



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE
MÉXICO**

Centro Universitario UAEM Ecatepec

**Realidad Virtual en la Educación: Matemáticas
en Tercer Grado de Primaria**

TESIS

Para obtener el título de
Ingeniero en Computación

PRESENTA:

C. Maldonado Rodríguez Carlos Alberto

C. Zamora Sánchez Ricardo

Tutor Académico:

M. en I.S.C. Alejandra Morales Ramírez

Revisores:

Dr. En C. Rodolfo Zolá García Lozano

M. en I.S.C. Juan de Jesús Amador Reyes



Ecatepec de Morelos, Estado de México, Junio 2017



CARTA DE CESIÓN DE DERECHOS DE AUTOR

El (la) que suscribe **C.C. CARLOS ALBERTO MALDONADO RODRIGUEZ Y RICARDO ZAMORA SANCHEZ** Autor del trabajo escrito de evaluación profesional en la opción de **TESIS** con el título **“REALIDAD VIRTUAL EN LA EDUCACION: MATEMATICAS EN TERCER GRADO DE PRIMARIA”** por medio de la presente con fundamento en lo dispuesto en los artículos 5, 18, 24, 25, 27, 30, 32 y 148 de la Ley Federal de Derechos de Autor, así como los artículos 35 y 36 fracción II de la Ley de la Universidad Autónoma del Estado de México; manifiesto mi autoría y originalidad de la obra mencionada que se presentó en el Centro Universitario UAEM Ecatepec para ser evaluada con el fin de obtener el Título Profesional de **INGENIERIA EN COMPUTACION**

Así mismo expreso mi conformidad de ceder los derechos de reproducción, difusión y circulación de esta obra, en forma NO EXCLUSIVA, a la Universidad Autónoma del Estado de México; se podrá realizar a nivel nacional e internacional, de manera parcial o total a través de cualquier medio de información que sea susceptible para ello, en una o varias ocasiones, así como en cualquier soporte documental, todo ello siempre y cuando sus fines sean académicos, humanísticos, tecnológicos, históricos, artísticos, sociales, científicos u otra manifestación de la cultura.

Entendiendo que dicha cesión no genera obligación alguna para la Universidad Autónoma del Estado de México y que podrá o no ejercer los derechos cedidos.

Por lo que el autor da su consentimiento para la publicación de su trabajo escrito de evaluación profesional.

- a) Texto completo
- b) Por capítulo
- c) Solamente portada y tabla de contenido

<input checked="" type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>

Se firma presente en la ciudad de Ecatepec de Morelos, Estado de México, a los 2 días del mes de Junio de 2017.

CARLOS ALBERTO MALDONADO RODRIGUEZ

RICARDO ZAMORA SANCHEZ



UAEM | Universidad Autónoma
del Estado de México

Ecatepec de Morelos, Edo. De Méx., a 19 de Mayo de 2017
ASUNTO: VOTO APROBATORIO DE ASESOR

LIA. ADRIANA MORALES LICONA
JEFA DEL DEPARTAMENTO DE TITULACION DEL
CENTRO UNIVERSITARIO U.A.E.M ECATEPEC
P R E S E N T E

Por éste conducto me permito informarle que el (los) pasantes **C.C. CARLOS ALBERTO MALDONADO RODRIGUEZ y RICARDO ZAMORA SANCHEZ** con el número de cuenta **1025480 Y 1025497**, de la **INGENIERIA EN COMPUTACION**, ha concluido el desarrollo de su **TESIS**, con el título:

“REALIDAD VIRTUAL EN LA EDUCACION: MATEMATICAS EN TERCER GRADO DE PRIMARIA”

Manifiesto que el borrador del trabajo escrito reúne las características necesarias para ser revisado por la Comisión especial nombrada para tal efecto.


M. en I.S.C. **ALEJANDRA MORALES RAMIREZ**
NO. DE CÉDULA PROFESIONAL: **5782891**



PATRIA, CIENCIA Y TRABAJO

“2017, Año del Centenario de la Promulgación de la Constitución Política de los Estados Unidos Mexicanos”



www.uaemex.mx



UAEM | Universidad Autónoma
del Estado de México

Ecatepec de Morelos, Edo. De Méx., 2 de Junio de 2017
ASUNTO: VOTO APROBATORIO DE REVISORES

LIA. ADRIANA MORALES LICONA
JEFA DEL DEPARTAMENTO DE TITULACION DEL
CENTRO UNIVERSITARIO UAEM ECATEPEC
PRESENTE

Nos es grato comunicarle que el trabajo de **TESIS** titulado:

"REALIDAD VIRTUAL EN LA EDUCACION: MATEMATICAS EN TERCER GRADO DE PRIMARIA"

Que para obtener el título de: **INGENIERIA EN COMPUTACION**

Presenta la pasante: **CARLOS ALBERTO MALDONADO RODRIGUEZ y RICARDO**
ZAMORA SANCHEZ

Con números de cuenta: **1025480 y 1025497**

Cumplen con los requisitos teóricos-metodológicos suficientes para ser aprobada, pudiendo continuar con los trámites correspondientes para su impresión.

REVISORES

DR. RODOLFO ZOLA GARCIA LOZANO
CÉDULA PROFESIONAL: 2706467

M. en C.C. JUAN DE JESUS AMADOR REYES
CÉDULA PROFESIONAL: 9118134



PATRIA, CIENCIA Y TRABAJO

"2017, Año del Centenario de la Promulgación de la Constitución Política de los Estados Unidos Mexicanos"



www.uaemex.mx



Universidad Autónoma del Estado de México
Centro universitario UAEM Ecatepec

Ecatepec de Morelos, Estado de México a 2 de Junio de 2017
ASUNTO: IMPRESIÓN DE TRABAJO ESCRITO

C.C. CARLOS ALBERTO MALDONADO RODRIGUEZ Y RICARDO ZAMORA SANCHEZ
PASANTE DE LA INGENIERIA EN COMPUTACION
DEL CENTRO UNIVERSITARIO UAEM ECATEPEC
P R E S E N T E

Por este medio le comunico a usted que al haber cubierto los trámites correspondientes al desarrollo del trabajo escrito bajo la modalidad **TESIS** con el fin de obtener el Título Profesional, se le aprueba la **IMPRESIÓN DE SU TRABAJO** con el título:

“REALIDAD VIRTUAL EN LA EDUCACION: MATEMATICAS EN TERCER GRADO DE PRIMARIA”

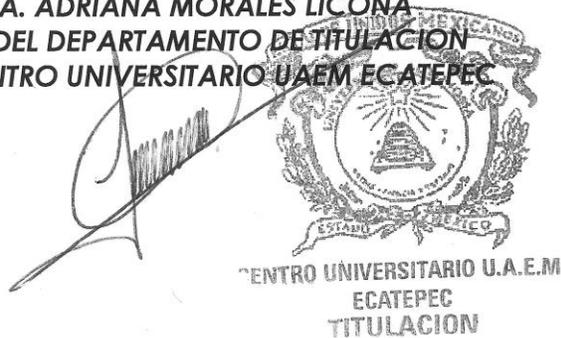
Con el objetivo de establecer la fecha de Evaluación Profesional, le recuerdo que la presentación final del trabajo escrito es de su completa responsabilidad.

Sin más por el momento, reciba un cordial saludo.

PATRIA, CIENCIA Y TRABAJO

“2017, Año del Centenario de la Promulgación de la Constitución de los Estados Unidos Mexicanos”

LIA. ADRIANA MORALES LICONA
JEFA DEL DEPARTAMENTO DE TITULACION
DEL CENTRO UNIVERSITARIO UAEM ECATEPEC



Agradecimientos de Carlos

Papá y Mamá, gracias por su apoyo incondicional que me impulso a ser la persona que ahora soy, por creer siempre en mí, de igual forma por mostrarme que en esta vida se pueden lograr las cosas que uno se propone. Papá, has sido y serás siempre mi superhéroe favorito por mostrarme como luchar para salir adelante, gracias por brindarme la oportunidad de estudiar y prepararme para tener una mejor vida. Mamá, te llevas el premio de la mejor mamá del universo por haber forjado la persona que ahora soy, ustedes son un ejemplo a seguir para mí.

Michel, mi amor, eres la persona que siempre elegiría. Gracias por apoyarme siempre en mis decisiones y formar parte de esta etapa, tú eres mi fuente de inspiración más grande. Te amo.

Hermanos, agradezco infinitamente haber llegado a este mundo y formar parte de esta hermosa familia, tantos momentos felices que alimentaron mi desarrollo personal y académico. Son los mejores hermanos del universo, los quiero mucho.

Primo Frank y familia, ustedes son el tipo de personas que agradezco tener a mi lado. Gracias por brindarme su apoyo y permitirme vivir a su lado durante mi estancia en Monterrey. Durante ese periodo me di cuenta que eras más que mi primo, te considero mi hermano. Cada consejo lo tomo como parte de mi crecimiento personal y académico, los llevo en mi corazón.

Maestra Ale gracias por su gran apoyo y paciencia durante el desarrollo de este proyecto.

Omegavolts, esto no hubiera sido posible si el destino no nos hubiera unido. Gracias por cada momento de diversión, cada plática y cada locura.

Por último, me gustaría que este trabajo sirva como motivador para todos, principalmente a las personas que aprecio y que se den cuenta que todo se puede lograr, solo es cosa de que se lo propongan.

Agradecimientos de Ricardo

A mi Familia

A mis padres Juana y Gregorio por ser un ejemplo a seguir y salir adelante, por darme la oportunidad de estudiar, lograr mis metas y estar ahí en todo momento, a sus consejos para crecer como persona y profesionalmente ahora y a lo largo de la vida. A mis hermanos María Teresa, Sandra y Carlos, que me apoyaron y alentaron en continuar y crecer profesionalmente. A mi novia Mariela Alejandra, por su apoyo, sus consejos, sus pláticas y permitir compartir esta etapa con ella. ¡Gracias a ustedes!

A mi Alma Máter

A la Universidad Autónoma del Estado de México, C. U. UAEM Ecatepec y a cada uno de los Profesores que gracias a sus experiencias y conocimientos me dieron la oportunidad de aprender de cada uno de ellos, por sus consejos de crecer profesionalmente y compartir a la sociedad lo que ellos nos han compartido en estos 5 años de carrera universitaria. ¡Gracias mi Alma Máter, Gracias Profesores!

A mis Amigos

A Omegavolts por estar día con día conviviendo como compañeros, amigos y hermanos, en el apoyo y la confianza que me brindaron, en compartir sus conocimientos durante la estancia y sobre todo, por estar ahí y compartir grandes cosas tanto escolares como personales. Un privilegio el haberlos conocido en esta etapa de la vida. ¡Gracias Omegavolts!

A la Maestra en I.S.C. Alejandra Morales Ramírez y Amigo Carlos Maldonado

Por darme la oportunidad de trabajar con ustedes ésta tesis, que a pesar de estar en dudas desde un principio y de descuidarla al final del camino no dejaron de confiar en mí, por su tiempo para que esto se realizara y llevara a cabo, por aconsejarme y alentarme a no dejar de lado este proyecto. ¡Gracias Maestra y Amigo!

Índice

Introducción.....	1
Capítulo 1. Marco teórico	
1.1. Realidad Virtual.....	9
1.1.1. Concepto de la RV.....	9
1.1.2. Tipos de RV.....	9
1.1.2.1. RV no inmersiva.....	10
1.1.2.2. RV inmersiva.....	10
1.1.3. Componentes de la RV.....	11
1.1.3.1. Mundo virtual.....	12
1.1.3.2. Inmersión.....	12
1.1.3.3. Retroalimentación.....	12
1.1.3.4. Interactividad.....	13
1.2. Características de software enfocado a la educación.....	13
1.3. Condiciones en donde se puede aplicar la RV.....	16
1.4. Aplicaciones de la realidad virtual.....	17
1.5. Juegos serios.....	21
1.6. Recursos artísticos y tecnológicos.....	22
1.7. Técnicas de modelado 3D.....	23
1.7.1. Modelado de caja.....	24
1.7.2. Modelado escultórico.....	24
1.7.3. Modelado de superficies y curvas NURBS.....	25
1.7.4. Modelado a través de displacement maps y height maps.....	26
1.7.5. Modelado por modificadores.....	26
1.7.6. Modelado por clonación.....	27
1.7.7. Modelado booleanico.....	28
1.7.8. Modelado con blue prints.....	28
1.7.9. Modelado de letras y números.....	29
1.7.10. Modelado simétrico.....	30

1.8. Herramientas de modelado 3D.....	30
1.8.1.Autodesk Maya.....	31
1.8.2.Blender.....	31
1.8.3.DAZ Studio.....	32
1.9. Herramientas para generar y editar texturas.....	33
1.9.1.GIMP.....	33
1.9.2.Crazy Bump.....	34
1.10. Motores de representación gráfica 3D.....	35
1.10.1. Unity 3D.....	35
1.10.2. Unreal Engine.....	36
1.10.3. Cry Engine.....	37
1.11. La RV como recurso de enseñanza-aprendizaje.....	38

Capítulo 2. Metodología de desarrollo

2.1. Metodología.....	40
2.2. Fase de análisis.....	40
2.2.1.Identificación de requerimientos.....	41
2.2.2.Propuesta de desarrollo.....	41
2.2.3.Especificación de requerimientos.....	41
2.3. Fase de diseño.....	41
2.3.1.Modelado conceptual.....	42
2.3.2.Modelado de navegación.....	42
2.3.3.Descripción de componentes.....	42
2.3.4.Modelado de eventos.....	43
2.4. Fase de desarrollo.....	43
2.4.1.Producción de recursos artísticos.....	44
2.4.2.Animación e integración.....	44
2.4.3.Producción de recursos tecnológicos.....	44
2.4.4.Pruebas.....	44
2.5. Fase de evaluación.....	45
2.5.1.Valoración por el cliente / usuario.....	45
1.5.2.Fase de cierre y entrega final.....	45

Capítulo 3. Aplicación de la metodología

3.1. Fase de análisis.....	46
3.1.1. Identificación de requerimientos.....	46
3.1.2. Propuesta de desarrollo.....	49
3.1.3. Especificación de requerimientos.....	53
3.2. Fase de diseño.....	54
3.2.1. Modelado conceptual.....	55
3.2.1.1. Escenarios.....	55
3.2.1.2. Objetos 3D.....	57
3.2.2. Modelado de navegación.....	66
3.2.2.1. Diagrama de navegación de escenarios.....	67
3.2.2.2. Diagrama de ubicación de misiones.....	68
3.2.2.3. Diagramas de navegación de misiones.....	68
3.2.2.4. Storyboard.....	75
3.2.3. Descripción de componentes.....	84
3.2.4. Modelado de eventos.....	86
3.3. Fase de desarrollo.....	87
3.3.1. Producción de recursos artísticos.....	88
3.3.1.1. Generación de un Low Poly a partir de un High Poly.....	88
3.3.1.1.1. Ventajas de un objeto Low Poly contra High Poly....	101
3.3.1.2. Creación de una textura a partir de varias imágenes.....	102
3.3.1.3. Creación de personajes en DAZ Studio.....	111
3.3.1.4. Optimización con Simplygon.....	116
3.3.2. Animación.....	121
3.3.2.1. Animación en DAZ Studio.....	121
3.3.2.2. Animación en Unity 3D (Animator).....	127
3.3.3. Integración.....	132
3.3.4. Producción de recursos tecnológicos.....	139
3.3.5. Pruebas.....	153
3.3.6. Empaquetado.....	156

3.4. Fase de evaluación.....	161
3.4.1. Valoración por el cliente / usuario.....	161
Conclusiones.....	165
Trabajo a futuros.....	168
Anexo 1. Reactivos de los temas de fracciones en la prueba ENLACE 2013.....	169
Referencias.....	173

Índice de tablas

Tabla 1.	Resultados 2013 de la primaria Justo Sierra.....	4
Tabla 2.	Resultados 2013 de la primaria Adolfo López Mateos.....	4
Tabla 3.	Resultados 2013 de la primaria Revolución.....	5
Tabla 4.	Información sobre temas y reactivos de la prueba ENLACE del área de matemáticas de la escuela Aquiles Serdán.....	7
Tabla 5.	Información sobre temas y reactivos de la prueba ENLACE del área de matemáticas de la escuela Lázaro Cárdenas.....	7
Tabla 6.	Características de aplicaciones de software enfocadas a la educación.....	13
Tabla 7.	Porcentajes de aprendizaje a través de los sentidos.....	39
Tabla 8.	Requerimientos funcionales y no funcionales.....	41
Tabla 9.	Descripción de componentes.....	42
Tabla 10.	Programa de estudio del tercer bloque de la materia de matemáticas en educación básica.....	47
Tabla 11.	Programa de estudio del cuarto bloque de la materia de matemáticas en educación básica.....	47
Tabla 12.	Programa de estudio del quinto bloque de la materia de matemáticas en educación básica.....	48
Tabla 13.	Propuestas de misiones.....	52
Tabla 14.	Requerimientos funcionales y no funcionales.....	54
Tabla 15.	Bocetos de personajes.....	58
Tabla 16.	Bocetos de elementos del asteroide Vecindario.....	60
Tabla 17.	Bocetos de elementos del asteroide Glorieta.....	61
Tabla 18.	Bocetos de elementos del asteroide Laguna.....	62
Tabla 19.	Bocetos de elementos del asteroide Parque.....	64
Tabla 20.	Componentes de escenario “Vecindario”.....	84
Tabla 21.	Componentes de escenario “Glorieta”.....	84
Tabla 22.	Componentes de escenario “Laguna”.....	85
Tabla 23.	Componentes de escenario “Parque”.....	85
Tabla 24.	Eventos del usuario.....	86

Tabla 25.	Eventos de componentes.....	87
Tabla 26.	Comparación entre High Poly y Low Poly.....	102
Tabla 27.	Porcentaje de optimización en los personajes.....	121
Tabla 28.	Problemas detectados en la fase de integración.....	154
Tabla 29.	Problemas detectados en la fase de producción de recursos tecnológicos...	155
Tabla 30.	Evaluación de los requerimientos funcionales.....	155
Tabla 31.	Observaciones funcionales y visuales de la aplicación.....	156
Tabla 32.	Instrumento para medir el nivel de aceptación de la RA.....	162
Tabla 33.	Instrumento para medir el nivel de aceptación de la RV en los alumnos....	163
Tabla 34.	Instrumento para medir el nivel de aceptación de la RV en los alumnos desde el punto de vista del profesor.....	164

Índice de diagramas

Diagrama 1.	Metodología de desarrollo evolutivo para escenarios tridimensionales	40
Diagrama 2.	Fase de desarrollo.....	43
Diagrama 3.	Modelo de navegación.....	67
Diagrama 4.	Ubicación de misiones.....	68
Diagrama 5.	Navegación de la misión Galactic Labyrinth.....	69
Diagrama 6.	Navegación de la Misión Cosmic Apples.....	70
Diagrama 7.	Navegación de la Misión Spatial Coins.....	71
Diagrama 8.	Navegación de la Misión Stellar Milk.....	72
Diagrama 9.	Navegación de la Misión Moon's Pizza.....	73
Diagrama 10.	Navegación de la Misión Rainbow Fishes.....	74

Índice de storyboards

Storyboard 1.	Bienvenida parte 1.....	77
Storyboard 2.	Bienvenida parte 2.....	78
Storyboard 3.	Bienvenida parte 3.....	79

Storyboard 4.	Misión Cosmic Apples parte 1.....	80
Storyboard 5.	Misión Cosmic Apples parte 2.....	81
Storyboard 6.	Misión Cosmic Apples parte 3.....	82
Storyboard 7.	Misión Cosmic Apples parte 4.....	83

Índice de gráficas

Gráfica 1.	Línea de tiempo nivel de logro.....	6
------------	-------------------------------------	---

Índice de figuras

Figura 1.	RV no Inmersiva.....	10
Figura 2.	RV inmersiva.....	11
Figura 3.	Aplicación médica de la RV.....	18
Figura 4.	Entrenamiento de vuelo con RV.....	19
Figura 5.	Videojuegos con RV.....	19
Figura 6.	Sistema de RV para la representación de caída libre.....	20
Figura 7.	Sistema de RV para entrenamiento de conductas asertivas.....	21
Figura 8.	Modelado de caja.....	24
Figura 9.	Modelado escultórico.....	25
Figura 10.	Modelado con NURBS.....	25
Figura 11.	Modelado con height maps.....	26
Figura 12.	Deformador Lattice.....	27
Figura 13.	Modelado por clonación.....	27
Figura 14.	Operaciones booleanas.....	28
Figura 15.	Modelado con blueprints.....	29
Figura 16.	Modelado de letras y números.....	29
Figura 17.	Modelado simétrico.....	30
Figura 18.	Autodesk Maya.....	31
Figura 19.	Blender.....	32

Figura 20.	DAZ Studio.....	33
Figura 21.	Gimp.....	34
Figura 22.	Crazy Bump.....	34
Figura 23.	Unity 3D.....	36
Figura 24.	Unreal Engine.....	37
Figura 25.	Cry Engine.....	38
Figura 26.	Diseño 2D del asteroide Glorieta.....	50
Figura 27.	Diseño 2D del Asteroide Laguna.....	50
Figura 28.	Diseño 2D del Asteroide Vecindario.....	51
Figura 29.	Diseño 2D del Asteroide Parque.....	51
Figura 30.	Boceto de escenario “Glorieta”.....	55
Figura 31.	Boceto de escenario “Parque”.....	56
Figura 32.	Boceto de escenario “Laguna”.....	56
Figura 33.	Boceto de escenario “Vecindario”.....	57
Figura 34.	Vista en primera persona.....	75
Figura 35.	Tipos de mapas de salida que genera Transfer Maps.....	89
Figura 36.	Objeto con alto LOD.....	89
Figura 37.	Objeto con bajo LOD.....	90
Figura 38.	Mapa UV de Low Poly.....	91
Figura 39.	Mapeado de referencia con checker UV.....	91
Figura 40.	High Poly texturizado.....	92
Figura 41.	High poly y low poly en la misma posición espacial.....	92
Figura 42.	Herramienta Transfer Maps.....	93
Figura 43.	Source y Target Meshes.....	93
Figura 44.	Mapas de salida.....	95
Figura 45.	Asignación de los mapas de salida.....	95
Figura 46.	Maya Common Output.....	97
Figura 47.	Opciones Avanzadas.....	98
Figura 48.	Método de búsqueda.....	98
Figura 49.	Sub objeto generado por la envoltura.....	99
Figura 50.	Personalización de la envoltura.....	99
Figura 51.	Diffuse Map de salida.....	100

Figura 52.	Normal Map de salida.....	100
Figura 53.	High Poly.....	101
Figura 54.	Low Poly.....	101
Figura 55.	Modelo tridimensional de un pino con dos texturas.....	103
Figura 56.	Modelo tridimensional de un pino sin texturas.....	103
Figura 57.	Texturas de un pino.....	104
Figura 58.	Creando una capa nueva en Gimp.....	104
Figura 59.	Importando imágenes a Gimp.....	105
Figura 60.	Imágenes agregadas como capas nuevas.....	105
Figura 61.	Caja de herramientas de Gimp.....	106
Figura 62.	Ajustando imágenes en la capa de referencia.....	106
Figura 63.	Exportando imagen en Gimp.....	107
Figura 64.	Textura resultante de la unión de dos imágenes.....	107
Figura 65.	Creación de un material tipo Lambert.....	107
Figura 66.	Asignación de nombre a un material tipo Lambert.....	108
Figura 67.	Atributos de un material Lambert.....	108
Figura 68.	Selección del tipo de textura.....	109
Figura 69.	Atributos del archivo del canal de color.....	109
Figura 70.	Selección del archivo para el canal de color.....	110
Figura 71.	Modelo tridimensional no mapeado.....	110
Figura 72.	Mapeado de UVs.....	111
Figura 73.	Modelo tridimensional de un pino con una textura.....	111
Figura 74.	Tienda en línea de DAZ Studio.....	113
Figura 75.	Biblioteca de contenido de DAZ Studio.....	113
Figura 76.	Figura base del paquete “Toon Generations”.....	114
Figura 77.	Figura base con ropa y accesorios.....	114
Figura 78.	Modificadores del paquete “Toon Generations”.....	115
Figura 79.	Propiedades de los elementos que integran el personaje.....	116
Figura 80.	Personaje “Charlie” en Daz Studio.....	116
Figura 81.	Accediendo a Simplygon.....	117
Figura 82.	Inicio de sesión en Simplygon.....	117
Figura 83.	Panel Simplygon.....	118

Figura 84.	Número de polígonos del personaje Guía Gus.....	118
Figura 85.	Porcentaje de optimización.....	119
Figura 86.	Optimizando al personaje en el servidor.....	119
Figura 87.	Descargando al personaje optimizado.....	120
Figura 88.	Personaje optimizado.....	120
Figura 89.	Jerarquía del sistema óseo.....	122
Figura 90.	aniMate.....	122
Figura 91.	Agregando inicio de animación.....	123
Figura 92.	Marcador “End of Animation”.....	123
Figura 93.	Marcador “Playhead”.....	124
Figura 94.	Creación de un key frame.....	124
Figura 95.	Animation Baking.....	125
Figura 96.	Exportación de personaje con animación.....	126
Figura 97.	Parámetros de la exportación.....	126
Figura 98.	FBX creado en Daz Studio.....	127
Figura 99.	Modelo tridimensional “Lechería”.....	127
Figura 100.	Ventana Animation.....	128
Figura 101.	Creando una nueva animación.....	128
Figura 102.	Estado de grabación en la ventana Animation.....	129
Figura 103.	Componente Animator.....	129
Figura 104.	Agregando una curva en animación.....	130
Figura 105.	Estableciendo el tiempo de duración de una animación.....	130
Figura 106.	Componente Transform.....	131
Figura 107.	Botón Play de la ventana Animation.....	131
Figura 108.	Desactivando el estado de grabación.....	132
Figura 109.	Organización de librerías de proyecto.....	133
Figura 110.	Paquete “Character Controller”.....	133
Figura 111.	Creación de folder en Unity.....	134
Figura 112.	Biblioteca de proyecto.....	134
Figura 113.	Paquete FBX del terreno “Glorieta”.....	135
Figura 114.	Proceso de importación.....	135
Figura 115.	Contenido del paquete “Suelo_glorieta”.....	136

Figura 116.	Jerarquía de la escena en Unity.....	136
Figura 117.	Conflicto de tipo de textura.....	137
Figura 118.	Configuración de textura Normal Map.....	137
Figura 119.	Conversión de escala de grises a Normal Map.....	138
Figura 120.	Efecto de Normal Map en pasto.....	138
Figura 121.	Integración de los cuatro escenarios.....	139
Figura 122.	Carpeta de Scripts.....	140
Figura 123.	Creación del script Fraccion.....	141
Figura 124.	Script Fraccion.....	141
Figura 125.	Entorno de desarrollo “MonoDevelop”.....	142
Figura 126.	Script de la clase Fracción.....	143
Figura 127.	Elementos UI del Canvas.....	145
Figura 128.	Inicializador de misiones.....	146
Figura 129.	Posibilidad de peso insuficiente.....	146
Figura 130.	Posibilidad de peso correcto.....	147
Figura 131.	Posibilidad de peso excedido.....	147
Figura 132.	Script Iniciar_Mision_Manzanas.....	148
Figura 133.	Script Recolectar Manzanas.....	151
Figura 134.	Script AllMissionStatus.....	153
Figura 135.	Configuración de calidad gráfica sección 1.....	157
Figura 136.	Configuración de calidad gráfica sección 2.	157
Figura 137.	Ventana Build Settings.....	158
Figura 138.	Escenas en construcción del proyecto.....	158
Figura 139.	Asignación del icono de la aplicación.....	159
Figura 140.	Configuración de resolución y presentación.....	160
Figura 141.	Nombre y ruta de almacenamiento de la construcción de la aplicación....	160
Figura 142.	Ejecutable de la aplicación.....	161

Introducción

En la actualidad la tecnología se ha convertido en una herramienta de gran importancia para el ser humano, siendo útil para muchas actividades cotidianas, desde el entretenimiento hasta las actividades laborales. Por ejemplo, en el área educativa a pesar de que los libros y apuntes constituyen el soporte de la enseñanza; las computadoras personales, los Smartphone, las tabletas electrónicas y el internet se están empleando para facilitar la docencia de los profesores y el aprendizaje de los estudiantes, incluyéndose de esta manera en el entorno educativo.

Otra de las herramientas que se está utilizando en el ámbito educativo en distintas áreas curriculares es la Realidad Virtual (RV). Definida como una tecnología que permite crear ambientes tridimensionales en los que es posible interactuar con cualquier objeto a través de múltiples canales sensoriales (por ejemplo: vista, oído y tacto).

Dicha tecnología ofrece una serie de ventajas pedagógicas importantes (Cubillo, Martín, Castro & Colmenar, 2014) como las siguientes:

- Experimentos o prácticas que no pueden ser realizados debido a los costes del equipamiento, a la relación entre el número de equipos disponibles y los alumnos matriculados
- La disponibilidad de las instalaciones, ya sea por espacio y/o tiempo
- La realización de experimentos complejos y peligrosos, en muchas ocasiones no son realizados debido a que pueden provocar lesiones en caso de que ocurra algún fallo, la RV proporciona la interacción con modelos virtuales en tiempo real y ver los resultados obtenidos en el mundo virtual.

- Permite la observación de experimentos o fenómenos que ocurren tras un largo periodo de tiempo (meses, años, décadas...etc.) en segundos como por ejemplo, las leyes de Mendel, aunque también nos permite el caso contrario facilitando la observación de aquello que transcurre en un instante.

Debido a que la RV ofrece ventajas pedagógicas en el proceso de enseñanza-aprendizaje para hacer este proceso más interesante y divertido, ya que puede generar una enorme gama de actividades e ideas para implementar teorías y prácticas o una combinación de ellas generando motivación en los estudiantes, convirtiéndose en un factor muy importante sobre todo en estudiantes que tienen problemas de atención (Kaufmann, & Schmalstieg, 2006).

En México son pocos los trabajos que se han creado con RV en el área de educación. En este sentido, se buscó un área curricular de oportunidad para desarrollar una aplicación didáctica de RV, a través del análisis de los resultados de la prueba ENLACE 2012 y 2013 que evalúa el dominio de los conocimientos adquiridos en español, matemáticas, ciencias e historia. Sin embargo, para delimitar el presente trabajo solo se tomaron en cuenta los resultados del área de matemáticas específicamente de tercer año de primaria.

Cabe señalar que la prueba se evalúa con base en el porcentaje de reactivos correctos de los alumnos, clasificados en cuatro niveles INSUFICIENTE, ELEMENTAL, BUENO Y EXCELENTE.

De manera general se encontró lo siguiente:

1. 38.8% de los estudiantes que realizaron la prueba de nivel primaria en el año 2013 obtuvieron un resultado de nivel “ELEMENTAL”.
2. Comparando los resultados obtenidos en el año 2013 con respecto a los años 2012 y 2011, se observa un mayor porcentaje de alumnos situados en la

clasificación de “EXCELENTE”, siendo 19.8%, 17.3% y 11% respectivamente (Enlace, 2013).

3. De los 8, 463, 445 alumnos que hicieron la prueba en 2013, el 12.4% (1, 049,467) se encuentran en el rango de "INSUFICIENTE", mientras que en 2012 el número de alumnos ubicados en este nivel fue del 13% (1, 100,248) de en este mismo nivel.
4. En la prueba del 2013 para el cálculo de los ‘Puntajes’ se excluyen los casos de los alumnos que contestaron menos del 50% de las preguntas (Enlace, 2013).

Analizando los resultados se puede mencionar que es evidente que el nivel de matemáticas nacional se encuentra en nivel básico de conocimientos. Con estos resultados es notable que al paso de los años existe un aumento en el nivel de conocimientos pero no es lo suficiente para indicar que es una mejora constante o gradual del nivel de logro.

Posteriormente para conocer los resultados más cercanos al entorno social en el que nos desenvolvemos día con día, se analizaron los resultados obtenidos por estudiantes de tercer grado de tres escuelas primarias del municipio de Ecatepec de Morelos en el Estado de México, encontrándose lo siguiente:

Con respecto a los resultados de la Escuela Primaria Justo Sierra, se puede observar en la tabla 1, que el 46.8% de los alumnos se encuentran en un nivel ELEMENTAL, el 12.9% en INSUFICIENTE y el 29% se halla en el nivel BUENO y solo el 11.3% se ubica en un nivel EXCELENTE.

De forma general en la tabla 1, se muestra que en el área de matemáticas los resultados se encuentran apenas por encima de la media (542 de 1000 puntos posibles).

Tabla 1. Resultados 2013 de la primaria Justo Sierra (Fuente: <http://goo.gl/8CYZKk>)

Tercer grado		Escuela: Justo Sierra											
Asignatura	Puntaje promedio 2013	Porcentaje de alumnos por nivel de logro											
	El 99.9% de los alumnos que presentaron la prueba obtuvieron resultados entre 200 y 800 puntos.	Insuficiente Necesita adquirir los conocimientos y desarrollar las habilidades de la asignatura evaluada.			Elemental Requiere fortalecer la mayoría de los conocimientos y desarrollar las habilidades de la asignatura evaluada.			Bueno Muestra un nivel de dominio adecuado de los conocimientos y posee las habilidades de la asignatura evaluada.			Excelente Posee un alto nivel de dominio de los conocimientos y las habilidades de la asignatura evaluada.		
		2011	2012	2013	2011	2012	2013	2011	2012	2013	2011	2012	2013
Español	532	7.5%	25.8%	9.6%	57.0%	40.3%	50.4%	31.8%	30.6%	38.4%	3.7%	3.2%	1.6%
Matemáticas	542	25.7%	20.3%	12.9%	45.0%	48.8%	46.8%	23.9%	62.0%	29.0%	5.5%	4.9%	11.3%
Formación cívica y ética	482			30.3%			32.8%			34.4%			2.5%

Mientras que en la Escuela Primaria Adolfo López Mateos (tabla 2), el 48.9% de los estudiantes se encuentran en el nivel ELEMENTAL y el 10.9% en INSUFICIENTE, sin embargo, los resultado muestran un 7.2% más alto en el nivel EXCELENTE y 17 puntos más en el promedio general que la Escuela Primaria Justo Sierra.

Tabla 2. Resultados 2013 de la primaria Adolfo López Mateos (Fuente: <http://goo.gl/VCPQxp>)

Tercer grado		Escuela: Adolfo López Mateos											
Asignatura	Puntaje promedio 2013	Porcentaje de alumnos por nivel de logro											
	El 99.9% de los alumnos que presentaron la prueba obtuvieron resultados entre 200 y 800 puntos.	Insuficiente Necesita adquirir los conocimientos y desarrollar las habilidades de la asignatura evaluada.			Elemental Requiere fortalecer la mayoría de los conocimientos y desarrollar las habilidades de la asignatura evaluada.			Bueno Muestra un nivel de dominio adecuado de los conocimientos y posee las habilidades de la asignatura evaluada.			Excelente Posee un alto nivel de dominio de los conocimientos y las habilidades de la asignatura evaluada.		
		2011	2012	2013	2011	2012	2013	2011	2012	2013	2011	2012	2013
Español	533	10.1%	12.2%	7.5%	48.3%	57.1%	60.2%	34.8%	27.6%	31.2%	6.7%	3.1%	1.1%
Matemáticas	559	22.8%	11.2%	10.9%	47.8%	61.2%	48.9%	23.9%	22.4%	21.7%	5.4%	5.1%	18.5%
Formación cívica y ética	560			45.7%			20.7%			31.5%			2.2%

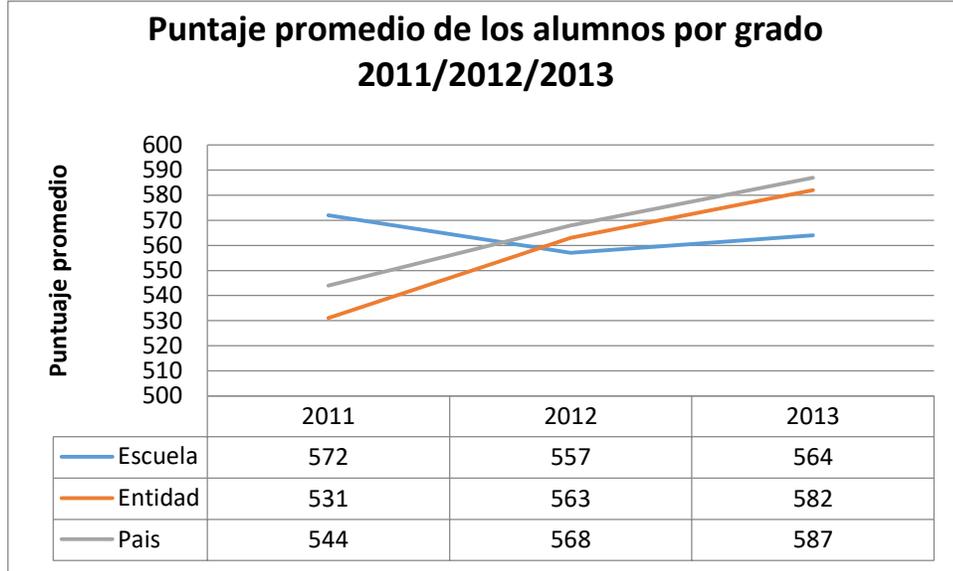
Los resultados de la Escuela Primaria Revolución (tabla 3) indican que el 46.5% de los estudiantes se encuentran en nivel ELEMENTAL, el 12.8% en INSUFICIENTE, resultados muy similares a los dos casos anteriores. Sin embargo, muestran un número mayor de alumnos en nivel EXCELENTE con 23.3%. Este último porcentaje muestra que más de sus estudiantes de tercer grado de primaria están mejor preparados. Y en cuanto a los resultados generales se encuentra por encima de la mitad de los puntos totales con 564 puntos y colocándose en un nivel de logro básico.

Tabla 3. Resultados 2013 de la primaria Revolución (Fuente: <http://goo.gl/LFWsCG>)

Tercer grado		Escuela: Revolución											
Asignatura	Puntaje promedio 2013	Porcentaje de alumnos por nivel de logro											
	El 99.9% de los alumnos que presentaron la prueba obtuvieron resultados entre 200 y 800 puntos.	Insuficiente Necesita adquirir los conocimientos y desarrollar las habilidades de la asignatura evaluada.			Elemental Requiere fortalecer la mayoría de los conocimientos y desarrollar las habilidades de la asignatura evaluada.			Bueno Muestra un nivel de dominio adecuado de los conocimientos y posee las habilidades de la asignatura evaluada.			Excelente Posee un alto nivel de dominio de los conocimientos y las habilidades de la asignatura evaluada.		
		2011	2012	2013	2011	2012	2013	2011	2012	2013	2011	2012	2013
Español	552	9.0%	16.5%	11.6%	36.0%	34.1%	39.5%	32.4%	34.1%	38.4%	22.5%	15.4%	10.5%
Matemáticas	564	10.8%	9.8%	12.8%	34.2%	46.7%	46.5%	31.5%	28.3%	17.4%	23.4%	15.2%	23.3%
Formación cívica y ética	488			37.2%			23.3%			32.6%			7.0%

Sin lugar a duda el nivel de conocimientos va en decadencia aunque la tendencia hasta el momento indique que el número de estudiantes con nivel excelente aumenta, el número de estudiantes con nivel insuficiente igual aumenta y de esta manera se va creando en conjunto un nivel bajo de conocimientos.

Graficando los resultados de los puntajes obtenido de la mejor escuela en este caso la Primaria Revolución con respecto a los resultados por entidad y país desde el año 2011 al 2013 (Gráfica 1), se puede observar que a nivel escuela el puntaje ha disminuido del año 2011 al año 2012 y aumentado 7 puntos al año 2013 mientras que a nivel entidad y nacional se mantiene un ritmo de aumento moderado.



Gráfica 1. Línea de tiempo nivel de logro (Fuente: <http://goo.gl/LFWsCG>)

Considerando que el ritmo de aumento en el puntaje a nivel escuela fuese constante, un nivel de logro aceptable sería conseguido en 34 años con un puntaje de 802 puntos, sin embargo, el aumento sería lento o posiblemente decaería a un nivel de logro más bajo que el actual.

Algo parecido sucede con la entidad, pues se encuentra en un nivel bajo en cuanto a matemáticas se refiere, ya que de los 1000 puntos posibles, solo se han obtenido 582 promediando todas las escuelas registradas en la SEP. De igual forma, el país se encuentra en condiciones que no rebasan los 600 puntos de logro colocando a México en un nivel de educación elemental.

Después de observar los resultados anteriores, se decidió analizar la información por grupos de una escuela para conocer los temas en específico que representan mayor dificultad para los estudiantes. Estos datos se encuentran organizados en reactivos y temas junto con la cantidad de alumnos que fallaron en contestar las preguntas relacionadas con cada tópico.

En las tablas 4 y 5, se muestran datos de la escuela primaria Aquiles Serdán y Lázaro Cárdenas, sobre los temas y reactivos de la prueba ENLACE del área de matemáticas en donde comenten más errores los estudiantes.

Tabla 4. Información sobre temas y reactivos de la prueba ENLACE del área de matemáticas de la escuela Aquiles Serdán (Fuente <http://goo.gl/exgAvb>).

