. Si se desea ver a más detalle el código, se lo puede encontrar en el Anexo B.

3.4.1 GAMEOBJECT PERSONAJE

El GameObject Personaje permite el manejo del Objeto –personaje principal– dentro del videojuego mediante las entradas del Usuario. Las entradas se manejan de acuerdo al gameplay establecido, es decir deben manejar las colisiones con objetos, las animaciones corresponder a las acciones que se tomen, aumentar o disminuir la vida, determinar el número de items o enemigos vencidos.

3.4.1.1 Clase ThirdPersonController.

La clase ThirdPersonController permite manejar las acciones del personaje de acuerdo a las entradas de usuario o a su interacción con el entorno. Maneja la gravedad, verifica los collider, recibe mensajes de elementos del entorno como enemigos y se comunica a la Clase ThirdPersonAnimator.

3.4.1.2 Clase ThirdPersonAnimator.

La clase ThirdPersonAnimator se encarga de manejar las animaciones,

basándose en las entradas que recibe a través del usuario.

3.4.1.3 Clase ThirdPersonStatus.

La clase ThirdPersonStatus controla cuanta vida tiene el personaje, cuantos enemigos le faltan por vencer, cuantos items necesita recoger, es la encargada de comunicar al nivel, el lanzamiento de un evento, o la muerte del personaje.

3.4.1.4 Clase ThirdPersonAttack.

La clase ThirdPersonAttack.js permite al personaje hacer daño al enemigo. Si el usuario presiona la tecla de ataque se detectara a todos los enemigos que se encuentren en el radio de ataque. Si un enemigo se encuentra en el radio de ataque se envía un mensaje a la clase EnemyDamage.js con la cantidad de daño que infringe el personaje.

3.4.2 GAMEOBJECT CÁMARA.

"El Objeto cámara se encarga de permitir al usuario ver el mundo"⁸³. La cámara es un objeto por default dentro de Unity 3d ya que sin él no se podría apreciar los niveles en construcción, en este caso específico se extiende la funcionalidad de la cámara para adaptarla al gameplay.

3.4.2.1 Clase OrbitCamera.

La clase OrbitCamera camera permite controlar con el mouse la cámara que rota alrededor del target. La cámara sigue al personaje a una distancia y rota a una velocidad definida. Además maneja el ángulo máximo y mínimo con el eje.

3.4.2.2 Clase GameHuD.

La clase GameHud muestra el estado del personaje. Hace una llamada a la clase ThirdPersonStatus para obtener la cantidad de vida del personaje. La clase obtiene el tamaño de la pantalla, ubica las imágenes que representan la en un sector que se determine.

⁸³ Tomado de la definición de cámara en Unity Scripting Reference,
<http://docs.unity3d.com/Documentation/ScriptReference/Camera.html>

3.4.3 GAMEOBJECT ENEMIGO

El Game Object del enemigo representa al objeto que combatirá con el personaje, maneja inteligencia artificial simple mediante máquinas de estado finito.

3.4.3.1 Clase Enemy

La clase Enemy implementa dos tipos de enemigos diferenciados por el tipo de ataque. Realiza una llamada a la posición de un objetivo, determinando la distancia, persigue al objetivo con una animación y velocidad determinadas. Si el objetivo está en la distancia del rango del ataque, realiza la animación correspondiente a esta acción. El ataque dependerá del tipo de enemigo. En el caso del enemigo melee el ataque se efectúa directamente, envía un mensaje a la clase ThirdPersonStatus para establecer el daño sobre el objetivo. En el tipo de enemigo distancia instanciara un GameObject que permite impactar al objetivo, con la misma funcionalidad del enemigo melee establece el daño.

3.4.3.2 Clase EnemyDamage

La clase EnemyDamage se encarga de manejar el estado de la vida que tendrá un Enemigo, controla si el enemigo recibe daño, si es destruido; al ser destruido instanciara un objeto con la animación de su muerte.

3.4.3.3 Clase BossSamuro

La clase BossSamuro maneja determinadas acciones de un enemigo o Jefe. Hace una llamada al GameObjectPlayer para determinar su posición y lanzar un ataque. Realiza diferentes tipos de ataques los cuales se van intercambiando con el tiempo.

3.4.3.4 Clase BombaJS

La clase BombaJS se encarga de realizar un movimiento parabólico cuando se lanza un proyectil al Player, además controla que el objeto sea destruido en un tiempo determinado.

3.4.4 GAMEOBJECT PICKUP

Los objetos Pickup incrementan o disminuyen algún status del personaje. La clase hace uso de un collider para poder tocar al personaje y activar su funcionalidad. Se hace una llamada a la clase ThirdPersonStatus, dependiendo del tipo de pickup que sea, se usa una de las funciones de esa clase. Al usarse el objeto es destruido invocando un efecto.

3.4.5 GAMEOBJECT LEVELSTATUS

El GameObject Level Status instancia escenas de corte. Hace una llamada a través de la clase ThirdPersonStatus.js, obteniendo los objetos necesarios para ejecutar las acciones en el nivel.

3.4.6 GAMEOBJECT PAUSE MANAGER

El GameObject Pause Manager se encarga de pausar el juego. Detiene las acciones que sucedan y llama al menú de opciones. Para pausar el juego se usa las funciones de renderizado que se encuentran dentro del SceneManager, se cambia el tiempo de render a 0 para que no siga avanzando. La función OnGui() permite llamar al menú de Opciones

3.4.7 GAMEOBJECT MISCELANEOS

Los objetos misceláneos no incorporan clases, pero incluyen llamadas a distintas funciones dentro del juego. Un caso en concreto es GameObject ItemCollector el cual hace una llamada a la función AddItem() de la clase ThirdPersonStatus.js la cual aumenta el número de items en uno. Otro ejemplo representa las montañas e inclusive el piso los cuales incorporan funciones para que el personaje no resbale.

3.4.8 GAMEOBJECT DIALOGS

Los Game ObjectDialogs son los encargados de mostrar los diálogos en la escena. Cada GameObject llama a un conjunto de librerías que se encuentran disponibles dentro de la tienda de activos de Unity.

3.4.9 UNITYSERIALIZER

UnitySerializer es una librería de clases disponible en la AssetStore de Unity. Estas clases permite transformar los objetos en archivos XML, los cuales pueden ser almacenamos dentro de las preferencias de jugador o en un archivo externo.

3.4.10 SCENE MANAGER

El SceneManager es la instancia de la escena. Unity maneja librerías para la construcción de la escena que permite la construcción grafica de la misma. El Scene Manager se encarga de crear todos los objetos y comunicarlos mediante distintas librerías por eso todos los objetos se relacionan con él.

3.4.11 PRUEBAS

3.4.11.1 Pruebas de Desarrollo [63]

Las pruebas de desarrollo se realizan a medida que se construye la aplicación, se aplican para detectar errores en una etapa de desarrollo temprana y corregirlos de manera oportuna. Además al realizar estas pruebas se evalúa los requerimientos funcionales de la aplicación.

3.4.11.1.1 Pruebas de Unidad

Verifica la unidad más pequeña del desarrollo, es decir son de alcance restringido para descubrir errores en los límites de las clases generadas. Para las pruebas de unidad se construye una escena con un plano, en esta escena sólo se maneja una cámara. Se prueba los componentes que forman a los distintos GameObjects. En el caso del videojuego muchos componentes no se pueden probar de manera aislada, en la Tabla 3.25 se presenta un ejemplo de caso de prueba de unidad para la clase ThirdPersonController.js, la totalidad de estas pruebas están detalladas en el Anexo A.

Caso de prueba Único	ThirdPersonController.js
Propósito	Identificar errores en el movimiento en la clase ThirdPersonController.JS
Pre-requisitos	Construir la escena de Pruebas con un Plano y una Cámara
Pasos y Condiciones de Ejecución	<ol style="list-style-type: none"> 1. Ingresar el GameObjectPlayer en la escena de prueba. 2. Añadir un Componente Controller y un componente CapsuleCollider al Objeto3D 3. Añadir en Componente ThirdPersonController al GameObjectPersonaje 4. Ejecutar la Aplicación 5. Probar el movimiento con las teclas correspondientes: A,S,D,W, SpaceBar
Resultado Esperado	<p>Las teclas A realiza un movimiento hacia la izquierda La tecla S realiza el movimiento hacia atrás La Tecla D realiza el movimiento hacia la derecha La tecla W realiza el movimiento hacia adelante La tecla Space Bar hace saltar al personaje</p>
Resultado Obtenido	Las teclas tienen la funcionalidad correcta
Notas	Ninguna
Responsable Diseño	Andrés Cevallos
Responsable Ejecución	Andrés Cevallos

Tabla 3.25 Ejemplo Prueba de Unidad ThirdPersonController [8]

Resumen de Resultados de pruebas de unidad:

A continuación se presenta un resumen de los resultados sobre la ejecución de las pruebas en el transcurso del desarrollo.

Resumen de errores:

- Errores en las animaciones GameObject Personaje.
- Errores en la posición de la cámara
- Errores de animaciones en los GameObjectEnemy.
- Errores en el script CharacterController, el personaje salta demasiado alto y atraviesa las paredes.

Corrección de errores:

- Se corrigió las animaciones del personaje y los enemigos. El error se debe a que los objetos tienen varias animaciones y al no guardar una posición inicial, se superpone la posición de la última animación ejecutada. Se guardó una posición inicial por cada animación.
- Se corrigió la posición de la cámara.
- Se corrigió el script CharacterController.

3.4.11.1.2 *Pruebas de Integración*

La prueba de integración toma componentes a los cuales se les aplicó una prueba de unidad. Se las realiza mientras se construye el sistema, todos los componentes deben integrarse a un GameObject de una forma correcta. Puede producirse en dos formas, una integración incremental ascendente o una descendente. Se usa la integración ascendente empezando con módulos de bajo nivel, se prueban en grupo y ascienden a medida que se realizan las pruebas. Tanto para las pruebas de integración como para las de regresión se establece un diagrama con los módulos del sistema. En el caso del videojuego las clases que lo conforman. El diagrama se presenta en la Figura 3.94.

