

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MEXICO
UNIDAD ACADÉMICA PROFESIONAL NEZAHUALCÓYOTL



Licenciatura en Ingeniería en Sistemas Inteligentes

“Diseño y Desarrollo de Entorno Virtual Tridimensional basado en un Agente Inteligente Reactivo en apoyo al Sector Cultural”

Tesis que para obtener el título de:

Ingeniero en Sistemas Inteligentes

Presenta:

Tania Jezabel López García

Asesora:

Dra. Rosa María Rodríguez Aguilar

Cd. Nezahualcóyotl, México, Marzo de 2018

Nuestra intuición sobre el futuro es lineal. Pero la realidad de la tecnología informática es exponencial, y eso hace una gran diferencia. Si doy 30 pasos de manera lineal, llego a 30. Si doy 30 pasos exponencialmente, llego a mil millones.

- **Ray Kurzweil.**

A

GRADECIMIENTOS

La gratitud es un hermoso valor inculcado y debe ser expresado plenamente del corazón; reservado para aquellos que acompañaron o estuvieron presentes en nuestra vida. Primeramente, a Dios gracias, que me permites sonreír ante todos mis logros que son resultado de tu ayuda, y cuando caigo y me pones a prueba, aprendo de mis errores y me doy cuenta que todo responde a una mejoría como ser humano, y a crecer de diversas maneras.

Quiero extender mi más grande y profundo agradecimiento a mis Padres; gracias a mi madre por estar dispuesta a aconsejarme y motivarme cada largo y agotador día de estudio; gracias a mi padre por siempre desear y anhelar lo mejor para mi vida, gracias por cada consejo y por cada una de las palabras de apoyo que me han guiado durante mi camino. Muchas gracias por sus palabras de aliento, por su ejemplo, por su lucha y dedicación, por su responsabilidad, por su generosidad, por su preocupación, por ser mi más grande inspiración, por ser los principales promotores de mis sueños, por cada día confiar y creer en mí y en mis expectativas.

Todos sus imparables esfuerzos y su gran amor incondicional, han sido mi motor a lo largo de mi vida. Quienes me han educado y me han proporcionado todo y cada cosa que he necesitado. Las enseñanzas y valores que me han inculcado y que aplico día con día. Los amo y sin ustedes, nada soy.

Gracias a mi familia, quienes siempre se mostraron interesados y atentos con cada avance para concluir con esta Tesis. Mis hermanos que son un ejemplo de lucha y esfuerzo; mi hermana que siempre es un gran ejemplo de constancia, dedicación, compromiso, responsabilidad y trabajo inagotable. También, quiero agradecer a mis amigos, quienes siempre se mostraron comprensibles, atentos y dispuestos a ayudar; descubrí grandes amistades en este camino y siempre estaré agradecida por ello. Específicamente, gracias por tus consejos y guía Alberto, aprecio y valoro mucho tu amistad.

Por último, pero no menos importante, quiero agradecer la ayuda constante e invaluable de mi tutora de Tesis, Dra. Rosa María Rodríguez A., sin su organización, enfoque y conocimiento basto no hubiese podido direccionar y culminar mi proyecto. Quiero agradecer también, por los valiosos aportes a mis sinodales: Mtra. Alejandra Puebla Urbina, gran profesora y amiga, y al Dr. Ricardo Rico Molina, excelente persona y académico. En general, gracias infinitas a mis profesores de carrera, quienes siempre estuvieron dispuestos a compartir su conocimiento de la mejor manera.

Gracias a la vida por la culminación de esta etapa, gracias a todas las personas que me apoyaron y creyeron en la realización de este proyecto, que representa un importante escalón en el largo camino de mi preparación profesional.

R ESUMEN

Paseo RV-CNA es una aplicación elaborada mediante técnicas de desarrollo de software y juegos para fungir como herramienta auxiliar de difusión de espacios culturales, específicamente del inmueble Centro Multimedia. El desarrollo de esta tesis está influenciada por las tecnologías de Realidad Virtual (RV), que se han vuelto relevantes en el contexto de aplicaciones multimedia de bajo coste.

El objetivo del presente proyecto es la inclusión de un mayor número de personas para conocer inmuebles culturales relevantes con una oferta educativa de calidad; se pensó en una aplicación de escritorio con RV, cuya funcionalidad no dependa de dispositivos específicos con los que pueda o no contar el público.

Actualmente se encuentra en desarrollo las aplicaciones con este tipo de tecnología, a pesar de no existir una metodología formal para la creación de este tipo de sistemas. Sin embargo, existen metodologías de sistemas de información que se han podido adaptara al desarrollo de estas aplicaciones. En este caso, se utilizó la metodología basada en prototipos, ya que su estructura se adaptaba muy bien a las especificaciones del presente sistema. Esta metodología comprende dos etapas de desarrollo a) visión general del sistema (cuenta con la mayor parte de las indicaciones del producto final) y b) retroalimentación y detalles, (programación de los eventos requeridos, pruebas de funcionalidad, modificación de observaciones, etc.).

Por otro lado, los procesos internos estuvieron definidos mediante la adaptación de una arquitectura de videojuegos, cuyos módulos determinaron el comportamiento del sistema en conjunto con el desarrollo de la técnica de IA, específicamente la implementación de un agente inteligente reactivo, cuya función principal es fungir como auxiliar en la navegación del usuario dentro del entorno creado para la exploración del recinto elegido, *Cenart*.

Para poder probar la aplicación creada y poder comprobar nuestro objetivo, se utilizaron algunos métodos de prueba de usabilidad de sistema informáticos, los cuales son válidos para la evaluación de un producto con las características propias de un sistema en RV. Gracias a los datos recabados en la prueba de usabilidad del sistema desarrollado, se puede concluir que, aunque el sistema presenta deficiencias, estas no son significativas para el objetivo final de la aplicación. Es decir, se cuenta con un sistema funcional y aceptado por una muestra de 32 personas de la población considerada como objetivo, con una media de 85% en una evaluación formal con distintos métodos.

A

STRACT

Walking RV-CNA is an application elaborated through a development techniques of software and games to serve as an auxiliary tool for the dissemination of cultural spaces, specifically the Multimedia Center buildings. The development of this thesis is influenced by Virtual Reality (VR) technologies, which have become relevant in the context of low-cost multimedia applications.

The objective of this project is the inclusion of a greater number of people for knowing relevant cultural buildings offering a quality of educational of properties; it was thought a desktop application with RV, which the functionality does not depend on specific devices with which the public can or cannot count.

By the moment the applications with this type of technology are developing, in spite of unexisting formal methodology for the creation. However, there are methodologies of information systems that have been able to adapt the development of these applications. In this case, the methodology based on prototypes was used, since its structure adapted very well to the specifications of the present system. This methodology includes two stages of development a) general view of the system (it has most of indications at the final product) and b) feedback and details, (programming the required events, functionality tests, modification of observations, etc.).

On the other hand, the internal processes were defined through the adaptation of AI technique of videogames, specifically the implementation of an intelligent reactive agent architecture, which the modules are determined by the behavior of the system in conjunction with the development, which the main function is perform as an assistant in user navigation within the created environment for the exploration of the chosen site, in this case Cenart.

In order to test the application created and be able to verify our objective, some usability computer systems test methods were used, which are valid for the evaluation of a product with the characteristics of RV system. Thanks to the data collected in the usability test into the developed system, it could be concluded, although the system has deficiencies, these are not significant for the final objective of the application. In other words, there is a functional system accepted by a sample of 32 people from the target population, with an average of 85% in a formal evaluation with different methods.

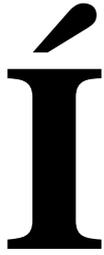
C ONTENIDO

RESUMEN	V
ÍNDICE DE FIGURAS	XII
ÍNDICE DE TABLAS	XV
CAPÍTULO I: INTRODUCCIÓN	17
1.1 PLANTEAMIENTO DEL PROBLEMA	18
1.2 HIPÓTESIS	19
1.3 JUSTIFICACIÓN	20
1.4 OBJETIVOS	26
1.4.1 Objetivo General	26
1.4.2 Objetivos Específicos	26
1.5 LIMITACIONES Y ALCANCES	27
1.6 DESARROLLO DE LA DOCUMENTACIÓN	28
CAPÍTULO II: ANTECEDENTES Y ESTADO DEL ARTE	29
2.1 ANTECEDENTES	30
2.2 TECNOLOGÍAS PRECURSORAS EN REALIDAD VIRTUAL	32
2.2.1 SENSORAMA	32
2.2.2 ESPADA DE DAMOCLES	33
2.2.3 ASPEN MOVIE MAP	34
2.2.4 VRML (VIRTUAL REALITY MODELING LANGUAGE)	34
2.3 REALIDAD VIRTUAL: DESARROLLO HISTÓRICO	36
2.4 TRABAJOS RELACIONADOS EN LA EDUCACIÓN	39
2.4.1 DESARROLLO DE MUNDOS VIRTUALES EN EL ÁREA DE MÁS SOBRE CIENCIA DEL MUSEO VIRTUAL 3D DEL REHILETE	39
2.4.2 SISTEMA DE REHABILITACIÓN CARDIACA AUMENTADO POR REALIDAD VIRTUAL	41
2.4.3 ANÁLISIS Y DISEÑO DE LA INTERFAZ PARA UN SISTEMA DE APRENDIZAJE COLABORATIVO APOYADA POR UN AGENTE TUTOR INTELIGENTE	43
2.4.4 ANIMACIÓN Y RECONSTRUCCIÓN SOBRE LA ZONA PREHISPÁNICA DE MONTE ALBÁN, OAXACA	44
2.4.5 MODELO DE UN ENTORNO VIRTUAL INTELIGENTE BASADO EN LA PERCEPCIÓN Y EL RAZONAMIENTO DE SUS ELEMENTOS CON UN PERSONAJE PARA LA GENERACIÓN DE REALISMO	46
CAPÍTULO III: MARCO TEÓRICO	47
3.1 SISTEMAS EN REALIDAD VIRTUAL	48
3.1.1 ENTORNOS VIRTUALES TRIDIMENSIONALES (EV3D)	51
3.1.1.1 Modelo Geométrico	52
3.1.1.2 Modelo Cinemático	53
3.1.1.3 Modelo Comportamental	55
3.1.2 ENTORNOS VIRTUALES INTELIGENTES TRIDIMENSIONALES (EVI3D)	58

3.1.3 SIMULACIÓN COMPORTAMENTAL INTELIGENTE EN EV	60
3.1.3.1 Búsqueda Heurística	60
3.1.3.2 Búsqueda Local	62
3.1.3.3 Agentes Inteligentes	64
3.1.4 AGENTES INTELIGENTES EN ENTORNOS VIRTUALES	65
3.1.4.1 Introducción	65
3.1.4.2 ¿Qué es un Agente inteligente?	66
3.1.4.2.1 Agentes de Interfaz	69
3.1.4.2.2 Agentes Deliberativos	70
3.1.4.2.3 Agentes Reactivos	71
3.1.4.2.4 Agentes Híbridos	73
3.1.4.2.5 Agentes Virtuales Inteligentes	75
3.1.4.3 Propiedades de los Entornos	76
3.4 PROCESO DE MODELADO 3D	77
3.4.1 INTRODUCCIÓN	77
3.4.2 CONTEXTO MATEMÁTICO	77
3.4.3 CICLO DE PRODUCCIÓN 3D	78
3.4.3.1 Pre-Producción	78
3.4.3.2 Producción	79
3.4.3.3 Post-Producción	79
3.4.4 ETAPAS PRINCIPALES DE MODELADO 3D	80
3.4.4.1 Modelado	80
3.4.4.1.1 Modelos constituidos por polígonos	80
3.4.4.1.2 Modelos determinados por curvas matemáticas	81
3.4.4.1.3 Operaciones Booleanas	82
3.4.4.2 Materiales	83
3.4.4.2.1 Parámetros Principales en los Materiales	83
3.4.4.2.2 Manejo de múltiples materiales	85
3.4.4.2.3 Materiales especiales	86
3.4.4.3 Texturas	87
3.4.4.4 Iluminación	88
3.4.4.4.1 Iluminación Local	88
3.4.4.4.2 Iluminación General	88
3.4.4.5 Ambiente	89
3.4.5 SOFTWARE DE MODELADO 3D	90
3.4.5.1 AutoCAD	91
3.4.5.2 Maya	91
3.4.5.3 Sketchup	92
3.4.5.4 Revit	93
3.4.5.5 3DS Max	94
3.4.5.6 Blender	95
3.4.5.7 Tabla Comparativa y Selección Final	96
3.5 DESARROLLO INTERACTIVO: UNITY	97
3.5.1 APLICACIÓN DE ASSETS	98
3.5.1.1 Escalas y Unidades	98
3.5.1.2 Archivos y Objetos	99
3.5.1.3 Meshes	99
3.5.1.4 Texturas	100
3.5.1.5 Materiales	101
3.5.2 NAVEGACIÓN Y PATHFINDING	102
3.5.3 ÁREA DE RENDERIZACIÓN	103
3.5.3.1 Cubemap	103
3.5.3.2 Oclusión Portals	103
3.5.3.3 Skybox	103

3.5.3.4 LOD Group	104
3.5.4 ÁREA DE MEMORIA	104
3.5.5 SCRIPTING	105
3.5.5.1 Lenguajes de Programación permitidos en Unity	106
3.5.5.1.1 Lenguaje C#	106
3.5.5.1.2 JavaScript	106
3.5.5.1.3 Boo	107
3.5.5.1.4 Unity Script	107
3.5.5.1.5 Tabla Comparativa y Elección Final	107
CAPÍTULO IV: ARQUITECTURA Y METODOLOGÍA DE DESARROLLO	109
4.1 DESCRIPCIÓN DE LA METODOLOGÍA IMPLEMENTADA	110
4.1.1 INVESTIGACIÓN PRELIMINAR	111
4.1.2 FASE A)	112
4.1.2.1 Requerimientos y Especificaciones	112
4.1.2.2 Diseño y Construcción	113
4.1.2.3 Evaluación	116
4.1.2.4 Modificación	118
4.1.3 FASE B)	119
4.1.3.1 Diseño Detallado	119
4.1.3.2 Codificación de Programas	120
4.1.3.3 Prueba de Sistema	120
4.1.3.4 Implementación y Mantenimiento	120
4.1.4 MODELO DE INTEGRACIÓN DE UN AGENTE REACTIVO	121
4.2 IMPLEMENTACIÓN Y DESARROLLO	126
4.2.1 ARQUITECTURA DE SISTEMAS DE REALIDAD VIRTUAL	126
4.2.1.1 Componentes de un Sistema de RV	126
4.2.1.2 Funcionamiento de un Sistema de Realidad Virtual	129
4.2.1.3 Niveles de simulación, interacción e inmersión	130
4.2.2 ARQUITECTURA ENTORNO VIRTUAL INTELIGENTE CON AGENTES	133
4.2.2.1 Introducción	133
4.2.2.2 Requerimientos funcionales y no funcionales	134
4.2.2.3 Estructura General del Sistema	135
4.2.2.4 Descripción del Sistema	136
4.2.2.4.1 Módulo de Comunicación	136
4.2.2.4.1.1 Sensores y Actuadores	136
4.2.2.4.1.2 Interacción	137
4.2.2.4.1.3 Simulación	138
4.2.2.4.2 Modelo Geométrico Tridimensional	139
4.2.2.4.2.1 Definir objetos y personaje dentro del EV	140
4.2.2.4.2.2 Modelado 3D	141
4.2.2.4.2.3 Asignación de texturas y materiales	145
4.2.2.4.2.4 Definición del Nivel de Física	147
4.2.2.4.3 Módulo Gráfico	148
4.2.2.4.3.1 Procesamiento de la Mesh Geometry	149
4.2.2.4.3.1 Rendimiento Gráfico	151
4.2.2.4.4 Módulo de Movimiento	154
4.2.2.4.4.1 Propiedades	154
4.2.2.4.4.1 Eventos	155
4.2.2.4.5 Módulo de Comportamiento	155
4.2.2.4.5.1 Objetivos	157
4.2.2.4.5.2 Creencias	158
4.2.2.4.5.3 Acciones posibles	158

4.2.2.4.6 Técnica de IA Incorporada: Agente Reactivo Basado en Modelos	160
4.2.2.4.6.1 Descripción Detallada del “Agente Usuario”	160
4.2.2.4.6.2 Algoritmo Simplex Nelder-Mead	162
4.2.2.4.6.3 Sistema de Navegación.	165
4.2.2.4.7 GUI (Graphical User Interface)	167
4.2.2.4.7.1 Desarrollo de la interfaz 2D para el sistema CNA RV	167
4.2.2.4.7.2 Diseño de Interfaz	168
4.2.2.4.7.3 Interfaz orientadas a la Realidad Virtual	171
CAPÍTULO V. PRUEBAS Y RESULTADOS	172
5.1 CRITERIOS DE ACEPTACIÓN DE UN PRODUCTO SOFTWARE	173
5.2 MÉTODOS DE USABILIDAD DE SOFTWARE	174
5.2.1 EFICIENCIA DE LOS MÉTODOS CLÁSICOS DE EVALUACIÓN APLICABLES	175
5.3 CARACTERÍSTICAS RELEVANTES EN LA EVALUACIÓN DE UN SISTEMA EN RV	176
5.4 DESARROLLO DE LA EVALUACIÓN DE USABILIDAD	178
5.4.1 ASPECTOS MEDIBLES EN LA EVALUACIÓN DE USABILIDAD.	179
5.4.2 JUSTIFICACIÓN DE LOS MÉTODOS IMPLEMENTADOS	181
5.4.2.1 ESPECIFICACIONES DEL TEST	182
5.4.2.2 ESPECIFICACIONES DEL CUESTIONARIO	184
5.4.3 CONDICIONES DE APLICACIÓN	185
5.4.4 PARTICIPANTES	186
5.4.5 PERFIL DE LOS PARTICIPANTES	189
5.5 RESULTADOS OBTENIDOS	189
5.5.1 ANÁLISIS DE DATOS: CUESTIONARIO	190
5.5.2 ANÁLISIS DE DATOS: TEST DE VALIDACIÓN	192
5.5.3 ASPECTOS LIMITANTES	199
CAPÍTULO VI. CONCLUSIONES Y FUTURAS LÍNEAS DE INVESTIGACIÓN	200
6.1 FUTURAS LINEAS DE INVESTIGACIÓN	203
BIBLIOGRAFÍA	204
APÉNDICE A: TEST DE USABILIDAD	209

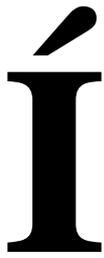


NDICE DE FIGURAS

<i>Fig. 1.1 Asistencia a Cursos o Talleres Culturales (ENCCUM2012). Elaboración propia.</i>	22
<i>Fig. 1.2 Asistencia a Recintos Culturales y Espectáculos. Elaboración propia.</i>	23
<i>Fig. 1.3 Medios de comunicación de difusión de Actividades Culturales. Elaboración propia.</i>	24
<i>Fig. 1.4 Gráfica del conocimiento de si se imparten o se han impartido actividades culturales. Elaboración propia.</i>	25
<i>Fig. 1.5 Opinión de los encuestados acerca de la calidad de talleres y/o cursos. Elaboración propia.</i>	25
<i>Fig. 2.1 Maquina Sensorama de Heilig (s.f.) (Recuperado de: Blog stampaprint.fr, publicación 2016)</i>	33
<i>Fig. 2.2 Maquina "Espada de Damocles" (s.f.) (Recuperado de: geekster.be, 2016)</i>	33
<i>Fig. 2.3 Project Aspen Moviemap, M.I.T. (1981) (Recuperada de: naimark.net)</i>	34
<i>Fig. 2.4 Figuras Geométricas creadas con el lenguaje VRM (Ortega, 2011)</i>	35
<i>Fig. 2.5 Interfaz de la App Museo del Rehilete (Cruz Rodríguez, 2003)</i>	39
<i>Fig. 2.6 Programación de eventos en objetos 3D con VRML. (Cruz Rodríguez, 2003).</i>	40
<i>Fig. 2.7 Representación de Usuario dentro del Entorno Virtual InTrainer (Ortiz S. Miguel A., 2010)</i>	41
<i>Fig. 2.8 Esquema del sistema de guía tipo resorte.</i>	42
<i>Fig. 2.9 Interfaz General del Software de Aprendizaje (A. Garduño Enrique, 2012)</i>	43
<i>Fig. 2.10 Vista de la Cámara de un Ángulo Externo. (J. Manuel Jiménez y Luis F. Mendoza, 2014)</i>	44
<i>Fig. 2.11 Textura del entorno y textura de transparencia para el mapa. (J. Manuel Jiménez y Luis F. Mendoza, 2014)</i>	45
<i>Fig. 2.12 Rutas organizadas por canales para mostrar la posición del usuario en tiempo real. (J. Manuel Jiménez y Luis F. Mendoza, 2014)</i>	45
<i>Fig. 2.13 Interfaz de Software basado en metodología SAMI (M. Santiago Sandra, 2015)</i>	46
<i>Fig. 3.1 Aplicaciones y Usos de la RV. (Adaptación de Flores C. Jesús A, 2014).</i>	48
<i>Fig. 3.2 Esquema Clásico de Renderización soportada por Hardware de Procesamiento Gráfico dentro de un EV3D. Elaboración propia.</i>	52
<i>Fig. 3.3 a) Esquema de Carga de un Modelo Geométrico en un EV. b) Ejemplo de grafo de escena jerárquico. Elaboración propia.</i>	53
<i>Fig. 3.4 Principales características cinemáticas dentro de un EV3D. Elaboración propia.</i>	54
<i>Fig. 3.5 Tabla comparativa del Modelo Comportamental en sus dos categorías. Elaboración Propia.</i>	56
<i>Fig. 3.6 Ciclo de comportamiento de un personaje inteligente. Elaboración propia.</i>	59
<i>Fig. 3.7 Representación de búsqueda heurística A*. Elaboración propia.</i>	62
<i>Fig. 3.8 Representación de Búsqueda Local. Elaboración Propia.</i>	63
<i>Fig. 3.9 Arquitectura General de un Agente. (Adaptación de Inteligencia Artificial por José T. Palma)</i>	68
<i>Fig. 3.10 Diagrama de Control dentro de un Agente BDI. (Adaptación de Inteligencia Artificial por José T. Palma)</i>	71
<i>Fig. 3.11 Diagrama de Control de un Agente Reactivo (Adaptación de Inteligencia Artificial por José T. Palma)</i>	72
<i>Fig. 3.12 Arquitectura de Subsunción. (Adaptación de Inteligencia Artificial por José T. Palma)</i>	73
<i>Fig. 3.13 Arquitectura Híbrida con Tres Capas. (Adaptación de Inteligencia Artificial por José T. Palma)</i>	74
<i>Fig. 3.14 Esquema del Ciclo de producción 3D. (González Morcillo, Carlos & Vallejo Fernández, David, 2009).</i>	78
<i>Fig. 3.15 Ejemplos de bocetos de estructuras arquitectónicas y personajes para Modelado 3D. (Autor: Alejandro García Melero, 2014)</i>	79

Fig. 3.16 Ejemplos de primitivas predefinidas en un Software de Modelado 3D. (Blender Foundation, 2017)	80
Fig. 3.17 Aplicación de operaciones booleanas en figuras 3D. (Ton Roosendaal, 2014)	82
Fig. 3.18 Principio básico en la renderización de una imagen. (Ton Roosendaal, 2014)	83
Fig. 3.19 Diferentes tipos de malas en diferentes vistas con materiales distintos. (Blender Foundation, 2017)	85
Fig. 3.20 Tratamiento de vértices mediante efectos Halo en modelado 3D. (Ton Roosendaal, 2014 & Bungie Studios, 2001)	86
Fig. 3.21 Aplicación de lens flares en imágenes en 3D (Ton Roosendaal, 2014 & David Jones, 2013)	86
Fig. 3.22 Mapeo de textura en un objeto 3D. (Chandrajit Bajaj & Insung Ihm, 2013)	87
Fig. 3.23 Iluminación local sobre objeto 3D. (Arnold Gallardo, 2001)	88
Fig. 3.24 Iluminación global sobre la escena. (Arnold Gallardo, 2001)	88
Fig. 3.25 Efectos de ambiente niebla y estrellado. (Ton Roosendaal, 2014)	89
Fig. 3.26 Figura Geométricas Básicas en Software de Modelado 3D. (Blender Foundation, 2016)	90
Fig. 3.27 Barra de Menú de Software Maya (Recuperada de: FormacionProfesional.info, 2015)	92
Fig. 3.28 Interfaz de Sketchup (Intec, 2014)	93
Fig. 3.29 Interfaz de Revit (Autodesk, 2014//Licencia CC BY-NC-SA 3.0)	93
Fig. 3.30 Interfaz de 3D Max Studio (Francisco Calero, 2012)	94
Fig. 3.31 Interfaz de Software Blender. (Blender Foundation).	96
Fig. 3.32 Interfaz de Unity. (Unity Technologies)	97
Fig. 3.33 Vista Simple de la memoria utilizada a través de Unity. (Captura de Pantalla Software Unity)	104
Fig. 3.34 Porcentajes de aplicación de lenguajes de programación en Unity. (Unity Blogs, 2014).	108
Fig. 4.1 Esquema de la Metodología orientada a Prototipos. (Elaboración propia).	110
Fig. 4.2 Esfera aplicando diferentes funciones de transformación: mover, girar, escalar respectivamente. (Elaboración propia mediante el software Blender).	115
Fig. 4.3 Diagrama de interfaces del sistema. (Elaboración propia).	117
Fig. 4.4 Estructura de un Agente Inteligente Simple. (Elaboración propia).	121
Fig. 4.5 Esquema de un Agente Reactivo Simple. (Elaboración propia).	122
Fig. 4.6 Esquema de un Agente Reactivo basado en Modelos. (Elaboración Propia).	124
Fig. 4.7 Arquitectura de un Sistema de Realidad Virtual. (Adaptación de Brunet, 2012).	126
Fig. 4.8 Estructura General del Sistema CNA RV. (Elaboración Propia).	135
Fig. 4.9 Esquema General de un Objeto Tridimensional. (Elaboración Propia).	139
Fig. 4.10 Planos del inmueble y modelo 3D (vista wireframe). (Elaboración propia).	141
Fig. 4.11 Diagrama de transición de escenas de la aplicación. (Elaboración propia).	144
Fig. 4.12 Texturas predominantes utilizadas en el modelo 3D. (Elaboración propia).	145
Fig. 4.13 Ejemplo de creación de un material. (Elaboración propia).	145
Fig. 4.14 Proceso de Texturización con UV Mapping en Blender. (Elaboración propia).	146
Fig. 4.15 Esquema de jerarquización computacional en una aplicación 3D. (Adaptación de Sivunen & Nordback, 2014).	148
Fig. 4.16 Proceso gráfico de los shaders. (Ziff Davis, 2017).	151
Fig. 4.17 Ventana game con las estadísticas de renderizado. (Elaboración propia).	152
Fig. 4.18 Información de ventana para renderizar estadísticas. (Adaptación Unity Technologies, 2017).	152
Fig. 4.19 Ventana "Profiler" con el estudio de ejecución del entorno virtual. (Elaboración propia a partir de Unity 5.5.2).	153
Fig. 4.20 Algunos mensajes que aparecen a lo largo del recorrido en el entorno. (Elaboración propia).	155
Fig. 4.21 Cámara y controlador en primera persona de Unity. (Unity Technologies, 2017).	156
Fig. 4.22 Máquina de estados del comportamiento lógico en el modelo 3D. Elaboración propia.	159
Fig. 4.23 Diagrama detallado del "agente usuario" basado en modelos. (Elaboración Propia).	160
Fig. 4.24 Diagrama de flujo del Algoritmo NM. (Elaboración propia).	163

<i>Fig. 4.25 Diagrama de la Clase de MonoBehaviour de Unity. (Adaptación Unity Videojuegos, 2014).</i>	166
<i>Fig. 4.26 Pantallas de Interfaz principal de la app CNA-RV. (Elaboración propia).</i>	167
<i>Fig. 4.27 Conectividad de eventos entre pantallas de la interfaz principal. (Elaboración propia).</i>	169
<i>Fig. 4.28 Diagrama Detallado de MonoBehaviour y GUI. Adaptación de (Richard Fine, 2012).</i>	170
<i>Fig. 5.1 Esquema de grado de aceptación de un producto. (Adaptación de Pereira, 2002).</i>	173
<i>Fig. 5.2 Descripción gráfica de configuración del entorno físico de aplicación. (Elaboración propia).</i>	186
<i>Fig. 5.3 Dispositivos implementados en la aplicación del Test de Usabilidad.</i>	186
<i>Fig. 5.4 Gráfica acerca del uso de internet respecto al nivel escolar. (INEGI, 2016).</i>	187
<i>Fig. 5.5 Usuario de computadora en el Edo. De México, por nivel escolar. (INEGI, 2016)</i>	188
<i>Fig. 5.6 Usuarios de computadora por entidad federativa seleccionada, clasificados por nivel de escolaridad. (INEGI, 2015)</i>	188
<i>Fig. 5.7 Promedio de los porcentajes obtenidos por nivel escolar y sexo. (Elaboración propia).</i>	191
<i>Fig. 5.8 Promedios por aspectos evaluados clasificados por nivel de escolaridad. Elaboración propia.</i>	197
<i>Fig. 5.9 Análisis de la desviación estándar y calificaciones generales por reactivo. Elaboración propia.</i>	198



NDICE DE TABLAS

<i>Tabla 1.1 Interés en Actividades Culturales más representativas. Elaboración propia.</i>	24
<i>Tabla 2.1 Avance Histórico de la Realidad Virtual. Adaptación de Cardozo y Ortigueira (2004 y 2009).</i>	38
<i>Tabla 3.1 Niveles de interacción en relación funcional usuario-sistema RV. (Adaptación de Luna, Jaqueline, 2012)</i>	50
<i>Tabla 3.2 Seudocódigo de algoritmo de búsqueda heurística A*. Elaboración propia.</i>	61
<i>Tabla 3.3 Inconvenientes y posibles soluciones encontradas en el algoritmo de búsqueda local.</i>	63
<i>Tabla 3.4 Tabla comparativa de métodos de simulación comportamental inteligente en EV. Elaboración Propia.</i>	65
<i>Tabla 3.5 Esquema General de funcionamiento de un Agente de Interfaz. Elaboración propia.</i>	70
<i>Tabla 3.6 Ciclo de ejecución de un Agente Reactivo. Elaboración propia.</i>	72
<i>Tabla 3.7 Tipos de shaders básicos en materiales para modelos 3D. (Adaptación de: Mike Geig, 2013)</i>	84
<i>Tabla 3.8 Propiedades generales del sombreado en materiales para modelos 3D. (Adaptación de: Mike Geig, 2013)</i>	85
<i>Tabla 3.9 Tabla comparativa de Software's 3D. Elaboración propia.</i>	96
<i>Tabla 3.10 Propiedades de Skybox en Unity. (Unity Technologies, Publication 5.3-Q 2016)</i>	103
<i>Tabla 3.11 Tabla comparativa de lenguajes de desarrollo en Unity. Elaboración propia.</i>	108
<i>Tabla 4.1 Análisis del modelo del agente reactivo. (Elaboración propia).</i>	123
<i>Tabla 4.2 Seudocódigo del algoritmo d un Agente Reactivo Basado en Modelos. Elaboración propia.</i>	125
<i>Tabla 4.3 Acciones primitivas entre entorno virtual y la visión lógica del sistema. (Elaboración propia).</i>	137
<i>Tabla 4.4 Elementos que constituyen una simulación por computadora. (Mejía Luna Jaqueline, 2012)</i>	138
<i>Tabla 4.5 Distribución del Modelo del Entorno Virtual. (Elaboración Propia).</i>	140
<i>Tabla 4.6 Funciones de propiedades físicas implementadas. (Adaptación de Unity Technologies, 2017).</i>	148
<i>Tabla 4.7 Función GetComponent de Unity para el manejo de mesh. (Unity Technologies, 2017).</i>	150
<i>Tabla 4.8 Atributos modificables en la función mesh. (Unity Technologies, 2017).</i>	150
<i>Tabla 4.9 Pseudocódigo de las funciones de controlador de personaje principal. Adaptación de trabajo Controlador en Ira persona (Lynn Thompson, 2015).</i>	157
<i>Tabla 5.1 Métodos de Usabilidad para el Desarrollo de Software. (Elaboración propia).</i>	174
<i>Tabla 5.2 Eficiencia de los Métodos aplicables según los Criterios de Usabilidad. (Zumaquero, 2010)</i>	175
<i>Tabla 5.3 Métodos de Evaluación de Usabilidad y su Aplicación a Diferentes Tipos de Software. (Zumaquero, 2010)</i>	176
<i>Tabla 5.4 Estándares de usabilidad de sistemas informáticos establecidos por ISO, 2da Parte. (Pereira, 2002)</i>	178
<i>Tabla 5.5 Aspectos a Evaluar del Sistema CNA RV. (Elaboración Propia).</i>	180
<i>Tabla 5.6 Especificaciones del Test de Validación. (Adaptación de Pereira, 2002).</i>	182
<i>Tabla 5.7 Reactivos del Test propicios para el presente sistema "CNA RV". (Elaboración propia).</i>	184
<i>Tabla 5.8 Reactivos del Cuestionario. (Elaboración propia).</i>	184
<i>Tabla 5.9 Ficha de Detalles Técnicos de Muestreo. (Elaboración propia).</i>	187

<i>Tabla 5.10 Grupos creados a partir de la muestra general. (Elaboración Propia).</i>	187
<i>Tabla 5.11 Perfiles de Participantes basados en la experiencia con app. (Elaboración Propia).</i>	189
<i>Tabla 5.12 Perfil de los participantes de prueba. Elaboración propia.</i>	190
<i>Tabla 5.13 Datos Obtenidos mediante la Técnica de Cuestionario. Elaboración Propia.</i>	191
<i>Tabla 5.14 Calificaciones y promedio de la evaluación obtenida mediante el Test de Usabilidad (Nivel Primaria). Elaboración Propia.</i>	192
<i>Tabla 5.15 Calificaciones y promedio de la evaluación obtenida mediante el Test de Usabilidad (Nivel Secundaria). Elaboración Propia.</i>	193
<i>Tabla 5.16 Calificaciones y promedio de la evaluación obtenida mediante el Test de Usabilidad (Nivel Bachillerato). Elaboración Propia.</i>	194
<i>Tabla 5.17 Calificaciones y promedio de la evaluación obtenida mediante el Test de Usabilidad (Nivel Universidad). Elaboración Propia.</i>	195
<i>Tabla 5.18 Análisis comparativo de promedios generales por aspectos evaluados y sus reactivos. Elaboración propia.</i>	197

C

APÍTULO I: INTRODUCCIÓN

La tecnología moderna se ha convertido en un fenómeno total para la civilización, la fuerza determinante de un nuevo orden social en el que la eficiencia no es una opción sino una necesidad impuesta a toda actividad humana.

- Jacques Ellul

1.1 PLANTEAMIENTO DEL PROBLEMA

En México, existen variados espacios de cultura y de aprendizaje, enfocados a diferentes disciplinas en diferentes áreas del conocimiento; “espacios de cultura donde muchas veces se ha realizado una verdadera revolución didáctica mucho más potente que la llevada a cabo dentro de las aulas.” (Fernández, 2003)

En el presente trabajo nos enfocaremos, sobre la problemática que representa el que los objetivos de los responsables de la educación y la cultura muchas veces no vayan vinculados a un fin común dentro de la formación integral de los individuos.

Para confirmar esto, basta con consultar la encuesta estadística sobre el Consumo Cultural de México, que realiza INEGI, donde: “En el conjunto de los sitios y eventos culturales sobre los que se recabó información se incluyen espacios como: teatro, cine, sitios históricos y religiosos, sitios arqueológicos, parques naturales y áreas protegidas, museos, casas de cultura, bibliotecas, hemerotecas o archivos históricos, entre otros sitios o eventos culturales; los cuales fueron seleccionados a partir de la identificación de las necesidades de información no disponible con la desagregación requerida, para su posterior incorporación a la CSCM.” (INEGI, Encuesta Nacional de Consumo Cultural en México , 2012)

Dicha información destaca actividades como las visitas realizadas a algún sitio o evento cultural, al menos durante una ocasión dentro del periodo de un año, en un rango de “6 años y más”, así como el número total de asistencias y el gasto realizado (INEGI, 2012).

Los datos relevantes son que entre la población que oscila entre los 12 y 29 años, solo visitan espacios culturales por lo menos 1 vez al año, y el interés de la población en el aprendizaje de habilidades en áreas afines a la cultura; “a través de su participación en cursos y talleres relativos a temas como cine o video, radio o televisión, música, teatro, danza, pintura, escultura, fotografía, canto, dibujo, manualidades y artesanías, actuación, diseño, o literatura”, es representado solo por el 12%, es decir, uno de cada diez. Esto puede deberse a diversos factores tales como, la desinformación acerca de los inmuebles que brindan espacios culturales, y así mismo, las actividades, cursos y talleres que ahí se imparten.

En el presente siglo XXI la percepción de la realidad es un concepto híbrido que relaciona entornos materiales y virtuales generando un nuevo paradigma en la presentación y adquisición del conocimiento. Entre las tecnologías digitales, que actualmente son las más influyentes en los procesos de comunicación y relación con nuestro entorno, se encuentra la Realidad Virtual (RV), que proporcionaría una visión mucho más atractiva al proceso de aprendizaje, haciéndola más amigable e interactiva al usuario (Barbero, 2000 citado en Millán, 2005).

Se propone el desarrollo de una exhibición tridimensional de carácter interactivo implementado a un sitio público y de interés nacional, esto con el fin de conocer las

instalaciones de dicho espacio y con esto promover las actividades que se realizan ahí. La propuesta se estará llevando a cabo por medio de la Inteligencia Artificial, particularmente hablando, mediante la ayuda de agentes inteligentes.

1.2 HIPÓTESIS

La utilización de tecnologías de Realidad Virtual aplicada mediante un algoritmo orientado a Agentes Inteligentes, permitirá el desarrollo de un entorno virtual que hará posible interactuar más de cerca con espacios culturales, fundamentales en la formación integral del individuo.

1.3 JUSTIFICACIÓN

En nuestro país existen muchos y variados inmuebles públicos enfocados a la cultura, los cuales no tienen la difusión masiva que deberían tener, por tal motivo resulta importante crear nuevas alternativas donde personas desde cualquier parte del mundo, tengan la oportunidad de conocer e interactuar en estos sitios.

En la actualidad, muchos aspectos de la vida cotidiana se han visto inmersos en el desarrollo de nuevas tecnologías que permiten visualizar e interactuar con la información de manera más atractiva e interesante. Al hablar de Realidad Virtual (RA), se contempla interactuar con una potente tecnología que lleva, relativamente, poco tiempo en desarrollo, sin embargo, son muchos los ámbitos donde ha dejado ver sus grandes características, entre estos destaca la creación de videojuegos y mundos virtuales.

La creación de un entorno virtual engloba muchas variantes, y en el presente trabajo, no solo se contemplara las herramientas software que se utilizaran para el desarrollo, sino también, el público al que va dirigido, el impacto social que se pretende generar en los diferentes ámbitos sociales, y la metodología empleada para crear la arquitectura de un proyecto de esta índole.

Para tal propósito, se desarrollara, para el mejor funcionamiento del sistema, la utilización de un Agente Inteligente, el cual se encargara de la toma de decisiones, basándose en un conocimiento adquirido por medio de la interacción con su entorno. En este caso, se pueden considerar dos tipos de agentes inteligentes:

- Agente reactivo
- Agente de interfaz inteligente.

El sistema va dirigido para el público en general, ya sea que solo tenga curiosidad de conocer las instalaciones e información acerca de los seminarios, cursos, certificaciones que se imparten dentro del inmueble antes de hacer una visita presencial o, simplemente no disponga de los recursos monetarios, o de tiempo para conocer el inmueble en una fecha inmediata, también a un público más específico interesado en RV, Videojuegos y, en general al arte implementado por medios electrónicos y nuevas tecnologías.

Todo esto se tiene contemplado con el propósito de fomentar un interés más amplio hacia las actividades que se imparten dentro los espacios culturales en la Ciudad de México, donde se concentra gran parte de la población y resulta un sitio de interés para visitar por habitantes de otros estados, incluso, de otros países.

Es por ello, que se eligió un inmueble cultural de gran relevancia ubicado en la Ciudad de México, el cual, está dedicado a la preservación del arte y la cultura. Se trata del Centro Nacional de las Artes (Cenart).

Cenart “es una institución dedicada a la difusión, investigación, formación, impulso, debate y enseñanza de arte, la cultura y la interdisciplina”. (Artes, 2017). Cuenta con las

instalaciones recreativas y académicas necesarias para disfrutar de una gran y variada programación artística, así mismo, para desarrollar una vida académicamente de calidad.

Cuenta con un programa único en su tipo, que abarca desde cursos y talleres para niños hasta estudios de posgrado en educación artística. Además, concentra cuatro escuelas de educación profesional, las cuales pertenecen al Instituto Nacional de Bellas Artes (INBA), con las disciplinas de: danza, teatro, música o artes plásticas, junto con un Centro de Capacitación Cinematográfica. Por el lado de capacitación profesional, Cenart cuenta también con cuatro centros nacionales de investigación dedicados a la:

- Danza
- Artes plásticas
- Teatro
- Música
- Multimedia

En lo que al presente trabajo respecta, se ha puesto especial atención en el Centro Multimedia, el cual realiza proyectos de enseñanza y experimentación en la rama de las artes electrónicas y las nuevas tecnologías aplicadas en el arte. Actividades, que como se verá más adelante, no han logrado generar gran interés en la población, hablando a nivel nacional y a nivel local. (Artes, 2017)

En este sentido, se busca cubrir la divulgación y promoción, por medio de tecnologías electrónicas, no solo de los inmuebles que conforman el Cenart, sino también de las actividades que se ofrecen en cada uno de sus centros de investigación y enseñanza. Cabe mencionar, que se busca que los usuarios interesados en estos temas, conozcan el inmueble y su arquitectura, ya que el Cenart, así como sus centros y escuelas, compone una de las obras de arquitectura contemporánea más sobresalientes de México, esto se debe a que conjunta edificios de destacados creadores como Ricardo Legorreta, Teodoro González de León, Enrique Nortén, Luis Vicente Flores, Javier Calleja, Alfonso López Baz y Javier Sordo Madaleno. (Artes, 2017)

La falta de difusión de estos espacios puede deberse a muchos factores, sin embargo, la divulgación por medios electrónicos de cualquier producto, actividad, taller, curso, etc se ha vuelto una buena estrategia, ya que en los últimos años se ha visto un gran incremento en la utilización de internet y tecnologías electrónicas en la población de México, principalmente en la ciudad, como se verá a continuación.

Datos de la más reciente Encuesta Nacional de Consumo Cultural de México (ENCCUM) realizada por el Instituto Nacional de Estadística y Geografía (INEGI) en colaboración con el Consejo Nacional para la Cultura y las Artes (CONACULTA), hablan de un bajo consumo en las actividades de artes plásticas y visuales, artesanías y danza. (INEGI, Encuesta Nacional de Consumo Cultural en México, 2012)

Específicamente, en el sector en el que se interesa contribuir es en la difusión de cursos, talleres y/o certificaciones que se ofrecen en los espacios culturales. Ya que, según

la ENCCUM¹ por INEGI, la formación en actividades culturales en temas como: cine y/o video, radio y/o televisión, música, teatro, danza, fotografía, escultura, canto, pintura, dibujo, artesanías y/o manualidades, actuación, diseño y literatura; no representa gran interés dentro de la población, sin embargo, se muestra que el 12% de la población encuestada, ha asistido al menos un ocasión durante el año a estos cursos o talleres. (INEGI, Encuesta Nacional de Consumo Cultural en México , 2012)



Fig. 1.1 Asistencia a Cursos o Talleres Culturales (ENCCUM2012). Elaboración propia.