Matemáticas	Reactivos contestados erróneamente por más del 60% del grupo	Reactivos contestados erróneamente entre el 60% y 40% del grupo	Reactivos contestados erróneamente por menos del 40% del grupo
Medida	50,75,96		
Números y sistemas de numeración	12,17,44,45,46,69,91	10,11,18,19,97,98	70,80
Problemas aditivos		71,92	
Problemas multiplicativos	16,21,23,48,93,94	15,20,49,72,73,74,95,99,100,101,102	13,14,22,47

Tabla 5. Información sobre temas y reactivos de la prueba ENLACE del área de matemáticas de la escuela Lázaro Cárdenas (Fuente: <http://goo.gl/4SRGUR>)

Matemáticas	Reactivos contestados erróneamente por más del 60% del grupo	Reactivos contestados erróneamente entre el 60% y 40% del grupo	Reactivos contestados erróneamente por menos del 40% del grupo
Medida	50,75,96		
Números y sistemas de numeración	11,12,19,44,45,46,69,81	10,17,18,19,97,98	90,98
Problemas aditivos		71,92	
Problemas multiplicativos	13,16,21,94,95,99	15,20,23,48,49,72,73,74,93,100,101,102	14,22,47

Según las estadísticas mostradas, los temas en los que se cometen más equivocaciones son: medidas, números y sistemas de numeración, y problemas multiplicativos, estando implícito en estos los temas de fracciones (Anexo 1). Los errores en estos temas son tan frecuentes que el 60% de los alumnos o más tienden a tener respuestas incorrectas cuando se les presenta un problema de ese tipo. En este sentido, se puede concluir que más de la mitad de los alumnos se encuentran en un nivel de logro bajo en los temas de fracciones.

Motivos por los cuales, el presente proyecto toma esta área de oportunidad para desarrollar una aplicación de RV enfocada al tema de fracciones, que se imparte durante el proceso de enseñanza – aprendizaje a los alumnos de tercer grado de primaria, en México.

El presente documento se encuentra estructurado por cuatro capítulos. En el Capítulo 1, se mencionan las bases teóricas de la RV. En el Capítulo 2, se describe la

metodología que fue utilizada durante el desarrollo de la aplicación de RV. En el Capítulo 3, se explica cómo fueron implementadas cada una de las fases de la metodología en el desarrollo de la aplicación de RV empezando con el análisis, donde se creó una perspectiva del contexto de la aplicación y un plan para cubrir las necesidades del caso planteado; seguida de la fase de diseño donde se definieron las características visuales y funcionales de la aplicación; posteriormente se presenta la fase de desarrollo, donde se menciona como fue el modelado de los recursos artísticos, recursos tecnológicos y las pruebas. Finalmente en la etapa de evaluación se propone aplicar a futuro un instrumento con la finalidad de conocer y analizar el nivel de aceptación y satisfacción que despierta este tipo de tecnología tanto para los profesores como para los estudiantes de tercer año de primaria. Y finalmente, se mencionan las conclusiones obtenidas.

Capítulo 1

Marco teórico

1.1 Realidad Virtual

En la actualidad la RV ha tomado bastante auge debido al bajo costo que ha tenido el despliegue de esta tecnología, lo que ha implicado que se implemente en distintas áreas como por ejemplo en el entretenimiento, la formación, la difusión, la capacitación, el diseño, la terapia y la educación.

1.1.1 Concepto de la RV

La RV es un campo de la visualización científica, compuesta por un conjunto de dispositivos electrónicos y objetos virtuales tridimensionales que reaccionan con la interacción del usuario de forma que se asemeje lo más posible a la realidad (Pérez, Zabre & Islas, 2004). Estos elementos en conjunto pueden generar dos tipos de RV: inmersiva y no inmersiva, ambas dando la posibilidad de simular y apoyar a las necesidades que resulten difíciles o poco convenientes el ser humano en la vida real.

1.1.2 Tipos de RV

Los tipos de RV se diferencian según el nivel de inmersión con el usuario: a) RV no inmersiva o de escritorio y b) La RV inmersiva.

1.1.2.1 RV no inmersiva

El usuario mediante una computadora de escritorio puede ver el espacio virtual y navegar en él a través del teclado y el ratón.

La realidad virtual de escritorio hace su uso más factible y sencillo al no utilizar dispositivos de alto costo.

Ejemplo de ésta son las aplicaciones que utilizan Internet y los videojuegos con objeto de interactuar en tiempo real con diferentes personas en espacios y ambientes tridimensionales (Ramos, 2003).



Figura 1. RV no Inmersiva (Fuente: <https://4tololo.ru/content/13483>).

1.1.2.2 RV inmersiva

Este tipo de RV ofrece inmersión total al usuario y pone en tentativa sus habilidades mediante el uso de diferentes equipos de interacción como los sensores, guantes, visores de RV, diademas e incluso CAVES (Zhan, 2011).

Un ejemplo de ellos son los videojuegos, que recrean un mundo virtual donde el ambiente es impredecible, percibiendo a través de los estímulos sensoriales una sensación de formar parte del entorno en el que se encuentra virtualmente.

La interacción que proporciona es mayor a la RV no inmersiva y se tiene la oportunidad de transformar el mundo virtual y realizar tareas que en el mundo real es imposible de efectuar.



Figura 2. RV inmersiva (Fuente: <https://www.emaze.com/@AFLQFCZR/VIDEOJUEGOS>).

1.1.3 Componentes de la RV

La RV tiene cuatro elementos indispensables para que el usuario experimente un buen acercamiento a ella. Los elementos son los siguientes:

- Mundo Virtual
- Inmersión
- Interacción
- Retroalimentación

1.1.3.1 Mundo Virtual

El mundo virtual es una representación digital del mundo real almacenada en la memoria de un ordenador, por medio de objetos tridimensionales (personajes, terrenos, casas, arboles, carros, etc.) que constituyen información espacial.

Este elemento debe tener un comportamiento interactivo con el usuario y que reaccione a sus acciones efectuadas en tiempo real. Con estos comportamientos el usuario obtiene una inmersión en el mundo virtual.

Los mundos virtuales tienen grandes alcances, ya que pueden ser implementados para ambientes de cualquier índole como, entrenamientos de vuelo, intervenciones quirúrgicas, terapias psicológicas, apoyo a la educación, etc.

1.1.3.2 Inmersión

La palabra inmersión puede tener muchos significados según el contexto en el que se aplique. Pero en la RV se puede definir como el conjunto de estímulos auditivos, visuales, táctiles, olfativos manipulados con una gran variedad de tecnologías de punta que hacen creer al usuario que se encuentra realmente en el mundo virtual, olvidándose del espacio físico donde se encuentra.

1.1.3.3 Retroalimentación

La retroalimentación es un elemento muy importante en la RV porque involucra todas las reacciones correspondientes a las acciones del usuario inmerso en el mundo virtual. Estas reacciones son simulaciones del mundo real por ejemplo, tocar algún objeto tridimensional que emita un sonido pregrabado como el de un platillo de una batería o las cuerdas de una guitarra. Estas reacciones se suman a la impresión generada en las personas de que realmente se encuentra en otro espacio físico.

1.1.3.4 Interactividad

La retroalimentación anteriormente descrita es el efecto de la interacción del usuario con el mundo virtual en tiempo real. Esta interacción se puede describir como las decisiones que toma el usuario mientras navega virtualmente en los escenarios tridimensionales. Un ejemplo de la interacción es la acción de prender un foco, el usuario tendría que presionar un interruptor en el mundo virtual como lo haría en el mundo real para encender la luz.

1.2 Características de software enfocado a la educación

Con base en las consultas realizadas en diferentes bases de datos de artículos científicos que proporciona nuestra casa de estudios, se extrajeron datos relevantes de diferentes trabajos de investigación sobre el desarrollo de software enfocado a la educación como realidad virtual, videojuegos, y programas de ordenador, generando así un análisis sistémico del año 2005 al 2014. La tabla 6 resume la información obtenida:

Tabla 6. Características de aplicaciones de software enfocadas a la educación.

Característica de software enfocado a la educación	Shi, Y. L. 2014,	Hui-Zhen, R., & Zong-Fa, L. 2013	Rahimi, A., Golshan, N. & Mohebi, H. 2013	Aymerich-Franch L. 2012	Ibanez, M., et al. 2011	Zhan, Z. 2011	Wang, G. & Wu, H. 2008	Hannes, K. & Schmalstieg, D. 2006	Dickey, M. 2005	Garrido, J. 2013	Fernández, M. 2011	Gramigna, A. González, J. 2009	González, S., & Blanco, F. 2008	Rodríguez, W. & Pallares M. 2006	Morejón, S. 2011	A. Collins & R. Halversont 2009	Espinoza R. 2009	Caro, M., et. al. 2009	Organista-Sandoval, J. 2010
	Realidad Virtual									Videojuegos					Software /Programas				
Inmersivo, el sistema deberá tener la capacidad de estimular los sentidos del usuario, para que se sienta parte del ambiente y asumiendo el protagonismo que se proyecta.	X	X	X	X	X	X	X	X	X	X									

El usuario podrá interactuar con el ambiente y los objetos de forma igual o parecida al mundo real.	X	X	X		X	X		X		X									
El contenido de los escenarios se debe presentar de manera estructurada.	X	X	X		X	X	X	X											
Los escenarios con situaciones realistas y/o atractivas, sean estos intuitivos, dinámicos y sorprendidos.	X	X	X	X	X		X	X	X			X							
Limpieza de los escenarios, cuando un objeto ya no sea utilizado o necesario para realizar cierta actividad, tendrá que ser eliminado del entorno.							X												
Los estímulos al usuario, se realizan a través de la obtención de recompensas (objetos) o habilidades (poderes).		X					X					X							
Los personajes representados en primera persona presentan mayor percepción del usuario.	X					X		X											
Los objetos que se encuentren dentro de los escenarios y con los cuales tendrá que interactuar el usuario serán representados en 3D, lo que permitirá una mejor percepción por parte de los estudiantes (debido a que los objetos se pueden manipular directamente).	X		X	X	X	X		X	X			X							X
Los avatares en tercera persona tienen la posibilidad de ser personalizados y representan una experiencia más simbólica.					X			X				X							
La comunicación con el usuario puede ser a través de mensajes y respuestas de textos y audio.								X	X										
Emplear un lenguaje adecuado dependiendo del tema.																			X
Espacios crecientes de sorpresas, lo que implica situaciones (niveles progresivos y graduales) de diferente dificultad.	X							X	X			X	X						
Que contenga muchas actividades para que el usuario decida qué es lo que quiere hacer (auto-admitirse ritmos y tiempos).	X			X	X				X			X							
Que exista la posibilidad de equivocarse y volver a intentarlo, sin que esto sea percibido como una actividad obligatoria, más bien que se presente como un desafío (retos)									X			X							
Mostrar resultados al finalizar cada actividad o módulo para ver el avance que se ha tenido.	X		X						X										
Tener en cuenta si se usará algún tipo de guía (ayuda académica) o será con los conocimientos adquiridos.					X					X		X					X	X	
Retroalimentación, evaluación o diagnósticos, que permitan a los	X		X			X	X		X	X							X	X	

- El contenido del software debe resultar familiar, estructurado y ordenado para que el aprendizaje sea el más óptimo posible y así mismo debe ser acorde a la realidad según sea el caso de estudio.
- El software educativo puede representar fenómenos que no sean posibles experimentar en el aula de clases.
- El software en primera persona genera una mayor percepción para el usuario que los de tercera persona.
- La comunicación con el usuario debe contener un lenguaje apropiado para el tema en cuestión a través de audio, textos y/o imágenes, por ejemplo, al inicio de cada actividad plantear el propósito que se espera cumplir.
- El software debe incluir muchas actividades, permitiendo al usuario decidir cuales realizar y en el orden que él desee. También debe permitirle equivocarse y mostrarle retroalimentación (ejemplos, imágenes, videos cortos, etc.) para que sienta que es un reto en lugar de una actividad obligatoria.
- Debe proveer estímulos para motivar el aprendizaje como recompensas o poderes por completar las actividades, teniendo estas un incremento gradual en el nivel de dificultad. Así mismo, debe mostrar estadísticas para verificar el grado de avance.

1.3 Condiciones en donde se puede aplicar la RV

A continuación se enlistan los casos en que se puede utilizar la tecnología de RV para desarrollar software educativo según Pantelidis (1996).

Cuando:

- La enseñanza o el entrenamiento en el mundo real puede ser peligrosa, imposible o inconveniente.

- Pueden suceder errores significativos por parte del alumno o aprendiz en el mundo real. Estos errores pueden ser, devastadores o desmoralizadores para el alumno, perjudiciales para el ambiente, o costosos.
- Para fortalecer el aprendizaje de los estos temas el profesor debe mostrar ejemplos referentes a cada tema, estos pueden ser aplicados con RV.
- La interacción con el modelo es igual o más motivadora que la interacción con la situación real.
- La realización de una clase atractiva requiere viajes, dinero y/o logística.
- Se desea lograr experiencias compartidas en un grupo.
- Se desea crear un entorno simulado para lograr los objetivos de aprendizaje.
- Es necesario hacer perceptible lo imperceptible.
- Quieren desarrollarse entornos participativos y de actividades que solo pueden ser generados por ordenador.
- Las tareas a enseñar requieren destrezas manuales o movimientos físicos.
- Es esencial hacer el aprendizaje más interesante y divertido. La motivación es un factor muy importante, sobre todo en estudiantes que tienen problemas de atención.
- Es necesario proporcionar a una persona discapacitada la oportunidad de realizar experimentos y actividades que de otra manera no podría realizar.

1.4 Aplicaciones de la RV

La RV tiene sus orígenes décadas atrás en diferentes áreas del conocimiento con el empleo de simulaciones por computadora por ejemplo, en los años 50's fue desarrollado el "Sensorama" por Morton Heiling con el que estimulaba la vista, el olfato, el tacto y el sentido auditivo. Otro de los antecedentes más notables es el cine ya que siempre ha intentado crear formatos de imagen y audio que hagan sentir al espectador parte de la escena (Olguín, Rivera & Hernández, 2006).

A continuación se describe como ha intervenido la RV en la actualidad en diferentes áreas.

- Aplicaciones médicas

Los sistemas de RV en la medicina son utilizados para el entrenamiento de intervenciones quirúrgicas que tienen un alto grado de dificultad y que son muy riesgosas para el paciente en la vida real.



Figura 3. Aplicación médica de la RV (Fuente: <https://redef.com/author/529b6b12a30191a24400017b>).

- Aplicaciones militares

En la milicia se destacan las aplicaciones de simulaciones de vuelo y manejo para entrenar a los soldados, reduciendo o eliminando la pérdida de vidas.



Figura 4. Entrenamiento de vuelo con RV (Fuente: http://www.allprojectors.ru/ap_module/content/article/3013/11183).

- Videojuegos

En el mundo de los videojuegos la realidad virtual se usa mayormente para el entretenimiento. Los sistemas de inmersión utilizados en este ámbito han evolucionado, implementando nuevas tecnologías como lo son sistemas de reconocimiento de movimientos corporales y de voz, el uso de vibradores en los mandos, entre otros, para obtener un grado de inmersión mayor.



Figura 5. Videojuegos con RV (Fuente: http://www.elspanol.com/videojuegos/20160316/109989024_0.html).

- Aplicaciones académicas

En la educación el uso de la RV se emplea para la enseñanza de diferentes asignaturas, por ejemplo en la materia de física se representan simulaciones de fenómenos naturales que son riesgosos o difíciles de comprender o asociar los conceptos del tema en cuestión. Además, ofrece la generación de un mayor interés en los alumnos para aprender.



Figura 6. Sistema de RV para la representación de caída libre.

- Aplicaciones psicológicas

En esta rama existen entornos tridimensionales para tratar muchos tipos de fobias como lo son la aracnofobia, claustrofobia, etc. De igual manera ayudan a superar trastornos psicológicos por abusos de cualquier tipo sin poner en riesgo la salud o integridad de los pacientes.

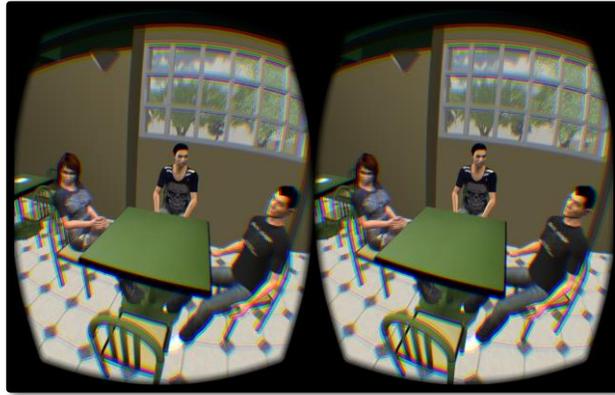


Figura 7. Sistema de RV para entrenamiento de conductas asertivas.

1.5 Juegos Serios

Según Iglesias (2011), no existe una taxonomía universal aceptada para los videojuegos, sin embargo, se han tratado de clasificar tomando en cuenta distintos factores como:

- **El tema:** Se refiere a la ambientación, la historia, y el contenido cultural del juego.
- **La audiencia:** Se refiere al público al cual está destinado el juego
- **El propósito:** Se refiere al propósito de los creadores del juego. Este propósito puede ser solamente entretener, obtener dinero, resolver un problema, o transmitir un mensaje entre otros. Los juegos con el propósito de resolver un problema son llamados juegos serios, e incluyen desde juegos para publicidad hasta juegos educativos.

Los juegos serios son aquellos juegos que intentan lograr un fin más allá del entretenimiento. Se definen formalmente como una aplicación de software interactiva que (Alvarez, 2008):

- Tiene por objetivo proponer un desafío.
- Es divertido de jugar o ejecutar.
- Incorpora conceptos de puntajes y medición de resultados.
- Le otorga al usuario habilidades, conocimientos o actitudes que pueden ser aplicadas en el mundo real.

Este tipo de juegos pueden ser utilizados para diversos fines tales como (Alvarez & Rampnoux, 2007):

- **Edutainment** (Entretenimiento educativo). Transmiten el conocimiento a través de un enfoque lúdico.
- **Advergaming** (Publicidad y juego). Es la práctica de usar videojuegos para publicitar una marca, producto, organización o idea.
- **Edumarket games** (Educación y publicidad). Su enfoque está en enseñar principalmente al público infantil una perspectiva de las labores sociales por ejemplo, ser policía o vendedor de helados, aunado a la publicidad de la organización policiaca o la marca de helados.
- **Political games** (Política). Simulan una situación social en la cual se resuelven problemas que involucran la justicia.
- **Training and simulation games** (Entrenamiento y simulación). Son videojuegos que representan situaciones del mundo real en un mundo virtual. El propósito de este tipo de juegos no es ganar, sino simplemente divertirse o alcanzar algunos objetivos y de este modo obtener experiencia que resulta significativa en la vida real.

1.6 Recursos artísticos y tecnológicos

Los recursos artísticos son el conjunto de texturas, modelos tridimensionales o bidimensionales, animaciones, secuencias de audio y video (Watkins, 2011).

Mientras que los recursos tecnológicos son los scripts. A continuación se mencionan algunos ejemplos de cada uno de ellos:

- **Modelos** (sillas, mesas, cubiertos, alimentos, personajes, planos, estructuras arquitectónicas, etc.).
- **Secuencias de audio y video** (sonidos ambientales, música, cinemáticas, conversaciones de interacción y contextuales).
- **Texturas** (piedra, metales, corteza de árbol, letreros, etc.).
- **Animaciones** (animaciones corporales, faciales, transformaciones, etc.).
- **Scripts** (Archivos que contienen códigos que dan instrucciones al ordenador para realizar una acción en el software).

Al integrarse dentro de un motor de representación gráfica funcionan de forma conjunta para conseguir al final una simulación de un ambiente real.

1.7 Técnicas de modelado 3D

Son un conjunto de procedimientos utilizados durante la producción de objetos, estructuras o terrenos tridimensionales mediante un programa de computadora especializado.

En la actualidad existe una variedad de técnicas que han sido útiles para casos específicos y otras que han sido más factibles para la gran mayoría de casos, por ejemplo, para la producción de películas animadas o videojuegos.

De igual manera para el desarrollo de sistemas con RV se requiere que los recursos artísticos tengan el menor número de polígonos posibles, ya que a menor número de polígonos el consumo de recursos va a ser menor y el rendimiento del equipo de cómputo resulta más eficiente. Las técnicas que más destacan se describen a continuación:

1.7.1 Modelado de caja

El modelado de caja o también conocido como “*Box Modeling*”, consiste en comenzar con una primitiva típicamente un cubo o una esfera que el artista va modificando según la forma deseada. Lo anterior se puede lograr con ayuda de las diversas herramientas y opciones que incluyen los programas de modelado 3D, como por ejemplo, la extrusión y el uso del escalado, rotación y traslación de caras, bordes y vértices.

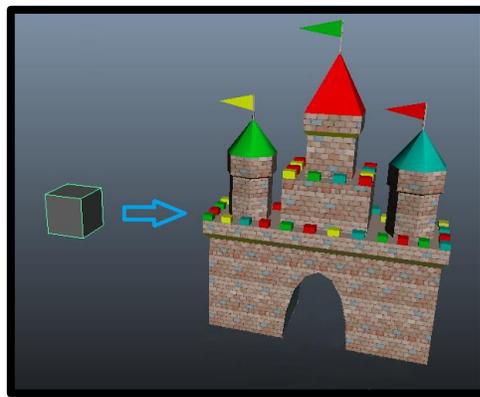


Figura 8. Modelado de caja.

1.7.2 Modelado escultórico

Esta técnica consiste en comenzar con una primitiva al igual que el Box Modeling pero la diferencia es que se utilizan herramientas que simulan una presión sobre el objeto produciendo más geometría, es algo similar a trabajar con arcilla. Con esta técnica se obtienen asombrosos modelos pero la desventaja es que no es conveniente utilizarla en proyectos que se ejecutan en tiempo real, debido a que el nivel de detalle y el consumo de recursos es alto.



Figura 9. Modelado escultórico (Fuente: <https://mangojambo.wordpress.com/page/2/>).

1.7.3 Modelado con superficies y curvas NURBS

Non-uniform rational B-spline (NURBS) por sus siglas en inglés, es una técnica práctica porque es más automatizada que un modelado poligonal. Permite generar modelos tridimensionales con más precisión y detalle como por ejemplo, automóviles y aviones. Sin embargo, no es útil para hacer renderización en tiempo real.

Cabe señalar que si se modelan objetos 3D con esta técnica y los requerimientos del proyecto requieren modelos poligonales, con ayuda del software de modelado se puede hacer la conversión, no obstante el número de geometría aumenta (Autodesk Maya, 2017).

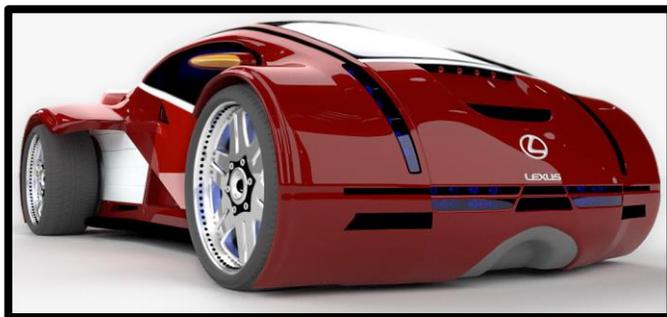


Figura 10. Modelado con NURBS (Fuente: http://descargas.pntic.mec.es/mentor/visitas/DemoModeladoBlender/111_tipos_de_modelado.html).

1.7.4 Modelado a través de displacement maps y height maps (terrenos)

Es una técnica en la que se utilizan texturas en escalas de grises para crear relieves o profundidades como escenarios con montañas, valles, lagos, etcétera añadiendo geometría adicional en los objetos, generalmente utilizada para hacer renderización de aspecto realista.

Cabe señalar que las texturas en esta técnica generan geometría a diferencia del bump mapping que solo crea la ilusión, esta última es más recomendable para proyectos que se ejecutan en tiempo real (Maya, 2017).

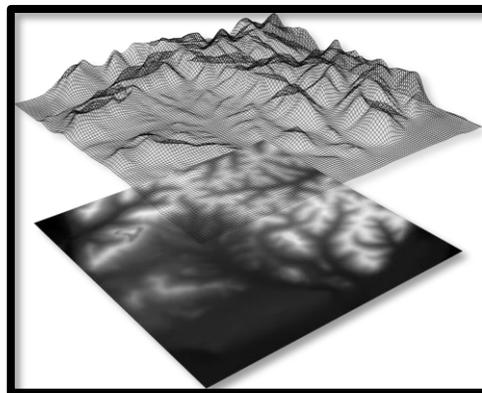


Figura 11. Modelado con height maps (Fuente: <http://blog.freemachines-project.com/tag/heightmap/>).

1.7.5. Modelado por modificadores

Es una técnica utilizada para moldear y dar más detalle a la forma de los objetos a través de los siguientes modificadores que se encuentran incluidos en las herramientas de los programas de modelado 3D:

- 1) *Noise*
- 2) *Extrusión*
- 3) *Lattice*

- 4) *Twist*
- 5) *Ripple*
- 6) *Displace*

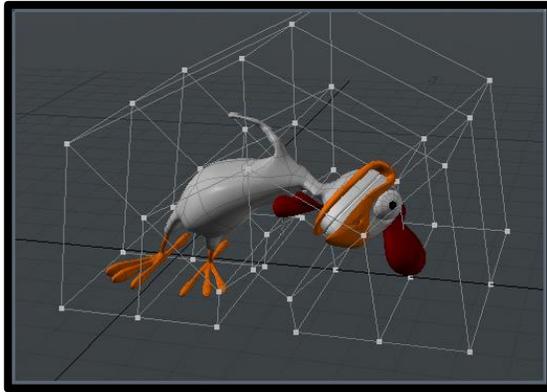


Figura 12. Deformador Lattice (Fuente: <http://mymodo.ru/lesson/slide-hinge-constraintogranichenie-skolzheniya>).

1.7.6. Modelado por clonación

Esta técnica consiste en realizar copias de objetos y duplicarlos para generar contenido 3D como: cercas, rocas, ladrillos, etc. Existen diferentes tipos: instancia, parent, copy.

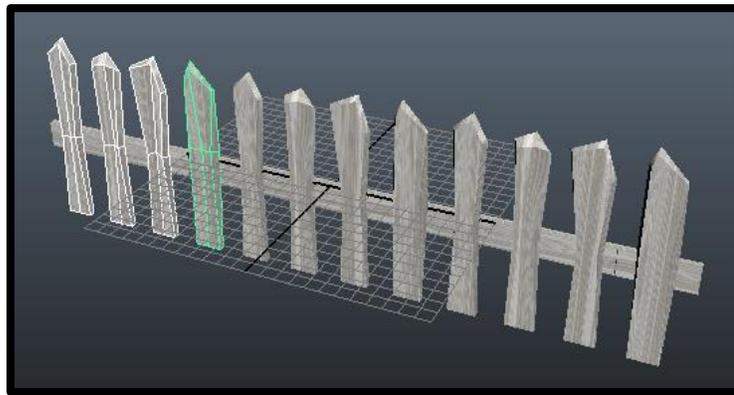


Figura 13. Modelado por clonación.

1.7.7. Modelado booleanico

En muchos casos esta técnica es aplicada para facilitar y agilizar el modelado mediante la utilización de operaciones booleanas, aunque se debe tener cuidado ya que su uso puede causar geometría no adecuada para el renderizado o para el motor gráfico de representación. (Maya, 2017). Existen tres operaciones booleanas (Figura 14).

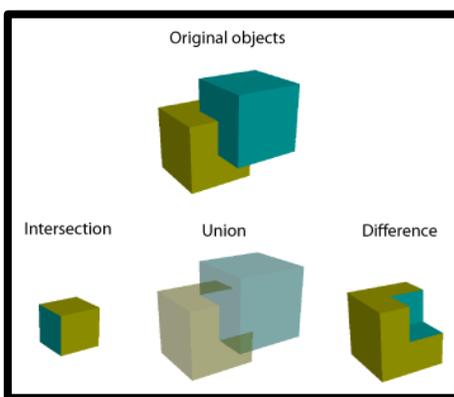


Figura 14. Operaciones booleanas (Fuente: <https://therese3d.com/2014/11/09/modelling-process-diary/>).

1.7.8. Modelado con blueprints

Es una técnica rápida y eficiente, en la cual se utilizan plantillas (blueprints) que corresponden a una serie de dibujos de un mismo objeto o personaje en diferentes perspectivas (vistas frontal, lateral y superior), usadas como base para el modelado. En esta técnica se inicia con una primitiva a la cual se le aplican modificadores para obtener el resultado final (Alvarado, Barría, Cordella & Lechuga, 2007).

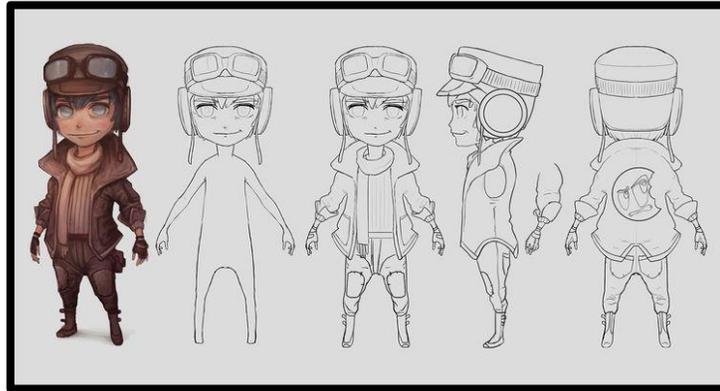


Figura 15. Modelado con blueprints (Fuente: <https://es.pinterest.com/pin/309622543111301712/>).

1.7.9. Modelado de letras y números

Es una técnica de modelado que permite generar textos a partir de curvas con algunas herramientas de los programas de modelado. Esta técnica es fácil de usar y rápida ya que solo se introduce el texto en la herramienta y el programa de modelado realiza los cálculos para dar grosor al texto.



Figura 16. Modelado de letras y números.

1.7.10. Modelado simétrico

Esta técnica es utilizada para generar geometría simétrica con la ayuda de herramientas que proporcionan los programas de modelado 3D, que realizan la función de espejo. Un ejemplo de su uso es el desarrollo de personajes (Figura 17), en donde se modela únicamente un lado del personaje y la otra mitad se genera de forma automática e idéntica.

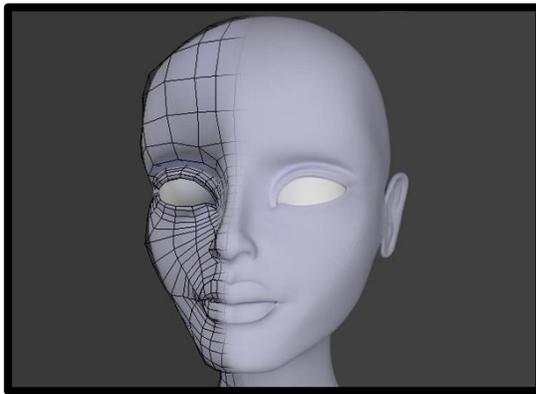


Figura 17. Modelado simétrico (Fuente: <https://es.pinterest.com/pin/523754631644602699/>).

1.8 Herramientas de modelado 3D

Estas herramientas son programas de ordenador especializados para el desarrollo de objetos y/o estructuras tridimensionales que permiten una amplia representación gráfica en la producción de estos. Algunas de estas herramientas son: 3DStudio Max, Autodesk Maya, Blender, AutoCAD, SketchUp, entre otros. A continuación se mencionan algunas herramientas destacadas con una breve descripción de sus principales características.

1.8.1 Autodesk Maya

Es una herramienta para crear contenido digital 3D como personajes, animaciones, efectos visuales, videojuegos y simulaciones (Maya, 2017).

En Autodesk Maya se pueden realizar diversas tareas para producir contenido como:

1. Crear modelos 3D
2. Colocación de huesos a personajes.
3. Animaciones
4. Dinámicas
5. Pintar y efectos de pintura
6. Luz, sombras y renderizado



Figura 18. Autodesk Maya (Fuente: <http://blbhome.blogspot.mx/2014/05/autodesk-motionbuilder-2015-free.html>).

1.8.2 Blender

Blender es una aplicación multiplataforma de código abierto para la creación 3D que ha sido probada en distintos sistemas operativos como Linux, Windows y Macintosh utilizando como interfaz OpenGL y además brinda la posibilidad de realizar diversas tareas como: modelar, colocar huesos, animar, simular, renderizar, componer

imágenes, seguimiento de movimientos con la cámara, edición de video y creación de videojuegos (Blender, 2017).

Cabe resaltar que esta herramienta está siendo desarrollada por cientos de personas al rededor del mundo incluyendo animadores, artistas modeladores, científicos y personas que lo hacen por entretenimiento, ya que tiene la ventaja de contar con un API que permite mejorar la aplicación y escribir herramientas más especializadas.



Figura 19. Blender (Fuente: http://madinggt.blogspot.mx/p/blog-page_12.html).

1.8.3 DAZ Studio

Es un programa producido por la empresa DAZ 3D, para el desarrollo de arte tridimensional, animaciones y como una herramienta de renderizado. Este programa utiliza la plataforma “Genesis” que brinda soporte a diferentes tipos de archivos de modelos tridimensionales, además de ser compatible con otros programas de modelado 3D, como: Autodesk Maya, 3D Studio Max y Poser.

La empresa proporciona la versión Pro en su página de internet, la cual se obtiene después de un registro. Además cuenta con una tienda en línea, donde se pueden obtener modelos de forma gratuita y muchos más que se pueden comprar.



Figura 20. DAZ Studio.

1.9 Herramientas para generar y editar texturas

Actualmente en el mercado del diseño 3D existen muchas aplicaciones para realizar texturas de buena calidad. Cada aplicación tiene ventajas y desventajas sobre otras como es el caso de las siguientes:

1.9.1 Gimp

GIMP es un acrónimo de GNU Image Manipulation Program. Es un programa de distribución gratuita y de código abierto para tareas como retoque fotográfico, composición y creación de imágenes.

Se encuentra disponible para varios sistemas operativos como GNU/Linux; OS X, Windows, entre otros. Además, es una aplicación que puede ser personalizada debido a que cuenta con la adaptación de diversos plugins que permiten realizar un trabajo más sofisticado y de mejor calidad (GIMP, 2017).

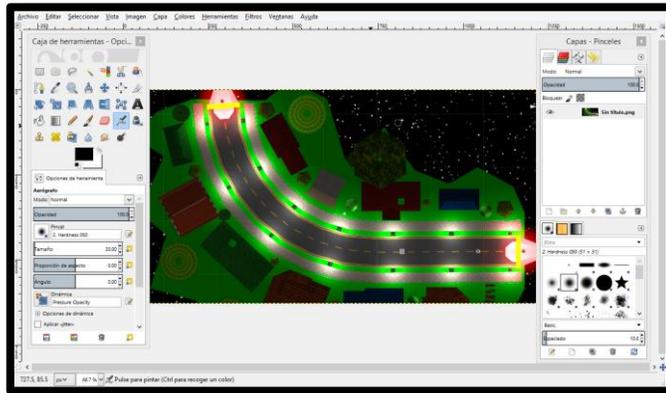


Figura 21. Gimp

1.9.2 Crazy Bump

Es una potente herramienta para la creación de mapas de normales a partir de una imagen de algún material por ejemplo rocas (textura), con un costo actual de 49 dólares la licencia para estudiantes.

Considerada como una herramienta muy práctica que permite ahorrar tiempo y esfuerzo para los texturizadores, ya que su interfaz provee controles para modificar los valores de los parámetros del mapa de normales hasta que se obtiene el efecto deseado.

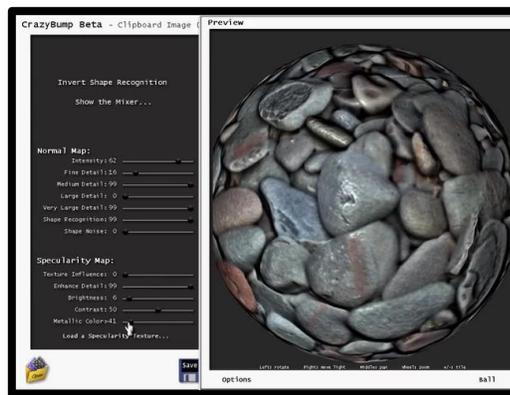


Figura 22. Crazy Bump (Fuente: <http://www.crazybump.com/>).

1.10 Motores de representación gráfica 3D

Hoy en día el crecimiento en la implementación de la RV ha traído consigo una gran variedad de motores gráficos que han ido evolucionando y mejorando con el paso del tiempo. A continuación se describen algunos que destacan en la industria.

1.10.1 Unity 3D

Es un motor de videojuegos multiplataforma líder para Mac OSX y Windows con las siguientes características:

- a) Unity ofrece licencias para todo tipo de desarrolladores que van desde la gratuita (uso personal) hasta la empresarial, los precios mensuales aproximados actualmente son 35 dólares la plus, 125 dólares la profesional y la versión empresarial tiene un costo que está sujeto a las necesidades de la empresa. Unity no cobra regalías en ninguna de sus licencias, permitiendo recaudar \$100,000 pesos en su versión personal, \$200,000 en la plus y en la versión Pro y empresarial son ilimitados.
- b) Su licencia gratuita a pesar de tener restricciones sigue siendo muy potente para el desarrollo de productos académicos.
- c) Tiene una interfaz sencilla, fácil de usar y personalizable por sus usuarios.
- d) Permite desarrollar productos para veintisiete plataformas, por ejemplo: iOS, Android, Windows, Linux, PlayStation, Wii U, Facebook, Xbox, etc.
- e) Provee la capacidad de crear proyectos de RV, videojuegos, simulaciones, etc.
- f) Soporta diversos lenguajes de programación como C#, JavaScript.
- g) Es compatible con modelos tridimensionales desarrollados en 3D Studio Max, Autodesk Maya, Blender, Cinema 4D y Cheetah3D, SketchUp y puede leer

archivos con extensiones .FBX (Film Box), .dae (Collada), .3ds, .dxf, .obj, y .skp.



Figura 23. Unity 3D.

1.10.2 Unreal Engine

Es una suite de herramientas para el desarrollo de videojuegos para Mac OSX y Windows. Actualmente es uno de los motores de videojuegos más potentes en el mercado ya que está diseñado y es distribuido en estudiantes, desarrolladores independientes y profesionales. Cuenta con las siguientes características:

1. Se pueden crear desde videojuegos en 2D hasta aplicaciones de RV y videojuegos de gama alta en dispositivos móviles con iOS y Android, consolas como PlayStation, Xbox y en PC's.
2. Permite crear renderización de alta calidad de construcciones arquitectónicas, simulaciones, películas digitales, etc.
3. Es un software que ofrece licencias de forma gratuita.
4. Se deben ofrecer regalías del 5% de los ingresos al superar los 3,000 dólares.
5. Ofrece compatibilidad para los modelos 3D desarrollados en Maya, Blender y 3D Studio Max con extensión .FBX, así como soporte para los lenguajes de programación C++ y Boo (Unreal Engine, 2017).

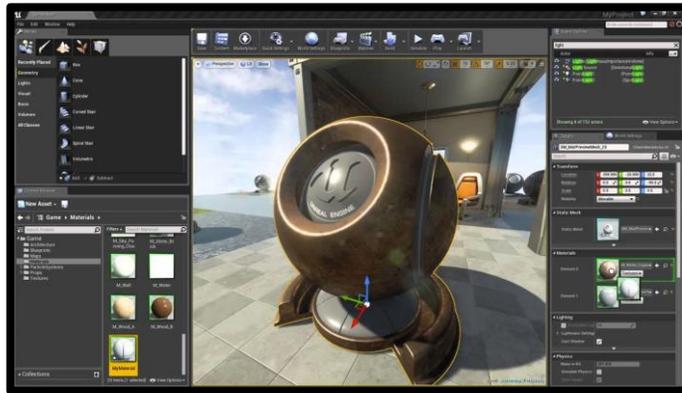


Figura 24. Unreal Engine (Fuente: <https://es.pinterest.com/intvoid/unreal-engine-4-tutorials/>).

1.10.3 Cry Engine

Es un motor de videojuegos que va creciendo en el mercado junto a otros motores, desarrollado por la empresa “CryTek”, cuenta con las siguientes características:

1. Permite crear aplicaciones como videojuegos, entornos de RV, simulaciones, etc.
2. Ofrece licencias para el público en general de forma gratuita, cabe resaltar que los usuarios no están obligados a ofrecer regalías por sus creaciones como con el motor Unreal Engine.
3. Actualmente tiene soporte para las siguientes plataformas: Xbox, Oculus Rift, Windows, PlayStation y Linux.
4. Permite la integración de modelos 3D creados con Autodesk Maya, Blender, 3D Studio Max y CryBlend con extensión .FBX (CryEngine, 2017).

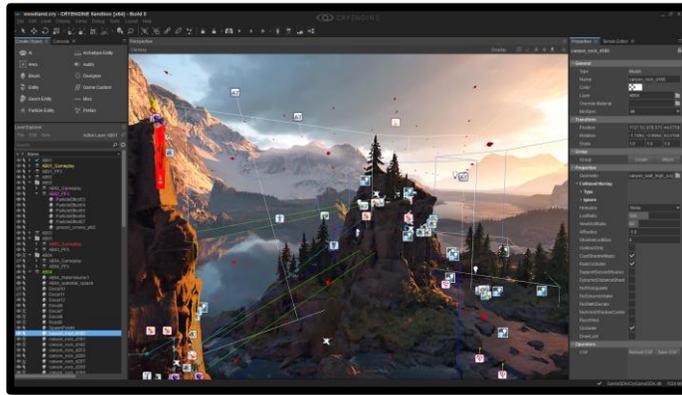


Figura 25. Cry Engine (Fuente: <http://www.3dvirtualand.com/cryengine5>).

1.11 La RV como recurso de enseñanza-aprendizaje

Numerosas aplicaciones educativas se basan en la tecnología de RV y emplean la teoría de que un conocimiento se retiene mucho mejor cuando se experimenta directamente que cuando simplemente se ve o se escucha.