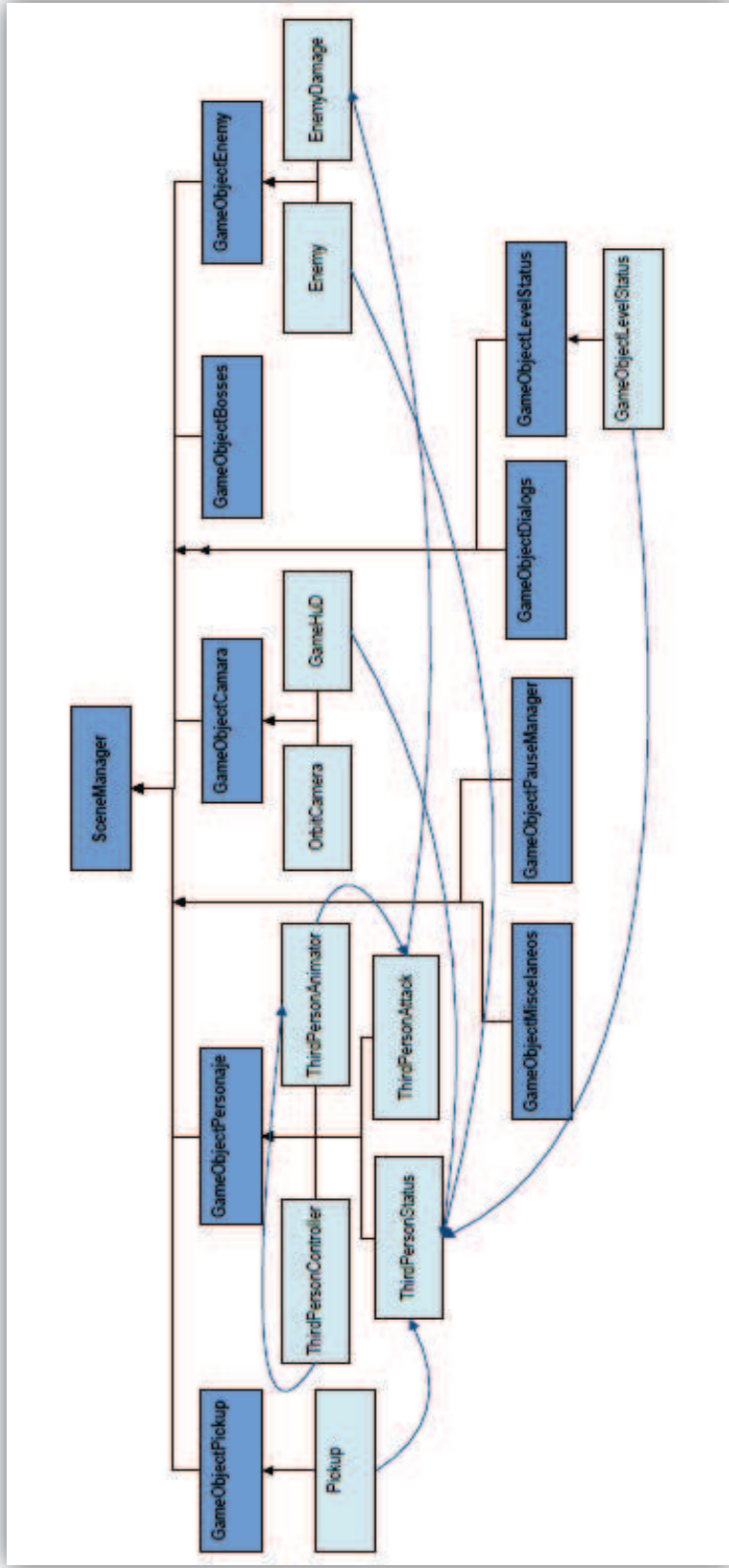


Figura 3.94 Esquema de Módulos del Videojuego. Fuente: Microsoft Office Visio [8]

El Figura 3.94 representa la comunicación que existe entre cada una de las clases. En la Tabla 64 se muestra un ejemplo de prueba de integración de la clase Enemy.js. Las pruebas se detallan en anexos.

Caso de prueba Único	Enemy.js
Propósito	Probar la Distancia de persecución y ataque de los GameObjectEnemy
Pre-requisitos	Tener la escena de prueba con el GameObjectPersonaje y el GameObjectEnemy
Pasos y Condiciones de Ejecución	<ol style="list-style-type: none"> 1. Ejecutar la aplicación 2. Activar los Gizmos del GameObjectEnemy 3. Entrar en la Distancia de Persecución 4. Entrar en la distancia de Ataque
Resultado Esperado	El enemigo persigue al objetivo cuando entra a la distancia de persecución. El enemigo ataca cuando entra en la distancia de ataque
Resultado Obtenido	El resultado es el esperado
Notas	Se debe incrementar la distancia de persecución
Responsable Diseño	Andrés Cevallos
Responsable Ejecución	Andrés Cevallos

Tabla 3.26 Ejemplo Prueba de Integración Enemy.js [8]

Resumen de Resultados de pruebas de integración:

A continuación se presenta un resumen de los resultados sobre la ejecución de las pruebas en el transcurso del desarrollo. Las pruebas de integración se realizan uniendo componentes a un nivel o escena.

Resumen de errores:

- Errores en la aplicación de daño del componente Enemy.js al componente ThirdPersonStatus.js.
- Errores en las llamadas a GameObjectDialogos por parte del componente DialogTrigger.js.
- Errores en la ejecución de componentes.
- Errores en las llamadas a Menús por parte del componente.
- Errores en el almacenamiento de archivos por parte del GameObjectSave Manager.

Corrección de errores:

- Se corrigió el daño y las llamadas a los menús.

3.4.11.1.3 *Pruebas de Regresión*

Cada vez que se agrega un nuevo módulo se crean nuevas salidas y entradas se establecen nuevas rutas de flujos de datos, esto puede causar problemas. Las pruebas de regresión se realizan para verificar que los nuevos componentes no afecten los previos. Se aplican en el mismo sentido que las de integración, pero es aplicada a todos los módulos.

4. CAPÍTULO 4: FASE DE POSTPRODUCCIÓN

4.1 DISEÑO DE CRITERIOS DE CALIDAD Y EVALUACIÓN

La evaluación del software a través de criterios de calidad, proporciona un mecanismo de medición. Gracias a la medición se puede cuantificar el producto desarrollado y mejorarlo de manera continua en un ciclo de desarrollo. Por esta razón la creación de métricas de evaluación es importante, para tener una idea de cómo se ha sido desarrollado el producto y de qué manera mejorarlo [63].

4.1.1 TERCER ENTREGABLE R3 - TERCER SPRINT CONTINUACIÓN

En este capítulo se retoman las tareas definidas en el tercer sprint del entregable R3, que no son parte del tercer capítulo. En este sprint se definen las tareas para la realización de pruebas alfa y beta.



Figura 4.1 R3-Tercer Sprint Tareas [8] [48]

4.1.1.1 Detalle de las tareas Entregable R3 - Tercer Sprint

A continuación se detalla las tareas nuevas que son parte de R3- Tercer Sprint, las que se repiten no se las detalla.

Nombre : Pruebas Alfa

Tipo : Historia de usuario

Clasificación : 8

Descripción : Se realiza pruebas con internas con el equipo de desarrollo, cuando el producto ha sido terminado.

Notas :

Fecha de envío : 30/08/2013 19:10:10

Fecha de aceptación : 30/08/2013 19:10:26

Fecha de estimación : 30/08/2013 19:10:37

Fecha de planificación : 30/08/2013 19:10:49

En progreso desde : 30/08/2013 19:10:49

Fecha de terminación : 28/10/2013 19:37:27

Creador : Marcelo Agila

Característica : Pruebas

77

Pruebas Alfa

8 Terminada

Figura 4.2 Tarea Pruebas Alfa [8] [48]

<i>Tarea</i>	<i>Subtarea</i>	<i>Estado</i>	<i>Descripción</i>	<i>Responsable</i>	<i>Inicio</i>	<i>Fin</i>
Pruebas Alfa	Pruebas de Resistencia	Terminada	Se debe ejecutar el videojuego con una cantidad inusual de recursos.	Andrés Cevallos	29/10/2013	29/10/2013
	Pruebas de Desempeño	Terminada	Se debe medir los recursos utilizados. Se realiza junto con la prueba de resistencia	Andrés Cevallos	29/10/2013	29/10/2013
	Pruebas de Interfaz Gráfica	Terminada	Se realiza todas las pruebas relacionadas a la interfaz, objetos, posición colisiones, menús, etc.	Andrés Cevallos	29/10/2013	29/10/2013
	Pruebas en Tiempo Rea	Terminada	Se realizan los tres tipos de pruebas en conjunto; Tareas: Evalúan la lógica del funcionamiento del videojuego; Comportamiento: Evalúan el sistema en general, las acciones ejecutadas, etc.; Sistema: evalúan las interrupciones y eventos.	Andrés Cevallos	29/10/2013	29/10/2013

Tabla 4.1 Subtareas Pruebas Alfa [8] [48]

Nombre : Pruebas Beta

Tipo : Historia de usuario

Clasificación : 9

Descripción : Se diseña criterios de calidad de evaluación multimedia y para videojuegos y se elabora una encuesta para los usuarios finales.

Notas :

Fecha de envío : 30/08/2013 19:21:25

Fecha de aceptación : 30/08/2013 19:22:04

Fecha de estimación : 30/08/2013 19:22:20

Fecha de planificación : 30/08/2013 19:22:34

En progreso desde : 30/08/2013 19:22:34

Fecha de terminación : 31/10/2013 19:38:00

Creador : Marcelo Agila

Característica : Pruebas

78 Pruebas Beta
3 Terminada

Figura 4.3 Tarea Pruebas Beta [8] [48]

Tarea	Subtarea	Estado	Descripción	Responsable	Inicio	Fin
Pruebas Beta	Definición de Criterios Multimedia	Terminada	Se define los criterios multimedia relacionados con los requerimientos funcionales	Marcelo Agila	01/11/2013	01/11/2013
	Definición de Criterios Videojuegos	Terminada	Se define criterios afines a los videojuegos los más generales y se los relaciona con los requerimientos funcionales	Marcelo Agila	01/11/2013	01/11/2013
	Creación de Encuesta	Terminada	Se crea una encuesta para los usuarios finales, tomando cada criterio definido, multimedia y de videojuegos.	Marcelo Agila	01/11/2013	01/11/2013

Tabla 4.2 Subtareas Pruebas Beta [8] [48]

Una vez que se ha definido y detallado las tareas restantes del entregable R3 tercer sprint, se procede con la documentación del trabajo realizado en cada una de las tareas en el presente documento.

4.1.2 EVALUACIÓN

4.1.2.1 Pruebas del Sistema [63]

Las pruebas del sistema se realizan sobre la aplicación y se dividen en dos grupos: Pruebas Alfa y Pruebas Beta.

4.1.2.1.1 Pruebas Alfa

Las pruebas alfa son internas, se las realiza con el equipo de desarrollo y se dividen en las siguientes categorías:

- **Resistencia**

Ejecuta programa de tal manera que requiera una frecuencia o cantidad anormal de recursos. En la Tabla 64 se muestra las pruebas de resistencia para los niveles de juego.

Caso de Prueba	Prueba de Resistencia
Descripción de la prueba	Prueba de Resistencia en los Niveles del juego
Ambiente de Prueba	Laptop Samsung core i7 6GB de RAM
Pasos a Seguirse	Correr la aplicación en los sectores que más triángulos contengan dentro de la aplicación y medir los frames por segundos obtenidos
Resultados Obtenidos	Escena MainMenu triángulos máximos 150mil, frames x segundo 65 Escena Begginig triángulos máximos 200mil, frames x segundo 69 Escena VillageLevel triángulos máximos 150mil, frames x segundo 48~20
Notas	Es necesario reducir el número de triángulos en el nivel VillageLevel o cambiar los terrain collider por objetos 3d
Responsable de Diseño	Andrés Cevallos
Responsable de Ejecución	Andrés Cevallos

Tabla 4.3 Pruebas de Resistencia Niveles Videojuego [8]

Resumen de Resultados de pruebas de resistencia:

A continuación se presenta un resumen de los resultados sobre la ejecución de la prueba.

Resumen de errores:

- En ciertos lugares del juego los frames por segundo disminuyen drásticamente.

Corrección de errores:

- Se optimizó los triángulos y se modificó los terrain collider por objetos 3D.

- Desempeño

Prueba el desempeño del software en tiempo de ejecución, se vinculan a las pruebas de resistencia, miden exactamente el uso de recursos. En la Tabla 4.4 se muestra las pruebas de desempeño.