En la categoría que hace referencia a la edad de los encuestados, resalta el hecho del porcentaje relativamente equilibrado entre los grupos de 12-17 años con el 25% y de 18-29 años con el 24%, siguiendo de cerca el grupo de 30-49 años con 21%, mientras tanto la población de 6-11 y 50 o más años son de los que menos asistieron con un porcentaje de 19% y 10%, respectivamente. (INEGI, Encuesta Nacional de Consumo Cultural en México , 2012)

En la Ciudad de México (CDMX), la Secretaria de Cultura, es la encargada de diseñar programas y acciones de investigación, formación, difusión, promoción, y preservación del arte y la cultura. “Tiene como misión garantizar a la ciudadanía el ejercicio de sus derechos culturales, con base a la ejecución de acciones de concertación y participación activa de las comunidades”. (Cultura, 2015)

Por ello, se realizó la encuesta de consumo cultural en la CDMX, con el fin de obtener información que pueda mostrar de una manera más fidedigna los niveles de consumo cultural por parte de la población en esta ciudad, específicamente. Se realizaron un total de 120 entrevistas por las 16 delegaciones que conforman la CDMX. Se distribuyeron 1920 cuestionarios por las 16 delegaciones, la técnica empleada para la recolección de los datos fue por entrevistas personales. Esta encuesta nos muestra información muy importante no solo del consumo cultural, sino también, de varios factores que son muy

¹ La información que respecta a la cifras porcentuales y en general los datos estadísticos son obtenidos por la Encuesta Nacional de Consumo cultural de México 2012 realizada por INEGI, las ilustraciones son adaptaciones del mismo trabajo.

importantes a considerar para crear un verdadero interés hacia el consumo de nuestras actividades culturales.

En la siguiente figura se muestra la gráfica que representa la asistencia a espectáculos y/o recintos culturales según la encuesta de consumo cultural de la CDMX², en donde se enfoca la atención en lo que respecta a los programas que ofrecen la apreciación o la enseñanza de actividades culturales, en este sentido, se puede resaltar los bajos índices de asistencia, de manera focalizada, es decir, solo en la CDMX.



Fig. 1.2 Asistencia a Recintos Culturales y Espectáculos. Elaboración propia.

En la siguiente tabla, se muestra el interés que hay por parte de la población en tomar cursos o talleres culturales, incluso, estudiar alguna de manera profesional, nuevamente, se tomaron en cuenta las actividades más relacionadas con la cultura, según las misma encuesta.

	Estudio	Actualmente Estudia	SI le interesa	NO le interesa
Danza	3.8	1.8	32.9	57.0
Música	4.4	2.0	49.4	40.4
Teatro	2.5	0.6	33.6	59.1
Bibliotecología	0.3	0.2	10.6	83.1
Literatura y letras	0.6	0.5	31.7	62.2
Artes Visuales	0.9	0.3	27.7	65.8
Gestión Cultural	0.2	0.1	22.2	71.1

² Las gráficas presentadas son adaptaciones basadas en la información mostrada en la Encuesta de Consumo Cultural de la Ciudad de México 2015, realizada por la Secretaria de Cultura.

Historia o Historia del Arte	1.1	10.1	62.3	5.4
------------------------------	-----	------	------	-----

Tabla 1.1 Interés en Actividades Culturales más representativas. Elaboración propia.

Para reforzar la iniciativa de realizar una aplicación para medios electrónicos, se contempla la siguiente gráfica donde se muestra los principales medios de comunicación por los cuales la población se entera de actividades culturales. Sobresale el hecho de que gran parte de la población utiliza internet, por lo cual queda implícito que este mismo porcentaje de población, deben tener acceso a una computadora o algún dispositivo electrónico.

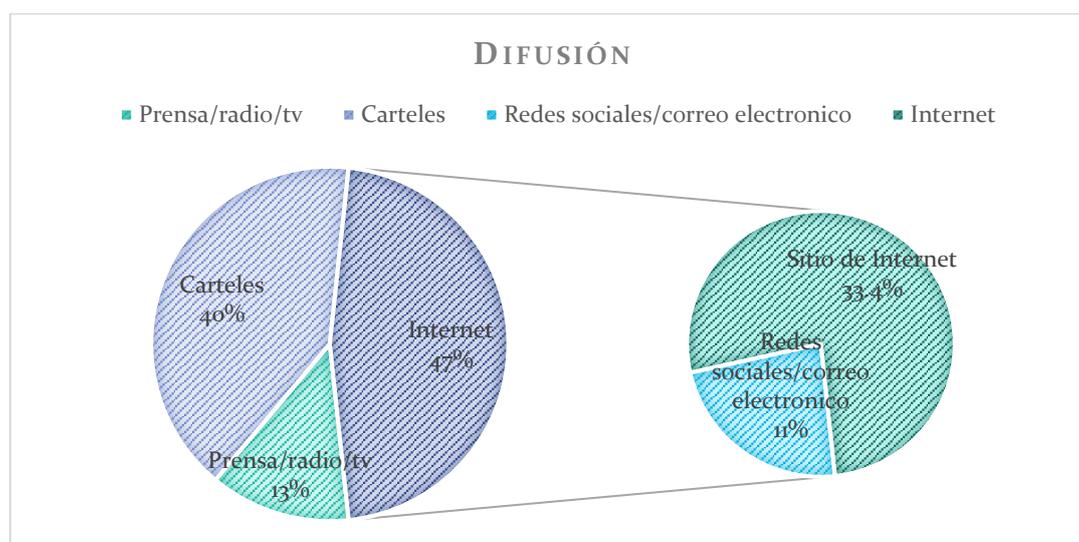


Fig. 1.3 Medios de comunicación de difusión de Actividades Culturales. Elaboración propia.

Es importante mencionar que una de las más importantes iniciativas del presente proyecto, es proporcionar el fácil acceso a una aplicación de realidad virtual que, como se explicara a lo largo del documento, es, dependiendo de sus características, de muy caro acceso.

Una aplicación de RV con las características que se plantean en el presente proyecto, suele generar una mayor aceptación en el público. Pues el principal objetivo de la RV es desarrollar un espacio virtual atractivo para el público de todas las edades, independientemente de la información que este ofrezcas; es por esto que la interfaz y la presentación de cualquier sistema, principalmente de este ámbito, es fundamental. (Luna, 2012)

Esta tipo de tecnologías permiten al usuario desplazarse en un entorno “ficticio”, sin moverse físicamente, independientemente de si se trata de una experiencia realista, según los dispositivos que se utilicen. Los sentidos del usuario deben ser previstos y estimulados por la aplicación para que el entorno que se percibe sea lo más parecido a la realidad. Sin embargo, esto no es lo más importante en un sistema de RV, pues aunque la noción de presencia se refiere a la sensación que experimenta el usuario de estar “dentro del entorno”, tiene mayor relevancia en el realismo que ofrece la aplicación por si misma

mediante las imágenes generadas, este proceso se explicara más a fondo en el *capítulo III*.

Otros datos importantes a considerar, es la apatía o simple desinformación de la población por este tipo de actividades, ya que la encuesta también revela un porcentaje de más de la mitad de la población encuestada, que no sabe acerca de estos talleres o cursos culturales.



Fig. 1.4 Gráfica del conocimiento de si se imparten o se han impartido actividades culturales. Elaboración propia.

En cuanto a la población que tiene alguna idea acerca de la impartición de estas actividades culturales, talleres o cursos, se obtuvieron los siguientes resultados donde la mayor parte de la población no sabe, o no le interesa.

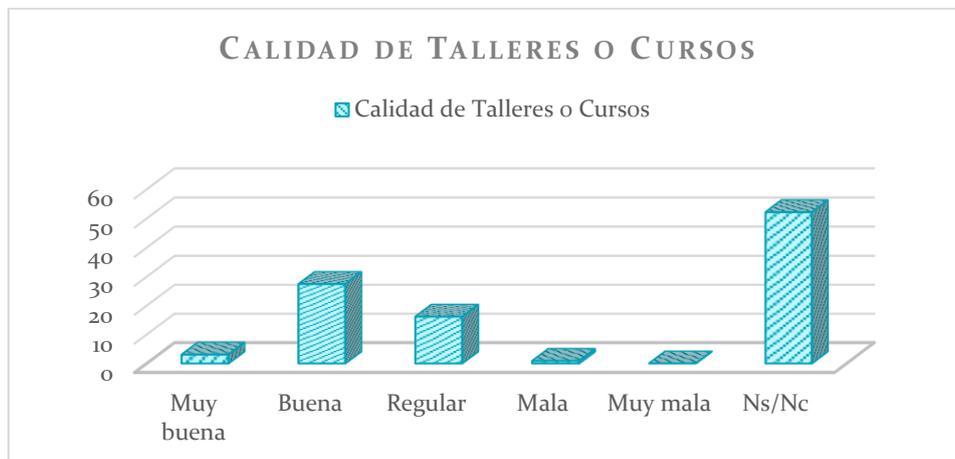


Fig. 1.5 Opinión de los encuestados acerca de la calidad de talleres y/o cursos. Elaboración propia.

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

Diseño y desarrollo de un sistema en Realidad Virtual, mediante la construcción de un entorno virtual asistido por un agente inteligente reactivo que facilitará la interacción con el usuario y sus necesidades en la toma de decisiones; dicha aplicación ayudarán a la divulgación del conocimiento de forma interactiva.

1.4.2 OBJETIVOS ESPECÍFICOS

- Investigar sobre software y herramientas de modelado 3D.
- Bosquejo y realización a partir de la extracción del inmueble.
- Documentación referente a: dimensiones dentro del recinto, espacios, cuartos, medidas para próximas exposiciones y/o visitas.
- Recabar información acerca de las actividades a realizarse en el inmueble como: talleres impartidos, trabajos importantes y expositores representativos
- Obtención de un recorrido en imágenes planas a través del espacio de difusión artística público.
- Recolección de dimensiones específicas, colores, texturas, materiales, etc.
- Desarrollo del algoritmo que permitirá la interacción del usuario con el sistema.
- Diseño de los Agentes Inteligente implementados en base al algoritmo definido en el paso anterior.
- Programación del paseo virtual con los componentes antes descritos.

1.5 LIMITACIONES Y ALCANCES

Dentro del presente proyecto se contemplan diversas ramas del conocimiento como:

- Arte
- Softwares de diseño
- Multimedia
- Fotografía

De las cuales, se tiene un conocimiento básico, sin embargo, se cuenta con las bases para poder desarrollar un proyecto de la magnitud contemplada.

El área de programación y matemáticas aplicadas es fundamental para el desarrollo del presente trabajo, contemplando herramientas novedosas con las que se pueda simplificar los métodos de desarrollo.

Pese a la experiencia que se tiene en los principales puntos de desarrollo, debido a la falta de práctica en proyectos de diferentes ámbitos involucrados, han surgido algunas dificultades significativas. Una de las principales es respecto a la estructuración del código, pues se ha tenido que diseñar y cambiar algunas clases para adaptarlas a los objetivos del proyecto, ya que la generación del algoritmo de un agente inteligente es primordial y tiene que adaptarse al funcionamiento del sistema completo.

La planificación previa ayudara a diseñar el comportamiento del sistema, sin embargo en el desarrollo se podrían presentar algunos inconvenientes que serán modificados de ser requerido así. Otra dificultad que podría presentarse es al momento de diseñar la interfaz, pues por un lado se utiliza un software para el desarrollo interactivo y otro para el sistema general. Se contempla también algunos problemas de compatibilidad en el momento de exportar e importar el modelo 3D para programar su comportamiento interactivo, ya que el programa de modelado 3D solo será utilizado para crear el modelo sin programar ningún evento para los objetos generados.

La limitación que se menciona en el desarrollo del modelo interactivo, deriva en la complejidad que representa simular los movimientos del usuario dentro de las instalaciones y el poder observar cierta información estática acerca de algunos sitios done se posicione el mouse

Generalmente se utilizan plugins para cubrir los efectos más utilizados en ambientes virtuales, sin embargo cuando se utilizan modelos externos la cantidad de plugins para aplicar son menores, así que se tiene que programar su comportamiento de manera rustica.

1.6 DESARROLLO DE LA DOCUMENTACIÓN

CAPÍTULO I. INTRODUCCIÓN. Este apartado se describen la problemática del proyecto, la justificación del porque se quiere desarrollar sustentado con información más actual y relevante del tema, el objetivo general y sus objetivos específicos.

CAPÍTULO II. ANTECEDENTES Y ESTADO DEL ARTE. El objetivo de este capítulo es lograr una pequeña remembranza acerca de todos los acontecimientos históricos que contribuyeron al nacimiento de esta nueva área de la tecnología: Realidad virtual, y los trabajos/proyectos relacionados que actualmente se han trabajado, prestando especial atención en los trabajos realizados académicamente que abordan una o más áreas de investigación vistas en este proyecto.

CAPÍTULO III. MARCO TEÓRICO (HERRAMIENTAS A UTILIZARSE). En el presente capítulo se introducen las principales áreas de investigación referentes al proyecto desarrollado (realidad virtual, agentes inteligentes, modelado 3D, programación), además se valoraran diferentes herramientas implementadas para el desarrollo de proyectos de esta índole.

CAPÍTULO IV. METODOLOGÍA Y ARQUITECTURA DE DESARROLLO. Se evaluara la metodología más adecuada para el desarrollo de interfaces interactivas utilizando la realidad virtual como principal recurso. También, se incluye el diseño y la implementación que aborda el desarrollo de la metodología descrita, en esta etapa se desarrollara todas las etapas para obtención del producto final.

CAPÍTULO V. PRUEBAS Y RESULTADOS. Se someterá a un método de evaluación, la interacción con el entorno creado.

CAPÍTULO VI. CONCLUSIONES Y FUTURAS LÍNEAS DE INVESTIGACIÓN. Descripción de los resultados obtenidos y la información de las pruebas realizadas. Así mismo, se relatan observaciones acerca del trabajo final y las posibles líneas de investigación para trabajos futuros.

REFERENCIAS BIBLIOGRÁFICAS

C

APÍTULO II: ANTECEDENTES Y ESTADO DEL ARTE

El avance de la tecnología se basa en hacerla encajar
de manera que en realidad ni siquiera la notas,
y así ésta es parte de la vida cotidiana.

- Bill Gates

2.1 ANTECEDENTES

Los seres humanos generamos conocimiento a partir de la interacción con nuestro entorno, lo que percibimos con nuestros sentidos al explorar la realidad física. Una de las maneras de crear información más importante es mediante lo que descubrimos usando nuestra visión. (Jesus Alberto Flores Cruz, 2014)

El desarrollo del presente proyecto representa la oportunidad de adentrarnos a un nuevo paradigma de presentación de la información, por medio de la implementación de un inmueble interactivo para la divulgación del conocimiento de áreas de la cultura específicas en nuestro país, complementando una formación impartida por el sistema educativo a cargo. Reconociendo así que ambas variantes del conocimiento pueden integrarse de manera adecuada para el desarrollo humano.

Cabe mencionar lo importante que supone actualmente, el ser partícipe del ciberespacio, ya que de manera inherente, nos invita a concebirnos como sus constructores y no sólo sus usuarios. Generar un ambiente virtual requiere de colocar objetos, visualizar los espacios, imaginar los movimientos e interacciones de las personas en él, lo cual deriva en las principales ventajas de contar con un entorno virtual tridimensional.

Un entorno virtual es una aplicación para computadora o cualquier tipo de dispositivo electrónico visual, que permite al usuario navegar e interactuar en un entorno tridimensional utilizando una interfaz, la cual, funge como intermediario usuario – maquina. (María del C. Ramos, 2007).

El cual, no solo ofrece los gráficos interactivos en tercera dimensión, sino también, un medio que nos permite cierto grado de inmersión y de presencia dentro de determinado espacio creado de manera artificial.

La escena virtual e interactiva en el diseño de sistemas, nos permite interactuar de una manera más amigable con aspectos comunes, como lo son, una pre-visita virtual a ese lugar que buscamos conocer en persona, hasta hacer la lectura de un libro más interesante y amena. Estas herramientas son muy impactantes, despiertan el interés del público en general, ayuda a la divulgación de sitios culturales, así como la realización de actividades, en talleres y proyectos, siendo este uno de los objetivos principales del presente trabajo.

“La explosión de mundos virtuales que se ha producido en los últimos años es un fenómeno cuyo inicio se remonta décadas atrás con el nacimiento de conceptos como la realidad virtual, la telepresencia o los lenguajes de programación para representar espacios tridimensionales.” (López, 2008).

En este sentido, se han trabajado considerables avances en los recientes años, dentro del enfoque de hacer mucho más amigable y accesible la visita a inmuebles de interés general.

En 2004, Google compra Earthview, un programa desarrollado en 2001, para crear el Google Earth, una representación del mundo que combina la potencia de las búsquedas de Google con imágenes de satélites, mapas, terrenos y edificios 3D. (Carmen Ortigueira España, 2009)

En 2005, Se anuncia el lanzamiento de Wii de la empresa Nintendo, (con el nombre en clave de "Revolution") la videoconsola que nace con la idea de conseguir una interacción antes nunca experimentada en una videoconsola entre el jugador y el videojuego. Así como "Virtual Boy" fue un fracaso, Wii a día de hoy ha sido un éxito rotundo. (Carmen Ortigueira España, 2009)

En 2007, Se presenta "Los simuladores, estrategia formativa en ambientes virtuales de aprendizaje", una propuesta innovadora que muestra la creación de mundos simulados que representan los cambios históricos que imponen las nuevas tecnologías de información. Es así como el llamado "e-learning" impacta la manera como se desarrolla el aprendizaje de tal forma que logra una gran influencia y un posicionamiento en el que la simulación juega un papel primordial. Así mismo, se aborda una nueva manera de representación de la información, utilizando lo más avanzado en tecnología. (Irma Lucía Franco Sepúlveda, 2007)

En 2008, se presenta Cyber Walk, una plataforma en la que podemos movernos sin problemas en los mundos virtuales y, lo novedoso, es que sigue nuestros pasos, avancemos en la dirección que avancemos. El dispositivo genera sensaciones táctiles vinculadas a imágenes virtuales mediante el ultrasonido. (Carmen Ortigueira España, 2009)

En 2012, se analiza la Realidad Virtual implementada en el campo de Psicología para el tratamiento de diferentes trastornos; "...Estas tecnologías ayudan a los pacientes a afrontar sus problemas a la vez que lo hacen en un contexto significativo seguro y controlado." Donde, además, se presentan las ventajas y desventajas que podría conllevar el uso de RV. (S. Quero, 2012)

En 2014, la cadena hotelera Marriot realizó una acción promocional en varias ciudades de Estados Unidos. Ubicó un par de "cabins de teletransportación" como las que hemos visto en series y películas, dotadas de la tecnología necesaria para "viajar" a destinos donde tiene hoteles. Así, a través de gafas de Realidad Virtual, películas en 360 grados, renders en 3D, los visitantes tenían la posibilidad de estar en una playa de Hawaii o en la terraza de la planta 42 de un edificio de Londres. (Rodríguez M. V., 2015)

En 2015, la oficina de turismo de British Columbia, Canadá, "The Wild Within VR Experience" de Destination BC, ofrece una experiencia de realidad virtual a sus visitantes en las ferias, presentaciones, congresos y exposiciones. A través de un vídeo, los sumerge en un juego en el que cada uno de ellos puede ir eligiendo su propio camino dentro del Parque Natural del Oso, en la costa del Pacífico, sorteando los accidentes naturales y los posibles peligros, pero acercándose a los atractivos de la zona. (Rodríguez M. V., 2015)

2.2 TECNOLOGÍAS PRECURSORAS EN REALIDAD VIRTUAL

La tecnología de la Realidad Virtual (RV), desde sus orígenes en el siglo XX, ha sido un determinante en la historicidad del desarrollo de Sistemas Inteligentes. Se habla tanto de las limitaciones visuales que las tarjetas gráficas imponían hace tan sólo quince años hasta su desarrollo hoy en día, momento en el que se ha pasado de simular el entorno en una pantalla de ordenador a formar parte de él. Se presentan además, algunas de las aplicaciones más conocidas de esta tecnología, convertida en un medio más de comunicación, gracias a los enormes esfuerzos de investigación que han conducido a un gran desarrollo de las computadoras (Martínez F. J., 2011).

El término de RV se comenzó a utilizar a partir de los años 60 y 70 del siglo XX. La RV, no es un término que se haya creado de la noche a la mañana, tuvieron que intervenir varios y novedosos proyectos para que en conjunto crearan esta nueva área de la tecnología. Aunque los inventores de esas tecnologías ni siquiera podían imaginarlo en su momento cada uno de los pasos que dieron permitió años después concebir que podían crearse espacios tridimensionales donde interaccionar con otros usuarios. Sin saberlo fueron los pioneros de un campo que estaba todavía por descubrir. (López C. , 2008)

2.2.1 SENSORAMA

Este invento, que parece de una tecnología del siglo XXI, tiene más de 50 años desde su aparición. Fue allá en 1957 cuando la máquina Sensorama, pionera en este ámbito y diseñada por Morton Heilig, reproducía películas de cine en 3D. Se trataba de una máquina con funcionamiento mecánico que proyectaba imágenes estereoscópicas (en tres dimensiones) sobre un aparato similar a las máquinas de salón recreativo. (Moriel, 2015)

La ilusión de la tridimensionalidad se realizaba grabando las imágenes al mismo tiempo con tres cámaras de 35 mm. Las utilizadas en el cine comercial y proyectándolas en el equipo. Al mismo tiempo la máquina reproducía viento, cierto movimiento e incluso reproducía aromas para dar una experiencia más realista. La primera película grabada para el sistema reproducía un paseo en bicicleta por Brooklyn (NY).

Lamentablemente Heilig no pudo conseguir financiación para rodar nuevas (y carísimas) películas para reproducir en su máquina, de forma que Sensorama pasó a ser una curiosidad más en la historia de la técnica y de la realidad virtual. Aun así se trata de una máquina que causó sensación en su momento por lo realista que eran sus proyecciones. Y sigue haciéndolo en algunas máquinas que siguen en funcionamiento décadas después. (López C. , 2008)

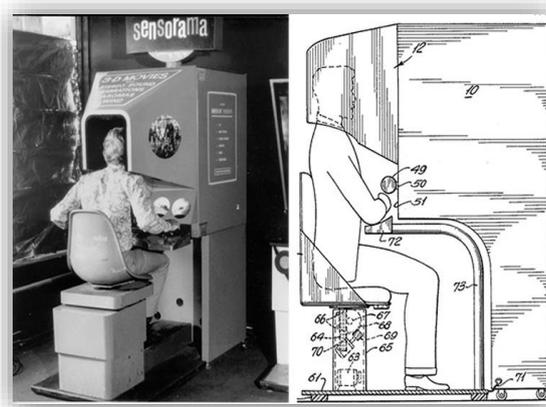


Fig. 2.1 Maquina Sensorama de Heilig (s.f.) (Recuperado de: Blog stampaprint.fr, publicación 2016)

2.2.2 ESPADA DE DAMOCLES

El aparato desarrollado en 1968 por Ivan Sutherland, fue uno de los primeros en esta categoría. Más tarde daría nombre a una categoría, los llamados HUD (Head-up display). Son aparatos que montados en la cabeza como unas gafas permiten ver información como si fuera una pantalla. Con el tiempo fueron evolucionando e incluso se desarrolló su utilización para uso militar. Como uso militar tiene la capacidad de mostrar información al mismo tiempo que se tiene visión del mundo real, así por ejemplo, un piloto de caza podría tener información de los aparatos del avión sin tener que dirigir la mirada hacia el tablero de instrumentos.

La máquina sólo reproducía polígonos y la capacidad gráfica era muy limitada. La técnica que había detrás del invento era relativamente sencilla, se situaba una cámara en la habitación que se quería reproducir y la cámara imitaba los movimientos de la cabeza que el operador realizaba. Al mismo tiempo el sistema mostraba una reproducción poligonal del espacio bastante pobre. (López C. , 2008)



Fig. 2.2 Maquina "Espada de Damocles" (s.f.) (Recuperado de: geekster.be, 2016)

2.2.3 ASPEN MOVIE MAP

En el campo de la tecnología tridimensional, un proyecto precursor fue el Aspen Movie Map, creado en 1977 por el MIT (Massachusetts Institute of Technology). Se trataba del primer documental interactivo, consistía en un estabilizador giroscópico con cámaras de parada de 16 mm fue montado en la parte superior de un coche y una quinta rueda con un codificador que disparaba las cámaras cada 10 pies.

La filmación se realizó diariamente entre las 10 AM y las 2 PM para minimizar las discrepancias de iluminación. El coche condujo con cuidado por el centro de la calle para registrar los cortes. Además de las imágenes básicas de "viaje", se recolectaron experimentos de cámara panorámica, miles de imágenes fijas, audio y datos. El sistema de reproducción requería varios reproductores laserdisc, un ordenador y una pantalla táctil. Se usaron lentes de gran capacidad angular para filmar, y se hicieron algunos intentos de reproducción ortoscópica. La técnica utilizada es muy similar a la que años después ha permitido a Google realizar la "streetview", aplicación conocida como Google maps. (Naimark, 2006)



Fig. 2.3 Project Aspen Moviemap, M.I.T. (1981) (Recuperada de: naimark.net)

2.2.4 VRML (VIRTUAL REALITY MODELING LANGUAGE)

En 1995, surge la primera versión del VRML (Virtual Reality Modeling Language. "Lenguaje para Modelado de Realidad Virtual"), la cual fue la base para crear el que se considera uno de los primeros ambientes virtuales, Cybertown. Este proyecto consistía en un chat tridimensional. Sin embargo, no se encontró el uso masivo que se esperaba de tan innovador proyecto.

El VRML ha sido utilizado extensivamente en el ámbito académico y la investigación, tal es así que ha sido el origen de un nuevo formato, el cual fue llamado: X3D. Además, los estándares para representar información en tres dimensiones que fueron formulados, siguen utilizándose actualmente, especialmente en aplicaciones médicas, militares y educativas. (López C. , 2008)

Su mecanismo de trabajo, logra evitar la transmisión de archivos gráficos directamente en los problemas de red, envía la tarea complicada a la máquina local, a fin de reducir la carga de la red, y lograr la ejecución, del tridimensional interactivo posible en Internet. El usuario necesita un plug-in para observar la escena VRML en el navegador. Entre los plug-in para un navegador VRML, se pueden encontrar los primeros en desarrollar una plataforma 3D diseñada para Internet, tales como: Blaxxun Contact, CosmoPlayer, entre otros.

Los objetos en la escena VRML pueden llamarse nodos, los cuales tienen diferentes propiedades, esto incluye la geometría, el material, la textura, la luz, la vista y así sucesivamente. Generalmente, se describe los objetos y sus atributos a partir de la ejecución en el escenario. El nodo constituye la escena virtual según la regla determinada. El nodo en el gráfico de la escena se utilizan para demostrar los objetos virtuales desde el sentido de la visión y el oído, que siguen una jerarquía y se combinan, esto contribuye a el rendimiento del efecto dentro del espacio del escenario. (Yang Kuang, 2015)

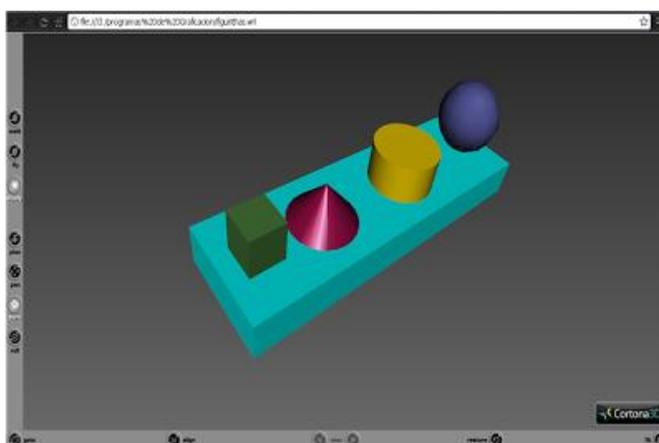


Fig. 2.4 Figuras Geométricas creadas con el lenguaje VRM (Ortega, 2011)

2.3 REALIDAD VIRTUAL: DESARROLLO HISTÓRICO

El ámbito de la RV ha ido creciendo en los últimos años de manera significativa, y con ello, la aparición de herramientas software para el desarrollo de la misma. Muchas veces, al hablar de RV, en general se relacionan los aspectos más superficiales de esta, herramientas como los cascos de visualización estereoscópica o los guantes de datos son los que prevalecen en la memoria colectiva al hacer referencia a este tema. Las definiciones sobre lo que es la RV, son numerosas, ya que se habla de un nuevo paradigma, no solo para la programación de software sino, también, para la representación e interacción de datos.

Entre muchas definiciones importantes, se encuentra la siguiente, que a propio punto de vista es una de las más acertadas: “Realidad Virtual es la simulación de un ambiente real o imaginario que puede ser experimentado en tres dimensiones, proporcionando una experiencia interactiva en tiempo real con video, sonido e incluso retroalimentación táctil” (Olguin Carbajal, Rivera Zárate, & Hernández Montañez, 2006)

Actualmente, es indiscutible que con el apoyo de diversos ejemplos de RV, se evidencia la indiscutible ayuda prestada en la mejora de la calidad de vida en muy diversos campos de la actividad diaria, la educación, el ocio, la ciencia, etc. Realidad Virtual, además, potencia la imaginación y libera la mente.

El mundo empresarial ha comenzado a pensar en ellos como una plataforma para relacionarse con sus potenciales clientes. Esta relación es una más del llamado Social Media Marketing, con el cual las empresas intentan acercarse a las redes sociales, el último gran boom en Internet. (López C. , 2008)

La gran inclusión de mundos virtuales que se ha producido en los últimos años en diferentes ámbitos del conocimiento, es un fenómeno cuyo inicio, como ya se vio anteriormente, se remonta décadas atrás con el nacimiento de conceptos como la realidad virtual, la telepresencia o los lenguajes de programación para representar espacios tridimensionales.

Aunque la RV es una tecnología relativamente nueva, se han alcanzado grandes avances en el desarrollo de varios proyectos innovadores. Uno de lo más sobresalientes, es la creación de mundos virtuales, los cuales surgieron de maneras muy diversas, algunos aparecieron como una extensión de los chats, para mostrar una imagen de cada uno de los participantes y hacer el espacio más interactivo. Otros fueron extensiones de juegos que lanzaron una versión para jugar en línea. En general se suele considerar a los siguientes como los pioneros.

AÑO	ACONTECIMIENTO
1930	El Link Flight Trainer fue creado, un simulador de vuelo basado en un Sistema Neumático para simular los movimientos y entrenar a los pilotos
1965	Surge el concepto de Realidad Virtual, cuando Iván Sutherland (hoy miembro de Sun Microsystems Laboratories) publicó un artículo titulado "The Ultimate Display", en el cual describía el concepto básico de la Realidad Virtual. El trabajo inicial del doctor Sutherland fue básico para investigaciones subsecuentes en este terreno.
1968	Ivan Sutherland y David Evans crean el primer generador de escenarios con imágenes tridimensionales, datos almacenados y aceleradores. En este año se funda también la sociedad Evans & Sutherland.
1971	Redifon Ltd en el Reino Unido comienza a fabricar simuladores de vuelo con displays gráficos. Henri Gouraud presenta su tesis de doctorado "Despliegue por computadora de Superficies Curvas".
1972	General Electric, bajo comisión de la Armada norteamericana, desarrolla el primer simulador computarizado de vuelo. Los simuladores de vuelo serán un importante renglón de desarrollo para la Realidad Virtual.
1976	P. J. Kilpatrick publica su tesis de doctorado "El uso de la Cinemática en un Sistema Interactivo Gráfico".
1979	A principios de los 80s la Realidad Virtual es reconocida como una tecnología viable. Jaron Lanier es uno de los primeros generadores de aparatos de interfaz sensorial, acuñó la expresión "Realidad Artificial", también colabora en el desarrollo de aparatos de interface VR, como guantes y visores.
1984	Mike Mc Greevy y Jim Humphries desarrollaron el sistema VIVED (Representación de un Ambiente Virtual, Virtual Visual Environment Display) para los futuros astronautas en la NASA.
1986	Habitat es el que se considera primer mundo virtual como tal. Fue desarrollado por Lucasfilm para el ordenador estrella de la época, el Commodore 64. El mundo no era en tres dimensiones pero si contaba ya con representaciones de sus usuarios, avatares. El software fue vendido y revendido entre compañías hasta que finalmente fue comprado por Compuserve que lanzó el juego con el nombre de Worldsaway.
1991	La compañía W. Industries desarrollaron los Virtuality, e instalados en los salones recreativos de EE.UU. El equipo incluía unos cascos y gafas de visionado. En este mismo periodo aparecerán numerosos modelos emulando cabinas de vuelo o conducción. En este mismo año, sale el primer programa destinado a los usuarios para la confección de ambientes virtuales 3D. En el mismo año se estrena un programa infantil en TVE llamado "El rescate del talismán" en el cual los concursantes debían guiar a un compañero con los ojos vendados por unos escenarios virtuales

1992	SUN hace la primera demostración de su Portal Visual, el ambiente VR de mayor resolución hasta el momento. En mayo de este mismo año, sale el primer juego cuya perspectiva de los gráficos era en primera persona Wolfenstein.
1992	Electronic Visualization Lab (EVL) de la universidad de Illinois, Chicago, ideó en 1992, el concepto de una habitación con gráficos proyectados desde detrás de las paredes y suelo, apareciendo CAVE (Cave Automatic Virtual Environment).
1993	Silicon Graphics (SGI) anunció un motor de Realidad Virtual. Se creó CitySpace, fue uno de los primeros mundos virtuales creados por el usuario, es decir, en él se permitía diseñar objetos e incluso avatares y edificios. Tan sólo estuvo operativo tres años, hasta el año 1996.
1994	La Sociedad de Realidad Virtual fue fundada. IBM y Virtuality anunciaron el sistema V-Space. Virtuality anunció su sistema serie 2000. División hizo una demostración de un sistema integrado de Realidad Virtual multiplataforma en IITSEC, Orlando. Antena 3 es la primera cadena de televisión española en introducir espacios virtuales en sus programas.
1995	Nintendo saca la primera consola de realidad virtual llamada "Virtual Boy" cuyos gráficos eran en 3D en rojo y negro. Pero estas serían un fracaso comercial era demasiado grande y frágil, el continuado durante varios minutos podía producir dolor de cabeza.
1996	The Palace se trataba de un servicio de chat que comenzó a ofrecer la posibilidad de diseñarse un personaje que le representara en las salas de conversación. También se podía modificar la apariencia, la ropa e incluso poner complementos y algunos objetos. Y aunque no es conceptualmente un mundo virtual avanzó algunas de las posibilidades que estarían disponibles en los mundos virtuales que se crearon posteriormente.
1997	Se desarrolla para la US Army's STRICOM un dispositivo que permite caminar, correr y moverse en un reducido espacio en todas las direcciones posibilitando experimentar el movimiento real en un cabina.
2003	Se crea el famoso mundo virtual en 3D "Second Life" donde por medio de un programa pc, los usuarios o residentes, pueden moverse por él, relacionarse, modificar su entorno y participar en su economía.
2004	Google compra Earthview, un programa desarrollado en 2001, para crear el Google Earth, una representación del mundo que combina la potencia de las búsquedas de Google con imágenes de satélites, mapas, terrenos y edificios 3D.
2005³	Se anuncia el lanzamiento de Wii de la empresa Nintendo, (con el nombre en clave de "Revolution") la videoconsola que nace con la idea de conseguir una interacción antes nunca experimentada en una videoconsola entre el jugador y el videojuego. Así como "Virtual Boy" fue un fracaso, Wii a día de hoy ha sido un éxito rotundo

Tabla 2.1 Avance Histórico de la Realidad Virtual. Adaptación de Cardozo y Ortigueira (2004 y 2009).

³ Avances más representativos en el desarrollo de la Realidad Virtual y la escena 3D en diversos ámbitos del conocimiento. Elaboración propia, adaptado de (Cardozo, 2004), (Carmen Ortigueira España, 2009).

2.4 TRABAJOS RELACIONADOS EN LA EDUCACIÓN

2.4.1 DESARROLLO DE MUNDOS VIRTUALES EN EL ÁREA DE MÁS SOBRE CIENCIA DEL MUSEO VIRTUAL 3D DEL REHILETE

En el año 2003, se crea el proyecto llamado Museo Virtual 3D El Rehilete, que se basó fundamentalmente en la virtualización del Museo, donde participaron pasantes de la Licenciatura en Sistema, en un convenio con el Gobierno de Hidalgo, mediante el Centro de Investigación en Tecnologías de Información y Sistema.

El Museo del Rehilete es una institución educativa y cultura fundado por el gobierno del Estado de Hidalgo, el cual presta servicio educativo no escolarizado, interactivo e innovador, lo cual se tiene contemplado para incentivar la asistencia y aumentar el conocimiento científico, cultural, tecnológico y artístico.

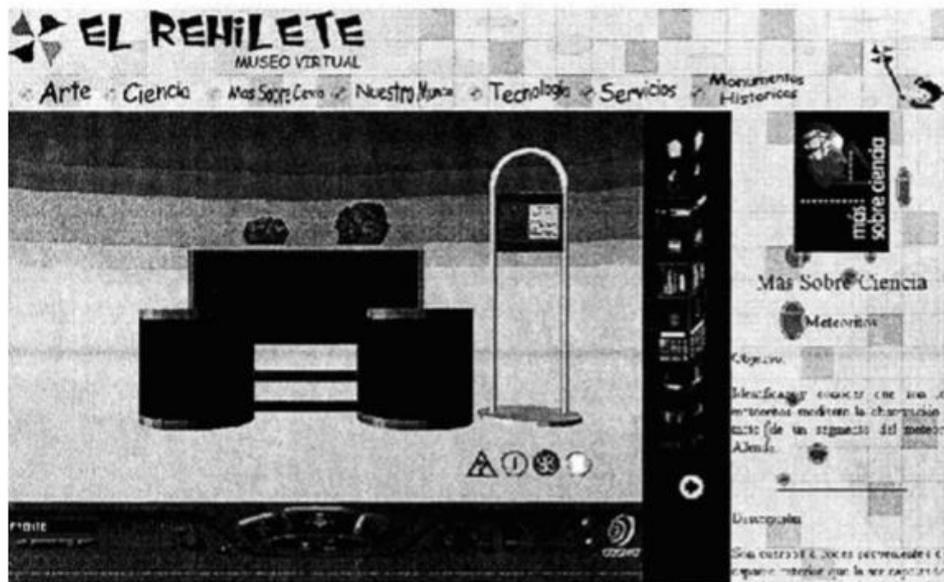


Fig. 2.5 Interfaz de la App Museo del Rehilete (Cruz Rodríguez, 2003)

Se desarrolló el Museo el rehilete ubicado en Pachuca, Hidalgo, haciendo uso de técnicas de modelado interactivo de Realidad Virtual para ser consultado mediante internet e internet2.

El proyecto fue desarrollado dentro de las instalaciones del CITIS. (Rodríguez M. Á., 2003), motivados principalmente para ofrecer una opción novedosa para conocer el museo en cuestión. Las exhibiciones desarrolladas contienen un grado de interactividad aceptable; es desarrollado en gran parte, mediante el lenguaje de programación **VRML** (*Ver Capítulo II y III*), así como JavaScript para definir comportamientos más complejos.

El funcionamiento del sistema, depende de una conexión a internet, un navegador que permita leer las funciones VRML, y para la presentación de las animaciones generadas mediante Macromedia Flash, será necesario FlashPlayer.

Para crear el menú principal que permitirá integrar el paseo virtual en la página web, se utilizó Macromedia Flash para el diseño mediante la programación con JavaScript. Para el correcto funcionamiento y presentación de los objetos 3D, se añadió “internet space builder”, el cual es una herramienta de Parallel Graphics que ayuda a texturizar con mayor facilidad el modelo 3D.

El texto presentado a lo largo del sistema interactivo, es mediante imágenes editadas en Photoshop, es decir, no se utilizaron archivos de texto, sino imágenes previamente echas para presentar texto fijo.

Para modelar los objetos y escenas 3D, se utilizó el programa 3D Studio Max, para simplificar los tiempos de creación, ya que mediante este software los modelos 3D, se diseñan de forma gráfica.

Como IDE, se utilizó VrmIpad, el cual ayuda a la programación en VRLM, pues mediante su estructura de programación a través de nodos, permite programar los objetos de forma más clara y organizada, para manejar los recursos implementados (texturas, sonidos, etc.).

Cabe señalar que para el desarrollo del sistema, no solo se utilizó un solo lenguaje de programación, pues debido a su naturaleza de aplicación web, se debió implementar también HTML, aunado a VRML y JavaScript, se logró montar un paseo virtual en su propia página web.

Tal como ya se mencionó, la estructura del sistema fue basada en la programación de nodos, para definir eventos y comportamientos de los objetos 3D. A continuación se muestra parte del código para declarar estos argumentos en el sistema.

```
PROTO BouncingBall {
  field SFFloat bounceHeight 1.0
  field SFTime bounceTime 1.0
  /*
  DEF Ball Transform {
    children [
      Shape {..}
    ]
  }
  DEF Clock TimeSensor {
    cycleInterval IS bounceTime
    startTime 0.0
    stopTime 0.0
    loop TRUE
  }
  DEF Bouncer Script f
  field SFFloat bounceHeight IS bounceHeight
  */
  ROUTE Clock.fraction_changed TO Bouncer.Set_fraction
  ROUTE Bouncer.valueChanged TO Ball.set_translation
}
```

Fig. 2.6 Programación de eventos en objetos 3D con VRML. (Cruz Rodríguez, 2003).

2.4.2 SISTEMA DE REHABILITACIÓN CARDIACA AUMENTADO POR REALIDAD VIRTUAL

En 2010, se desarrolla InTrainer, por Ortiz Salcedo Miguel A., un sistema con un Ambiente Virtual enfocado a la medicina, desarrollado mediante Realidad Virtual. La finalidad de dicho proyecto, es ayudar a los pacientes a realizar sus actividades de entrenamiento físico mediante la inmersión en el entorno virtual diseñado.

El sistema monitorea a los pacientes mediante la adquisición de información biomédica, siendo así capaz de reaccionar y actuar dependiendo de la información adquirida. Utiliza, también, tecnologías especializadas para lograr una experiencia capaz de involucrar al usuario haciéndolo crear sus metas a corto plazo, también creando un ambiente agradable con mensajes y la interacción requerida para los trabajos en conjunto.



Fig. 2.7 Representación de Usuario dentro del Entorno Virtual InTrainer (Ortiz S. Miguel A., 2010)

La información obtenida a través de los sensores utilizados, es procesada para modificar la experiencia del usuario de forma constante, adaptándose al estado y necesidades actuales del paciente. Aunado a esto, el sistema genera archivos de log (información que recopila datos del estado y funcionamiento en general, también registra información relacionada con el trabajo realizado de cada paciente en cada una de las sesiones en las que participe), lo cual, resulta útil para detectar errores o mal funcionamiento, pues cada incidencia está registrada dentro del código con mensajes de información específica de la naturaleza del error detectado. (Salcedo, 2010)

El desarrollo de la interfaz del sistema fue elaborada con lenguaje C#, mediante Windows Forms y el IDE Visual Studio. El diseño se enfocó en darle al usuario una experiencia completamente intuitiva dentro de la interfaz y del recorrido virtual, para ello se utilizaron botones básicos para realizar las funciones disponibles para el usuario. El sistema permite al usuario crear un perfil personalizado para guardar sus

avances en diferentes actividades; los grandes grupos de opciones disponibles se categorizan en: entrenamiento y perfil.

La creación del entorno virtual donde el usuario interactúa, es creado utilizando Unity, en específico, un demo de una isla elaborada con gran nivel de detalle, a la cual se le hicieron ciertas modificaciones menores para adecuarse a las especificaciones del proyecto; entre las cuales se encuentran: ambiente, nuevas superficies de colisión para guiar al/los participante(s) y esquema de la cámara.

Se crearon 4 avatares diferentes para reconocer a los usuarios registrados dentro del entorno virtual, los mismos que fueron creados en 3DS Max. También, se les incorporó procesamiento físico y capacidades de colisión.

Para hacer el recorrido se asignó un “guía” el cual maneja una unión tipo amortiguado, el cual se puede describir como un resorte que une a los participantes activos con el guía. Las capsulas de colisión de cada avatar está configurada para que mantengas una distancia considerable entre ellos mismo.

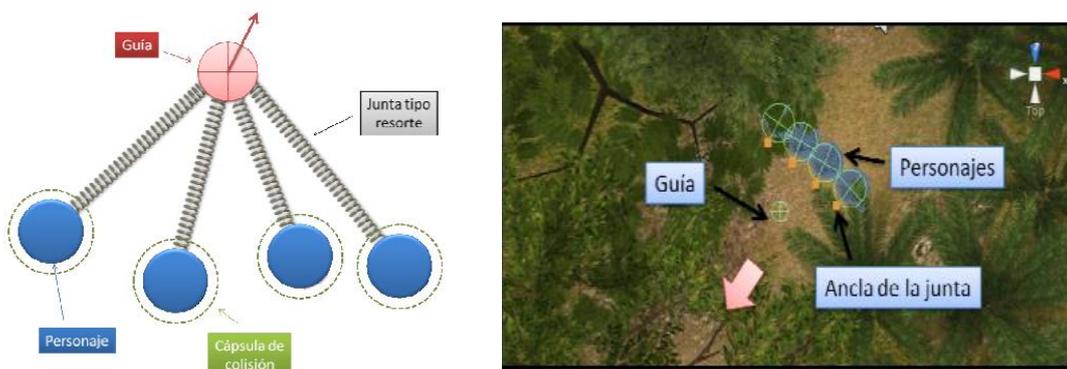


Fig. 2.8 Esquema del sistema de guía tipo resorte.