La base de esta teoría es el concepto del conocimiento en primera persona, según en la cual un individuo adquiere la mayoría de los conocimientos de su vida diaria mediante experiencias que se resuelven de manera subjetiva. Al contrario de las técnicas educativas tradicionales, en las cuales se pone una perspectiva en tercera persona. El conocimiento se transmite a través del maestro resultando indirecto, frecuentemente se abusa del método expositivo, donde el alumno solo se convierte en un ser receptivo y pasivo, llegando a limitar su aprendizaje a un ejercicio de reproducción e ignorando su motivación.

En este sentido, la RV presenta una de sus mayores ventajas, al mantener al estudiante motivado e interesado a seguir aprendiendo y adquirir experiencias que junto con ellas se obtiene el conocimiento (Zapatero, 2011). Por ejemplo: el usuario al estar inmerso en un escenario interactivo en el cual puede desenvolverse con actividades didácticas, le ayuda a crear su propio conocimiento y experiencias. Con el

cual podrá explorar y moverse libremente en un entorno controlable, realístico, repleto de recursos y emociones, los alumnos adquieren conocimiento conceptual y práctico, esto motiva: la reflexión, la imaginación, cooperación e investigación en ellos, además ayuda a la percepción de una educación más humana, en comparación del uso de fríos libros de texto y máquinas (Shi, 2014).

Tabla 7. Porcentajes de aprendizaje a través de los sentidos. (Ferres, 1992. Cit. En Zapatero, 2011)

¿Cómo aprendemos?	
Gusto	1%
Tacto	1.5%
Olfato	3.5%
Oído	11%
Vista	83%

En la tabla 7, se puede observar que el 83% del aprendizaje se obtiene a través de la vista, con lo que se puede decir que la RV apoya en gran medida la enseñanza-aprendizaje ya que hasta el momento se ha especializado más en el sentido de la vista, tacto y oído, esto no quiere decir que el sentido del olfato y gusto no se implementan ya que actualmente están en proceso de desarrollo.

Capítulo 2

Metodología de desarrollo

2.1 Metodología

A continuación se describen las fases y actividades correspondientes a la metodología de desarrollo evolutivo para escenarios tridimensionales, que será utilizada para la creación del presente proyecto (Morales, et al., 2015):

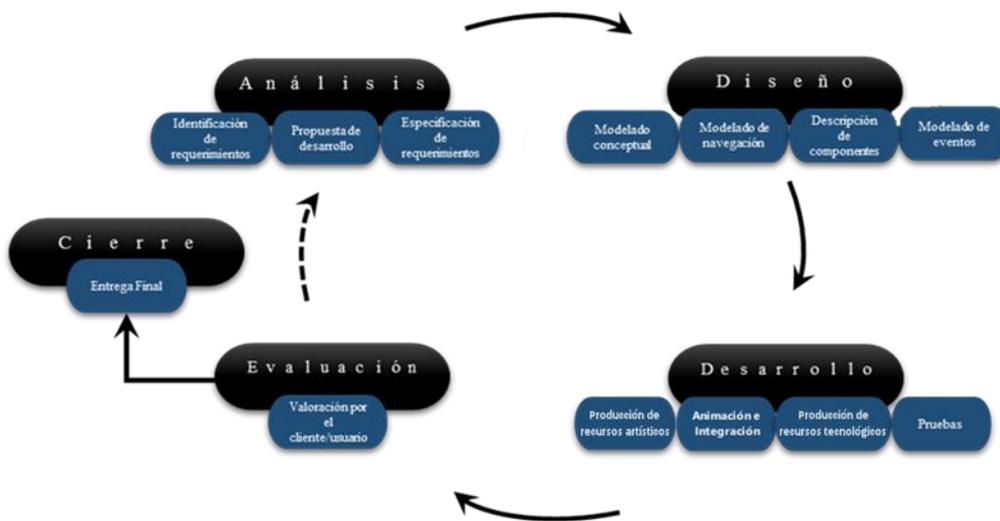


Diagrama 1. Metodología de desarrollo evolutivo para escenarios tridimensionales

2.2 Fase de análisis

La finalidad de esta fase, es tener una perspectiva clara del contexto en la cual se va enfocar la aplicación educativa. Para lo cual, es necesario definir y comprender las necesidades del cliente a través de tres actividades: identificación de requerimientos, propuesta de desarrollo y especificación de requerimientos.

2.2.1 Identificación de requerimientos

Se analiza el caso planteado por el cliente, para visualizar las necesidades que deberán ser cubiertas para los usuarios.

2.2.2 Propuesta de desarrollo

En esta actividad, se forjan los diferentes planes factibles de desarrollo que cubran las necesidades del caso planteado y cumplan con las restricciones de recursos y tiempo.

2.2.3 Especificación de requerimientos

Especificar de forma clara cuáles serán los alcances, atributos, requerimientos y limitantes del producto a desarrollar, es decir, los requerimientos funcionales y no funcionales del producto.

Tabla 8. Requerimientos funcionales y no funcionales.

Descripción general del producto			
Objetivo:			
Requerimientos			
No.	Descripción	Funcional	No funcional
1			

2.3 Fase de diseño

En esta fase, se crean los modelos conceptuales y se definen las características explícitas e implícitas de cada uno de los componentes (espacios arquitectónicos, objetos, personajes, escenarios, guiones, dinámicas, secuencias de audio y video).

2.3.1 Modelado conceptual

Partiendo de la información generada en la fase anterior, el equipo de diseñadores crea el concepto (modelos o bocetos) que será la base para desarrollar los escenarios tridimensionales que conformarán la aplicación.

2.3.2 Modelado de navegación

Definir la forma en que el usuario podrá recorrer e interactuar con los elementos que componen la aplicación o sistema de RV, así como la técnica de representación que se utilizará (tipo de vista del usuario: primera persona o tercera persona).

2.3.3 Descripción de componentes

Su propósito es hacer una descripción detallada de los elementos que integran la aplicación de realidad virtual para identificar sus características físicas particulares, el tipo de interactividad y su comportamiento dentro de los escenarios. Para ello, se recomienda el uso de tablas de descripción de componentes, las cuales servirán para que los modeladores tridimensionales puedan revisar de manera inmediata los principales atributos que conforman cada elemento.

Tabla 9. Descripción de componentes.

Identificación:	Tipo de plantilla:	Descripción de objetos de ambientación				
	Objeto:	Características:			Cantidad:	Escenario:
		Color	Forma	Textura		
001						

2.3.4 Modelado de eventos

En esta actividad se detallan los sucesos que serán detectados durante la visita virtual y las acciones correspondientes que los usuarios, los objetos, y/o entornos deberán realizar. Es prudente advertir que si se desea un alto nivel de inmersión el modelo de eventos será más complejo.

2.4 Fase de desarrollo

El objetivo de esta fase es construir el producto. Para ello, se trabajara de forma iterativa e incremental para lograr una versión ejecutable de los escenarios de la aplicación al finalizar cada iteración, cabe resaltar, que si los elementos desarrollados no cumplen con los criterios establecidos se podrán retomar a las actividades anteriores con el objeto de modificar los errores.



Diagrama 2. Fase de desarrollo.

En la fase de desarrollo se seguirá la secuencia de actividades que se muestran en el diagrama 2 y se puntualizan a continuación:

2.4.1 Producción de recursos artísticos

Con el uso de herramientas de software para modelado tridimensional se crean espacios arquitectónicos, objetos, personajes, y terrenos que integran los escenarios de la aplicación didáctica, así como los materiales y texturas que serán aplicados a los componentes tridimensionales a fin de aumentar su realismo. De igual modo se desarrollan las secuencias de audio y video necesarias para complementar la representación de la información deseada.

2.4.2 Animación e integración

En lo que respecta a la actividad de animación se producen secuencias de movimiento, cambio de posición o transformación de parámetros de algunos componentes según los modelos de navegación y eventos. Posteriormente, en la etapa de integración todos los componentes tridimensionales, secuencias de audio y video son incorporados en los escenarios o ambientes virtuales.

2.4.3 Producción de recursos tecnológicos

Con la información proveniente de la etapa de diseño, se codifican y/o editan los eventos que disparan las acciones de respuesta establecidas para cada uno de ellos y para las dinámicas.

2.4.4 Pruebas

Una vez finalizada cada iteración, se verifica el funcionamiento y la integración de los recursos artísticos y tecnológicos con la finalidad de obtener información útil, en aras de mejorar el producto en aquellos aspectos concernientes a los parámetros funcionales que fueron establecidos en las fases de análisis y diseño.

2.5 Fase de evaluación

2.5.1 Valoración por el cliente / usuario

Cuando se considera que los componentes desarrollados se han integrado de manera adecuada en cada escenario y que estos conforman de manera coherente la representación virtual de cada actividad didáctica el producto es considerado como versión Alfa, la cual debe ser puesta a consideración del cliente a fin de confirmar que cumpla con los requerimientos planeados al inicio del proyecto.

Finalmente, la retroalimentación del cliente dará como resultado una versión Beta del producto, que se pone a consideración del usuario con pocos conocimientos acerca de la navegación en entornos virtuales, para valorar algún fallo no conocido por el equipo de desarrollo; con la finalidad de que el producto pueda ser liberado.

2.5.2 Fase de Cierre y Entrega Final

En esta fase se entrega el producto final de la aplicación, lista para implementarse en un ambiente con usuarios y condiciones reales, la cual será susceptible a recibir un plan de mantenimiento y/o actualización de contenidos o componentes según los recursos trazados en el proyecto.

Capítulo 3

Aplicación de la metodología

3.1 Fase de análisis

En esta fase se describieron tres actividades fundamentales para el desarrollo del proyecto: identificación de requerimientos, propuesta de desarrollo y especificación de requerimientos. Las cuales, tienen como objetivo tener como resultado un contexto claro y definido acerca del producto que se quiere desarrollar.

3.1.1 Identificación de requerimientos

De acuerdo al nivel de logro que muestran los estudiantes de educación básica en la prueba ENLACE en el área de matemáticas explicados con anterioridad, se decidió elegir el tema de fracciones debido a que los resultados exponen que un poco más de la mitad de estudiantes tanto a nivel de entidad como a nivel nacional tienden a cometer mayores equivocaciones cuando se les presentan problemas de números, repartos con fracciones, medidas y problemas multiplicativos.

En este sentido, primeramente para identificar los requerimientos se analizó el programa de matemáticas actual de la SEP de tercer grado de primaria, ya que es el nivel en donde se empieza a introducir a los estudiantes en el tema de fracciones matemáticas, a continuación se muestran algunos bloques del programa de estudio (tablas 10 – 12).

Tabla 10. Programa de estudio del tercer bloque de la materia de matemáticas en educación básica. (Secretaría de Educación Pública, 2015).

Bloque III		
Aprendizajes esperados	Ejes	
	Sentido numérico y pensamiento algebraico	Manejo de la información
<ul style="list-style-type: none"> Resuelve problemas de reparto cuyo resultado sea una fracción de la forma $m/2n$. Utiliza el algoritmo convencional para resolver sumas o restas con números naturales. 	<p>Números y sistemas de numeración</p> <ul style="list-style-type: none"> Uso de fracciones del tipo $m/2n$ (medios, cuartos, octavos, etc.) para expresar oralmente y por escrito medidas diversas. Uso de fracciones del tipo $m/2n$ (medios, cuartos, octavos, etc.) para expresar oralmente y por escrito el resultado de repartos. Identificación de la regularidad en sucesiones con números, ascendentes o descendentes, con progresión aritmética para continuar la sucesión o encontrar términos faltantes. <p>Problemas aditivos</p> <ul style="list-style-type: none"> Estimación del resultado de sumar o restar cantidades de hasta cuatro cifras, a partir de descomposiciones, redondeo de los números, etcétera. Determinación y afirmación de un algoritmo para la sustracción de números de dos cifras. <p>Problemas multiplicativos</p> <ul style="list-style-type: none"> Resolución de problemas de división (reparto y agrupamiento) mediante diversos procedimientos, en particular el recurso de la multiplicación. 	<p>Análisis y representación de datos</p> <ul style="list-style-type: none"> Resolución de problemas en los cuales es necesario extraer información explícita de diversos portadores.

Tabla 11. Programa de estudio del cuarto bloque de la materia de matemáticas en educación básica. (Secretaría de Educación Pública, 2015).

Bloque IV		
Aprendizajes esperados	Ejes	
	Sentido numérico y pensamiento algebraico	Manejo de la información
<ul style="list-style-type: none"> Resuelve problemas que implican identificar la regularidad de sucesiones con progresión aritmética. Resuelve problemas que implican efectuar hasta tres operaciones de adición y sustracción. Resuelve problemas que impliquen dividir mediante diversos procedimientos. 	<p>Números y sistemas de numeración</p> <ul style="list-style-type: none"> Identificación de escrituras equivalentes (aditivas, mixtas) con fracciones. Comparación de fracciones en casos sencillos (con igual numerador o igual denominador). Identificación de la regularidad en sucesiones con figuras, con progresión aritmética, para continuar la sucesión o encontrar términos faltantes. <p>Problemas aditivos</p> <ul style="list-style-type: none"> Resolución de problemas que impliquen efectuar hasta tres operaciones de adición y sustracción. <p>Problemas multiplicativos</p> <ul style="list-style-type: none"> Identificación y uso de la división para 	<p>Análisis y representación de datos</p> <ul style="list-style-type: none"> Identificación de ángulos como resultado de cambios de dirección. Obtención de ángulos de 90° y 45°, a través del doblado de papel. Reproducción de los ángulos en papel.

	resolver problemas multiplicativos, a partir de los procedimientos ya utilizados (suma, resta, multiplicación). Representación convencional de la división: $a \div b = c$.	
--	--	--

Tabla 12. Programa de estudio del quinto bloque de la materia de matemáticas en educación básica. (Secretaría de Educación Pública, 2015).

Bloque V		
Aprendizajes esperados	Ejes	
	Sentido numérico y pensamiento algebraico	Manejo de la información
<ul style="list-style-type: none"> Utiliza unidades de medida estándar para estimar y medir longitudes. 	<p>Números y sistemas de numeración</p> <ul style="list-style-type: none"> Elaboración e interpretación de representaciones gráficas de las fracciones. Reflexión acerca de la unidad de referencia. <p>Problemas aditivos</p> <ul style="list-style-type: none"> Resolución de problemas sencillos de suma o resta de fracciones (medios, cuartos, octavos). <p>Problemas multiplicativos</p> <ul style="list-style-type: none"> Desarrollo y ejercitación de un algoritmo para la división entre un dígito. Uso del repertorio multiplicativo para resolver divisiones (cuántas veces está contenido el divisor en el dividendo). 	<p>Análisis y representación de datos</p> <ul style="list-style-type: none"> Comparación por tanteo, del peso de dos objetos y comprobación en una balanza de platillos. Trazo de segmentos a partir de una longitud dada.

Una vez que se examinó el contenido de cada bloque, en la siguiente lista se resumen los temas encontrados correspondientes a las fracciones matemáticas:

1. Uso de fracciones del tipo $m/2n$ para expresar medidas diversas.
2. Uso de fracciones del tipo $m/2n$ para expresar resultados de reparto.
3. Comparación de fracciones con igual o diferente denominador.
4. Identificación de fracciones equivalentes (aditivas y mixtas).
5. Resolución de problemas sencillos de suma y resta con fracciones del tipo $m/2n$

3.1.2 Propuesta de desarrollo

Con base en los conocimientos adquiridos en los estudios de ingeniería y a la experiencia obtenida en cursos especializados se decidió trabajar con RV, aunado a los beneficios que ofrece esta tecnología en comparación con otras que son utilizadas para desarrollar aplicaciones educativas.

Por lo que, la propuesta consiste en crear una aplicación de RV que sirva como una herramienta de apoyo didáctico para la enseñanza aprendizaje del tema de fracciones, es decir, un juego serio de entretenimiento educativo, el cual estará conformado por cuatro escenarios llamados Asteroides (Glorieta, Parque, Laguna y Vecindario) y un personaje principal (usuario), el cual ayudará a los demás personajes a resolver problemas en distintas actividades (misiones) cotidianas, que involucren la solución de los temas de fracciones que se encuentran incluidos en el programa académico de la SEP de tercer año de primaria.

El contexto de los cuatro asteroides está conformado por distintos lugares reales con aspecto de fantasía, en donde se presentan diversas misiones que se pretenden sean contestadas por el usuario, con la finalidad de reforzar los conocimientos adquiridos en el aula de clases. Las cuales están distribuidas de la siguiente manera:

El escenario Asteroide Glorieta (figura 26) está representado por los siguientes elementos:

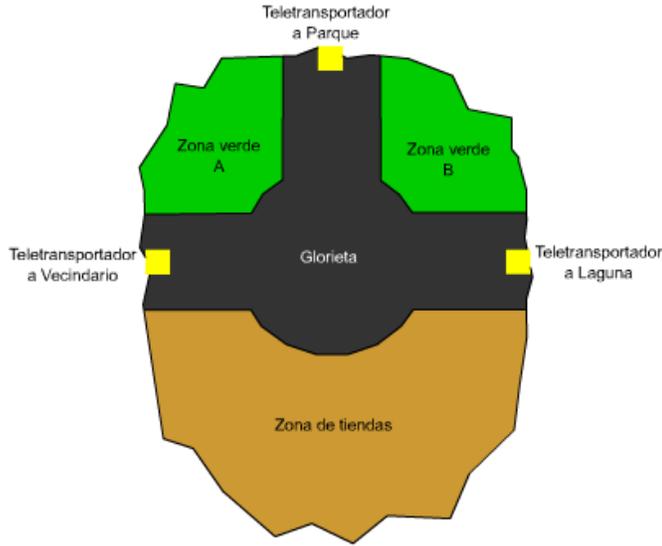


Figura 26. Diseño 2D del asteroide Glorieta.

- Una glorieta en el centro del escenario con tres caminos diferentes. Cada camino llevara a un asteroide distinto.
- Zonas verdes
- Una zona de tiendas

El escenario Asteroide Laguna (figura 27) está representado por los siguientes elementos:

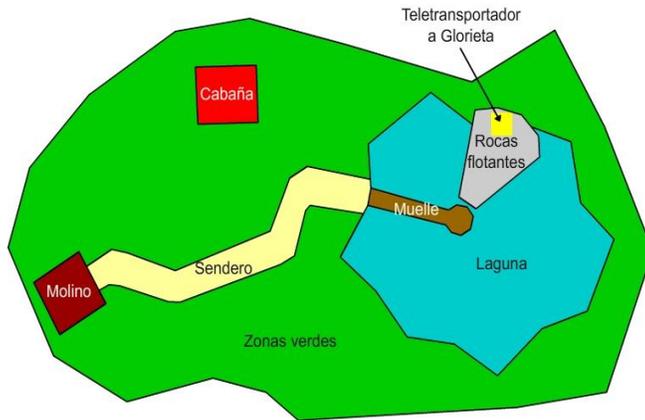
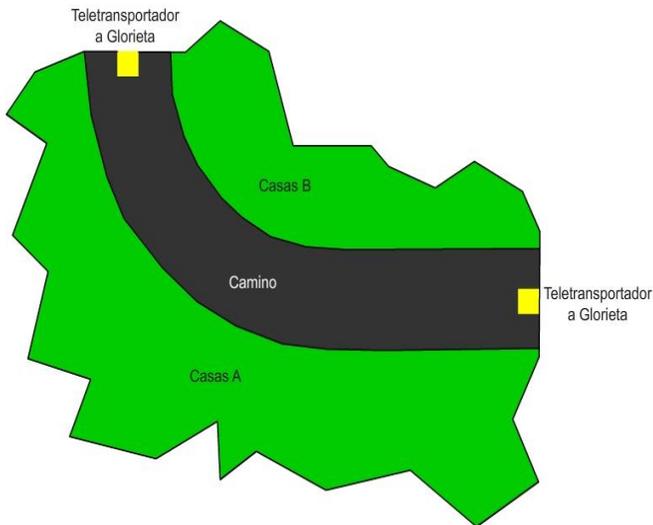


Figura 27. Diseño 2D del Asteroide Laguna.

- h) Una laguna con un muelle
- Un molino
 - Zonas verdes
 - Una cabaña
- Rocas flotantes que simulan una escalera para llegar a un teletransportador

El escenario Asteroide Vecindario está representado por los siguientes elementos:

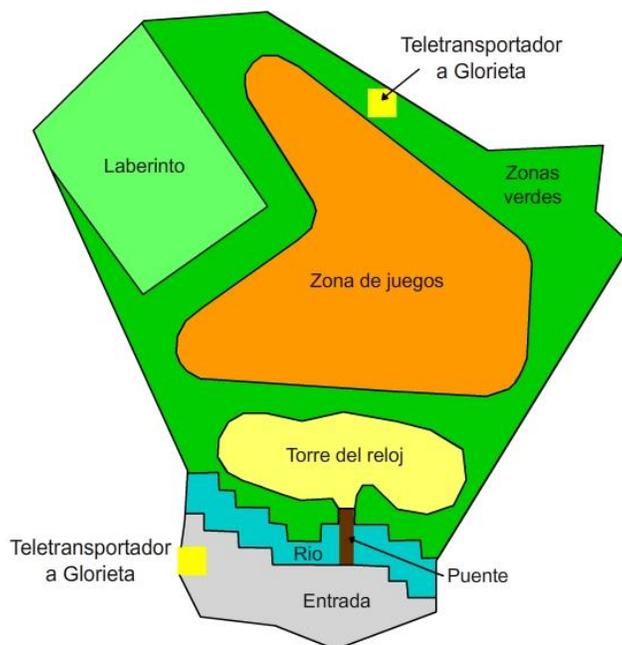


i) Un camino de asfalto que en ambos extremos tendrá un teletransportador

- Casas
- Zonas verdes

Figura 28. Diseño 2D del Asteroide Vecindario.

El escenario Asteroide Parque está representado por los siguientes elementos:



- Un laberinto
- Zona de juegos
- Un puente que atravésara un río
- Una entrada a él con forma de castillo
- Un camino de arena con una torre de reloj en el centro

Figura 29. Diseño 2D del Asteroide Parque.

Se pretende que el usuario inicie en el escenario Asteroide Glorieta en donde el guía Gus le da la bienvenida y los objetivos que se deben de cumplir en cada Asteroide. Posteriormente el usuario decide que misiones realizar de acuerdo a los temas vistos en el aula de clases.

Cabe mencionar que para la propuesta de las misiones se consultaron libros de matemáticas de tercer grado de primaria para analizar el tipo de ejercicios que se manejan en el curso y de este modo obtener ideas.

Dentro de los diferentes asteroides se propone representar las siguientes actividades didácticas que se encuentran estructuradas en la tabla 13.

Tabla 13. Propuestas de misiones.

Misiones				
Tema	Bloque	Escenario	Ubicación	Descripción
Medidas	III	<ul style="list-style-type: none"> Vecindario Glorieta 	<ul style="list-style-type: none"> Casas Zona de tiendas 	El lechero necesita ayuda para repartir dos pedidos diferentes de leche en dos casas del vecindario que se encuentran ubicadas a cierta distancia de la calle. Por ejemplo en la casa que está del lado derecho a los $\frac{3}{4}$ de la calle.
		<ul style="list-style-type: none"> Glorieta 	<ul style="list-style-type: none"> Zona de tiendas 	Ayudar al señor de la tienda a acomodar productos por peso en el mostrador. El usuario debe arrastrar los productos que correspondan al peso indicado en cada una de las repisas del mostrador.
Reparto	III	<ul style="list-style-type: none"> Glorieta 	<ul style="list-style-type: none"> Zona de tiendas 	El usuario debe recolectar las monedas de los cofres que se encuentran distribuidos en los 4 asteroides. Al juntar 5 monedas el usuario debe dirigirse a la pizzería a comprar una pizza para repetirla con sus compañeros en partes iguales.
		<ul style="list-style-type: none"> Parque 	<ul style="list-style-type: none"> Zona de juegos 	Memorama de repartos. El memorama tendrá cartas con caras de personas y fracciones. Por ejemplo si el usuario levanta una carta con las caras de tres personas, entonces debe buscar la fracción que indique la porción que le toca a cada una de ellas, en este caso sería $\frac{1}{3}$.
Comparación	IV	<ul style="list-style-type: none"> Laguna 	<ul style="list-style-type: none"> Muelle 	El usuario deberá pescar dos de los peces que se encuentran en la laguna, uno debe ser el de mayor peso y el otro de menor.
		<ul style="list-style-type: none"> Laguna 	<ul style="list-style-type: none"> Zonas verdes 	La rana está perdida y quiere regresar al lago. Para que regrese al lago el usuario debe comparar que fracción es la menor. La rana llegara al contestar tres problemas. Para el primer problema debe comparar dos fracciones, para el segundo tres y para el tercero cuatro.
Equivalencias	IV	<ul style="list-style-type: none"> Parque 	<ul style="list-style-type: none"> Zona de juegos 	El usuario debe disparar a monedas que se deslizan lentamente por unas repisas. Estas monedas tendrán fracciones en su centro. El usuario debe disparar únicamente a las fracciones

				que sean equivalentes a $\frac{1}{2}$
		<ul style="list-style-type: none"> • Glorieta • Vecindario • Laguna • Parque 	<ul style="list-style-type: none"> • Zona de tiendas • Todas las zonas verdes 	El usuario debe ayudar al pizzero a hacer una pizza. El usuario deberá conseguir los ingredientes y la porción adecuada al peso que indique el pizzero.
Suma y resta	V	<ul style="list-style-type: none"> • Laguna 	<ul style="list-style-type: none"> • Zonas verdes • Cabaña 	La abuelita Domi necesita ayuda para recoger $1\frac{3}{4}$ Kg de manzanas saltarinas de las zonas verdes del asteroide Laguna. El usuario debe buscar y capturar las manzanas y regresar a la cabaña a entregarlas cuando tenga el peso correcto. En la interfaz aparecerá un marcador indicando el peso que ha juntado con cada manzana, el peso de cada manzana aparecerá en la interfaz cuando de clic sobre ella.
		<ul style="list-style-type: none"> • Parque 	<ul style="list-style-type: none"> • Zona de juegos 	Aparecerán restas de fracciones y el usuario debe seleccionar el resultado correcto para que avance su caballito
Combinación de todos los temas	III, IV y V	<ul style="list-style-type: none"> • Parque 	<ul style="list-style-type: none"> • Laberinto 	El usuario debe cruzar el laberinto para llegar al tesoro (recompensa). En el trayecto del laberinto se cruzara con una serie de retos diversos, uno de cada tema. Estos retos se encontraran en cada una de las 5 puertas bloqueadas y que se desbloqueen al resolver el problema.

Lo anterior, con la finalidad de que el estudiante pueda aplicar sus conocimientos adquiridos y observar en qué actividades cotidianas puede implementarlos, de esta manera se pretende que a través de estas experiencias los estudiantes puedan alcanzar un aprendizaje significativo.

Cabe señalar que para el diseño, se tomaron como base la película “Lorax” y los videojuegos “Mario Galaxy” y “Mario Kart” que se encuentran en la clasificación Everyone, es decir en donde el contenido es apto para todas las edades y manejan el mínimo de violencia representada con caricatura o fantasía (ESRB, 2017).

3.1.3 Especificación de requerimientos

Con base en la información establecida en la identificación de requerimientos y en la propuesta de desarrollo, en esta actividad se especificaron cuáles son los alcances, atributos y limitantes de la aplicación de RV que se va a desarrollar, ya que no se contaba con un cliente como tal, a través de los requerimientos funcionales y no funcionales, (tabla 14).

Tabla 14. Requerimientos funcionales y no funcionales.

Descripción general del producto	Aplicación didáctica de RV no inmersiva enfocada al tema de fracciones que se encuentra en los bloques III, IV y V de la materia de matemáticas de tercer grado de primaria.		
Objetivo	Crear una aplicación de RV que sirva como una herramienta de apoyo didáctico para la enseñanza aprendizaje del tema de fracciones que se encuentra distribuido en los bloques III, IV y V de la materia de matemáticas del tercer grado de primaria.		
Requerimientos			
No.	Descripción	Funcional	No funcional
1	La aplicación didáctica debe contar con características que permiten al usuario acceder a ella, desde cualquier equipo de cómputo.		✓
2	Se debe mostrar al usuario las instrucciones necesarias para realizar cada misión adecuadamente.	✓	
3	Se debe recompensar al usuario cuando finalice de manera exitosa cada misión.	✓	
4	El usuario puede decidir qué rumbo tomar dentro de cada escenario.	✓	
5	El audio de la aplicación didáctica deberá ser claro y con alta fidelidad.	✓	
6	El contexto de la aplicación en general debe combinar lo realista con la fantasía.	✓	
7	La interacción del usuario con los personajes de la aplicación será a través diálogos textuales manipulables a través del mouse y teclado.	✓	
8	El usuario tendrá la posibilidad de reintentar cada misión si no la supera.	✓	
9	El personaje principal deberá reaccionar a las instrucciones del usuario.	✓	
10	El contexto de los escenarios debe integrar lugares que sean familiares para los usuarios.	✓	
11	Se deben de abordar todos los temas de fracciones que se encuentren en el programa de tercer grado de educación básica.	✓	
12	El personaje protagonista será un Boy Scout.	✓	
13	La vista del usuario será con perspectiva de primera persona.	✓	
14	La apariencia de los personajes será de un aspecto de fantasía.	✓	
15	Las misiones deberán representar actividades que los usuarios posiblemente realizan o han realizado en su vida cotidiana.	✓	
16	La interacción con los objetos de las misiones se realizara con la ayuda del mouse. Por ejemplo: mover y seleccionar.	✓	

Nota: Donde los requerimientos funcionales hacen referencia a las funciones que el sistema será capaz de realizar. Es decir, las transformaciones que el sistema realiza sobre las entradas para producir salidas. Mientras que los no funcionales son características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, etc.

3.2 Fase de Diseño

En esta fase se desarrolla el concepto del entorno de la aplicación y se establece como será su contenido, definiendo las características de cada elemento que lo conformaran

como son objetos bidimensionales y tridimensionales, personajes, escenarios, secuencias de audio, guiones y dinámicas.

3.2.1 Modelado conceptual

Después de realizar el estudio de los requerimientos en la fase de análisis se crean los bocetos de los componentes tridimensionales que conformaran los escenarios de la aplicación didáctica.

3.2.1.1 Escenarios

En esta actividad se dibujaron bocetos de los cuatro escenarios propuestos:

- a) Asteroide Glorieta

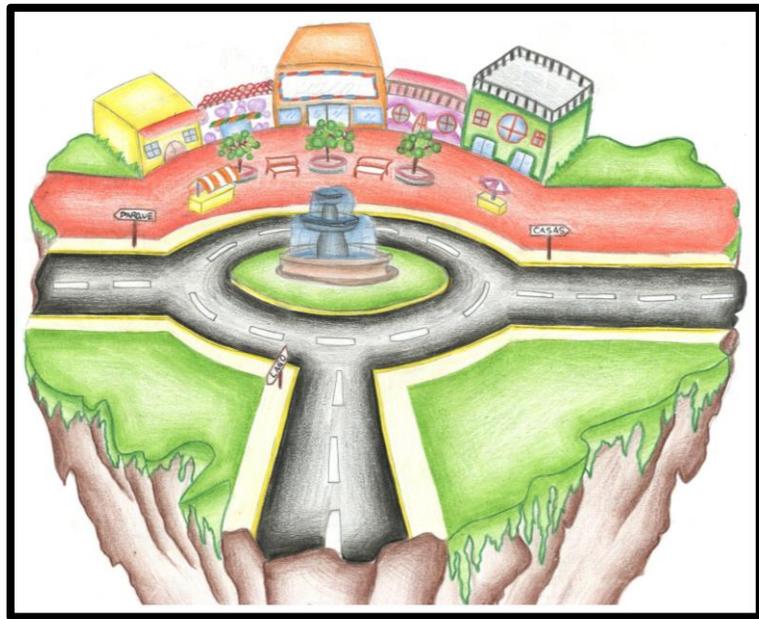


Figura 30. Boceto de escenario "Glorieta".

b) Asteroide Parque

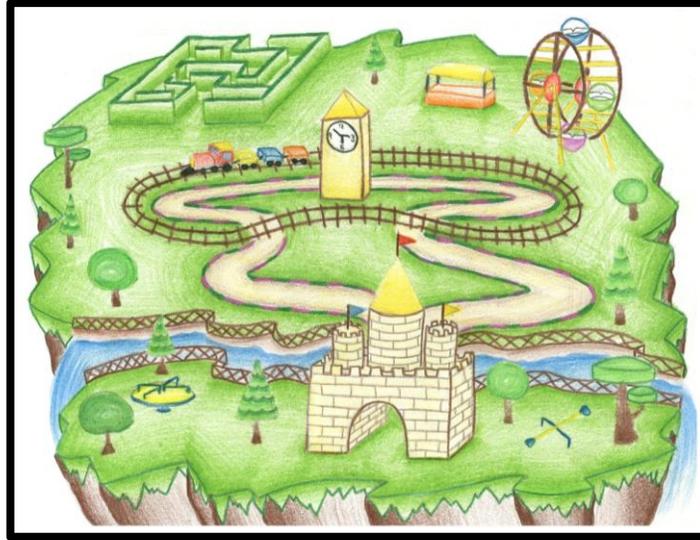


Figura 31. Boceto de escenario "Parque".

c) Asteroide Laguna

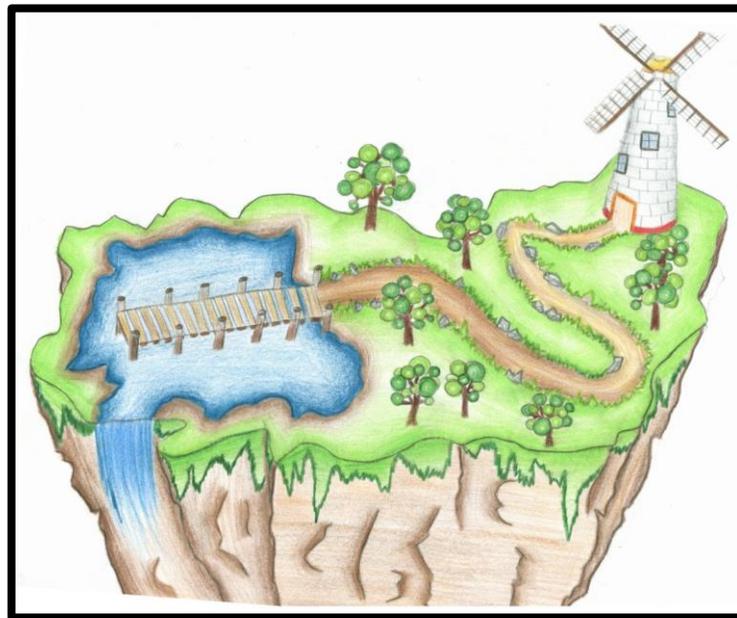


Figura 32. Boceto de escenario "Laguna".

d) Asteroide Vecindario

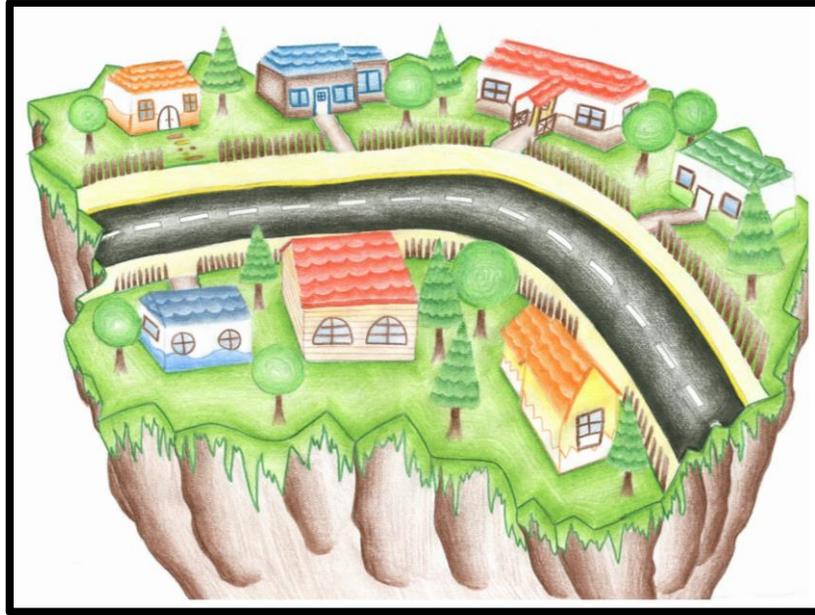


Figura 33. Boceto de escenario “Vecindario”.

3.2.1.2 Objetos 3D

En esta actividad se diseñaron todos los objetos de ambientación e interacción, los cuales se clasificaron en escenarios y personajes como se observa a continuación:

a) **Personajes** (tabla 15)

Tabla 15. Bocetos de personajes.

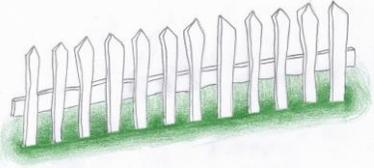
Personajes		
Objeto	Descripción	Tipo de elemento
	<p>“Charlie”</p> <p>Es el compañero del usuario en la aventura.</p>	<p>Ambientación NPC (Non Player Controller)</p>
	<p>“Michelle”</p> <p>Es la compañera del usuario en la aventura.</p>	<p>Ambientación NPC</p>
	<p>“Guía Gus”</p> <p>Es el guía del usuario, Charlie y Michelle durante la aventura. Su propósito es dar instrucciones de las misiones que debe cumplir el usuario.</p>	<p>Interacción NPC</p>

	<p style="text-align: center;">“Abuelita Domi”</p> <p>Es una abuelita que vive en la cabaña del asteroide Laguna. Ella tiene algunas complicaciones con el uso de fracciones y una gran pasión por las manzanas.</p>	<p style="text-align: center;">Ambientación NPC</p>
	<p style="text-align: center;">“Lechero Ritchie”</p> <p>Es el dueño de la lechería del asteroide Glorieta. Él suministra de leche a las casas del asteroide Vecindario.</p>	<p style="text-align: center;">Ambientación NPC</p>
	<p style="text-align: center;">“Pizzero Alex ”</p> <p>Es el dueño de la pizzería del asteroide Glorieta. Su pasión es preparar deliciosas pizzas para los habitantes de todos los asteroides.</p>	<p style="text-align: center;">Ambientación NPC</p>

b) Escenarios

- Asteroide Vecindario (tabla 16)

Tabla 16. Bocetos de elementos del asteroide Vecindario.

Asteroide Vecindario		
Objeto	Descripción	Tipo de elemento
	<p>Casa del vecindario</p> <p>Las casas del vecindario serán utilizadas para formar una comunidad en el asteroide Vecindario.</p>	Ambientación
	<p>Lámpara</p> <p>Este objeto brindará iluminación en los asteroides.</p>	Ambientación
	<p>Cercas</p> <p>Proporcionará contexto y estética al asteroide.</p>	Ambientación
	<p>Flores</p> <p>Serán utilizadas para dar un aspecto más colorido en algunas zonas verdes.</p>	Ambientación

- Asteroide Glorieta (tabla 17)

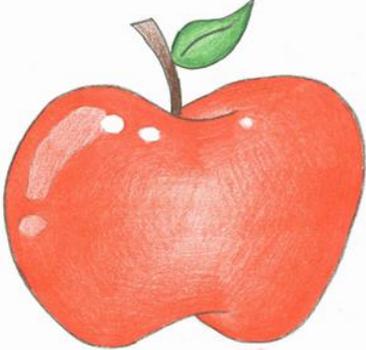
Tabla 17. Bocetos de elementos del asteroide Glorieta.

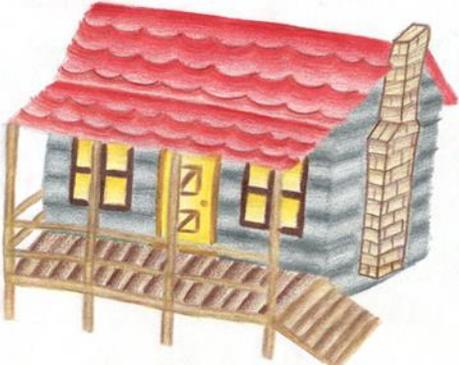
Asteroide Glorieta		
Objeto	Descripción	Tipo de elemento
	<p>Teletransportadores</p> <p>Serán utilizados para la navegación del usuario entre asteroides.</p>	Interacción
	<p>Letrero de dirección</p> <p>Indicaran la dirección en la que se encontrara cada escenario.</p>	Ambientación
	<p>Fuente de agua</p> <p>Proporcionará contexto y estética al asteroide.</p>	Ambientación

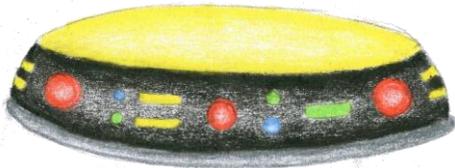
	<p style="text-align: center;">Local Pizzería</p> <p>Este elemento se incluirá como local de la zona de tiendas para ser utilizado en la misión "Moon's Pizza".</p>	<p style="text-align: center;">Interacción</p>
	<p style="text-align: center;">Cofre de monedas</p> <p>Estarán ubicados en distintas zonas de los asteroides y contendrán algunos tesoros.</p>	<p style="text-align: center;">Interacción</p>

- Asteroide Laguna (tabla 18)

Tabla 18. Bocetos de elementos del asteroide Laguna.