Caso de Prueba	Prueba de Desempeño
Descripción de la prueba	Prueba de Desempeño en los Niveles del juego
Ambiente de Prueba	Laptop HP dv6000 procesador AMD Thuron 1.9GHz 3GB de RAM Windows Vista
Pasos a Seguirse	Correr la aplicación en el momento de carga y ver cuanta memoria y procesador consume. Correr la aplicación en el momento de carga de niveles y observar la memoria consumida. Correr la aplicación en el momento de más triángulos y observar los resultados. Detectar zonas en que el framing decaiga.
Resultados Obtenidos	En el momento de carga la aplicación consume 30000 KB de memoria RAM. En el momento de carga de niveles la aplicación consume el 80% de procesador, y 170000KB de memoria. En zonas de altos triángulos se tiene un consumo de 60% del procesador y, y 150000KB de memoria.
Notas	Es necesario reducir el número de triángulos en el nivel VillageLevel o cambiar los terrain collider por objetos 3d
Responsable de Diseño	Andrés Cevallos
Responsable de Ejecución	Andrés Cevallos

Tabla 4.4 Pruebas de Desempeño Niveles Videojuego [8]

Resumen de Resultados de pruebas de desempeño:

A continuación se presenta un resumen de los resultados sobre la ejecución de la prueba.

Conclusiones:

- El juego puede ser aplicado para computadores con un procesador doble núcleo de 1.9Ghz sin ningún inconveniente.
- Se estima además que para jugar sin complicaciones es necesario 1GB de memoria RAM

- Interfaz Gráfica

Se debe probar todos los objetos que son relevantes para la GUI. En la Tabla 4.5 se muestra un ejemplo de prueba de interfaz de la clase Menu.js.

Caso de prueba Único	Menus.js
Propósito	Comprobar que el tipo de letra usado para los menús sea el mismo en todos los escenarios
Pre-requisitos	Se deben tener los menús construidos
Pasos y Condiciones de Ejecución	<p>Iniciar la aplicación</p> <p>Revisar el menú Principal</p> <p>Revisar el menú de Opciones</p> <p>Revisar el menú de Opciones dentro del juego</p>
Resultado Esperado	Los menús manejan el mismo tipo de letra en todas las interfaces
Resultado Obtenido	El menú de Opciones dentro del juego no tiene el mismo tipo de letra
Notas	El menú de Opciones dentro del juego no tiene el mismo tipo de letra
Responsable Diseño	Andrés Cevallos
Responsable Ejecución	Andrés Cevallos

Tabla 4.5 Prueba de Interfaz Menu.js [8]

Resumen de Resultados de pruebas de Interfaz:

A continuación se presenta un resumen de los resultados sobre la ejecución de la prueba.

Resumen de errores:

- La letra de los menús no se visualiza de manera correcta.
- La posición de la cámara está muy alejada del personaje.
- No existe un botón para salir del juego.

Corrección de Errores:

- Se modificó el tipo de letra de los menús Opciones dentro de juego.
- Se corrigió y ajusto la posición de la cámara.
- Se debe añadir un botón de salida del juego dentro del menú de opciones principales.

4.1.3 CRITERIOS DE CALIDAD

Los criterios de calidad deben estar basados principalmente en el cumplimiento de los requerimientos funcionales del sistema. Estos requerimientos son la base para establecer una cuantificación de calidad. Se han dividido los criterios en: criterios multimedia y criterios de videojuegos.

4.1.3.1 Criterios de calidad videojuegos

Se han seleccionado los siguientes criterios de calidad generales, en base a la experiencia como usuarios de videojuegos.

4.1.3.1.1 Gráficas

Las gráficas comprenden la calidad de las animaciones, modelados, los escenarios, efectos especiales, iluminación del terreno, fluidos como agua, fuego, lava, entre otros. Este criterio de calidad cumple con los requerimientos funcionales Re5 y Re8 (ver Tabla 2.1).

4.1.3.1.2 Jugabilidad

Se relaciona con el gameplay del videojuego, comprende el control del personaje principal, debe ser intuitivo. Los ángulos de la cámara deben enfocar al personaje y guardar una buena perspectiva junto con su movimiento, entre otros. Cumple con los requerimientos funcionales Re1, Re2, Re3 y Re4 (ver Tabla 2.1).

4.1.3.1.3 Efectos y Sonido

El sonido ambiental debe estar relacionado con la temática del videojuego. Los efectos acompañan a las gráficas, efectos como relámpagos, golpes le dan mucha más profundidad a la acción. Cumple con los requerimientos funcionales Re3 y Re5 (ver Tabla 2.1).

4.1.3.2 Criterios de calidad Multimedia [64]

4.1.3.2.1 Simplicidad

La interfaz gráfica debe ser amigable con el usuario, sencilla e intuitiva. Los elementos como letras botones, etc., guardar relación con la interfaz general del videojuego. Cumple con los requerimientos funcionales Re1, Re2 y Re7 (ver Tabla 2.1).

4.1.3.2.2 Coherencia

Las interfaces deben corresponder a las acciones realizadas, por ejemplo acceder a un menú. La secuencia de las acciones debe ser la correcta a medida que el jugador interactúa con el videojuego. Cumple con los requerimientos Re6 y Re7 (ver Tabla 2.1).

4.1.3.2.3 Claridad

Los colores de los menús deben ser atractivos para el usuario, las fuentes claras con fondos que faciliten la lectura y evitar la sobrecarga de colores. Cumple con los requerimientos Re5 y Re7 (ver Tabla 2.1).

4.1.3.2.4 Adaptabilidad

El diseño debe ser simple para que usuarios con diferentes habilidades, no tengan problemas desenvolviéndose dentro del videojuego. La portabilidad del videojuego es esencial, se debe adaptar a las características específicas de los equipos en los que sea ejecutado. Cumple con los requerimientos Re1, Re2, Re5 y Re7 (ver Tabla 2.1).

4.1.3.2.5 Pruebas Beta

Para el diseño de estas pruebas se deben considerar los criterios de calidad que guardan relación con los requerimientos funcionales.

- *Diseño pruebas de Jugabilidad*

Para diseñar las pruebas de jugabilidad se ha tomado en cuenta los criterios de calidad multimedia y de videojuegos, es así como se ha asignado pesos a cada criterio de calidad, sobre un total de 100, la división de los criterios de calidad y los pesos se los presenta en la Tabla 4.6.

<i>Criterios de Calidad</i>		<i>Pesos</i>
<i>Contenido Multimedia</i>	<i>Simplicidad</i>	15
	<i>Coherencia</i>	10
	<i>Claridad</i>	10
	<i>Adaptabilidad</i>	10
<i>Videojuego</i>	<i>Gráficas</i>	20
	<i>Jugabilidad</i>	20
	<i>Efectos y Sonido</i>	15
	Total	100

Tabla 4.6 Pesos de Criterios de Calidad [8]

La encuesta se la ha diseñado siguiendo las definiciones de cada criterio de calidad, se realizó preguntas acerca de cada criterio. La encuesta realizada es de tipo Likert⁸⁴ impar con 5 puntos: Muy Mala, Mala, Regular, Buena, Muy Buena. El detalle del diseño de la encuesta presentada a los usuarios a través de la herramienta de formularios de Google Docs se presenta en la Figura 4.4

⁸⁴ Escala denominada así por Rensis Likert , comúnmente utilizada en cuestionarios y encuestas de investigación.

ENCUESTA					
Objetivo: Recabar información de la jugabilidad del videojuego Azrael Awakening obteniendo datos que permitan cuantificar el cumplimiento de los requerimientos funcionales y algunos elementos en el diseño multimedia.					
Indicaciones:					
1. Marque la respuesta que considere apropiada					
2. Debe marcar una sola respuesta					
3. Existen cinco opciones para calificar:					
1	2	3	4	5	
Muy Mala	Mala	Regular	Buena	Muy Buena	
					1 2 3 4 5
SIMPLICIDAD					
1. La facilidad para ubicar al personaje dentro del screen es?					
2. El movimiento de la cámara a una perspectiva deseada es?					
3. La usabilidad de los menús es?					
4. La facilidad para usar la interfaz es?					
COHERENCIA					
5. La llamada a acciones a través de los menus es?					
6. La secuencia del videojuego es?					
7. La comprensión del texto de la historia del videojuego es?					
8. La correspondencia de eventos respecto a las acciones y lo desplegado en pantalla es?					
CLARIDAD					
9. La facilidad para leer los textos durante el juego es?					
10. La facilidad para distinguir los objetos uno de otro es?					
11. La iluminación del escenario es?					
12. La mezcla de colores de la interfaz es?					
ADAPTABILIDAD					
13. Si se altera la resolución de la pantalla la calidad del videojuego es?					
14. La no exigencia de conocimientos previos es?					
GRÁFICAS					
15. La calidad de los objetos, texturas, animaciones es?					
16. La calidad de los efectos, iluminación es?					
17. La fluidez en el desarrollo del videojuego es?					
18. La calidad de escenarios es?					
JUGABILIDAD					
19. La facilidad de aprendizaje de los controles del personaje es?					
20. La facilidad de aprendizaje de la mecánica del videojuego es?					
21. El grado de diversión al jugar el videojuego es?					
22. La dificultad del videojuego es?					
EFFECTOS Y SONIDO					
23. La calidad de los sonidos incorporados es?					
24. La emotividad que denotan los sonidos en las acciones es?					
25. La correspondencia de sonidos con las acciones es?					
26. La distinción de sonidos(efectos, voces, ambiente) es?					

Figura 4.4 Diseño de Encuesta [8]

4.1.4 FINALIZACIÓN DEL ENTREGABLE R3 RESULTADOS DE LA METODOLOGÍA

Una vez que se ha cerrado con el tercer sprint del entregable R3, se puede obtener las gráficas que corresponden al desarrollo de la metodología empleada sobre el videojuego.




▼▲	Clasificación: ▼▲ Valor ▼▲	Tipo ▼▲	Característica ▼▲	Descripción ▼▲	Esfuerzo ▼▲	Historias asociadas ▼▲	Historias terminadas
	1 0	Funcional	Diseño de Interfaces	Las interfaces son muy importantes, deben ser muy intuitivas para que el usuario se sienta cómodo, mucho más si se trata de un videojuego. Proporcionan el entorno visual con el cual el usuario se comunica con nuestra aplicación; en este caso un videojuego.	32	9	9
	2 0	Funcional	Arte Conceptual	Es el conjunto de características que definen el aspecto del videojuego.	19	7	7
	3 0	Funcional	Construcción de personajes	Todo lo relacionado con la construcción del personaje desde los bocetos hasta el objeto 3D final.	227	18	18
	4 0	Funcional	Lógica del Videojuego	Define todas las acciones que puede desarrollar el personaje y los NPCs en el videojuego, es decir todas las acciones dependan o no del jugador.	95	9	9
	5 0	Funcional	Construcción de Ambientes	La creación de ambientes incluyen a todos los objetos necesarios para crear un nivel, es decir los objetos que intervienen para la creación del terrain, la cámara, luces, etc.	157	9	9
	6 0	Funcional	Pruebas	Todas las pruebas necesarias para la depuración del videojuego en la etapa de desarrollo	263	14	14

Figura 4.5 Características Videojuego Esfuerzo en Puntos estimados [8] [48]

Leyenda:

- (1) Los colores definen las características, las historias asociadas corresponden al número de tareas que empleo determinada característica.
- (2) El esfuerzo empleado se mide en días.