La ruta que diseñada para hacer el recorrido se puede atravesar a modo de circuito que fue creado a partir de Cubic Spline Curves, las cuales son un tipo de representación del modelado gráfico. A grandes rasgos, es un método con el cual se describen curvas entre dos puntos, que a su vez contiene puntos adicionales para definir la geometría de la curva final.

En conclusión, el proyecto de Ortiz Salcedo muestra una técnica diferente trabajando con Unity, uno de los programas elegidos para el desarrollo del presente proyecto, lo cual muestra un panorama y alcances diferentes en la misma línea de investigación, datos que serán tomados a lo largo del presente trabajo.

2.4.3 ANÁLISIS Y DISEÑO DE LA INTERFAZ PARA UN SISTEMA DE APRENDIZAJE COLABORATIVO APOYADA POR UN AGENTE TUTOR INTELIGENTE

En 2012, se presenta el resultado de un análisis metodológico con el propósito de definir los requerimientos de funcionalidad en un sistema mediante una interfaz que permite el aprendizaje colaborativo, para ello se implementan agentes inteligentes, en específico un agente pedagógico con características reactivas.

En dicho trabajo, se abordan diversas perspectivas metodológicas que corresponden a diferentes campos del conocimiento, entre ellos: agentes pedagógicos, modelos instruccionales y modelos para la creación de una interfaz colaborativa.

Se explica el comportamiento de un agente pedagógico que tiene implícito la reactividad que lo dota de un comportamiento flexible que le permite adaptarse al entorno donde se encuentra “inmerso”, mejor dicho, con el que interactúa, de esta manera poder desarrollar el diseño de un agente de esta categoría. (Garduño, 2012)



Fig. 2.9 Interfaz General del Software de Aprendizaje (A. Garduño Enrique, 2012)

El prototipo de sistema realizado para ejemplificar dicho proyecto, muestra una interfaz donde el usuario interactúa con el tutor inteligente y los participantes de sesión.

Las características reactivas presentadas en el agente inteligentes son de enseñanza y aprendizaje las cuales están presentadas en la interfaz de la siguiente manera: función que permite realizar un seguimiento del alumno y trazar un plan de trabajo, se crea un modelo de estudiante en base a la arquitectura propuesta por Laureano y Arriaga (2000), por último, se basa en un conjunto de estrategias didácticas que debe incluir todo agente tutor, según Lester (2001).

2.4.4 ANIMACIÓN Y RECONSTRUCCIÓN SOBRE LA ZONA PREHISPÁNICA DE MONTE ALBÁN, OAXACA

En el año 2014, Juan Manuel Jiménez Gutiérrez y Luis Fernando Mendoza, desarrollan una aplicación 3D interactiva, con la cual es posible realizar un paseo virtual por la Ciudad Prehispánica de Monte Albán.

Estas son unas ruinas ubicadas en el estado de Oaxaca, México. Dicho paseo interactivo no solo muestra la arquitectura estas importantes ruinas, sino también ofrece información histórica de la ciudad y permite experimentar parte del entorno de esta ciudad prehispánica.

Esta aplicación se desarrolla dentro de un gran campo de exploración dentro de la Realidad Virtual, ya que se utilizan nuevas tecnologías para recrear espacios importantes dentro de la historicidad de México, que representa un sitio de turismo y educación para todo el público.



Fig. 2.10 Vista de la Cámara de un Ángulo Externo. (J. Manuel Jiménez y Luis F. Mendoza, 2014)

El objetivo de esta aplicación realizada con el software de diseño 3D Studio Max de Autodesk es mostrar al público una parte significativa de la historia de México desde una perspectiva innovadora, utilizando los mínimos recursos computacionales y/o económicos. (Juan Manuel Jiménez Gutiérrez, 2014)

Las especificaciones tales como texturas, sonidos, animaciones, comportamientos, eventos, fueron creados mediante 3D Studio Max. Los modelos están contruidos a base de polígonos, y la estructura de programación es orientada a objetos, como lo es habitualmente. Las animaciones implementadas han sido creadas para funcionar en tiempo real.

Este paseo virtual tiene la peculiaridad de mostrar un avatar visible que representa al usuario, y otros personajes que interactúan dentro de la escena pero no con el usuario. El

avatar/personaje principal tiene definido cierto comportamiento que se efectuara cada vez que el usuario de una instrucción que lo invoque, por ejemplo, si el usuario quiere correr el avatar lo interpretara y hará el movimiento definido.

Dentro del sistema, también se puede encontrar un mapa del sitio, el cual funciona mediante texturas que simulan las características del entorno. La primera textura presenta el contorno de los edificios correspondientes a la zona arqueológica, la segunda textura se utiliza para crear la transparencia expuesta a la primera textura, de esta forma solo se

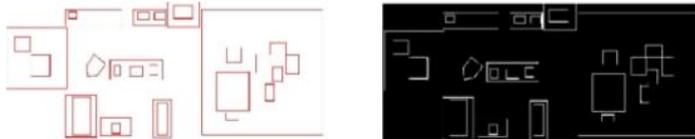


Fig. 2.11 Textura del entorno y textura de transparencia para el mapa. (J. Manuel Jiménez y Luis F. Mendoza, 2014)

visualizan los contornos de la zona, y finalmente la textura restante se usa como un color solido que representara el objetivo dentro del mapa, el cual también será la posición actual del usuario.

Las rutas posibles dentro del mapa, se generan mediante operaciones matemáticas que representan indicadores que van calculando las coordenadas y rutas dentro de la zona arqueológica en base a la posición actual del usuario.

En la Fig. 2:11, se muestra como el autor del sistema distribuyo los indicadores: el indicador “A” genera la proyección del mapa en la cámara mientras el usuario este en el paseo virtual, el cual contiene varios atributos que contribuyen en varias funciones dentro del sistema, el indicador “B” se encuentra en $(0.4527 * A) + 72.63$, un canal con una única entrada: el componente en X del vector de posición del avatar de usuario, el indicador “C” sirve como integrador para crear la relación de variable de posición con el objetivo, por último el indicador “D” señala el canal objetivo, este contiene la imagen del objetivo y los atributos que proporcionaran las coordenadas del objetivo en el sistema de coordenadas de la pantalla donde se proyecta la cámara. (Juan Manuel Jiménez Gutiérrez, 2014)

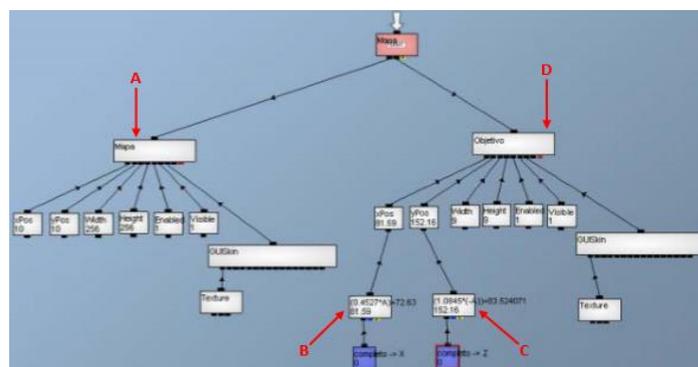


Fig. 2.12 Rutas organizadas por canales para mostrar la posición del usuario en tiempo real. (J. Manuel Jiménez y Luis F. Mendoza, 2014)

2.4.5 MODELO DE UN ENTORNO VIRTUAL INTELIGENTE BASADO EN LA PERCEPCIÓN Y EL RAZONAMIENTO DE SUS ELEMENTOS CON UN PERSONAJE PARA LA GENERACIÓN DE REALISMO.

En el año 2015, es creado el modelo de un entorno virtual desarrollado en el campo de “Entornos Virtuales Inteligentes”, los cuales son conformados por la interacción de tecnologías involucradas en el desarrollo de gráficos interactivos 3D y técnicas de Inteligencia Artificial. (Santiago, 2015)

La problemática que este proyecto aborda, es principalmente la falta de un estándar para desarrollar sistemas en esta área del conocimiento. Por lo cual, se propone un modelo geométrico y cinemático para fundamentar el modelo del entorno virtual, y así potencializar el rendimiento del Entorno Virtual Inteligente respecto a sus niveles cognitivos y comportamentales, respectivamente. (Santiago, 2015)



Fig. 2.13 Interfaz de Software basado en metodología SAMI (M. Santiago Sandra, 2015)

Cabe mencionar que para desarrollar este modelo se implementaron algunas técnicas de Inteligencia Artificial aplicadas al razonamiento y la percepción, a su vez fueron evaluadas y comparadas utilizando distintas métricas, para así escoger la Red Neuronal Artificial en el que se basa el modelo. (Santiago, 2015)

De forma demostrativa, se desarrolló en paralelo un sistema experimental para comprobar la eficacia del modelo propuesto.

C

APÍTULO III: MARCO TEÓRICO

Vivimos en una sociedad exquisitamente dependiente
de la ciencia y la tecnología, en la cual casi nadie
sabe nada acerca de la ciencia y la tecnología.

- Carl Sagan

3.1 SISTEMAS EN REALIDAD VIRTUAL

La Realidad Virtual (RV), es una tecnología que permite la comunicación hombre-máquina, para que los seres humanos puedan interactuar, por ejemplo, con una computadora y sus datos complejos. Se ha llegado a conseguir el objetivo combinando tecnologías informáticas modernas con tecnología de detección y medición, tecnología de comunicación, tecnología de simulación y tecnología microelectrónica. (Qing-yun Dai, 2013)

Los expertos en innovación tecnológica han considerado 2016 como el año de despegue de la RV y lo cierto es que la mayoría de las grandes empresas tecnológicas han decidido desarrollar productos o servicios relacionados con esta novedosa forma de consumo de contenidos e interacción en los entornos digitales. (Robles, 2017)

Entre las definiciones de RV conocidas dentro de la literatura, se encuentra (Lozano & Calderón, 2004) los cuales afirman que la RV se conoce como la tecnología computacional cuyo objetivo es representar ambientes tridimensionales los cuales, generalmente, son experiencias visuales en una pantalla o en un dispositivo estereoscópico⁴. En algunas ocasiones incluyen sensaciones auditivas, sensoriales, etc.; actualmente las definiciones al respecto no varían mucho, y hacen referencia a las mismas características.

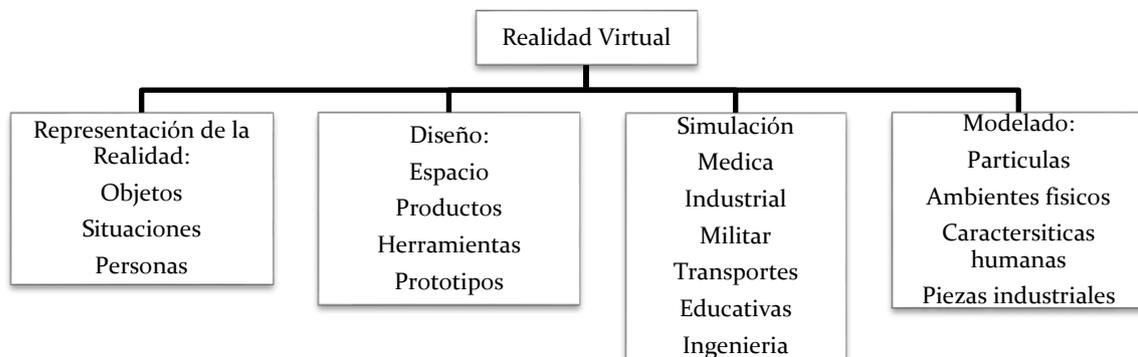


Fig. 3.1 Aplicaciones y Usos de la RV. (Adaptación de Flores C. Jesús A, 2014).

En la *Fig. 3.1* se pueden observar algunas de las aplicaciones dentro de diversas áreas del conocimiento más utilizadas en RV. Retomando la definición de Lozano & Calderón, se puede concluir que un entorno virtual es una aplicación/sistema para dispositivos electrónicos, que permite al usuario, por medio de la implementación de tecnologías de RV, navegar e interactuar en un entorno tridimensional, es aquí donde se observa la aplicabilidad de la RV en el presente trabajo.

⁴ La visión estereoscópica es una facultad físico - psicológica, que permite ver en tercera dimensión aquellos objetos que contempla mediante su visión binocular (Ministerio de Ciencia, 2013).

Dentro del desarrollo de sistemas de RV se ha normalizado una clasificación, en la cual la diferencia que predomina es en cuan inmerso el usuario puede encontrarse con cada uno de estos tipos: (Luna, 2012)

- **Sistemas RV de escritorio:** Estos sistemas son los más comunes, se pueden encontrar en los juegos o aplicaciones de avanzadas de escritorio. Consiste en presentar imágenes en 3D sin necesidad de equipos especiales o específicos para RV. La imagen puede estar en un monitor o directamente proyectada en otra superficie.
- **Cabina de simulación:** De igual manera a los sistemas de escritorio con la diferencia que el usuario experimenta un ambiente especial dentro de una cabina. Ejemplos típicos son los simuladores de vuelo.
- **Realidad aumentada:** Este sistema permite al usuario valerse del mundo real pero a través de cristales que complementan las imágenes con diagramas, textos o referencias. Se suele ver esto en las películas con soldados futuristas, solamente que no es cosa del futuro
- **RV en segunda persona (o unencumbered systems):** En estos sistemas sí ya se utilizan HMD e involucran respuestas en tiempo real como consecuencia de las acciones detectadas por los guantes, cascos y otros dispositivos conectados al usuario. No es un sistema de inmersión completo porque el usuario se ve a sí mismo dentro de la escena.
- **Sistemas de telepresencia:** fundamentalmente para realizar tareas a distancia. El usuario opera a control remoto, las acciones que él realiza son repetidas por un agente remoto que hace las veces del usuario, a la vez que el sistema simula, para el usuario, el ambiente en el que se está moviendo el agente. Ejemplos típicos son los robots operados a distancia para: desarmar bombas, hacer cirugías, explorar en ambientes de alto riesgo para la vida humana, etc.
- **Sistemas inmersos:** Llegando por fin los sistemas inmersos que simulan de manera más “real” la realidad. Valiéndose para esto de todos los dispositivos conocidos, HMD, guantes de datos, seguidores de movimiento, chalecos de movimientos, etc. La diferencia con la telepresencia es que no se están repitiendo las acciones del usuario por un agente, de lo contrario sería un sistema de telepresencia inmerso.

Los puntos anteriores describen sistemas de RV en diversos *niveles de inmersión*, sin embargo, esta no es la única característica importante en este tipo de sistemas. La interactividad desempeña un papel muy importante en su funcionamiento, y para lo cual, también existen una clasificación que muestra los diferentes *niveles de interactividad*.

En este contexto, la interacción apunta a la posibilidad de cambiar la linealidad de los eventos dentro de un sistema, los cuales se generan mediante ciertos estímulos interpretados por los sensores que recaban información; así entonces, el mundo virtual (personajes y/u objetos) reaccionaran a las acciones del usuario y, en muchos casos, de estos mismos. El usuario, también responderá a los cambios generados por sus acciones,

de esta manera se cumple el ciclo interactivo, donde gran parte del “curso de acción” del sistema en RV, dependerá de este. (Luna, 2012)

La interacción está dada por dos aspectos importantes que definirán su complejidad, los cuales son; *dinámica del ambiente y navegación*. La primera habla sobre las reglas que determinan el comportamiento de los componentes del mundo virtual en la interacción con el usuario, generalmente intercambiando información; el segundo término, navegación, se refiere a la capacidad del usuario para moverse dentro del mundo virtual independientemente, las restricciones son generadas por el programador, las cuales permitirán diferentes grados de libertad en las acciones del usuario (caminar, correr, nadar, volar, etc.). Dentro de la navegación también se toma en cuenta el punto de vista del usuario, es decir, el posicionamiento de visión: primera persona, tercera persona, ambos. (Luna, 2012)

En las aplicaciones de RV, generalmente se implementa una *interacción en tiempo real*, lo cual quiere decir que la respuesta del sistema a la acción del usuario será cero, es decir el menor retardo posible. Esto generara una respuesta en un tiempo imperceptible para el sistema sensorial humano. Para conseguir la imperceptibilidad de respuesta se han estudiado varios casos para los sentidos receptores del ser humano. En el primer caso, se pretende mantener la sensación de movimiento continuo, lo cual se logra mediante una secuencia de imágenes generadas, mínimo, a 10 Hz (10 hercios o 10 veces por segundo), sin embargo, lo recomendable es mantenerse en 25 a 30 Hz, el máximo óptimo se considera en 60 Hz o más. (Luna, 2012)

En RV el nivel de interacción en relación con la funcionalidad entre usuario-sistema, se categoriza en tres niveles, los cuales hacen referencia al grado de intervención que el usuario dispone en el entorno virtual (véase *Tabla 3.1*).

Nivel	Descripción
Explorativa	Este nivel es representado comúnmente por los recorridos, tales como las caminatas (walk-throughs) y vuelos (fly-throughs) a través de entornos virtuales. Se caracteriza principalmente por la nula modificación al entorno o cualquier objeto dentro de este, que pueda realizar el usuario, es decir, puede seleccionar lo que desea ver, como y cuando quiere verlo. El usuario solo puede modificar la posición y orientación del punto de vista virtual (avatar y/o sujeto virtual), sin alterar el entorno que explora.
Manipulativa	Es el siguiente nivel de interacción, permite una relación más activa con el entorno. El sistema permite al usuario manipular los objetos y sus componentes, desde un punto de vista geométrico. Dicho de otra forma, el usuario podrá mover, rotar, y escalar objetos, sin modificar el entorno virtual; lo cual supone una “reconfiguración”, sin un cambio a nivel funcional o reactivo.
Contributiva	Es el máximo nivel de interactividad considerado en un entorno virtual, dado a sus características complejas permite al usuario realizar cambios en el estados funcional y reactivo del sistema. Es posible modificar los objetos del entorno, desde su geometría hasta el estado mismo de este, por lo cual este elemento resultara diferente a su presentación original.

Tabla 3.1 Niveles de interacción en relación funcional usuario-sistema RV. (Adaptación de Luna, Jaqueline, 2012)

3.1.1 ENTORNOS VIRTUALES TRIDIMENSIONALES (EV3D)

Formalmente a este tipo de sistemas, se le conoce como un *Entorno Virtual Tridimensional (EV3D)* el cual hace referencia a la elaboración de objetos particulares que posteriormente serán utilizados dentro de una escena. Existen diferentes tipo de primitivas poligonales y de geometría, los cuales representan la información visualizada en 3D, por otro lado la “escena” es representada en una estructura denominada “grafo de escena”, el cual es un grafo dirigido acíclico de nodos que contiene los datos que determinan un escenario virtual y controlan su proceso de sombreado. (Gonzalo Luzardo, 2014)

En un EV3D, es un ordenador quien se encarga de generar impresiones sensoriales, las cuales son enviadas al usuario a través sus sentidos. El nivel de inmersión, que es la sensación de presencia dentro del entorno, es determinado por la calidad de las impresiones generadas por el ordenador. Para crear una experiencia mucho más realista, el usuario tendría que recibir información hacia todos sus sentidos mediante una alta resolución, calidad y consistencia, de manera constante. Sin embargo, en el mercado predominan las aplicaciones que estimulan solo uno o unos cuantos sentidos y generalmente la calidad es muy baja en las representaciones. (V. Juřik, July 2016) En este contexto, se han considerado tres importantes niveles de inmersión:

- **Entornos Virtuales de Escritorio.** Son los entornos virtuales más comunes, también conocidos como “ventana del mundo”. El nivel de inmersión en este tipo de aplicaciones es básico, sin embargo no pierde de vista la calidad. No se involucra hardware especializado para RV, solo se necesita de un monitor convencional para mostrar la imagen vista desde una perspectiva en primera persona hacia el exterior, como si se tratase de una ventana.
- **Entornos Virtuales Ojo de Pez.** Este tipo de aplicación es muy parecida a la anterior, se basan en el seguimiento de la cabeza, es decir, la aplicación muestra una imagen desde la perspectiva del movimiento de la cabeza del usuario, de esta manera la sensación de presencia en el EV es mucho mayor. Estas aplicaciones utilizan monitores convencionales o gafas LCD con visión estereoscópica para presentar la información.
- **Sistema Inmersivos.** Se trata de la versión más actual de entornos virtuales, estos permiten al usuario adentrarse totalmente en el entorno generado por ordenador mediante el uso de un dispositivo HMD, el cual, muestra una vista estereoscópica de la escena creada de acuerdo a la posición y orientación del usuario. Estos sistemas, suelen incluir características que potencializan su efecto, como audio especializado, interfaces táctiles y/o sensoriales, detección de movimientos físicos, etc.

En el presente trabajo, se utilizara la categoría de Entorno virtuales 3D básica, que, como ya se mencionó, son las aplicaciones de escritorio. Desde un punto de vista computacional, independientemente del modelo que se esté interesado en realizar

virtualmente, para la generación de un EV3D se deben tener en cuenta tres modelos formalizados que conformaran la aplicación informática los cuales son: modelo geométrico, modelo en tiempo de ejecución o comportamental y un modelo de interacción con el usuario. (Gonzalo Luzardo, 2014)

3.1.1.1 Modelo Geométrico

Este modelo es el encargado del bajo nivel gráfico, lo cual quiere decir que se encarga de los distintos tipos de formatos gráficos utilizados y del modelo interno empleado para la ejecución de las ordenes de dibujado, también conocido como render, los cuales resumen las propiedades elementales para la visualización de cualquier EV3D. Estos entornos generalmente, están conformados en base a un conjunto de primitivas poligonales, lineales, de texto, superficies planas/curvas, etc, las cuales constituyen la información espacial que se puede apreciar visualmente en 3D. (Santiago, 2015)

La implantación de hardware destinado al procesamiento de esta información visual ha ayudado a que los sistemas gráficos de tiempo real sean capaces de dibujas miles de polígonos por segundo lo cual atiende perfectamente la demanda de gran variedad de aplicaciones gráficas creadas en la actualidad. (Santiago, 2015)

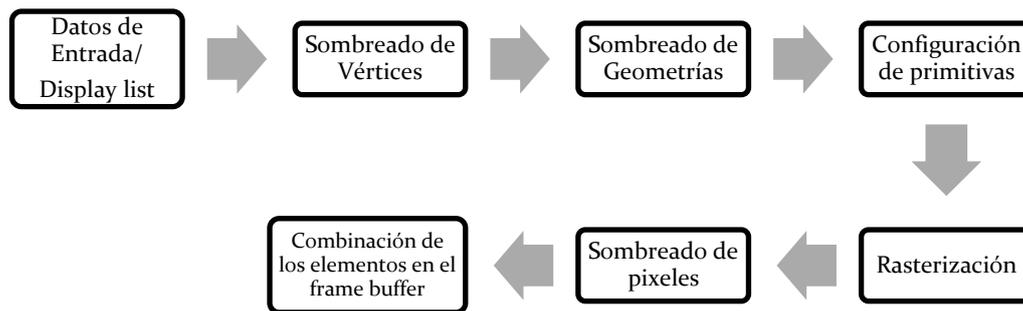


Fig. 3.2 Esquema Clásico de Renderización soportada por Hardware de Procesamiento Gráfico dentro de un EV3D. Elaboración propia.

La Fig. 3.2 Muestra el proceso clásico de renderización, también conocido como tubería gráfica, en un dispositivo que permita el procesamiento gráfico, es decir, la técnica de rasterización⁵ soportada por estos dispositivos, donde se acepta como entrada estructuras de datos que representan escenas 3D y como resultado proporciona una imagen raster 2D, la cual, es captada por una cámara virtual que se localiza dentro del escenario 3D. Algunas de las técnicas de rasterización aceptadas como estándares en el campo de gráficos por ordenadores son OpenGL y Direct3D. (Gonzalo Luzardo, 2014)

Es importante mencionar que mientras los EV3D fueron aumentando en complejidad, el modo de dibujo convencional resultaba no tan eficiente para la organización y mantenimiento de las escenas recreadas, lo que abrió paso a la creación de una nueva estructura conocida como **grafo de escena** (scene graph, Fig. 17b). El

⁵ Proceso por el cual una imagen descrita en un formato gráfico vectorial se convierte en un conjunto de pixeles o puntos para ser desarrollados en un medio de salida digital, como una pantalla de computadora o dispositivo electrónico. (Vargas, 2012)

objetivo de esta estructura es organizar jerárquicamente la geometría incluida en un escenario 3D, haciendo más sencillo el recorte automático de las áreas no visibles, de esta manera se aumenta el número de imágenes generadas por segundo. (Santiago, 2015)

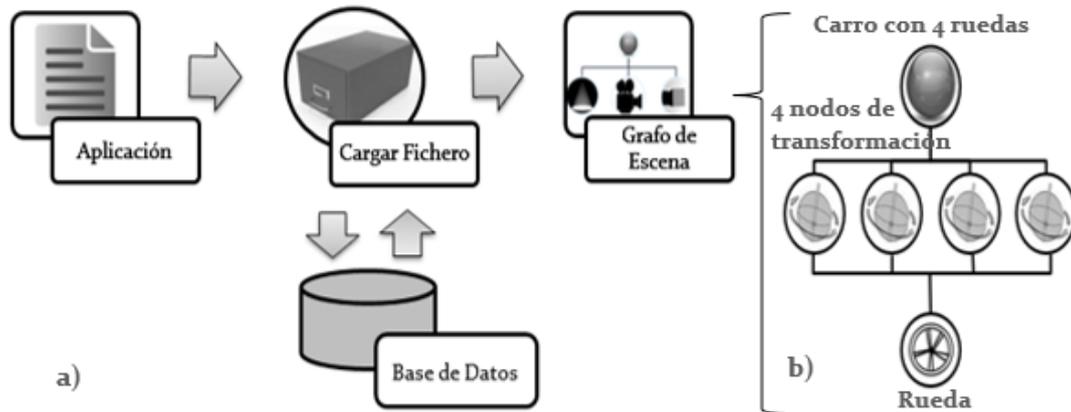


Fig. 3.3 a) Esquema de Carga de un Modelo Geométrico en un EV. b) Ejemplo de grafo de escena jerárquico. Elaboración propia.

Lenguajes de modelado de escenas 3D, como el clásico VRML, así como gran parte de las API's (Application Programmer's Interface) actuales para el soporte del modelo geométrico, como OpenGL-Performer, OpenInventor, Java3D, etc., se basan en este tipo de estructura (*Fig. 3.3 b*) para la organización de la escena, agrupando sus nodos de forma jerárquica. Es un ejemplo de la simulación de un carro con 4 ruedas, el cual necesita de un nodo geométrico más cuatro nodos de transformación, esto permitirá que cada rueda pueda tener un comportamiento independiente entre ellas. Por otro lado, la *Fig. 3.3 a*, representa el proceso de como carga un fichero 3D, este creará un grafo de escena básico que contendrá los modelos geométricos creados, luz de ambiente, y una cámara, de forma que los modelos incluidos se iluminen para su correcta visualización. (Ibañez, 2004)

3.1.1.2 Modelo Cinemático

Al hacer referencia a la capacidad de los objetos, situados dentro la escena virtual, de recrear un movimiento físico realista se le conoce como **Cinemática**. Esta provee propiedades a los objetos para controlar sus movimientos, ya sean previstos o el resultado de simulaciones que en un ambiente real serían **complicados, costosos o riesgosos** de realizar.

En los EV3D resulta una característica muy importante la calidad de la similitud que tenga con la vida real, ya que en el mundo natural los movimientos que describen a un objeto donde actúan las leyes físicas como la gravedad o inercia, contribuyen a la lógica y coherencia de su comportamiento. Por otro lado, el EV3D llegará a ser tan convincente como el movimiento que los objetos que los conforman.

La relación que tengan los elementos entre sí dentro de un EV, hacen que la masa sea un factor fundamental, pues este permite la influencia de la inercia, perturbación, gravedad, choque, y otras fuerzas físicas que podrían intervenir en un EV, estos factores y la capacidad de colisionar entre sí, ayudan a los objetos a que tengan movimientos naturales y posean control sobre el entorno. (Santiago, 2015)

Particularmente, la propiedad de colisionar es un evento en los objetos primordial dentro de los EV3D, el cual puede ser el de una entidad que se traslada a lo largo de alguna superficie, un elemento interceptando a otro, etc. Es importante considerar en todos estos casos las características cinemáticas que deben cumplir los objetos involucrados.

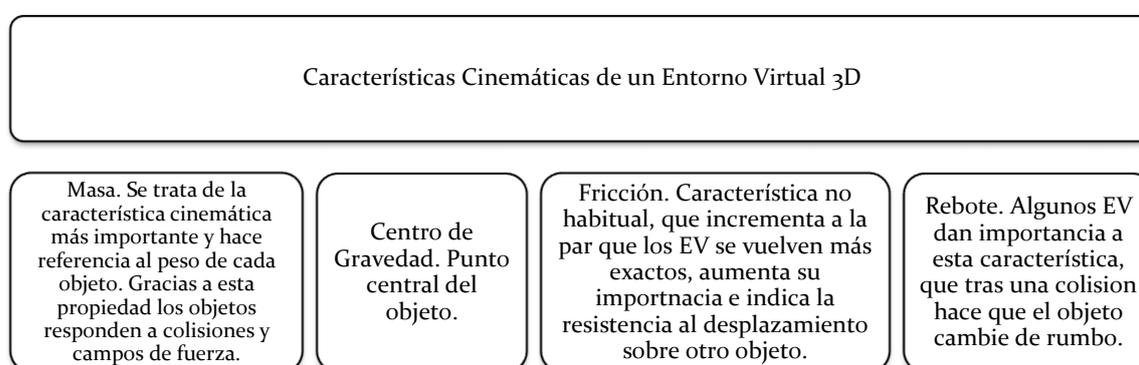


Fig. 3.4 Principales características cinemáticas dentro de un EV3D. Elaboración propia.

Otros aspectos que abarca el modelo cinemático son la **orientación y pivote** de cada uno de los objetos creados, el cual representa el punto céntrico en el espacio del EV3D y les permite desplazarse, rotar, y modificar su tamaño dentro de la escena 3D.

Las **articulaciones** también desempeñan un papel fundamental en el movimiento de los personajes en escena. Su primordial uso es controlar objetos orgánicos, no es un elemento estático, por el contrario, se expande a diversos tipos de usos dentro de la cinemática. Las articulaciones son objetos que no son visibles dentro de la escena, sin embargo, controlan los objetos que se visualizan de forma precisa. Al utilizar articulaciones, es necesario implementar funciones que ofrecen mayor fluidez y precisión en los movimientos; para ello, las **cadena cinemática relativa e inversa** constituyen elementos mecánicos virtuales que manejan las articulaciones de manera más sencilla.

Para manejar los elementos que no pueden ser representados por figuras poligonales, por ejemplo el agua, aire o fuego son utilizadas **partículas**, las cuales son pequeños componentes con características específicas que permiten implementar este tipo de elementos. En un EV3D es un emisor o superficie, quien se encarga de generar estas partículas cuyas características que elevan su rango de dinamismo, las cuales son: (Santiago, 2015)

- **Conservación.** Conserva el impulso que tiene. Si la partícula no tiene esta característica puede tener movimientos más naturales, de lo contrario su desplazamiento sería infinito y no representaría ningún comportamiento del mundo real, exceptuando un movimiento simulado en el espacio exterior
- **Colisión.** Es la capacidad de interactuar cinemáticamente con otros objetos poligonales dentro de la escena
- **Figura.** La forma fundamental de una partícula es el punto, sin embargo, las partículas pueden tener diferentes figuras, por ejemplo: líneas, esferas, círculos, etc., aunado a esto, toman color y transparencia
- **Dirección.** Es la primera característica que genera el emisor o superficie al crear la partícula, pues esta debe llevar una dirección específica, lo cual no implica un movimiento posterior.
- **Vida útil.** Es la vida útil que implica el tiempo de existencia de una partícula, posterior a su nacimiento. Es un factor importante, porque al no ser considerado, pueden saturar el entorno y sus recursos, lo que finalmente terminara colapsando.
- **Impulso.** Al asignar un valor a esta característica, hará que la partícula se mueva a esa velocidad, en la dirección que fue creada.

3.1.1.3 Modelo Comportamental

Dentro de un EV3D, es el encargado del comportamiento de este como aplicación informática en tiempo de ejecución, dicho de otra manera, modela el comportamiento dinámico de los objetos dentro del EV3D. Al hablar de “comportamiento” en este modelo, se refiere a la modificación de una o más de sus propiedades básicas como puede ser posición, color, visibilidad u orientación. Estos comportamiento pueden estar especificados dentro del fichero del modelo geométrico (*Fig. 3:3a*), para ser representado en el grafo de escena indicado.

De acuerdo a la literatura encontrada al respecto, este modelo se puede dividir en dos categorías: (Ibañez, 2004)

Modelo Comportamental Dependiente	Modelo Comportamental Independiente
<ul style="list-style-type: none"> ➤ Para satisfacer la necesidad de un modelo comportamental más flexible y robusto, los entornos dirigidos por eventos son los más utilizados. ➤ Este modelo comportamental maneja los cambios del EV3D independientemente de su visualización. ➤ El comportamiento de un objeto dentro de la escena puede ser definido en base a sus reacciones ante los eventos producidos en el entorno (acciones del usuario, actuaciones de algún personaje) 	<ul style="list-style-type: none"> ➤ En los entornos basados en el tiempo existe un tiempo absoluto, este se inicia generalmente al comienzo de una simulación y es utilizado para determinar el inicio y el fin de los cambios producidos. ➤ La orientación y posición, por ejemplo, de un avión que vuela alrededor de una ciudad, se animan de manera sencilla con funciones dependientes del tiempo transcurrido a partir de empezar el vuelo.

<p>inteligente, colisiones entre objetos, cambios en la iluminación)</p> <ul style="list-style-type: none"> ➤ Los eventos necesarios para la simulación son guardados de forma general en una cola, para después ser dirigidos hacia los objetos que demandan su recepción. ➤ Este modelo de comunicación es más adaptable a la simulación de mundos 3D con “enfoque casual” en su comportamiento, lo cual es porque proporciona un método común para controlar los cambios del entorno a partir sus causas. ➤ La mayoría de entorno o motores de juegos constituyen actualmente los EV3D dirigidos por eventos. 	<ul style="list-style-type: none"> ➤ El dinamismo del EV3D tiene implícito en su desarrollo variadas funciones matemáticas o modelos dinámicos, encargados de los cambios graduales de color, posición, etc., para los objetos o personajes que componen el entorno. ➤ Los sistemas basados en un tiempo absoluto, en general son utilizados en EV3D orientados al entretenimiento, donde el modelo dinámico del objeto simulado (avión, carro, etc) aporta la mayor parte del dinamismo del entorno, donde los demás elementos que conforman la escena son estáticos utilizados para decoración. ➤ Dificulta la integración de personajes u objetos 3D autónomos.
---	---

Fig. 3.5 Tabla comparativa del Modelo Comportamental en sus dos categorías. Elaboración Propia.

En el **modelo comportamental dependiente** visto (Fig. 3.5), se habla de un **modelo dirigido por eventos** utilizados mayormente en entornos de **naturaleza altamente reactiva**, el comportamiento de los personajes se fundamenta en este tipo de sistemas de encaminamiento global, el cual controla la recepción y envío de los cambios admitidos, así consigue un modelo de comunicación que hace más sencilla la integración de personajes autónomos tridimensionales, pues en su tiempo de ejecución mantiene el flujo de información entre el personaje y su entorno. Los alcances de este modelo comportamental se adaptan de forma precisa al objetivo del presente proyecto, por lo cual se considera implementarlo en el desarrollo del mismo.

Por otro lado, en la gran mayoría de EV3D generados, existe la expectativa de la interacción libre con los componentes del entorno. Para ello, la mayoría de los EV3D basados en algún **grafo de escena** colocan al usuario dentro del mismo, lo cual define el modelo de interacción que estará centrado en el usuario. La ventaja de esta estructura es al momento de dibujar en la escena, pues el grafo siempre tendrá información actual del usuario, haciendo que la imagen sea renderizada desde el punto de vista del usuario. Lo cual, significa que, por ejemplo, al utilizar un modelo de un vehículo terrestre o aéreo, se coloque al usuario en un nodo de transformación que simule el movimiento que tendría dentro de algún vehículo, cual sea que se utilice, y así producir la sensación de ir dentro.

Finalmente, resultado del análisis de la literatura citada, el comportamiento de los objetos de la escena se refiere a la alteración en tiempo de ejecución de las propiedades básicas de los elementos que conforman el entorno, en consecuencia de la interacción del usuario con otros objetos del mismo entorno, estos objetos, muchas veces, carecen de comportamiento propio que simulen la realidad. Sin embargo, actualmente se trabaja en herramientas para EV3D para proveer de comportamiento específico y detallado a estos

objetos dentro de la escena⁶. Gonzalo Luzardo (2014), por ejemplo, el uso de técnicas de **inteligencia artificial**, en específico, **agentes inteligentes**.

Cabe mencionar, que tal como los modelos descritos anteriormente que admiten la gran mayoría de los EV3D, al integrar agentes en EV3D se tienen que implementar modelos, de igual forma, que permitan la comunicación personaje↔entorno y usuario↔entorno, apropiados al tipo de personaje que se piensa recrear.

Retomando, es importante recordar que este proceso señala al desarrollo de un mundo virtual, el cual se reconoce como un entorno donde se pueden hacer muchas cosas que no son posibles en realidad, dependiendo el objetivo del proyecto, se puede dejar objetos suspendidos en el espacio, volar, etc. Lo cual, lleva a examinar la relación entre las simulaciones y la realidad, pues algo fundamental es definir si el diseño del ambiente se relaciona con la realidad o si el mundo virtual existe por sí solo (Crookall, Oxford & Saunders, 1987). Se debe entender el propósito del ambiente para saber cómo se pueden usar simulaciones, para crear prototipos de diseños. (Thompson, 2017)

En la actualidad, el uso de simulaciones y/o modelos en RV o realidad aumentada (RA) es cada vez más recurrente ya que a satisfecho muchas necesidades es diversos ámbitos. Gredler (2004), es uno de los autores que ha examinado la relación entre las simulaciones y el aprendizaje; tal relación ha dado paso a un nuevo paradigma de enseñanza, donde las TIC's son más recurrentes. En este contexto, se hace referencia a las simulaciones por su efectividad en diferentes situaciones de aprendizaje: las simulaciones ofrecen una nueva faceta de aprendizaje experimental y simbólico; por lo cual diseñar una simulación de manera efectiva, implicaría una simulación donde sea posible la aplicación del conocimiento de diversas ramas del conocimiento. (Thompson, 2017)

Otro aspecto importante de mencionar es que para desarrollar un mundo o ambiente virtual con todas las etapas que este implica existen diferentes tipos de programas que hacen esta tarea más accesible al público experto y principiante, donde el espacio de trabajo es muy similar a un entorno 3D. Desde las coordenadas que rigen el proceso de creación hasta los movimientos programados y por programar. Por lo cual se puede deducir que quien maneja bien cualquier programa de desarrollo 3D también sabe cuáles son las consideraciones y características que existen dentro de un entorno virtual. (Thompson, 2017) En lo próximo del documento, se profundizara en los programas, herramientas y técnicas utilizadas en este ámbito.

⁶ Una escena en el ámbito 3D, hace referencia a un archivo que contiene toda la información necesaria para identificar y posicionar todos los modelos, cámaras y luces para su posterior renderización. (Blanco, 2010)

3.1.2 ENTORNOS VIRTUALES INTELIGENTES TRIDIMENSIONALES (EVI3D)

Pese a ello y dejando de lado la modificación de lo visual de los objetos debido a la interacción con el usuario, se debe hacer énfasis en la forma en la que un objeto y su entorno puedan comunicarse, esto aumenta el nivel de interacción que pueda ejecutarse dentro del entorno virtual. La cuestión ahora es elegir cuando estas interacciones son iniciadas por el objeto y cuando por el entorno que lo rodea.

Para la tarea de asignar comportamientos complejos a los objetos y su entorno como resultados de su propia interacción surge la siguiente generación de los EV3D. Son aquellos cuyo objetivo principal es aumentar las capacidades de comportamiento e interactivas de los EV3D, lo que logra mediante la incorporación de herramientas de Inteligencia Artificial interactuando con el sistema gráfico, con esto alcanza una nueva característica derivando así en un sistema de distribución de eventos de forma asincrónica⁷, la cual es la que lo diferencia de un entorno virtual convencional y de ahí su ventaja en sistemas de alta complejidad. Esta categoría dentro de los EV3D, se le conoce como **Entorno Virtuales Inteligentes 3D (EVI3D)**, donde más allá de los efectos visuales para dotar de realismo al entorno, se ocupa de la experiencia del usuario dentro de este. La experiencia de navegar por un entorno virtual 3D estático, donde los objetos visibles incluidos no poseen un comportamiento propio o de respuesta con el usuario, puede hacer que este no se sienta lo suficientemente inmerso dentro del entorno, limitando los alcances de las tecnologías de realidad virtual que puedan verse involucradas y el propio EV3D. (Santiago, 2015)

Todo esto representa de manera clara, la convergencia que se ha generado entre la **Inteligencia Artificial (IA)** y los EV3D. La inserción de técnicas de IA interactuando con sistema de sistemas gráficos de apoyo a los EV3D, aporta: (Xin Bai, 2013)

- **Optimiza la interactividad.** Este tipo de entornos requiere de un alto nivel de interacción y ayuda al usuario, por lo cual resulta necesario recurrir a una “asistencia inteligente”, la cual buscara nuevas capacidades de interacción para los objetos dentro del entorno y el/los usuario(s). También añade un factor fundamental, para lograr una interacción compleja, que es el reconocimiento de las acciones ejecutadas por el usuario, de esta manera el EV3D puede adaptarse a su comportamiento en base a esas acciones. (Xin Bai, 2013)
- **Una nueva alternativa en la representación del conocimiento.** Actualmente, el dibujado y representación de entornos virtuales en ordenador, ha cambiado de manera significativa, con la aparición de nuevas y potentes herramientas, además de esto, también evoluciono en su estructura pues ahora el modelo basado en grafos de escena permite mayor control en el desarrollo de EV. Más aun, con la actual demanda de integrar personajes inteligentes al entorno, lo cual implica el

⁷ No tiene intervalo de tiempo constante entre cada evento, específicamente en un sistema es la característica en la que el usuario/receptor recibe una respuesta por cada envío de datos que ejecuta de forma no constante (IPN, Upiicsa/Fundamentos de la Computación-Unidad II).

mantenimiento de la representación simbólica (entorno), en el cual el/los agente(s) o personaje(s) puedan retroalimentar sus propios conocimientos. (Xin Bai, 2013)

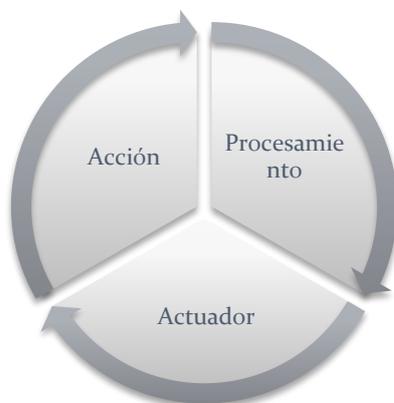


Fig. 3.6 Ciclo de comportamiento de un personaje inteligente. Elaboración propia.

En los personajes dentro de un entorno, la información que maneja los agentes de simulación abarca distintos niveles de representación. El nivel básico recolecta la información geométrica acerca del entorno (tamaños, posiciones, direcciones, colores, propiedad que sean visibles de objetos y actores 3D). Mientras tanto, la representación simbólica del entorno permitirá a los personajes y/o agentes resolver de manera más óptima los inconvenientes que se llegaran a suscitar en el EV3D. Este tipo de representación ayudara en la programación de soluciones a muchas limitantes en lo que respecta a la interactividad con el usuario. (Xin Bai, 2013)

- **Otra opción para efectuar la simulación física.** En general, los comportamientos dinámicos adjuntos a objetos y/o personajes son descritos en base a algún modelo físico, capaz de probar el control de este en tiempo real. Aunque, por otro lado, la causalidad⁸ de las acciones producidas en este tipo de entornos, sugiere la revisión de modelados de comportamiento simbólico, los cuales son capaces de controlar esta variabilidad comportamental en distintas situaciones. De esta manera, las máquinas de estados, sistemas de reglas, bases de hechos, y otras características reactivas, pueden ser incluidos para el control de los niveles de libertad de distintos mecanismos del entorno, por ejemplo: ventanas, puertas, muebles, luces, escaleras, etc. (Xin Bai, 2013)

⁸ Relación de causa-efecto. Real Academia Española. También descrita como el inicio u origen de algo. Este principio se utiliza para nombrar a la relación entre una causa y su efecto, el cual se utiliza en el ámbito de la física, estadística y filosofía.