Asteroide Laguna		
Objeto	Descripción	Tipo de elemento
	<p style="text-align: center;">Manzana saltarina</p> <p>Las manzanas se incluirán dentro de las zonas verdes del asteroide y tendrán la peculiaridad de ser saltarinas.</p>	<p style="text-align: center;">Interacción</p>

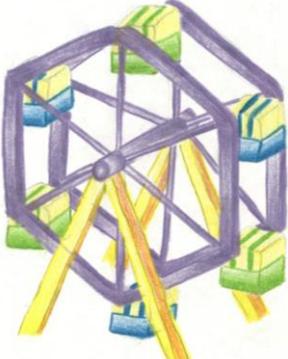
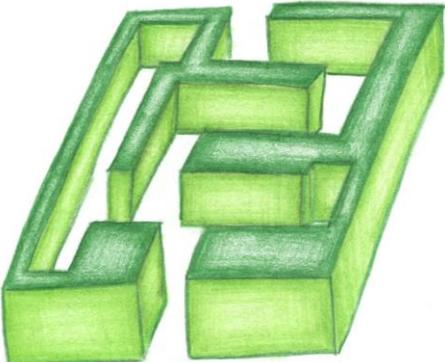
	<p>Muelle</p> <p>Este elemento se diseñó para dar vista a la laguna del escenario.</p>	<p>Ambientación</p>
	<p>Árbol</p> <p>Proporcionará contexto y estética al asteroide.</p>	<p>Ambientación</p>
	<p>El molino estará ubicado en la loma del asteroide Laguna para mejorar su estética.</p>	<p>Ambientación</p>
	<p>Cabaña</p> <p>La cabaña será el hogar de la abuelita Domi, será utilizada para dar vista al escenario.</p>	<p>Ambientación</p>

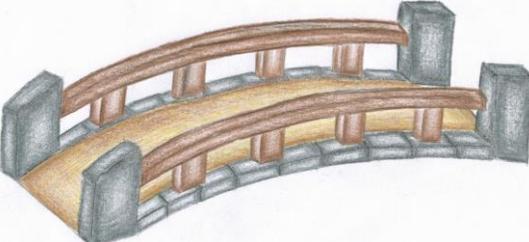
	<p>Teletransportador de piso</p> <p>Será utilizado para teletransportar al personaje al asteroide glorieta</p>	<p>Interacción</p>
	<p>Bote</p> <p>Proporcionará contexto y estética al asteroide.</p>	<p>Ambientación</p>

- Asteroide Parque (tabla 19)

Tabla 19. Bocetos de elementos del asteroide Parque.

Escenario Parque		
Objeto	Descripción	Tipo de elemento
	<p>Carrito de hot dogs</p> <p>Proporcionará contexto y estética al asteroide.</p>	<p>Ambientación</p>

	<p>Tren</p> <p>Proporcionará contexto y estética al asteroide.</p>	<p>Ambientación</p>
	<p>Rueda de la fortuna</p> <p>Proporcionará contexto y estética al asteroide.</p>	<p>Ambientación</p>
	<p>Torre del reloj</p> <p>Proporcionará contexto y estética al asteroide.</p>	<p>Ambientación</p>
	<p>Laberinto</p> <p>Este objeto será utilizado para la misión "Galactic Labyrinth".</p>	<p>Interacción</p>

	<p>Tienda de disparos</p> <p>Este objeto será utilizado como sitio para realizar la misión "Spatial Coins".</p>	<p>Ambientación</p>
	<p>Moneda</p> <p>Este objeto será utilizado como objetivo en las misiones "Spatial Coins" y "Moon's Pizza".</p>	<p>Interacción</p>
	<p>Banca</p> <p>Proporcionará contexto y estética al asteroide.</p>	<p>Ambientación</p>
	<p>Puente</p> <p>Proporcionará contexto y estética al asteroide.</p>	<p>Ambientación</p>

3.2.2 Modelado de navegación

En esta etapa se desarrollaron los diagramas que describen como están organizados e interrelacionados los escenarios y la forma en que el usuario podrá recorrer los

lugares que integran la aplicación didáctica y como interactuara con los componentes tridimensionales que intervienen en las misiones propuestas en la fase de análisis.

3.2.2.1 Diagrama de navegación de escenarios

La navegación que el usuario podrá realizar en los cuatro escenarios se muestra en el Diagrama 3.

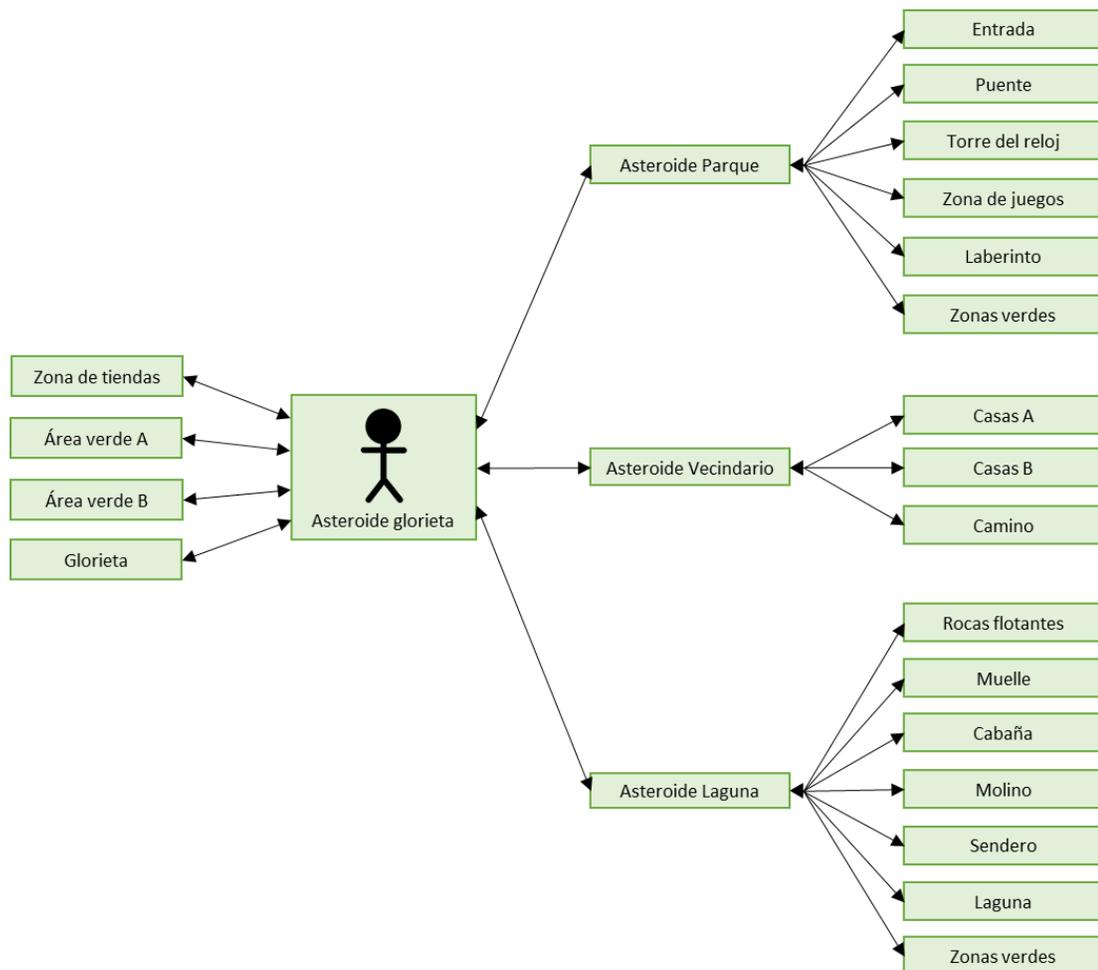
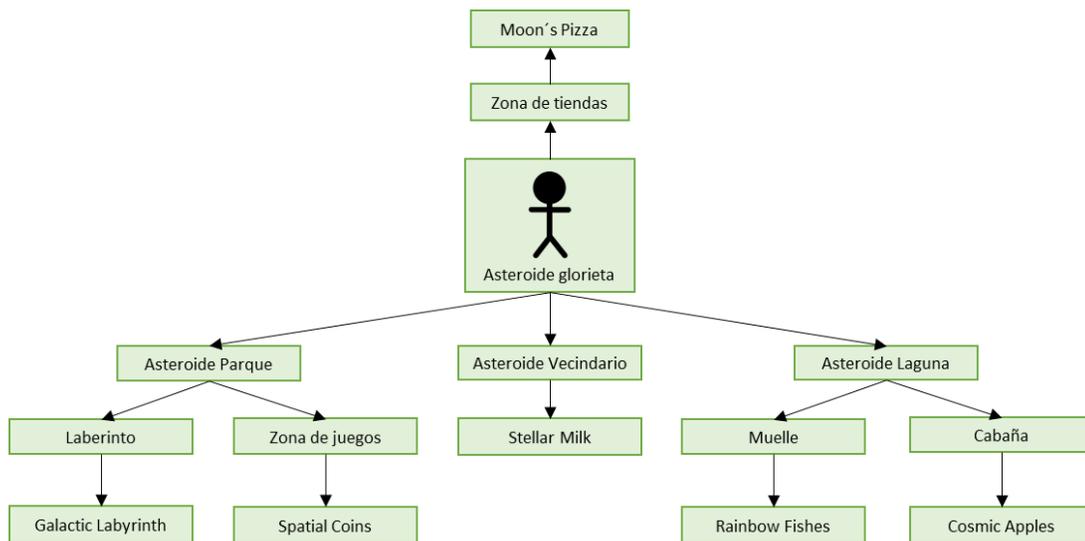


Diagrama 3. Modelo de navegación.

Como se observa en el diagrama, el usuario inicia en el asteroide Glorieta, a partir de este punto se podrá desplazarse en las diferentes áreas de cada escenario para completar las misiones.

3.2.2.2 Diagrama de ubicación de misiones

En este diagrama se especifica dónde estarán ubicadas las misiones que el usuario puede realizar dentro de cada escenario. Por ejemplo, en la cabaña del asteroide Laguna se encuentra la misión Cosmic Apples.



3.2.2.3 Diagramas de navegación de misiones

En este apartado se crearon los diagramas que describen la navegación que el usuario puede realizar en las misiones y sus posibilidades que pueden tener dependiendo de las decisiones que tome (Diagrama 5 – 10), por ejemplo: realizar erróneamente o correctamente alguna actividad tendrá diferentes consecuencias (volverlo a intentar, aumento o disminución de puntos, felicitaciones, etc.).

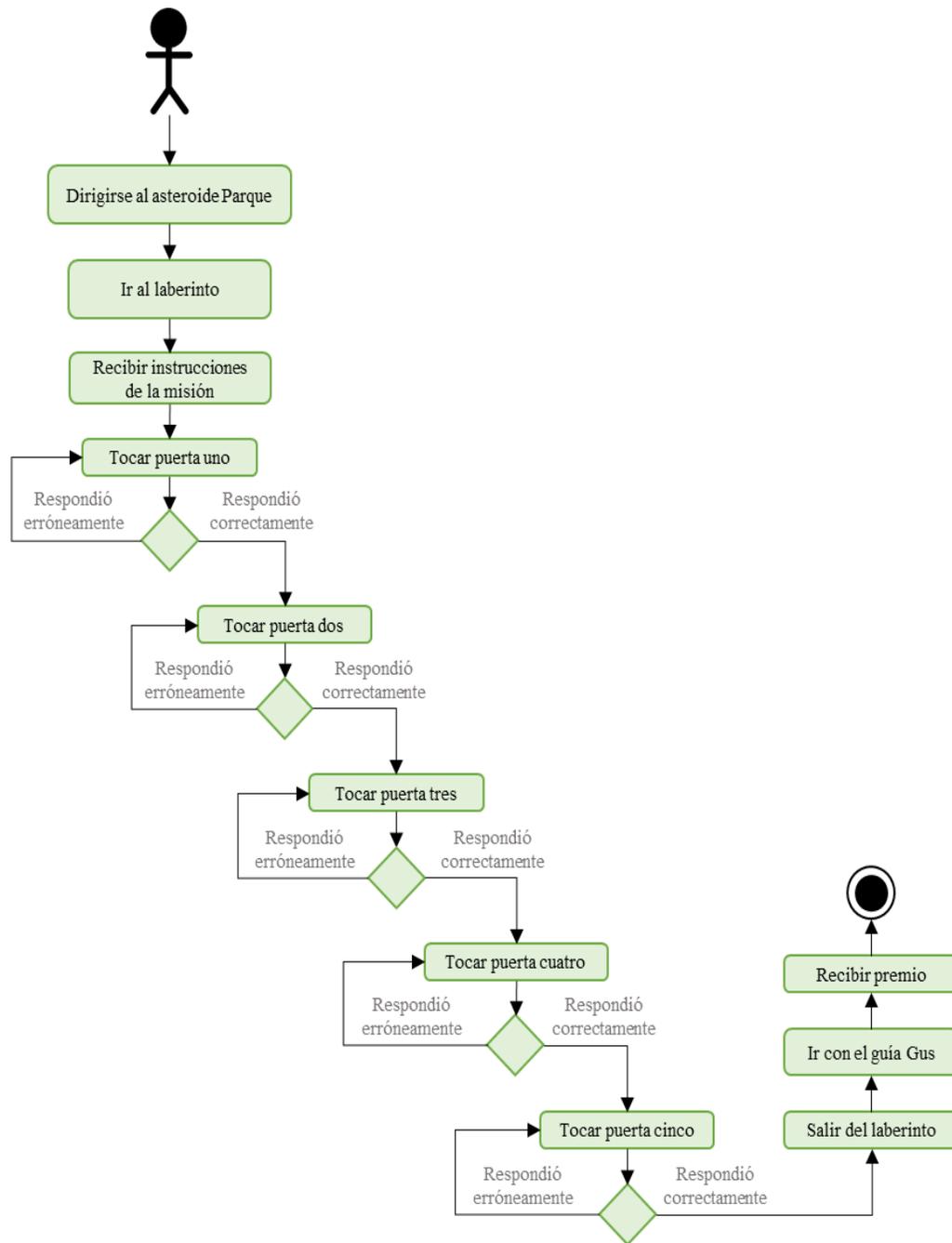


Diagrama 5. Navegación de la misión Galactic Labyrinth.

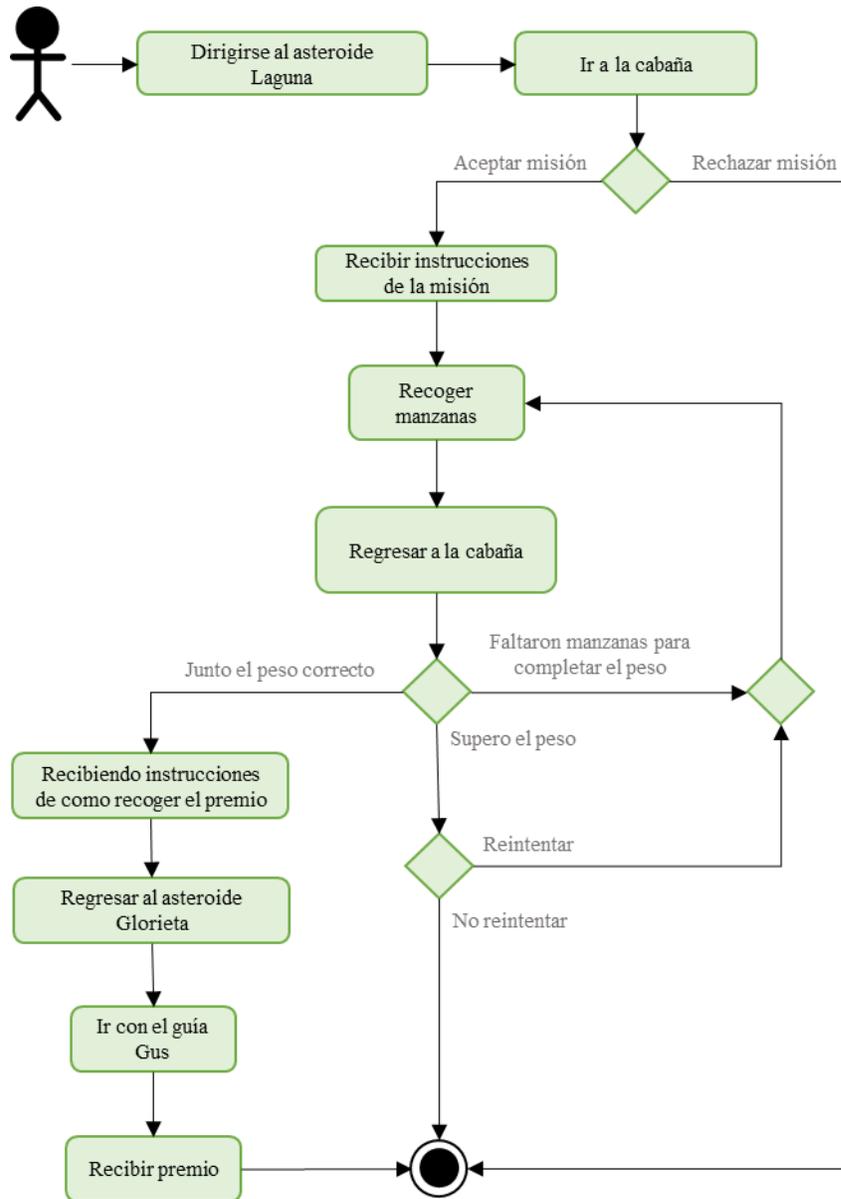


Diagrama 6. Navegación de la Misión Cosmic Apples.

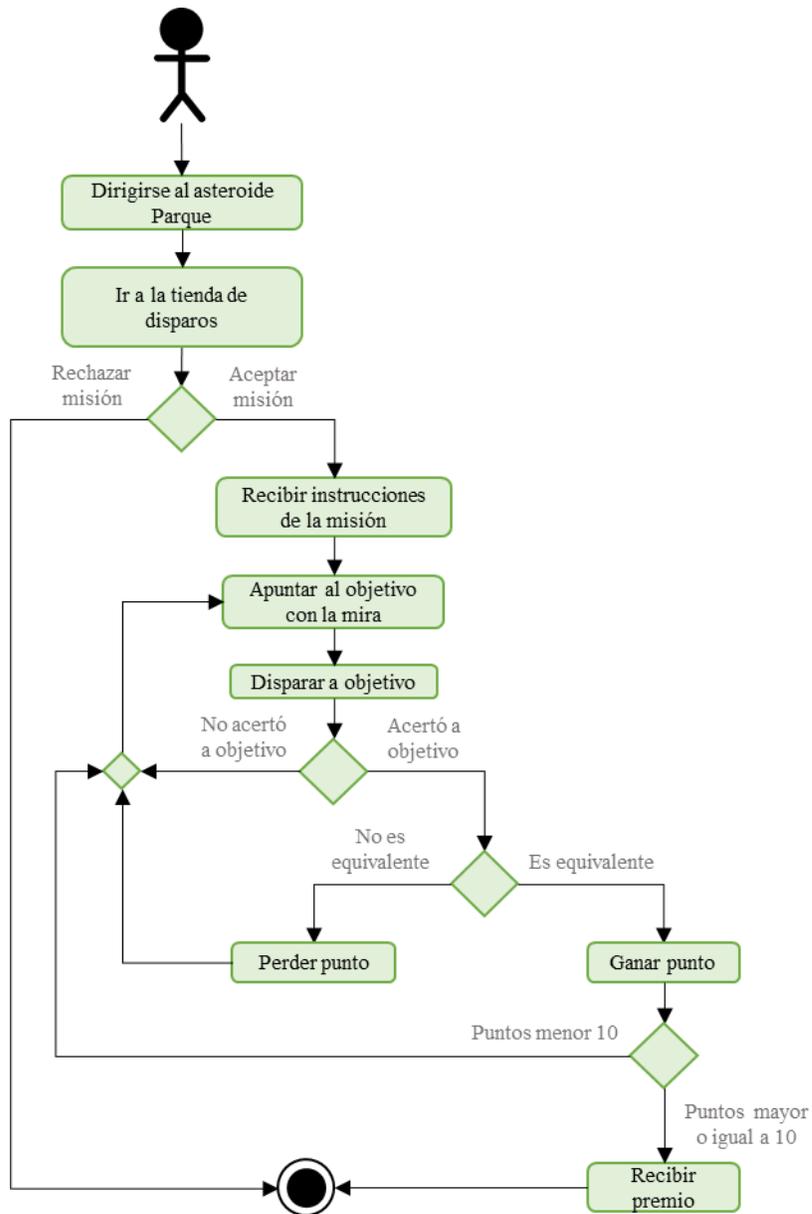


Diagrama 7. Navegación de la Misión Spatial Coins.

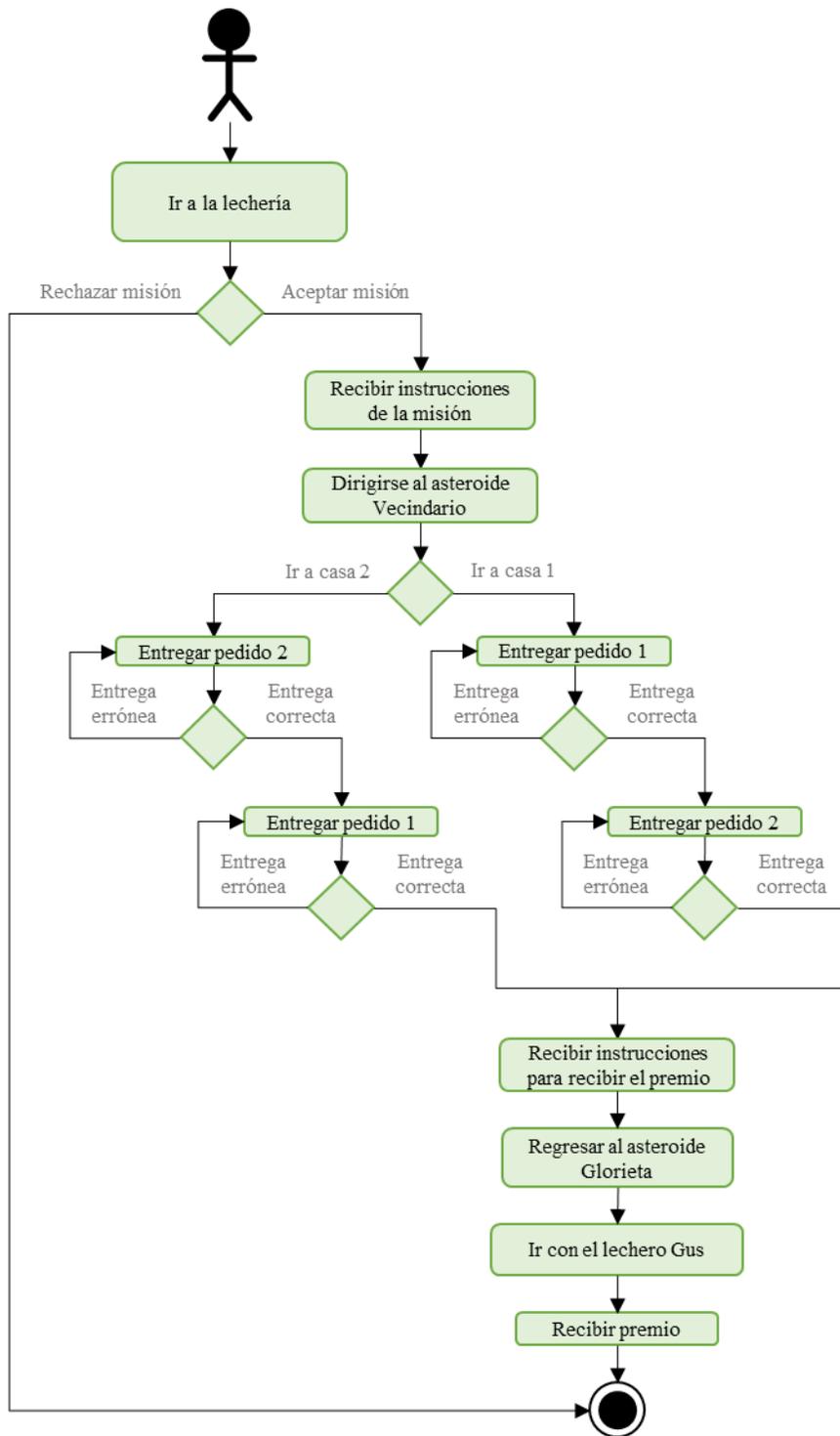


Diagrama 8. Navegación de la Misión Stellar Milk.

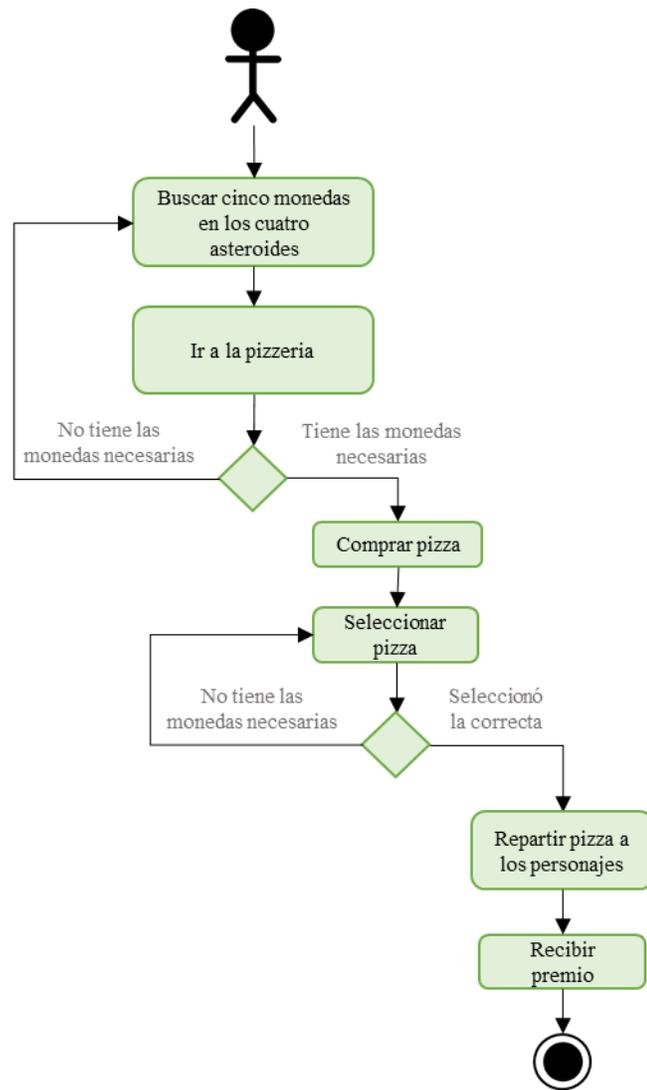


Diagrama 9. Navegación de la Misión Moon's Pizza.

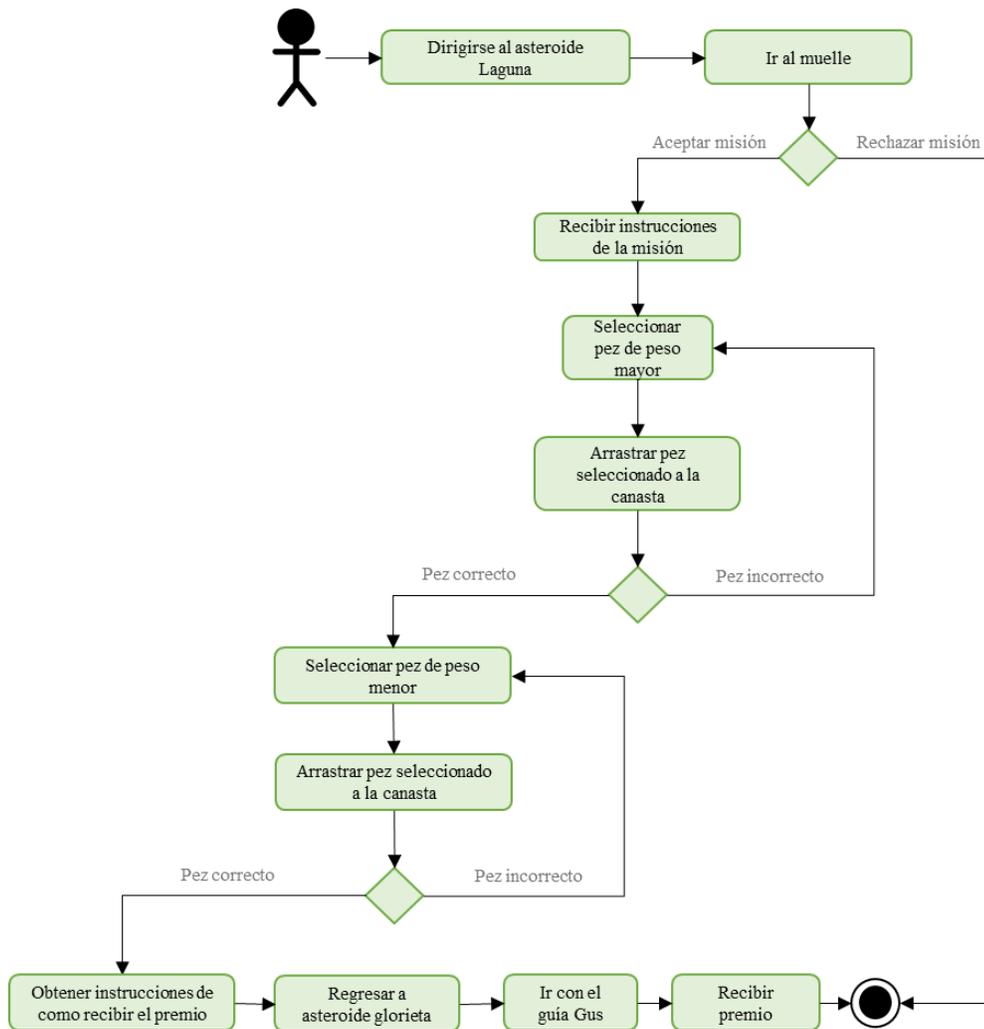


Diagrama 10. Navegación de la Misión Rainbow Fishes.

Cabe mencionar que para navegar a través de todos los escenarios el usuario manejará una vista en primera persona (Figura 34) en la cual se ve el mundo desde la perspectiva del protagonista, de este modo se obtiene un mejor aprendizaje (Vera, Ortega & Burgos, 2003).

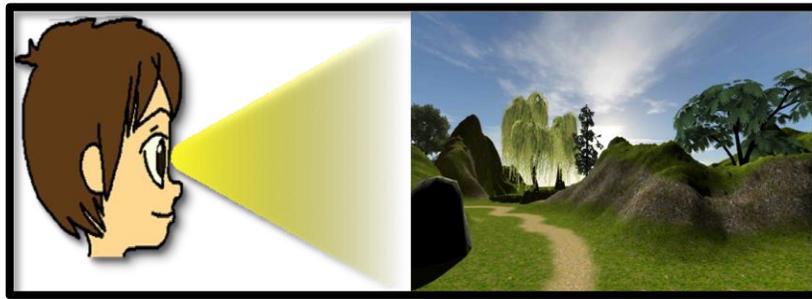


Figura 34. Vista en primera persona.

3.2.2.4 Storyboard

El storyboard suele ser de gran ayuda, porque a través de bocetos, proporciona la secuencia del guion y un acercamiento visual de la aplicación. Este puede ser tan extenso que llega a complementar y en algunas ocasiones sustituir completamente al guion (Guijón, 2006).

En el caso del presente proyecto, se realizó un formato (ejemplo: Storyboard 1) con la finalidad de tener una estructura ordenada de las historias que serán incluidas dentro de la aplicación didáctica e incluye los siguientes datos en la cabecera:

- c) **Nombre del proyecto:** Nombre de la aplicación.
- d) **Descripción:** Descripción general de la actividad.
- e) **Actividad:** Nombre de la actividad o misión.
- f) **Página:** En esta sección se utiliza el formato de número de página siguiente “número de página actual”-”número total de páginas”.
- g) **Número de secuencia cronológica.** Para el cuerpo del formato se utilizaron recuadros que contienen los bocetos, la descripción de lo que representan y un número que identifica la secuencia de la escena.

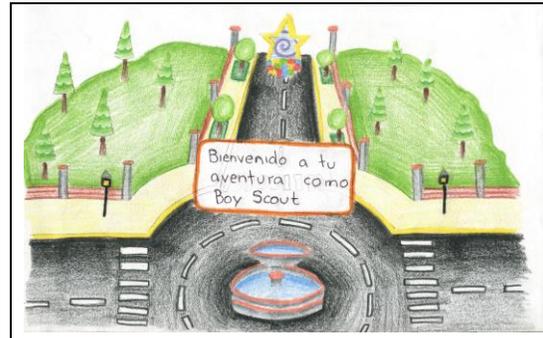
A continuación se muestra el Storyboard de la bienvenida al usuario cuando este inicia la aventura Galaxy Math dentro de la aplicación didáctica (Storyboard 1).

Cabe señalar que es conveniente desarrollar cada uno de los Storyboard de cada misión con el objetivo facilitar a los desarrolladores el trabajo, sin embargo, por cuestiones de tiempo solo se realizó el Storyboard de la misión Cosmic Apples (Storyboard 2).

Storyboard	
Nombre del proyecto: Galaxy Math Descripción: Bienvenida y presentación al usuario (Boy Scout), tanto de los escenarios que conformaran la aplicación como de los objetivos que se deberán de cumplir para la obtención de medallas.	Actividad: Bienvenida Página: 1 – 3



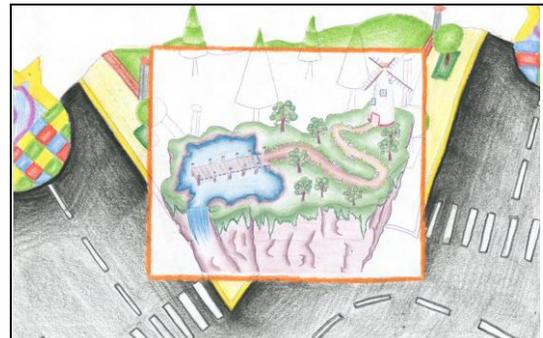
- Interfaz de inicio.



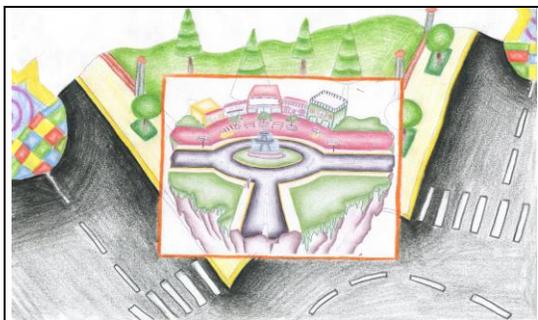
- Mensaje de bienvenida.



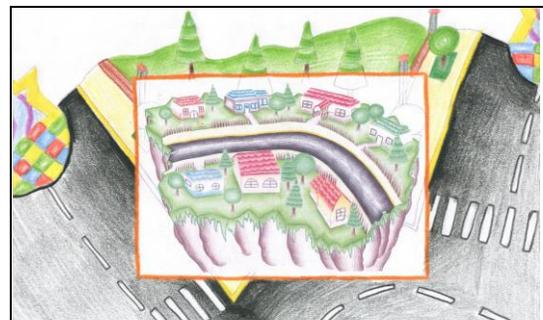
- Descripción general de la aplicación.



- Presentación del escenario "Asteroide Laguna".

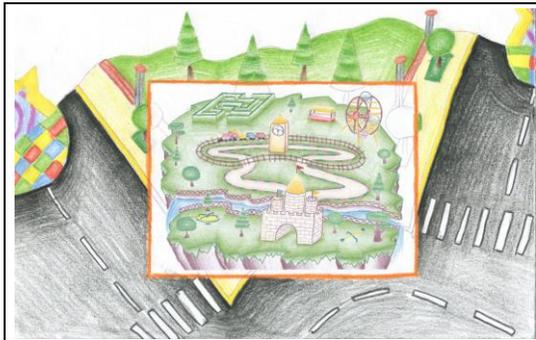


- Presentación del escenario "Asteroide Glorieta".



- Presentación del escenario "Asteroide Vecindario".

Storyboard	
Nombre del proyecto: Galaxy Math Descripción: Bienvenida y presentación al usuario (Boy Scout), tanto de los escenarios que conformaran la aplicación como de los objetivos que se deberán de cumplir para la obtención de medallas.	Actividad: Bienvenida Página: 2 – 3



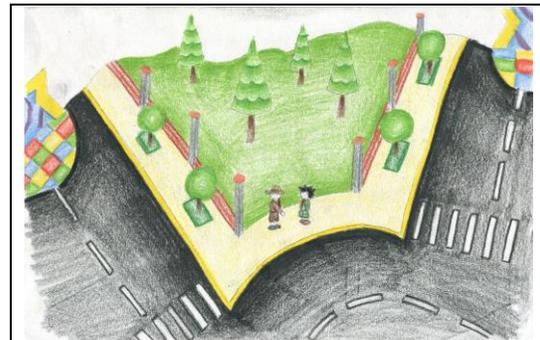
- Presentación del escenario "Parque".



- Fin de la bienvenida.



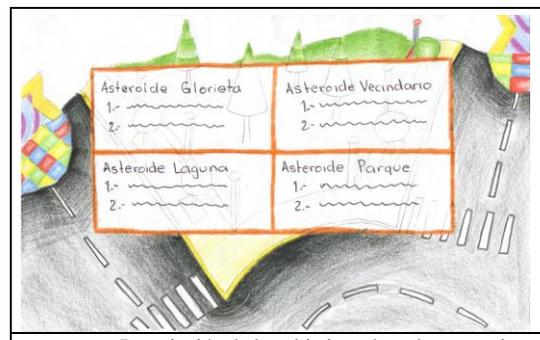
- Comienzo de la aventura y primera instrucción que recibirá el usuario para comenzar el juego.



- Acercamiento del usuario con el guía Gus.



- Saludo del guía Gus al usuario.



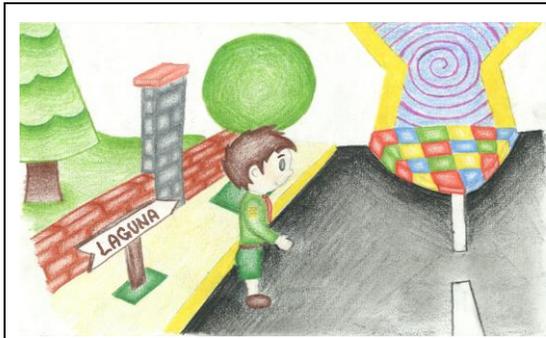
- Descripción de los objetivos de cada escenario.

Storyboard	
Nombre del proyecto: Galaxy Math	Actividad: Bienvenida
Descripción: Bienvenida y presentación al usuario (Boy Scout), tanto de los escenarios que conformaran la aplicación como de los objetivos que se deberán de cumplir para la obtención de medallas.	Página: 3 – 3



- Presentación de la instrucción para la obtención de medallas.

Storyboard	
Nombre del proyecto: Galaxy Math	Actividad: Misión Cosmic Apples
Descripción: El usuario (Boy Scout) ayudará a la abuelita Domi a recolectar 1 $\frac{3}{4}$ kg. de manzanas saltarinas que se encuentran distribuidas dentro del asteroide Laguna.	
Página: 1 – 4	



1. El usuario se dirige al teletransportador que lo llevara al asteroide Laguna.



2. El usuario entra al teletransportador.



3. El usuario llega al asteroide y tendrá que dirigirse al inicializador de misiones, el cual podrá ser identificado por el conjunto de partículas que lo componen.



4. Una vez que se encuentre ubicado en el inicializador, la aplicación comenzara a mostrar las instrucciones para resolver la misión. Y la cámara cambiara de tercera persona a primera persona.

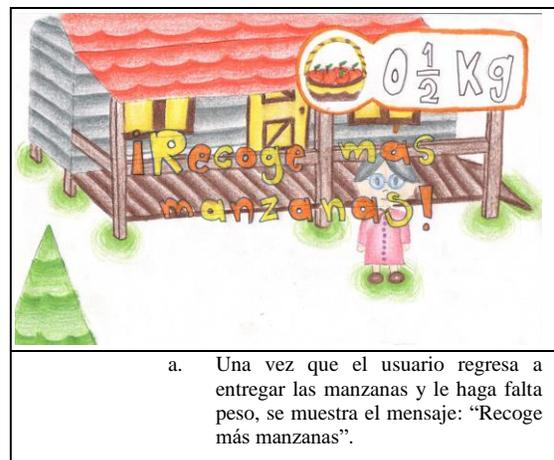
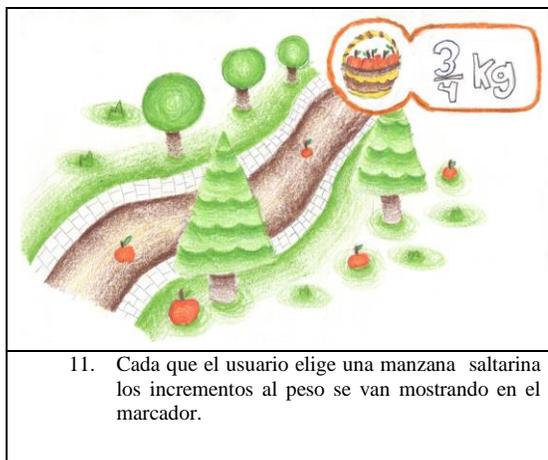
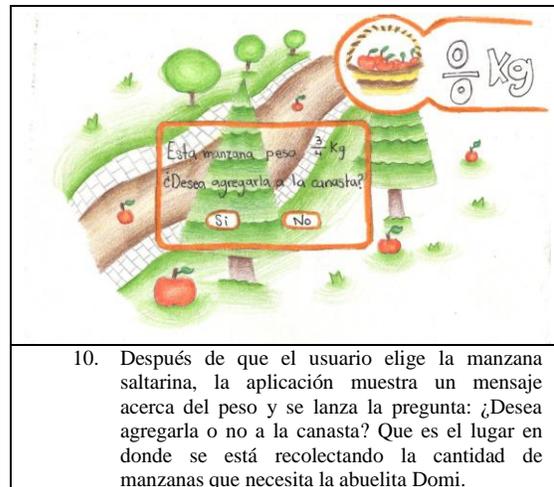
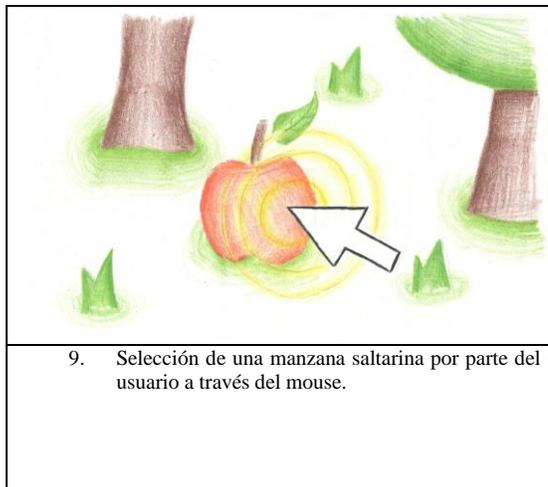


5. Mensaje que contendrá la descripción de la actividad que tendrá que resolver el usuario.



6. Mensaje de las instrucciones para que el usuario pueda tanto buscar y recolectar las manzanas saltarinas, así como para reiniciar la misión en el caso de que se requiera.