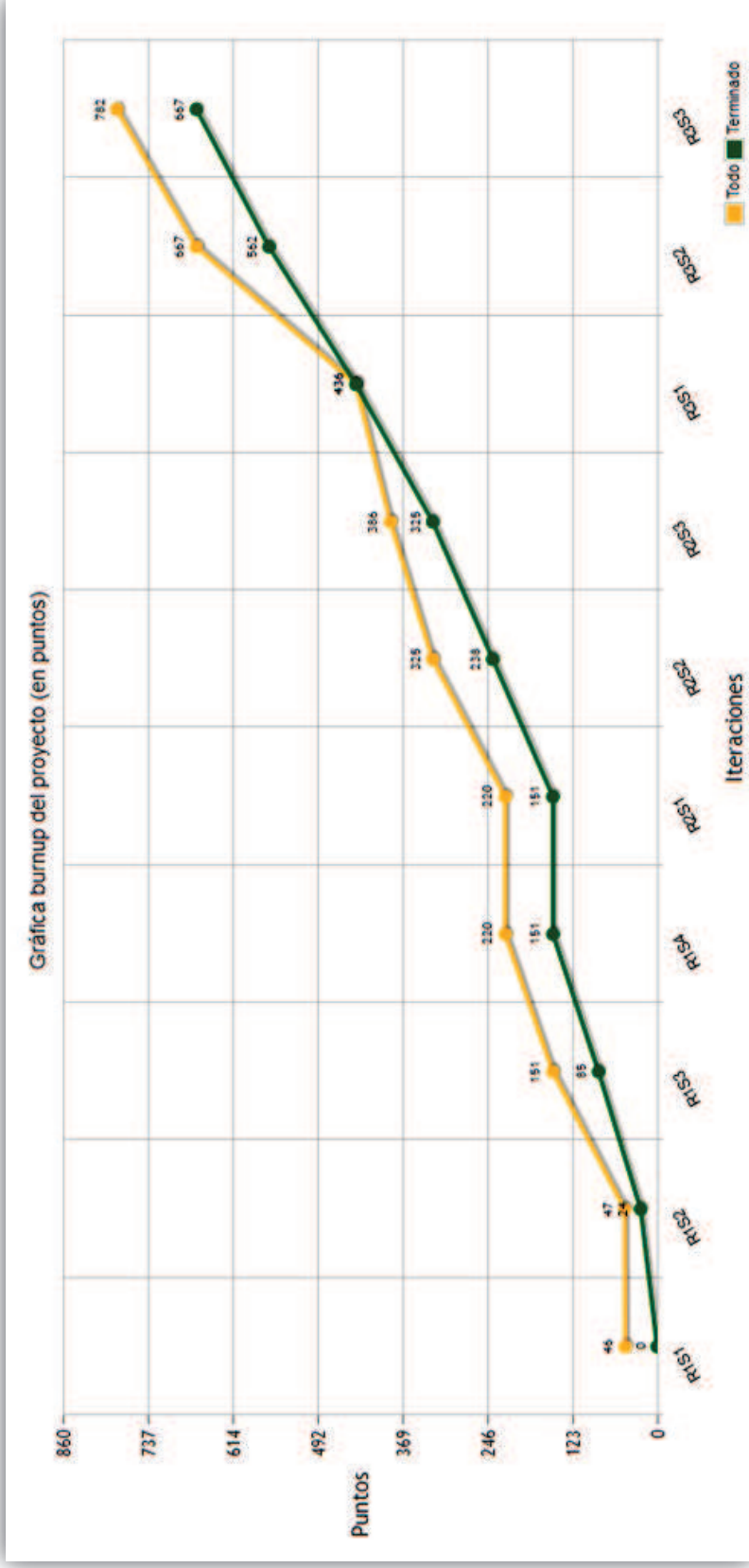


Figura 4.6 Diagrama BurnUp del Videojuego en Puntos. Puntos Vs Iteraciones [8] [48]

Leyenda:

- (1) La línea amarilla representa el total de las tareas. Los puntos corresponden al esfuerzo
- (2) La línea verde representa las tareas completadas. Los puntos corresponden al esfuerzo

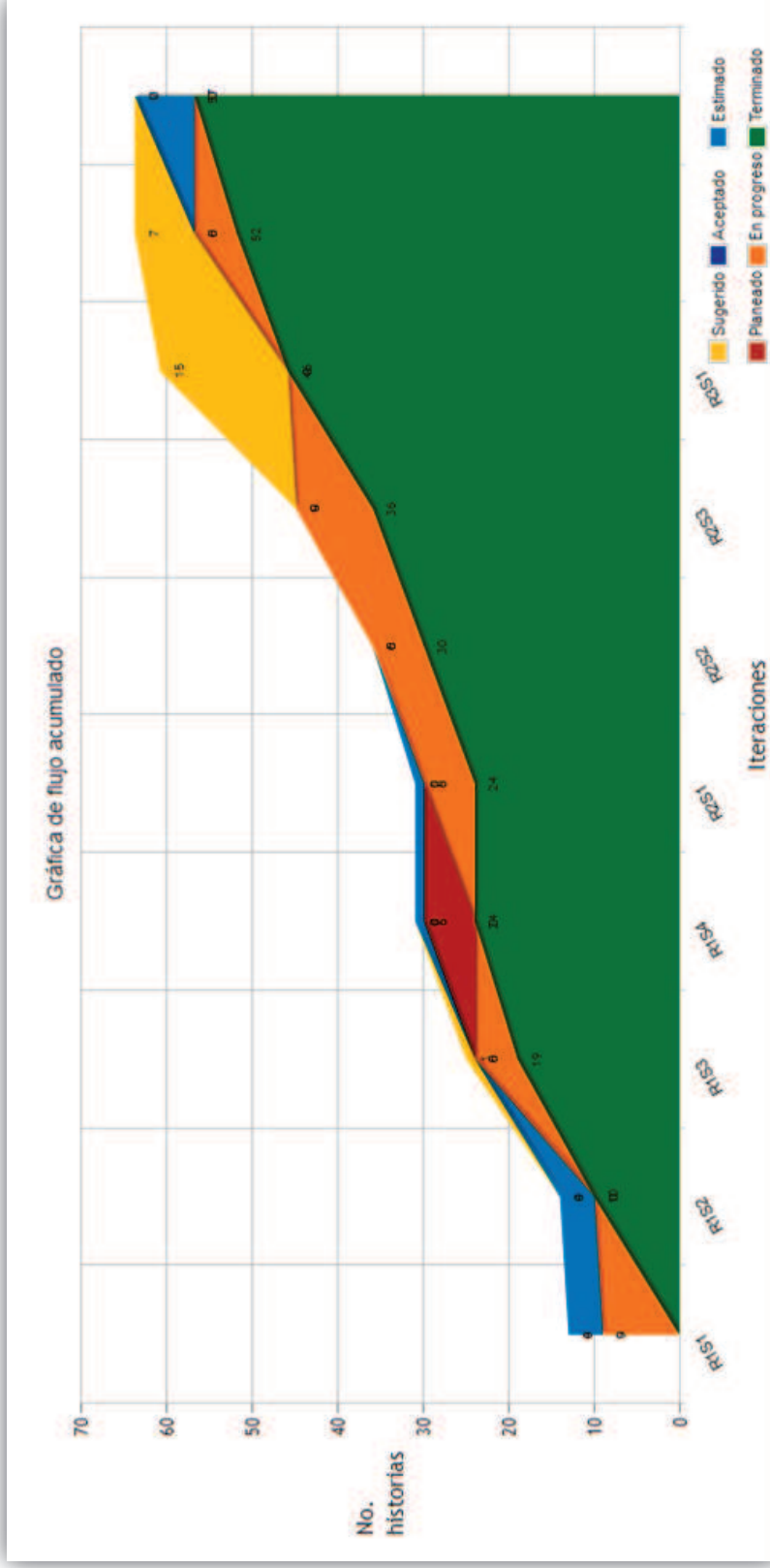


Figura 4.7 Flujo Acumulado Videojuego Iteraciones vs Nro. de Historias [8] [48]

Leyenda:

- (1) Color amarillo, historias de usuario sugeridas
- (2) Color Azul, Historias estimadas
- (3) Color celeste, Historias estimadas en tiempo
- (4) Color Rojo, Historias planeadas
- (5) Color Tomate, Historias en progreso
- (6) Color Verde, Historias terminadas

4.2 DETALLE DE DISTRIBUCIÓN DEL VIDEOJUEGO

Para la distribución del videojuego se usó internet, se procedió a crear enlaces a archivos alojados en servidores como DropBox. La encuesta fue hecha en Google Docs con la ayuda de la herramienta para creación de formularios.

4.2.1 BLOG VIDEOJUEGO

El blog de videojuego fue creado en e-blogger definiendo lo fundamental acerca de la construcción de personajes y desarrollo del videojuego. El enlace para acceder al blog es el siguiente: azr43lawakening.blogspot.com. En el blog se detalla cómo obtener el videojuego.



Figura 4.8 Captura Blog Videojuego [8]

4.2.2 DISTRIBUCIÓN

El videojuego se distribuyó a través de enlaces en las redes sociales, amigos y familiares realizaron las encuestas online.

El enlace para la realización de la encuesta es el siguiente: <https://docs.google.com/forms/d/1vlygYR9Chyc7B09vbf8S49NKYnFN44gzs6EZCCpCMDw/viewform>, una vez que el usuario finaliza la encuesta, puede visualizar los resultados acumulados. Las respuestas con los datos totales se presentan a manera de gráfico.

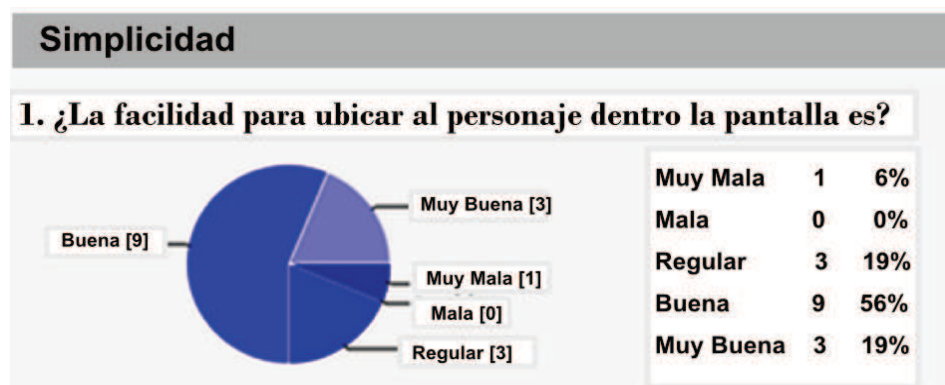
4.3 RECOLECCIÓN DE INFORMACIÓN DE ENCUESTAS DE JUGABILIDAD

El resultado de las encuestas es procesado con la herramienta de Google Docs para el procesamiento de formularios. La encuesta es un formulario, creado con preguntas relevantes a los criterios de calidad, detalladas en el diseño de encuestas; de esta manera se puede estimar una medida aproximada de la calidad del producto.

Gracias a las opciones de configuración de los formularios, se estableció preguntas con únicas respuestas, definidas en un rango del 1 al 5.