3.1.3 SIMULACIÓN COMPORTAMENTAL INTELIGENTE EN EV

Dentro de los Entornos Virtuales Inteligentes, es de gran importancia la técnica implementada para el modelo comportamental y cinemático de el/los personaje(s), quien determinara la manera en que el usuario interactúa con el entorno y sus componentes, como ya se ha mencionado anteriormente. Estas técnicas han permitido la expansión en el mercado de hardware y software especializado, con los que el usuario es capaz de tener mayor interacción con entornos cada vez más complejos. Así mismo, el área de ingeniería que trata de simular la inteligencia (IA) se ha convertido en un importante referente al proporcionar técnicas que brinden al usuario entidades y/o personajes que simulen un comportamiento inteligente dentro de entornos virtuales diversos en los que se quieran involucrar, permitiendo así, la interacción (conversar, empatizar, etc) con personajes dentro de la escena. (Giuseppe De Giacomo, 2014)

La creación de EVI3D tiene una relación muy estrecha con la IA, como ya se ha visto, sobre todo para la generación de aplicaciones de entretenimiento, pues es cada vez más común la demanda, por parte de los usuarios, de vivir experiencias nuevas mediante la interacción con entidades inteligentes. Precisamente, aquí reside el principal reto con el cual se ha enfrentado la generación de nuevas técnicas en IA y nuevas tecnologías, y es encontrar adversarios/compañeros inteligentes. Al hablar de inteligencia en estas entidades se hace referencia a su capacidad de responder en defensa o como lo requiera la situación, simular un comportamiento real, un ser con virtudes, capaz de improvisar en base a lo aprendido, que posea debilidades, entre otras características que lo dotarían de inteligencia. (Giuseppe De Giacomo, 2014)

Actualmente, la IA y sus nuevos paradigmas han hecho posible encontrar en diversas aplicaciones de videojuegos, entidades virtuales fuertes, débiles, que son capaces de conspirar, planear, anticiparse, equivocarse, coordinarse, interactuar entre ellos mismos y con el usuario. Definitivamente estas técnicas de IA que se dedican a concebir un comportamiento inteligente en estas aplicaciones, han marcado una pauta en la búsqueda de olvidar que se trata de una “simulación”, intensificando la parte interactiva para lograr la sensación real, omitiendo que lo que se tiene enfrente es software.

A continuación se hará un breve repaso de las principales áreas de desarrollo que han surgido gracias a la IA para ser implementadas en entornos virtuales interactivos, desarrollados principalmente como videojuegos de alta interactividad.

3.1.3.1 Búsqueda Heurística

Esta técnica consiste en un algoritmo de búsqueda el cual trabaja en base a una representación de ramificaciones en forma de árbol, es decir, la heurística es formalizada como reglas para elegir (buscar) las “ramas” entre los posibles estados, por las cuales, es más probable de llegar a una solución óptima para el problema planteado. La búsqueda heurística es empleada cuando un problema puede no contar con una solución exacta dado que se presenten ambigüedades en la información disponible respecto al problema, otro

tipo de problema común se refiere a cuando se tiene una solución exacta pero el costo computacional que requiere es excesivo. (Hollinger, 2016-2017)

Para lograr encontrar tal camino entre las opciones disponibles se crea una función de evaluación $g(n)$, en función al tipo de búsqueda se obtendrán ciertos parámetros. Las principales características de este tipo de búsqueda son: (Hollinger, 2016-2017)

- Excluyen estados no prometedores y sus ramificaciones.
- Encuentras la soluciones más óptimas (el camino más corto)
- Dada la limitación de información disponible en ocasiones las reglas creadas pueden ser inexactas
- Aborda la complejidad del problema guiando la búsqueda por los caminos más competentes dentro del espacio de búsqueda

En la simulación de un comportamiento inteligente es comúnmente utilizado el algoritmo de **búsqueda heurística A***. La idea general de este tipo de algoritmo se trata de encontrar el camino más corto entre un punto “A” a un punto “B”, ubica la primera “mejor búsqueda” con la evaluación de la función dada. Crea una cola de nodos de búsqueda ordenados por prioridad inicialmente es el “estado inicial”, la prioridad es definida por la función $f(n)$. (Hollinger, 2016-2017)

mientras cola!= o & !=objetivo:

obtener el mejor estado x de la cola

si x!=objetivo

entonces genera los n hijos de x

& guarda ruta con cada nodo

aplica $f(n)$ a cada nuevo nodo y

se agrega a la cola

elimina duplicaos en la cola evaluación

con la $f(n)$ para elegir el mejor.

Fin si

Fin mientras

Tabla 3.2 Seudocódigo de algoritmo de búsqueda heurística A. Elaboración propia.*

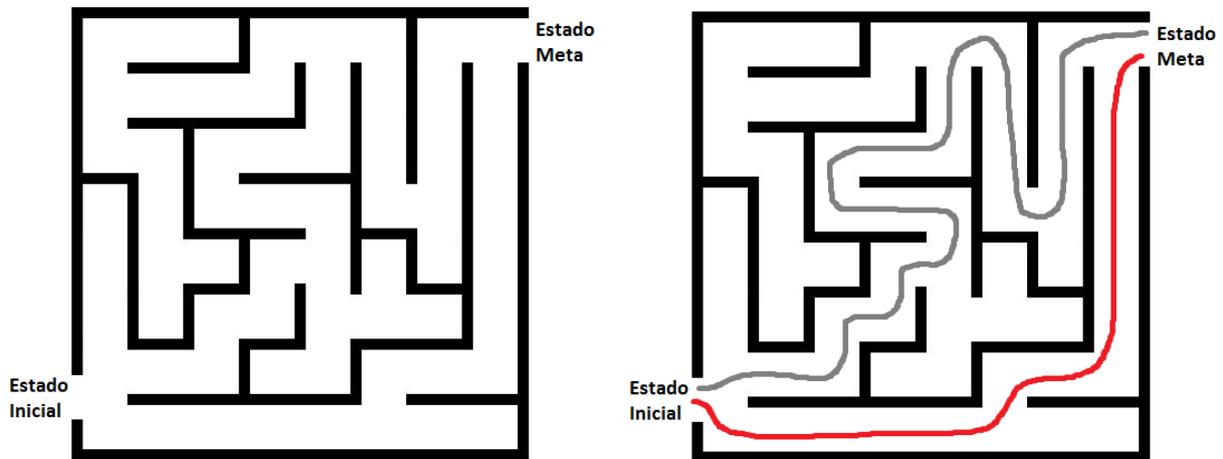


Fig. 3.7 Representación de búsqueda heurística A*. Elaboración propia.

Una entidad virtual que interacciona con su entorno debe desplazarse en él, un ejemplo claro es en un videojuego donde el personaje se encuentra en un barco, debe llegar a los botes salvavidas que se encuentran en la cubierta antes de que este se hunda y muera ahogado, él se encuentra en una habitación en la planta baja, puede llegar subir por el elevador, escaleras, o recorriendo los pasillos. El algoritmo heurístico A* resulta una manera eficiente de determinar que camino es el más conveniente basándose en las características de la circunstancias. (R. Mora, 2014)

Este tipo de técnica de IA, como ya se ha visto, es ideal cuando se tiene un objetivo específico dentro de un entorno virtual, dado los objetivos del presente proyecto, esta técnica es inadecuada para cumplir el objetivo.

3.1.3.2 Búsqueda Local

Los algoritmos de búsqueda local trabajan sobre un espacio de posibles soluciones, entre estas, siempre se trata de conseguir la mejor. Este tipo de búsqueda está caracterizado por empezar con una configuración inicial, habitualmente aleatoria, y se generan cambios pequeños a través de operadores hasta obtener un estado inmejorable. (Fredy Martínez, 2016-2017)

Las técnicas implementadas por este tipo de búsqueda tienen tendencia a encontrar “óptimos locales” que no representan la mejor solución. La solución óptima global es, en general, imposible de alcanzarse en un tiempo delimitado, esto se debe al tamaño del espacio de soluciones posibles. Los métodos utilizados en búsqueda local, se conocen como meta-heurísticas u optimización local.

Su comportamiento está definido por una función heurística que evalúa la calidad de la solución encontrada, sin embargo, esta no está sujeta a un coste. Esta función se implementa para descartar las soluciones que no tienen viabilidad, dentro del espacio de

búsqueda. Un factor importante involucrado, es la falta total de una memoria, pues no suele guarda el camino recorrido esto implica un gasto de memoria mínimo, sin embargo, esto también supone problemas en el algoritmo, bucles.

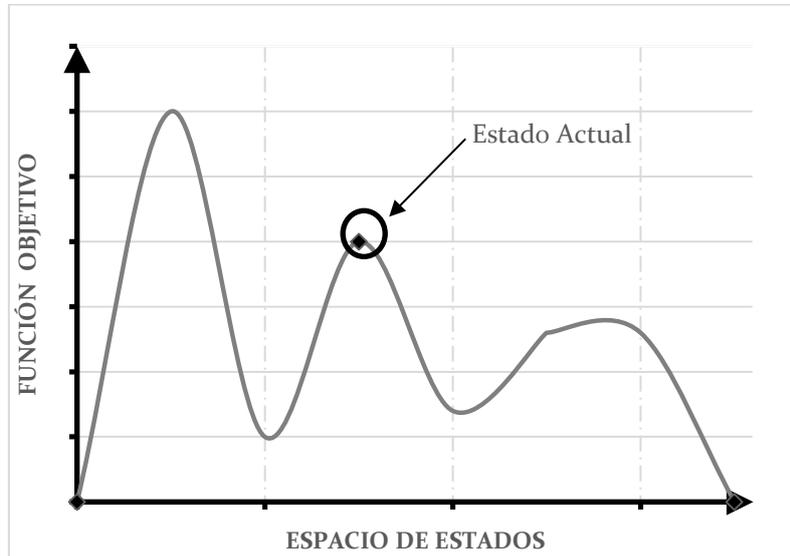


Fig. 3.8 Representación de Búsqueda Local. Elaboración Propia.

La Fig. 3.8, representa la búsqueda local por ascensión de colinas, una de las técnicas más implementadas en este tipo de búsquedas. Las características de esta función heurística definen la calidad del resultado y su rapidez de búsqueda.

Inconvenientes	Solución
<p>Máximo local. Determina que todos los vecinos tienen una función heurística inferior</p> <p>Meseta. Todos los vecinos poseen la misma función heurística que el nodo actual.</p> <p>Crestas. Estas generan una secuencia de “máximos locales”, lo cual dificulta la navegación para otro tipo de algoritmos.</p>	<p>Se vuelve a un nodo anterior y se sigue con el proceso en otra dirección.</p> <p>Reinicia la búsqueda en otro punto.</p> <p>Se aplican dos o más operadores previamente a la selección del camino a seguir.</p> <p>Se puede dividir el espacio de búsqueda en distintas regiones y examinar las más competentes.</p>

Tabla 3.3 Inconvenientes y posibles soluciones encontradas en el algoritmo de búsqueda local.

Este tipo de algoritmos de búsqueda son muy populares en la simulación de comportamientos en entornos interactivos, aunque es importante destacar que la solución encontrada por este tipo de algoritmos será la mejor que pueda obtener entre un número limitado de posibles soluciones, ya que la búsqueda local no es capaz de encontrar el óptimo y calidad de dicho resultado, pues este será el mejor en función de los operadores que haya diseñado el programador y la función de evaluación, así mismo, el tiempo que implica obtener el resultado. Por lo cual, se determina que este tipo de algoritmo no cumple con las características para crear las características necesarias para el enfoque

comportamental e instruccional, requerido en el desarrollo del presente trabajo, ya que aplicado en el entorno virtual, derivaría en algunas limitantes en la interacción con el usuario y un aumento en costos computacionales.

3.1.3.3 Agentes Inteligentes

Los agentes inteligentes surgen a partir de la necesidad de obtener un método capaz de hacer frente a cualquier problema que pueda ser resuelto mediante la IA. De manera simple, podrían ser descritos como “entidades” que dividen las tareas requeridas para satisfacer determinado objetivo. (Serkan Dincer, 2015)

A través de los sistemas basados en agentes inteligentes se pueden abordar diferentes áreas de esta y relacionándolas entre sí, de esta manera, la búsqueda, representación y solución de problemas, el aprendizaje, el razonamiento, y la IA distribuida pueden ser enfrentadas bajo el mismo paradigma. (Arsalan Heydarian, 2015)

El funcionamiento de los agentes puede ser muy básico hasta un nivel de alta complejidad, dependiendo de los modelos y enfoque de desarrollo. Estos trabajan de forma independiente, sin embargo en un **Sistema Multiagente**, se deben de comunicar entre sí, para abordar cualquier problema intercambiando información. Un agente estará en constante búsqueda de la información necesaria para llevar a cabo su función, lo cual se logra mediante su capacidad de contemplar y reaccionar ante su entorno. (Arsalan Heydarian, 2015)

En el desarrollo de un agente inteligente implementado en un sistema de realidad virtual se tiene que tomar en cuenta algunas características importantes: (Anu Sivunen, 2014)

- **Inteligencia.** Se trata del grado de razonamiento y conducta aprendida, es decir, la habilidad del agente para aceptar las decisiones del usuario y realizar objetivo específicos. Es la convergencia del razonamiento, aprendizaje, planificación y respuesta hacia el entorno que lo rodea.
- **Agencia.** Es el grado de autonomía y autoridad determinada en el agente. La cual será representada por el grado de interacción con servicios, datos, objetos, con el entorno en general.
- **Movilidad.** Dentro de los Sistemas Multiagente, estos aparte de aprender y adaptarse a su entorno en término y objetivos del usuario, también los agentes viajan por la red para comunicarse.

En base a información recopilada acerca de las diferentes áreas de la IA que se dedican a desarrollar métodos/técnicas para implementarse en este tipo de sistemas de realidad virtual, se ha elaborado una tabla de análisis comparativa que pretender resaltar las propiedades más importantes para este trabajo.

Propiedad	Búsqueda Heurística	Búsqueda local	Agentes Inteligentes
Interactividad	Alta	Baja	Alta
Adaptabilidad	Media	Media	Alta
Optimalidad	Alta	Baja	Alta
Ligereza de procesamiento	Media	Baja	Media

*Tabla 3.4 Tabla comparativa de métodos de simulación comportamental inteligente en EV.
Elaboración Propia.*

Los Agentes Inteligentes, área de estudio de IA, han brindado las características necesarias para cumplir con el objetivo del sistema en realidad virtual planteado en el presente proyecto. Ya que es posible enfocar la construcción de un agente inteligente al modelo comportamental e instruccional que representara al usuario en el entorno virtual. En este contexto, se identificarán las técnicas, modelos y/o estructuras necesarias profundizando en el tema a continuación

3.1.4 AGENTES INTELIGENTES EN ENTORNOS VIRTUALES

3.1.4.1 Introducción

Las nuevas tecnologías aplicadas al patrimonio cultural ofrecen diferentes formas de acercamiento. Desde perspectivas históricas o arqueológicas, como por ejemplo la exploración de yacimientos, hasta perspectivas de ocio y entretenimiento, en las que el usuario participa como espectador, o siendo parte de la aplicación, creando una sensación de presencia en diferentes niveles. (Josu Ahedo Ruiz, 2013)

Son varios los modelos de programación que coexisten en la actualidad. Todos con diferentes formas de visualizar el mundo y lo que se percibe como realidad, con el fin de desarrollar instrucciones para las computadoras.

Los lenguajes de programación se pueden clasificar en diferentes tipos de acuerdo a su funcionalidad. Los modelos de programación más relevantes son: (Das, 2014)

- Estructurado
- Funcional
- Lógico
- Orientado a objetos
- Orientado a servicios
- Orientado a agentes
- Basado en restricciones

De los últimos avances reconocidos en el desarrollo de sistemas inteligentes, se encuentra los software's adaptativos, que están basados en la programación de agentes y se utilizan generalmente en Inteligencia Artificial. (Coca Bergolla, 2009)

Una de las herramientas que se utilizara en el actual trabajo para diseñar el sistema objeto de estudio se encuentra precisamente el paradigma orientado a agentes. Estos, como menciona (Gonzalo Luzardo, 2014), son piezas que se utilizan para desarrollar software de alta complejidad, su funcionamiento es muy similar a la programación orientada a objetos. Contextualizando un poco esta definición, dentro de un entorno virtual, (independientemente de su clase) incluirá un personaje virtual, el cual, dentro de este paradigma, deberá ser autónomo y formalmente quedara entendido bajo el concepto de "agente".

Es decir, un "sistema" de control con capacidad de toma de decisiones (reactivo, deliberativa, etc.), adaptabilidad, y respuesta en su entorno. De esta manera, los EV3D brindan grandes posibilidades para la creación de sistemas de realidad virtual inteligentes.

Sin embargo, dadas las especificaciones del trabajo, se utilizara un tipo de agente en específico, dentro del desarrollo del sistema existirá un apartado que se encargara de la generación de este, como ya se mencionó anteriormente, el enfoque será comportamiento del controlador que representa al usuario, así, de esta manera, brindar al usuario una experiencia más satisfactoria.

Con el desarrollo de un sistema basado en la utilización de un **agente inteligente reactivo**, se contempla apoyar la divulgación de espacios culturales, los departamentos que conforman dicho inmueble sin limitarse solo en la exposición de instalaciones y estructura, sino también información general de talleres y proyectos de investigación.

A continuación se exponen algunos conceptos del paradigma de la programación orientación a agentes que se han considerado más importantes para el presente trabajo, para comenzar con un análisis del proyecto.

3.1.4.2 ¿Qué es un Agente inteligente?

Hasta principios de los años noventa del siglo pasado, los sistemas inteligentes solían ser programas aislados, orientados a resolver problemas muy concretos. La complejidad residía principalmente en el algoritmo y no tanto en la arquitectura del sistema. Sin embargo, en la actualidad la complejidad está en la diversidad de componentes que forman los sistemas. Tal es el caso de los **Agentes Inteligentes (AI)**, los cuales, proporcionan servicios, que se pueden combinar de múltiples maneras para crear nuevos servicios más versátiles, que muchas veces se pueden personalizar para cada usuario. (José T. Palma Méndez, 2008)

Propiamente no hay una definición universalmente aceptada. Ya que muchos investigadores del tema, tienden a coincidir que los agentes son entidades software con autonomía, pero en las características y propiedades, cada uno tiene diferente punto de vista en cuanto cuales son las más imprescindibles. Entre las acepciones más utilizadas se encuentran:

Un agente es cualquier cosa capaz de percibir su medio ambiente con la ayuda de sensores y actuar en ese medio utilizando actuadores. (Stuart J. Russell, 2004)

Smith (1994) señalo como definición que un agente es una entidad de software persistente, enfocado a un propósito específico. Los agentes tienen sus propias instrucciones para realizar las tareas. Tiene un objetivo puntual que hace que se distingan de todo Aplicaciones multifuncionales; Los agentes son típicamente mucho más pequeños. (Pozo, 2008)

Wooldridge y Jennings (1995) menciona que un agente inteligente es una entidad software que, basándose en su propio conocimiento, realiza un conjunto de operaciones para satisfacer las necesidades de un usuario o de otro programa, bien por iniciativa propia o porque alguno éstos se lo requiere. (Gonzalo Luzardo, 2014)

Garijo (2002) menciona, en una definición mucho más completa, que el concepto de agente caracteriza a una entidad software con una arquitectura robusta y adaptable que puede funcionar en distintos entornos o plataformas computacionales y es capaz de realizar de forma autónoma tareas específicas según su función, intercambiando información con otros agentes humanos o computacionales. Las principales características de un agente son:

- Funcionamiento continuo y autónomo
- Comunicación con el entorno y con otros agentes (posiblemente humanos), por medio de un lenguaje o formalismo de comunicación
- Robustez
- Adaptabilidad como capacidad de realizar objetivos y tareas en distintos dominios de forma incremental y flexible (Pozo, 2008).

Actualmente, las definiciones de lo que es un agente no han variado mucho desde que se formularon. Partiendo de los axiomas anteriores y algunos otros que se encuentran en la literatura podemos enfatizar algunas propiedades que caracterizan a los agentes:

- **Reactividad.** Los agentes pueden percibir su entorno (p. ej. el mundo físico, un usuario detrás de una interfaz gráfica, aplicaciones en la red, u otros agentes) y responder oportunamente a cambios que se produzcan en el mismo
- **Autonomía.** Los agentes pueden trabajar sin la intervención directa del usuario y tienen cierto control sobre sus acciones y estado interno

- **Interactividad.** Para cumplir con sus objetivos un agente puede necesitar la colaboración de otro agente en quienes delega o comparte la realización de las tareas.
- **Iniciativa.** El comportamiento de los agentes lo determina los objetivos que se pretendan alcanzar, por lo tanto, pueden ejecutar acciones no solo como respuesta al entorno

Para definir la arquitectura interna de los agentes se debe tener en cuenta principalmente dos aspectos importantes: autonomía y reactividad, que aparecen en todos los modelos de agente. De esta manera, en una primera aproximación, todo agente es una entidad que percibe eventos de su entorno y que puede actuar sobre él.

El agente tiene una visión parcial del entorno, limitada por sus sensores, que le permiten obtener información sobre determinados eventos. En el caso de un Agente Web, para los sensores y actuadores, tendrá que acceder a servicios web utilizando determinados protocolos de comunicación.

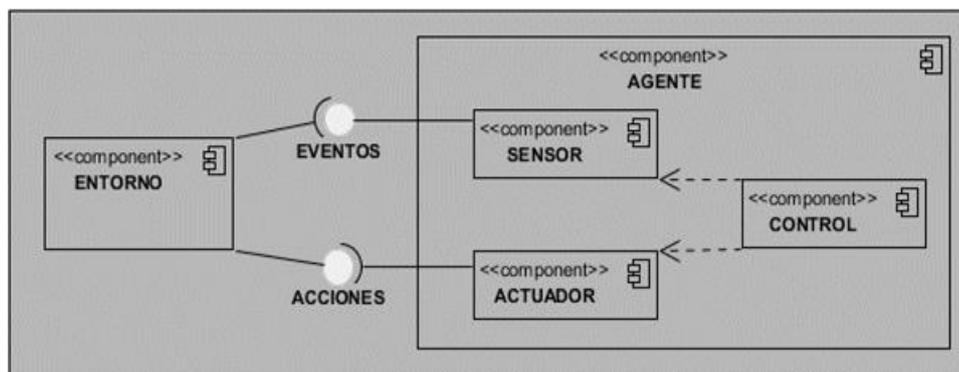


Fig. 3.9 *Arquitectura General de un Agente.* (Adaptación de *Inteligencia Artificial* por José T. Palma)

Wooldridge (1995) señala que esta percepción y actuación limitada establece que la visión del mundo que tiene un agente es parcial, por lo tanto, existe un cierto grado de incertidumbre. El agente tiene que decidir qué acciones tomar teniendo en cuenta el modelo del mundo que va construyéndose a partir de la información que consigue por los sensores. Así la arquitectura del agente, especialmente el componente de control, es una arquitectura software de un sistema de toma de decisiones. (Gonzalo Luzardo, 2014)

Russel y Norvig (2003), aseguran, por tanto, lo importante que es tener en cuenta que un agente puede fallar, ya sea en la acción que decide tomar o en el momento en que decide tomarla. Un agente debe utilizar la información que obtiene al observar el entorno cuando ha ejecutado alguna acción, para evaluar si se logró el objetivo esperado, y con ello, intentar mejorar en sus decisiones futuras.

La estructura del componente que controla los sensores y actuadores, maneja y procesa el modelo del mundo, es muy importante para un mejor funcionamiento, para

ello, se han propuesto algunas arquitecturas que se dividen tres principales tipos: deliberativas, reactivas e híbridas. (José T. Palma Méndez, 2008)

3.1.4.2.1 Agentes de Interfaz

Los agentes de interfaz son frecuentemente utilizados en el desarrollo de sistemas con implicaciones cognitivas. Un agente de interfaz, según (Lincicum, 2003) y (Serenko, 2006), puede definirse como una entidad visual reactiva, colaborativa, y autónoma asistido por una serie de procesos que hace posible la comunicación directa con el usuario de una forma atractiva socialmente. Los agentes de interfaz fungen como intermediarios entre el usuario y las tareas disponibles con la computadora, su principal función consiste en facilitar la interacción usuario-máquina y que esta sea una experiencia más agradable. Por tal motivo, para lograr alcanzar su objetivo, este tipo de agentes tiene muy presentes ciertos aspectos del usuario y la toma de decisiones en el sistema, para crear una relación tal que no sea perceptible (en medida de la complejidad implementada) que se está comunicando con una máquina. (Mabanza, 2016)

Existen variadas aplicaciones para los agentes de interfaz y también se clasifican respecto a la implementación de estos, por ejemplo, se pueden utilizar en medicina, entretenimiento, educación, etc. Cuando se implementan en la educación se despliega una categoría conocida como agentes educativos, cuyas propiedades se adaptan a las tareas especializadas en esta rama. (Mabanza, 2016)

A continuación se dan algunos ejemplos reales de las aplicaciones de los agentes de interfaz:

- **Comercio electrónico.** Un agente puede fungir como un asistente de comercio que proporcionaría asesoramiento sobre algún producto.
- **Educación.** El agente se encarga de asesorar al usuario para el cumplimiento de alguna tarea educativa/pedagógica.
- **Entretenimiento.** El agente puede participar como un personaje autónomo dentro de un videojuego.

En el presente trabajo, se realizan todas las actividades disponibles en el sistema mediante un solo agente inteligente, que en cooperación con otros eventos y procesos del entorno hacen posible un grado de interactividad básico, sin embargo, un agente de interfaz podría dotar al sistema de un alto nivel de complejidad influenciando los procesos para un funcionamiento más óptimo en el cumplimiento de sus objetivos.

Retomando los tipos de agentes de interfaz, según la literatura encontrada, son conocidos de diferente manera, por ejemplo agente de conversación, agente encarnado, agente de interfaz animado, sin embargo al hacer referencia al mismo tipo de agente se le conoce formalmente, en la comunidad científica, como agente de interfaz. Lo cual podría variar dependiendo sus características y objetivos, no obstante, entran en la misma categoría de agentes.

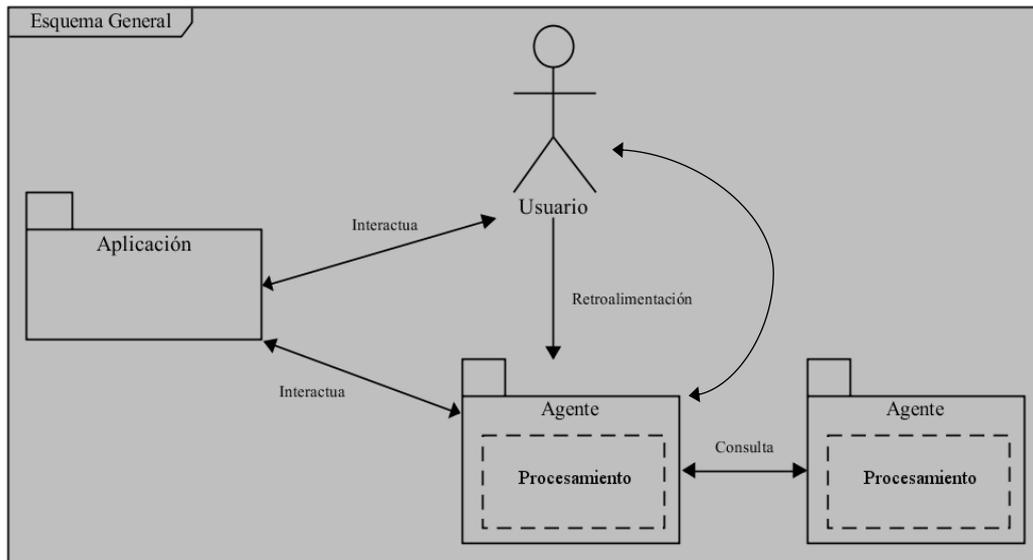


Tabla 3.5 Esquema General de funcionamiento de un Agente de Interfaz. Elaboración propia.

Los agentes de interfaz suelen enfatizar la adaptabilidad y autonomía para así cumplir con el objetivo del usuario. En otras palabras, estos agentes prestan apoyo al usuario que está aprendiendo una nueva actividad o información específica. Algunas técnicas mediante las cuales el agente aprende son:

- Mediante la observación e imitación del usuario y entorno.
- Con la retroalimentación positiva o negativa del usuario y entorno.
- Recibiendo instrucciones explícitas del usuario.
- Colaborando con otros agentes.

Es así como el agente interactúa con el usuario, convirtiéndose en una clase de asistente personal para él, y de esta manera, lo ayuda a cumplir con tareas específicas.

3.1.4.2.2 Agentes Deliberativos

Las arquitecturas deliberativas utilizan modelos de representación simbólica del conocimiento y suelen basarse en la teoría clásica de planificación. Los agentes tienen un conjunto de objetivos y, dependiendo del estado del mundo, generan planes para alcanzar los objetivos.

Una vez elegida este tipo de arquitectura, el primer paso a efectuar, es una descripción simbólica adecuada del problema, para que esta pueda ser entendida por el agente y pueda llevar a cabo las tareas signadas en el tiempo preestablecido. Debido a la complejidad de los algoritmos de manipulación simbólica, es un aspecto muy importante, en especial si se tiene en cuenta que los agentes se desenvuelven en entorno reales, en los que frecuentemente tienen que responder a los estímulos en tiempo real.

(Rao y Georgeff, 1991) mencionan a la BDI (Belief, Desire, Intention), es la arquitectura deliberativa más extendida, y se caracteriza por el hecho de que los

agentes que la implementan están dotados cada uno de un estado mental. En este sentido, el estado mental es un conjunto de creencias, deseos e intenciones.

De manera general, esta arquitectura define que las creencias representan el conocimiento que el agente tiene del entorno, los deseos son los objetivos que persigue el agente, en otras palabras, su percepción ideal del mundo, por lo tanto, para lograr los objetivos establecidos a partir de las creencias existentes es necesario definir un mecanismo de planificación que permita identificar las intenciones.

Dentro de esta arquitectura hay dos factores que se deben contemplar, los cuales son la gestión y el control del estado mental.

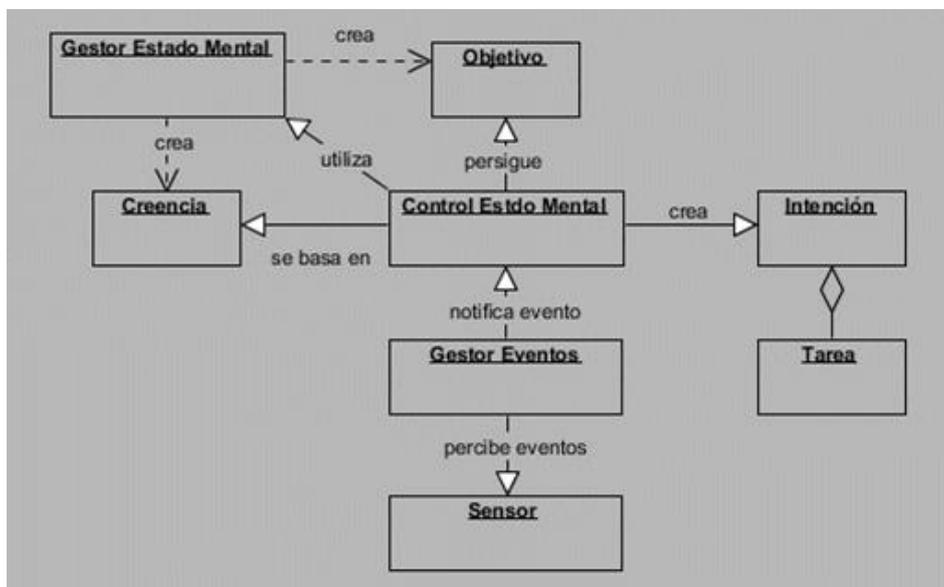


Fig. 3.10 Diagrama de Control dentro de un Agente BDI. (Adaptación de Inteligencia Artificial por José T. Palma)

Tal como muestra la Fig. 3.10, los eventos del entorno, capturados por los sensores, son notificados al control de estado mental que decidirá cómo afecta al gestor de estado mental.

➤ **Gestor de Estado Mental**

Proporciona los mecanismos para crear, modificar y destruir entidades del estado mental, manteniendo su consistencia.

➤ **Control de Estado Mental**

Encapsula el mecanismo de decisión y genera las intenciones del agente: planes o secuencias de tareas que el agente intentara ejecutar para lograr satisfacer sus objetivos.

3.1.4.2.3 Agentes Reactivos

La utilización de una representación simbólica y mecanismos de toma de decisiones puede resultar costosa en tiempo y recursos, por lo cual algunos investigadores han

considerado necesario buscar modelos más efectivos de representación del conocimiento y de comportamiento para los agentes. Esta asunción puede ser válida en situaciones donde las acciones del agente se producen como respuesta directa a los eventos del entorno, sin requerir un proceso deliberativo complejo. En estos casos, la inteligencia surgiría por la interacción de múltiples agentes, aunque cada uno tuviera un comportamiento muy sencillo.

Los Agentes Reactivos se basan en un ciclo de percepción y acción (estímulo-respuesta), reaccionan a la evolución del entorno, del cual no hay una representación explícita así como de otros agentes y sus capacidades. (José T. Palma Méndez, 2008)

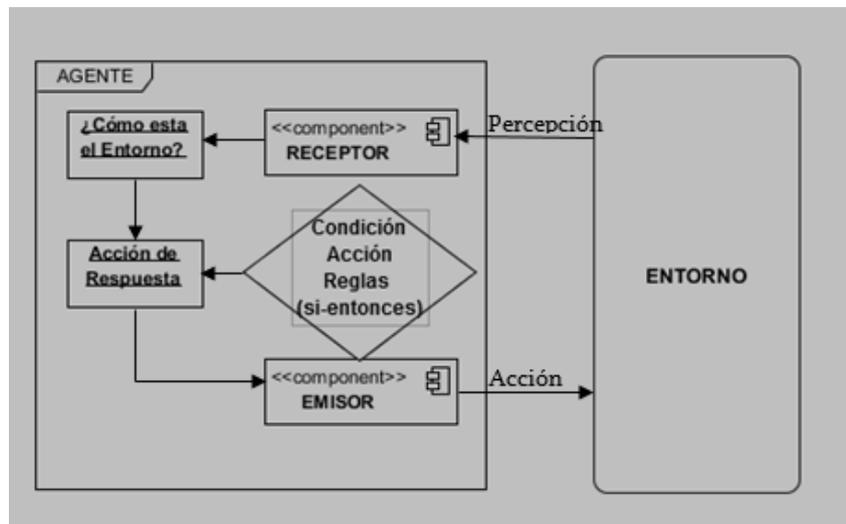


Fig. 3.11 Diagrama de Control de un Agente Reactivo (Adaptación de *Inteligencia Artificial* por José T. Palma)

Los AR deciden la acción a realizar teniendo en cuenta su historia de interacciones con el entorno. Crean una secuencia de estados del entorno o secuencia de percepciones y se guarda como estado interno del agente.

El ciclo de ejecución de un AR bien puede ser representado por un ciclo que repite una acción mientras se cumpla la condición asignada. En la ejecución hay dos factores que deben ser considerados los cuales son las reglas (situación-acción) y un conjunto de percepciones:

```

Mientras(verdadero){
    Estado=interpretar_entrada (percepción);
    Regla=correspondencia (estado/reglas);
    Ejecutar(regla-acción);
Fin Mientras
}

```

Tabla 3.6 Ciclo de ejecución de un Agente Reactivo. Elaboración propia.

Uno de los ejemplos más representativos en este tipo de arquitecturas, es la propuesta de (Brooks, 1991) con su arquitectura de subsunción, la cual asevera que no es necesario que los agentes tengan una representación simbólica y que pueden razonar sobre ella para que el conjunto del SMA (Sistema Multiagente) pueda ofrecer un comportamiento inteligente, sino que el comportamiento inteligente pueda surgir de la interacción entre los agentes.

En una arquitectura de subsunción las tareas que definen un comportamiento están organizadas en jerarquías de capas, de menor a mayor nivel de abstracción. Se trata de módulos de comportamiento que realizan tareas, carece de representación y razonamiento simbólico. El comportamiento se puede implementar como un conjunto de reglas/acción donde la “situación” se toma directamente de la percepción, y como ya se mencionó anteriormente, no sufre ninguna alteración a representaciones simbólicas. (Lujak, 2015)

Jerarquía de subsunción

- Se pueden ejecutar varios comportamientos simultáneamente
- Para elegir entre ellos se usa la jerarquía de subsunción
- Los comportamientos están ordenados por capas
- Los comportamientos de las capas más bajas (mayor prioridad) inhiben a los de las capas superiores



Fig. 3.12 Arquitectura de Subsunción. (Adaptación de Inteligencia Artificial por José T. Palma)

Como se puede visualizar, los comportamientos son simples y se pueden implementar como autómatas sencillos o como reglas de tipo situación-acción, que asocian acciones a la aparición de determinados eventos de entrada (Fig. 3.12).

3.1.4.2.4 Agentes Híbridos

En un esfuerzo por satisfacer las limitaciones de las arquitecturas puramente reactivas y deliberativas se han propuesto sistemas híbridos que pretenden mezclar los mejores

aspectos de ambos modelos (José T. Palma Méndez, 2008). Una de las primeras propuestas consiste en construir un agente compuesto de dos subsistemas: uno deliberativo, que utilice un modelo simbólico y que genere “planes de ejecución”, y otro reactivo, que reacciona ante los eventos que tengan lugar en el entorno, que no requiera un mecanismo de razonamiento complejo. Estos subsistemas se pueden utilizar para definir una estructuración en dos ejes:

- **Vertical:** solo una capa tiene acceso a los sensores y actuadores
- **Horizontal:** todas las capas tienen acceso a los sensores y a los actuadores

Se trata de capas que se organizan jerárquicamente con información sobre el entorno a diferentes niveles de abstracción. Generalmente, es este tipo de arquitecturas, se contemplan tres niveles principalmente:

- **Reactivo:** Es el considerado de “más bajo nivel”, y es el encargado de tomar decisiones en respuesta de los eventos recibidos del entorno donde se ejecutan. Suele implementarse en arquitecturas de subsunción.
- **Conocimiento:** Es el nivel intermedio, trata el conocimiento que el agente posee del medio, generalmente, con la ayuda de una representación simbólica del mismo.
- **Social:** Capa de alto nivel, maneja los aspectos sociales del entorno, tanto de la información de otros agentes, así como sus deseos, intenciones, entre otras características.

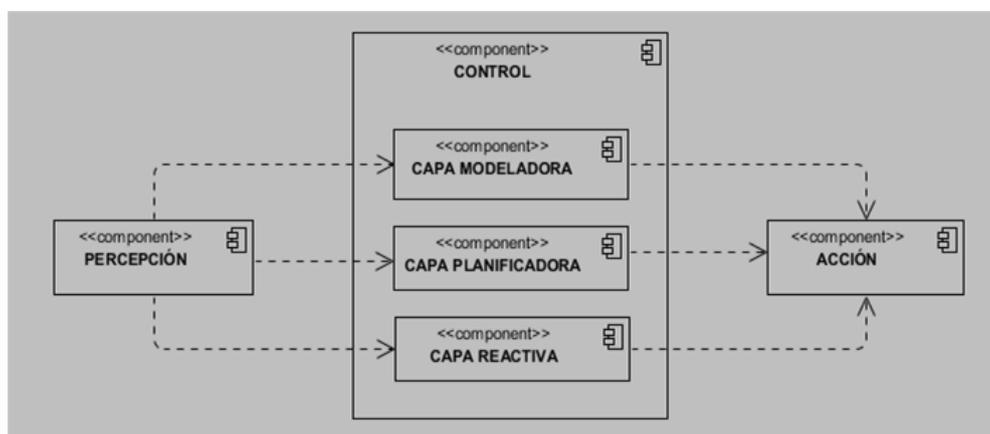


Fig. 3.13 Arquitectura Híbrida con Tres Capas. (Adaptación de Inteligencia Artificial por José T. Palma)

El comportamiento general del agente está definido por la interacción entre los niveles mencionados, sin embargo, esta interacción cambia de una arquitectura a otra. Tal como se puede apreciar en la Fig. 3.13, donde se muestra un tipo de arquitectura que entra en el tipo de arquitecturas denominadas TouringMachine, según (Ferguson, 1992). Se habla de una arquitectura compuesta por tres capas horizontales, las cuales son:

- **Capa Reactiva:** (respuesta inmediata a cambios dentro del entorno, como reglas situación-acción, o entrada de senso-salida de actuador)
- **Capa Planificadora:** (comportamiento proactivo basado en esquemas de planes)
- **Capa Modeladora:** (modelo del mundo para anticipar conflictos y generar nuevos objetivos para la capa planificadora)

Estas capas están integradas en un subsistema de control basado en reglas que definen si hay que inhibir entradas y salidas, y determinan que capa tiene control sobre el agente.

3.1.4.2.5 Agentes Virtuales Inteligentes

Las herramientas y técnicas para la creación de sistemas gráficos complejos, independientemente del área, han aumentado en auxilio del demandante mercado de estas aplicaciones, tal es el caso de los **Agentes Virtuales Inteligentes (AVI)**, se surgen a partir del estudio de los EVI3D, conforme fueron avanzando este tipo de entornos, los agentes adquirieron nuevas propiedades según las exigencias. Estos están basados en AI que se desenrollan dentro de un EV3D. Generalmente, se les conoce como **Agentes Virtuales Autónomos (AVA)**, surgen a partir del estudio de los EVI3D, conforme fueron avanzando este tipo de entornos, los agentes adquirieron nuevas propiedades según las exigencias. Estos están basados en AI que se desenrollan dentro de un EV3D. Generalmente, se les conoce como **Agentes Virtuales Autónomos (AVA)**. (Kelly S. Hale, 2015)

Como se ha mencionado a lo largo de este capítulo, los AI se pueden ver como “entidades particulares” que actúan en un sistema, en este caso en particular se un EV3D, poseen el control sobre su comportamiento y movimientos dentro del entorno. También, cuentan con la habilidad, como todos los AI, de realizar procesos consecutivamente que les proporcionan datos para conocer cómo y qué objetivos perseguir.

Los AVI cuentan con características representativas de los Entornos Virtuales y los AI, logrando complementarse mutuamente

- Residen en un entorno de ejecución de simulación tridimensional
- Tiene la capacidad de ejecutar sus comportamientos de manera gráfica
- Identifican los cambios que se producen en el entorno y responden en consecuencia
- Tiene una representación gráfica del mundo en el que habitan y pueden adaptarse y reaccionar a lo que los rodea.

3.1.4.3 *Propiedades de los Entornos*

Al igual que existe variedad de enfoques para los entornos virtuales para desarrollar sus características. Existen, también, gran variedad de propiedades que determinan la información que ofrecen a los diferentes módulos de funcionamiento del sistema. Algunas de las propiedades más relevantes, según (Hernández, 2017) en cualquier entorno virtual son:

Totalmente Observable. También conocido como accesible, se caracteriza principalmente por proporcionar acceso al estado completo del entorno continuamente por medio de los sensores, los cuales detectan los aspectos más influyentes para generar una respuesta. Este tipo de entornos brindan la información necesaria para que el agente no requiera tener un estado interno para tener conocimiento de lo que sucede en el entorno.

Parcialmente Observable. Esta propiedad es conocida por brindar poca accesibilidad al entorno, pues debido al ruido y a la inexactitud de los sensores existentes, no hay una buena comunicación con el entorno y el sistema en general. Esto implica, la existencia insuficiente de sensores para poder analizar el entorno completo.

Estocástico. La propiedad que convierte al entorno estocástico se presenta cuando el estado consecutivo de un agente no se puede determinar a partir de la situación actual, por lo cual no es determinista.

Determinista. Esta propiedad es todo lo contrario al anterior mencionado, ya que este se fundamenta cuando el estado consecutivo del entorno, está completamente determinado por la situación actual y la respuesta generada por el agente.

Estratégico. Esta propiedad es una variación del determinista, ya que este también posee esa cualidad determinista, sin embargo no puede efectuarse en las acciones de otros agentes.