Storyboard	
Nombre del proyecto: Galaxy Math Descripción: El usuario (Boy Scout) ayudará a la abuelita Domi a recolectar $1 \frac{3}{4}$ kg. de manzanas saltarinas que se encuentran distribuidas dentro del asteroide Laguna.	Actividad: Misión Cosmic Apples Página: 2 – 4



Storyboard	
Nombre del proyecto: Galaxy Math	Actividad: Misión Cosmic Apples
Descripción: El usuario (Boy Scout) ayudará a la abuelita Domi a recolectar $1\frac{3}{4}$ kg. de manzanas saltarinas que se encuentran distribuidas dentro del asteroide Laguna.	Página: 3 – 4



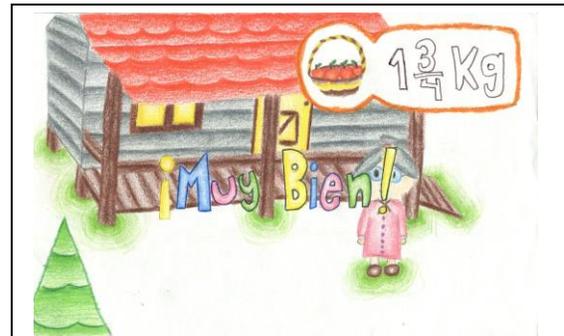
12. Si el usuario al entregar las manzanas supera el peso, se muestra el mensaje: "Superaste el peso".



12.1) Si el usuario supera el peso establecido en la misión, se mostrara el mensaje de que lo vuelva a reintentar.



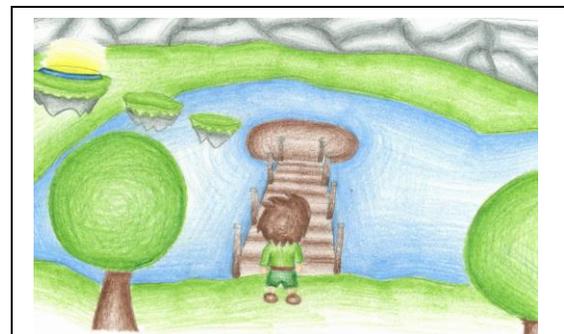
12.2) Si el usuario acepta volver a intentarlo el marcador se reiniciara en 0.



13. Cuando el usuario entrega el peso corrector, se muestra el mensaje: "Muy bien".



14. La abuelita Domi lo felicita y le indica la instrucción para recibir la medalla.



15. El usuario se dirige al teletransportador para regresar al asteroide gloriaeta.

STORYBOARD

Nombre del proyecto: Galaxy Math	Actividad: Misión Cosmic Apples
Descripción: El usuario (Boy Scout) ayudará a la abuelita Domi a recolectar $1 \frac{3}{4}$ kg. de manzanas saltarinas que se encuentran distribuidas dentro del asteroide Laguna.	
Página: 4 – 4	



16. Teletransportador para llegar al escenario "Glorieta".



17. Llegando al escenario "Glorieta".



18. Llegando con el personaje "Guía Gus"



19. Felicitaciones del personaje "Guía Gus" al usuario por concluir la actividad.



20. Entrega de la medalla y término de la actividad.

3.2.3 Descripción de componentes

En esta fase de la metodología se describieron las características de los objetos contenidos en cada escenario que conforman el proyecto. A continuación se muestran los objetos clasificados por escenarios (tabla 20 – 23).

Tabla 20. Componentes de escenario “Vecindario”

Identificación:	Tipo de plantilla:	Descripción de objetos del escenario “Vecindario”			
	Objeto:	Características:			Cantidad:
		Color	Forma	Textura	
Transportador_estrella	Teletransportador	Amarillo/negro	Estrella	Piso de colores/Metal	2
Casa 1	Casa	Beige/Naranja	Cuadrado	Concreto/Tejas/Madera	1
Casa 2	Casa	Café/Verde/Azul	Rectangular	Concreto/Tejas/Madera	1
Casa 3	Casa	Gris/Café	Cuadrado	Ladrillo/Madera	1
Casa 4	Casa	Azul/Café	Rectangular	Madera/Tejas	1
Casa 5	Casa	Rojo/Café	Rectangular	Madera/Tejas	1
Casa 6	Casa	Blanco/Naranja	Cuadrado	Concreto/Tejas/Madera	1
Casa 7	Casa	Rojo/Café	Rectangular	Ladrillo/Tejas/Madera	1
Casa 8	Casa	Verde/Blanco	Rectangular	Concreto/Tejas/Madera	1
Casa 9	Casa	Gris/Naranja	Rectangular	Ladrillo/Tejas/Madera	1
Casa 10	Casa	Verde/Gris	Rectangular	Madera/Tejas	1
Pino	Árbol	Verde/Café	Cono	Madera/Hojas	8
Arbol_redondo	Árbol	Verde/Café	Esférica	Madera/Hojas	8
Arbol_3	Árbol	Verde/Café	Amorfa	Madera/Hojas	3
Arbol_4	Árbol	Verde/Café	Redonda	Madera/Hojas	3
Cerca	Cerca	Blanco	Rectangular	Madera	21
Lampara_piso	Lámpara	Gris	Cilíndrica	Metal	14
Terreno_casas	Piso del escenario	Verde	Amorfa	Pasto	1
Street 90	Carretera/banquetas con giro de 90 grados	Gris/Blanco	Redonda	Asfalto, concreto	1
Street recta	Carretera/banquetas rectas	Gris/Blanco	Rectangular	Asfalto, concreto	1
Cofre	Cofre	Café/Naranja	Rectangular	Madera/Metal	1

Tabla 21. Componentes de escenario “Glorieta”

Identificación:	Tipo de plantilla:	Descripción de objetos del escenario “Glorieta”			
	Objeto:	Características:			Cantidad:
		Color	Forma	Textura	
Transportador_estrella	Teletransportador	Amarillo/negro	Estrella	Piso de colores/Metal	3
Fuente	Fuente de agua	Gris/Naranja	Ovoide	Concreto	1
Basurero	Basurero	Amarillo/Verde	Cilíndrica	Metal	2
Flores	Flores de colores	Amarillo/Morado/Blanco	Ovoide	Pétalos, tallo	71
Mesa	Mesa con sombrilla	Rojo/Café	Ovoide	Madera, tela, plástico	2
Arbol_redondo	Árbol	Verde/Café	Esférica	Madera, hojas	15
Pino	Árbol	Verde/Café	Cono	Madera, hojas	10

Arbol_3	Árbol	Verde/Café	Cilíndrica	Madera, hojas	2
Arbol_4	Árbol	Verde/Café	Esférica	Madera, hojas	4
Macetero	Macetero	Café	Cilíndrica	Concreto	4
Pared_rejaGrande	Pared con rejas	Café/Negro	Rectangular	Concreto, metal	8
Pared_reja_chica	Pared con rejas	Café/Negro	Rectangular	Concreto, metal	2
lampara_piso	Lámpara	Gris	Cilíndrica	Metal	18
Lamp_pared	Lámpara	Gris	Cuadrada	Metal	2
letrero_1	Letrero parque	Café	Rectangular	Madera	1
letrero_2	Letrero laguna	Café	Rectangular	Madera	1
letrero_4	Letrero vecindario	Café	Rectangular	Madera	1
local_pasteleria	Pastelería	Beige	Rectangular	Concreto	1
Pizzería	Pizzería	Blanco/Naranja	Rectangular	Concreto	1
Lechería	Lechería	Azul	Rectangular	Concreto	1
Tienda	Tienda	Blanco/Azul	Rectangular	Concreto	1
Acuario	Acuario	Beige	Rectangular	Ladrillo	1
Cofre	Cofre	Café/Naranja	Rectangular	Madera/Metal	2

Tabla 22. Componentes de escenario “Laguna”

Identificación:	Tipo de plantilla:	Descripción de objetos del escenario “Laguna”			Cantidad:
	Objeto:	Características:			
		Color	Forma	Textura	
Transportador_piso	Teletransportador	Azul/Blanco	Cilíndrica	Metal	1
Casa del lago	Cabaña	Café	Rectangular	Ladrillo, Madera	1
Apple	Manzana	Rojo	Esférica	Cascara de manzana	30
lancha	Lancha	Azul/Verde	Rectangular	Metal	1
Lamp_pared	Lámpara	Gris	Cuadrada	Metal	3
Rocas_Orillas	Rocas grandes	Gris/Verde	Rectangular	Roca, Pasto	24
Roca2	Rocas pequeñas	Gris/Verde	Rectangular	Roca, Pasto	3
Letrero_3	Letrero Glorieta	Café	Rectangular	Madera	1
Carpa_arcoiris	Pez	Azul/Verde/Amartillo	Pez	Colores	12
Molino	Molino	Café/Gris	Cilíndrica	Ladrillo, Madera	1
Muelle	Muelle	Café	Hexagonal	Madera	1
Nighttime Simple Water	Agua	Azul	Plana	Agua	1
Cofre	Cofre	Café/Naranja	Rectangular	Madera/Metal	1

Tabla 23. Componentes de escenario “Parque”

Identificación:	Tipo de plantilla:	Descripción de objetos del escenario “Parque”			Cantidad:
	Objeto:	Características:			
		Color	Forma	Textura	
Transportador_estrella	Teletransportador	Amarillo/Negro	Estrella	Piso de colores, Metal	2
Arbol_3	Árbol	Verde/Café	Cilíndrica	Madera, Hojas	7
Arbol_redondo	Árbol	Verde/Café	Esférica	Madera, Hojas	28
Arbol_4	Árbol	Verde/Café	Esférica	Madera, Hojas	4
Laberinto	Laberinto	Verde	Cuadrado	Hojas	1

Banca	Banca	Café	Rectangular	Madera	6
Entrada_castillo	Castillo	Gris/Verde/Amarillo/Rojo	Rectangular	Ladrillo	1
Lampara2	Lámpara	Negro	Cilíndrica	Metal	5
reloj_parque	Torre con reloj	Amarillo	Rectangular	Concreto	1
PuenteMadera	Puente	Café	Rectangular	Madera	1
Lampara_piso_2	Lámpara	Negro	Cilíndrica	Metal	9
Cofre	Cofre	Café/Naranja	Rectangular	Madera, Metal	2
Tren	Tren	Rojo	Rectangular	Metal	1
JuegoDisparos	Tienda de disparos	Amarillo/Rojo/Azul	Rectangular	Tela	1
Rueda_fortuna	Rueda de la fortuna	Azul/Morado/Verde	Redonda	Metal	1
Carrito hamburguesas	Carrito de hamburguesas	Gris/Naranja	Rectangular	Metal, Plástico	1
Agua rio	Agua	Azul	Plana	Agua	1
Camino parque	Camino	Café	Redonda	Tierra	1

3.2.4 Modelado de eventos

En esta etapa se describieron los eventos que serán detectados por la aplicación didáctica a través de los comportamientos que el usuario o los componentes realicen. Cabe mencionar que el modelo de eventos tiene una estrecha relación con el modelo de navegación, debido a que los eventos generados son derivados del rumbo que el usuario elija tomar y de los componentes que intervienen en su aventura.

Debido a que es una aplicación de RV no inmersiva o de escritorio, solo son detectados por el sistema eventos del usuario que se generan por medio del mouse o teclado (tabla 24).

Tabla 24. Eventos del usuario

Tipo de plantilla:	Descripción de eventos durante la navegación del usuario dentro de los entornos	
Eventos	Acción	
Mover mouse	Girar cámara del motor de representación sobre el eje de las X y Y.	
Clic izquierdo	Seleccionar componentes	
Tecla de dirección izquierda	Desplazamiento del personaje hacia la izquierda	
Tecla de dirección derecha	Desplazamiento del personaje hacia la derecha	
Tecla de dirección arriba	Desplazamiento del personaje hacia adelante	
Tecla de dirección abajo	Desplazamiento del personaje hacia atrás	
Tecla espaciadora	Salto del personaje	

Y los eventos detectados por los componentes tridimensionales generados por el usuario (personaje principal) se describen en la tabla 25.

Tabla 25. Eventos de componentes

Componente Tridimensionales	Eventos detectados	Condiciones	Acción
Inicializador de misiones	Colisión con el personaje	Si la misión no está activa	Activar misión
		Si la misión esta activa y el usuario desea comprobar sus avances	Comprobar misión
Teletransportador	Colisión con el personaje	Siempre se cumple	Cargar nuevo escenario
Cofre	Clic izquierdo del mouse	Si el cofre está cerrado	Abrir tapa cofre
Casa 7	Colisión con el personaje	Si aún no recibe el pedido de leche	Mostrar leches disponibles
Casa 10	Colisión con el personaje	Si aún no recibe el pedido de leche	Mostrar leches disponibles
Manzana saltarina	Clic izquierdo del mouse	Si la misión esta activa	Sumar peso al marcador
Monedas de fracción	Clic izquierdo del mouse	Si el usuario dio clic en una moneda correcta	Sumar punto
		Si el usuario dio clic en una moneda incorrecta	Restar punto
Puerta de laberinto	Colisión con el personaje	Si está cerrada	Mostrar pregunta
Teletransportador de piso	Colisión con el personaje	Siempre se cumple	Cargar escenario Glorieta
Pez	Clic izquierdo del mouse	Si la misión esta activa y el usuario no ha dado clic sobre otro pez	Pescar y Comparar peso
	Cursor sobre su maya	Si la misión esta activa	Mostrar peso
Pizza	Clic izquierdo del mouse	Si la misión esta activa	Recoger pizza
	Cuando se suelta el clic izquierdo	Si la misión esta activa	Soltar pizza

3.3 Fase de Desarrollo

En esta fase se desarrollaron los recursos artísticos y tecnológicos en el siguiente orden:

- Se modelaron y texturizaron los componentes tridimensionales
- Se animaron los componentes tridimensionales si lo requerían
- Se integraron en Unity conforme se iban produciendo
- Se programaron los scripts necesarios para cada misión

Cabe mencionar que si un recurso artístico no cumplía con las características establecidas, se regresaba al paso en el que resulto deficiente para corregirlo y cumplir con lo establecido.

3.3.1 Producción de recursos artísticos

En esta fase se modelan los objetos, texturas, videos y secuencias de audio con herramientas especializadas como “Autodesk Maya” para generar los componentes tridimensionales, “Gimp” y “Crazy Bump” para las texturas.

Para generar los componentes tridimensionales se utilizaron dos métodos:

3.3.1.1 Generación de un Low Poly a partir de un High Poly

En la producción de los recursos artísticos se utilizó la herramienta llamada “*Transfer Maps*” (Transferencia de mapas). Esta herramienta permite generar los siguientes mapas (Figura 35):

- ***Normal Map*** (Mapa de normales). Mapa utilizado para simular una superficie 3D.
- ***Displacement Map*** (Mapa de desplazamiento). Este tipo de mapa es utilizado para desplazar los puntos de la superficie de acuerdo a los valores de esta textura generando geometría al objeto. Este tipo de texturas resulta muy cara ya que genera excesiva geometría.
- ***Diffuse Map*** (Mapas de color). Este tipo es utilizado para proporcionar los aspectos de color a un objeto.
- ***Shaded Map*** (Mapa de sombreado). Este tipo de mapa contiene información de color y sombreado de un objeto de acuerdo a las luces utilizadas.
- ***Alpha Map*** (Mapa de transparencia). Este mapa es utilizado para generar texturas con transparencias.
- ***Ambient Occlusion Map*** (Mapa de oclusión ambiental). Este mapa es similar al mapa de sombreado, tomando en cuenta la reflexión de la luz y la oclusión de los objetos cercanos a la superficie.

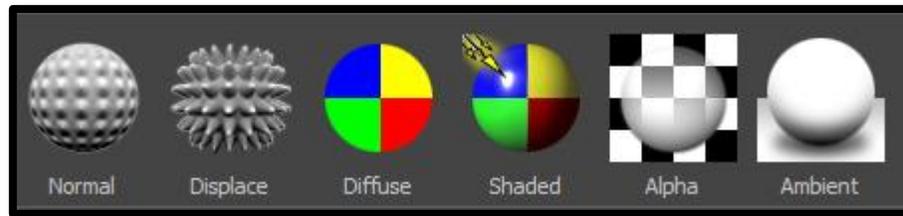


Figura 35. Tipos de mapas de salida que genera Transfer Maps

Estos mapas se generan a partir de un modelo *High Poly* que transmite sus propiedades a otro objeto optimizado geoméricamente, obteniendo así un modelo con apariencia similar al objeto *High Poly* que consume menos recursos en el equipo de cómputo sobre el cual se renderiza. A continuación se describe el proceso de creación de estos mapas, el cual fue utilizado para el desarrollo de la mayoría de los objetos de la aplicación.

En este ejemplo se describe paso a paso como se realizó la generación de mapas de color y normales para obtener un objeto *Low Poly*.

- Se modeló el objeto tridimensional con un alto nivel de detalle (LOD por sus siglas en inglés *Level Of Detail*), a este objeto se denomina *High Poly* debido a que tiene un gran número de polígonos.

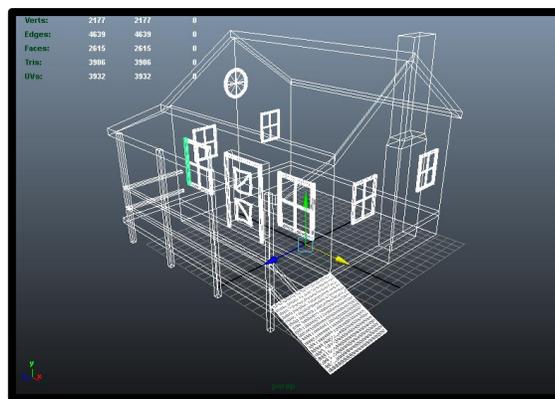


Figura 36. Objeto con alto LOD.

- Se modeló otro objeto con el menor número de polígonos posibles o un bajo LOD con la técnica de modelado poligonal. Este objeto tiene las mismas dimensiones que el modelo *High Poly*, este objeto con bajo número de polígonos se denomina *Low Poly*.

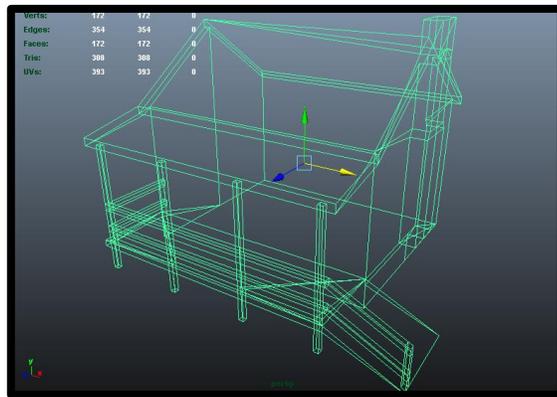


Figura 37. Objeto con bajo LOD.

- Se realizó la proyección de la superficie de la cabaña *Low Poly* en un plano 2D. A este proceso se le conoce como mapeado UV, en el cual la letra “U”, hace referencia al eje “X” y letra “V” al eje “Y”.

El resultado de este proceso se conoce como mapas UV, los cuales envuelven la superficie de los polígonos.

Para realizar este proceso, se seleccionaron las caras de la cabaña que se deseaban proyectar y posteriormente se accedió al menú “*Create Uvs>Planar mapping*” (figura 38, lado izquierdo), para realizar la proyección. Posteriormente se abrió el Editor de Uvs dando clic en “*Edit Uvs > UV Texture Editor*” (figura 38, lado derecho), donde se acomodaron las proyecciones realizadas.

En este paso se acomodaron los UV de modo que no se superpongan unos con otros para obtener mapas óptimos del *low poly* (figura 38, lado derecho).

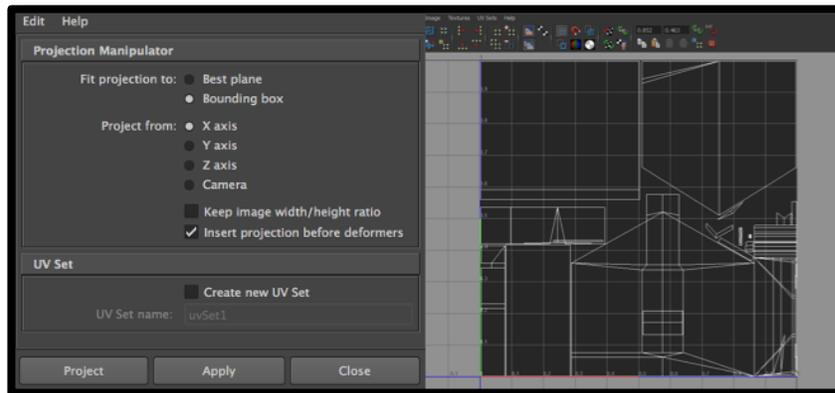


Figura 38. Mapa UV de Low Poly

- Se realizó el mapeado del objeto *High Poly* con una imagen de referencia llamada “*Checker UV*” para tener una vista más precisa de cómo se iban a mapear las texturas a utilizar sobre el objeto.

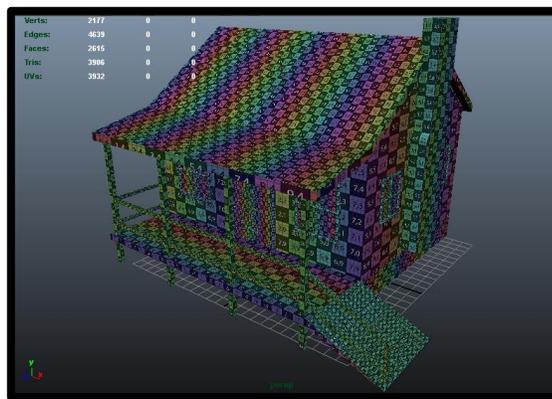


Figura 39. Mapeado de referencia con checker UV.

Nota. Se debe generar un material nuevo para cada parte del objeto que requiera una textura diferente.

- Se asignaron las texturas correspondientes a cada material generado en el proceso anterior.

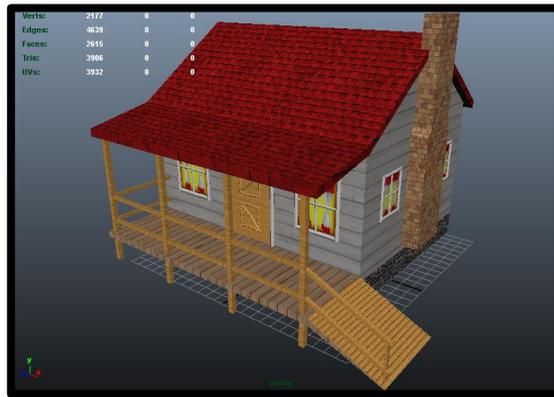


Figura 40. High Poly texturizado

- Se colocó el *Low Poly* en la misma posición espacial del *High Poly*.

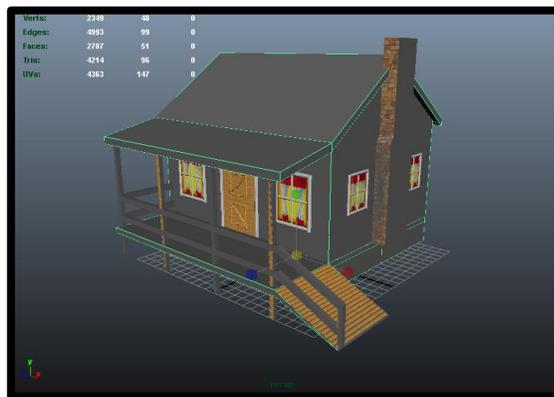


Figura 41. High poly y low poly en la misma posición espacial.

- Se accedió a las herramientas de *Rendering* y abrir la herramienta “*Transfer maps*”.

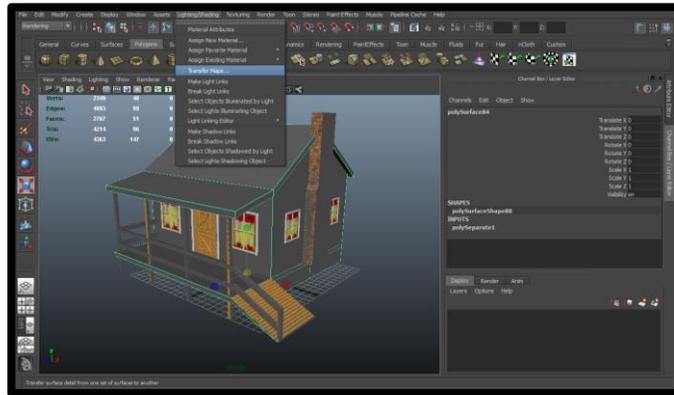


Figura 42. Herramienta Transfer Maps.

- Se seleccionó los objetos que componen el *High Poly* y establecerlos como “*Source Meshes*” (objeto del cual se quieren copiar sus propiedades), dando clic en Add Selected y el objeto *Low Poly* como “*Target Mesh*” (objeto al que se le desea aplicar las características del *Source Mesh*).

Nota. Por defecto al tener seleccionado el low poly, maya lo establece como target mesh al abrir la herramienta.

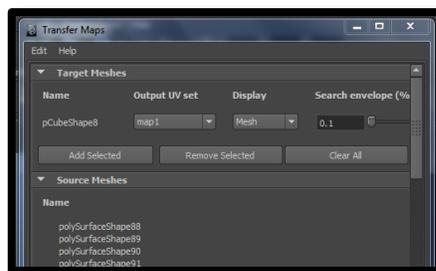


Figura 43. Source y Target Meshes.

- En *Output maps* (Mapas de salida) se seleccionó *Diffuse Map* y *Normal Map*. A continuación se describe su funcionalidad y como fueron configurados los parámetros en los componentes tridimensionales.

Diffuse Map

Este tipo de mapa contiene uno o más materiales que son agregados dentro de un solo canal de color. El mapa se generó de acuerdo al *UV set* del *Target Mesh*. La generación de este mapa es útil para reducir el número de materiales utilizados en un objeto.

Normal Map

Este mapa captura la información de las normales de la superficie de los *Sources Meshes* y es generado de acuerdo al mapeado del objeto *Low Poly*. En este mapa solo se escribió la ruta de almacenamiento y el formato de la imagen.

Para el objeto creado en esta sección se seleccionaron y configuraron los siguientes parámetros:

- La opción *Include materials* se deselecciono porque los materiales de los *Source Meshes* no tienen *bump mapping* que puedan transferir su información al nuevo mapa.
- En el parámetro *map space* se seleccionó *object space* porque el objeto no tendrá movimiento en la aplicación y no *tangent space* ya que se utiliza cuando el objeto tendrá animación.
- En *Use Maya Common Settings* se estableció que el mapa utilice los parámetros de la sección *Maya Common Output*.

De igual manera se designó la ruta donde se almacenó la imagen así como el formato de la misma.

Las dos imágenes se dejaron con *Use Maya Common Settings* para que tuvieran los mismos parámetros en sus dimensiones.

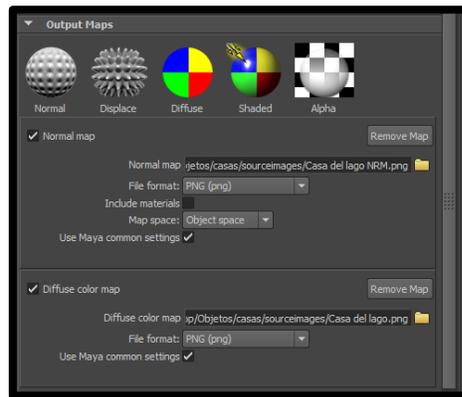


Figura 44. Mapas de salida.

- En la sección *Connect Output Maps* se especifica cómo se desea que el mapa a crear aparezca en el *Low Poly*. Para este ejemplo se dejó seleccionado *Connect maps to Shader* para que Maya asigne los mapas creados al material del *Low Poly* automáticamente; al canal *diffuse map* le asigna el mapa de color y el mapa de normales al canal del *bump map*. En *Connect maps to*, se dejó seleccionado *New Shader* para que cree un nuevo material.

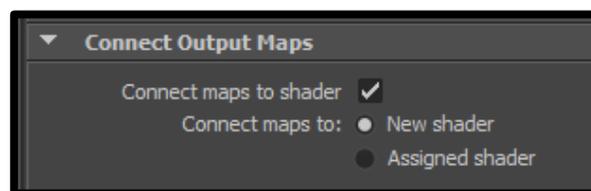


Figura 45. Asignación de los mapas de salida.

- En *Maya Common Output* se configuran las dimensiones de los mapas, el tipo de transferencia, calidad del muestreo, filtro, relleno de píxeles y las caras que son simétricas en el *Source Mesh*.

1. En las dimensiones de los mapas se configuran los siguientes parámetros:
 - a. *Map Width* es la medida en píxeles a lo ancho del mapa
 - b. *Map height* es la medida de la altura del mapa en píxeles
 - c. *Keep aspect ratio* es para mantener el aspecto de la textura, se escalarán proporcionalmente altura y anchura

2. En el tipo de transferencia (*Transfer in*) se puede seleccionar uno de los tres tipos:
 - a. *World Space*. Se usa cuando el low y high poly tienen dimensiones diferentes. Los dos objetos deben estar situados en el mismo lugar espacial para obtener los mapas de manera óptima.
 - b. *Object Space*. Este parámetro es útil cuando los objetos son de las mismas dimensiones, y los objetos pueden estar separados al realizar la transferencia pero deben tener congeladas sus posiciones, rotaciones y tamaños.
 - c. *UV Space*. Se utiliza cuando los objetos son de diferentes proporciones o diferentes formas. Este parámetro ignora todas las transformaciones de los objetos y toma como referencia los UVs.

- En la calidad de muestreo se especifica el número de muestras por píxel del *High Poly* para el mapa de salida. Si se aumenta el número de muestreo, la calidad del mapa de salida será alta, siempre y cuando las texturas del *High Poly* sean de buena calidad. Se deben tener en cuenta los recursos computacionales, ya que para una calidad alta se tarda más en generar los mapas.

- El Tamaño del filtro controla la interpolación de cada pixel en el mapa de salida. Para este objeto se utiliza un tamaño igual a 3 para obtener un mapa más nítido.
- El tipo de filtro es utilizado para eliminar el ruido o el pixelado de las líneas. El tipo Gaussian es un filtro ligeramente más suave. Los filtros *Triangular* y *Box* son suave y muy suave respectivamente.
- El parámetro *Fill texture seams* agrega pixeles alrededor de los UVs para evitar un mapeado no deseado o transparencias en los resultados.
- El siguiente parámetro se deshabilito ya que no es útil para este modelo, es más usado cuando se modela un personaje porque tienen partes del cuerpo que son simétricas y podrían compartir los mismos mapas en ambos lados por ejemplo la cara.

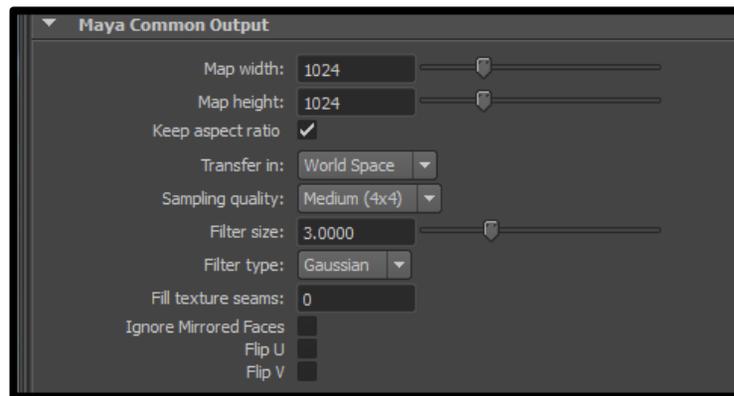


Figura 46. Maya Common Output.

- Las opciones avanzadas son utilizadas para indicarle a Autodesk Maya como debe realizar la búsqueda de la textura por medio de la envoltura generada. La configuración para este ejemplo se muestran en la figura 47.

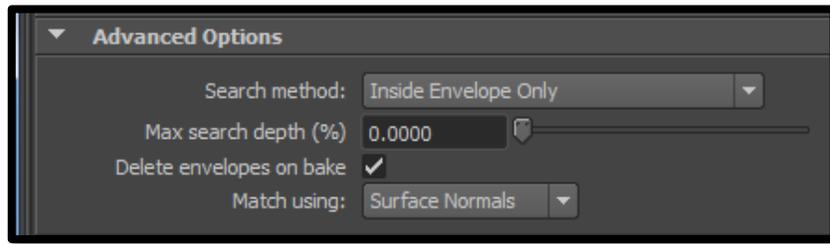


Figura 47. Opciones Avanzadas.

A continuación se describen los cuatro parámetros:

- El método de búsqueda se refiere a como actuara la envoltura al buscar las texturas. La figura 48 muestra los tres casos en los que se puede usar.

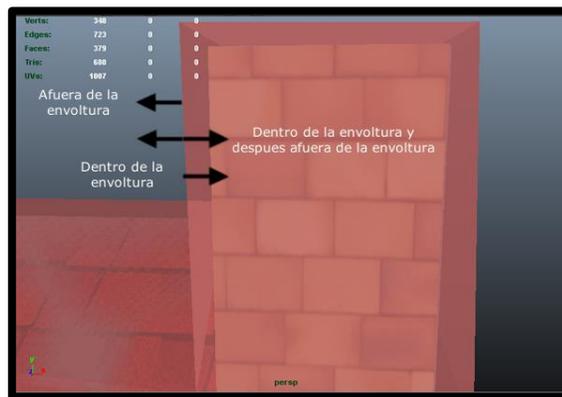


Figura 48. Método de búsqueda

- El porcentaje de la profundidad de búsqueda establece el límite de distancia de búsqueda.
- *Delete Envelope on Bake* se dejó activado para que al finalizar la transferencia de mapas se eliminara la envoltura ya que esta se crea como un objeto hijo del Low Poly (figura 49).
- El parámetro *Match Using* tiene dos opciones:

- *Geometry Normals*. Se usa para superficies de bordes suaves del Low Poly.
- *Surface Normals*. Se usa para superficies de bordes duros del Low Poly.

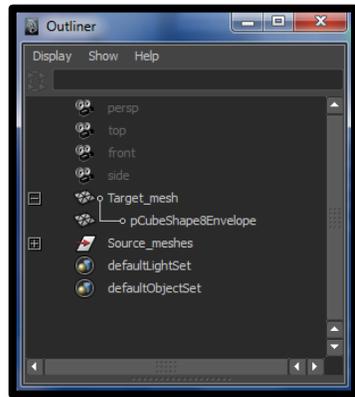


Figura 49. Sub objeto generado por la envoltura.

- En la sección de *Display* la opción *Both* es utilizada para que las caras de la envoltura pueda ser manipulada y personalizada mediante transformaciones. En este caso, se modificó la envoltura hasta que cubriera completamente el *High Poly*.

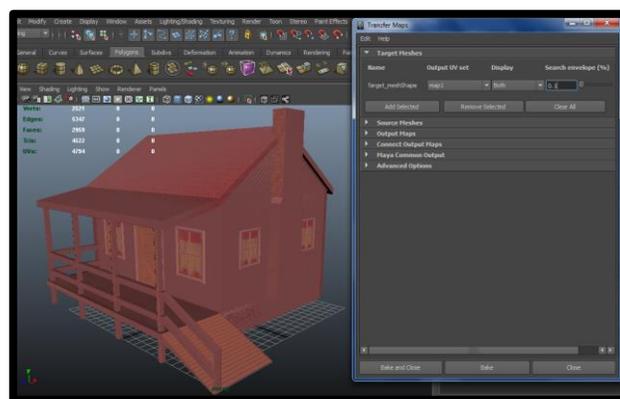


Figura 50. Personalización de la envoltura.

- Para finalizar se da clic en *Bake and close* para generar los mapas y cerrar la herramienta al finalizar la creación. Los resultados se pueden observar en las figuras 51 y 52.



Figura 51. Diffuse Map de salida

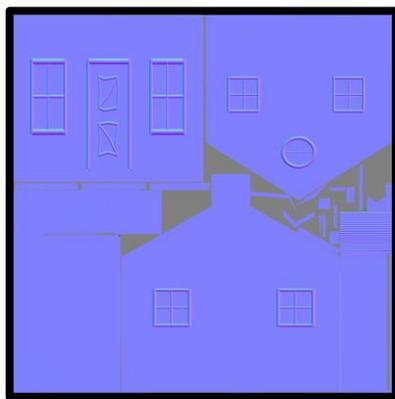


Figura 52. Normal Map de salida

Los resultados visuales son similares en ambos objetos, en las figuras 53 y 54 se observa.

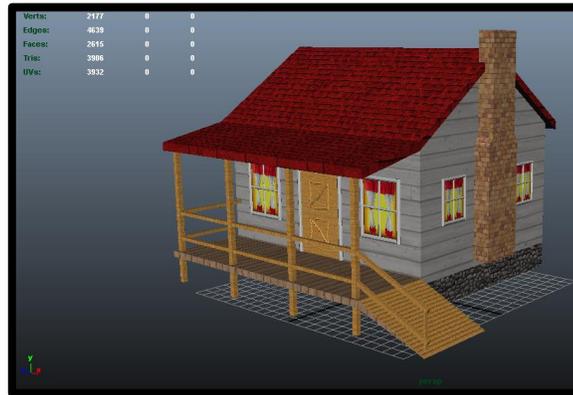


Figura 53. High Poly

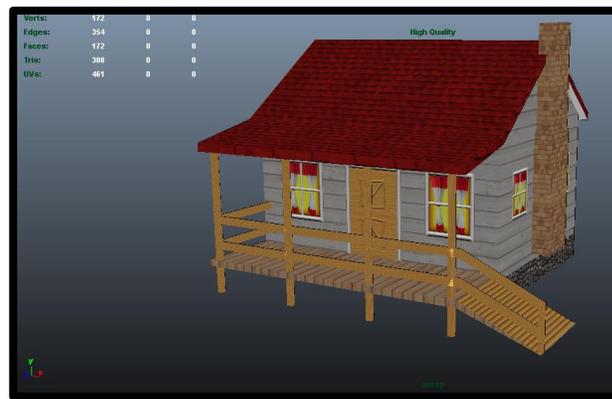


Figura 54. Low Poly

3.3.1.1.1 Ventajas de un objeto Low Poly contra high Poly

La herramienta *Transfer Maps* además de facilitar y reducir en gran parte la generación de distintos tipos de mapas para un objeto de bajo poligonaje, proporciona grandes ventajas como las que se observan en la tabla 26.

Tabla 26. Comparación entre High Poly y Low Poly.

	Low Poly	High Poly
Número de vértices	172	2177
Número de caras	172	2615
Número de bordes	354	4639
Número de UVs	461	3932
Peso del archivo FBX	 cabaña_lowpoly.fbx FBX File 357 KB	 cabaña_highpoly.fbx FBX File 5,02 MB
Número de materiales	1	8
Tiempo de renderizado en Maya	1 segundo	2 segundos

El uso de esta herramienta demuestra que es posible la obtención de un objeto con menos polígonos con características visuales similares a las de un objeto *High Poly*. Optimizando en gran medida el número de polígonos y de texturas para representar los materiales necesarios y por consiguiente el tamaño del archivo FBX.

Cabe mencionar que esta técnica es útil para el desarrollo de videojuegos y entornos de RV ya que proporciona un mejor rendimiento debido a que el uso de recursos computacionales se minimiza.

3.3.1.2 Creación de una textura a partir de varias imágenes

Este método consiste en la unión de dos o más imágenes para producir un collage como una sola textura y así reducir el número de texturas de un objeto.

Este método es útil y sencillo de realizar en comparación con el anterior, además se puede reducir el tiempo de producción de recursos artísticos. Esto no quiere decir que sea aplicable en todos los casos o el más conveniente, ya que cada método tiene sus pros y sus contras.

Para demostrar la funcionalidad de este método partiremos del modelo tridimensional de un pino, el cual tiene dos texturas independientes que representan el tronco y las hojas (Figura 55).



Figura 55. Modelo tridimensional de un pino con dos texturas.