Los resultados obtenidos al día 08/11/2013, se presentan a manera de Figuras en su totalidad. Los criterios evaluados son los siguientes:

Multimedia



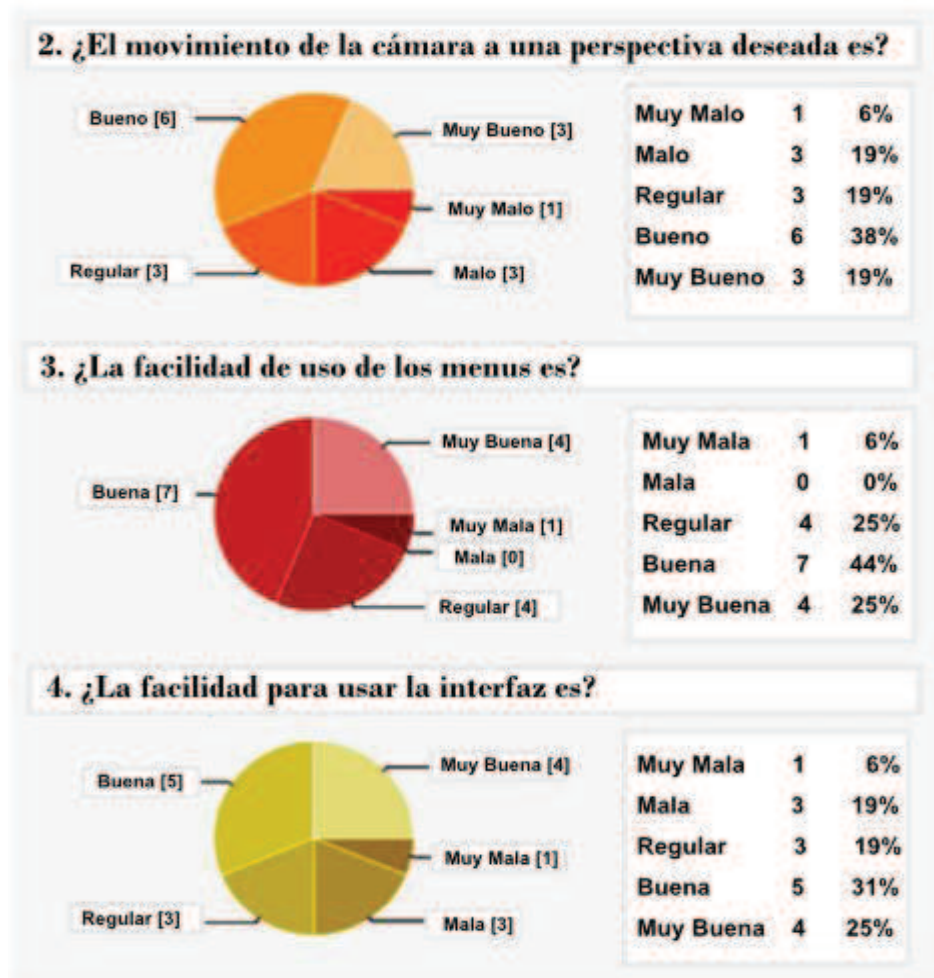
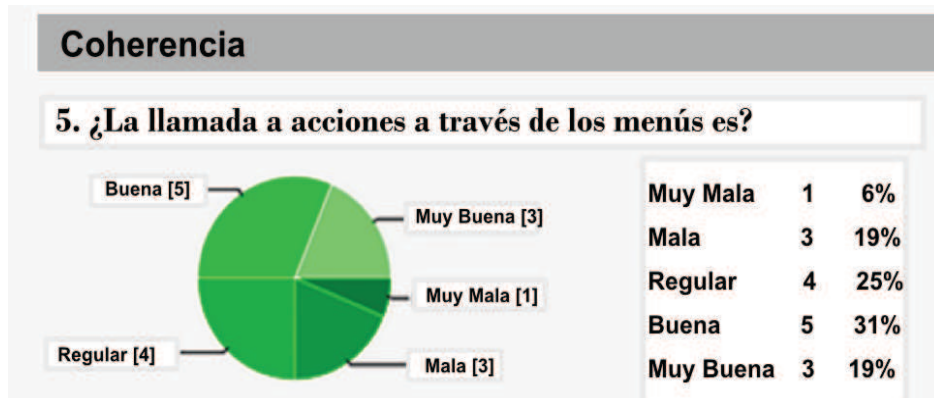