Dinámico. Tal como su nombre lo indica, esta propiedad dota al entorno de gran dinamismo pues puede cambiar mientras el agente este procesando una respuesta. Estos módulos dinámicos se encuentran continuamente a la expectativa de las decisiones tomadas por el agente.

Estático. Esta propiedad no permite al entorno cambiar, por lo cual el agente no necesita tener una visión actualizada del entorno y no contempla el tiempo de toma de decisión.

Secuencial. Los entornos secuenciales, son definidos por la influencia que la decisión actual puede tener sobre futuras decisiones. Por esta razón, se debe pensar en las futuras posibilidades para controlar los efectos y aprovechar esta propiedad.

Episódico. El funcionamiento de un agente se puede dividir en módulos que trabajan en diferentes niveles, en el primero mediante la información obtenida realiza una percepción de la situación enfrentada y en el próximo nivel determina una acción. Estas decisiones no influyen a las decisiones futuras.

3.4 PROCESO DE MODELADO 3D

3.4.1 INTRODUCCIÓN

EL modelo geométrico es una parte indispensable en cualquier entorno virtual. El ordenador y las diversas herramientas software que se utilizan en este ámbito permiten generar cualquier tipo de formas, desarrollar cualquier tipo de características superficiales, efectos especiales, etc; con ello se logra expresar cualquier idea y/o concepto de manera gráfica por medio de imágenes 2D y 3D dotadas (generalmente) de movimiento.

Para el desarrollo de un modelo geométrico se aplican diferentes etapas de desarrollo y, en general, los programas de modelado 3D implementan en sus interfaces ciertos conceptos matemáticos básicos, con los cuales el desarrollador debe estar familiarizado, muchos de estos conceptos se mencionaron dentro de los subtemas de *Entornos Virtuales* (sub 3.1.1, 2, 3), sin embargo, no se profundizó mucho en sus significados, por lo cual se explican a continuación.

3.4.2 CONTEXTO MATEMÁTICO

Como ya se mencionó, el proceso de modelado tiene implícito conceptos matemáticos para poder aplicarlos correctamente en el ámbito 3D y así permitir una mejor comunicación entre plataformas de desarrollo, según sean los objetivos del proyecto.

Es necesario tener en cuenta y dominar los fundamentos geométricos, es decir, entender y aplicar conceptos como vector, sistema de coordenadas, normal, polígono, etc. de manera intuitiva, reconociéndolos a lo largo del proceso. A continuación se explican algunos de los conceptos más recurrentes: (Blanco, 2010)

- **Sistema de coordenadas.** Se trata de un conjunto de valores que permiten definir unívocamente la posición de algún punto específico ubicado en el espacio geométrico respecto a un punto inicial conocido como “origen”. En el ámbito 3D, la escena se identifica precisamente por las coordenadas en **3 dimensiones**, es decir, el espacio de trabajo que, comúnmente, es conocido como sistema de coordenadas universal o mundo.
- **Vectores.** Cuando se hace referencia a un vector, se habla de un segmento de recta dirigido dentro del espacio. El cual, posee una serie de características entre ellas están: el punto exacto respecto al cual actúa el vector, llamado **origen**, la longitud del vector es el **módulo**, la orientación en el espacio se conoce como **dirección**, el **sentido** es el indicador representado por la pequeña punta de flecha que se encuentra sobre el vector.
- **Segmento.** Es la parte de una línea recta ubicada entre dos puntos indicados sobre esta. Facilita la localización de partes específicas dentro del espacio.
- **Normal.** Es la superficie representante de un punto concreto dentro del espacio. Es una parte fundamental de cualquier modelo 3D, y se debe tener muy en cuenta,

pues determinara el peso, tiempo de renderización, e incluso la vista final que este pueda tener.

- **Polígono.** Se trata de una figura plana y cerrada conformada por 3 segmentos (o más) unidos entre sí por sus extremos formando vértices y aristas. En la creación de estos polígonos, se contempla la creación de regulares e irregulares.

Una vez que se tengan presentes y entendidas algunas de la definiciones más recurridas en el ámbito 3D, el funcionamiento será mucho más sencillo de asimilar, puesto que la creación de un modelo 3D, dentro de un sistema de RV es fundamental, ya que dentro del archivo que generara, se especificara la información necesaria para ver los objetos creados en 3D, esto mediante los datos generados de la geometría (forma del objeto) y los atributos superficiales del objeto (material, color, etc). (Blanco, 2010)

3.4.3 CICLO DE PRODUCCIÓN 3D

Cuando se trata de un modelo 3D, es posible desarrollarlo de forma estándar para ser implementado como un “subsistema” o de una manera enfocada, por ejemplo un videojuego, una animación, o una película. (González Morcillo, Carlos y Vallejo Fernández, David, 2009), proponen las fases de producción de la siguiente manera utilizando un software específico, sin embargo estas no distan demasiado de las fases implementadas en general, para modelos tridimensionales. Este ciclo de producción se divide en las siguientes fases:

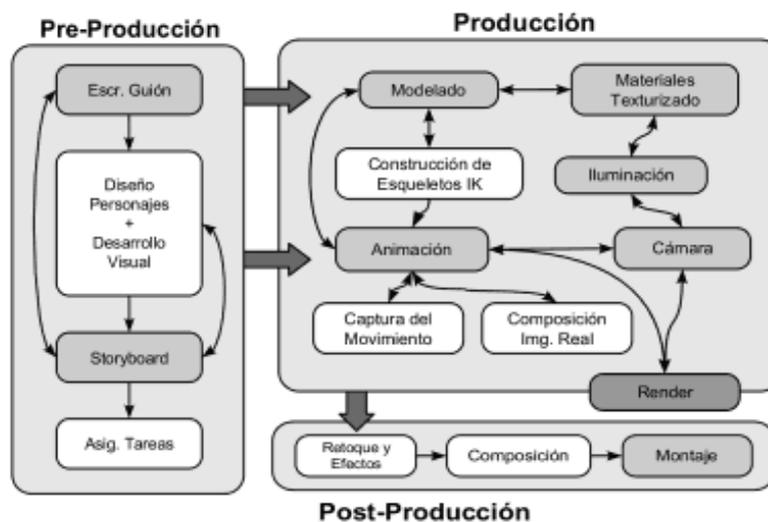


Fig. 3.14 Esquema del Ciclo de producción 3D. (González Morcillo, Carlos & Vallejo Fernández, David, 2009).

3.4.3.1 Pre-Producción

En esta fase comúnmente se inicia con la creación de un guion, ya sea para pequeños o grandes proyectos, siempre debe existir ya que presupone los objetivos del modelo. En muchos de los softwares de modelado existen módulos (storyboard) específicos para el control de transformación visual que derivara el guion creado. Es importante mencionar

que este guion tiene como finalidad proporcionar la información necesaria al o los desarrolladores para determinar la dirección visual y estilo del proyecto. Es aquí donde se seleccionan los colores predominantes en cada una de las escenas creadas, ya sea imitando una estructura existente o simplemente guiándose por los objetivos del guion creado, de igual forma, se crearan los bocetos de personajes que se hayan contemplado incluir. (González Morcillo Carlos, 2009)

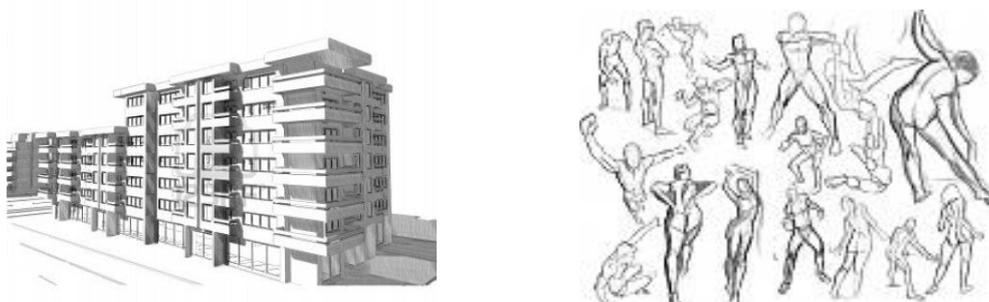


Fig. 3.15 Ejemplos de bocetos de estructuras arquitectónicas y personajes para Modelado 3D. (Autor: Alejandro García Melero, 2014)

La etapa final de esta fase es la creación del **storyboard**, el cual, como se mencionó anteriormente, es el proceso de pasar el guion creado a imágenes. En base a estas, se trabajara con representaciones denominadas unidades que posteriormente se trataran como bloques de trabajo. El storyboard se encarga de la posición de cámaras, acciones, en un aspecto superficial, sin poder controlar detalles técnicos de la ejecución. (González Morcillo Carlos, 2009)

3.4.3.2 Producción

Esta fase de la producción de un modelo 3D, engloba las etapas más engorrosas del proyecto, en este ciclo en específico se encuentran varias etapas que bien pueden ser omitidas dependiendo de los objetivos del proyecto.

Entre las etapas imprescindibles de esta fase se encuentra el proceso de **modelado**, implementación y creación de **materiales** y **texturas**, **iluminación** y **renderizado**. Conceptos que se aplican en cualquier software de modelado 3D, y que se describirán más adelante. Esta fase determinara todos los aspectos visuales y de manera muy general, aspectos de comportamentales del modelo. (González Morcillo Carlos, 2009)

3.4.3.3 Post-Producción

La fase terminal, toma como información inicial las imágenes creadas en la etapa anterior mediante el proceso de renderizado y las subdivide, de manera que sea posible aplicar una serie de filtros y modificaciones (de ser necesarias) para arrojar el resultado final, es decir, las imágenes definitivas que serán utilizadas.

Estas imágenes son organizadas en capas para la estructura final del archivo. Las características que puedan tener “extras”, como por ejemplo, el campo general, la profundidad de una vista, etc, se aplican desde la creación de otras capas. Además, se

contemplan los efectos de sistemas de partículas para poder incorporarse (p. ej. fuego, lluvia, nieve, tierra, etc.). (González Morcillo Carlos, 2009)

3.4.4 ETAPAS PRINCIPALES DE MODELADO 3D

Tal como se ha visto, el modelado 3D es una representación de coordenadas, las cuales constituyen estructuras protegidas por una textura. Esta representación de coordenadas conforma una malla, cuya simplicidad debe ser su principal atributo. Posteriormente, se le añadirán las propiedades características de cada figura creada. Básicamente, un modelo se conforma de esta manera y las características principales de este son determinadas mediante: (Ton Roosendaal, 2014)

3.4.4.1 Modelado

El proceso de modelado es primordial pues permite la síntesis de un objeto por medio de una estructura generada. De esta manera se pueden crear modelos matemáticos que representan fenómenos físicos y/o geometrías de un objeto determinado. Dicho de otra manera, el proceso de modelado 3D, se encarga de generar modelos completos que puedan ser manipulados algorítmicamente por una computadora. (González Morcillo Carlos, 2009). En base a lo anterior, se asume como primer paso la creación de un modelo, para lo cual existen diversas técnicas. (Ton Roosendaal, 2014)

3.4.4.1.1 Modelos constituidos por polígonos

En el presente proyecto se utiliza este tipo de modelado representado por una estructura de polígonos. Por ejemplo, una pirámide está compuesta por 4 triángulos alrededor de una base cuadrada, cada figura es un polígono. De igual forma, con una figura cónica o redonda, puede ser representada por polígonos. Este tipo de objetos creados a partir de polígonos, se dividen en:

- **Primitivas.** Las primitivas básicas son los cubos, conos, cilindros, anillos, tubos, pirámides, plano, esferas.
- **Primitivas extendidas.** Cubo redondeado, tanque de aceite, capsulas, forma C, L, anillo ondulado, prisma, nudo toroide, hose.

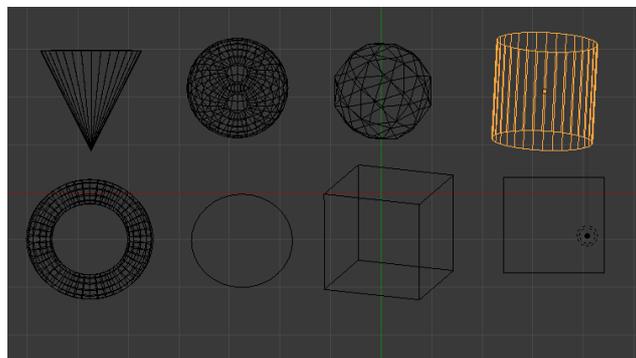


Fig. 3.16 Ejemplos de primitivas predefinidas en un Software de Modelado 3D. (Blender Foundation, 2017)

El modelado poligonal ofrece ventajas tales como: velocidad de procesamiento, alta exactitud en la definición de modelos, varios niveles de detalle en la geometría de los objetos, etc. En este contexto, se debe tener mucha precaución en el número de polígonos que conforman un modelo, pues entre más detallado sea este, mayor cantidad de polígonos se generaran lo que implica un alto número de frames por segundo para el procesamiento. (González Morcillo Carlos, 2009)

3.4.4.1.2 Modelos determinados por curvas matemáticas

Al crear un modelo 3D es importante tener clara la finalidad de este, pues determinara el tipo de modelado más conveniente para tal objetivo. Al trabajar con superficies planas el modelado poligonal resulta de mucha ayuda por su linealidad, sin embargo cuando se trata de superficies curvas con un nivel importante de detalle, se requerirá un alto número de polígonos, lo cual no es beneficioso para el proyecto final. (Ton Roosendaal, 2014)

Como posible solución para la creación de superficies curvas sin la utilización de polígonos, aparecen los modelos definidos por curvas matemáticas. Las curvas pueden ser representadas de manera exacta mediante ecuaciones, estas pueden ser evaluadas y transformadas a polígonos (generalmente triángulos) para su representación final. Algunos ejemplos de este tipo de modelado son:

Nurbs. Una de las técnicas para construir mallas de alta complejidad es mediante Nurbs, las cuales son de aspecto curvado o natural, se crea a partir de splines que a su vez pueden ser implementados por diversos métodos, a partir del anidado de splines se crea la malla.
Patch. Basa su técnica en curvas bezier-splines, crea su estructura básica, es decir, a partir de un “esqueleto” por secciones para formar en conjunto una superficie completa, estas subdivisiones son conocidas como parches que pueden ser sometidos a diferentes procedimientos. (Ton Roosendaal, 2014)

Algunas de las ventajas que ofrece el modelado mediante curvas son:

- **Animación.** Al crear una animación implementando curvas adquieren una forma suave y simple para manipular. Esto es importante en la transición de fotogramas para generar una animación. (González Morcillo Carlos, 2009)
- **Representación.** Suelen ser mucho más simples que los polígonos en cuanto a su estructura, lo cual, como ya se ha mencionado en la sección de “Entornos Virtuales”, es muy útil en el ahorro de memoria, principalmente en dispositivos móviles que, comúnmente, cuentan con una memoria limitada. (González Morcillo Carlos, 2009)
- **Versatilidad.** Una superficie curva puede convertirse en una malla poligonal, lo cual es principalmente útil para determinar los niveles de detalle del modelo. Las primitivas suelen ser más suaves y continuas que sus equivalentes representaciones conformadas por un conjunto de polígonos. (González Morcillo Carlos, 2009)

Sin embargo, cabe mencionar que para los modelos constituidos por polígonos existen varios operadores para desarrollar diferentes técnicas de modelado.

3.4.4.1.3 Operaciones Booleanas

Cuando se trabaja con modelos a base de mayas poligonales existen una serie de operaciones booleanas aplicables, las cuales son acciones específicas que solo se pueden utilizar en este tipo de objetos. En estos procesos, es importante prestar especial atención en las normales (caras) de cada objeto, puesto que la dirección de vista suele cambiar en la creación del modelo por las operaciones aplicadas, modificando así la vista final del modelo creado (materiales, texturas). Las operaciones booleanas no afectan a los operandos originales, por lo tanto el resultado que este arroja se convierte automáticamente en un nuevo objeto, en la mayoría de software de modelado 3D. (Ton Roosendaal, 2014)

Las operaciones booleanas se utilizan seleccionando exactamente dos mallas (bordes, vértices, aristas), se despliega el menú para seleccionar las operaciones posibles y se selecciona la que es requerida. Las operaciones booleanas disponibles en el menú emergente son:

- **Intersección.** Al aplicar esta operación entre mallas, se genera un nuevo que representa el volumen completo de los objetos involucrados.
- **Unión.** Esta operación crea un nuevo objeto con una superficie que representa el volumen de los objetos originales seleccionados.
- **Diferencia.** Cuando se aplica este tipo de operación se tiene que tener en cuenta especialmente el orden en que se seleccionan los objetos, pues esto orden influirá en el resultado final que arroje. Ya que el objeto seleccionado primero, denominado “activo” se restara el segundo. En otras palabras, la superficie completa representara el volumen que corresponde al objeto denominado “inactivo”, sin tomar en cuenta al “activo”.

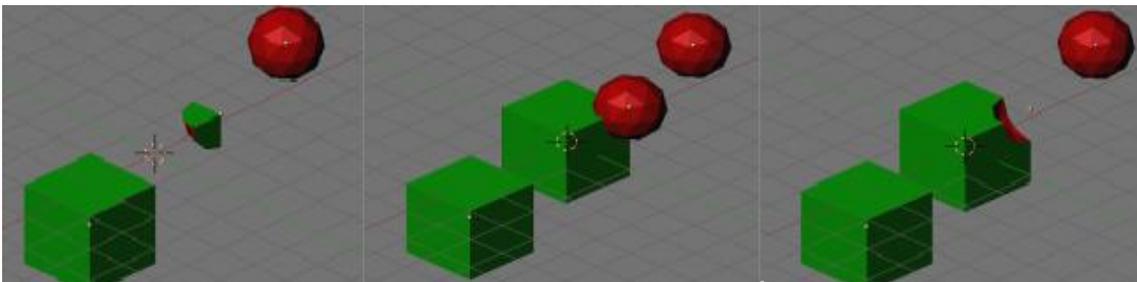


Fig. 3.17 Aplicación de operaciones booleanas en figuras 3D. (Ton Roosendaal, 2014)

Cuando se efectúan cualquiera de las operaciones booleanas pueden derivar en un número considerable de polígonos más en comparación con los polígonos de los objetos originales, particularmente en lo objetos y/o mallas más complejos. También se debe tener en cuenta que los nuevos polígonos, habitualmente, suelen ser de baja calidad pues su malla se vuelve alargada y delgada, y muchas veces, muy pequeños; lo cual interviene

en la calidad de la textura y/o material del objeto en su representación final. (Ton Roosendaal, 2014)

3.4.4.2 Materiales

EL proceso de diseñar e implementar un material para un objeto creado, es necesario y muy similar independientemente de que software de modelado 3D se utilice. Un factor muy importante en la presentación de cualquier material es la luz dentro de la escena, por ejemplo, el motor de render en Blender permite una configuración detallada en la interacción de las luces simuladas y las superficies de los objetos creados, por medio de una serie de parámetros del material en cuestión. El resultado arrojará una imagen renderizada, la cual se trata de una proyección de la escena en una superficie “imaginaria” llamada **plano de proyección**. Este plano es equivalente a la córnea en un ojo humano, aunque la luz que recibe es simulada, para renderizar la imagen se debe confirmar que la luz de la escena está chocando con el plano de proyección completamente.

Para verificar esto, se debe seguir el rayo de la luz simulada (línea recta) desde el punto en el plano de proyección y el punto focal (posición de cámara) hasta que impacte sobre la superficie de algún objeto en la escena, así se determina que luz impactará en ese punto. (Ton Roosendaal, 2014)

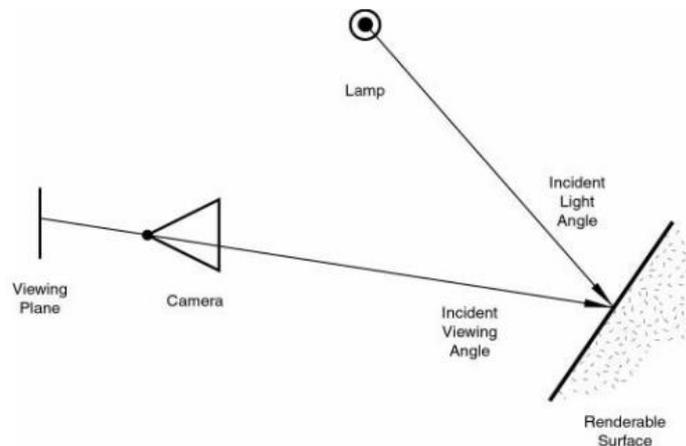


Fig. 3.18 Principio básico en la renderización de una imagen. (Ton Roosendaal, 2014)

Uno de los principios básicos en la renderización de una imagen consiste en, tal como indica la Fig. 3.18, cuando un rayo de luz choca contra cualquier punto de una superficie, genera dos tipos de fenómenos: difusión y reflexión especular. Estas se distinguen entre sí por la relación entre el ángulo de luz incidente y el ángulo de luz reflejada. (Ton Roosendaal, 2014)

3.4.4.2.1 Parámetros Principales en los Materiales

Los materiales también pueden ser descritos como contenedores para **sombreadores (shaders)** y **texturas** que se pueden aplicar en el modelo generado, en gran medida la apariencia de los materiales depende de que sombreado se seleccione; así mismo, los shaders siempre tienen alguna función que comparten. En este contexto, la textura de un

modelo determinara lo que se dibuja en la superficie, y el sombreado se encarga del cómo se dibuja. (Geig, 2013)

En otras palabras, un material contiene propiedades y texturas, de esta manera los shaders determinan que propiedades y texturas puede tener un material. De esta manera, se asume que gran parte de la información que requiere un material para su creación, es respecto al shader de este.

Sombreado (Shader)	Descripción
Difuso	Es el tipo de sombreado por defecto. Se trata del sombreado más básico para los materiales. Este permite que la luz se distribuya uniformemente a través de la superficie del objeto.
Especular	Cuando se aplica una textura especular el objeto en cuestión se muestra brillante. Si se necesita hacer que un objeto simule reflejar mucha luz, este es el tipo de sombreado que se necesita (p. ej. un objeto metálico).
Bumped	Generalmente, cuando se usan en conjunto otros shaders, se utilizan los shaders desplazados. Por ejemplo: bumped-difuso o bumped-especular. Estos shaders un mapa normal para brindar a una textura plana un efecto 3D, de esta manera se puede detallar los modelos en su apariencia sin aplicar modelado complejo.

Tabla 3.7 Tipos de shaders básicos en materiales para modelos 3D. (Adaptación de: Mike Geig, 2013)

Los parámetros de los sombreadores disponibles en la gran mayoría de software especializado, permite determinar la intensidad de dos fenómenos muy importantes: **difusión** y **especularidad** (Tabla 3.7). Estos fenómenos son muy importantes ya que describen el comportamiento físico en la virtualidad, lo cual se explica de la siguiente manera: si A es la energía lumínica que afecta a un objeto, $Ref.$ (reflejo) multiplicado por A , da como resultado la energía difundida (difusión) y $Espc$ (especularidad) multiplicado por A es la energía reflejada especularmente, entonces se debe cumplir $Ref + Espc < 1$, de lo contrario, el objeto irradia más energía de la que recibe. (Ton Roosendaal, 2014)

Es importante tener en cuenta que según el sombreador que se elija, este nos permitirá configurar distintos parámetros, sin embargo, existen propiedad comunes entre los sombreadores, algunas de estas propiedades se presentan en la Tabla 3.8.

Propiedad	Descripción
Color Principal	Esta propiedad del sombreado, define el color de la luz predominante que brilla en el objeto. Esto no implica el cambio de color del objeto en sí, tan solo hace que el objeto parezca diferente. Un ejemplo de esto es cuando se aplica color principal amarillo sobre un objeto con textura azul, el resultado será un objeto con apariencia verde, y así sucesivamente.
Color Especular	El “color especular” determina el color de las partes brillantes de un objeto especular, por default este color es blanco, sin embargo este se puede cambiar a consideración del usuario, ya sea que este tenga la intención de que la

	apariciencia simule un determinado color de luz brillando sobre el objeto en cuestión.
Brillo	Esta propiedad sirve para definir la cantidad de brillantez en un objeto especular.
Textura	Es aquí donde se determina que textura se aplicara al modelo creado.
Normal Map	Tal como su nombre lo dice, este apartado contiene el “mapa normal” que se aplicara al modelo, el cual puede ser utilizado para aplicar relieves a un modelo. Esta cualidad es aplicada sobre todo cuando se pretende dar un mayor nivel de detalle mediante el cálculo de la iluminación.
Tiling	En esta propiedad se define la frecuencia con la que una textura se repetirá en un modelo, es decir, cuantas veces se multiplica y aparece en los ejes “x” y “y”.
Offset	Permite definir si existe un hueco entre los bordes del objeto y la textura aplicada.

Tabla 3.8 Propiedades generales del sombreado en materiales para modelos 3D. (Adaptación de: Mike Geig, 2013)

3.4.4.2 Manejo de múltiples materiales

En el modelado 3D, es común generar objetos que puedan ser manipulados por partes, de esta manera es posible definir un material diferente para cada una, de ser requerido. Sin embargo, en ocasiones resulta más útil un objeto modelado desde una malla, sin perder la propiedad de estar compuesta de varios materiales. Por ejemplo, una planta que posea un material específico para la planta en sí, la tierra y la maceta, tratándose de una sola malla. (Ton Roosendaal, 2014)



Fig. 3.19 Diferentes tipos de malas en diferentes vistas con materiales distintos. (Blender Foundation, 2017)

Cabe mencionar que si el material que será aplicado contiene algún tipo de textura (rasposa, lisa, relieves), estas seguirán vinculadas aun después de convertir en un solo objeto el modelo completo. Sin embargo, este proceso puede ser revertido para poder editar el material y texturas individualmente, lo cual, suele ser conveniente cuando se trata de un objeto con diversos materiales y texturas sin olvidar convertirlo en un solo objeto para facilitar su manipulación dentro de la escena 3D.

3.4.4.2.3 Materiales especiales

Muchos de los materiales que se pueden encontrar en la vida diaria y que se pretenden simular, no solo están conformados de materiales y su textura, por lo cual muchos de softwares de modelado 3D, ofrecen más herramientas de personalización. Dentro estas personalizaciones, se encuentra un paradigma diferente de manejar materiales, ya que no solo se trabaja con las caras de los modelos, sino con sus vértices. (Slick, 2016)

Un ejemplo claro, se muestra en el programa Blender con las opciones de “Halo”, donde mediante la renderización de los vértices se crea un nuevo efecto en la imagen ().



Fig. 3.20 Tratamiento de vértices mediante efectos Halo en modelado 3D. (Ton Roosendaal, 2014 & Bungie Studios, 2001)

Por otro lado, también es importante mencionar que mucho de lo que percibe el ojo humano para reconocer una imagen como “real”, es gracias a los procesos llevados a cabo por artefactos como la cámara fotográfica, tales como: desenfoque, profundidad, los conocidos lens flares, etc. Este último, provee al espectador la información necesaria para que crea que es una imagen autentica, salida de la vida real. (Ton Roosendaal, 2014)

Para conseguir un lens flare, se trabaja mediante diferentes opciones las aristas de las mallas, como en el caso anterior. Estos pueden ser alterados en sus propiedades de manera conveniente para crear el efecto deseado, de tal modo que la imagen presentada simule una escena física legitima.

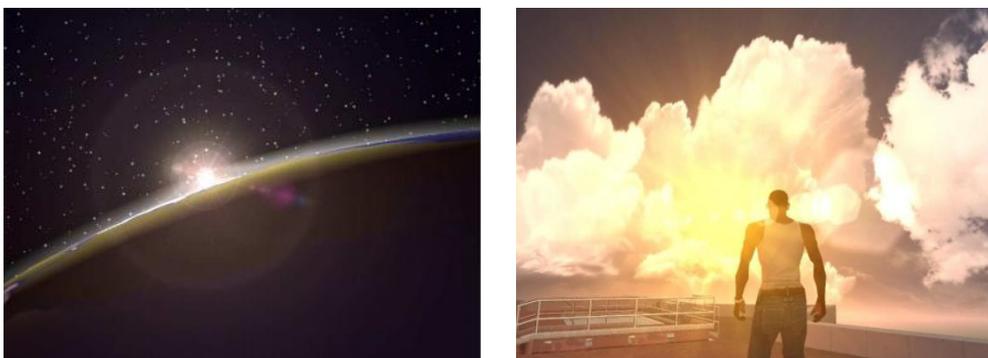


Fig. 3.21 Aplicación de lens flares en imágenes en 3D (Ton Roosendaal, 2014 & David Jones, 2013)

3.4.4.3 Texturas

Como se mencionó en el tema pasado, las texturas son parte muy importante en la creación del material para un objeto, por lo cual estas deben tener características precisas para complementar formalmente el material. La aplicación de texturas en un objeto se realiza mediante un proceso llamado mapeo, el cual resulta una técnica muy potente, mediante la cual los objetos tridimensionales adquieren un aspecto más complejo y realista. El **mapeo de texturas**, ayuda a crear efectos visuales “envolviendo” patrones de imagen 2D sobre la superficie de un modelo 3D. (Jahanzeb Hafeez, 2017)

Las texturas pueden ser de diferentes tipos, dependiendo sus características básicas. Por ejemplo, las texturas procedurales que generalmente se tratan de metales, son creadas mediante fórmulas matemáticas, imágenes compuestas, o mapas de entorno para crear una representación que simule reflexiones y refracciones en la percepción de estas. El proceso de mapeado, representa la relación entre **material** y **textura**; es un proceso bidireccional, pues por un lado la información que se envía a la textura debe ser detallada, para, posteriormente, especificar el efecto que la textura aplicara sobre el material. (Ton Roosendaal, 2014)

En el presente trabajo, la técnica de texturización será la más habitual, como ya se mencionó, se trata de una imagen 2D adherida a un objeto 3D, por lo cual se tomaron las respectivas muestras de las texturas utilizadas en el inmueble real para, mediante un programa de tratamiento de imágenes, se modificó para tener una buena calidad. Las características incorporadas como relieves y perspectivas, en la configuración del programa de modelado 3D, garantiza un gráfico bastante realista.

La evaluación de las texturas genéricas, según Ton Roosendaal (2014), habitualmente se lleva a cabo durante el proceso computacional de la representación gráfica. Por otro lado, los modelos con texturas procedurales ofrecen una representación más compacta, evaluando las texturas según se solicite durante el proceso de mapeado de textura, lo cual puede implicar en un procesamiento más tardado a que se solicite directamente las texturas pre-muestradas que se encuentran almacenadas en arrays simples. Esta información es relevante ya que se contemplan aspectos de procesamiento y viabilidad, como trabajo futuro, a implementarse como una aplicación móvil.

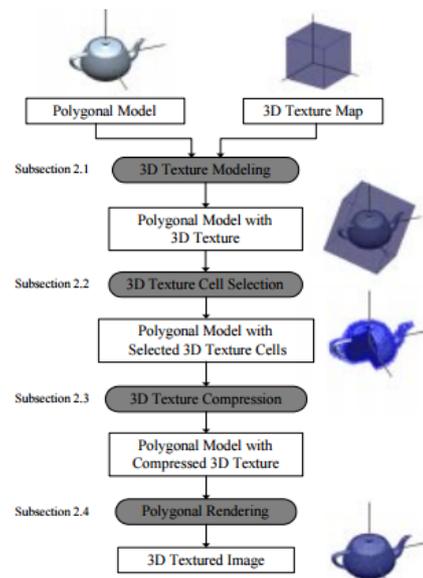


Fig. 3.22 Mapeo de textura en un objeto 3D. (Chandrajit Bajaj & Insung Ihm, 2013)

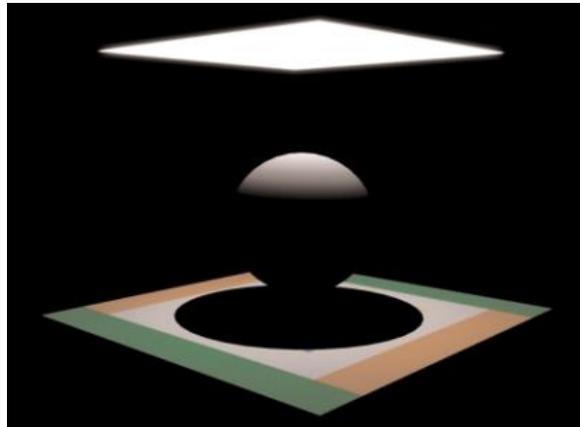
3.4.4.4 Iluminación

La iluminación es un aspecto muy importante en cualquier creación 3D, no importa la escala, la iluminación siempre repercutirá en el resultado final de toda la escena; ya que, la escena mejor modelada y texturizada con la mayor precisión podría arruinarse si carece de un buen esquema de iluminación. (Ton Roosendaal, 2014)

La física de las ondas lumínicas, es simulada por el renderizado de trazado de rayos, el cual, junto con la radiosidad, son procesos lentos que varían según el motor de render que sea utilizado. Los tipos de luces más utilizados son: lamp light, sun light y spot light; las cuales afectan directamente la luz y las que definen texturas directamente para la luz, esto con la finalidad de crear un efecto más personalizado.

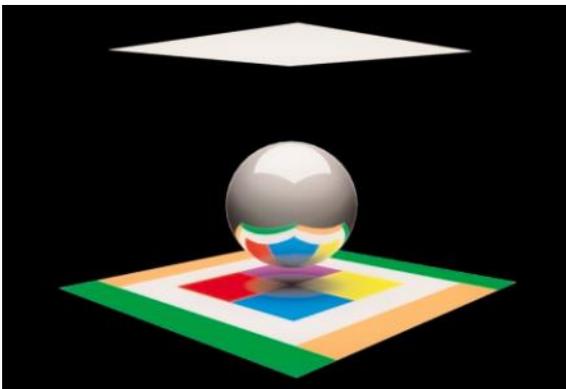
3.4.4.4.1 Iluminación Local

Cuando se habla de iluminación local se hace referencia a aquella iluminación directa sobre cualquier objeto desde una fuente de luz, es decir, no toma en cuenta la transferencia de luz completa, solo afecta a los objetos visibles en la escena. Este tipo de luz/iluminación descuida las reflexiones entre los objetos y la propagación de esta, pues “rebotan” en el ambiente. Es decir, la iluminación local calcula la luz enfocándose al objeto, la trayectoria que va desde la fuente de luz hasta choca con la superficie del objeto en cuestión. Por lo cual, se puede decir que es un modelo de transferencia de luz incompleto. (Gallardo, 2001)



*Fig. 3.23 Iluminación local sobre objeto 3D.
(Arnold Gallardo, 2001)*

3.4.4.4.2 Iluminación General



*Fig. 3.24 Iluminación global sobre la escena.
(Arnold Gallardo, 2001)*

La iluminación global, a diferencia de la local, amplía su rango de efecto, es decir, el cálculo de transferencia de luz es más completo. Esta iluminación aplica la transferencia de luz reflejada en la escena completa. Este efecto es gracias a dos factores aunados, el primero simula el comportamiento luminoso de la luz especular perfecta, el restante se encarga de simular el comportamiento de la luz difusa perfecta.

Como consecuencia, este efecto es conocido como especular ideal, y es el más utilizado para gráficos virtuales realistas. Sin embargo es tan amplia la gama de aplicaciones que ha pasado de fotorealismo a la simulación perfecta, precisión compleja de una imagen, lo cual, beneficia al desarrollo de la virtualidad. (Gallardo, 2001) Existen variadas técnicas y métodos de iluminación, no obstante, en el presente trabajo se implementara este tipo de iluminación acorde a la finalidad del proyecto.

3.4.4.5 Ambiente

La mayoría de los softwares comparten la particularidad de ofrecer opciones para trabajar con el fondo de la escena de nuestra creación 3D, en caso de así requerirlo. Generalmente, cuando se trata de un modelo donde la interactividad se aplicara dentro de este, no se da tanta importancia al ambiente pues este, pocas o nulas veces será contemplado por el consumidor.

Existen variadas opciones que se puede aplicar por defecto en el desarrollo de cualquier modelado 3D; algunos de los más utilizados son los que simulan un ambiente natural, el cual incluye todas las características que este implicaría. Por ejemplo, un ambiente natural traería consigo una luz tenue que afectara la escena y los objetos, lo cual, también es configurable.

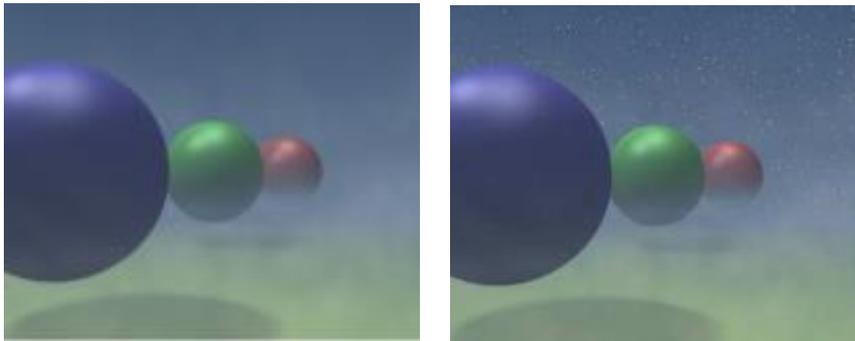


Fig. 3.25 Efectos de ambiente niebla y estrellado. (Ton Roosendaal, 2014)

3.4.5 SOFTWARE DE MODELADO 3D

El modelado en 3D es una herramienta digital de gran impacto dentro de las nuevas tecnologías, del cual, su desarrollo será muy importante en los próximos años. No solamente por la importancia que está adquiriendo la impresión 3D, sino también porque se trata de un mercado que se está ampliando cada vez más.

Generalmente, independientemente del área de conocimiento donde se efectuó, el proceso de modelado constituye la simplificación de un objeto del mundo real, para ser estudiado o representado. De esta manera, se conocen modelos matemáticos que simplifican fenómenos físicos o modelos meteorológicos para la predicción de eventos naturales, entre otros. Se puede definir este proceso como el encargado de crear modelos consistentes que puedan ser manejados algorítmicamente en una computadora, este modelo de construcción se aborda en diferentes etapas, partiendo de entradas básicas y aplicando una serie de operadores sobre ellas. Estas entidades básicas pueden ser formas geométricas primitivas (calculadas de forma algorítmica o mediante una ecuación matemática) u obtenidas mediante herramientas de captura, por ejemplo un escáner 3D. (Carlos González Morcillo, 2012)

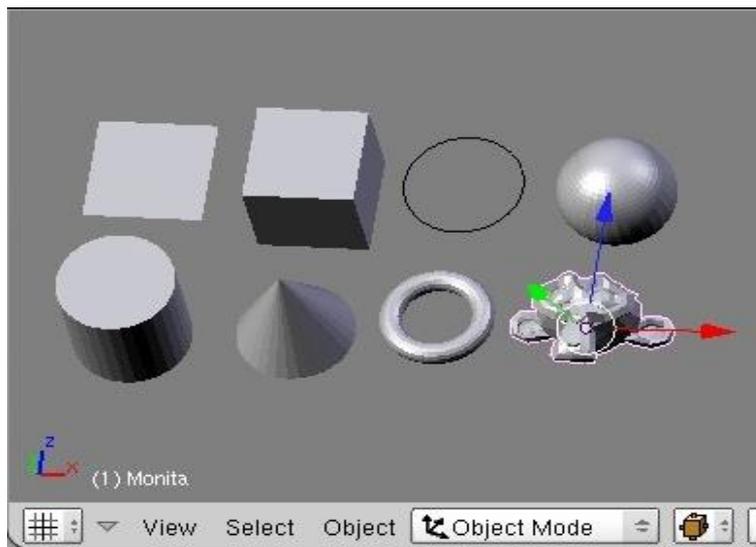


Fig. 3.26 Figura Geométricas Básicas en Software de Modelado 3D. (Blender Foundation, 2016)

Para poder expresar y comunicar de un modo correcto algunos aspectos dentro del ámbito 3D, se necesita tener claro y dominar los fundamentos geométricos, es decir, asimilar conceptos como “sistema de coordenadas”, “vector”, “normal” etc. de una manera intuitiva.

La geometría de un objeto 3D se puede describir de diversas maneras: ecuación matemática de una o varias superficies, una serie de vértices conectados formando polígonos adyacentes, etc. A la hora de visualizar un objeto en tiempo real se suele recurrir a la descripción poligonal por razones de eficiencia y por las actuales arquitecturas hardware de visualización y creación 3D.

3.4.5.1 AutoCAD

En términos oficiales, Autodesk AutoCAD es un software de diseño implementado mediante una computadora para modelado en dos y tres dimensiones. En la actualidad, es desarrollado y comercializado por la empresa Autodesk. (Felipe Díaz del Castillo Rodríguez, 2015)

AutoCAD es un software altamente reconocido en el diseño asistido por computadora CAD, por sus siglas en inglés (Computer Aided Design). Las características de las creaciones como: tamaño, contorno y la forma, son almacenados en dibujos electrónicos ya sean bidimensionales o tridimensionales.

Este tipo de software de diseño, le ofrece al usuario una rápida forma de manipulación en sus creaciones, ya que al realizar su diseño puede efectuar rápidas modificaciones, abreviando el proceso que implicaría en un dibujo hecho a mano con lápiz y papel.

El CAD, sustituye sin problema el tablero, las escuadras, las estilográficas y las plantillas implicada en un diseño. Lo cual, amplía mucho más las posibilidades de creación del artista y facilita así muchas de las dificultades al presentar alguna idea. Claramente, con el CAD, se olvida las tareas laboriosas, tediosas y repetitivas, los borrones, manchas de tinta, hasta comenzar desde cero.

Entre las ventajas que trae consigo el sistema CAD se encuentran:

- La división de la superficie para así poder utilizar un color o material diferente en cada subdivisión.
- Las distintas vistas/perspectivas del diseño creado.
- La fácil manipulación de las formas creadas, posición, rotación, etc.
- Medidas exactas y datos relevantes respecto al espacio de trabajo y objetos creados: volumen, superficie, centro de gravedad, inercia, etc.

En la actualidad, en el mercado se encuentra gran variedad de sistemas CAD, generalmente manejando dos modalidades de desarrollo, el trabajo 2D y 3D, incluso, en alguno programas se puede simular su comportamiento para definir características que tendría.

AutoCAD, es uno de los programas más utilizados en el mundo del diseño en sus diferentes áreas de conocimiento, arquitectos, ingenieros, diseñadores son el público de este potente software reconocido internacionalmente por sus grandes capacidades de edición. (Felipe Díaz del Castillo Rodríguez, 2015)

3.4.5.2 Maya

Autodesk Maya, es un programa de modelado y animación 3D por excelencia en este ámbito, y al igual que el programa anterior, también pertenece a la familia de Autodesk. Este software representa gran trabajo de investigación y desarrollo, donde ha ido

creciendo a partir de mezclas de diversas tecnologías 3D, e incluso, de aportaciones de grandes empresas relacionadas con el entretenimiento, por ejemplo Wall Disney con sus conocimientos en animación.

Es uno de los software's para modelado 3D más potentes en la actualidad, ya que no solo lo respalda la empresa consolidada en este ámbito, sino también, la tecnología que engloba las herramientas que ofrece, las cuales se desarrollan a través de composiciones poligonales, renderizado, animaciones y efectos especiales aplicados en diversas secuencias.

Maya es un programa basado en un arquitectura abierta, lo cual posibilita que a través de su versátil API (capacidad de comunicación que este tiene entre sus componentes de software), la comunicación con la gran mayoría de programas especializados en diseño gráfico, como lo es Autodesk 3ds Max; también, facilita la interacción con scripts de programación, generalmente, en Python. (FormacionProfesional.info, 2016)

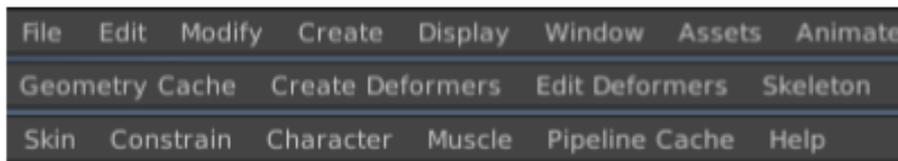


Fig. 3.27 Barra de Menú de Software Maya (Recuperada de: FormacionProfesional.info, 2015)

3.4.5.3 Sketchup

Se trata de un sistema de manejo sencillo, el cual se puede descargar de manera gratuita y es recomendable si se está iniciando en el mundo del modelado 3D. Existen versiones con herramientas más avanzadas, las cuales se encuentran restringidas a un público dispuesto a pagar por estas, sin embargo, la versión gratuita es suficiente para grandes renders y animaciones.

Sketchup no solo genera modelos 3D, también ofrece una potente herramienta llamada LayOut que funciona para generar documentos formados por varias páginas con diferentes vistas desglosadas de un modelo en particular.