A continuación se describe paso a paso como se fue desarrollado el objeto tridimensional utilizando este método.

1. Se modeló un objeto con el menor número de polígonos posibles.

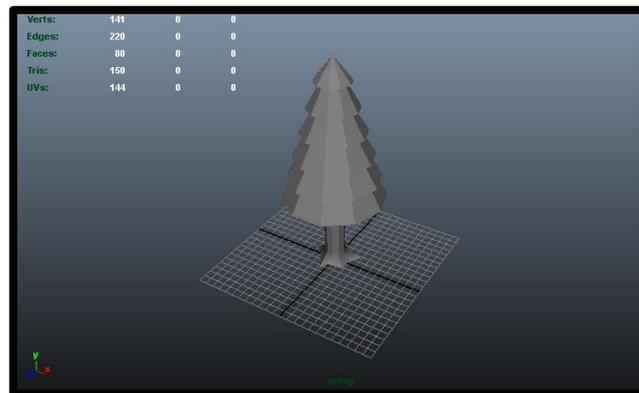


Figura 56. Modelo tridimensional de un pino sin texturas.

2. Se crearon y buscaron en páginas de imágenes de internet, las imágenes que fueron utilizadas como texturas para el objeto. En este caso se utilizaron dos,

una que se descargó de la red para el tronco del árbol y la otra que fue creada para las hojas.

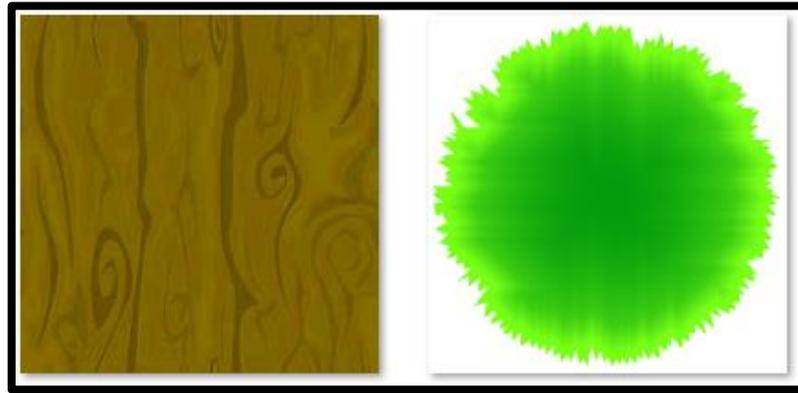


Figura 57. Texturas de un pino.

Nota. Se recomienda que las imágenes descargadas de internet sean de páginas que ofrezcan contenido gratuito, ya que en caso contrario se pueden violar los derechos de autor.

3. Se utilizó el editor de imágenes Gimp, en el cual se creó una capa con las dimensiones 512 x 512 píxeles, esta capa fungió como referencia de tamaño para acomodar las imágenes dentro de ella. Para este ejemplo se añadió un canal alfa ya que la textura de las hojas lo requiere.

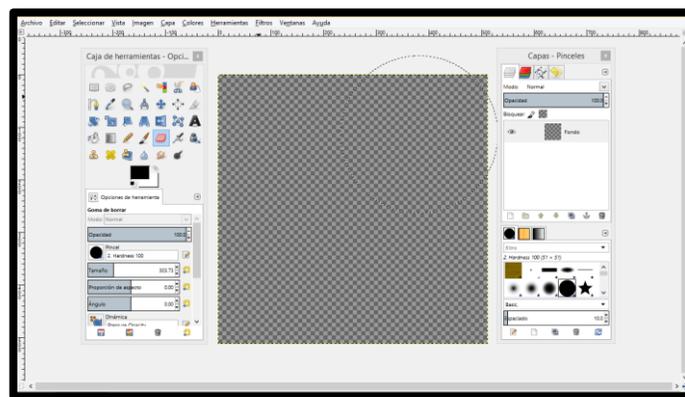


Figura 58. Creando una capa nueva en Gimp.

Nota. Se recomienda que las imágenes utilizadas como texturas tengan una relación de aspecto 1:1 para poder realizar un mejor mapeado de los modelos tridimensionales.

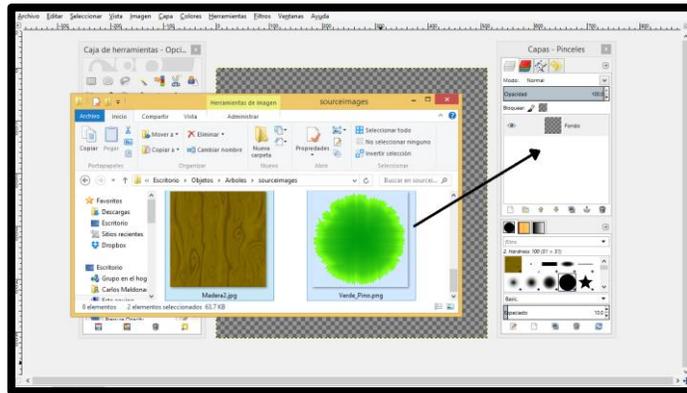


Figura 59. Importando imágenes a Gimp.

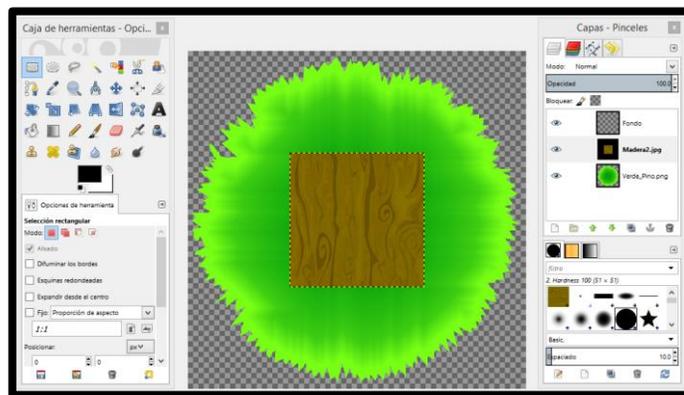


Figura 60. Imágenes agregadas como capas nuevas.

4. Se acomodaron las capas de las imágenes de modo que ambas cubrieran la capa de referencia, utilizando las herramientas de la ventana de caja de Gimp.

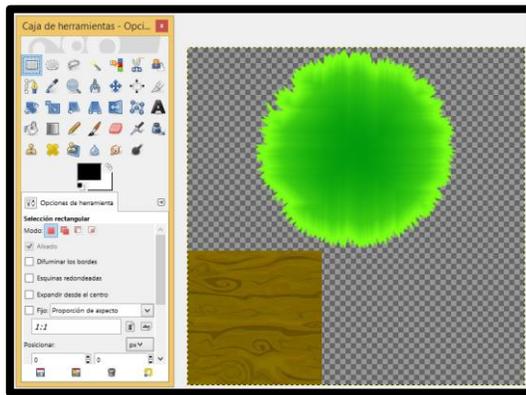


Figura 61. Caja de herramientas de Gimp.

5. Para rellenar el espacio sobrante se escaló la capa que tiene la imagen de madera a 512 pixeles para que lo cubriera.

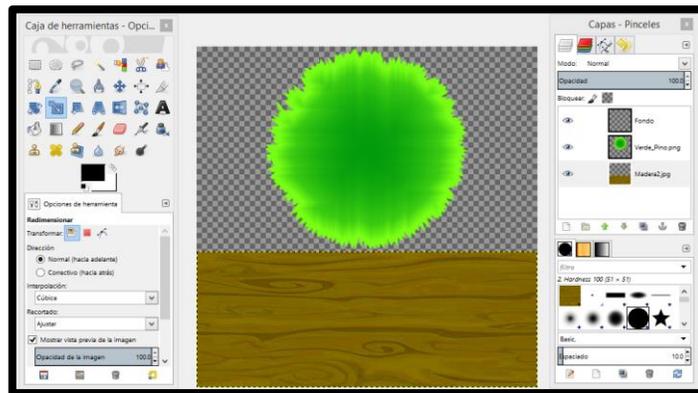


Figura 62. Ajustando imágenes en la capa de referencia.

6. Se exportaron las capas como imagen PNG, para que guarde los valores de transparencia (Figura 63) y a esta imagen se le llama Pino.png (Figura 64).

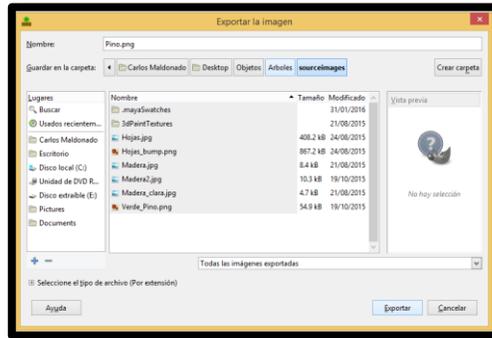


Figura 63. Exportando imagen en Gimp.

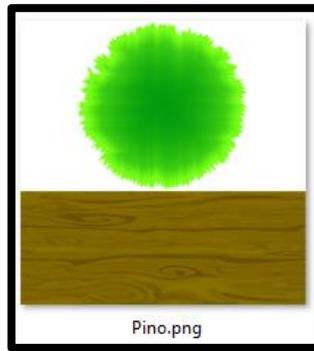


Figura 64. Textura resultante de la unión de dos imágenes.

7. En Autodesk Maya se asignó un nuevo material tipo Lambert a modelo tridimensional del pino (Figura 65), posteriormente se le dio el nombre “Pino” (Figura 66).



Figura 65. Creación de un material tipo Lambert.

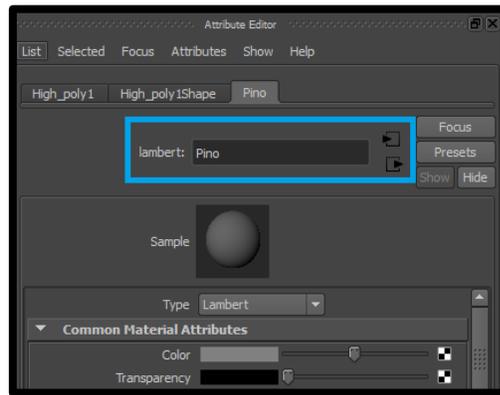


Figura 66. Asignación de nombre a un material tipo Lambert.

8. Se dio clic en el icono “Checker Box” que se encuentra a la derecha del atributo color.

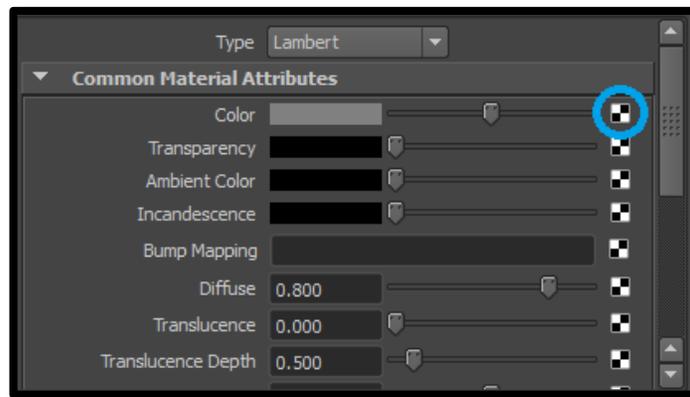


Figura 67. Atributos de un material Lambert.

9. Se abrió una ventana llamada “Create Render Node”, en la cual se muestran todos los tipos de conexiones que se pueden hacer a un material, en este caso, se seleccionó un tipo “File” para poder asignar una imagen, guardada en la carpeta “Image Source” de nuestro proyecto creado.

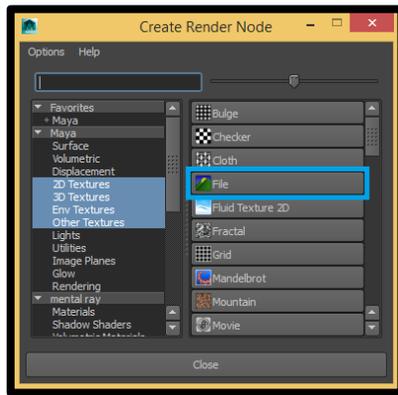


Figura 68. Selección del tipo de textura.

10. A continuación se mostraron los atributos del archivo en el editor de atributos y posteriormente se dio clic en el botón “*Image Name*” (Nombre de imagen) que se encuentra en el “*Attribute Editor*” (Editor de atributos).

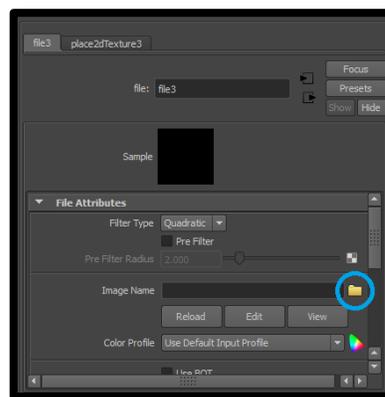


Figura 69. Atributos del archivo del canal de color.

11. Posteriormente se abrió una ventana en la cual se seleccionó la imagen generada en el paso 7.

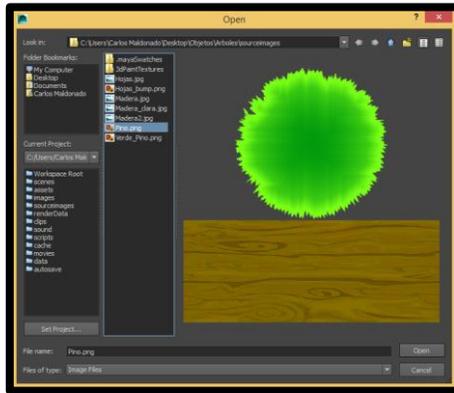


Figura 70. Selección del archivo para el canal de color.

En la figura 71 se observa que la textura no está bien mapeada, ya que aún no se mapeaban los UVs del pino.

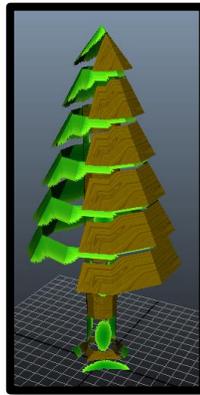


Figura 71. Modelo tridimensional no mapeado.

12. Se prosiguió a mapear los UVs del pino, para lo cual, se abrió el Editor de UVs.

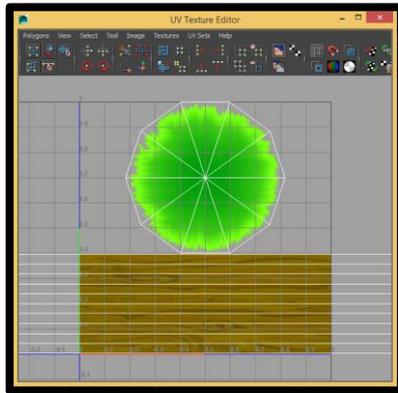


Figura 72. Mapeado de UVs.

La figura 73 muestra el resultado final, en el cual se puede observar que es idéntico al modelo inicial que tenía dos texturas.

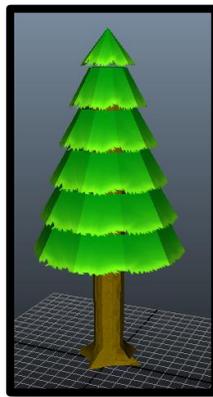


Figura 73. Modelo tridimensional de un pino con una textura.

3.3.1.3 Creación de personajes en DAZ Studio

Para la creación de los personajes que integran la aplicación se utilizó el programa DAZ Studio, ya que ayudó a agilizar el tiempo de modelado de los personajes, así como las animaciones. Cabe mencionar que para la creación de los personajes se

compró un paquete en la tienda en línea debido a los recursos de tiempo del proyecto, además de facilitar el modelado y creación de huesos de los personajes. Este paquete se llama “Toon Generations Mega-Bundle”, el cual contenía lo siguiente:

- 3 Toon Generations Base. Figura base de personaje, modificadores corporales y modificadores faciales.
- 4 Toon Generations Clothig Pack 1. Ropa para el personaje y zapatos.
- 5 Toon Generations Clothig Pack 2. Ropa para el personaje y zapatos.
- 6 Toon Generations Family Pack 1. Accesorios (lentes, bolsas de mano, oso de peluche, maletín).
- 7 Toon Generations Family Pack 2. Accesorios (lentes, bigote, aretes).
- 8 Toon Generations Hair Pack 1. Cabelleras de hombre y mujer con diferentes estilos.
- 9 Toon Generations Hair Pack 2. Cabelleras de hombre y mujer con diferentes estilos.
- 10 Toon Generations Strange Pack. Accesorios y ropa (para zombi, mago, superhéroe y hippie).

A continuación se describe el proceso que se llevó a cabo para obtener el paquete y crear a los personajes.

- Para realizar la compra, se accedió a la página oficial de DAZ Studio y posteriormente a la sección “Shop” (tienda) donde se buscó el paquete, el cual fue adquirido con una oferta del 26% de descuento, resultando un costo final de 74.96 dólares (figura 74).

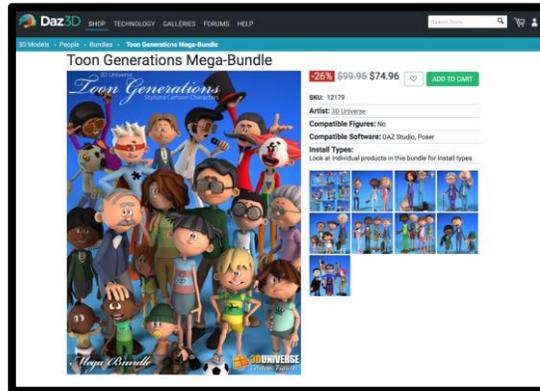


Figura 74. Tienda en línea de DAZ Studio.

- Una vez importado el paquete en DAZ Studio, se accede a su contenido en la sección “Products” (productos) del panel “Content Library” (Biblioteca de contenido) (figura 75).



Figura 75. Biblioteca de contenido de DAZ Studio.

- Ahora se procede a crear la figura base del personaje, para esto nos dirigimos al paquete “Toon Generation Base” donde se encuentra el elemento “Base Figure” (figura base), al cual se le da doble clic para agregarlo a la escena (figura 76).

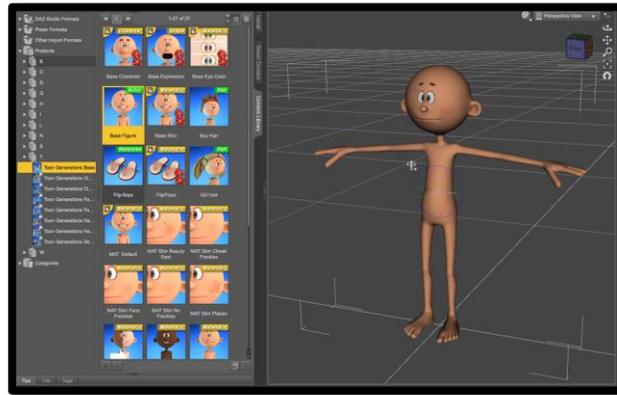


Figura 76. Figura base del paquete “Toon Generations”.

- Una vez que se tiene la figura base, lo que sigue es agregar ropa, accesorios y zapatos, los cuales se encuentran dentro de los paquetes Toon Generations Clothing Pack 1 y 2. Para adherirlos a la figura base se da doble clic sobre el elemento que se desea (figura 77).



Figura 77. Figura base con ropa y accesorios.

En la figura 77 se observa que en el panel “Scene” (escena), se agregaron en forma jerárquica los elementos que hasta el momento conforman el personaje.

- Para modificar los rasgos faciales y corporales, se accede al panel “Content Library” donde se encuentran los modificadores de tipo “Material” y “Shape” (se distinguen por tener un icono de dos dados rojos) (figura 78), los cuales permiten hacer los cambios a los elementos que se encuentran en la escena. Cabe mencionar que para aplicar los cambios, se debe seleccionar en el panel “Scene” el elemento deseado y posteriormente dar doble clic sobre el modificador.



Figura 78. Modificadores del paquete “Toon Generations”.

- Una vez que el personaje tiene los rasgos faciales y corporales adecuados, se cambia el color de la ropa. Para esto se accede al panel “Surfaces” (superficies) y se selecciona dentro del panel “Scene” el elemento al que se le desee cambiar el color. Posteriormente se puede observar que se despliegan las propiedades del elemento seleccionado, dentro de las cuales se encuentra “Diffuse Color”, esta propiedad permite cambiar el color del material, en este caso se cambió a color café la parte superior del suéter (figura 79).

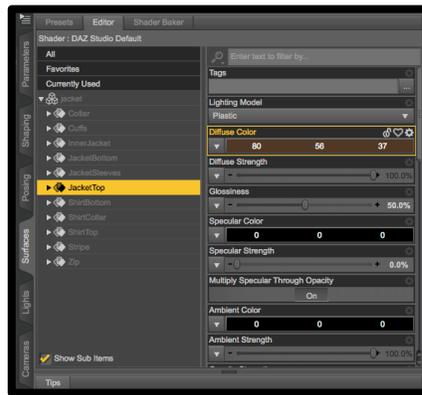


Figura 79. Propiedades de los elementos que integran el personaje.

La figura 80 muestra el resultado después de haber realizado los cambios necesarios de color y forma en el personaje.



Figura 80. Personaje “Charlie” en Daz Studio.

3.3.1.4 Optimización con Simplygon

Una parte esencial en el desarrollo de aplicaciones para tiempo real, es la optimización, ya que con ella se mejora el rendimiento de estas en el equipo de

cómputo. En el presente proyecto se utilizó la herramienta llamada “Simplygon”, la cual fue descargada como plugin para Unity.

A continuación se describe el proceso utilizado en los personajes para reducir el número de polígonos:

1. Acceder a Simplygon desde el menú Window (figura 81).

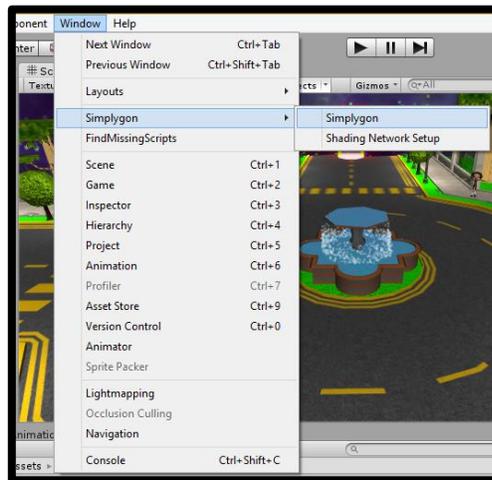


Figura 81. Accediendo a Simplygon.

2. Iniciar sesión con una cuenta de Simplygon,



Figura 82. Inicio de sesión en Simplygon.

En la figura 83 se puede observar que después de iniciar sesión, Simplygon cuenta con 4 secciones:

- Quick Start. Para hacer una optimización de manera rápida, solo ajustando el porcentaje de polígonos que se desea conservar del personaje original.
- Advanced Settings. Para hacer una optimización con una configuración avanzada.
- Manage Jobs. Para mostrar el progreso de la optimización.
- User Account. Para mostrar información del usuario y cerrar sesión.

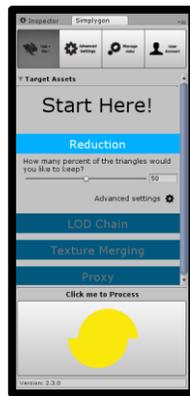


Figura 83. Panel Simplygon.

3. Una vez iniciada la sesión, se selecciona el personaje a optimizar del panel “Project” y Simplygon mostrará el número de polígonos que tiene el elemento seleccionado (figura 84).



Figura 84. Número de polígonos del personaje Guía Gus.

- Ahora se ajusta el porcentaje que los polígonos que se desean conservar en el personaje optimizado, tomando como referencia el número de polígonos del personaje original (figura 85).

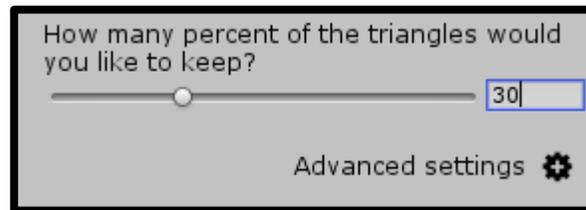


Figura 85. Porcentaje de optimización.

- Dar clic en el botón "Clic me to Process" (clic para procesar) para comenzar con la reducción de polígonos. Se puede observar el progreso en la sección "Manage Jobs (figura 86).

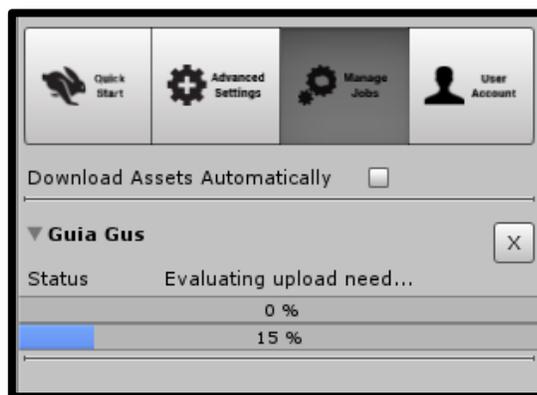


Figura 86. Optimizando al personaje en el servidor.

- Dar clic en el botón download (Descargar), este paso se puede evitar si antes de comenzar con la optimización se habilita el check box "Download Assets Automatically" (Descargar assets automáticamente). Cabe resaltar que este

paso se realiza debido a que el elemento a optimizar se sube al servidor de Simplygon para ser procesado y se descarga optimizado.

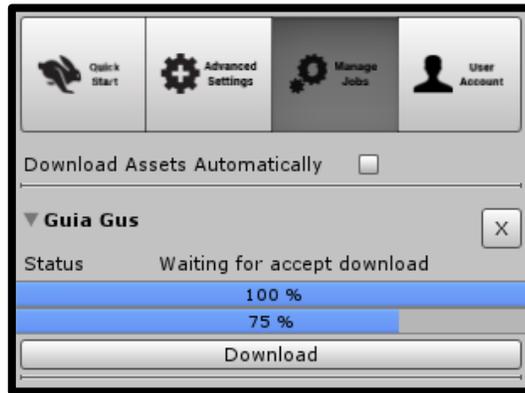


Figura 87. Descargando al personaje optimizado.

7. Ya que se ha completado el proceso de descarga, se crea la carpeta “LODs” en la ventana “Project” la cual contiene el personaje optimizado y una carpeta llamada “Materials”, que en caso de haber optimizado los materiales, ahí se almacenaran (figura 88).



Figura 88. Personaje optimizado.

Se puede verificar el número de polígonos después de la optimización, en este caso resultaron 24171 triangles (triángulos) o 12085 quads (cuadros).

La tabla 27 muestra el número de polígonos en quads que resultaron después de la optimización en cada personaje. Cabe resaltar que el número de polígonos se redujo considerablemente sin perder el aspecto o romper la geometría de los personajes.

Tabla 27. Porcentaje de optimización en los personajes.

Personaje	Porcentaje optimización	Número de polígonos originales	Numero de polígonos después de optimizar
Abuelita Domi	30 %	42386 quads	12716 quads
Charlie 1	19 %	41952 quads	7971 quads
Charlie 2	19 %	39263 quads	7460 quads
Lechero Ritchie	24 %	37195 quads	8927 quads
Guía Gus	30 %	40283 quads	12085 quads
Michel	22 %	56200 quads	12364 quads
Pizzero Alex	30 %	37110 quads	11133 quads

3.3.2 Animación

Al finalizar la fase de producción de recursos artísticos se procede a animar los objetos tridimensionales que lo requieren con el tipo de animación “frame by frame”. Para este proceso se utilizaron los programas, Daz Studio y Unity 3D.

3.3.2.1 Animación en Daz Studio

El uso de Daz Studio en el proyecto fue para realizar las animaciones de los personajes. A continuación se enumeran los pasos para la animación de uno de los personajes:

1. Identificación de huesos de los personajes y debido a que fueron creados en este programa se generan y clasifican de manera automática. En la figura 89 se observa la jerarquía que tienen los huesos.



Figura 89. Jerarquía del sistema óseo.

2. Crear la animación con una herramienta de Daz Studio llamada “aniMate” (figura 90), esta herramienta se encuentra ubicada en Window>Panes (Tabs)>aniMate.

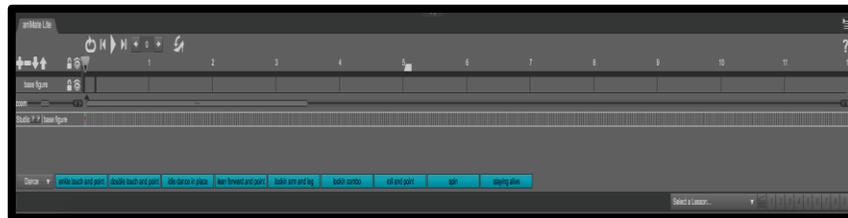


Figura 90. AniMate.

3. Seleccionar la figura base y en el área del “time line” de aniMate se agrega una marca de inicio dando clic derecho sobre esa área y en “Add Start”.

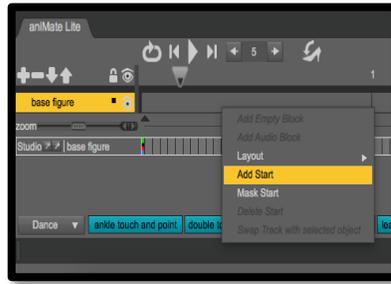


Figura 91. Agregando inicio de animación.

4. Mover el marcador “End of animation” (figura 92) hasta la cantidad de segundos que requiera la animación.

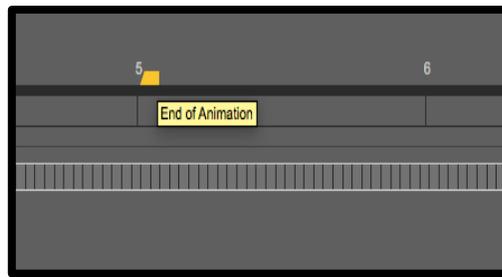


Figura 92. Marcador “End of Animation”.

5. Agregar los key frames que almacenan la información de posición y rotación de los huesos del personaje para este caso, de tal manera que estos funcionaran como puntos de referencia de inicio y final para cada movimiento (por ejemplo, el movimiento de cabeza del personaje se puede repetir en una animación varias veces y para cada movimiento se agrega un nuevo key frame), de este modo se forma un movimiento natural y fluido en el personaje.

Para agregar un key frame se mueve el marcador “Playhead” al segundo 1 (figura 93).

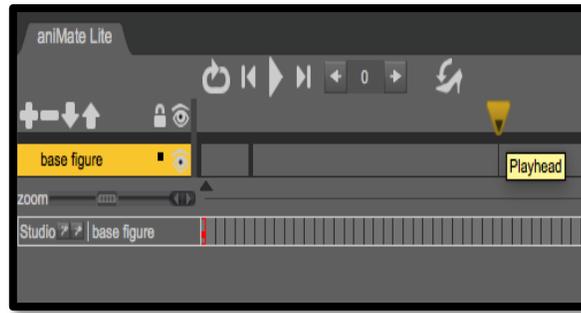


Figura 93. Marcador “Playhead”.

6. Realizar algún movimiento en un hueso del personaje se observa que en el frame se ha agregado un recuadro verde que indica el almacenamiento de los valores de posición y rotación del hueso en el key frame del segundo 1. Esto se realiza para cada hueso que se desee mover en ese tiempo.



Figura 94. Creación de un key frame.

7. Continuar sucesivamente con este procedimiento para completar la animación. Cabe mencionar que para este caso el último key frame debe ser igual al primero de cada hueso para que el movimiento sea cíclico y no sea notorio el inicio de la animación.
8. Al tener la animación completa la podemos exportar en formato FBX para utilizar el personaje con la animación en la aplicación didáctica. Para esto se debe dar clic derecho en aniMate y después en “Bake To Studio Keyframes”.

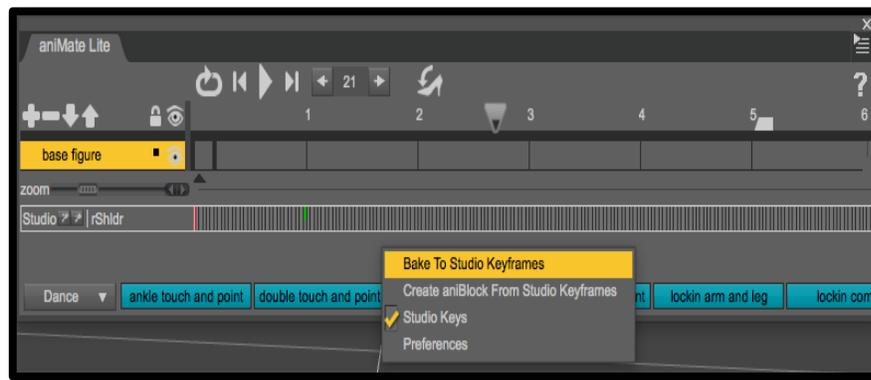


Figura 95. Animation Baking.

9. Exportar la animación, dando clic en File>Export. En esta sección se asigna el nombre del archivo FBX y la ruta donde será almacenado.

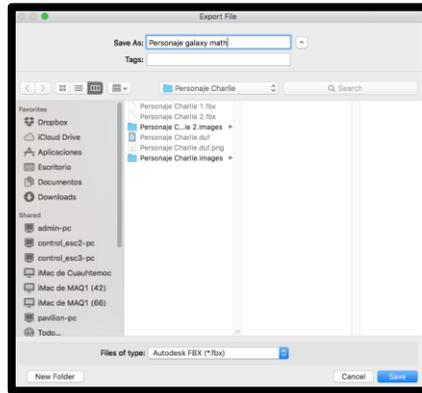


Figura 96. Exportación de personaje con animación.

10. Al definir lo anterior se configuran los parámetros de la exportación, en los cuales se indica que contenido de la escena se desea exportar, la versión del archivo FBX, si se desean empaquetar los materiales en una carpeta, si se desea unir la ropa con la figura que contiene el esqueleto.



Figura 97. Parámetros de la exportación.

11. Aceptar los parámetros de exportación y verificar que se haya creado el archivo FBX (figura 98).



Figura 98. FBX creado en Daz Studio.

3.3.2.2 Animación en Unity 3D (Animator)

Las animaciones realizadas en el motor de representación fueron para los recursos artísticos que no tuvieran movimientos complejos (por ejemplo, rotaciones, traslaciones, escalados y cambios de color). Para ejemplificar como se realizan estas animaciones, se muestra el caso del giro del bote de leche que está situado en el techo de la lechería. A continuación se describe el procedimiento para realizar una animación.

1. Una vez integrado el modelo de la lechería en el escenario (figura 99), seleccionar el bote de leche únicamente, ya que el recurso está compuesto por dos objetos (el bote y la construcción).



Figura 99. Modelo tridimensional “Lechería”.

2. Dirigirse a la ventana “Animation” en Unity, que nos servirá para crear y manipular las animaciones.



Figura 100. Ventana Animation.

3. Dar clic en botón “*REC Animation*” (Grabar animación) para crear una nueva animación, a la cual debemos darle un nombre y una ubicación de almacenamiento. En este caso, se le asignó el nombre “Giro bote leche” y se almaceno dentro de la librería donde se encuentra el modelo.

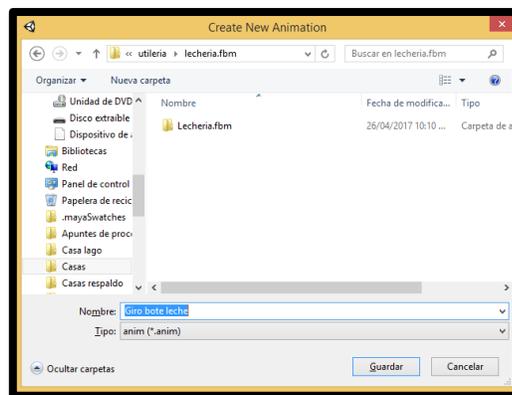


Figura 101. Creando una nueva animación.

4. En la figura 102 se observa que el botón “*REC Animation*” se ha coloreado de un color rojo tenue, esto indica que a partir de este momento podemos

comenzar con el proceso de animación, además, en el inspector podemos observar que se agregó un componente “Animator” al bote de leche (Figura 103), este sirve para asociar la animación creada con el modelo.

Cabe resaltar que existen dos tipos componentes para animación “Animation” y “Animator”, este último brinda más facilidad para el control de múltiples animaciones, controlando los estados y transiciones.

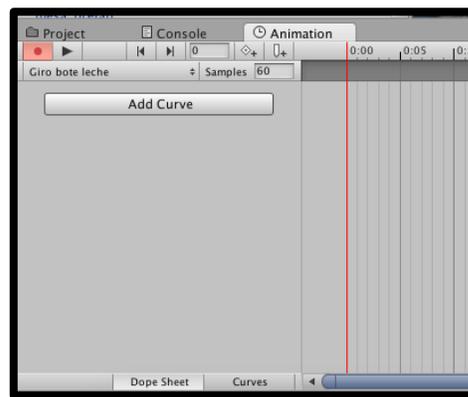


Figura 102. Estado de grabación en la ventana Animation.

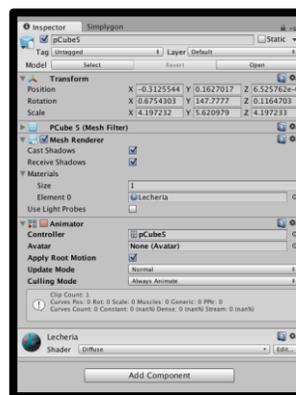


Figura 103. Componente Animator.

5. Agregar una curva seleccionando lo siguiente AddCurve>Transform>Rotation (Figura 104), se seleccionó rotation ya que lo que queremos es que la animación sea girar 360 grados el bote de leche en cierto tiempo.

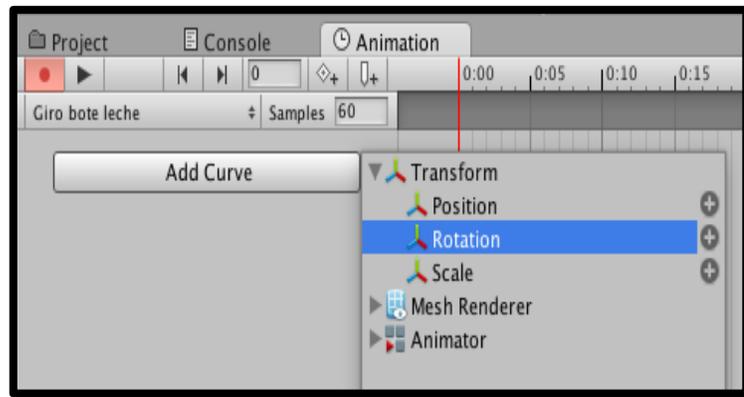


Figura 104. Agregando una curva en animación.

6. Una vez creada la curva se observa que se ha agregado una curva de rotación llamada “pCube5: Rotation”. Lo que prosigue es establecer el tiempo de duración de la animación. Para hacer esto se debe mover el marcador con forma de rombo hasta el número de segundos que durara la animación, en este caso se definieron tres segundos (Figura 105).

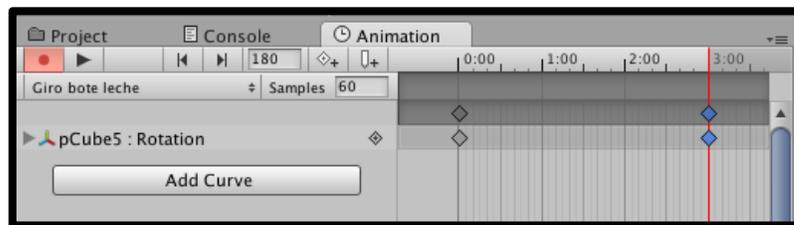


Figura 105. Estableciendo el tiempo de duración de una animación.

7. Asignar en el inspector de Unity los grados que va a girar en los tres segundos, para esto podemos identificar que el parámetro Rotation del componente Transform del bote de leche esta de color rojo (Figura 106) porque se encuentra en estado de grabación. Como se quiere que el bote de leche gire 360 grados, entonces se suman los 360 grados a los grados que tiene el eje Y.

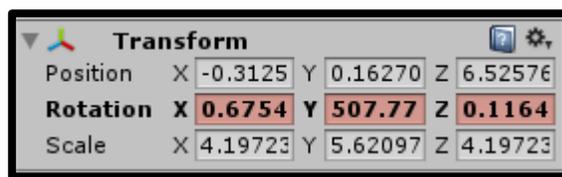


Figura 106. Componente Transform.

8. Comprobar que la animación realice el giro de 360 grados en los tres segundos. Para esto dar clic izquierdo sobre el icono de “Play” (Reproducir) en la ventana *Animation* y el motor de representación realizara una renderización de la animación.

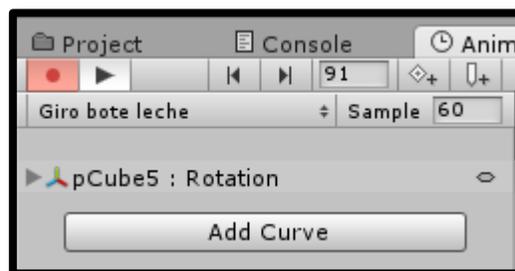


Figura 107. Botón Play de la ventana Animation.

9. Para finalizar la animación dar clic izquierdo sobre el botón *REC Animation* para desactivar el estado de grabación.

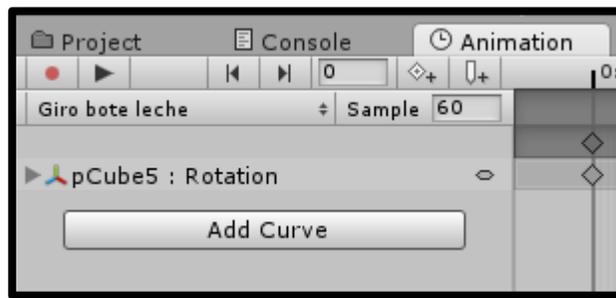


Figura 108. Desactivando el estado de grabación.