Figura 4.9 Resultados Preguntas Simplicidad. Fuente: Google Docs



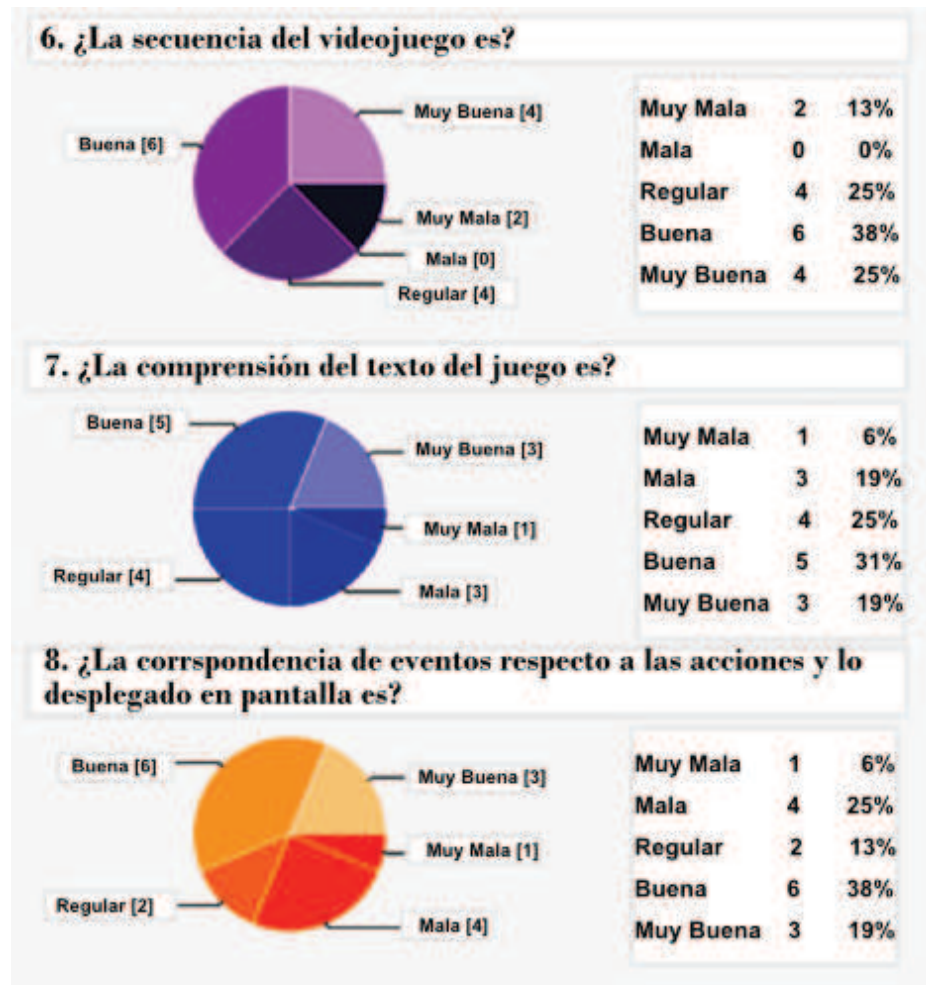


Figura 4.10 Resultados Preguntas Coherencia. Fuente: Google Docs

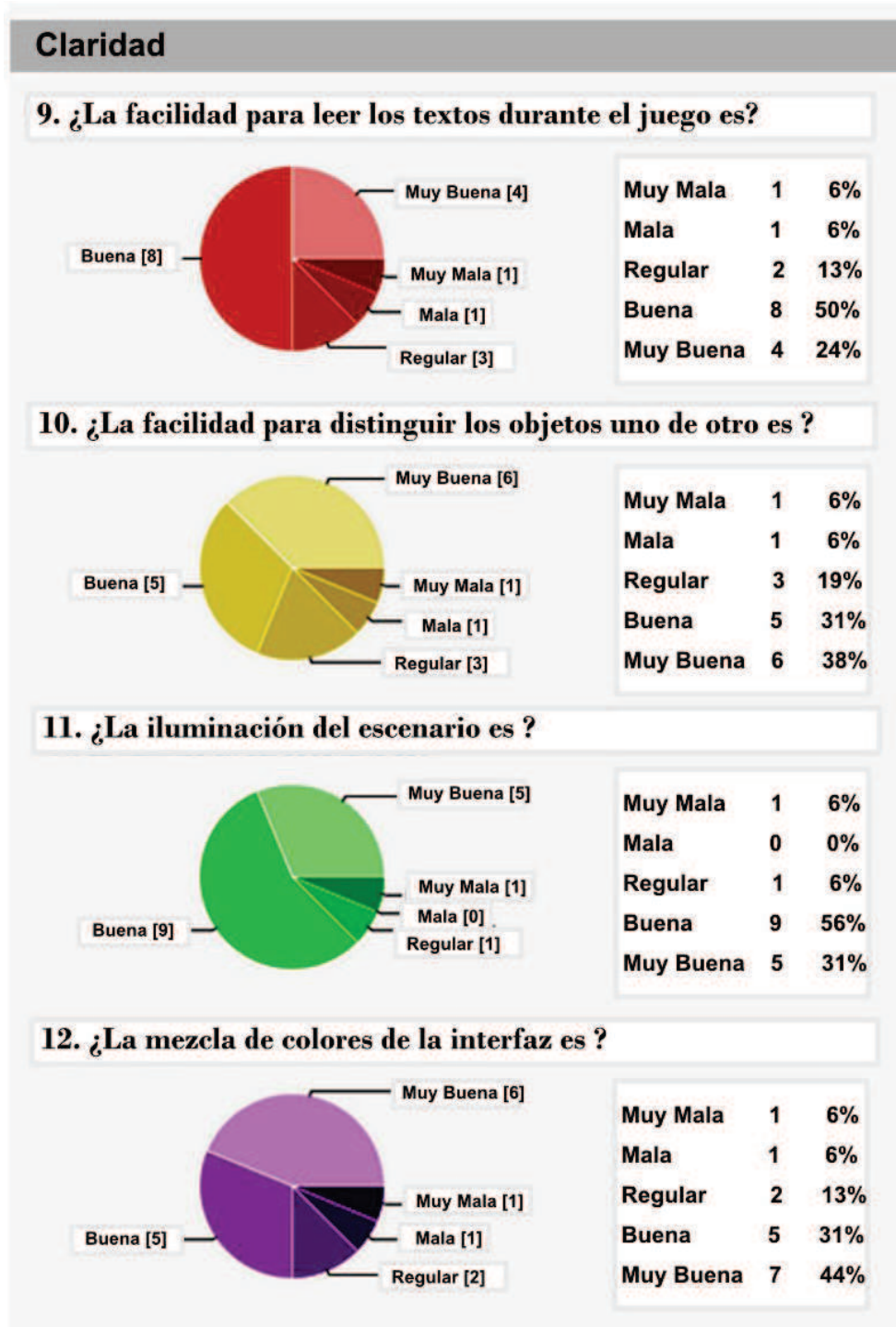


Figura 4.11 Resultados Preguntas Claridad. Fuente: Google Docs

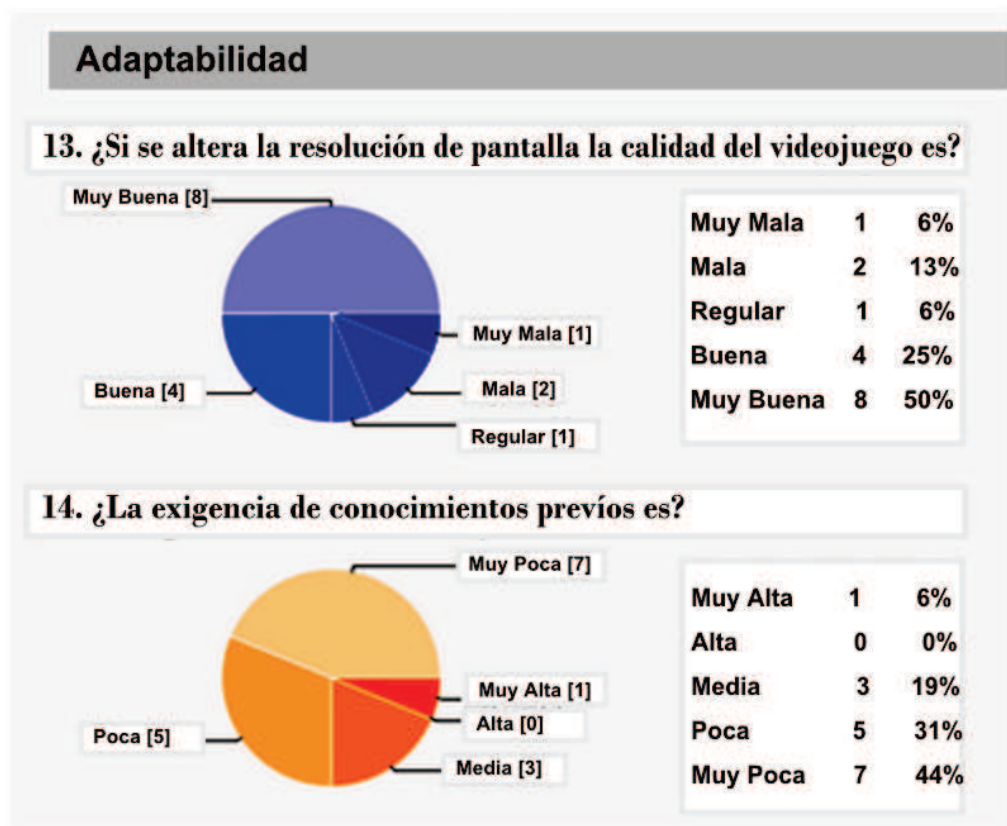
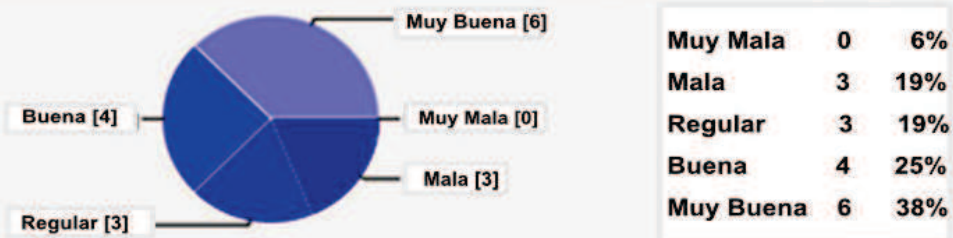


Figura 4.12 Resultados Preguntas Adaptabilidad. Fuente: Google Docs

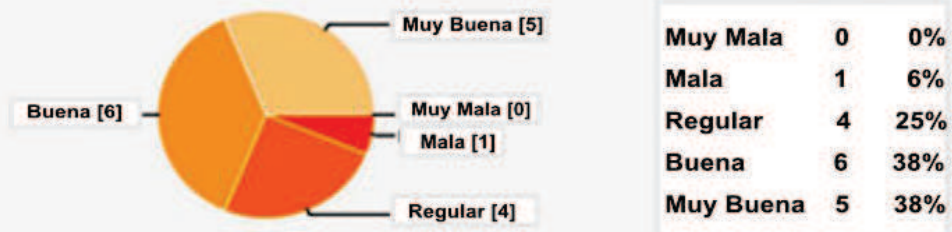
Videojuegos

Jugabilidad

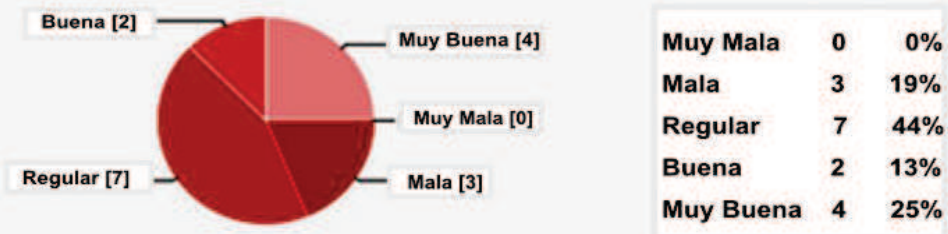
19. ¿La facilidad de aprendizaje de los controles del personaje es?



20. ¿La facilidad de aprendizaje de la mecánica del videojuego es?



21. ¿El grado de diversión al jugar el videojuego es?



22. ¿La dificultad del videojuego es?



Figura 4.13 Resultados Preguntas Jugabilidad. Fuente: Google Docs

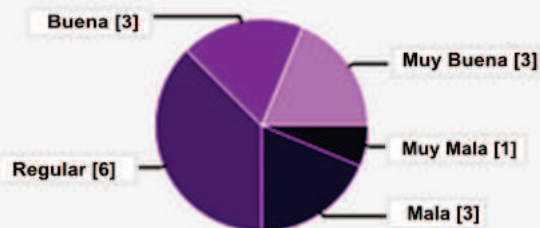
Efectos y Sonido

23. ¿La calidad de los sonidos incorporados es?



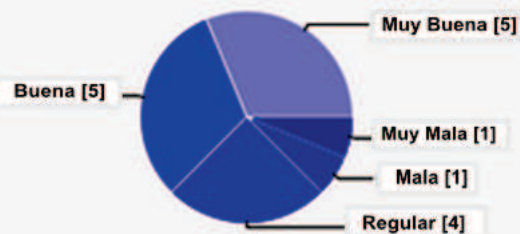
Muy Mala	1	6%
Mala	0	0%
Regular	6	38%
Buena	6	38%
Muy Buena	3	19%

24. ¿La emotividad que denotan los sonidos en las acciones es?



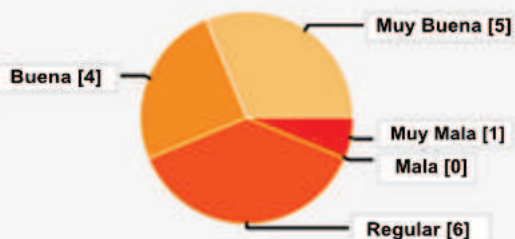
Muy Mala	1	6%
Mala	3	19%
Regular	6	38%
Buena	3	19%
Muy Buena	3	19%

25. ¿la correspondencia de los sonidos con las acciones es?



Muy Mala	1	6%
Mala	1	6%
Regular	4	25%
Buena	5	31%
Muy Buena	5	31%

26. ¿La distinción de los sonidos (efectos, voces ambiente) es?



Muy Mala	1	0%
Mala	0	0%
Regular	6	38%
Buena	4	25%
Muy Buena	5	31%

Figura 4.14 Resultados Preguntas Efectos y Sonido. Fuente: Google Docs

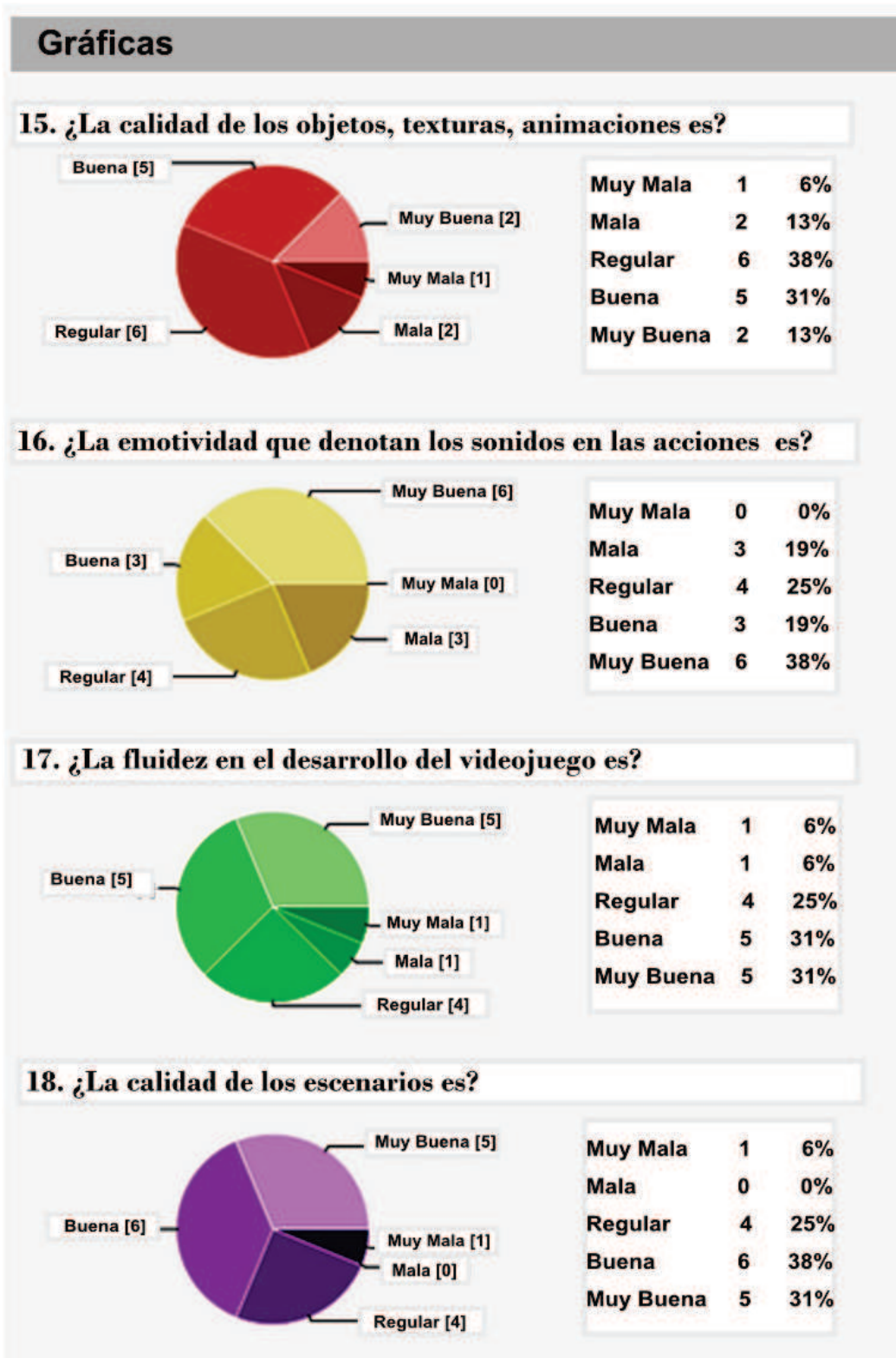


Figura 4.15 Resultados Preguntas Gráficas. Fuente: Google Docs

4.4 ANÁLISIS DE RESULTADOS

Para el análisis de resultados se ha seleccionado las opciones más votadas por los encuestados. De esta manera se diseña una tabla con los criterios de calidad, la opción más votada (del 1 al 5: Muy Mala[1], Mala[2], Regular[3], Buena [4], Muy Buena[5]), el número de votantes de ésta opción, el porcentaje sobre el total de encuestados y los promedios individuales y generales de los criterios evaluados. Los datos obtenidos se los aprecia en la Tabla 4.7.

Resultados de la Evaluación			Total Encuestados		16		Promedio Individual	Total/5
Multimedia	Simplicidad	Mayor Opción Seleccionada	4	4	4	4	4	4,31
		Porcentaje	56%	38%	44%	31%		
		Número de Personas	9	6	7	5		
	Coherencia	Mayor Opción Seleccionada	4	4	4	4	4	
		Porcentaje	31%	38%	31%	38%		
		Número de Personas	5	6	5	6		
	Claridad	Mayor Opción Seleccionada	4	4	4	5	4,25	
		Porcentaje	50%	31%	56%	44%		
		Número de Personas	8	5	9	7		
	Adaptabilidad	Mayor Opción Seleccionada	5	5			5	
		Porcentaje	50%	44%				
		Número de Personas	8	7				
Videojuegos	Gráficas	Mayor Opción Seleccionada	4	5	4,5	4	4,38	3,88
		Porcentaje	31%	38%	63%	38%		
		Número de Personas	5	6	10	6		
	Jugabilidad	Mayor Opción Seleccionada	5	4	3	3	3,75	
		Porcentaje	38%	38%	44%	63%		
		Número de Personas	6	6	7	10		
	Efectos y Sonido	Mayor Opción Seleccionada	3,4	3	4,5	3	3,50	
		Porcentaje	76%	38%	63%	38%		
		Número de Personas	12	6	10	6		

Tabla 4.7 Evaluación Resultados Encuesta [8]

Los resultados obtenidos de acuerdo al diseño, da una calificación general de 4.31/5 en la evaluación de criterios Multimedia y 3.88/5 en la evaluación de criterios videojuegos. El videojuego tiene una calidad aceptable, para los

criterios evaluados. Se debe mejorar la funcionalidad de los criterios que obtuvieron baja puntuación.

5. CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Sobre la base de la investigación bibliográfica realizada acerca de la industria de videojuegos se concluye que el principal factor para que no se desarrolle es la falta de conocimiento sobre su rentabilidad; además la poca difusión de los productos desarrollados por las empresas, también ha contribuido a que los sectores privados y el gobierno no se interesen en invertir en la industria. La mayoría de desarrolladores de videojuegos lo hacen por pasatiempo y las pocas empresas de nuestro país en este ámbito, desarrollan productos a medida de los clientes, en su mayoría extranjeros.
- El uso de metodologías ágiles para desarrollar la aplicación permitió tener un producto funcional, sin necesidad de realizar una documentación exhaustiva, más adaptable a cambios en el transcurso de su desarrollo; ahorrando tiempo y recursos.
- El uso de SCRUM permitió incorporar el proceso de construcción de elementos multimedia, al desarrollo del videojuego. La pila de requerimientos usada en el proyecto permitió incorporar las tareas de manera sencilla.
- Para el desarrollo del videojuego se usaron diferentes herramientas, el modelado de personajes y añadir lógica a la aplicación, fueron pilares fundamentales del proyecto. Las herramientas seleccionadas son intuitivas, capaces de relacionarse entre sí a través de la exportación/importación de objetos, de fácil manejo y abundante documentación.
- En el presente trabajo se usó herramientas libres o herramientas con licencia de uso sin costo, lo cual disminuyó la cantidad de dinero que debía ser invertida. Esto puede beneficiar a proyectos que no tengan suficiente financiamiento.

- La elaboración de diseños de bocetos bidimensionales, facilitó la construcción de los personajes 3D, a diferencia de la construcción de los assets (para la construcción de terrenos), que no contaban con bocetos previos para su construcción.
- El uso de diagramas UML en el presente proyecto, contribuyó a un correcto entendimiento de la lógica que se deseaba implementar, además de la integración de los componentes con cada uno de los objetos de la escena.
- La asignación de responsables para cada tarea realizada en este proyecto facilitó el cumplimiento de las tareas en los tiempos estipulados.
- La construcción y animación de los modelos tridimensionales fueron las tareas que se extendieron la duración del proyecto. Es útil utilizar material no exclusivo para construir un videojuego. Se puede encontrar distintos modelos en páginas como: 3deOcean, OpenGameArt.org; o canciones para videojuegos en páginas como: Incompetech o Youtube. Esto permite ahorrar tiempo en el desarrollo del videojuego.
- Las pruebas de validación de datos, no son necesarias para la realización de un videojuego. No existe E/S de datos, además de las E/S de los controles de usuario.
- Se realizaron pruebas de validación contra los requerimientos de la aplicación, para verificar la correcta ejecución de los controles y la funcionalidad. Al establecer las pruebas en un desarrollo tradicional, se busca errores en módulos independientes; en un videojuego los módulos están relacionados entre sí, por esta razón los errores son más notorios al ejecutar la aplicación.
- Las pruebas se las realizó en cada iteración; esto permitió corregir errores antes del lanzamiento de la versión beta del producto para los usuarios finales.
- Las pruebas con los usuarios, han permitido conocer los defectos, no visibles en el desarrollo, que presenta la aplicación. Al realizar correcciones sobre estos errores se mejora la calidad del producto final.
- Aplicar ingeniería mejora el proceso de desarrollo de un videojuego, estima los tiempos de desarrollo de la aplicación, establece ciclos, organiza tareas; en

general realiza una correcta planificación y ordenamiento de las actividades a cumplir.

5.2 RECOMENDACIONES

- Realizar una socialización acerca de los productos desarrollados por empresas ecuatorianas y el alcance que tiene la industria de los videojuegos a nivel global. De esta forma se puede fomentar el desarrollo de este tipo de aplicaciones en el país.
- Desarrollar videojuegos para dispositivos móviles o browser incrementa la probabilidad de difundirlos. La tecnología cada vez se inclina más a la movilidad con servicios disponibles en la nube. Aunque las características del videojuego desarrollado en este proyecto sean para un computador de escritorio, se puede modificarlas; haciéndolo funcional para un browser o un dispositivo móvil.
- Continuar con estudios relacionados al desarrollo de videojuegos, mejora los procesos en su construcción, es necesario adaptarse a los cambios de las tecnologías y la forma de desarrollar este tipo específico de productos.
- Usar SCRUM para añadir características de cualquier metodología, adaptarlas al ciclo de desarrollo, generar documentación liviana y adaptarse a los cambios que se presenten.
- Usar herramientas open source para disminuir los costos de desarrollo en proyectos independientes similares a este proyecto.
- Realizar un diseño de personajes y escenarios muy detallado, para facilitar al artista modelador 3d su trabajo. Es conveniente realizar un bosquejo 2d para construir los modelos 3d.
- Usar material no exclusivo construido por terceras personas, permite disminuir el tiempo en el desarrollo de un proyecto. La mayoría del material disponible es de buena calidad con un costo relativamente bajo, los desarrolladores independientes generalmente los usan para la construcción de assets, música, personajes secundarios 3d, texturas, entre otras posibilidades.

- Realizar pruebas continuas al código, con el fin de mejorar la calidad del producto.

GLOSARIO

Rayos Catódicos: corrientes de electrones observados en tubos de vacío que se equipan por lo menos con dos electrodos, un cátodo y un ánodo.

PDP-1: primer computador de la serie PDP de la Digital Equipment, famoso por ser el computador más importante de la cultura hacker en el MIT.

Atari: empresa que desarrolla, publica y distribuye videojuegos para consolas.

Joystick: proviene del inglés joy, alegría, y stick, palo. Dispositivo de control de dos o tres ejes que se usa desde una computadora o videoconsola.

Sears: cadena estadounidense de tiendas departamentales fundada por Richard Warren Sears en el siglo XIX.

Survival Horror: en español Terror de Supervivencia, es un subgénero perteneciente al género de los videojuegos conocido como aventuras de acción.

Remakes: se refieren a producciones nuevas que reproducen un trabajo anterior modificando ciertos aspectos.

Role Playing Game: juego de Rol, en el que los jugadores asumen el rol de los personajes a medida que se desarrolla una historia.

Shin Nihon Kikaku: empresa japonesa dedicada al desarrollo de videojuegos.

Script Creation Utility for Maniac Mansion: creado como motor para el videojuego Maniac Mansion.

Rumble Pack: dispositivo creado por Nintendo, diseñado para el mando de la Nintendo 64 proporcionaba vibración en la acción del videojuego.

Vista Isométrica: esta vista es usada para crear la sensación de profundidad en los videojuegos 3D.

Giga Disk Read-Only Memory: es un formato de discos óptico desarrollado por Yamaha para la compañía de videojuegos SEGA.

First Person Shooter: es un género de videojuegos que centra a un jugador en un arma de combate, estos videojuegos son conocidos como shooter games.

Play Station Portable: videoconsola portable de la multinacional Sony Computer Entertainment.

High Definition Multimedia Interface: es una norma de audio y video digital cifrado sin compresión.

MPEG-2 Audio Layer III: formato de compresión de audio digital.

Advanced Audio Coding: formato informático de señal digital de audio basado en un algoritmo de compresión por pérdida, obteniendo mayor grado de compresión.

Estereoscópico: es una técnica capaz de recoger información visual y crear una ilusión de profundidad en una fotografía o película, se crea creando una imagen diferente para cada ojo. Fue inventada por Sir Charles Wheatstone en 1840.

MicroConsole: es un dispositivo conectado al televisor que permite la transmisión de los datos desde internet de los videojuegos almacenados en la nube de Onlive.

Onlive: sistema de distribución de videojuegos bajo demanda, con videojuegos almacenados computados y renderizados en línea.

Videogamers: jugador de videojuegos, que juega o se informa con gran dedicación e interés.

Streaming: distribución multimedia a través de una red de computadoras, el usuario consume el producto al mismo tiempo que lo descarga de manera continua sin interrupción con ayuda de un buffer de datos, que almacena datos y los muestra al usuario.

Poligonales: son gráficos que la computadora almacena con posición de puntos, líneas y caras, construyendo un polígono, en un espacio de 3 dimensiones.

Texturizados: se refiere a las imágenes colocadas sobre una o más caras de un polígono.

GPU: Graphics Processing Unit o Unidad de Procesamiento de Gráficos, es un coprocesador dedicado al procesamiento de gráficos u operaciones de coma flotante para aligerar la carga de trabajo del procesador central.

Shading o Iluminación: implica la simulación de computadora del comportamiento de las caras de un polígono cuando son iluminadas por una fuente de luz virtual.

Diseño Artístico: conceptualización de personajes, escenarios, criaturas etc.

Interfaz Grafica: también conocida como GUI, actúa como un conjunto de imágenes

y objetos que representa la información y acciones disponibles al usuario.

Motor Grafico: también llamado Motor de Videojuego, implementa rutinas de programación que permiten el diseño, la creación y la representación de un videojuego

Blue Lizard Games: empresa desarrolladora de videojuegos con oficinas en Quito y Canadá.

Metodologías: en ingeniería de software es el marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información.

Artefactos: es un producto tangible resultante del proceso de desarrollo de software por ejemplo: casos de uso, diagramas de clase, etc.

Renderizar: render en inglés, se refiere al proceso de generar una imagen desde un modelo, interpreta una escena en 3 dimensiones y la plasma en una imagen bidimensional.

Colisiones: refiere a la simulación del choque entre dos objetos a los que se les da propiedades como por ejemplo la fuerza.

DirectX: es una colección de API desarrolladas para facilitar las tareas relacionadas con multimedia, especialmente programación de juegos y video sobre Microsoft.

Unreal Development Kit: versión sin costo del Motor Unreal Engine para promover el desarrollo de novatos sobre este motor.

OpenGL: Open Graphics Library, define una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos en 2D y 3D.

Plugins: es una aplicación que se relaciona con otra para agregarle nueva funcionalidad.

Bump Mapping: técnica en computación gráfica para simular golpes y arrugas en la superficie de un objeto.

Reflection Mapping: en computación gráfica es una eficiente técnica de iluminación basada en imagen, para aproximar la apariencia de una superficie reflectante por medio de una imagen de textura pre-calculada.