La potencialidad de este programa, reside precisamente, en los plug-in (complementos) que se pueden añadir a sus funcionalidades.

Es así, como Sketchup, se posiciona como uno de los mejores programas de diseño 3D, que sin sacrificar la calidad de procesos y herramientas, permite emplear una interfaz bastante simple e intuitiva. (InTec, 2014)

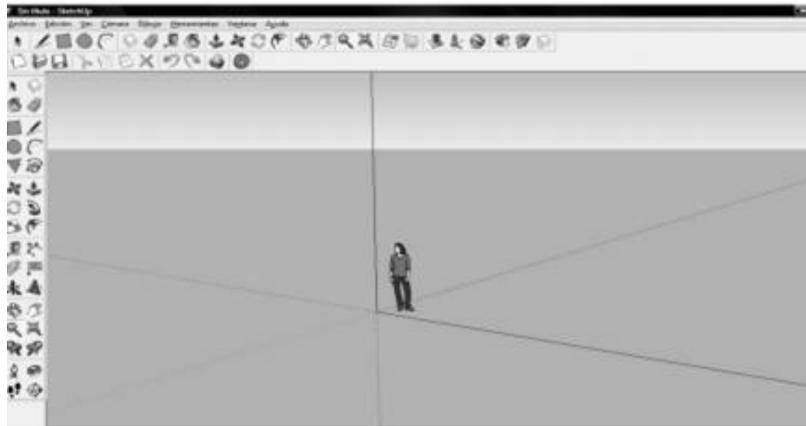


Fig. 3.28 Interfaz de Sketchup (Intec, 2014)

3.4.5.4 Revit

Revit es un software especializado en diseño arquitectónico e ingeniería estructural, particularmente del Building Information Modeling (BIM). Funciona basándose en bocetos conceptuales de adquisición gratuita, análisis de energía sostenible, funciones en colaboración y trabajos compartidos.

Revit, pertenece a Autodesk, lo cual precede un software de calidad, que permite el modelado de información de construcción, donde se admiten el diseño, los dibujos y tablas de planificación, los cuales son requerimientos en un proyecto de construcción. BIM, el Modelo de Información de Construcción, contribuye con la información referente al diseño, cantidades, envergadura, y fases que se necesitan en un proyecto.

Este Software, maneja un modelo en donde cada plano de dibujo, vista 3D o 2D y tabla de planificación representa información originaria de la base de datos del modelo de construcción interno. Como se mencionó en un principio, este tipo de software resulta ser más metódico y específico para ciertas actividades dentro del mundo de la ingeniería por el modelo en el que se basa. (Autodesk, 2014)

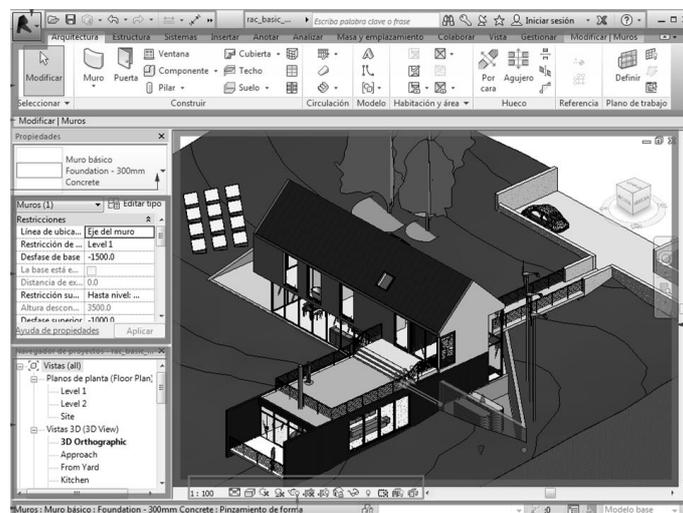


Fig. 3.29 Interfaz de Revit (Autodesk, 2014//Licencia CC BY-NC-SA 3.0)

3.4.5.5 3DS Max

Un derivado más de Autodesk, se trata de un potente software implementado para la creación de animaciones, logotipos, diseños 3d, imágenes avanzadas. Contiene una gran gama de herramientas que permite la creación de estos objetos, de una forma más sencilla, proporcionando vistas de diferentes perspectivas para visualizar la creación y así ayudar al desarrollo. (Martínez J. R., 2015)

Es uno de los programas más utilizados para la animación y diseño en tres dimensiones, siendo uno de los primeros programas especializados, a lo largo del tiempo se convirtió en un referente para la producción de cine, video y diseño multimedia.

Este programa incorpora una gran variedad de sofisticada herramientas para crear imágenes y animaciones en tres dimensiones, manipulables en un entorno de trabajo bastante sencillo de utilizar. 3D Max, tiene la peculiaridad de trabajar con un paradigma de desarrollo orientado a objetos, lo cual quiere decir, que todos los componentes creados dentro de la escena y las operaciones que efectúe el usuario, se les asigna un nombre y una serie de características/propiedades que las hacen irrepetibles dentro de la escena. (Mario Luis Arce, 2012)

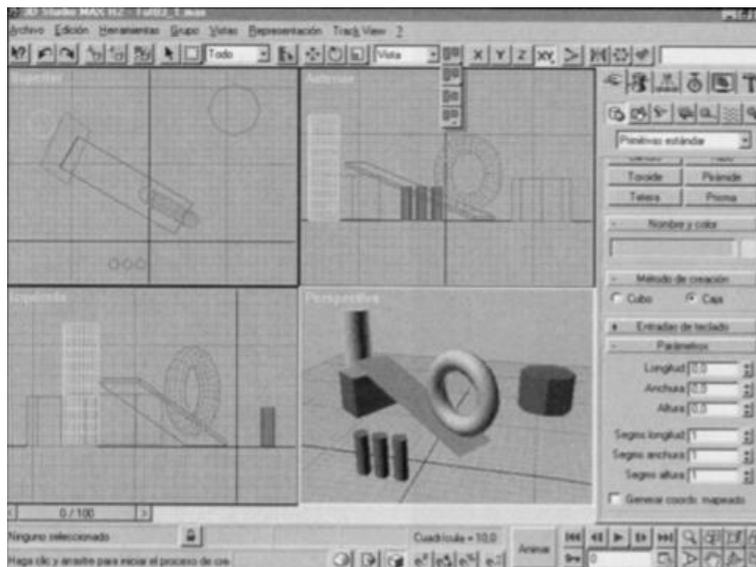


Fig. 3.30 Interfaz de 3D Max Studio (Francisco Calero, 2012)

Otro importante aspecto de 3D Max Studio, es su arquitectura basada en plug-in, con la que todo, incluyendo la interfaz, representa un módulo externo. Los plug-in controlan todas las fases del proceso de creación de imágenes, el modelado, la representación, la postproducción de video, etc. Aunado a ello, los programadores pueden crear sus propios plug-in, interfaz, u otra característica no incluida, utilizando el kit de desarrollo de software SDK.

3.4.5.6 Blender

Actualmente existen gran variedad de aplicaciones de modelado 3D, sin embargo Blender rápidamente se posiciona como una de las más solicitadas. Las características que lo definen son la capacidad para crear modelos, animación, renderizado fotorrealista y la edición de vídeo. Para los usuarios más avanzados, el software también permite crear juegos completos en 3D, editar videos y simulaciones realistas.

Es importante mencionar que Blender es primeramente destinado al modelado 3D de objetos para después situarlo en representaciones del mismo, para ello Blender ofrece la posibilidad de aplicar texturas y materiales, manejar luces dentro de la escena, edición de manera sencilla, etc. También incluye las tecnologías de más relevancia dentro del diseño 3D tal como: textos, meta-objetos, mallas, superficies, curvas y modelado escultórico.

Blender permite desarrollar animaciones de los modelos creados, como mecánicas de movimiento de componentes robóticos hasta efectos más detallados como emisiones de partículas para explosiones, sin dejar de lado las cinemáticas para el comportamiento de personajes. Cuenta con las herramientas necesarias para animar gases, fluidos, cuerpos blandos, pelo, partículas, etc.

Este software también incorpora un motor de juegos llamado Blender Game, que funge como simulador de experiencias físicas, paseos virtuales o cualquier creación interactiva. Blender game, es una modalidad de trabajo dentro de la interfaz del programa, donde al igual que la modalidad estándar donde los paneles y parámetros están orientados al diseño de imágenes fijas y animaciones, también brinda herramientas especializadas.

Además, brinda la tecnología para algún montaje de alguna producción, pues cuenta con su propio editor de video. Y no se puede dejar de mencionar la potente tecnología que ofrece, de edición de nodos, la cual permite manipular cada uno de los efectos por separado y no como en muchos otros programas de edición donde se encuentran de manera secuencial, esto da más libertad a la hora de manipularlos.

Por último, pero no menos importante, cuenta con una consola para programar en lenguaje Python, ya que se trata de software libre se puede extender hasta cubrir las necesidades de cada proyecto. Cualquier script puede ser creado y añadido para desempeñar funciones específicas, y así adaptarse a las necesidades del usuario. Tal como se puede apreciar, Blender es uno de los programas más potentes y versátiles para el modelado 3D, por lo cual es elegido para crear el modelado del inmueble que se tiene contemplado para el presente proyecto. (INTEF, 2012)

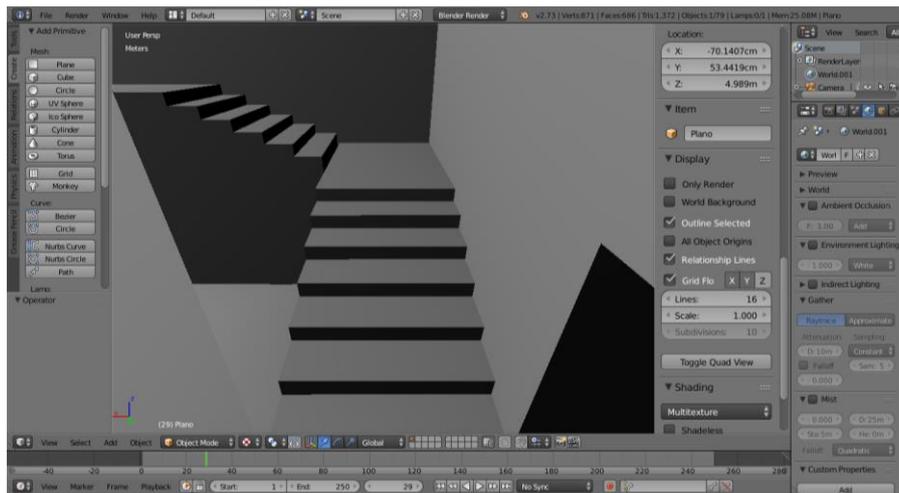


Fig. 3.31 Interfaz de Software Blender. (Blender Foundation).

3.4.5.7 Tabla Comparativa y Selección Final

La selección final se basó en el análisis de la literatura de trabajos realizados en diferentes software de modelado 3D y con la experiencia propia en la mayoría de los software's contemplados, contemplando que estos programas están en constante mejora se les asigno una observación objetiva de acuerdo a la información recabada. Los conceptos básicos visto anteriormente, respecto a la utilización de software's de modelado, apuntan a que las características que posee cada programa están definidas por su especialización.

Los software's vistos hasta ahora, son los más populares según un estudio realizado por Bim MakeOver (2012), servicio de impresión 3D y un estudio de tesis de Valdez (2012), cuyo objeto de estudio ha sido la aplicación en sí misma. Entre las diferencias más notables, se rescata la accesibilidad a estas aplicaciones, ya que muchas de estas tienen una versión free más y menos completas, lo cual se calificó con Alta, media y baja calidad en las herramientas que estas versiones ofrecen. También es significativa la posibilidad de estas para crear modelos 3D cada vez más complejos, con mayor detalle y por ende resultan más pesados. Sin embargo, esta última característica también se ve influenciada por la potencia de la computadora donde se ejecute.

CARACTERÍSTICA	AUTOCAD	MAYA	SKETCHUP	REVIT	3DS MAX	BLENDER
Compatibilidad	MEDIO	ALTO	ALTO	MEDIO	ALTO	ALTO
Recursos computacionales	ALTO	MEDIO	MEDIO	ALTO	MEDIO	MEDIO
Calidad de animaciones	MEDIO	ALTO	MEDIO	MEDIO	ALTO	ALTO
Geometría compleja	ALTO	ALTO	MEDIO	BAJO	ALTO	ALTO
Assets disponibles	ALTO	MEDIO	MEDIO	MEDIO	ALTO	ALTO
Accesibilidad	MEDIO	MEDIO	ALTO	BAJO	ALTO	ALTO

Tabla 3.9 Tabla comparativa de Software's 3D. Elaboración propia.

3.5 DESARROLLO INTERACTIVO: UNITY

Una de las fases más importantes del presente proyecto es la incorporación de la interactividad en el sistema, actualmente, existe ya una variada gama de software que ofrecen potentes tecnologías para hacer de esta actividad un proceso un poco más sencillo. Sin embargo, se ha optado desarrollar esta fase tan importante en el programa Unity, por ser el que más se ajusta a nuestras necesidades.

Unity 3D es un motor gráfico 3D, el cual cuenta con versiones para PC y MAC, que se ideó especialmente para el desarrollo de juegos, aplicaciones interactivas, visualizaciones y animaciones tridimensionales.

Creada por Unity Technologies, fundada en el año 2004 por David Helgason (CEO), Nicholas Francis (CCO), y Joachim Ante (CTO) en Copenhague, Dinamarca. Su éxito, se ha debido principalmente al enfoque de la compañía, que se ha preocupado por las necesidades de los desarrolladores independientes, haciendo una plataforma de desarrollo accesible para todo el público.

“A partir del boom de los smartphones aproximadamente en el 2008, Unity fue los primeros motores en respaldar la plataforma de iPhone. Actualmente, Unity es utilizado por el 53.1% de los desarrolladores según la encuesta realizada por “Game Developer Tecnología Móvil y Social”, la cual también menciona el 65% de los encuestados trabajan en títulos para móvil, mientras el 64% tiene contemplado lanzar próximos juegos y app para dispositivos móviles.” (Alejandro Cerón Cardona, 2014)

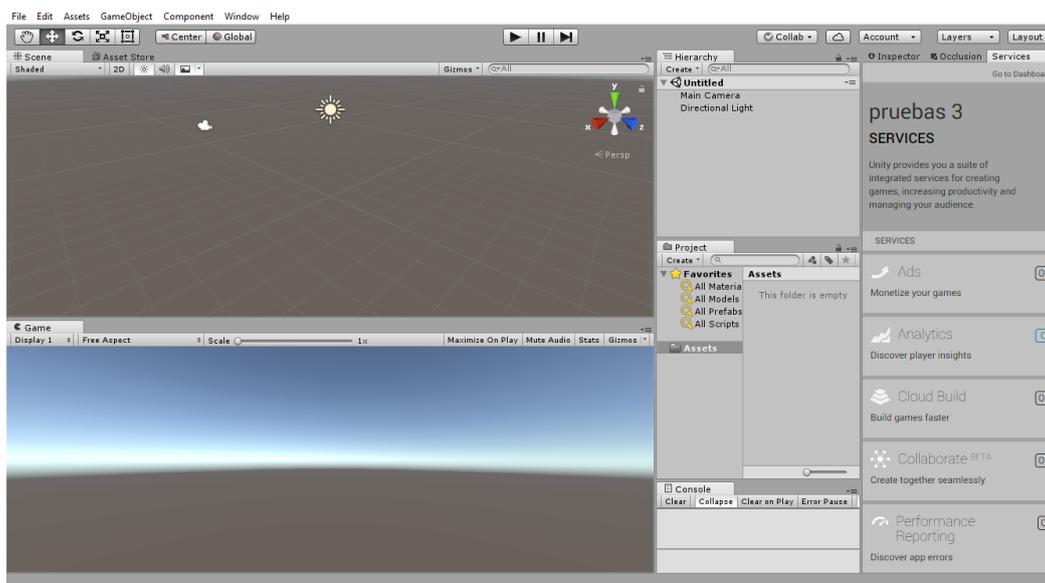


Fig. 3.32 Interfaz de Unity. (Unity Technologies)

Unity es potente software que ofrece gran variedad de herramientas de desarrollo para, tanto a nivel físico como a nivel Algunas de estas herramientas se describen a continuación.

3.5.1 APLICACIÓN DE ASSETS

Un asset es una representación de cualquier ítem que puede ser utilizado en la creación de algún proyecto. Es decir, un asset podría venir de un archivo creado afuera de Unity, tal como un modelo 3D elaborado con otro software, un archivo de audio, o algún otro tipo de archivo que acepte Unity. (Technologies, 2017) Es importante mencionar que no todas las extensiones de archivo que son aceptadas por Unity, cuentan con el mismo nivel de calidad, pues esto depende de cuanta compatibilidad hay entre el software involucrado y Unity.

Dentro de los tipos de asset, se encuentran aquellos que pueden ser generadores desde el propio Unity, tal como un algoritmo de movimiento mejor (animator controller), una nueva textura (render texture), o crear un nuevo audio. Algunos de los archivos que acepta Unity son para importar modelos (maya, max, fbx), de sonido (wav, mp3, mp4), de imagen (psd, bmp, tif, jpg). (Technologies, 2017)

Los asset más utilizados en la creación de proyectos en esta plataforma, son modelos 3D texturizados desde una gran variedad de fuentes. Algunas de las características y recomendaciones que se deben de tomar en cuenta para manipular estos modelos de manera eficiente en Unity se describen a continuación. (Technologies, 2017)

3.5.1.1 Escalas y Unidades

La mayoría de los programas de modelado 3D, suelen trabajar por default a una escala 1:1, así que salvo esta escala sea configurada por otra, lo que va “dibujando” en el software es de proporciones reales. Sin embargo, a la hora de exportar estos modelos e importarlos a otro software se pueden ver alteradas estas especificaciones, por lo cual es importante siempre seguir las recomendaciones dadas por el software donde se manipulara el proyecto, es este caso, Unity.

- Ajusta las unidades del sistema y del proyecto para que el software de origen funcione consistentemente con Unity, por ejemplo el sistema métrico, es decir, utilizar la unidad de medida real del objeto a exportar y utilizar esa misma medida en Unity.
- Trabajar la iluminación y la simulación de física a escala para lograr mayor convergencia entre toda la escena/escenas.
- Tener presente que el sistema de unidades con el que trabaja cada software de origen de los modelos.

- Específicamente para los tipos de archivo **FBX**⁹ y para archivos de aplicaciones 3D, Unity maneja un escalado diferente cuando son importados, sin embargo esto se puede modificar en la configuración de escala de importación en el Inspector.
- A manera de prueba para comprobar la métrica, se puede exportar con la escena seleccionada, una figura, por ejemplo un cubo de 1 metro para ver si coincide en Unity, o hacerlo coincidir si es necesario.

3.5.1.2 Archivos y Objetos

Los archivos generados por programas de modelado 3D suelen contemplar una gran variedad de extensiones para exportar sus proyectos y así ampliar su rango de compatibilidad con otros programas para su manipulación. Para optimizar esta acción se deben tener en cuenta algunos factores que podrían determinar el funcionamiento, calidad, y simpleza del proyecto creado.

- Es importante nombrar a los objetos en la/las escena(s) de forma única e irrepetible para ser localizados de manera rápida y detectar problemas en meshes (mallas) específicos del proyecto.
- Evitar usar símbolos especiales (“#\$%”).
- Asignar nombres sencillos y representativos para objetos y archivos, de esta manera las operaciones sobre estos (cortar, duplicar, borrar) serán más rápidas y exactas.
- Mantener jerarquías simples e intuitivas para la organización de los objetos creados en el proyecto.
- Al utilizar proyectos grandes en la aplicación 3D generada, es recomendable tener un archivo de trabajo que esté por fuera del directorio de trabajo de Unity. Esto ayudara al ahorro tiempo consumido en actualizaciones y en importar datos innecesarios.

(Technologies, 2017)

3.5.1.3 Meshes

En el apartado de EV3D, se mencionó el modelo geométrico el cual, en este caso, está conformado a base de polígonos. Un polígono está compuesto por caras en forma triangular, las cuales son conocidas en este ámbito como mallas (meshes). Estas son muy importantes en el procesamiento del modelo, por lo cual se sugiere contemplar algunos aspectos importantes. (Technologies, 2017)

- Para ahorrar tiempo en el renderizado y disminuir costes, se deben usar solo los polígonos necesarios. (Technologies, 2017)
- Optimizar la geometría de los objetos si éstos contienen demasiados polígonos.

⁹ Se trata de un dibujo 2D o 3D guardado en formato de Autodesk, el cual mantiene la fidelidad total y funcionalidad del archivo original y es compatible con gran variedad de programas. Es comúnmente utilizado para la creación de interoperabilidad entre aplicaciones 3D.

- La utilización de polígonos espaciados uniformemente en edificios, paisajes y arquitectura ayudarán a extender la iluminación y evitarán problemas incómodos. (Technologies, 2017)
- En medida de lo posible, evitar implementar triángulos demasiado angostos y largos. (Technologies, 2017)

Particularmente se debe tener un mayor cuidado en la creación de las meshes, por los problemas que implicaría un objeto utilizado frecuentemente sin la optimización de sus polígonos (lentitud, calidad visual, etc). Los expertos en modelado y desarrollo en Unity, recomiendan, como una vital regla práctica, empezar la construcción de objetos de manera simple y solo agregar detalles donde sea necesario. “Siempre es más fácil agregar polígonos que removerlos”. (Technologies, 2017)

3.5.1.4 Texturas

Las texturas con dimensiones que sean potencias de 2, por ejemplo 512x512 o 256x1024, serán más eficientes y no necesitarán ser re-escaladas¹⁰ en tiempo de construcción. Hablando técnicamente, las dimensiones más altas disponibles en gran parte de las tarjetas gráficas y plataformas, son 2048x2048. De igual manera, se recomienda crear buenas texturas en base a una serie de especificaciones estipuladas en el desarrollo profesional. A continuación se mencionan a grosso modo algunos de estos lineamientos

- Utiliza archivos fuente de alta resolución, esto determina la calidad del proyecto de Unity (PSD o Gimp). Esto es importante ya que a la hora de manipular la textura siempre se podrá disminuir la resolución al archivo fuente, sin embargo, un archivo con baja resolución no se le podrá aumentar. (Technologies, 2017)
- Utiliza la salida de resolución de textura que se requiera en la escena (guarda una copia p.ej. un archivo PNG optimizado de 256x256 o un TGA¹¹). Se puede seleccionar la resolución, en base a dónde será vista la textura y en dónde será mapeada. (Technologies, 2017)
- Cree un apartado en el proyecto de Unity para almacenar los archivos de salida de texturas (p.ej. \Assets\texturas). (Technologies, 2017)
- Es muy importante verificar que el archivo de trabajo 3D que se está utilizando, esté referenciando a las mismas texturas, para mantener la consistencia si se desea guardar o exportar. (Technologies, 2017)
- Se puede hacer uso del espacio disponible en la textura seleccionada, pero se debe contemplar que diferentes materiales requieren de partes diferentes de la misma textura. Por lo cual, se puede terminar utilizando o cargando la misma textura en varias ocasiones. (Technologies, 2017)

¹⁰ Consiste en duplicar píxeles para ajustar una imagen que no tiene las características necesarias para ser representado en algún dispositivo específico, donde la calidad se ve disminuida.

¹¹ Formato de gráficos discretos creado por Televisión, este tipo de formato puede almacenar datos de imagen de 1 a 32 bits con una exactitud de pixel.

- Para implementar cortes (alpha) y elementos que requieran diferentes sombreadores, es de utilidad separar las texturas. (Technologies, 2017)
- Se pueden utilizar texturas embaldosadas (tiling textures), las veces que sean necesarias, luego se podrá utilizar una mejor resolución que se repita en el espacio. (Technologies, 2017)
- Se debe evitar del bitmap, los elementos donde sea más notorio que son repetitivos, teniendo en cuenta el contraste. Si se quiere agregar detalle implemente decals y objetos para atenuar las repeticiones. (Technologies, 2017)
- Unity se encargara de la compresión de la plataforma de salida; lo cual implica que, a menos que la fuente ya sea un JPG de la resolución correcta, es mejor emplear para las texturas un formato que no tenga pérdida de datos. (Technologies, 2017)
- Al crear una página de textura a partir de fotografías digitales, se debe reducir la página a pequeñas secciones individuales que se puedan repetir. Por ejemplo, no se necesitan una docena del mismo objeto (ventana, puerta) consumiendo espacio de textura. De esta manera se puede tener mayor detalle de pixeles para un único objeto que se repite. (Technologies, 2017)

3.5.1.5 Materiales

Cuando se modelan objetos 3D, por default, la forma tiene un color asignado el cual suele corresponder al color de la capa (layer) donde se ha generado. Uno de los objetivo del modelado 3D es crear escenas fotorrealistas, es decir, emular de la manera más realista, los efectos atmosféricos, lumínicos, texturas, y otras características visuales. Específicamente, al momento de exportar estas texturas, y materiales, se debe tomar en cuenta algunas observaciones para optimizar la manipulación de estas en Unity.

- Los materiales utilizados en cada una de las escenas o escena deben ser organizados y nombrados de manera representativa. De esta manera, encontrar y editar los materiales será mucho más sencillo cuando haya sido importado a Unity. (Technologies, 2017)
- Siempre estará la opción de crear materiales en Unity, siempre y cuando sea necesario para el proyecto. (Technologies, 2017)
- No todas las configuraciones para materiales en el paquete original exportado pueden ser importadas a Unity. (Technologies, 2017)
- Colores difusos, texturas difusas y nombres son generalmente aceptados. (Technologies, 2017)
- Configuraciones de modelo de shader, mapas especular y normal, otras texturas secundarias y material de sustancia no serán reconocidas e importadas. (Technologies, 2017)

3.5.2 NAVEGACIÓN Y PATHFINDING

Unity tiene integrado un sistema de navegación que permite crear personajes que pueden moverse de forma predeterminada en la escena, esto se logra mediante la utilización de meshes de navegación que son creadas automáticamente de la geometría. El sistema de navegación de Unity consiste en los siguientes componentes:

- **NavMesh.** Se trata de una estructura de datos que describe las superficies explorables del entorno y permite encontrar el camino a una ubicación predeterminada. (Technologies, 2017)
- **El NavMesh Agent.** Ayuda a crear personajes que se evitan entre si mientras se mueven. Los Agents tienen cierto nivel de razonamiento acerca de su entorno, y mediante el NavMesh saben cómo evitar obstáculos, sin embargo, su comportamiento es muy básico y no aprenden a través la información que perciben. (Technologies, 2017)
- **Off-Mesh Link.** Es un componente que hace posible incorporar “atajos de navegación” los cuales no pueden representados utilizando una superficie caminable, por ejemplo, abrir una puerta, saltar una valla, pueden descritos a través de estos enlaces. (Technologies, 2017)
- **NavMesh Obstacle.** Ayuda a describir obstáculos que se mueve, donde el personaje debe evitar estos para seguir su desplazamiento por el mundo virtual. (p. ej. un barril o una caja controlada por el sistema de física). (Technologies, 2017)

Existe una opción llamada **Navigation Area**, que resulta ser muy importante para determinar el área de navegación y los costos que esta implica, es decir, define que tan difícil sería caminar por determinada área dependiendo de su compuesto. Las áreas con bajo costo son preferidas durante un **path finding** (encuentra caminos), adicionalmente, cada NavMesh Agent tiene un área que puede ser utilizada para especificar las áreas donde el “agente” puede moverse.

El costo permite controlar que áreas de **pathfinder** (encuentra caminos) son más viables cuando se encuentra un camino. Un ejemplo podría ser, si se configura el costo de una área a 3.0, esto implicaría que desplazarse por esta área es considerado tres veces mayor que otra ruta diferente. (Technologies, 2017)

Al analizar el comportamiento de esta función, pathfinder, se puede crear una relación con el algoritmo de **búsqueda heurística A***, donde la idea general es encontrar el camino más óptimo para llegar a determinado destino. La función en Unity, coloca un punto que represente cada polígono, el cual será tratado como un nodo, para así poder relacionar los polígonos entre si y calcular el camino más corto entre nodos. Sin embargo, no está limitado a funcionar de esta manera, pues una de las ventajas de las funciones que ofrece Unity, es la posibilidad de edición, donde se pueden mejorar el funcionamiento del algoritmo para obtener los objetivos propios.

3.5.3 ÁREA DE RENDERIZACIÓN

Los componentes de renderización, estos componentes tienen que ver con renderizar elementos del juego de la interfaz de usuario. La iluminación y efectos especiales también están incluidos en este grupo.

3.5.3.1 *Cubemap*

Este componente, dentro de la renderización resulta importante, ya que se trata de una colección de seis texturas cuadradas que representan las reflexiones sobre un entorno. Los seis cuadrados conforman las caras de un cubo imaginario que rodea un objeto, donde, cada cara representa la vista hacia las direcciones de los ejes del mundo (arriba, abajo, izquierda, derecha, adelante y atrás).

Los cubemaps son utilizados habitualmente para capturar reflexiones o “alrededores” de objetos, por mencionar algunos, skyboxes y reflejos del ambiente. Existen otras técnicas, por ejemplo generar el cubo desde los contenidos de una escena de Unity utilizando un script. La función **Camera.RenderToCubemap** puede grabar las seis imágenes de la cara desde cualquier posición deseada en la escena. Esto facilita la creación de los efectos dentro de una escena.

3.5.3.2 *Oclusión Portals*

En cualquier juego o aplicación donde se le permita al usuario interactuar y explorar un entorno, es necesario generar primitivas de oclusión que se puedan abrir y cerrar en tiempo de ejecución, que podría ser representado por una puerta que representa un obstáculo hasta que esta permite al personaje pasar sin perder las normas físicas. Para ello Unity, ofrece portales de oclusión.

3.5.3.3 *Skybox*

El skybox, es utilizado para ambientar la escena, se puede decir que es una envoltura alrededor de la escena entera el cual muestra como el mundo se vería más allá de su geometría. (Technologies, 2017)

Propiedad	Función
Tint Color	Define el color de tinte
Exposure	Permite ajustar la intensidad del brillo del skybox
Rotation	Modifica la rotación alrededor del eje del skybox
Front	Contiene las texturas utilizadas para cada vista de cara de cubo que almacena al skybox. Se debe tener especial atención en la colocación de estas texturas en las “ranuras” correctas

Tabla 3.10 Propiedades de Skybox en Unity. (Unity Technologies, Publication 5.3-Q 2016)

Los skyboxes son renderizados alrededor de la escena completa, esto con la finalidad de dar una impresión de un escenario complejo visto a lo lejos de la geometría.

3.5.3.4 LOD Group

Cuando se trabaja con escena cada vez más grandes y complejas, el rendimiento se convierte en un factor muy importante. Por lo cual, este proceso se suele manejar mediante mallas con diferentes niveles de detalle, directamente ligados a la distancia que se encuentre la cámara del objeto. A esto se le conoce como nivel de detalle (**LOD Group**). (Technologies, 2017)

Estos grupos, denominados “grupos LOD”, son utilizados para administrar el nivel de detalle (LOD) de **GameObjects** (objetos en escena). EL nivel de detalle es una técnica implementada para la optimización, la cual utiliza varias mallas para un objeto, estas mallas representan el mismo objeto con un detalle decreciente en la geometría. La idea es que las mallas de bajo detalle se muestran cuando el objeto se encuentra lejos de la cámara y la diferencia sea imperceptible. Otra ventaja de utilizar LOD de manera selectiva, es que las mallas con geometría más sencilla son mejor exigentes para el hardware, es decir, requieren menor consumo de recursos.

3.5.4 ÁREA DE MEMORIA

Al crear una aplicación para cualquier tipo de dispositivo (escritorio, celular, Tablet. etc.) es importante valorar el uso de memoria. Unity, ofrece dos modos para monitorear el uso de memoria de la aplicación creada.

El modo simple, muestra como utiliza la aplicación desarrollada la memoria disponible.

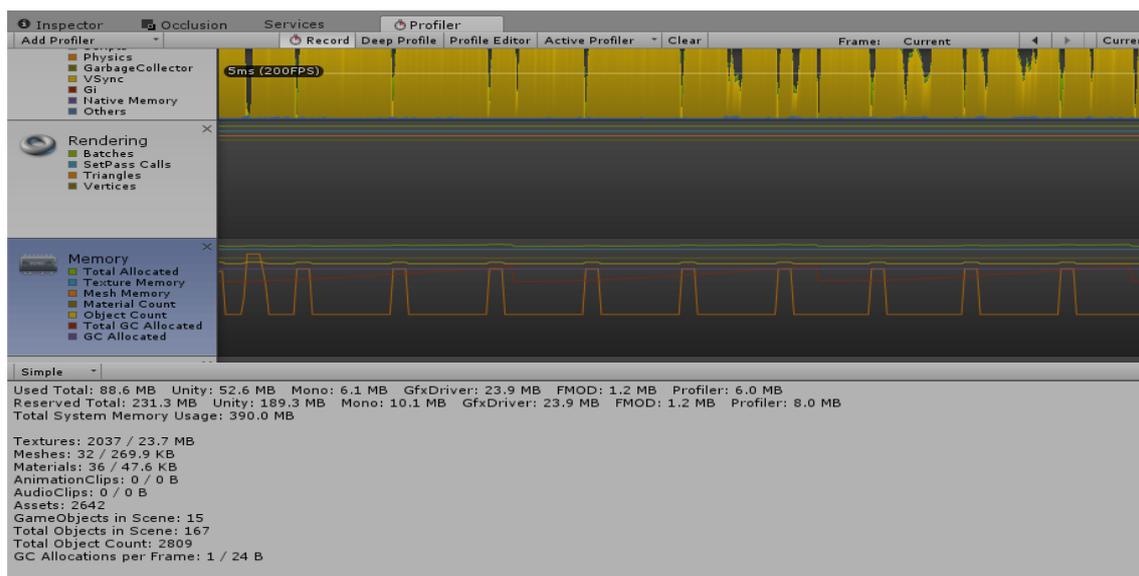


Fig. 3.33 Vista Simple de la memoria utilizada a través de Unity. (Captura de Pantalla Software Unity)

Unity, reserva grupos de memoria para ciertas asignaciones, esto con el objetivo de evitar pedirle al Sistema Operativo memoria con frecuencia. La cantidad de áreas reservadas y cuanto se usa está representada por las siguientes secciones. (Technologies, 2017)

- **Unity.** Es la cantidad de memoria rastreada por las asignaciones en el código nativo de Unity.
- **Mono.** Tamaño de la pila completa y el tamaño de la pila usada por Managed Code, en esta memoria, se recoleccióna la basura.
- **GfxDriver.** Es el monto estimado de la memoria que el controlador utiliza en texturas, rendertargets, shaders, y meshdata.
- **FMOD.** Se trata del uso de memoria calculado para los controladores de audio.
- **Profiler.** La memoria utilizada por el profilerdate, muestra los números que no siempre coincidirán con los mostrados por otros componentes como Task Manager, pues hay usos que no tienen seguimiento por el perfilador de memoria. Esto incluye memoria usada por algunos controladores y por código ejecutable.

También son mostradas algunas estadísticas de memoria para algunos assets/object más comunes, se incluye el conteo y la memoria utilizada (memoria principal y de video), algunas de estas son texturas, meshes, materiales, animaciones, audio, y object count (número total de objetos creados). (Technologies, 2017)

3.5.5 SCRIPTING

Este servicio dentro de Unity, es de gran ayuda para los desarrolladores, pues aunque Unity permite controlar el comportamiento de los GameObjects asignándoles componentes predefinidos y mediante la configuración de sus propiedades a través de una interfaz gráfica, muchas veces para los objetivos de un proyecto estas características son insuficientes si se pretende aplicar una mecánica más compleja.

Para satisfacer estas necesidades, Unity ofrece la posibilidad de implementar scripts. Un script es un programa relativamente pequeño que se puede asociar a un gameobject, como si fuese otro componente del mismo, añadiéndole alguna funcionalidad concreta que se haya definido por el autor del script. Es precisamente aquí donde radica la libertad de programar las acciones de cualquier aplicación que se cree, permitiendo mejorar las herramientas que ofrece el programa. (Murray, 2014)

En un contexto técnico, los scripts pueden ser explicados como un fichero de texto plano donde se han escrito procedimientos y órdenes en alguno de los lenguajes de programación que Unity permite, estos se van ejecutando según se haya configurado el orden, por ejemplo al ejecutarse un evento como un impacto o un movimiento, o en respuesta a alguna acción del usuario. (Murray, 2014)

Es por esto que la creación e implementación de scripts es fundamental en la creación de cualquier aplicación, incluso la más sencilla necesita scripts para responder a los estímulos de entrada que genera el usuario y asegurar que los eventos se ejecuten en el momento indicado. Otra característica relevante de los scripts, es la posibilidad de ser

usados para crear efectos gráficos, controlar el comportamiento físico de los GameObjects, incluso, se pueden alguna técnica de Inteligencia Artificial para el/los personaje(s) de la aplicación. Precisamente, esta opción es parte del desarrollo del presente trabajo, pues se desarrollara un agente inteligente reactivo en auxilio al modelo comportamental e instruccional del usuario en el EV.

Unity utiliza un tiempo de ejecución estándar mono para scripting, este todavía controla sus propias técnicas y prácticas para poder acceder al motor de juego desde los scripts. También, permite al desarrollador activar y desactivar eventos de la aplicación o juego, cambiar propiedades del componente en el tiempo y responder a las acciones del usuario de la forma que se determine.

Los scripts son considerados assets, sin embargo, a diferencia de la mayoría de los assets, estos son creados directamente desde Unity.

3.5.5.1 Lenguajes de Programación permitidos en Unity

Para generar un entorno virtual lo más realista posible, todos los componentes de este deben de trabajar en conjunto, comunicándose óptimamente entre sí, en RV a estos componentes se le denominan “objetos”, que pueden tener comportamientos específicos dentro de la escena. Para determinar las acciones que se ejecutaran dentro del entorno virtual y comportamiento de cada objeto. Para ello, Unity maneja algunos lenguajes de programación que por su estructura nos permite mayor control dentro del entorno virtual.

3.5.5.1.1 Lenguaje C#

C# es un lenguaje de programación simple pero eficaz, diseñado para escribir aplicaciones empresariales. El lenguaje C# es una evolución de los lenguajes C y C++. Utiliza muchas de las características de C++ en las áreas de instrucciones, expresiones y operadores.

C# presenta considerables mejoras e innovaciones en áreas como seguridad de tipos, control de versiones, eventos y recolección de elementos no utilizados (liberación de memoria).

En el desarrollo de cualquier proyecto en RV es importante tener en cuenta que software se va a necesitar, ya que los lenguajes de alto nivel como lo es C#, pueden ayudar a ejecutar opciones más específicas para cada objeto. En C#, todo puede ser tratado como un objeto, por su similitud con Java, es considerado programación orientada a objetos (POO).

3.5.5.1.2 JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas (interactivas). Una página web dinámica o interactiva, es aquella que incorpora diversos efectos en el contenido como: texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. Es decir, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

La inigualable popularidad de JavaScript como lenguaje de programación de aplicaciones web se ha extendido a otras aplicaciones y otros entornos no relacionados con la web. Por ejemplo, Unity es un motor de juegos que permite incluir y desarrollar scripts utilizando JavaScript, que dentro del entorno denomina UnityScript, los cuales pueden activar/desactivar eventos del proyecto en desarrollo, modificar propiedades del componente en el tiempo y responder a la petición del usuario de la forma que se desee.

3.5.5.1.3 Boo

El lenguaje Boo, está diseñado e implementado principalmente para el motor de juegos Unity. Boo es un lenguaje orientado a objetos de tipos estáticos escrito para el CLR, sintaxis inspirada en Python y hay especial énfasis en la productividad y extensibilidad del lenguaje y su compilador

Sus principales características son: dinamismo, sus generadores, multi-métodos, macros, closures, currificación, Funciones de primera clase, etc.

3.5.5.1.4 Unity Script

En Unity se emplea un lenguaje especialmente desarrollado para la plataforma, que aunque tiene cierta similitud con JavaScript en su sintaxis, también tienen importantes diferencias. Un ejemplo de esto, es que UnityScript implementa clases, mientras que JavaScript maneja objetos, sin el concepto de clase como tal (aunque de cierta manera se simulan).

Las diferencias entre UnityScript y JavaScript, son primordialmente semánticas. Es mejor mantener a UnityScript como un lenguaje meramente de unidad para evitar confusiones de utilidad y manejo.

3.5.5.1.5 Tabla Comparativa y Elección Final

Aunque todos los lenguajes que soporta el motor de juego Unity tienen estructuras similares, la sintaxis cambia, al igual que ciertas propiedades que resultan más útiles que otras. En datos al respecto de la utilización de estos lenguajes en Unity existe, una gráfica al respecto que contemplan los 3 lenguajes básicos de desarrollo en Unity.

En la *Fig. 3:34* se muestra un porcentaje bajo en la utilización del lenguaje Boo, el cual requiere del uso de una cantidad de recursos computacionales considerable, por lo que los desarrolladores de juegos en Unity, han dejado de apoyar el desarrollo de herramientas con este lenguaje. (Blogs, 2014)

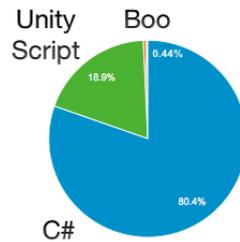


Fig. 3.34 Porcentajes de aplicación de lenguajes de programación en Unity. (Unity Blogs, 2014).

Las estadísticas indican que el mayor trabajo de desarrollo en programación con Unity, es mediante el lenguaje C#. Como consecuencia de esto, la documentación respecto al desarrollo de Assets, para implementar en el desarrollo de proyectos, y soporte de estas son predominantes en foros y páginas oficiales. En el caso de UnityScript, lenguaje que fue desarrollado específicamente para la plataforma, también existe gran cantidad de herramientas y soporte para el desarrollo de estas, sin embargo, gran parte de la comunidad sigue optando por otras opciones de programación disponibles. (Blogs, 2014)

Actualmente, la página oficial de Unity, muestra bastante material en C# y otro lenguaje contemplado en el presente proyecto, se trata de JavaScript, que por el contrario de lo que se podría esperar, tiene una estructura similar a C#, y suele brindar como ventaja la optimización de código. En la siguiente tabla se contemplan los más relevantes lenguajes encontrados en la literatura consultada a lo largo de la presente documentación. (Blogs, 2014)

PROPIEDADES	UNITY SCRIPT	BOO	JAVASCRIPT	C#
Documentación disponible	MEDIO	BAJO	ALTO	ALTO
Soporte	MEDIO	BAJO	ALTO	ALTO
Assets	BAJO	BAJO	ALTO	ALTO
Optimización	MEDIO	MEDIO	ALTO	ALTO
Gasto de Recursos Computacionales	BAJO	MEDIO	MEDIO	MEDIO

Tabla 3.11 Tabla comparativa de lenguajes de desarrollo en Unity. Elaboración propia.

C APÍTULO IV: ARQUITECTURA Y METODOLOGÍA DE

DESARROLLO

El diseño es una forma de vida, un punto de vista. Envuelve todo el complejo de la comunicación visual: talento, capacidad creativa, habilidad manual y conocimiento técnico. La estética y la economía, la tecnología y la psicología están intrínsecamente relacionadas en el proceso.

- Paul Rand

4.1 DESCRIPCIÓN DE LA METODOLOGÍA IMPLEMENTADA

En el desarrollo de cualquier sistema informático, como en este caso un entorno virtual con técnicas de inteligencia artificial, es de suma importancia para el éxito de su desarrollo, la implementación de una metodología adecuada a los objetivos del mismo.

Existen diferentes tipos de metodologías y cada una de ellas contiene sus propios métodos, procedimientos y herramientas con los cuales se pretende cumplir con ciertos objetivos dentro de un sistema, entre los cuales se encuentran: (Castellanos, 2012)

- Definir actividades que se realizaran en el sistema
- Unificar criterios en la organización para el desarrollo
- Proporcionar de puntos de control y revisión
- Asegurar el equilibrio y calidad del desarrollo y del sistema en general
- Satisfacer las necesidad del tipo de usuarios al que va enfocado el sistema
- Conseguir un mayor nivel de rendimiento y eficiencia del personal asignado al desarrollo
- Adaptarse a los tiempos y costos planteados en la planificación
- Crear adecuadamente la documentación relacionada con el sistema
- Proporcionar un mantenimiento sencillo posterior del sistema creado

En el presente proyecto, dado los objetivos que se definieron al principio, se utilizara la **Metodología Orientada a Prototipos**. La cual, permitirá ver los resultados genéricos en un corto plazo, y así seguir con el desarrollo de un método específico para generar la interactividad dentro del EV (Fig. 4.1).