3.3.3 Integración

Como se menciona anteriormente, la integración de los recursos artísticos tridimensionales se realizó de forma gradual y de este modo fue tomando forma la aplicación.

Antes de empezar a crear los escenarios es recomendable especificar los métodos de almacenamiento y distribución que son carpetas llamadas bibliotecas de proyectos, en las que se contendrán los recursos artísticos desarrollados y de esta manera se tenga un lugar organizado, de fácil acceso y donde cada integrante desarrollador pueda encontrar exactamente el archivo que corresponda a cada escena en la que está trabajando, haciendo más eficiente el proceso de producción.

La figura 109 muestra la organización de las librerías del proyecto, en la cual se observa como quedaron clasificados los recursos artísticos que integran la aplicación didáctica. Por ejemplo, la librería “Naturaleza” almacena árboles, flores, rocas.

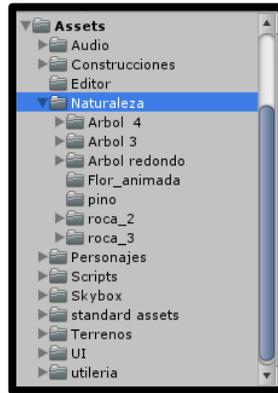


Figura 109. Organización de librerías de proyecto.

A continuación se describe el procedimiento que se realizó para formar los cuatro asteroides que conforman la aplicación didáctica.

- Una vez que se definió el método, se creó un nuevo proyecto en el motor de representación Unity, donde se importó el paquete que brinda Unity llamado “Character Controller”, el cual contiene una carpeta llamada “Standard Assets” que contiene objetos y códigos que permiten el movimiento en primera persona que se requiere para el proyecto.

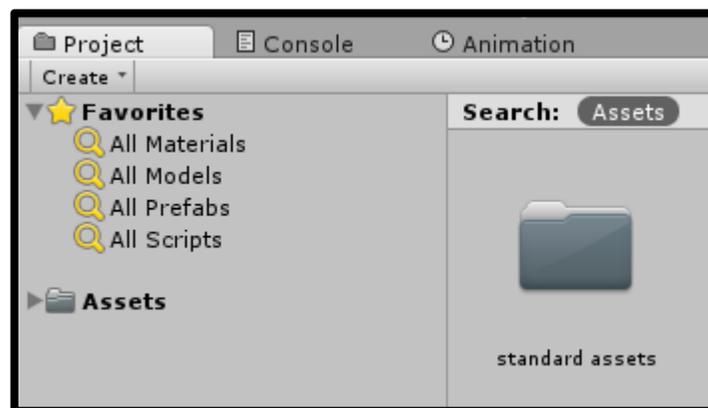


Figura 110. Paquete “Character Controller”.

- Dentro del proyecto se empiezan a crear las bibliotecas para integrar cada uno de los recursos artísticos en la ventana “Project”, haciendo clic derecho en “Assets” y se selecciona “Create” y posteriormente “Folder” (Figura 111).

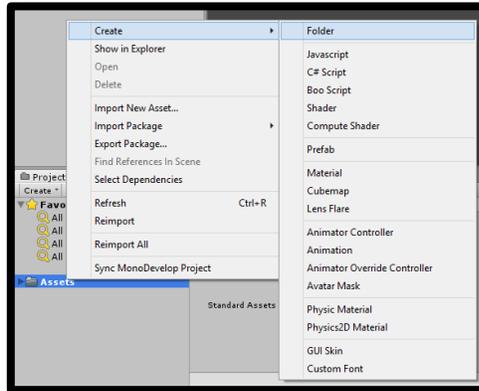


Figura 111. Creación de folder en Unity.

- Nombrar la librería creada, como ejemplo se tomó la llamada “Terrenos”, la cual contendrá los modelos del suelo de cada escenario (Figura 112).

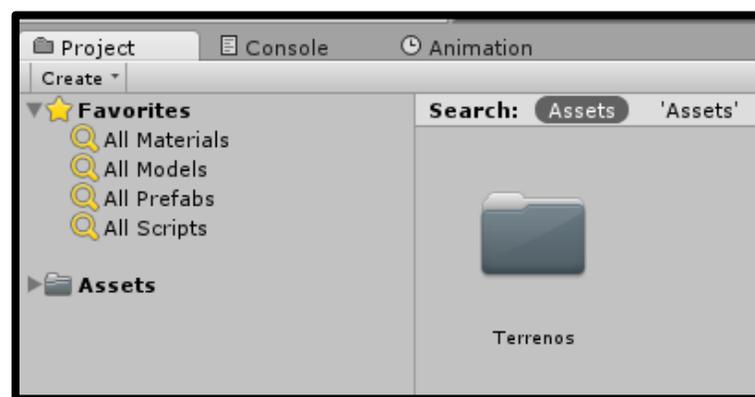


Figura 112. Biblioteca de proyecto.

- Importar los objetos tridimensionales de los terrenos de los escenarios que ya se encuentran empaquetados en formato FBX, arrastrándolos a la biblioteca creada.



Figura 113. Paquete FBX del terreno “Glorieta”.

Si se hizo de forma adecuada se observa que comienza el proceso de importación de geometría, texturas, materiales, normal maps y animaciones (Figura 114).

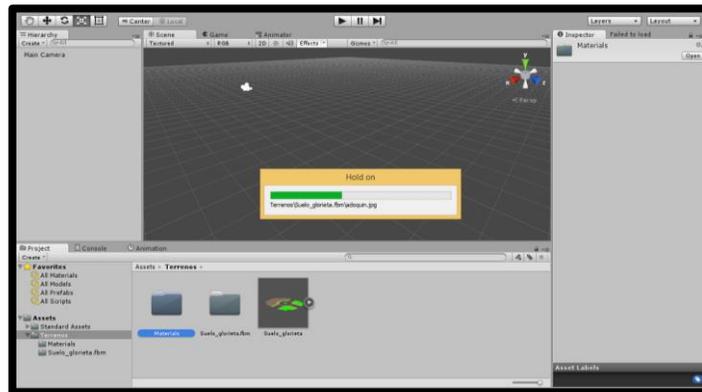


Figura 114. Proceso de importación.

La figura 115 muestra el contenido del paquete, el cual contiene una carpeta llamada “Materials”, “Suelo_glorieta” y el modelo tridimensional “Suelo_glorieta”.



Figura 115. Contenido del paquete “Suelo_glorieta”.

- Al finalizar la importación, ya podemos hacer uso del recurso artístico en la escena de Unity solamente arrastrando el modelo dentro de ella. Una vez el modelo tridimensional dentro de la escena se agrega dentro de la jerarquía que se encuentra del lado izquierdo (Figura 116).

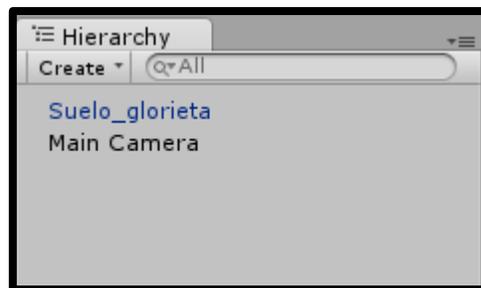


Figura 116. Jerarquía de la escena en Unity.

Se observa que hay un conflicto en Unity con el mapa de normales del recurso artístico (Figura 117).

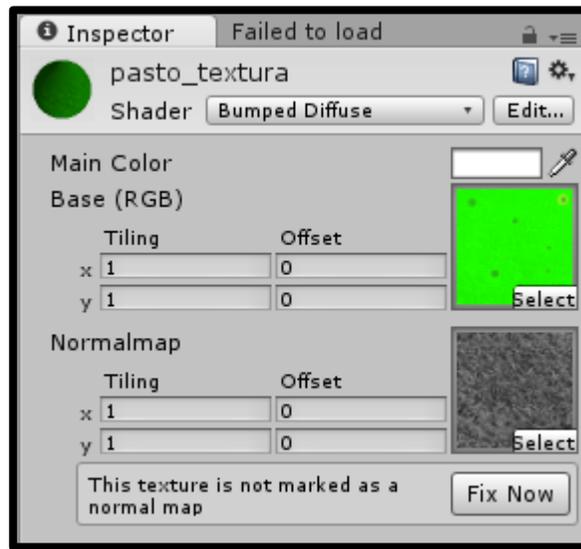


Figura 117. Conflicto de tipo de textura.

Este conflicto se da porque Unity trabaja con mapas de normales y no con imágenes en escala de grises. Para solucionar este conflicto se debe acceder a las propiedades de la imagen que se encuentra en la biblioteca, la figura 118, muestra cómo se configura y cambia el tipo de textura para que Unity realice una mejor representación.

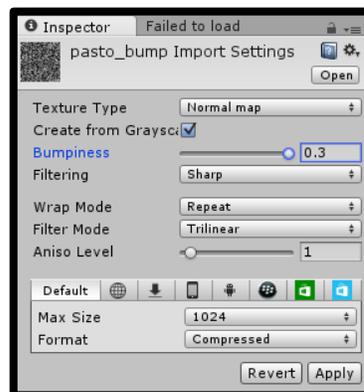


Figura 118. Configuración de textura Normal Map.

- Dar clic en el botón “Apply” para que Unity procese la imagen y convertirla de escala de grises a mapa de normales (Figura 119).

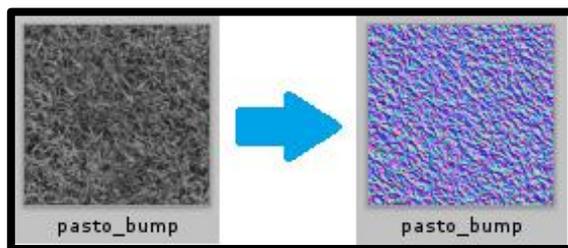


Figura 119. Conversión de escala de grises a Normal Map.

- Verificar en la escena de Unity el modelo tridimensional ahora tiene el aspecto de relieve en el pasto (Figura 120).

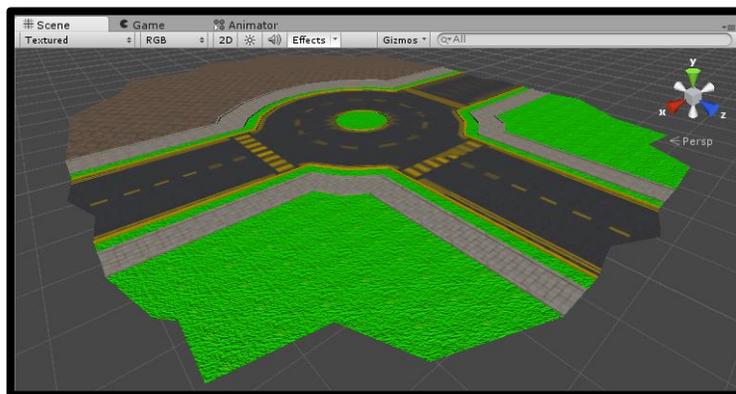


Figura 120. Efecto de Normal Map en pasto.

- De esta manera se siguieron importando los recursos artísticos al proyecto e integrándolos a las escenas con la ayuda de las herramientas de traslación, rotación y escalado para poder manipularlos y colocarlos en el lugar

establecido en la fase de diseño. En la figura 121 se muestran los cuatro escenarios.

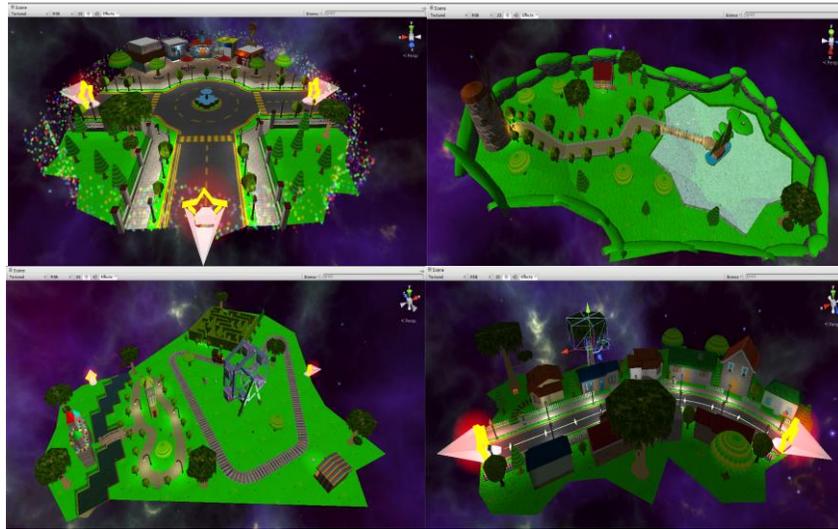


Figura 121. Integración de los cuatro escenarios.

3.3.4 Producción de recursos tecnológicos

En el desarrollo de esta fase se utilizó el lenguaje C# para generar los códigos (scripts) que realizan la tarea de interacción de la aplicación con el usuario y proporcionar la dinámica de algunos componentes 3D. Cabe mencionar que para agilizar el tiempo de producción se toma en cuenta el modelado de eventos y el modelado de navegación.

Para facilitar el presente proyecto y reducir el número de líneas en los códigos se desarrolló una clase llamada Fraccion, la cual contiene métodos que realizan operaciones aritméticas con números racionales. Los métodos contenidos en esta clase reciben dos objetos de la clase Fraccion como parámetros llamados numA y numB que devuelven un resultado dependiendo del método que se utilice. Estos objetos tienen como atributos tres variables de tipo entero las cuales son llamadas

numerador, denominador y entero debido a que estos son los elementos de un número racional en su forma de fracción.

Los métodos disponibles en esta clase se utilizan para:

- Sumar fracciones
- Restar fracciones
- Comparar si una fracción es mayor o menor que otra
- Comparar si dos fracciones son iguales
- Calcular el máximo común divisor
- Simplificar fracciones

Para crear el script en el proyecto se procede a crear una carpeta llamada Scripts en la raíz del proyecto (Figura 122).

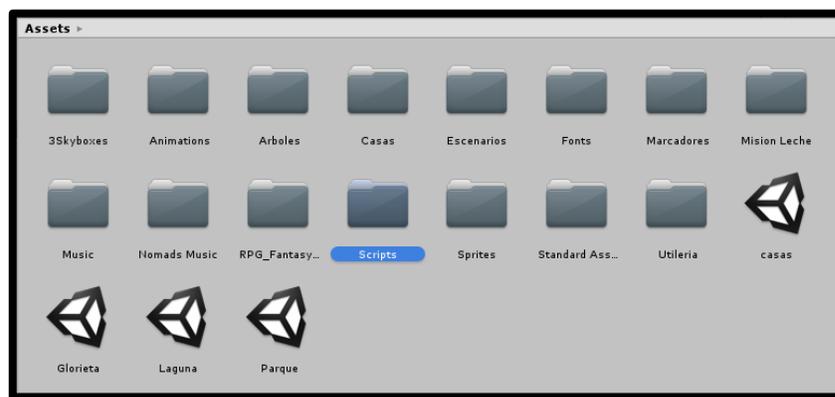


Figura 122. Carpeta de Scripts.

Al estar dentro de esta carpeta se crea el script dando clic derecho para seleccionar Create>C# Script (Figura 123).



Figura 123. Creación del script Fraccion.

Al realizar lo anterior, se procede a nombrar el script “Fraccion”, este nombre será el nombre de la clase de tal manera que no se pueden usar caracteres especiales o palabras reservadas de C#.



Figura 124. Script Fraccion.

Posteriormente se realiza la programación del script en el entorno de desarrollo que tiene Unity por default llamado “MonoDevelop”, para esto se da doble clic sobre el icono del script (Figura 125).

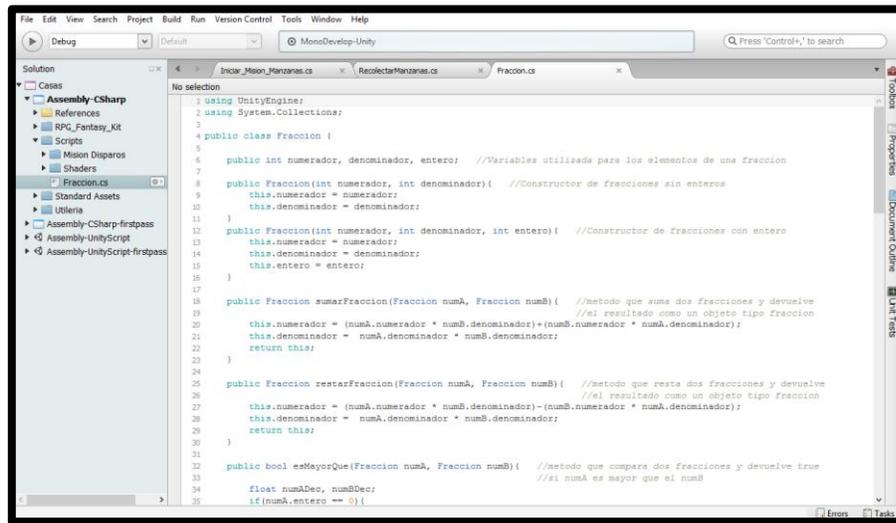


Figura 125. Entorno de desarrollo “MonoDevelop”.

A continuación se muestra el código empleado para la clase Fraccion (Figura 126), cabe mencionar que este script es utilizado como una librería de métodos y que no brinda algún comportamiento a los componentes tridimensionales, por lo tanto esta clase no deriva de la clase MonoBehaviour.

```

1 using UnityEngine;
2 using System.Collections;
3
4 public class Fraccion {
5
6     public int numerador, denominador, entero; //Variables utilizada para los elementos de una fraccion
7
8     public Fraccion(int numerador, int denominador){ //Constructor de fracciones sin enteros
9         this.numerador = numerador;
10        this.denominador = denominador;
11    }
12    public Fraccion(int numerador, int denominador, int entero){ //Constructor de fracciones con entero
13        this.numerador = numerador;
14        this.denominador = denominador;
15        this.entero = entero;
16    }
17
18    public Fraccion sumarFraccion(Fraccion numA, Fraccion numB){ //metodo que suma dos fracciones y devuelve
19        //el resultado como un objeto tipo fraccion
20        this.numerador = (numA.numerador * numB.denominador)+(numB.numerador * numA.denominador);
21        this.denominador = numA.denominador * numB.denominador;
22        return this;
23    }
24
25    public Fraccion restarFraccion(Fraccion numA, Fraccion numB){ //metodo que resta dos fracciones y devuelve
26        //el resultado como un objeto tipo fraccion
27        this.numerador = (numA.numerador * numB.denominador)-(numB.numerador * numA.denominador);
28        this.denominador = numA.denominador * numB.denominador;
29        return this;
30    }
31
32    public bool esMayorQue(Fraccion numA, Fraccion numB){ //metodo que compara dos fracciones y devuelve true
33        //si numA es mayor que el numB
34        float numADec, numBDec;
35        if(numA.entero == 0){
36            numADec = (float)numA.numerador / numA.denominador;
37        }else{
38            numADec = (float) (numA.entero*numA.denominador+numA.numerador) / numA.denominador;
39        }
40        if(numB.entero == 0){
41            numBDec = (float)numB.numerador / numB.denominador;
42        }else{
43            numBDec = (float) (numB.entero*numB.denominador+numB.numerador) / numB.denominador;
44        }
45        if (numADec > numBDec)
46            return true;
47        else
48            return false;
49    }
50
51    public bool esMenorQue(Fraccion numA, Fraccion numB){ //metodo que compara dos fracciones y devuelve true
52        //si numA es menor que el numB
53        float numADec, numBDec;
54        if(numA.entero == 0){
55            numADec = (float)numA.numerador / numA.denominador;
56        }else{
57            numADec = (float) (numA.entero*numA.denominador+numA.numerador) / numA.denominador;
58        }
59        if(numB.entero == 0){
60            numBDec = (float)numB.numerador / numB.denominador;
61        }else{
62            numBDec = (float) (numB.entero*numB.denominador+numB.numerador) / numB.denominador;
63        }
64        if (numADec < numBDec)
65            return true;
66        else
67            return false;
68    }
69

```

```

70 public bool esIgualQue(Fraccion numA, Fraccion numB){ //metodo que compara dos fracciones y devuelve true
71 //si numA es igual a numB
72     float numADec, numBDec;
73     if(numA.entero == 0){
74         numADec = (float)numA.numerador / numA.denominador;
75     }else{
76         numADec = (float)((numA.entero*numA.denominador)+numA.numerador) / numA.denominador;
77     }
78     if(numB.entero == 0){
79         numBDec = (float)numB.numerador / numB.denominador;
80     }else{
81         numBDec = (float)((numB.entero*numB.denominador)+numB.numerador) / numB.denominador;
82     }
83     if (numADec == numBDec)
84         return true;
85     else
86         return false;
87 }
88
89 public int mcdEuclides(Fraccion fr){ //metodo que realiza el algoritmo de Euclides y devuelve el MCD
90 //de una fraccion
91     int aux,residuo,num,den;
92     num = fr.numerador;
93     den = fr.denominador;
94     if( num < den ){
95         aux = num;
96         num = den;
97         den = aux;
98     }
99     residuo = num % den;
100    while( residuo != 0 ){
101        num = den;
102        den = residuo;
103        residuo = num % den;
104    }
105    return Mathf.Abs(den);
106 }
107
108 public Fraccion simplificarFraccion(Fraccion f){ //metodo recibe una fraccion y la devuelve simplificada
109     int mcd;
110     mcd = f.mcdEuclides (f);
111     f.numerador = f.numerador / mcd;
112     f.denominador = f.denominador / mcd;
113     return f;
114 }
115
116 public Fraccion simplificarFraccionConEntero(Fraccion fce){ //metodo para simplificar la fraccion
117 //obteniendo resultado una fraccion mixta
118     int mcd;
119     mcd = fce.mcdEuclides (fce);
120     fce.numerador = fce.numerador / mcd;
121     fce.denominador = fce.denominador / mcd;
122     fce.entero = Mathf.FloorToInt (fce.numerador / fce.denominador);
123     fce.numerador = fce.numerador - (fce.entero * fce.denominador);
124     return fce;
125 }
126 }

```

Figura 126. Script de la clase Fracción

Para describir esta actividad se eligió la misión llamada “Cosmic Apples” debido a que la programación de la aplicación didáctica es extensa. Se desarrollaron dos scripts que manipulan los elementos de la interfaz de usuario o Canvas (Figura 127) y la dinámica de la misión, por ejemplo, cuando el usuario da clic en una manzana y obtiene el peso de dicha manzana. Estos scripts fueron desarrollados para comunicarse entre sí y con el script Fraccion para pasarse valores entre ellos. Además se desarrolló otro script que administra las misiones (activar o desactivar misiones).



Figura 127. Elementos UI del Canvas.

Los scripts de la misión fueron nombrados de la siguiente manera:

- Iniciar_Mision_Manzanass.cs
- RecolectarManzanass.cs
- AllMissionStatus.cs

Los scripts Iniciar_Mision_Manzanass y RecolectarManzanass a diferencia del script Fracción derivan de la clase base MonoBehaviour la cual contiene los métodos Start (), Awake (), Update (), FixedUpdate (), OnGUI (), etc. que son utilizados por los componentes tridimensionales en Unity para tener algún comportamiento.

El script “Iniciar_Mision_Manzanass” es utilizado como el inicializador de la misión y se coloca como un componente del componente tridimensional “Inicializador de misiones” (figura 128).



Figura 128. Inicializador de misiones.

Este script también realiza las funciones de controlar las posibilidades en la misión, de las cuales se dan en tres casos:

- Cuando el usuario regresa al inicializador de misiones y le faltan manzanas para completar el peso, entonces el script le muestra un mensaje diciendo que le faltan manzanas.



Figura 129. Posibilidad de peso insuficiente.

- Cuando el usuario regresa al inicializador de misiones y tiene el peso correcto de manzanas, entonces el script le muestra un mensaje diciendo que lo hizo bien y posteriormente le da indicaciones para recibir la medalla.



Figura 130. Posibilidad de peso correcto.

- Cuando el usuario regresa al inicializador de misiones y se pasó del peso de manzanas requerido, entonces el script muestra un mensaje diciendo que se pasó y si desea reintentarlo.



Figura 131. Posibilidad de peso excedido.

A continuación se muestra el código desarrollado en el script `Iniciar_Mision_Manzanas`.

```

1 using UnityEngine;
2 using System.Collections;
3 using UnityEngine.UI;
4
5 public class Iniciar_Mision_Manzanas : MonoBehaviour {
6
7     public bool misionActivaA = false; //Variable para indicar el estado de la mision de las manzanas
8     public GameObject panelInstrucciones; //Panel para mostrar informacion en el canvas
9     public Text Instrucciones, mensajes; //Variable tipo Text para mostrar informacion en el canvas
10    public RecolectarManzanas rm; //Instancia para obtener acceso a las variables del Script RecolectaManzanas
11    Fraccion numRequerido = new Fraccion (3,4,1); //Objeto de la clase fraccion
12    private int cont; //Contador para llevar el conteo de las paginas de las instrucciones de la mision
13    public RawImage ImgManzanas; //Imagen para mostrar en el canvas
14    public Image MuyBien,Excedido; //Componentes para manipular las imagenes
15    private Vector3 posInicial; //Instancia para guardar la posicin inicial de los mensajes
16    private bool misionCumplida, pesoSuperado; //Variables para saber si se cumplio o no la mision
17    public Button btnCancelar; //Componente para manipular el boton cancelar
18
19    void Start(){
20        panelInstrucciones.gameObject.SetActive (false); //Ocultando iniialmente el panel de instrucciones
21        //Obteniendo acceso a la clase RecolectaManzanas
22        rm = GameObject.Find("Mision manzanas").GetComponent<RecolectarManzanas>();
23        ImgManzanas.gameObject.SetActive (false); //Ocultando la imagen que muestra cuales son las manzanas
24        MuyBien.gameObject.SetActive(false);
25        Excedido.gameObject.SetActive(false);
26        //Inicializando la instancia con la posicion inicial del gameobject llamado mensajes
27        posInicial = mensajes.transform.position;
28    }
29
30    void OnTriggerEnter(){
31        if (!misionActivaA) { //Si la mision aun no esta activada (false) se muestra el panel para iniciarla
32            Instrucciones.text = "La señora de la cabaña necesita 1 % de manzanas saltarinas." +
33                "\n\n ¿Quieres ayudarla a recolectarlas?"; //Agregando la cadena de texto
34            Instrucciones.gameObject.SetActive (true); //Mostrando el texto de las intrucciones
35            panelInstrucciones.gameObject.SetActive (true); //Mostrando el panel de instrucciones de la mision
36            cont = 2; //Asignacion del numero de paginas de las instrucciones
37            ImgManzanas.gameObject.SetActive (false); //Ocultando la imagen que muestra cuales son las manzanas
38            MuyBien.gameObject.SetActive(false);
39            Excedido.gameObject.SetActive(false);
40        }
41        //Si el peso de las manzanas es igual a 3/4 indica que es correcto
42        else if (numRequerido.esIgualQue(rm.resultado,numRequerido)) {
43            //Cambiando el estado de la mision porque se ha completado la mision y ya no necesita estar activa
44            misionActivaA = false;
45            MuyBien.gameObject.SetActive(true); //Mostrando la imagen con animacion que dice "Muy bien"
46            StartCoroutine ("esperar"); //Iniciando la corrutina
47            //Cambiando el estado de la variable a true para indicar que la mision se ha completado
48            misionCumplida = true;
49            //Desactivando el colisionador del gameobject que contiene este script
50            gameObject.GetComponent<BoxCollider>().enabled = false;
51            //Creando instancia de la clase AllMisionStatus
52            AllMissionStatus ams = GameObject.Find("Misiones").GetComponent<AllMissionStatus>();
53            //Cambiando el estado de la mision a true para recibir el premio y saber que la mision ya se realizo
54            ams.cosmicApples = true;
55        }
56        //Si el peso de las manzanas es menor al requerido indica que faltan
57        else if (numRequerido.esMenorQue(rm.resultado,numRequerido)){
58            //LLamada al metodo para escribir un mensaje en pantalla
59            StartCoroutine(escribirMensaje("¡TE FALTAN MANZANAS!"));
60        }
61        //Si el peso de las manzanas es mayor al requerido indica que ya se paso
62        else if (numRequerido.esMayorQue(rm.resultado,numRequerido)){
63            Excedido.gameObject.SetActive(true); //Mostrando la imagen con animacion que dice "te has pasado"
64            StartCoroutine ("esperar"); //Iniciando la corrutina
65            pesoSuperado = true; //Cambiando el peso de la variable que indica si se supero el peso indicado
66            //Desactivando el colisionador del gameobject que contiene este script
67            gameObject.GetComponent<BoxCollider>().enabled = false;
68        }

```

```

69     else{
70         //Si no se ha recogido ninguna manzana indica que recoja una manzana
71         StartCoroutine(escribirMensaje(";RECOGE UNA MANZANA!"));
72     }
73 }
74
75 public void botonSi(){ //Metodo para ocultar el panel de confirmacion y activar la mision
76     if (cont <= 0){ //Si el contador llega a cero se inicia la mision
77         if(misionCumplida){ //Si la mision fue realizada con exito realiza lo siguiente
78             panelInstrucciones.SetActive(false); //Ocultando el panel de las instrucciones
79             Destroy(gameObject); //Destruye el gameObject que contiene este script
80         }
81         else if(pesoSuperado){ //Si el peso fue superado realiza lo siguiente
82             rm.resultado.entero = 0; //Reiniciando los valores del resultado en el marcador
83             rm.resultado.numerador = 0;
84             rm.resultado.denominador = 0;
85             rm.enteros.text = rm.resultado.entero.ToString();
86             rm.numerador.text = rm.resultado.numerador.ToString();
87             rm.denominador.text = rm.resultado.denominador.ToString();
88             panelInstrucciones.SetActive (false); //Ocultando el panel de las instrucciones
89             //Habilitando el colisionador del gameObject que contiene este script
90             gameObject.GetComponent<BoxCollider>().enabled = true;
91         }
92         else{
93             panelInstrucciones.gameObject.SetActive (false); //Ocultando el panel de informacion
94             ImgManzanas.gameObject.SetActive (false);
95             //Al entrar al colisionador del objeto se cambia el estado de la mision a true (Activa)
96             misionActivaA = true;
97         }
98     }
99     else if(cont == 2){ //Si el contador es igual a dos se muestra el texto
100         Instrucciones.text="Instrucciones de juego \n\nDa un clic sobre las manzanas para recogerlas. "+
101             "\n\nCuando hayas juntado la cantidad requerida regresa con la señora de la cabaña " +
102             "para entregarlas. "+ "\n\nEn caso de haber superado la cantidad regresa a la cabaña para " +
103             "comenzar de nuevo la misión.";
104         cont--;
105     }
106     //Si el contador es igual a 1 se muestra la imagen y oculta el texto de la informacion
107     else if(cont ==1){
108         Instrucciones.gameObject.SetActive(false);
109         ImgManzanas.gameObject.SetActive(true);
110         cont--;
111     }
112 }
113
114 public void botonNo(){ //Metodo que utiliza el boton Cancelar para ocultar el panel de instrucciones
115     if (pesoSuperado) { //Si el peso fue superado realiza lo siguiente
116         misionActivaA = false; //Cambiano el valor de la variable a false para desactivar la mision
117         panelInstrucciones.gameObject.SetActive (false); //Ocultando el panel de instrucciones
118         //Habilitando el colisionador del gameObject que contiene este script
119         gameObject.GetComponent<BoxCollider>().enabled = true;
120         //Cambiano el valor de la variable a false para que se uede realizar de nuevo la mision al decidir x
121         pesoSuperado = false;
122     }
123     else {
124         panelInstrucciones.gameObject.SetActive (false); //Ocultando el panel de informacion
125     }
126 }
127
128 IEnumerator esperar(){
129     for(int i=0; i<5; i++){
130         yield return new WaitForSeconds(0.5f); //Esperando unos segundos
131     }
132     //Ocultando la imagen "Incorrecto" para que no aparezca los siguientes intentos
133     Excedido.gameObject.SetActive (false);
134     //Ocultando la imagen "Correcto" para que no aparezca los siguientes intentos
135     MuyBien.gameObject.SetActive (false);
136     if(misionCumplida){ //Si la mision fue cumplida con exito realiza lo siguiente

```

```

137     panelInstrucciones.SetActive(true); //Mostrando el panel de instrucciones
138     //Asignacion de una cadena de texto al gameObject de las instrucciones
139     Instrucciones.text = "Has cumplido la mision, dirigitete con el guia Ricardo para recibir tu medalla";
140     Instrucciones.gameObject.SetActive (true); //Mostrando el gameObject instrucciones
141     btnCancelar.gameObject.SetActive (false); //Ocultando el boton cancelar
142 }
143 else if(pesoSuperado){ //Si el peso fue superado realiza lo siguiente
144     panelInstrucciones.SetActive (true); //Mostrando el panel de instrucciones
145     Instrucciones.gameObject.SetActive (true); //Mostrando las instrucciones
146     Instrucciones.text = "¿Reintentar?"; //Asignacion de cadena de texto al gameObject instrucciones
147 }
148 }
149 IEnumerator escribirMensaje(string msj){ //Metodo para mostrar mensaje en el canvas
150     //Objeto de la clase Vector3 para obtener la posicion inicial de la interpolacion
151     Vector3 marcaInicial = posInicial;
152     //Objeto de la clase Vector3 para obtener la posicion final de la interpolacion
153     Vector3 marcaFinal = marcaInicial;
154     marcaFinal [1] = marcaFinal.y-50; //Restando 50 px para fijar una posicion final de la interpolacion
155     float fracJourney = 0f; //Cariable para indicar la posicion del objeto en la recta de interpolacion
156     mensajes.gameObject.SetActive (true); //Mostrando el mensaje que sera interpolado
157     mensajes.text = msj; //pasando la cadena de texto al gameObject tipo Text del canvas
158     while(fracJourney < 1){ //Mientras la variable fracJourney sea menor a 1 relaiza lo siguiente
159         yield return new WaitForSeconds(0.01f); //Espereando unos segundos para ejecutar la siguiente linea
160         //Calculando las posiciones de la interpolacion con el metodo Lerp
161         mensajes.transform.position = Vector3.Lerp (marcaInicial,marcaFinal,fracJourney);
162         //Incremento de la variable fracJourney para la siguiente posicion de la interpolacion
163         fracJourney = fracJourney + 0.01f;
164     }
165     for(int i=0; i<5; i++){
166         yield return new WaitForSeconds(0.5f); //Esperando unos segundos
167     }
168     mensajes.gameObject.SetActive (false); //Ocultando el texto de informacion
169 }
170 }

```

Figura 132. Script Iniciar_Mision_Manzanas.

El script RecolectarManzanas realiza la función de sumar los pesos de las manzanas seleccionadas por el usuario y envía la información al Canvas. A continuación se muestra el código.

```

1 using UnityEngine;
2 using System.Collections;
3 using UnityEngine.UI;
4 using System.Collections.Generic; //Librerias
5
6 public class RecolectarManzanas : MonoBehaviour {
7
8     public Text numerador, denominador, enteros, textInfo; //Componentes tipo texto para manipular el marcador
9     public GameObject panelConfirmacion, marcador; //Componentes del panel de confirmacion y manzanas
10    public Button btnSi; //Componente para manipular el boton SI
11    Fraccion numA = new Fraccion (0,0,0); //Instancias de la clase Fraccion para crear las fracciones
12    public Fraccion numB = new Fraccion (0,4,0);
13    public Fraccion resultado = new Fraccion (0,0,0);
14    public RaycastHit hit, hitaux; //Variable tipo Hit para detectar una colision con el rayo
15    public float velocidadY = 0.02f; //Variable tipo float para agregar la ferza que movera a las manzanas
16    public List<GameObject> manzanas = new List<GameObject>(); //Lista de manzanas activas
17    Iniciar_Mision_Manzanas imm; //Instancia de la clase Iniciar_Mision_Manzanas
18
19    void Start () {
20        //Agregando todos los gameobjects con tag igual a Manzanas que estan activas a la lista
21        manzanas.AddRange (GameObject.FindGameObjectsWithTag ("Manzanas"));
22        panelConfirmacion.gameObject.SetActive (false); //Ocultando inicialmente el panel de confirmacion
23        marcador.gameObject.SetActive (false); //Ocultando inicialmente el marcador de Kg de manzanas
24        //Obteniendo el acceso a la clase Iniciar_Mision_Manzanas con la instancia creada
25        imm = GameObject.Find("Inicializador de misiones").GetComponent<Iniciar_Mision_Manzanas>();
26    }
27
28    void Update () {
29        //Si la variable misionActivaA de la clase Iniciar_Mision_Manzanas es true se ejecuta la mision
30        if(imm.misionActivaA){
31            marcador.gameObject.SetActive(true); // Mostrando el marcador de Kg de manzanas
32            //Si el numero de elementos en la lista es mayor o igual a 1 ejecuta el metodo saltarinas()
33            if (manzanas.Count >= 1) {
34                saltarinas();
35            }
36            if(Input.GetMouseButtonDown(0)){ //Si se presiona el boton izquierdo del mouse
37                //Variable tipo Ray para capturar el clic del mouse en un pixel de la pantalla
38                Ray ray = Camera.main.ScreenPointToRay (Input.mousePosition);
39                //Si el rayo colisiono en una distancia menor o igual a 10 en el escenario
40                if(Physics.Raycast (ray, out hit, 10)){
41                    //Si la colision fue con los gameobjects con Tag "Manzanas"
42                    if(hit.collider.gameObject.tag == "Manzanas"){
43                        numB.numerador = Random.Range(1,3); //Generando un numero aleatorio entre 1 y 2
44                        //Informacion mostrada en el panel de confirmacion
45                        textInfo.text = ("Esta manzana pesa "+numB.numerador.ToString()+" / "
46                            +numB.denominador.ToString()+" Kg.\n\n¿Desea agregarla a la canasta?");
47                        panelConfirmacion.gameObject.SetActive(true); //Mostrando el panel de confirmacion
48                        //Hitaux se utiliza para guardar el hit de las manzanas ya que se actualiza con cada clic
49                        hitaux = hit;
50                    }
51                }
52            }
53        }
54        else{
55            marcador.gameObject.SetActive(false); //Ocultando el marcador
56            resultado.entero = 0; //Reiniciando los valores del resultado en el marcador
57            resultado.numerador = 0;
58            resultado.denominador = 0;
59            numerador.text = resultado.numerador.ToString(); //Pasando la fraccion resultado a los gameobjects
60            denominador.text = resultado.denominador.ToString();
61            enteros.text = resultado.entero.ToString();
62        }
63    }
64
65    public void botonNo(){ //Metodo para ocultar el panel de confirmacion
66        panelConfirmacion.gameObject.SetActive (false);
67    }
68

```

```

69 public void sumar(){ //Metodo para realizar la suma de la manzana seleccionada
70     agregarACanasta (numB,hitaux);
71 }
72
73 void agregarACanasta(Fraccion numB, RaycastHit hit){
74     if((resultado.entero | resultado.numerador) > 0){
75         //Si el numerador y los enteros de la fraccion resultado son mayores a 0
76         if(resultado.entero == 0){ //Si el entero del resultado es igual a 0
77             //El numerador de la fraccion numA sera igual al numerador de la fraccion resultado
78             numA.numerador = resultado.numerador;
79         }
80         else { //Si el entero del resultado es diferente de 0
81             //Si el entero de la fraccion resultado es diferente de 0 se convierte la fraccion del tipo n/Zn
82             numA.numerador = (resultado.entero * resultado.denominador) + resultado.numerador;
83         }
84         //El denominador del numA sera igual al denominador del resultado
85         numA.denominador = resultado.denominador;
86         //la fraccion resultado manda a llamar el metodo sumarFraccion para que devuelva la suma de las
87         //fracciones numA + numB
88         resultado = resultado.sumarFraccion(numA,numB);
89         //la fraccion resultado manda a llamar el metodo para simplificar el resultado a una fraccion mixta
90         resultado = resultado.simplificarFraccionConEntero(resultado);
91     }
92     //Si el entero o numerador de resultado son menores a 1 se iguala resultado a numB
93     else{
94         resultado.numerador = numB.numerador;
95         resultado.denominador = numB.denominador;
96     }
97     //Pasando la fraccion resultado a los gameobjects tipo Text que estan en el marcador de puntos
98     numerador.text = resultado.numerador.ToString();
99     denominador.text = resultado.denominador.ToString();
100    enteros.text = resultado.entero.ToString();
101    //Si el clic se realizo sobre un objeto con tag "Manzanas" realiza lo siguiente
102    if (hit.collider.gameObject.CompareTag("Manzanas")) {
103        manzanas.Remove(hit.collider.gameObject); //Remueve de la lista la manzana que recibio el hit
104        Destroy(hit.collider.gameObject); //Destruye el gameObject que recibio el hit
105    }
106    panelConfirmacion.gameObject.SetActive (false);
107 }
108
109 //Metodo para agregar una fuerza a las manzana y causar su movimiento
110 void saltarinas(){
111     //Recorrido de la lista para agregar una fuerza a las manzanas que estan en la lista
112     for(int i=0; i<manzanas.Count; i++){
113         Vector3 temp = manzanas[i].transform.position;
114         temp.y += velocidadY;
115         manzanas[i].transform.position = temp;
116     }
117 }
118 }

```

Figura 133. Script Recolectar Manzanas.