Parallax Mapping: también llamado mapeo de offset, o mapeo de desplazamiento virtual, es una mejora del bump mapping, las texturas tienen profundidad mucho más evidente lo que les da más realismo

BIBLIOGRAFÍA

- [1] Diario El Comercio, «Reportaje El desarrollo de videojuegos en el país tiene un freno con la piratería,» [En línea]. Available: http://www.elcomercio.ec/negocios/desarrollo-videojuegos-pais-freno-pirateria_0_527347326.html. [Último acceso: 11 08 2013].
- [2] P. Steven, TRIGGER HAPPY, Arcade Publishing, 2004.
- [3] Retro Informática, «Historia de los videojuegos,» [En línea]. Available: <http://www.fib.upc.edu/retro-informatica/historia/videojocs.html>. [Último acceso: 08 11 2013].
- [4] Otakufreaks, «Historia de los videojuegos,» [En línea]. Available: <http://www.otakufreaks.com/historia-de-los-videojuegos-el-origen-y-los-inicios>. [Último acceso: 08 11 2013].
- [5] Isopixel, «Historia de los videojuegos,» [En línea]. Available: <http://isopixel.net/archivo/2011/11/historia-de-los-videojuegos-infografia/>. [Último acceso: 08 11 2013].
- [6] ElOtroLado wiki, «Historia de los videojuegos,» [En línea]. Available: http://www.elotrolado.net/wiki/Historia_de_los_videojuegos. [Último acceso: 08 11 2013].
- [7] RinconSolero, «Historia de los videojuegos,» [En línea]. Available: http://www.rinconsolero.com/Rinconsolero.V2/historia_de_los_videojuegos.htm. [Último acceso: 08 11 2013].
- [8] M. AGILA y A. CEVALLOS, *Realizado por los autores*, Quito, 2013.
- [9] GamerZona, «Nuevo horizonte de las videoconsolas,» [En línea]. Available: <http://www.gamerzona.com/2012/03/17/son-xbox-720-wii-u-y-playstation-4-las-ultimas-consolas/>. [Último acceso: 08 11 2013].
- [10] Wikipedia, «OnLive,» [En línea]. Available: <http://es.wikipedia.org/wiki/OnLive>. [Último acceso: 08 11 2013].
- [11] 3D Juegos, «Videoconsolas,» [En línea]. Available: <http://www.3djuegos.com/foros/tema/381462/0/evolucion-de-graficos-y-consolas-en-toda-la-historia/>. [Último acceso: 08 11 2013].
- [12] Rinconsolero, «Videoconsolas,» [En línea]. Available: <http://www.3djuegos.com/foros/tema/381462/0/evolucion-de-graficos-y-consolas-en-toda-la-historia/>. [Último acceso: 08 11 2013].
- [13] Taringa, «Evolución de Polígonos en los videojuegos,» [En línea]. Available: <http://www.taringa.net/posts/juegos/7345421/Evolucion-en-la-tecnologia-grafica-en-los-videojuegos.html>. [Último acceso: 08 11 2013].
- [14] LaPS3, «Evolución de Polígonos en los videojuegos,» [En línea]. Available: http://www.laps3.com/foro/10_charla/177960-curiosidad_poligonos_en_los_videojuegos.html. [Último acceso: 08 11 2013].

- [15] Diario Hoy, «Reportaje Diario Hoy Ecuador en las Grandes Ligas de los Videojuegos,» [En línea]. Available: <http://www.hoy.com.ec/noticias-ecuador/ecuador-en-las-grandes-ligas-de-los-videojuegos-541469.html>. [Último acceso: 08 11 2013].
- [16] AESOFT, «Catálogo Software 2012-2013. Matriz de Sectores,» [En línea]. Available: <http://aesoft.com.ec/www/index.php/182-catalogo-2012-2013>. [Último acceso: 08 11 2013].
- [17] REDICI, «Acciones ejecutadas en la política de telecomunicaciones de Colombia,» [En línea]. Available: <http://www.redici.org/podcast/99-edicion043>. [Último acceso: 08 11 2013].
- [18] Dracomedia, «De Estadodo un poco. Estadísticas de los videojuegos,» [En línea]. Available: <http://dragonomega.wordpress.com/2011/07/31/estadisticas-de-videojuegos/>. [Último acceso: 08 11 2013].
- [19] M. J, RAPID APPLICATION DEVELOPMENT, New York: Macmillan Inc, 1991.
- [20] L. P. P. M. C. CANÓS José H., METODOLOGÍAS ÁGILES EN EL DESARROLLO DE SOFTWARE., Valencia: DSIC-Universidad Politécnica de Valencia.
- [21] Ambysoft Scott W. Amber, «Metodología AUP,» [En línea]. Available: <http://www.ambysoft.com/unifiedprocess/agileUP.html>. [Último acceso: 08 11 2013].
- [22] CodeProject, «Metodología SCRUM,» [En línea]. Available: URL: <http://www.codeproject.com/Articles/4798/What-is-SCRUM>. [Último acceso: 08 11 2013].
- [23] Proyectos Ágiles Org, «Fundamentos SCRUM,» [En línea]. Available: <http://www.proyectosagiles.org/fundamentos-de-scrum>. [Último acceso: 08 11 2013].
- [24] Itzcoalt Alvarez M, «Desarrollo Ágil con SCRUM,» [En línea]. Available: <http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:sg07.p02.scrum.pdf>. [Último acceso: 08 11 2013].
- [25] P. E. GUALOTUÑA COLLAGUAZO, *INTEGRACIÓN DE UXD (USER INTERFACE DESING) CON UNA METODOLOGÍA ÁGIL DE DESARROLLO DE SOFTWARE APLICADA A UN CASO DE ESTUDIO*, Quito: EPN, 2013.
- [26] C. REINOSO, «Métodos Ágiles,» [En línea]. Available: <http://carlosreynoso.com.ar/archivos/arquitectura/Metodos-Agiles.PDF>. [Último acceso: 08 11 2013].
- [27] A. MOLCEPERES, PROCESOS DE DESARROLLO: RUP, XP Y FDD.
- [28] Wikipedia, «Motor Gráfico,» [En línea]. Available: http://es.wikipedia.org/wiki/Motor_de_videojuego. [Último acceso: 08 11 2013].
- [29] Develop-online, «Motores Gráficos, Top10,» [En línea]. Available: <http://www.develop-online.net/news/tag/the-top-10-game-engines>. [Último acceso: 08 11 2013].
- [30] Unreal, «Motor Gráfico,» [En línea]. Available: <http://www.unrealengine.com>. [Último acceso: 08 11 2013].

- [31] Gamebryo, «Motores Gráfico,» [En línea]. Available: <http://www.gamebryo.com> . [Último acceso: 08 11 2013].
- [32] Crytech, «Motor Gráfico CryEngine,» [En línea]. Available: <http://www.crytek.com/cryengine>. [Último acceso: 08 11 2013].
- [33] UNITY, «Herramienta Unity3D,» [En línea]. Available: <http://www.unity3d.com>. [Último acceso: 08 11 2013].
- [34] Bliztech , «Motor Gráfico,» [En línea]. Available: <http://www.blitztech.com>. [Último acceso: 08 11 2013].
- [35] Herramientas modelado y animación 3D, «Herramientas modelado y animación 3D,» [En línea]. Available: <http://es.slideshare.net/gbgarcia/herramientas-de-modelado-3d-2455690>. [Último acceso: 08 11 2013].
- [36] Autodesk , «Autodesk 3DMax.,» [En línea]. Available: <http://www.autodesk.es/products/autodesk-3ds-max/overview>. [Último acceso: 08 11 2013].
- [37] Autodesk , «Autodesk Maya,» [En línea]. Available: <http://www.autodesk.es/products/autodesk-maya/overview>. [Último acceso: 08 11 2013].
- [38] Cinema 4D., «Cinema 4D.,» [En línea]. Available: <http://www.maxon.net/es/products/cinema-4d-studio.html>. [Último acceso: 08 11 2013].
- [39] BLENDER, «Herramienta Blender,» [En línea]. Available: <http://www.blender.org/>. [Último acceso: 08 11 2013].
- [40] Pixologic, «ZBrush,» [En línea]. Available: <http://pixologic.com/zbrush/>. [Último acceso: 08 11 2013].
- [41] Wikipedia, «Herramientas de diseño 2D,» [En línea]. Available: http://es.wikipedia.org/wiki/Dise%C3%B1o_asistido_por_computadora. [Último acceso: 08 11 2013].
- [42] InkScape ORG, «InkScape,» [En línea]. Available: <http://inkscape.org/?lang=es>. [Último acceso: 08 11 2013].
- [43] Adobe , «Adobe Photoshop.,» [En línea]. Available: <http://www.adobe.com/la/products/photoshop.html>. [Último acceso: 08 11 2013].
- [44] GIMP ORG, « Herramienta GIMP,» [En línea]. Available: <http://www.gimp.org/>. [Último acceso: 08 11 2013].
- [45] Corel Draw, «Corel Draw,» [En línea]. Available: <http://www.corel.com/corel/product/index.jsp?pid=prod4260069&cid=catalog20038&segid=10700003&storeKey=us&languageCode=en>. [Último acceso: 08 11 2013].
- [46] Autodesk, «Autodesk AutoCad,» [En línea]. Available: <http://www.autodesk.com/products/autodesk-autocad/overview>. [Último acceso: 08 11 2013].
- [47] Pixologic, «Herramienta Sculptris,» [En línea]. Available: <http://www.pixologic.com/sculptris>. [Último acceso: 08 11 2013].

- [48] ICESCRUM ORG, «Herramienta ICESCRUM,» [En línea]. Available: <http://www.icescrum.org>. [Último acceso: 08 11 2013].
- [49] ElOtroLado wiki, «Clasificación de los videojuegos,» [En línea]. Available: http://www.elotrolado.net/wiki/G%C3%A9neros_de_los_videojuegos. [Último acceso: 08 11 2013].
- [50] Wikipedia, «Clasificación de los videojuegos,» [En línea]. Available: http://es.wikipedia.org/wiki/G%C3%A9nero_de_videojuegos. [Último acceso: 08 11 2013].
- [51] T. MULLEN, ANIMACION DE PERSONAJES CON BLENDER, España: Anaya Diseño y Creatividad, 2008.
- [52] J. WILLAMSON, *LOW POLY GAME CHARACTER SERIES*, CGCookie Video Tutorial.
- [53] K. MacLeod, «Incompetech,» [En línea]. Available: <http://incompetech.com/>. [Último acceso: 08 11 2013].
- [54] A. S. Unity, «Assets Store Unity,» [En línea]. Available: <http://www.assetstore.unity3d.com> . [Último acceso: 08 11 2013].
- [55] 3. O. Envato, «3D Ocean Envato,» [En línea]. Available: <http://3docean.net> . [Último acceso: 08 11 2013].
- [56] Blend-Swap, «Blend-Swap,» [En línea]. Available: <http://blendswap.com> . [Último acceso: 08 11 2013].
- [57] O. G. A. Org., «Open Game Art,» [En línea]. Available: <http://opengameart.org>. [Último acceso: 008 11 2013].
- [58] L. Salvemini, *A COMPLETE BEGINNER'S INTRODUCTION TO RIGGING*, Blender Guru 2013..
- [59] Unity, «Unity Scripting,» [En línea]. Available: <http://docs.unity3d.com/Documentation/ScriptReference/>. [Último acceso: 08 11 2013].
- [60] UNITYTECHNOLOGIES, 3DPLATFORMER TUTORIAL, UnityAssetStore, 2012.
- [61] UNITYTECHNOLOGIES, PENELOPE, UnityAssetStore, 2011.
- [62] J. WITTAYABUNDIT, *UNITY3 HOTSHOT*, Birmingham UK: Pack Publishing, 2011..
- [63] R. PRESSMAN S., *INGENIERÍA DE SOFTWARE UN ENFOQUE PRÁCTICO*, México: Editorial McGrawHill Sexta Edición, 2005.
- [64] C. BERLOCH O., «EVALUACIÓN DE APLICACIONES MULTIMEDIA, CRITERIOS DE CALIDAD,» [En línea]. Available: Criterios Multimedia. [Último acceso: 08 11 2013].

ANEXOS

Los anexos se encuentran en el CD adjunto.

ANEXO A: Casos de Prueba

ANEXO B: Código Fuente de la Aplicación

ANEXO C: Herramienta ICESCRUM, junto con el archivo Tesis.xml que contiene todas las iteraciones del proyecto