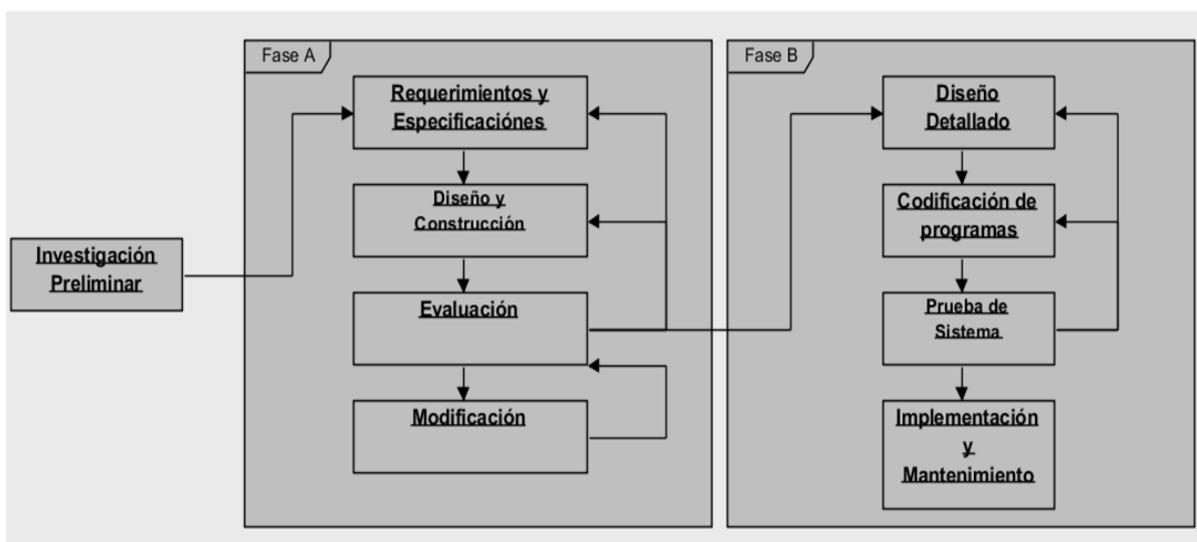


Fig. 4.1 Esquema de la Metodología orientada a Prototipos. (Elaboración propia).

4.1.1 INVESTIGACIÓN PRELIMINAR

Este apartado dentro de la metodología corresponde a la selección del sitio del cual se realizara el levantamiento tridimensional y la justificación del mismo. Como ya se ha mencionado al principio del presente proyecto, la problemática principal que se aborda es el bajo índice de interés y asistencia, que han proporcionado distintas encuestas a nivel nacional y a nivel local (específicamente datos de una encuesta realizada en la Ciudad de México). Se eligió como lugar de estudio la **Ciudad de México** por ser una de las ciudades más transitadas del país y donde se localizan muchos inmuebles culturales donde se imparten distintos talleres/cursos/certificaciones en temas relacionados. (Cultural, 2016)

Con 240 casas y centros culturales, y 8, 851,080, la Ciudad de México ofrece las opciones necesarias para empezar a crear una conciencia y un interés serio por los recursos culturales que se ofrecen al público. (Cultural, 2016) Entre estos inmuebles culturales se ha elegido el **Centro Nacional de las Artes (CENART)**, por ser uno de los espacios que conjuntan a su vez, centros de investigación e instituciones educativas artísticas y culturales de gran relevancia, específicamente, este proyecto se enfoca al centro de investigación denominado “**Centro Multimedia**”.

El Centro Multimedia es parte de CENART, órgano de la Secretaria de Cultura en México. Se trata de un área dedicada a la formación, investigación y experimentación de prácticas “artístico-culturales” que implementan nuevas tecnologías. Las actividades realizadas en este centro, se dividen en 5 espacios de desarrollo, los cuales son: **realidad virtual y videojuegos, publicaciones digitales, audio, imágenes en movimiento, gráfica digital, investigación en arte y tecnología, e interfaces electrónicas y robótica.** (Multimedia, s.f.)

Cuenta con varios proyectos académicos y artísticos nacional e internacionales, imparten algunas certificaciones en temas relacionados, se desarrollan dinámicas que generen nuevos usos y herramientas para los creadores y productores, culturales, etc.

Como se puede apreciar, es un centro muy completo de desarrollo cultural, es por ello que resulta de fundamental importancia la divulgación de estos conocimientos para el público en general, y así, dar espacio a la retroalimentación y fomentar la innovación artística-tecnológica en confluencia de variadas disciplinas a fines.

Un sistema de un entorno virtual 3D, posibilita la divulgación del conocimiento desde una visita no presencial, es decir, el usuario puede interactuar con el entorno simulado a distintos niveles de inmersión, en este caso por tratarse de una aplicación de escritorio el factor “inmersivo” es muy básico, sin embargo, la aplicación vista desde un dispositivo especializado para RV, como los Virtual Reality Glasses resulta una experiencia mucho más realista.

La aplicación está destinada, sobre todo, a personas interesadas en conocer este tipo de instituciones culturales de gran esparcimiento del conocimiento, que no cuenten con

los recursos monetarios, de tiempo, y sobre todo, geográficos, para hacer posible que desde cualquier lugar donde se ejecute esta aplicación, siempre y cuando se cumplan con las especificaciones para visualizarla (dispositivo electrónico, sistema operativo, memoria ram, etc.), los usuarios puedan hacer una visita virtual.

Un factor importante en un paseo virtual 3D, es la interactividad entre el usuario y este, para ello, se generara un agente inteligente reactivo, que le dará mayor independencia al usuario para recorrer las instalaciones del ya mencionado inmueble, sin tener que seguir un comportamiento lineal.

Retomando la estructura de la metodología, después de la investigación preliminar, resulta bastante conveniente para el desarrollo del presente proyecto las **dos fases en las que se divide**, las cuales fácilmente se pueden adaptar a las etapas que conforman el entorno virtual planteado. Tal adaptación derivo de la siguiente manera:

4.1.2 FASE A)

Las etapas que conforma esta fase son requerimientos y especificaciones, diseño y construcción, evaluación y modificación, las cuales están orientadas al desarrollo poligonal y visual del entorno virtual. Al término de esta fase se puede evaluar como prototipo, ya que representa la interfaz que maneja el usuario.

4.1.2.1 Requerimientos y Especificaciones

Como ya se mencionó, la principal justificación de este proyecto son los bajos índices de interés y asistencia por actividades y sitios culturales en temas específicos, por lo cual se plantea implementar una aplicación de un entorno virtual 3D que permita interactuar al usuario con los objetos que se encuentren en escena, los cuales a su vez le irán brindando información acerca del entorno.

Una característica fundamental en un entorno virtual 3D es la interactividad, el nivel de libertad que tenga el usuario dentro del entorno, por lo cual se implementara a su vez, un agente reactivo para poder dotar al usuario de movimientos los más naturales posibles.

Es importante destacar el nivel “básico de inmersión”, que está representado por el diseño del agente reactivo, el cual se encuentra simbolizado por el usuario y al mismo tiempo el usuario tiene una vista en primera persona para aumentar la sensación de estar situado dentro del entorno.

En cuanto las **especificaciones** del presente proyecto, se determinó un enfoque particular a características específicas dentro del entorno virtual, por ejemplo:

- Luces, sombras
- Calidad de texturas
- Calidad en los modelos poligonales
- Manejo de colisiones

- Simulaciones físicas
- Una interfaz intuitiva
- Interacción entorno-usuario
- Creación del agente reactivo
- Buen rendimiento en cualquier ordenador/computadora

Para lo cual es fundamental manejar herramientas de alta calidad, las cuales fueron seleccionadas, como ya se mencionó en el **capítulo III**, por las características de factibilidad, tiempo, conocimiento, etc., para obtener los resultados esperados. Las herramientas consideradas son:

- Blender
- Unity
- Photoshop
- Programación para scripts de interacción: C#
- Estructura de programación orientada a agentes.

Algunos, son requerimientos utilizados para realizar detalles como planos, fondos, ilustraciones dentro del programa por lo cual no se documentaron a fondo, sin embargo se irán mencionando a lo largo del desarrollo.

4.1.2.2 Diseño y Construcción

Esta etapa se enfocara a generar el entorno con el que interactúa el usuario principalmente que es el paseo virtual el cual está conformado a su vez por diversas variantes dentro de su de diseño y construcción

El diseño consiste en la representación plana de la arquitectura del lugar del cual se creara un modelo virtual 3D, para lo cual se necesitan

- **Planos.** Estos reflejaran los espacios incluidos dentro del inmueble, posiciones de puertas, ventanas, escalones, escaleras, muros, etc., y medidas de todos estos componentes.
- **Fotografías de arquitectura.** Mostraran el aspectos físico de los componentes de la arquitectura, detalles para ayudar a la concordancia del modelo en general.
- **Fotografías de texturas.** Se tomaran fotografías de alta calidad para recrear las texturas, dentro del paseo virtual, de muros, ventanas, objetos, pisos, puertas, etc.
- **Objetos dentro de los espacios.** Se contemplaran los modelos correspondientes a los muebles, plantas, objetos en general que se encuentren en cada uno de los espacios contemplados para el paseo virtual.
- **Materiales.** Se tomara en cuenta los materiales con lo que se elaboraron objetos, ventanas, vidrios, puertas, pisos para dar dotarlos de características realistas.

El proceso de modelado incluye todas estas variantes para la creación de cada uno de sus componentes. El modelo habla de la **geometría poligonal**, la cual consiste en la creación de polígonos tridimensionales los cuales son la base del modelado tridimensional de los elementos incluidos dentro del entorno, conocidos formalmente como objetos. En el lenguaje de los gráficos 3D por computadora, este modelo es un archivo que contiene información requerida para el proceso de “render”, el cual da las características que se podrán apreciar del objeto en 3 dimensiones, los cuales son:

- **Geometría.** Forma del objeto
- **Atributos de superficie del objeto en cuestión.** Esta información permite que el objeto se coloree de forma adecuada de manera que cree el factor realista en el objeto y se perciba de determinado material(vidrio, metal, plástico, etc)

Para el ordenador, la información de la geometría del modelo creado, especifica las “superficies” del objeto, enumerando los polígonos planos que comparten lados y vértices. Así mismo, se puede concebir que un modelo describe una malla, conocida en el ámbito de la graficación 3D, como mesh. Existen dos tipos principales de modelos 3D:

- **Modelos representados por polígonos.** Uno de los sistemas aceptados por los ordenadores para representar cualquier estructura son los ya mencionados, polígonos. Se considera un polígono a cualquier forma cerrada y plana (donde su primer y último vértice coincide perfectamente). Una figura genérica, por ejemplo: un cubo, está conformado por 6 caras, y cada una de estas es un polígono (cuadrado), en una pirámide, su base es también un polígono, pues es cuadrada. Incluso, las formas que están constituidas por líneas curvas también pueden ser representadas por polígonos, esta situación se ilustra en la vida cotidiana cuando se observa un balón de futbol que está compuesto por 12 pentágonos y 20 hexágonos.
- **Modelos definidos por sus curvas matemáticas.** Actualmente existen otras alternativas de modelado en donde el usuario no trabaja con polígonos, sino que es posible reemplazarlos utilizando superficies curvas definidas matemáticamente (p. ej. los llamados **Patch y Nurbs**). Este tipo de modelo funciona mediante una circunferencia que, si bien podría ser representada por un polígono de muchos lados, es representada como una función matemática entre dos variables ubicadas en un plano **X** y **Y**. Estas no son visualizadas por el usuario como fórmulas matemáticas complejas, es decir, funcionan de la misma forma que un programa vectorial donde resulta simple trazar líneas curvas perfectas, ya que en un modelador poligonal se disponen de diversas herramientas capaces de crear superficies curvas complicadas.

Dentro de la arquitectura del presente proyecto, y según los requerimientos de la arquitectura, el modelado se basará en la **creación, modificación y transformación de polígonos**.

Al respecto de trabajar con polígonos se tienen que tener en cuenta algunos aspectos básicos para poder manipular un modelo 3D poligonal, la manera de transformar su geometría para obtener el aspecto deseado basado en un objeto con ciertas características, es modificando sus vértices, extruyendo sus polígonos, dicho de otra manera, manipulando sus subcomponentes: vértices, aristas, bordes, polígonos y elementos.

Otro aspecto importante, son las **normales del polígono** modelado en 3D, un polígono posicionado en un sistema de coordenadas tiene una única orientación necesaria, es decir, tiene una primera vista en un único sentido; existe un vector imaginario que nace desde la superficie del polígono, y perpendicular a esta, el cual es llamado **normal** del polígono. Las normales resultan importantes en un modelo ya que determinan la parte visible de un polígono, que a su vez representa un objeto, dado que cada polígono podría tener incorporadas dos normales o más, apuntando en direcciones opuestas, la elección del lado hacia el cual la normal se proyecta establece la cara expuesta del polígono, la que será visible por el/los usuario(s).

Finalmente, la **transformación y modificación** de los polígonos, el concepto de transformación se refiere a la acción de mover, rotar y/o escalar, incluso de implementar la función de espejo, a un objeto dentro de la escena. Por otro lado, la modificación de los objetos habla de la aplicación de funciones que curvan el modelo, lo afilan, lo achatan. Ambos procesos están relacionados directamente con un pivote único que representa el sistema de coordenadas, el cual está presente en cualquier objeto creado. Este pivote se puede contemplar como punto único de referencia del objeto.

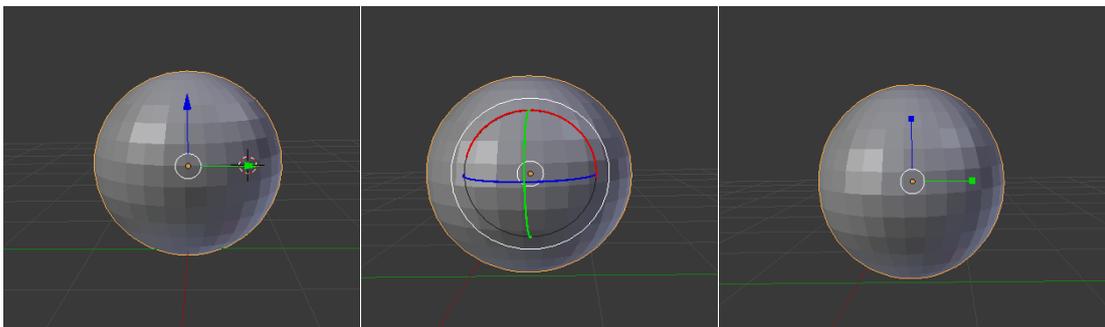


Fig. 4.2 Esfera aplicando diferentes funciones de transformación: mover, girar, escalar respectivamente. (Elaboración propia mediante el software Blender).

Al trabajar con polígonos también se reducen los tiempos computacionales, esto se debe a que el modelo de un polígono común está conformado por 4 vértices, y el motor de render siempre realiza las operaciones con el polígono más elemental que

está conformado por 3 vértices, de esta manera el motor de render¹² fraccionara el polígono en formas triangulares pues fraccionar un cuadrado en triángulos resulta más simple que fraccionar un triángulo en triángulos.

Cabe mencionar que en este proceso de modelado, cada vértice, de los polígonos creados, sea único en unir al polígono inmediato, es decir, al polígono más cercano de este pues el traslape de estos vértices implicaría consumir más procesos y operaciones en el motor de render. Las normales, también son primordiales pues son en base a quienes se determina la iluminación del ambiente, por lo cual se debe cuidar que apunten a la dirección correcta antes de exportar el modelo.

4.1.2.3 Evaluación

La evaluación consiste en poner a prueba el prototipo creado de manera general respecto a los requerimientos y especificaciones. Esto con el fin de detectar fallas tempranas en la aplicación y poder corregirlas para pasar a la próxima fase de la metodología de desarrollo. La evaluación se subdividió en algunas pruebas que aseguraran el funcionamiento del entorno virtual, las cuales son:

- **Prueba de navegación por la interfaz.** En el menú principal existen tres opciones, una enviara al usuario al paseo interactivo virtual, donde podrá conocer las instalaciones del inmueble e interactuar con ciertos objetos colocados estratégicamente para brindarle al usuario información clave acerca del espacio que está visitando, posteriormente, si el usuario quisiera saber más acerca del inmueble que está conociendo podrá elegir la siguiente opción donde habrá información acerca de las actividades que se imparten en los laboratorios, finalmente existirá una opción para salir de la aplicación.
- **Prueba comportamental.** La cual consiste en probar la libertad de movimiento del usuario con respecto al entorno en el que se mueve, específicamente, se utiliza la programación de un script basado en la estructura de un agente inteligente
- **Prueba de luces.** A lo largo del recorrido se han manejado luces artificiales y luz del ambiente, para una correcta visualización de texturas, colores, formas, etc., es importante comprobar que las luces estén en el orden correcto, con la intensidad adecuada para hacer el entorno lo más natural posible
- **Prueba de sonido.** Dentro del entorno es importante el sonido utilizado para la ambientación, el sonido de fondo, y el sonido que produce la interactividad con los objetos, incluso, el propio movimiento del usuario (p. ej. caminar)

¹² Proceso de producir imágenes desde una perspectiva de modelos tridimensionales, en una escena 3D, dicho de otra manera, se puede describir como el proceso de tomar una fotografía de la escena. Así mismo, una animación es una serie de renders secuenciados. (Mathieu Aubry, 2014)

- **Prueba de colisiones.** Durante el recorrido, y por el nivel de libertad que se brinda al usuario, este será capaz de recorrerlo en el orden que desee, sin embargo tiene que haber delimitaciones que involucren otros caminos no incluidos en el paseo, por lo cual es de suma importancia que el paseo se encuentre bien delimitado para no provocar un bucle cuando el usuario insista en acceder a zonas no permitidas
- **Prueba de movimiento y dirección.** El movimiento del usuario dentro del paseo virtual está determinado por las teclas básicas de movimiento, donde: ➔ derecha, ➜ izquierda, ⬆ arriba, ⬇ abajo, para desplazarse, mientras que para seleccionar una dirección para avanzar se utilizara el mouse del ordenador, se tomó la decisión de no incluir un avatar para representar al usuario, ya que una de las características básicas de un entorno virtual inmersivo (véase capítulo III), es precisamente que el usuario viva la experiencia virtual desde una perspectiva de primera persona.
- **Diseño y funcionamiento del menú.** El diseño de la estructura de funcionamiento de la aplicación resulta muy intuitivo gracias a al menú con títulos precisos.

Las pruebas de control son específicamente importante ya que dentro del paseo se encuentran definidos algunos eventos para que el usuario interactúe, sin embargo, es preciso verificar que el uso de las teclas de movimiento asignadas funcionan correctamente,

Aspectos Importantes a Considerar de una Interfaz Gráfica

Conocidas como GUI (Graphical User Interface), generalmente se controlan mediante el modo consola, ya sea interacción mediante el teclado del ordenador, o puede tratarse de una interfaz basada en texto. Es importante mencionar, que cualquier aplicación gráfica tiene implícito el uso de ciertos componentes, por ejemplo: contenedores, menús, botones, cuadros de texto, ventanas, imágenes, etc. (Arango, 2014)

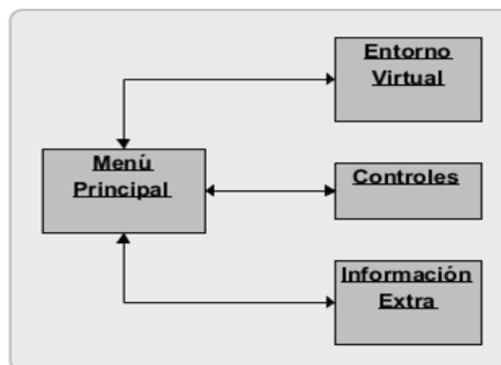


Fig. 4.3 Diagrama de interfaces del sistema. (Elaboración propia).

Como se observa en la *Fig. 4.3*, el diseño del menú está compuesto por las interfaces: **menú principal**, la cual permitirá elegir al usuario una de las tres opciones disponibles, **controles**, proporciona información acerca del movimiento dentro del entorno virtual, sobre teclas que se utilizaran e instrucciones de uso, **entorno virtual**, representación digital del inmueble seleccionado (Centro Multimedia) en el cual el usuario podrá interactuar, **información extra**, en este apartado la información acerca de las actividades realizadas en este espacio cultural se retoman de una forma más completa. El menú, desde un punto de vista más técnico, tiene como finalidad ayudar al usuario visualizar y manipular el paseo interactivo y sus atributos, la creación de la interfaz se compone de las siguientes aplicaciones:

- Ventanas gráficas para entrada y salida de datos
- Iconos
- Dispositivos de entrada (mouse, teclado)
- Interacción directa

Se tiene que comprobar que todas estas acciones se ejecuten de manera correcta para el funcionamiento óptimo de la aplicación.

Las pruebas, ya mencionadas, para verificar el correcto funcionamiento de la aplicación se hicieron con un ordenador de las siguientes características:

- Sistema Operativo Windows (Vista, 8, 10)
- Procesador Intel Core i7
- Memoria Ram 8GB
- 200 MB como mínimo en espacio de Disco Duro
- Tarjeta de video Intel HD Graphics 4400, memoria compartida 2048 MB.

Si la aplicación se ejecuta en un ordenador con características diferentes, podría tener un impacto no favorable en la calidad del entorno.

4.1.2.4 Modificación

En este punto se tiene ya elaborado el prototipo del resultado final de la aplicación, permitiendo conocer la estructura, interfaz y funcionamiento general para el usuario.

Si a lo largo de las etapas pasadas se detectaron errores o surgieron nuevas especificaciones y/o requerimientos, se modificarán de acuerdo a las nuevas incidencias. Es precisamente una de las ventajas de trabajar con una metodología orientada a objetos, ya que al enfocarse en los requerimientos de la aplicación puede reducir la incertidumbre ya que presenta resultados en una fase temprana de desarrollo para ser evaluados, con ello también quedan reducidos tiempos y costos.

4.1.3 FASE B)

Esta segunda etapa conforma la implementación de la geometría en el entorno, aspectos de interacción con el usuario implementando tecnologías de RV, y analizando últimos detalles de la interfaz en la integración de todas las etapas para su posterior ejecución.

4.1.3.1 Diseño Detallado

A lo largo de la construcción del prototipo del sistema, el diseño detallado queda implícito, pues este trata de la creación de los módulos encargados del funcionamiento del sistema, es decir, la programación que hace posible su funcionamiento complejo.

Por lo tanto, el sistema debe ser entonces rediseñado y documentado según los estándares de la organización y para ayudar a las mantenciones futuras. Esta fase de diseño técnico tiene dos etapas: por un lado, la producción de una documentación de diseño que especifica y describe la estructura del software, el control de flujo, las interfaces de usuario y las funciones y, como segunda etapa, la producción de todo lo requerido para promover cualquier mantención futura del software.

En esta etapa se definen los modelos de programación y la estructura necesaria para el sistema en general, como ya hemos visto el sistema en realidad virtual que se desarrollara es un Entorno Virtual 3D con un componente inteligente que representara al usuario dentro del entorno.

- **Interactividad dentro del entorno.** En el presente proyecto se dará un particular enfoque al desarrollo de la interactividad del entorno, aunque el software utilizado brinda muchas herramientas para este propósito, se desarrollara un algoritmo para crear un agente reactivo que permita al usuario una mayor libertad dentro del entorno sin perder de vista el objetivo.
- **Definición de eventos para los objetos del entorno.** Los objetos creados en el entorno virtual tienen definidos ciertos comportamientos, los cuales se ejecutaran con la interacción del usuario. Dado que uno de los objetivos del entorno virtual es proporcionar información acerca de lo que está viendo el usuario, se tiene que diseñar el repositorio de información dependiendo de cuán grande sea la cantidad de datos se utilizara una base de datos o archivos txt.
- **Estructura de navegación.** Una correcta estructura para la navegación el sistema, hará más intuitivo el manejo para el usuario, se cuenta con un menú donde puede ver la información de manera más detallada acerca del inmueble elegido, instrucciones de cómo moverse dentro del entorno y finalmente, la opción de visitar el entorno virtual disponible. Específicamente, la navegación dentro del entorno virtual será definida por los objetivos que se quieren alcanzar, aunque el usuario tendrá la libertad de elegir por donde quiere empezar y con qué quiere interactuar (dentro de las opciones

disponibles) la ruta de navegación tiene que cumplir con un objetivo, que es que el usuario conozca todas las instalaciones disponibles, los talleres, la información concreta que le proporcionara algunos objetos dentro del entorno, por lo cual serán restringidas algunas zonas, para hacer que el usuario dentro de su libertad tome el camino que debe tomar.

- **Documentos txt o Base de datos.** Este es una característica importante, pues es la técnica que se utilizara para guardar la información que estará disponible para el usuario. Sin embargo, no toda la información debe ser manejada a través de una B.D. ya que, debido a su simpleza, en algunos momentos se pueden utilizar archivos de texto plano.
- **Simulaciones.** Las simulaciones estarán divididas en dos módulos, el comportamiento de usuario será determinado por el algoritmo del agente reactivo que le proporcionara autonomía en sus movimientos. Por otro lado, las simulaciones del entorno en general, es decir, los objetos secundarios situados en el entorno con los que podrá interactuar el usuario, serán definidos por la configuración de sus propiedades básicas.
- **Animaciones.** En el caso de un sistema de realidad virtual las animaciones de los objetos son muy importantes, incluso, de los componentes de la interfaz, ya que una de las características principales es el enfoque dinámico, por lo cual se tiene principal atención a este aspecto, a bajo y nivel y alto nivel.

4.1.3.2 Codificación de Programas

La etapa de programación es una de las más importantes, pues te tienen que generar los algoritmos que sean capaces de satisfacer las necesidades del sistema. Tiene como principal objetivo, llevar a cabo lo estipulado en el diseño detallado.

4.1.3.3 Prueba de Sistema

Después de generar la programación del diseño detallado, se deberá poner a prueba el funcionamiento del sistema en general, principalmente en entorno virtual, pues es donde la interacción con el usuario es mucho más profunda. Los modelos en los que fue dividida la subetapa de simulación, deberán ser consistentes y armónicos respecto al sistema completo.

Se contemplara el cumplimiento de los cambios que se consideraron necesarios en el diseño detallado y la programación, para asegurar la corrección adecuada para el correcto funcionamiento respecto a los requerimientos estipulados al principio.

4.1.3.4 Implementación y Mantenimiento

La etapa final de la fase B, y de la metodología en general, es la implementación y mantenimiento. La instalación del sistema en formalidad para ser explorado, en este caso, el usuario que conoce el prototipo ya sabrá cómo manejarlo, y el usuario ajeno al proyecto deberá de explorar y manifestar que tan fácil fue navegar dentro del

sistema, si tuvo el nivel de inmersión esperado y si el desenvolvimiento dentro del entorno virtual, que es el componente más dinámico del sistema, ha resultado satisfactorio para conocer el inmueble del que se realizó el modelo tridimensional.

EL mantenimiento es una de las fases menos relevantes, pero necesarias para mantener el sistema funcional y si es necesario, realizar mejoras futuras. Sobre todo para este último propósito, ya que la fase A y la fase B se encargan del refinamiento del sistema en diferentes etapas, por lo cual ofrece mayor calidad en sus resultados finales.

Retomando, si se pretendieran hacer mejoras futuras, el proceso de prototipado tendría que ser repetido, que a su vez implicaría definir nuevos requerimientos. El sistema desarrollado mediante estas dos fases resulta ideal para satisfacer la atención que requiere cada componente que lo conforma, además que permite apreciar desde una etapa temprana un prototipo que muestra el funcionamiento grosso modo. Sin embargo, dado que implementara una técnica diferente para simular el comportamiento del usuario, existe la necesidad de utilizar un modelo de agentes inteligente para su creación.

4.1.4 MODELO DE INTEGRACIÓN DE UN AGENTE REACTIVO

Al hablar de agentes inteligentes en general, el número de actuadores que el módulo estratégico brinda, lleva en si muchas limitantes en las acciones que se pueden realizar en el entorno en el que se encuentra. En cambio cuando un agente forma parte del entorno virtual, debe cumplir con el control del avatar que representa al usuario, en este caso, el usuario es representado por el controlador en sí, pues una de las características de inmersión básica es que el usuario no pueda verse a sí mismo dentro del entorno, esto hace referencia a que se tiene una perspectiva en primera persona. (Arango, 2014)

Recordar que los agentes representan las entidades autónomas del sistema. Estos agentes pueden gestionar objetos que no estén representados dentro del EV, es decir, serán capaces de gestionar cuentas vinculadas al usuario, bases de datos, o cualquier otro elemento que no tenga representación virtual.

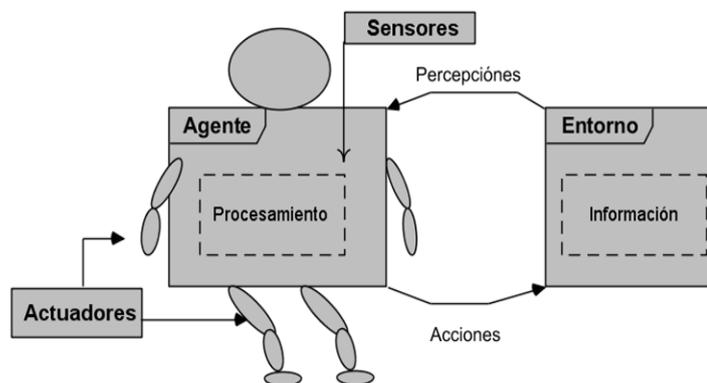


Fig. 4.4 Estructura de un Agente Inteligente Simple. (Elaboración propia).

Al implementar una arquitectura, deben de manifestarse el conjunto de primitivas disponibles para el agente, pues mediante estas se controlara en el entorno. Estas mediante su número y complejidad definirán el alcance del agente, por ejemplo, si se utilizara un avatar para el usuario y este tuviera la capacidad de gesticular, no habría necesidad de añadir emociones o control de gestos al agente, o como en este caso particular que no se usa ningún avatar, sino simplemente el propio agente inmerso en representación del usuario. (Thanos G. Stavropoulos, 2014)

Los sensores son fundamentales para percibir el entorno, y en su caso, pasa algo similar al caso anterior, ya que dependiendo la cantidad de cosas que proporcionen algún tipo de información apreciable por el agente, su comportamiento se adecua convenientemente más-menos “estados” existentes en el entorno. (Thanos G. Stavropoulos, 2014)

Para la creación de un agente inteligente inmerso en un entorno virtual, se debe tener siempre claro en qué tipo de situaciones participara el agente, de esta manera el EV podrá mandar la información necesaria a través de los sensores.

De manera general, al utilizar un agente inmerso en el EV, se necesitara conocer el entorno por el que se desplazara con el fin de poder ir de un lugar a otro sin presentarse obstáculos insuperables, además conocer dónde están los objetos con los que se podrá interactuar, dicho de otra manera, se necesitan conocer la ubicación de los actuadores.

Para satisfacer las necesidades interactivas del entorno respecto al controlador principal (representación del usuario), se ha elegido un tipo de agente, cuyas características, atiende perfectamente a los propósitos del presente trabajo.

Los **agentes reactivos**, son subdivididos en varias categorías según el enfoque de comportamiento, un agente reactivo simple actúa en función de las percepciones que está tomando en ese momento, sin contemplar las anteriores. Este tipo de actuación puede no ser tan útil ya que muchos entornos virtual requieren de guardar un historial para ir optimizando sus decisiones (p. ej. un juego de ajedrez).

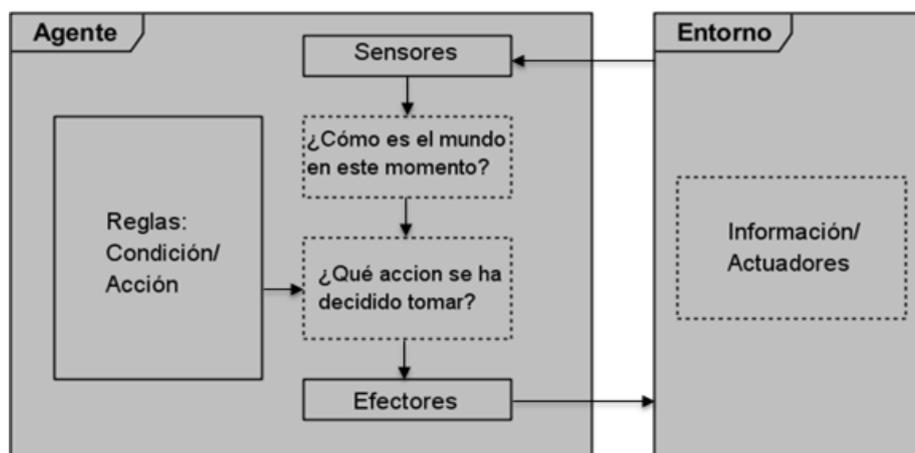


Fig. 4.5 Esquema de un Agente Reactivo Simple. (Elaboración propia).

La Fig. 4.5 muestra los componentes de funcionamiento de un agente reactivo simple, es, por definición, capaz de percibir su entorno y reaccionan ante los cambios que se producen en el mismo. El sistema encargado de la toma de decisión es, generalmente, quien determine el manejo de la variabilidad de situaciones del mundo simulado, detectando cambios que alteren el comportamiento en este, y desechando los datos irrelevantes.

Un claro ejemplo de esto, es el sistema reactivo de referencia, por (Reynolds, 1987), básicamente, funciona a partir de un sistema de reglas, el cual resulta ser una herramienta muy potente para crear un motor de inferencia capaz de tomar decisiones de sistema multiagentes en mundo dinámicos. En base a los cambios que se pueden efectuar en el entorno, los agentes, según Reynolds, serán auxiliados por técnicas de sensorización local para manejar la cantidad adecuada de información en cada ciclo de simulación, proceso que hace referencia al algoritmo de k-vecinos. Este tipo de sistemas han sido utilizados para el control comportamental de agentes, pues el paradigma de programación que proponen es adecuado para el control de un personaje inteligente. (Luz, 2014)

Mediante el análisis de los tipos de agentes reactivos que se encuentran en la literatura, se ha determinado que el más adecuado para utilizar dadas las especificaciones del software, es el modelo de agente reactivo basado en modelos.

Área de Desarrollo del Agente	Objetivo de implementación	Sensores	Actuadores	Entorno
Sistema en Realidad Virtual (Entorno Virtual Interactivo)	La divulgación de actividades y recintos culturales	Teclado, entrada de información acerca del movimiento. Mouse, selecciona el objeto con el que se quiere interactuar (de los objetos disponibles)	Flechas básicas de movimiento (↓,↑,←,→), manipulación de la orientación de avance según la perspectiva del controlador que representa al usuario.	Espacios inteligentes que determinan si han sido visitados, objetos interactivos que proporcionan información del entorno.

Tabla 4.1 Análisis del modelo del agente reactivo. (Elaboración propia).

Se trata de un modelo computacional que permite la simulación de acciones e interacciones de un individuo(s) autónomo dentro de un entorno, también permite definir los efectos que se produzcan en el sistema. En esta primera versión del sistema, el representante virtual del usuario en el entorno, está compuesto únicamente por un nodo con una determinada anchura y altura (espacio de colisión) que en interacción con otros

nodos de la escena, define el comportamiento del usuario, sin quitarle libertad pero cumpliendo con el objetivo (conocer todos los espacios y su información).

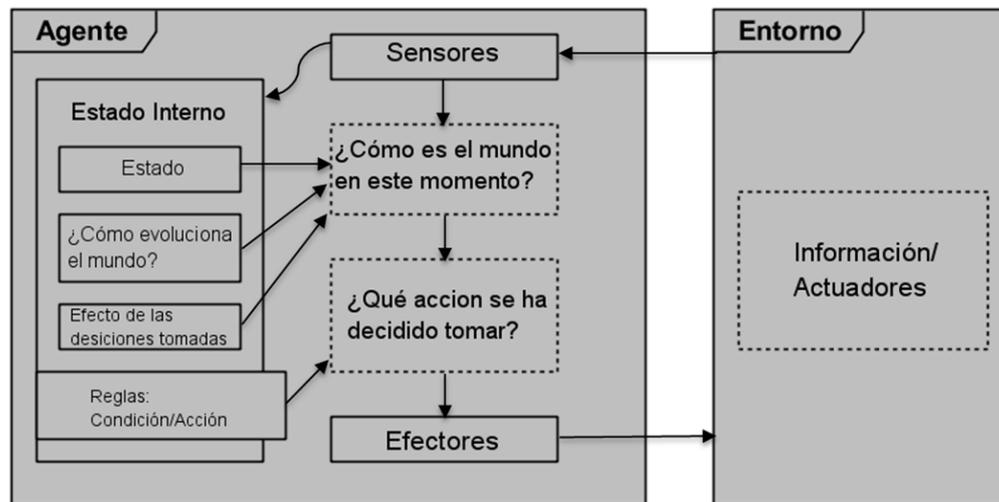


Fig. 4.6 Esquema de un Agente Reactivo basado en Modelos. (Elaboración Propia).

Es importante recordar, que un agente, independientemente de su aplicación, estará conformado por una arquitectura implementada a través de código, por lo cual es un pequeño programa, el cual actuara mediante una función que transforma secuencias de percepciones en acciones. (Andrea Bonsch, 2017). El agente reactivo que se desarrollara está basado en conocimientos, es decir, cuenta con una base de reglas (condición-acción) y hechos que incrementa proporcionalmente a la interacción, así mismo, esta base de reglas y hechos, funge como factor primordial en la toma de decisiones, tal como indica la Fig. 4.6. La cual muestra el esquema del agente reactivo basado en modelos, cuyos componentes en conjunto determinan su funcionamiento de la siguiente manera:

- **Estado Interno.** Este tipo de agente, requiere guardar algún tipo de perfil parcial del mundo virtual, la solución la implementa a través de algunos módulos que recolectan información al respecto del entorno que no son visibles, que en conjunto es lo que se denomina como “estado interno”.
 - **Estado.** Este módulo recopila información acerca de las propiedades del entorno, en respuesta a otro
 - **¿Cómo evoluciona el mundo?** Determina el cambio en la información de propiedades de un estado
 - **Efectos de las decisiones tomadas.** Vincula los cambios de un estado a otro, y la consecuencia que estas acciones trajeron consigo.
- **Reglas: condición-acción.** Es una etapa del proceso del agente de vital importancia, se trata del diseño de una base de datos las cuales contienen información (reglas) predefinidas que se van
- **Sensor.** Dentro del modelo de un agente reactivo, los sensores tienen un papel fundamental para la toma de decisiones, ya que estos son los que proporcionan la información necesaria para la evolución del sistema, pues es base a esta información que se permitirá la interacción con el usuario.

-
- **¿Cómo está el mundo en este momento?** Esta pregunta debe ser respondida mediante varios módulos que recopilan información acerca de los cambios que se han producido en el entorno, a partir de las acciones tomadas previamente aunados a los datos percibidos por los sensores que intervienen.
 - **¿Qué acción se ha decidido tomar?** En esta etapa se selecciona la acción que se va a implementar, basándose en la información previamente recibida.
 - **Efactor(s).** La decisión seleccionada es representada mediante el/los dispositivos efectores (monitor, bocinas)
 - **Agente.** En conjuntos todos estos módulos hacen posible el funcionamiento del agente inverso en el entorno virtual como representación del usuario. El agente, en este nivel, tiene una inteligencia limitada.
 - **Entorno.** El entorno es un factor muy importante en la interactividad con el usuario, ya que proporciona los objetos con los que, según la configuración y/o programación de sus propiedades,

mientras se cumpla condición

estado= interpretar_entrada (percepciones)

regla= vinculacion (estado, reglas)

ejecución de respuesta (regla, acción)

fin mientras

Tabla 4.2 Seudocódigo del algoritmo d un Agente Reactivo Basado en Modelos. Elaboración propia.

4.2 IMPLEMENTACIÓN Y DESARROLLO

Para el desarrollo de este sistema, se tiene prevista la aplicación de dos arquitecturas, la primera esta formalizada y presenta de manera general el funcionamiento del sistema y la segunda muestra el comportamiento de los procesos internos. Se propone trabajar con una estructura que fusiona ambas arquitecturas, ya que no se ha formalizado una arquitectura específica para el desarrollo de este tipo de sistemas en realidad virtual, y que además presenten como complemento una técnica de Inteligencia Artificial, en este caso específicamente, un agente inteligente.

4.2.1 ARQUITECTURA DE SISTEMAS DE REALIDAD VIRTUAL

Existen arquitecturas genericas para el desarrollo de sistemas en RV, en las cuales siempre son considerados imprescindibles algunos componentes. La siguiente arquitectura consiste en determinar los elementos hardware y software que determinaran el comportamiento de la presente aplicación, estableciendo relaciones entre los elementos correspondientes y cómo estos afectan los diferentes modelos de interacción, simulación e inmersión del sistema.

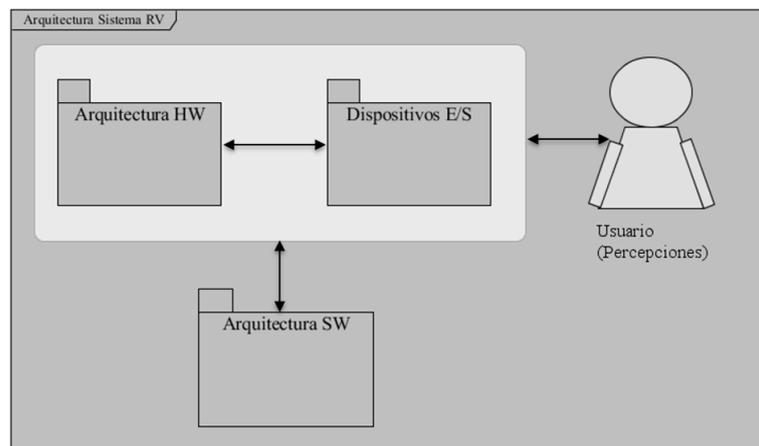


Fig. 4.7 Arquitectura de un Sistema de Realidad Virtual. (Adaptación de Brunet, 2012).

Es importante especificar estos tres módulos, dado del tipo de sistema de RV que se esta trabajando, tal como se ha mencionado a lo largo del documento, se desarrollara un EV basado en un agente inteligente, por lo cual a medida que se vayan desarrollando los componentes de la presente arquitectura general se irán vinculando, con la arquitectura que permitirá desarrollarlo.

4.2.1.1 Componentes de un Sistema de RV

Tal como se muestra en la *Fig. 4.7*, un sistema de realidad virtual se caracteriza por implementar elementos de hardware y software. De los cuales, se le da una particular importancia a los componentes hardware, sin embargo, dado la naturaleza de la aplicación en RV de escritorio que se esta desarrollando, se le dará mayor enfoque al desarrollo de la arquitectura software. Sin olvidar los componentes hardware que seran utilizados, quienes aportarán la información requerida mediante los sensores implicados.

En este contexto, el desarrollo de componentes software son los más relevantes en este trabajo, ya que se aborda el desarrollo del modelo geométrico 3D y los programas desarrollados para la simulación física (perspectivas de la cámara virtual, detección de colisiones, etc), comportamental (comportamiento de la representación virtual del usuario), de navegación (rutas marcadas, nivel de libertad, etc) y recolección de datos para los procesos internos del sistema.

Dispositivos de Entrada (sensores)

Estos dispositivos se encargan de capturar las acciones del usuario y enviar esta información a los módulos correspondientes para ser procesados. Los dispositivos considerados para este sistema son el teclado y el mouse, cuya interacción con el usuario permitirá adquirir los datos requeridos para el correcto funcionamiento de la técnica elegida para el modelo comportamental e instruccional del usuario, el cual es fundamental dado el enfoque del sistema. En este contexto, el presente sistema de realidad virtual cuenta con un nivel de inmersión básico, ya que no se tienen considerados muchos de los dispositivos hardware más representativos en esta área (guantes, cascos, etc).

Sin embargo, es importante mencionar que al contar con un modelo tridimensional la presente aplicación puede ser mejor apreciada con la utilización de lentes de realidad virtual.

Dispositivos de salida (efectores)

Los dispositivos de salida, son las representaciones perceptibles por el usuario de las respuestas producidas por la técnica de procesamiento implementada para la interacción con el usuario (audio, video, etc.) Estas representaciones son generadas por la computadora como estímulos para los sentidos del usuario (imágenes, sonido, etc.). Para estos efectores también existe una clasificación definida por el sentido del ser humano, al que va dirigido (visuales, auditivos, táctiles, etc). En este sentido, los efectores del sistema en cuestión, se simplifican a un monitor y bocinas (audifonos), de igual manera, existe la opción de visualizar el sistema por medio de lentes de RV.