El script AllMissionStatus administra las misiones, por ejemplo, si una misión ya fue realizada, la variable con el nombre de la misión cambia su valor a true para que la próxima vez que se cargue el escenario que contiene la misión ya no se carguen los componentes tridimensionales de dicha misión.

```

1 using UnityEngine;
2 using System.Collections;
3
4 public class AllMissionStatus : MonoBehaviour {
5
6     public bool cosmicApples; //Variables para saber si las misiones ya fueron cumplida
7     public bool galacticLabyrinth;
8     public bool spatialCoins;
9     public bool stellarMilk;
10    public bool rainbowFishes;
11    public bool moonPizza;
12
13    // Use this for initialization
14    void Start () {
15        //Metodo para evitar que se destruya el gameobject que contiene este script al cargar otro escenario
16        DontDestroyOnLoad (gameObject);
17    }
18
19    void Update () {
20    }
21
22
23    void OnLevelWasLoaded(int level){ //Metodo para detectar que escenario fue cargado
24        if (level == 2) { //Si el escenario cargado tiene como id el numero 2 (Laguna) realiza lo siguiente
25            print ("laguna fue cargada");
26            //Si la mision Cosmis Apples ya fue realizada evita que se cargue el inicializador de la mision
27            if(cosmicApples){
28                //encontrando y destruyendo el inicializador de la mision en la escena
29                Destroy(GameObject.Find("Inicializador de misiones"));
30            }
31        }
32    }
33 }

```

Figura 134. Script AllMissionStatus.

3.3.5. Pruebas

Las pruebas se realizaron durante (fase de integración y producción de recursos tecnológicos) y después del desarrollo de la aplicación. Estas pruebas fueron realizadas por los desarrolladores de la aplicación y por profesores que colaboran en este proyecto y que son integrantes del Cuerpo Académico TIC's y dispositivos electrónicos, así como encargados del laboratorio de RV del CU UAEM Ecatepec.

A continuación se describen los inconvenientes que se detectaron y cómo fueron solucionados (tablas 28 y 29).

Tabla 28. Problemas detectados en la fase de integración.

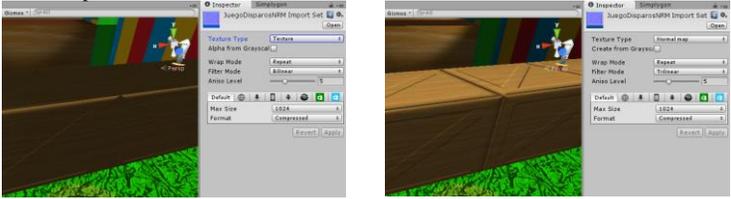
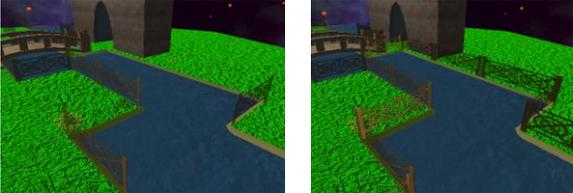
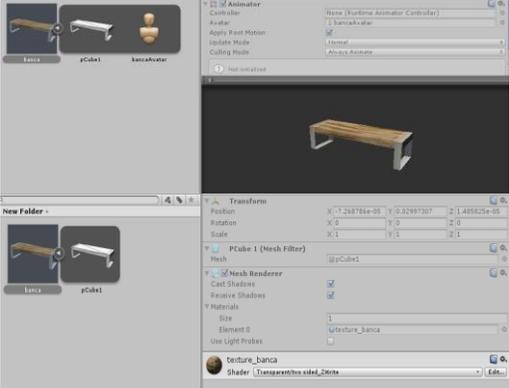
Problema	Descripción y corrección
<p>Los mapas de normales no eran detectados por Unity.</p>	<p>El problema se daba al momento de importar los archivos FBX, ya que Unity no detectaba los mapas de normales como tal. Para solucionarlo se accedió a las propiedades de la imagen de mapa de normales para cambiar de tipo Texture a tipo Normal Map.</p> 
<p>Objetos con normales transparentes.</p>	<p>El problema ocurría porque los modelos se creaban con una sola cara para representar cosas que eran idénticas de ambos lados, por ejemplo, el barandal como se observa en la imagen de la izquierda. En Unity solo renderizaba por un lado y el otro era transparente. Para solucionar esto, se creó un shader para mostrar de ambos lados las normales.</p> 
<p>Importaciones con componente Animator vacío.</p>	<p>Al importar un archivo FBX, el modelo tridimensional tenía un componente Animator vacío y un avatar, esto era un inconveniente debido a que no todos los recursos importados tenían animaciones, siendo componentes que ocupaban espacio e inservibles. Para solucionar en el inspector se desactiva la opción de import animation en la sección "Animations" y se cambia el tipo de animación a "None" en la sección de Rig.</p> 

Tabla 29. Problemas detectados en la fase de producción de recursos tecnológicos.

Problema	Descripción y corrección
Errores de sintaxis.	Estos problemas se detectaban al compilar los scripts, ya que la consola de Unity, muestra cuando existen este tipo de errores. La solución era visitar Unity Scripting o el API de C# para ver la sintaxis de las instrucciones y corregirlos.
Error al no asignar los componentes en los scripts utilizados.	Estos errores suceden cuando se ejecuta la aplicación y los componentes aún no han sido asignados en los scripts utilizados (figura izquierda), entonces Unity intenta acceder a ellos y al no detectar el componente requerido, muestra el error en la consola. Para solucionar este problema, se asignan todos los componentes en los scripts antes de ejecutar la aplicación (figura derecha). <div style="display: flex; justify-content: space-around; margin-top: 10px;">   </div>
Errores de lógica en la programación.	Este tipo de error impide que la aplicación realice lo que estaba previsto. El código puede compilarse y ejecutarse de manera correcta, pero el resultado de una operación puede generar un resultado no esperado. La solución a estos problemas es comprender el código e ir verificando valores durante su ejecución hasta encontrar la falla.

Una vez terminada la versión alfa se hizo un “*check list*” (Lista de verificación) para verificar que se cumpliera con los requerimientos funcionales propuestos en la fase de análisis. En la tabla 30 se puede observar que se cumplieron con todos los requerimientos, sin embargo hubo observaciones de redacción, de faltas ortográficas, visuales y de funcionamiento (tabla 31) por parte del equipo de trabajo, las cuales se atendieron de forma inmediata para que la aplicación en la siguiente fase pudiera ser evaluada por un grupo de usuarios.

Tabla 30. Evaluación de los requerimientos funcionales.

Requerimientos funcionales	Observaciones
Instrucciones en las misiones claras.	11. Algunas de las misiones no tenían redacción clara. 12. Existían errores ortográficos en las instrucciones.
Entrega de recompensas al usuario después de finalizar la misión.	✓
El usuario puede tomar el rumbo que quiera dentro de la aplicación.	✓
Audio claro y con fidelidad.	✓
El contexto tiene combinación de realista y caricatura.	✓
Interacción con los personajes con diálogos textuales.	✓
El usuario puede reintentar las misiones al no superarlas.	✓
El personaje principal reacciona a las instrucciones del usuario.	✓
Escenarios familiares para los usuarios.	✓
Aborda temas de fracciones.	✓
Vista en primera persona.	✓
Personajes con aspecto de fantasía.	✓
Las actividades son de la vida cotidiana o que puedan realizar los usuarios.	✓
Interacción por medio del mouse y teclado.	✓

Tabla 31. Observaciones funcionales y visuales de la aplicación.

Observaciones	Corrección
El movimiento de la cámara era muy rápido y causaba mareo.	Se accedió a los parámetros de los Scripts en la cámara y se bajó la sensibilidad.
Algunos marcadores de puntos eran demasiado grandes.	Se escalaron los marcadores a un tamaño más pequeño.
En un letrero decía “Casas” en lugar de “Vecindario”.	Se modificó la textura en GIMP, se cambió el texto y se volvió a asignar al material de los letreros en Unity. Cabe mencionar que el texto fue cambiado debido a que el nombre real del asteroide y que se menciona en la información de la aplicación es “Vecindario”.
La cámara no se bloqueaba al mostrar información y esto complicaba la lectura.	Se agregaron instrucciones en los Scripts para deshabilitar el movimiento de la cámara al aparecer alguna información en la aplicación.
Los usuarios sentían vértigo al acercarse a las orillas de los asteroides.	Se implementó el uso de partículas para crear una barrera y evitar el sentimiento de vértigo en los usuarios.

3.3.6 Empaquetado

El empaquetado se refiere a la forma en que va ser exportado un proyecto para ejecutarse en un sistema operativo específico. A continuación se muestra la configuración de los parámetros de empaquetado para la aplicación.

Para empezar con el empaquetado de la aplicación, se configura el nivel de calidad gráfica que Unity renderizará, a continuación se describe el procedimiento que se realizó en la para este caso:

- Acceder a la configuración de la calidad gráfica seleccionando Edit>Project>Settings>Quality. La figura 135 muestra cómo fueron configurados los parámetros.



Figura 135. Configuración de calidad gráfica sección 1.

Se observa que la configuración es una matriz en la cual las columnas hacen referencia a la plataforma y las filas a la calidad gráfica. Para este caso se estableció “Good” (buena) como valor de default para la segunda columna (Standalone), esto para que la aplicación tenga un buen rendimiento según las características del ordenador donde será ejecutada. Cabe mencionar que la columna Standalone incluye los sistemas operativos Linux, Mac OS y Windows.

- Configurar la segunda sección del inspector de calidad. Aquí se cambió el parámetro “Anti Aliasing” a “4x Multi Sampling” ya que ayuda a mejorar la representación visual de los bordes de los polígonos, sin embargo, esto incurre como un costo en el rendimiento del equipo de cómputo.



Figura 136. Configuración de calidad gráfica sección 2.

Después de haber configurado el nivel de calidad gráfica se procede a configurar la creación del proyecto. A continuación se describe el procedimiento que se realizó para la construcción de la aplicación.

1. Seleccionar File>Build Settings para abrir la ventana de configuración de construcción (figura 137).



Figura 137. Ventana Build Settings.

2. Arrastrar las escenas que integran el proyecto al área “Scenes In Build” (figura 138).



Figura 138. Escenas en construcción del proyecto.

En la figura 138 se observa que las escenas están indexadas, ya que el motor de representación asigna un identificador para hacer referencia a las escenas en los scripts de programación si es que se requiere. Este identificador también sirve para establecer el orden de carga de las escenas incluidas en la construcción.

3. Seleccionar el sistema operativo en el cual será ejecutada la aplicación, en este caso se seleccionó Windows de 64 bits (figura 138).

Cabe resaltar que las opciones “Development Build”, “AutoConnect Profiler” y “Script Debugging” son herramientas útiles para los desarrolladores, las cuales permiten verificar por ejemplo, el consumo de recursos de la aplicación (solo versión pro de Unity), compilar los Scripts.

4. Asignar un icono al archivo ejecutable. Para esto se debe importar la imagen que se desea a la librería adecuada dentro de la ventana “Project” y posteriormente dar clic en el botón “Player Settings” para abrir la “configuración del reproductor” (figura 139) en el inspector. Lo siguiente es arrastrar la imagen al campo “Default Icon”.

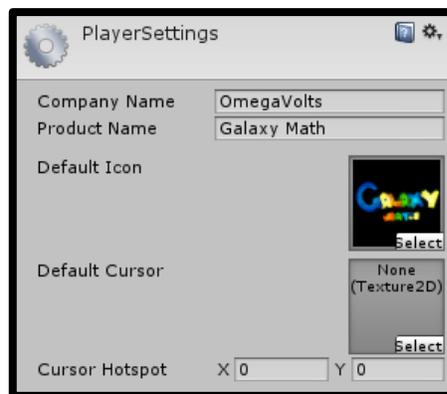


Figura 139. Asignación del icono de la aplicación.

5. En la sección “Resolution and Presentation” (Resolución y presentación) (figura 140) se dejaron habilitados los check box “Default Is Full Screen” (Pantalla completa por defecto) y “Default Is Native Resolution” (Resolución nativa por defecto), debido a que la aplicación podría ser ejecutada en diferentes tamaños y resoluciones de pantalla.

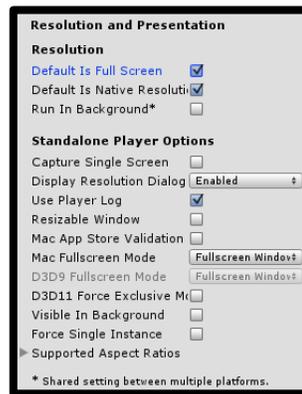


Figura 140. Configuración de resolución y presentación.

6. El siguiente paso es regresar a la venta “Build Settings” y dar clic en el botón “Build” (Construir) y posteriormente se abrirá una ventana en la cual se asigna el nombre del proyecto y la ubicación de almacenamiento (figura 141).

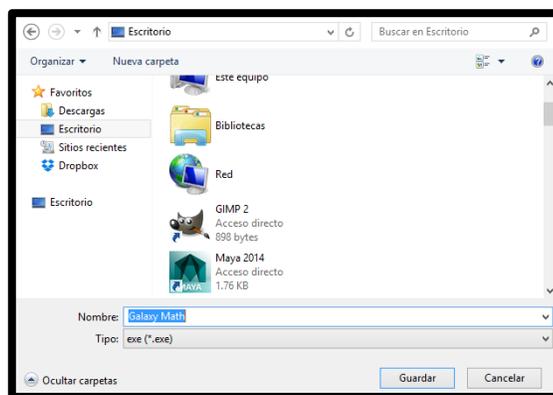


Figura 141. Nombre y ruta de almacenamiento de la construcción de la aplicación.

7. Al dar clic en el botón “Guardar”, inicia el proceso de construcción y finalmente se obtiene el archivo ejecutable (figura 142).

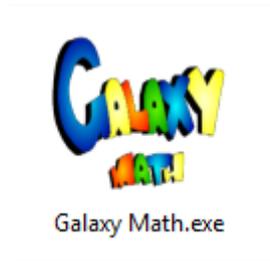


Figura 142. Ejecutable de la aplicación.

3.4 Fase de Evaluación

3.4.1 Valoración por el cliente / usuario

Después de realizar los cambios necesarios para que la aplicación estuviera en una versión Beta, se decidió buscar un instrumento que a futuro permitiera conocer y analizar el nivel de motivación, aceptación y satisfacción que despierta esta tecnología en experiencias formativas apoyadas con aplicaciones de RV tanto para los profesores como para los estudiantes de tercer año de primaria. Además de que facilitara conocer las dificultades técnicas de uso que pudiera tener la aplicación didáctica (tabla 32).

El cuestionario elegido fue el Modelo de aceptación de la RA (Realidad Aumentada) basado en el modelo TAM (Modelo de Aceptación de la Tecnología), diseñado por Cabero, Barroso y Llorente (2016), el cual está compuesto por 23 preguntas presentadas en escala tipo likert de siete opciones de respuesta 1) extremadamente probable; 2) muy probable; 3) ligeramente probable; 4) ninguno; 5) poco probable; 6) bastante improbable y 7) extremadamente improbable. Además contiene 4 reactivos sociodemográficos.

El cuestionario propuesto está constituido por siete escalas: 1) Utilidad percibida, 2) Facilidad de uso percibida, 3) Percepción de disfrute, 4) Actitud hacia el uso, 5) La intención de utilizar, 6) Calidad técnica del sistema y 7) Competencia tecnológica.

Tabla 32. Instrumento para medir el nivel de aceptación de la RA.

Genero	F	M	Edad							
Área curricular			Nivel de estudios							
Instrucciones	El cuestionario es anónimo y voluntario; sus respuestas se mantendrán confidenciales, por lo que se solicita su opinión sincera al respecto. No hay respuestas buenas o malas; todas son valiosas. Responda después de leer cuidadosamente cada enunciado. Debe responder en la casilla de la letra correspondiente (A, B, C, D, E, F o G).									
	A) extremadamente probable; B) muy probable; C) ligeramente probable; D) ninguno; E) poco probable; F) bastante improbable y G) extremadamente improbable									
Número	Pregunta			A	B	C	D	E	F	G
1	El uso de este sistema de RA podría mejorar mi aprendizaje en el aula.									
2	La utilización del sistema de RA durante las clases me facilitaría la comprensión de ciertos conceptos.									
3	Creo que el sistema de RA es útil cuando se está aprendiendo									
4	Creo que el sistema de RA es fácil de usar.									
5	Aprender a usar y manejar el sistema de RA no es un problema para mí.									
6	Mi interacción con el sistema de RA es claro y comprensible.									
7	Creo que el sistema de RA permite aprender jugando.									
8	Disfruté con el uso del sistema de RA.									
9	Aprender con un sistema de RA de este tipo es entretenido.									
10	El uso de un sistema de RA hace que el aprendizaje sea más interesante.									
11	Aprendiendo a través del sistema de RA me ha parecido aburrido.									
12	Creo que el uso de un sistema de RA en el aula es una buena idea.									
13	Me gustaría utilizar el sistema de RA en el futuro si tuviera la oportunidad.									
14	El uso de un sistema de RA me permitiría aprender por mi cuenta.									
15	Me gustaría utilizar el sistema de RA para aprender tanto los temas que se me han presentado como otros.									
16	Los objetos producidos en RA son estéticamente satisfactorios.									
17	Los objetos producidos en RA funcionan correctamente.									
18	Los objetos producidos en RA están bien estructurados y organizados.									
19	Los objetos producidos en RA son atractivos.									
20	Los objetos producidos en RA son seguros.									
21	De manera general me considero capacitado para el manejo técnico de los medios audiovisuales e informática.									
22	De manera general me considero capacitado para el manejo técnico de Internet.									
23	De manera general me considero capacitado para el manejo técnico de los objetos producidos en el sistema de RA.									

Una vez analizado el instrumento, se decidió modificar la redacción de las preguntas con el objetivo de que fueran más claras. A su vez se eligieron las preguntas que se

considera que los niños que se encuentran cursando tercer año de primaria puedan responder. Y las respuestas se redujeron a las siguientes opciones: si, más o menos y no (tabla 33)

Tabla 33. Instrumento para medir el nivel de aceptación de la RV en los alumnos.

Tipo de plantilla	Instrumento para medir el nivel de aceptación de la aplicación en los alumnos.					
Genero	F	M	Edad			
Área curricular			Nivel de estudios			
Instrucciones	El cuestionario es anónimo y voluntario; sus respuestas se mantendrán confidenciales, por lo que se solicita su opinión sincera al respecto. No hay respuestas buenas o malas; todas son valiosas. Responda después de leer cuidadosamente cada enunciado. Debe responder en la casilla de la opción correspondiente (Si, Más o menos o No)					
Número	Pregunta			Si	Más o menos	No
1	Creo que la aplicación de RV es fácil de usar.					
2	Aprender a usar y manejar la aplicación de RV no es un problema para mí.					
3	La interacción con la aplicación de RV es clara y comprensible.					
4	Disfruté la clase con el uso de la aplicación de RV.					
5	Ver el tema de la clase con una aplicación de RV de este tipo es entretenido.					
6	El uso de la aplicación de RV hace que la clase sea más interesante.					
7	Ver el tema de la clase con el uso de la aplicación de RV me pareció aburrido.					
8	Creo que el uso de una aplicación de RV en la clase es una buena idea.					
9	Me gustaría utilizar la aplicación de RV en el futuro si tuviera la oportunidad.					
10	Me gustaría utilizar la aplicación de RV para repasar el tema que se me ha presentado como otros en casa.					
11	Los objetos producidos en la aplicación de RV funcionan correctamente.					
12	Los objetos producidos en la aplicación de RV son atractivos.					
13	De manera general, me considero capacitado para el manejo técnico de los objetos producidos en la aplicación de RV.					

Para el caso de la aplicación del instrumento para los profesores participantes se propone dejar todas las preguntas y opciones de respuesta, ya que se considera que ellos si cuentan con la preparación suficiente para comprender el cuestionario completo (tabla 34).

Tabla 34. Instrumento para medir el nivel de aceptación de la RV en los alumnos desde el punto de vista del profesor.

Tipo de plantilla		Instrumento para medir el nivel de aceptación de la aplicación en los alumnos desde el punto de vista del profesor.						
Genero		F	M	Edad				
Área curricular					Nivel de estudios			
Instrucciones		<p>El cuestionario es anónimo y voluntario; sus respuestas se mantendrán confidenciales, por lo que se solicita su opinión sincera al respecto. No hay respuestas buenas o malas; todas son valiosas. Responda después de leer cuidadosamente cada enunciado. Debe responder en la casilla de la letra correspondiente (A, B, C, D, E, F o G).</p> <p>A) extremadamente probable; B) muy probable; C) ligeramente probable; D) ninguno; E) poco probable; F) bastante improbable y G) extremadamente improbable</p>						
Número	Pregunta	A	B	C	D	E	F	G
1	El uso de la aplicación de RV podría mejorar el aprendizaje de los niños en el aula.							
2	La utilización de la aplicación de RV durante la clase facilitaría a los niños la comprensión de ciertos conceptos.							
3	Creo que el uso de las aplicaciones de RV es útil cuando se está enseñando y los niños están aprendiendo.							
4	Creo que la aplicación de RV es fácil de usar.							
5	Aprender a usar y manejar la aplicación de RV no es un problema para mí.							
6	La interacción con la aplicación de RV es clara y comprensible.							
7	Creo que la aplicación de RV permite que los niños refuercen el tema visto en clase jugando.							
8	Disfruté la clase con el uso de la aplicación de RV.							
9	Impartir la clase con una aplicación de RV de este tipo es entretenido.							
10	El uso de la aplicación de RV hace que la clase sea más interesante.							
11	Enseñar el tema en clase con apoyo de la aplicación de RV me pareció aburrido.							
12	Creo que el uso de un sistema de RV en el aula es una buena idea							
13	Me gustaría utilizar la aplicación de RV en el futuro si tuviera la oportunidad.							
14	El uso de aplicaciones de RV les permitiría a mis alumnos repasar y reforzar los temas vistos en clase por su cuenta.							
15	Me gustaría utilizar la aplicación de RV para que mis alumnos repasar y reforzar tanto los temas que se me ha presentado como otros.							
16	Los objetos producidos en la aplicación de RV son estéticamente adecuados para el tema en cuestión.							
17	Los objetos producidos en la aplicación de RV funcionan correctamente							
18	Los objetos producidos en la aplicación de RV están bien estructurados y organizados							
19	Los objetos producidos en la aplicación de RV son atractivos.							
20	Los objetos producidos en la aplicación de RV no afectan en ningún sentido al usuario.							
21	De manera general, me considero capacitado para el manejo técnico de los objetos producidos en la aplicación de RV							
22	De manera general, me considero capacitado para el manejo técnico de Internet.							
23	De manera general, me considero capacitado para el manejo técnico de los objetos producidos en el sistema de RV.							

Conclusiones

Al finalizar este proyecto de tesis se llegaron a las siguientes conclusiones:

- Se cumplió con el objetivo señalado en este trabajo de tesis que fue desarrollar una aplicación de RV funcional enfocada al tema de fracciones, que se imparte durante el proceso de enseñanza – aprendizaje a los alumnos de tercer grado de primaria, en México como complemento de las herramientas educativas tradicionales.
- Ya que la metodología utilizada está enfocada al desarrollo tridimensional de visitas virtuales fue necesario complementar las actividades propuestas en las fase de diseño y desarrollo con lo siguiente:
 - Se agregó el Storyboard con la finalidad de poder facilitar el trabajo a los desarrolladores y agilizar el tiempo de creación de la aplicación en las etapas de modelado, integración y producción de recursos tecnológicos.
 - Se incluyeron los diagramas de navegación de las misiones para representar el recorrido que tiene que hacer el usuario en cada una de ellas, lo anterior permitió favorecer y acelerar la fase de producción de recursos tecnológicos. Ya que teniendo una referencia de cómo va a estar estructurada la misión y los eventos que van a suceder durante su realización, los desarrolladores trabajan de forma más segura y la aplicación no está sujeta a muchas modificaciones.

- En la etapa del modelado de eventos se crearon dos tablas que describen los eventos que son generados y detectados tanto por los usuarios como por los objetos que integran los escenarios respectivamente, además de describir las acciones que desencadenan cada uno de ellos dentro de la aplicación.
- Se generó una clase en C# tipo librería con la definición de los parámetros necesarios para el cálculo de las fracciones matemáticas (denominados, numerador y entero), así como los métodos de las operaciones que se realizan con los números racionales (suma, resta, comparación y equivalencias). Lo anterior, favoreció la optimización de la programación de las misiones.
- La fase de análisis y diseño son las más importantes en el desarrollo de proyectos, debido a que en estas se definen las bases necesarias para todo el concepto de la aplicación. Para el caso del presente trabajo, en estas fases se tuvo que invertir mucho más tiempo de trabajo de lo previsto, ya que no se tuvo un cliente que presentara directamente una problemática o un área curricular de oportunidad, de tal modo que:
 - Se tuvieron que consultar los resultados de la prueba ENLACE para encontrar un área de oportunidad.
 - Analizar un conjunto de artículos especializados para verificar que las características de los requerimientos funcionales fueran aptas para este tipo de aplicaciones de RV.
 - Consultar en la página de la SEP el contenido del programa de matemáticas de tercer año de primaria para verificar los temas específicos de las fracciones matemáticas que iban hacer abordados dentro la aplicación.

- El equipo de desarrollo tuvo que ver un conjunto de películas animadas y videojuegos de categoría “Everyone”, es decir aquellos que son aptos para todo público.

Durante el desarrollo del proyecto se tuvieron las siguientes experiencias:

- Se crearon directorios para clasificar todos los elementos que integran la aplicación, esto permitió tener un proyecto organizado y ubicar cualquier objeto más rápido.
- En el desarrollo de la misión Rainbow fishes, se implementó el uso de las ecuaciones paramétricas de la recta en el espacio:

$$\bullet \quad X = X_0 + at \quad Y = Y_0 + bt \quad Z = Z_0 + ct$$

- Donde cada ecuación pertenece a los ejes X, Y y Z, de un punto cualquiera sobre una recta en el espacio tridimensional.
- Debido a que la misión requería que los peces se movieran al dar clic sobre ellos y simular que eran pescados, se debía calcular la posición que ocupaban en el espacio.
- El trabajar con la versión para estudiante de Unity limita el uso de herramientas que podrían facilitar el desarrollo de los proyectos.
- Es recomendable que se cuente con especialistas de cada área de desarrollo de aplicaciones tridimensionales de RV, por ejemplo: analistas, diseñadores, modeladores, animadores, programadores, etcétera. ya que para un equipo de dos personas la creación de este tipo de proyectos es una tarea demasiado ardua.

Trabajos a futuro

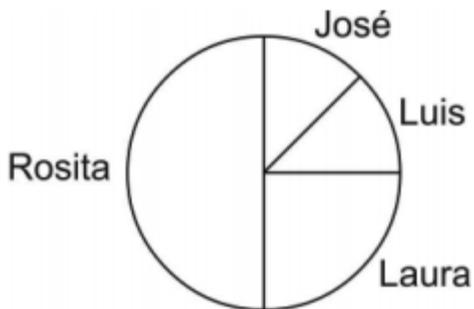
- Optimizar las dimensiones de las texturas para reducir el tamaño de la aplicación, ya que muchas de ellas son de 1024 x1024 pixeles.
- Agregar más actividades en la misión Spatial Coins, es decir, que el usuario pueda comparar más fracciones y no solo la fracción 1/2.
- Agregar controles de volumen.
- Agregar una sección en la cual el usuario pueda seleccionar la música.
- Agregar más indicadores en las misiones para hacer más fácil su realización.
- Agregar un mapa en donde el usuario pueda ver la ubicación de las misiones.
- Agregar más actividades en la misión Rainbow Fishes, es decir, que el usuario compare peces más de dos veces.
- En la misión Cosmic Apples, mostrar el peso de las manzanas cuando el usuario pase el cursor sobre una de ellas y al dar clic la agregue a la canasta.
- En la misión Galactic Labyrinth, hacer las actividades de una forma más dinámica.
- Agregar más misiones en la aplicación con los temas de fracciones: multiplicación y división.
- Eliminar las luces de las escenas de la aplicación con la herramienta “Lightmapping” de Unity y de esta manera reducir el consumo de recursos considerablemente.
- Probar la aplicación con un grupo de muestra de niños que cursen el tercer grado de primaria para validar su funcionamiento y contenido de la misma.
- Realizar pruebas con un grupo de muestra de niños que cursen el tercer grado de primaria para medir la satisfacción que genera en ellos.
- Probar la técnica de modelado 3D procedural, ya que podría reducir el tiempo de desarrollo de escenarios y así poder crear más detalles dentro de ellos.
- Adaptar la aplicación para dispositivos móviles.

Anexo 1

Reactivos de los temas de fracciones en la prueba ENLACE 2013.

Reactivo 11

Rosita repartió un pastel de la siguiente manera:



¿Qué parte del pastel le toco a rosita?

A) $\frac{1}{2}$

B) $\frac{1}{3}$

C) $\frac{1}{4}$

D) $\frac{1}{8}$

Reactivo 18

La mamá de Beto tenía una sandía y la cortó en dos partes iguales para reparirla entre Beto y su amigo Juan. Pero Beto invitó también a Paco y Pepe. La mamá le partió las dos mitades en partes iguales para darles a todos. ¿Qué parte de la sandía le tocó a cada uno de los niños?

A) $\frac{1}{4}$

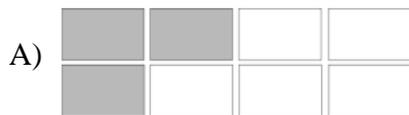
B) $\frac{2}{4}$

C) $\frac{3}{4}$

D) $\frac{4}{4}$

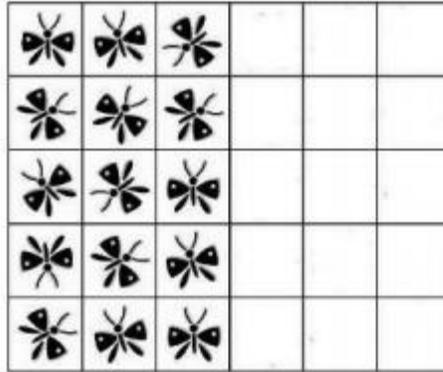
Reactivo 45

Rafael tiene una barra de amaranto con chocolate y quiere compartir con Alfonso tres octavos. ¿Cuál de las siguientes opciones representa la fracción de la barra que Rafael compartirá con Alfonso?



Reactivo 69

María está bordando un mantel como se ve en el siguiente dibujo:



¿Qué fracción representa lo bordado?

A) $\frac{1}{16}$

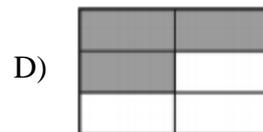
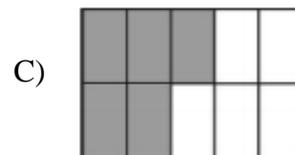
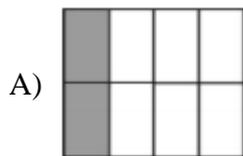
B) $\frac{1}{8}$

C) $\frac{1}{4}$

D) $\frac{1}{2}$

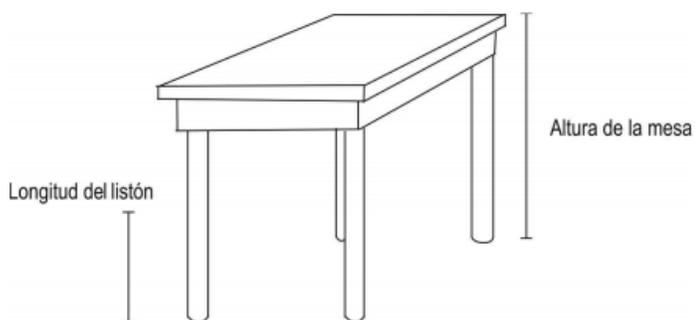
Reactivo 90

Observa las siguientes figuras geométricas que la maestra Laura dibujó. ¿Cuál de ellas, en su parte sombreada, representa $\frac{3}{4}$?



Reactivo 96

Mariana compró una mesa para su casa, utiliza un listón para medirla como se muestra en la siguiente ilustración:

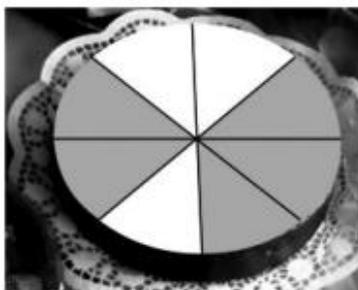


¿Cuántos listones mide la altura de la mesa?

- A) 3 listones.
- B) 2 y medio.
- C) 2 listones.
- D) 1 y medio.

Reactivo 97

En casa de Ciro partieron un pastel y él se comió la parte que se muestra en blanco:



¿Cuál de las siguientes fracciones representa lo que se comió Ciro?

- A) $\frac{8}{3}$ B) $\frac{1}{3}$ C) $\frac{3}{8}$ D) $\frac{5}{3}$

Referencias

- Alvarado, D., Barría, S., Cordella, R. & Lechuga, A. (2007). *Manual de Métodos de optimización para proyectos 3D*. Obtenido de: <http://www.optimizacion3D.info>
- Alvarez, J. (2008). “*Serious games Advergaming, edugaming, training and more*”. Francia. IDATE Consulting and research.
- Alvarez, J. and Rampnoux, O. (2007), “Serious game: just a question of posture?”, Proceedings of Artificial & Ambient Intelligence, AISB’07, Newcastle, Abril, pp. 420-423.
- Antonietti, A., Rasi, C., Imperio, E., & Sacco, M. (2000). The representation of virtual reality in education. *Education and Information Technologies*, 5(4), 317-327. DOI: 10.1023/A: 1012057608694
- Aymerich-Franch, L. (2012). Los juegos en entornos virtuales como herramientas de aprendizaje: estudio de la respuesta emocional de los participantes. *Sphera Pública*, 12,183-197
- Blender Foundation. (2017). About. Obtenido de: Blender.org: <https://www.blender.org/about/>
- Cabero, A. J., Barroso, O. J. & Llorente C. M. (2016). Technology acceptance model & realidad aumentada: estudio en desarrollo. *Revista lasallista de investigación*, 13 (2), 18-26.
- Collins, A., Halversont, R. (2009). The second educational revolution: rethinking education in the age of technology. *Journal of Computer Assisted Learning*, 26, 18-27. DOI: 10.1111/j.1365-2729.2009.00339.x
- CryEngine. (2017). CRYENGINE. The complete solution for next generation game development. Obtenido de Crytek: <https://www.cryengine.com/>

- Cubillo, A.,J, Martin, G.,S., Castro, G., M., Colmenar, S., A. (2014). Recursos digitales autónomos mediante realidad aumentada. *Revista Iberoamericana de Educación a Distancia*. 17(2), 241-274.
- Dickey, M. D. (2005). Three-dimensional virtual worlds and distance learning: two case studies of Active Worlds as a medium for distance education. *British journal of educational technology*, 36(3), 439-451. DOI: 10.1111/j.1467-8535.2005.00477
- Espinoza, E., A. (2009). Los mediadores pedagógicos en la enseñanza de las ciencias: la implementación de un programa educativo multimedia en la enseñanza del sistema circulatorio. *El hombre y la máquina*, 21(32), 20-37.
- Entertainment Software Rating Board, (2017). *ESRB Ratings*. Obtenido de: <http://www.esrb.org/ratings/>
- Fernández Ruiz, M. (2011). Elementos visuales expresivos en la interactividad del videojuego. *Razón y Palabra*, 15(75), febrero-abril.
- Garrido Miranda, J. M. (2013). Videojuegos de estrategia: algunos principios para la enseñanza. *Revista Electrónica de Investigación Educativa*, 15(1), 62–74.
- GIMP. (2017). The Free & open Source Image Editor. Obtenido de GNU Image Manipulation Program: <https://www.gimp.org/>
- González, C., Blanco, F. (2008). Emociones con videojuegos: incrementando la motivación para el aprendizaje. *Teoría de la Educación. Educación y Cultura en la Sociedad de la Información*, 9(3), 69-92.
- Gramigna, A, González, J. (2009). Videojugando se aprende: renovar la teoría del conocimiento y la educación. *Comunicar*. 17(33), 157-164.
- Guijón, C. I. (2006). *Sistema de Modelado Tridimensional basado en Trazos Libres: MOSKIS 3D* (Tesis de pregrado). Universidad De Castilla-La Mancha Escuela Superior De Informática, Castilla, España.
- Gurău, C. (2007). Virtual Reality Applications in Tourism. En Pease, W., Rowe, M. & Cooper, M. (Ed.), *Information and Communication Technologies*

in Support of the Tourist Industry (180-197). USA: IGI Global. DOI: 10.4018/978-1-59904-955-7.ch054.

- Hannes, K., Schmalstieg, D., & Wagner, M. (2006). Construct3D: A Virtual Reality Application for Mathematics and Geometry Education. *Education and Information Technologies*, 5(4), 263-276. DOI: 10.1023/A:1012049406877
- Hui-Zhen, R. & Zong-Fa, L. (2013). Application and Prospect of the Virtual Reality Technology in College Ideological Education. *In Intelligent Systems Design and Engineering Applications, 2013 Fourth International Conference on*. DOI: 10.1109/ISDEA.2013.434.
- Ibanez, M.B., Garcia, J.J., Galan, S., Maroto, D., Morillo, D. & Kloos, C.D. (2011). Design and Implementation of a 3D Multi-User Virtual World for Language Learning. *Educational Technology & Society*, 14(4), 2-10. DOI: 10.1080/09687760802526707.
- Kastoudi, D. (2011). Using a Quest in a 3D Virtual Environment for Student Interaction and Vocabulary Acquisition in Foreign Language Learning. *European Association for Computer-Assisted Language Learning (EUROCALL)*, 20, 87-89.
- Kaufmann, H., & Schmalstieg, D. (2006). Designing immersive virtual reality for geometry education. In *Virtual Reality Conference, 2006* (51-58). IEEE. DOI: 10.1109/VR.2006.48.
- Knutzen, K. B., & Kennedy, D. M. (2012). Designing the self: the transformation of the relational self-concept through social encounters in a virtual immersive environment. *Interactive Learning Environments*, 20(3), 271-292.
- Autodesk Maya. (2017). Basics. Obtenido de Autodesk knowledge network: <https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2017/ENU/Maya/files/GUID-6B531DDB-3440-4216-A322-FB6CD1EA83A1-htm.html?v=2017>

- Morales, A., Alviter, L. E., Hidalgo, C., Amador, J., Zúñiga, J. & Mejía, C. (2015). Metodología de desarrollo evolutivo de escenarios tridimensionales para la creación de una visita virtual para el Centro Universitario UAEM Ecatepec. *Revista Colombiana de Computación*, 16(1), 7-27.
- Morejón, S. (2011). El software educativo un medio de enseñanza eficiente. *Cuaderno de educación y desarrollo*. 3(29).
- Olguin, C.M, Rivera, Z.I, Hernández, M.E. (2006). Introducción a la Realidad Virtual. *Polibits*, (33), 11-15.
- Organista Sandoval, J. (2010). Análisis del uso de objetos de aprendizaje en las materias de matemáticas y física de bachillerato. *Revista Electrónica Sinéctica*, 34, 1-16.
- Papert, S. (1984). New Theories For New Learnings. *National Association of School PsycholOgists' Conference*. Canada, Estados Unidos.
- Pérez, M., Zabre E. & Islas E. (2004). Realidad Virtual: Un panorama general. *Boletín IIE, Instituto de Investigación Electrónicas*, 28 (2), 39-44.
- Rahimi, A., Golshan, N. & Mohebi, H. (2013). Virtual Reality as a Learning Environment in Iranian EFL Context: Personal, Technical, and Pedagogical, *Procedia - Social and Behavioral Sciences*, 136, 234 – 239. DOI: 10.1016/j.sbspro.2014.05.320.
- Ramos, N.M. (2003). Realidad Virtual. *Enterate*, 2(24). 23-25.
- Pallares Muñoz, M., R., Rodriguez Calderon, W. (2006). Diseño de un software educativo para el análisis de sistemas hidráulicos a presión en ingeniería civil. *Ingeniería e Investigación*.26 (003), 58-66.
- Secretaria de Educación Pública. (2015). Currículo Básica. Obtenido de Matemáticas:
<http://www.curriculobasica.sep.gob.mx/index.php/matematicas3?sid=253>
- Shi, Y. L. (2014). Application of Virtual Reality Technology in Medical Education. En Li, S., Jin, Q., Jiang, X. & Park, J. (Ed), *Frontier and Future*

Development of Information Technology in Medicine and Education (467-476). Netherlands: Springer. DOI: 10.1007/978-94-007-7618-0_46.

- Pantelidis, V. (1996). Suggestions on when to use and when not to use virtual reality in education. *VR in the schools* (2), 1-18.
- Unreal Engine. (2017). What is unreal engine 4. Obtenido de Unreal Engine: <https://www.unrealengine.com/what-is-unreal-engine-4>
- Wang, G., Wu, H. (2008). Research of Education Game Based on Virtual Reality. *College of Computer and information engineering, Zhejiang GongShang University* (2). 921-924. DOI 10.1109/PACIIA.2008.243
- Watkins, A. (2011). *Creating Games with Unity and Maya: how to develop fun and marketable 3D games*. Oxford: Elsevier.
- Zapatero, G. D. (2011). La realidad virtual como recurso y herramienta útil para la docencia y la investigación. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología* (6). 17-23.
- Zhan, Z. (2011). The Application of VR on Distance Education. En Chen, R. (Ed.), *Intelligent Computing and Information* (78-83). Berlin Heidelberg: Springer. DOI: 10.1007/978-3-642-18134-4_13