Computadora (Arquitectura Hardware)

Sin duda, la computadora es el personaje principal en la creación de este tipo de sistemas de escritorio, pues es el responsable de efectuar la simulación virtual y permitir la interacción con el usuario, mediante un modelo geométrico 3D, y los programas encargados de los modelos de comportamiento, navegación, cinemático, etc., que engloban la simulación física y sensorial en el entorno. No obstante, el procesamiento más pesado para la computadora es la simulación visual, es decir, la representación de imágenes a partir del modelo 3D generado. Las computadoras utilizadas para la generación de realidad virtual deben contar con una buena tarjeta de procesamiento gráfico, ya que mediante ésta, se cubre la mayor parte de las etapas del proceso de visualización.

Módulo Geométrico Tridimensional

En un sistema de RV es imprescindible permitir al usuario desplazarse por el entorno e interactuar con este, al tiempo que se aprecia el entorno desde diferentes ángulos. Para ellos es necesario tener una representación tridimensional, como se ha mencionado a lo largo del capítulo, el cual permitirá realizar cálculos de imágenes, activación de sonido, cálculo de colisiones, movimientos físicos, etc.

Técnicas de tratamiento de datos de entrada

Tal como se vio en el *Capítulo III*, existen variadas técnicas para desarrollar el algoritmo que permitirá el tratamiento interno de la información recibida del exterior, la cual permitirá la interacción del sistema con el usuario. Los módulos encargados de recolectar estos datos por medio de los sensores para su próximo proceso, se han descrito a través de un modelo de agente inteligente reactivo. Así mismo, se incluyen los controladores de dispositivos físicos (contemplados), por ejemplo, datos de posición y orientación del usuario virtual, generalmente el proceso consiste en transformar estos datos en un sistema de coordenadas, los cuales, posteriormente, serán filtrados y procesados por el algoritmo para reemplazar mejores posiciones respecto al objetivo final. De esta manera, el sistema requiere como información exterior, la dirección de desplazamiento, las vistas dentro dentro del modelo, posición actual, etc. (Brunet, 2012)

Simulación física y sensorial

Dentro de la técnica utilizada para el procesamiento del sistema de RV, se pueden encontrar módulos que determinan las modificaciones necesarias en la representación virtual de la escena, en base a las acciones que tome el usuario y la evolución interna del sistema. Un claro ejemplo, es cuando el módulo que se encarga de recolectar los datos, informa que el usuario tiene que hacer el movimiento para abrir una puerta (en caso que se cuente con un avatar o representación detallada del usuario dentro de la escena), mediante la transformación geométrica correspondiente al objeto.

Dentro de este tipo de funciones, la más básica calcula los parámetros de la cámara virtual en tiempo real, de acuerdo a los movimientos del usuario. Por otro lado, estos módulos también se encargan del cálculo de simulaciones físicas, como: colisiones, modificaciones, deformaciones, comportamientos y otras actualizaciones que pueda tener el entorno virtual en el tiempo que esta siendo visualizado.

También son contemplados módulos que se encargarán de la representación digital de las imágenes, estos son los que se encargarán de traducir señales para enviar finalmente estímulos para los sentidos del usuario. Particularmente, la simulación visual, tratándose de una aplicación de escritorio, se basa en algoritmos de visualización en tiempo real del modelo geométrico, lo cual quiere decir, que su entorno geométrico va cambiando con cada acción que implementa el usuario ya que es necesario ajustar la imagen a consecuencia.

Los algoritmos de visualización necesarios para las tareas de simulación, son diseñados y desarrollados dentro de la conceptualización de ciertas técnicas utilizadas para los sistema de RV, por lo cual dentro de su estructura es muy importante que esten representados estos módulos para detectar movimientos y acciones del usuario que desencadenen los eventos programados.

4.2.1.2 Funcionamiento de un Sistema de Realidad Virtual

En este apartado se describirá el funcionamiento general de un sistema de realidad virtual convencional (arquitectura software), de esta manera se podra apreciar de modo más preciso la relación que tienen los elementos hardware y software con el procesamiento interno del sistema.

Tal como se ha explicado a lo largo del *Capítulo IV*, toda acción implementada por el usuario motiva una respuesta en el entorno, esto se debe a que estas acciones son registradas en tiempo real por los dispositivos de entrada (sensores), los cuales recolectan la información solicitada, tal como se menciona anteriormente. Sin embargo, se sabe que esto datos son solicitados por módulos que procesan esta información, pero ¿quién determina que información va a pedir estos módulos?, ¿para qué?, ¿qué respuestas generaran?. Bien, el procesamiento interno de un sistema se basa en programar un algoritmo diseñado para cumplir con cierta tarea u objetivo, y este, a su vez, ayudar a cumplir un objetivo más grande, ese es el panorama que se tiene al tratarse de un sistema dinámico.

Resumiendo, la información que se recopila es procesada por la computadora por lo módulos recolectores de información (datos de entrada), cuando se tienen la acciones del usuario en un formato adecuado, se procesan por las secuencias programadas encargadas de la simulación, quienes determinarán los posibles cambios para el entorno virtual. Por ejemplo, si el usuario quiere correr para desplazarse, el modelo geométrico tendrá que virtualizar una imagen de este, mucho más rápida para que vaya acorde con el movimiento (perspectiva, dirección, etc) del representante virtual del usuario. La información digital que se genera para notificar posición, dirección, orientación del usuario, es traducida en una señal de video por la tarjeta gráfica de la computadora, para finalmente, presentarse como una imagen física en el dispositivo de visualiación.

Este proceso actúa en tiempo real, es decir, cumple un ciclo que se repetirá continuamente con una valta elocidad para que el usuario no sea capaz de percibir el tiempo que se toma en procesar la información enviada, ya que esto disminuye la interactividad con el sistema. Para ello es importante tener en cuenta y diferenciar las distintas frecuencias de actualización. Este proceso de estímulo-respuesta, es explicado cognitivamente en el desarrollo del agente para mejorar el análisis.

Cuando se habla del tiempo de respuesta, se hace referencia al coste de tiempo que implica desde que se percibe algún movimiento (definido) hasta que proporciona una

respuesta en consecuencia. Por lo tanto, es la capacidad de detectar, procesar y dar respuesta a un estímulo. Factores determinantes en el coste acción-respuesta:

- **Percepción.** Ver, oír o sentir el estímulo con seguridad es esencial para tener un buen tiempo de reacción. En una carrera de atletismo, cuando el juez da la señal de salida con una pistola, el sonido llega a los oídos de los atletas (perciben el estímulo) y saben que la carrera ha empezado.
- **Procesamiento.** Es necesario centrarse y entender bien la información para un adecuado tiempo de reacción. Siguiendo el ejemplo anterior, los corredores, al percibir la señal de salida, la distinguen del ruido ambiente y entienden que pueden empezar a correr (procesan el estímulo).
- **Respuesta.** La agilidad motora es necesaria para actuar ante el estímulo y tener un buen tiempo de respuesta. Cuando los atletas han percibido y procesado correctamente la señal, empiezan a mover las piernas (dan respuesta al estímulo).

4.2.1.3 Niveles de simulación, interacción e inmersión

Tal como se vio en el *Capítulo III*, en el apartado de entornos virtuales 3D, existen componentes importantes dentro de la arquitectura de un sistema de realidad virtual, en específico se consideran tres componentes elementales en este tipo de sistemas, los cuales son:

- Simulación interactiva
- Interacción implícita
- Inmersión sensorial

El nivel de simulación siempre estará directamente relacionado con la variedad de leyes físicas que se simulan en un entorno virtual. Normalmente, estos sistemas usan únicamente modelos de iluminación empíricos para lograr simular las leyes básicas percibidas visualmente, de esta manera proporcionar una imagen realista aceptable de la escena virtual, no obstante, existen modelos y técnicas para potencializar estas simulaciones, por ejemplo:

- **Cinética y dinámica.** Algunos sistemas (la gran mayoría) posibilitan la simulación de fuerzas de gravedad y leyes básicas de movimiento para los objetos, como la inercia, aceleración, etc. Un ejemplo sería, al tomar un objeto virtual y aventarlo contra la pared, el objeto tendría que rebotar, explotar, romperse (dependiendo sus propiedades) por efecto de la fuerza con la que se arrojó o permanecer íntegro, esto se determina por el grado de simulación implementado en el sistema.
- **Deformaciones.** Generalmente, los sistemas de realidad virtual consideran a los objetos con propiedades elementales inalterables, es decir, rígidos, sólidos, sin movimiento. Pero no en todos es así, algunos sistemas trabajan en los detalles,

calculando la deformación de los objetos en base a las fuerzas que actúan sobre este.

- **Detección de colisiones.** Siendo un sistema que permite el desplazamiento por un modelo tridimensional, resulta imprescindible detectar colisiones con los objetos interceptados y evitar de esta manera que dos objetos ocupen la misma posición en el espacio, también para poder interactuar con los objetos existentes en la escena
- **Simulación del comportamiento de un sistema físico.** Algunos sistemas de alta complejidad, simulan el comportamiento de un sistema para estudiar fenómenos físicos, por ejemplo movimientos de partículas o el funcionamiento de alguna red neuronal. Este tipo de simulaciones son llevadas a cabo mediante preprocesos que se encargan de desarrollar tareas específicas, dado las limitaciones que el proceso de cálculos en tiempo real. Por lo cual, el sistema de realidad virtual solo arroja la visualización de los resultados.

Una de las características del desarrollo de un sistema de realidad virtual, es la gran variedad de simulaciones que se pueden implementar, independientemente de que tipo de sistema se trate, muchas actividades pueden simularse en diferentes niveles de complejidad. La simulación de cualquier ley física, movimiento o comportamiento, tienen un efecto directo en la percepción, por lo cual el nivel de inmersión influye de gran manera.

El factor interactividad, siempre será muy importante en cualquier sistema de esta índole, sin embargo no es necesario. Particularmente, en el presente trabajo, se le ha dado especial importancia, pues de ello depende la variedad de acciones disponibles para el usuario que, evidentemente, tendrán una respuesta por parte del entorno virtual. Se toma como generalidad, según Jeff Murray (2014), que el nivel de interacción básico de un sistema de realidad virtual permite al usuario cierto grado de libertad para poder moverse por la escena y observar los objetos desde una perspectiva en primera persona y según la dirección que elija. No obstante, existen programas que ofrecen un nivel más avanzado de interactividad, por ejemplo, permiten al usuario interactuar directamente con los objetos que le rodean, desde abrir puertas, ventanas o muebles, tomar objetos, prender electrodomésticos, e incluso, manejar aviones o carros.

Elementos embebidos de la interacción implícita

La interacción implícita hace referencia en la recopilación de información acerca de las intenciones del usuario, la cual es inherente al movimiento “natural” de este. Tal como se mencionó anteriormente, una manera de efectuar esta interacción es, al utilizar dispositivos de realidad virtual específicos se puede capturar el movimiento del usuario que derivaría en una respuesta en el entorno virtual donde se encuentra inmerso.

Por lo cual, todo sistema de realidad virtual tiene que contemplar las posibilidades de acción que no tienen relación con un movimiento natural en la escena. Es importante mencionar las posibilidades de desarrollo que brindan estos dispositivos especiales para

realidad virtual en lo que respecta a la interactividad, un ejemplo claro es cuando un sistema utiliza reconocimiento de voz como medio para enviar información a la computadora de alguna acción que quiera emprender, este mecanismo sustituiría el teclado o el ratón (dispositivos más comunes para la comunicación con una computadora).

Independientemente del dispositivo que sea implementado para la interacción con la computadora en un sistema de realidad virtual, es importante mencionar que aunque exista una importante limitación para el usuario al dejar las posibilidades de interacción ser definidas por la trayectoria que decida elegir, esta acción es determinante para la interacción de muchas de las tecnologías señaladas (reconocimiento de voz, cascos de movimiento, guantes, etc). Generalmente, en las aplicaciones CAD¹³ (Computer Aided Design), gran parte del tiempo que invierte el usuario, está dirigida a la navegación permitida dentro de la escena. Este tipo de aplicaciones manejan un término denominado “tiempo de aprendizaje cero”, el cual sirve para indicar que la mayor parte de la interacción en el sistema se basa en movimientos que no necesitan un proceso instruccional que lo anteceda.

Elementos embebidos de la inmersión sensorial

La nivel de inmersión sensorial se encuentra relacionado, primeramente, de los sentidos del usuario al que va dirigido, los estímulos que proporciona el sistema, posteriormente, la calidad, alcance, velocidad, coherencia, y frecuencia, que estos son capaces de brindar.

Generalmente, estos sistemas, tienen contemplado los cinco sentidos humanos (tacto, vista, olfato, oído, gusto), sin embargo también se pueden desarrollar estímulos para otras percepciones del ser humano, por ejemplo, el equilibrio, cinestesia, etc. De los cuales, los más asistidos para la realidad virtual son la vista y el oído, tal como sugiere el presente trabajo. Por otro lado, en las aplicaciones de alto nivel de simulación e inmersión, el tacto, el equilibrio y la cinestesia son muy recurridos, por ejemplo en una simulación de manejo de un avión, para percibir las fuerzas físicas, como aceleración, altura, etc.

Tal como se puede deducir, por lo anterior expuesto, el nivel de inmersión es proporcional al número de sentidos que son estimulados por el sistema. Este proceso se agudiza, en medida que el sistema es capaz de aislar al usuario completamente del mundo real, y sumergirlo dentro del mundo virtual. El presente trabajo, presenta un sistema **semi-inmersivo**, pues aunque la representación geométrica 3D ofrece la posibilidad de ser visualizado en una alta calidad mediante lentes 3D, está basado primordialmente en la proyección en una pantalla, donde se puede percibir el entorno real que existe entre el ojo expectante y la pantalla.

¹³ El CAD se refiere al uso de programas informáticos para generar representaciones gráficas de objetos físicos, en 2da o 3ra dimensión. El cual es ampliamente utilizado para la animación computacional, efectos especiales, publicidad, y diversas aplicaciones industriales. (PLM-Software, 2013)

En los procesos de inmersión, las características de los estímulos, como la calidad en imágenes, sonido, formas, etc., la velocidad de actualización y tiempo de latencia (el cual se refiere al tiempo de espera entre acción y respuesta) y coherencia (entre la acción que queremos implementar y lo que se ve reflejado) son determinantes también, en el grado de inmersión sensorial. Entre los sistemas de inmersión de alto nivel, se han sustituido algunos problemas, por ejemplo, de desorientación por lo cual esta característica no siempre se pretende desarrollar a niveles altos por el desarrollador. Por otro lado, el desarrollo de algunos sistemas buscan simplemente aumentar la información sensorial que recibe el usuario, por lo cual, es necesario mantener una percepción real del entorno.

4.2.2 ARQUITECTURA ENTORNO VIRTUAL INTELIGENTE CON AGENTES

Dentro de la literatura de sistemas desarrollados en realidad virtual con influencia de agentes inteligentes, no se encuentran muchas arquitecturas formalizadas, sin embargo, se implementará una arquitectura genérica basada en modelos/módulos, prevista para entornos virtuales inteligentes, adaptándola a la metodología del sistema.

4.2.2.1 Introducción

A lo largo del presente trabajo, se han implementado metodologías para el desarrollo del sistema poniendo especial atención a los detalles. A continuación se muestran los procesos principales que rigen el funcionamiento del sistema en general, siguiendo la directrices marcadas por la arquitectura vista anteriormente. La cual permite desarrollar Entornos Virtuales Inteligentes con diferentes enfoques.

Específicamente, se ha realizado la implementación de esta arquitectura, no solo para probar su eficacia, sino también para poder facilitar la adopción de esta, por diferentes investigadores interesados en el tema. Así mismo, se proporciona datos relevantes que permitan el desarrollo del sistema a niveles de más complejidad, en todas sus características (interacción, simulación, preproceso, etc), de esta manera aumentar la utilidad en los diferentes ámbitos del desarrollo de sistemas inteligentes.

Se han desarrollado aquellas entidades más representativas para el funcionamiento del sistema, los cuales resultan con más relevancia por el enfoque del programa y sus probables futuras líneas de investigación. Tal como se menciona en la metodología, se organizó las diferentes funciones por medio de módulos de desarrollo los cuales se muestran en la Fig. 4:8. Gracias a la metodología orientada a prototipos, se ha podido desarrollar de manera detallada cada una de las etapas, desde el modelo geométrico hasta las técnicas implementadas para su funcionamiento, de esta manera se puede decir que ha simplificado los tiempos de desarrollo de este tipo de sistemas.

Se debe recordar, que aunque el presente proyecto no tenga un enfoque pedagógico, el usuario estará adquiriendo nuevo conocimiento acerca de temas en particular, información general acerca del inmueble del cual se generó un modelo 3D, objeto de estudio.

Resumiendo, a continuación se implementa una arquitectura para el desarrollo de Entorno Virtuales Inteligentes con Agentes, en este caso se profundizara en el desarrollo de un agente reactivo basado en modelos, pues es quien se encargara del comportamiento del usuario inmerso en el entorno, dandole la posibilidad de recorrer el modelo 3D en el orden que este disponga, sin perder de vista el objetivo, que es recorrer por completo el entorno virtual e interactuar con los objetos predefinidos incluidos en la escena. Cabe mencionar, que el presente trabajo se desarrolla dentro del marco de generación de sistema inteligentes con una propuesta diferente de desarrollo.

4.2.2.2 Requerimientos funcionales y no funcionales

Los requerimientos funcionales del sistema, son declaraciones de los servicios que brindara el sistema y las especificaciones para realizarlos.

Servicios:

- Interfaz Interactiva
- Funcionamiento intuitivo
- Sensación de inmersión en primera persona
- Representación visual de alta calidad (3D, 2D)

Especificaciones:

- Computadora arquitectura x64
- Procesador mayor a Core i5
- Sistema Operativo Windows 10 (vista, 7, 8)
- Memoria Ram 8GB
- 160 MB como mínimo en espacio de Disco Duro
- Tarjeta gráfica Intel HD Graphics 4400, memoria compartida 2048 MB
- Conexión a internet no es necesaria

Los requerimiento no funcionales del sistema, representan las herramientas utilizadas en el proceso de desarrollo y las características de implementación del software

Calidad de Software:

- Posibilita la utilización de lentes 3D para vizualizar mejor el entorno
- Robustez minima
- Escaso tiempo de respuesta por operación

Proceso de Desarrollo:

- Plataforma Windows 10
- Software de Modelado 3D Blender
- Software Unity
- Software Photoshop
- Lenguaje de programación C# para el desarrollo de técnica IA

4.2.2.3 Estructura General del Sistema

Tal como se menciona anteriormente, el objetivo principal de esta arquitectura es dividir el funcionamiento del sistema en entidades y módulos que puedan ser desarrolladas a largo plazo con nuevas intervenciones y se ajusten nuevos procesos.

La funcionalidad del Entorno Virtual Inteligente basado en un agente reactivo (en lo subsecuente “Sistema CNA RV”), se desarrolla principalmente por los siguientes módulos:

- Módulo de Comunicación
- Módulo Gráfico
- Módulo de Movimiento
- Módulo de Comportamiento

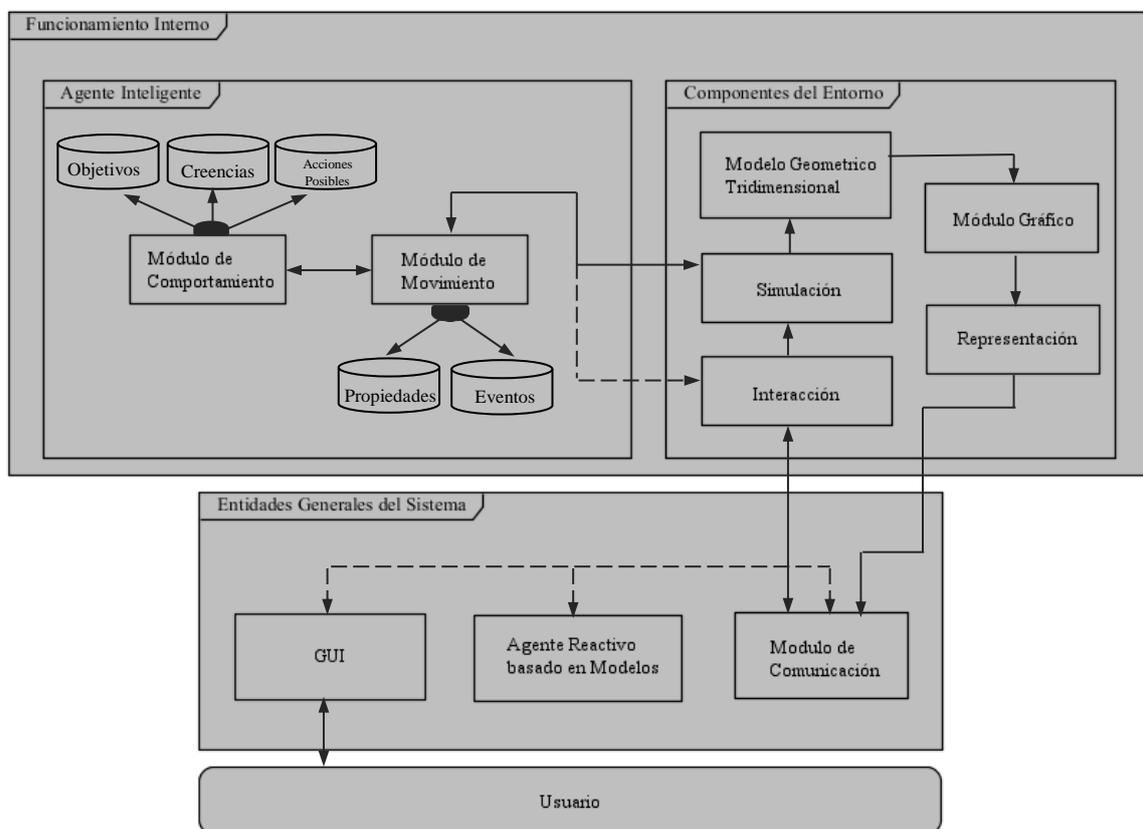


Fig. 4.8 Estructura General del Sistema CNA RV. (Elaboración Propia).

En la Fig. 4.8 se pueden distinguir a primera vista, tres bloques que representan la nivelación de la estructura, en primero lugar se encuentra el bloque del usuario, posteriormente las entidades generales del sistema, las cuales intervienen directamente con la interfaz, y finalmente, el bloque de funcionamiento interno. Este último hace referencia, a su vez, a dos “subprocesos”, uno encargado de la gestión del entorno virtual y otro encargado de la gestión del agente “user”, que determinarán la funcionalidad del sistema. Cada uno tiene relación implícita con otros componentes que asisten a estos procesos para llegar a su objetivo

4.2.2.4 Descripción del Sistema

A continuación se desarrollan los módulos principales que interfieren en la comunicación del sistema en general, como se comunican los procesos internos con la interfaz de usuario y su desarrollo. Así mismo, las relaciones existentes no representadas en la arquitectura general *Fig. 4.8*.

4.2.2.4.1 Módulo de Comunicación

Este módulo funciona para implementar una interfaz entre los subsistemas (modelo geométrico y agente reactivo) y el resto de la aplicación. Esta comunicación se genera gracias a una visión lógica del mundo, es decir, una representación del estado (agente reactivo) del entorno virtual, lo mismo sucede cuando se define el comportamiento de los integrantes del entorno.

En la comunicación a este nivel, no contemplan detalles como animaciones y colisiones de los objetos del entorno, sino tiene un enfoque a las acciones relevantes para el módulo que respecta al comportamiento del agente.

La comunicación que hay entre el agente user y los objetos de su entorno, no requiere mucho esfuerzo pues toma los datos directamente de las características predefinidas de los objetos con los que va a interactuar. Por lo cual, el módulo de comunicación, a través de las estancias de simulación e interacción, sencillamente invoca las propiedades de las primitivas involucradas en la visión lógica de la escena. Por otro lado, cuando el agente user tiene que interactuar con el usuario, la entidad correspondiente invoca las primitivas del simulador que a su vez, se retroalimenta del estado interno del agente.

4.2.2.4.1.1 Sensores y Actuadores

La implementación de la comunicación entre el entorno virtual y la visión lógica están representadas por entidades/módulos que interactúan entre sí. En este sentido, necesitan una conexión física que permita interpretar las acciones que desea ejecutar el usuario, por lo cual los sensores y actuadores, recabadores de información, son de vital importancia.

Los componentes encargados de recibir mensajes internos que, al ser procesados, derivan en acciones en el entorno, son posibles, en gran medida, a la percepción. Pues las percepciones y acciones que la visión lógica representara, cambian, y el módulo responsable del procesamiento de estas (junto con sus instancias) reemplazara estas modificaciones en el estado del mundo, actualizándolo constantemente.

En la *Tabla 4.3* se muestran algunas acciones básicas para la mecánica con la que se rigen los procesos del entorno virtual, la gran mayoría derivan en la acción de una entidad específica a otra o un módulo de acción completo.

Sensores	
usuario_Press	Evento que indica cuando el usuario ha seleccionado un objeto específico de su entorno o interfaz.

estancia_Visitada	Evento que indica cuando el usuario ya ha visitado determinado espacio dentro del entorno virtual.
usuario_Vista	Evento que indica el ángulo de visibilidad del entorno, por tratarse de una perspectiva en primera persona.
direc_Desplaza	Indica en que dirección se desplazara el usuario dentro del modelo geométrico tridimensional.
Actuadores	
direc_Vision	Manda una orden de hacia donde se tiene que posicionar la representación virtual del usuario (agente).
direc_Des	Orden de desplazamiento hacia determinado objetivo, por ejemplo hacia una puerta, ventana o algún documento predefinido para interactuar con la entidad inteligente.
modo_Des	Determina la manera en la que el usuario se desplazara por el entorno virtual, ya sea caminando o corriendo.
act_Objec	Envía un estímulo para activar una acción/evento de un objeto (predefinido) con el que se desea interactuar.
comunicaUsuario	Cuando el usuario pretende implementar alguna actividad que el agente inteligente procesa como “especial” para el cumplimiento del objetivo, este le manda mensajes de “advertencia”.

Tabla 4.3 Acciones primitivas entre entorno virtual y la visión lógica del sistema. (Elaboración propia).

4.2.2.4.1.2 Interacción

El proceso de interacción se encuentra relacionado con el módulo de comunicación y aunque en el esquema general aparece como una instancia sin ramificaciones, realmente se compone por otros “subprocesos” susceptibles al resultado de los módulos subsecuentes al que pertenece este. La forma en que se presentan los datos y procesos de la simulación, la interacción que existe con una simulación, son de suma importancia pues estos determinan la posibilidad de procesar cantidades de datos significativas, de forma resumida. Sin embargo, el presente sistema enfoca la mayor parte del procesamiento de la información obtenida, por medio de los sensores con los que se comunica el usuario, al agente reactivo “user”.

Todas las acciones posibles a implementar están basadas en la interacción típica en un entorno virtual (independientemente de su aplicación), donde el usuario, representado por un avatar o no, puede interactuar con el entorno, objetos y otras entidades (según sea el caso). Sin embargo, el nivel de interactividad determinará cuán involucrado está el usuario con el entorno y en base a ello poder modificar o no los objetos que lo rodean (*Capítulo III*), particularmente en esta entrega del proyecto se tiene contemplado un nivel de *interacción explotativa*. La gran mayoría de los casos, esta interacción es posible gracias a operaciones genéricas, que a su vez, se encuentran dependientes de la semántica que se implemente en cada entidad/objeto. Algunas de estas son:

- **Accionar.** El usuario podrá accionar algún objeto que se encuentre en la escena, esto derivara en algún evento programado por medio de sus propiedades semanticas o bien, algún algoritmo que especifique un evento en especial.
- **Recolectar.** El usuario por medio de su representante virtual, podrá recolectar objetos disponibles predefinidos, encontrados en su entorno (documentos, aparatos, etc)
- **Mirar.** Cuando el usuario se posicione en determinado lugar, podrá mirar demanera detallada el entorno o los objetos, siempre en primera persona.

4.2.2.4.1.3 Simulación

Este componente se encuentra relacionado con el módulo de comportamiento pues es quien determina las características físicas de los objetos presentados en el entorno. El programa responsable de la simulación se desarrolla de la siguiente manera. Tal como se muestra en la estructura general del esquema, para generar una simulación coherente con lo que sucede en el entorno se debe recurrir al módulo que almacena el estado del mundo, es decir, la tarea que desarrolla el agente inteligente.

De esta manera cuando se ejecutan ciertas acciones en el entorno virtual, acciones emprendidas por el usuario que interactua (abrir una puerta), se provocan cambios en la “vista lógica”, la cuál es actualizada gracias a la simulación. Así mismo, estos cambios son desarrolladas en el entorno virtual, para modificar su presentación acorde a las decisiones tomadas. Existen distintos tipos de simulación por computadora basadas en teorías estadísticas y matemáticas que proveen herramientas para el análisis objetivo para optimizar los cálculos de la computadora. Whiker & Sigelman (1991), mencionan los elementos fundamentales que componen una simulación por computadora. (Luna, 2012)

Asunciones	Parámetros	Variables de entrada (independientes)	Algoritmos	Variables de salida (dependientes)
El razonamiento sobre el cual se rige el fenómeno de estudio, esta basado en este conjunto de “hipótesis”, las cuales también constituyen el esqueleto del modelo.	Se refiere a las variables de control, su valor se mantiene fijo para poder identificar como las variables independientes afectan el comportamiento de la variables dependientes.	Son los valores ingresados a la simulación cuando esta se inicializa. Estos valores afectaran directamente a las variables dependientes.	Los algoritmos implementados son la parte mas importante en el sistema, pues estos convierten las entradas en salidas, siguiendo reglas de decisión que se han especificado. Contituye la operatividad del modelo y sus procesos.	Las salidas son calculadas mediante los algoritmos que procesan las variables de entrada, sin ser modificadas por ningun tipo de intervención externa.

Tabla 4.4 Elementos que constituyen una simulación por computadora. (Mejía Luna Jaqueline, 2012)

La simulación, también, se encuentra dirigido al módulo gráfico para llevar a cabo su función, ya que en esta instancia, el simulador conoce la respuesta que se debe

implementar en base a los estímulos obtenidos, así, mediante las primitivas del modelo geométrico del entorno, se encarga de la visualización que será enviada al usuario.

Según los datos proporcionados por los módulos y estancias adecuados, la representación virtual del usuario “agente user” puede decidir detener al usuario ante alguna actividad específica, para darle advertencias (p. ej. si ya visitó determinado espacio, no permitir que pierda tiempo visitándolo nuevamente), informar algún error, dar pistas o información acerca de su entorno, o simplemente permitir que siga la actividad elegida sin ningún evento en especial.

4.2.2.4.2 Modelo Geométrico Tridimensional

El modelo geométrico es una parte esencial en el presente trabajo, pues en este apartado será definido el aspecto gráfico mediante un software de modelado 3D, tal como se menciona en el *Capítulo III*. Un entorno virtual en dos dimensiones conlleva un proceso completamente diferente a uno en tres dimensiones, pues para el primero no hay más que utilizar herramientas de diseño básicas, ya que solo se implementan imágenes planas, sin embargo, para la creación de tres dimensiones, existe un grado mayor de complejidad pues es necesario implementar una herramienta capaz de exportar la geometría fidedigna en un formato simple y manipulable en cualquier plataforma. Para la creación del modelo, se ha seleccionado los elementos necesarios para generar el diseño final del entorno virtual, de acuerdo a las características que se describen en la metodología del sistema.

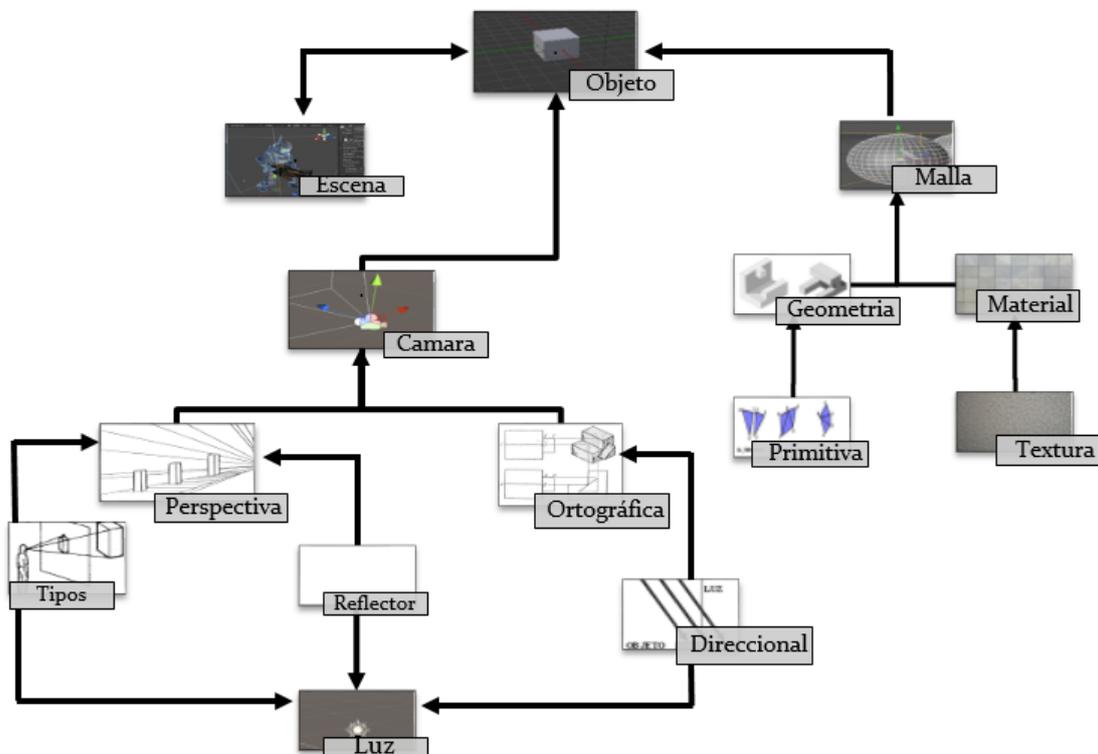


Fig. 4.9 Esquema General de un Objeto Tridimensional. (Elaboración Propia).

Dentro de un modelo 3D los componentes que se desarrollan siguen una jerarquía funcional, tal como se presenta en la figura Fig. 4.9. Detrás de la creación de cualquier entidad u objeto. La geometría consta de los modelos compuestos por polígonos, tal como

se menciono anteriormente, esta es la técnica implementada en el modelo del presente proyecto.

4.2.2.4.2.1 Definir objetos y personaje dentro del EV

Es importante tener en cuenta la cantidad de objetos que contendrá el entorno de esta manera se pueden asignar comportamientos complejos y básicos, según sea requerido. A continuación se definen y describen las áreas que formaran parte del paseo virtual interactivo, esto con la finalidad de seleccionar los espacios más relevantes para el diseño del modelo 3D, y así poder determinar la precisión y alcance del proyecto.

	Nombre	Características ¹⁴	Objetos
Laboratorios	Realidad Virtual y Videojuegos	Desarrollo e Investigación de los alcances y usos de nuevas tecnologías de simulación visual.	Sillas, Escritorios, Computadoras, Puertas, Ventanas, Muebles
	Interfaces electrónicas y robótica	Diseño y Desarrollo de circuitos electrónicos y estructuras mecánicas de bajo costo.	Sillas, Escritorios, Computadoras, Puertas, Ventanas, Muebles
	Imágenes en Movimiento	Experimentación, investigación y documentación de la imagen en movimiento.	Sillas, Escritorios, Computadoras, Puertas, Ventanas, Muebles
	Investigación en arte y tecnología	Investigación, reflexión, formación, y divulgación de enfoques acerca del arte/tecnología	Sillas, Escritorios, Computadoras, Puertas, Ventanas, Muebles
	Publicaciones digitales	Experimentación y desarrollo de sitio web, publicaciones digitales e impresas. Diseño de interfaces para distintos proyectos.	Sillas, Escritorios, Computadoras, Puertas, Ventanas, Muebles
	Gráfica Digital	Experimentación, creación e investigación de la imagen fija. Automatización de la tecnología informática y técnicas de estampado tradicionales.	Sillas, Escritorios, Computadoras, Puertas, Ventanas, Muebles
	Audio	Desarrollo e investigación de herramientas tecnológicas que pueden aplicarse con fines creativos en las artes sonoras e interdisciplinarias.	Sillas, Escritorios, Computadoras, Puertas, Ventanas, Muebles
Espacios Comunes	Galeria Manuel Felguerez	Espacio dirigido a la promoción y exhibición de manifestaciones de arte digital y electrónico por artistas de México y del extranjero.	Sillones
	Foro A/B	Foros de exposición, promoción y exhibición.	Exposiciones
	Sala de Juntas	Sala de juntas de Centro Multimedia.	Mesa de juntas, Sillones, Muebles
	Oficinas	Oficina de directivos.	Sillas, Escritorios, Computadoras
	Comedor	Espacio común para trabajos, exposiciones, talleres, etc.	Mesas, Bancas, Sillones

Tabla 4.5 Distribución del Modelo del Entorno Virtual. (Elaboración Propia).

¹⁴ La información presentada en la columna de características ha sido extraída de la página oficial del Centro Multimedia. (Multimedia, s.f.)

4.2.2.4.2.2 Modelado 3D

El modelado por polígonos es la expresión más básica de los modelos tridimensionales generados por computadora, los cuales están constituidos por vértices, aristas y caras. La cantidad de polígonos que contiene cada modelo es fundamental, ya que tiene relación directa con la calidad gráfica del modelo, así mismo con el comportamiento cinemático del entorno.

También se ha considerado el costo de memoria que la cantidad de polígonos implica, lo cual resulta beneficioso para la aplicación pues se trata de un modelo estructural, por lo cual las relaciones que podrían existir con otros modelos complejos en el entorno es casi nula. En la *Fig. 4.10* se puede observar el plano con medidas reales del inmueble CNA, del cual se realizó el modelo 3D, este también se puede observar en la parte derecha en una vista que permite visualizar la estructura mostrando solo sus bordes, de esta manera se puede apreciar con mayor exactitud de la unión de los vértices en la arquitectura a modo de generar menos polígonos en el modelo final.

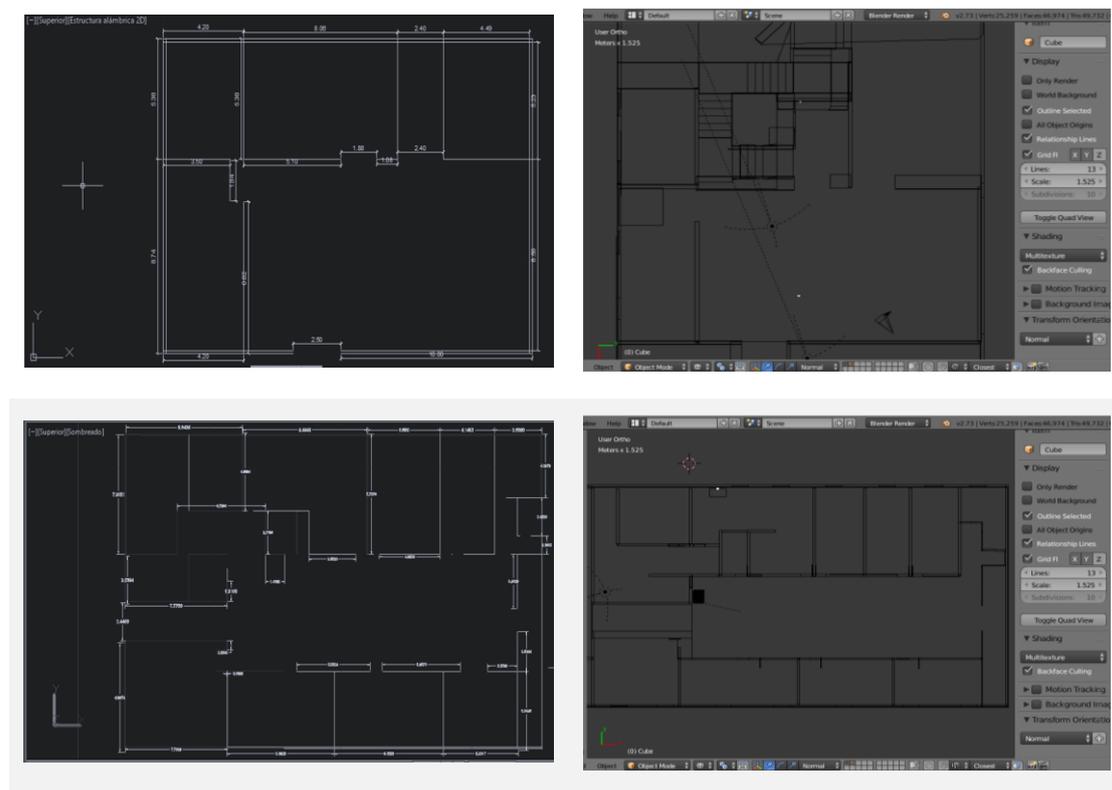


Fig. 4.10 Planos del inmueble y modelo 3D (vista wireframe). (Elaboración propia).

Cabe mencionar, por el contexto de desarrollo de la presente aplicación, que la manipulación y tratamiento de los modelos realizados en Blender, fue posible gracias a tres operaciones básicas de cálculo vectorial. Pues la técnica de geometría poligonal implementada, en el entorno virtual, ayuda a que los modelos 3D consuman lo mínimo en costos computacionales, así el procesador puede calcular de forma rápida las ecuaciones que forman parte de los objetos del entorno.

Las transformaciones realizadas mediante las operaciones vectoriales, afectan a cada polígono editable y cada uno de sus subconjuntos, tal como se explicará a continuación.

- **Traslación.** Se aplica cuando se necesita el movimiento de un objeto de un punto desde la posición $P_1 = (x_1, y_1, z_1)$ a la posición $P_2 = (x_2, y_2, z_2)$. Cuya operación es representada por la matriz siguiente: (Juan Manuel Jiménez Gutiérrez, 2014)

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix}$$

Donde t_x, t_y, t_z son correspondientes a las distancia traslacional respecto a los puntos de los ejes x, y, z . La traslación abarca todos los puntos del objeto seleccionado, lo cual quiere decir, que dentro de un entorno tridimensional el objeto adquiere una nueva posición cuando cada uno de sus puntos son trasladados. Pues al estar creados los objetos por un conjunto de superficies poligonales, se debe aplicar la traslación completa de cada uno de sus vértices de sus superficies, tal tarea se simplifica con algunas herramientas de software implementado para modelar.

- **Rotación.** Para aplicar el movimiento de rotación a un objeto, se le debe asignar un eje de rotación, generalmente los objetos se definen respecto al origen del sistema de coordenadas (0,0,0), lo cual permite mantenerse en armonía con los demás objetos de su entorno, sin embargo, el movimiento de rotación lo hará sobre su propio eje. (Juan Manuel Jiménez Gutiérrez, 2014)

Matricialmente, los movimiento de rotación en un ambiente 3D, se expresan de la siguiente manera:

Respecto al eje "X"

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\text{sen } \theta & 0 \\ 0 & \text{sen } \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix}$$

Respecto al eje "Y"

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & \text{sen } \theta & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen } \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix}$$

Respecto al eje "Z"

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\text{sen } \theta & 0 & 0 \\ \text{sen } \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix}$$

Lo cual nos dice que dependiendo del eje en el cual se aplique la rotación se definirá el plano de movimiento. Por ejemplo, cuando se gira sobre el eje x (el cual es un punto en el origen del sistema de coordenadas y_1, z_1 ; y_2, z_2) el plano de giro será sobre YZ.

- **Escalación.** La escalación es el cambio de tamaño a un objeto, el cual se efectúa sobre sus tres ejes, sin perder su posición original. Los parámetros de escalado son: $x_2 = x_1 * s_x$, $y_2 = y_1 * s_y$, $z_2 = z_1 * s_z$

La matriz de escalado en una posición $P(x_1, y_1, z_1)$, respecto a un eje coordenado, está representado por: (Juan Manuel Jiménez Gutiérrez, 2014)

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix}$$

La organización espacial determinada por el grafo de escena, hace la organización de los datos más sencilla, tanto para acceder a estos como para invocarlos para el proceso del recorrido virtual y la interacción del usuario con este.

El esquema derivado de la generación de escenas de la aplicación se presenta en la Fig. 4.11, se ha diseñado de manera que el usuario perciba una navegación intuitiva dentro de esta aplicación. Tal como su nombre sugiere, muestra las diferentes opciones disponibles y en el flujo de pantallas. Estas opciones se encuentran disponibles para consultar información acerca del inmueble, y los espacios tridimensionales que conforman el recorrido virtual; también existe un apartado, en el menú principal, que muestra las instrucciones de uso de la aplicación.

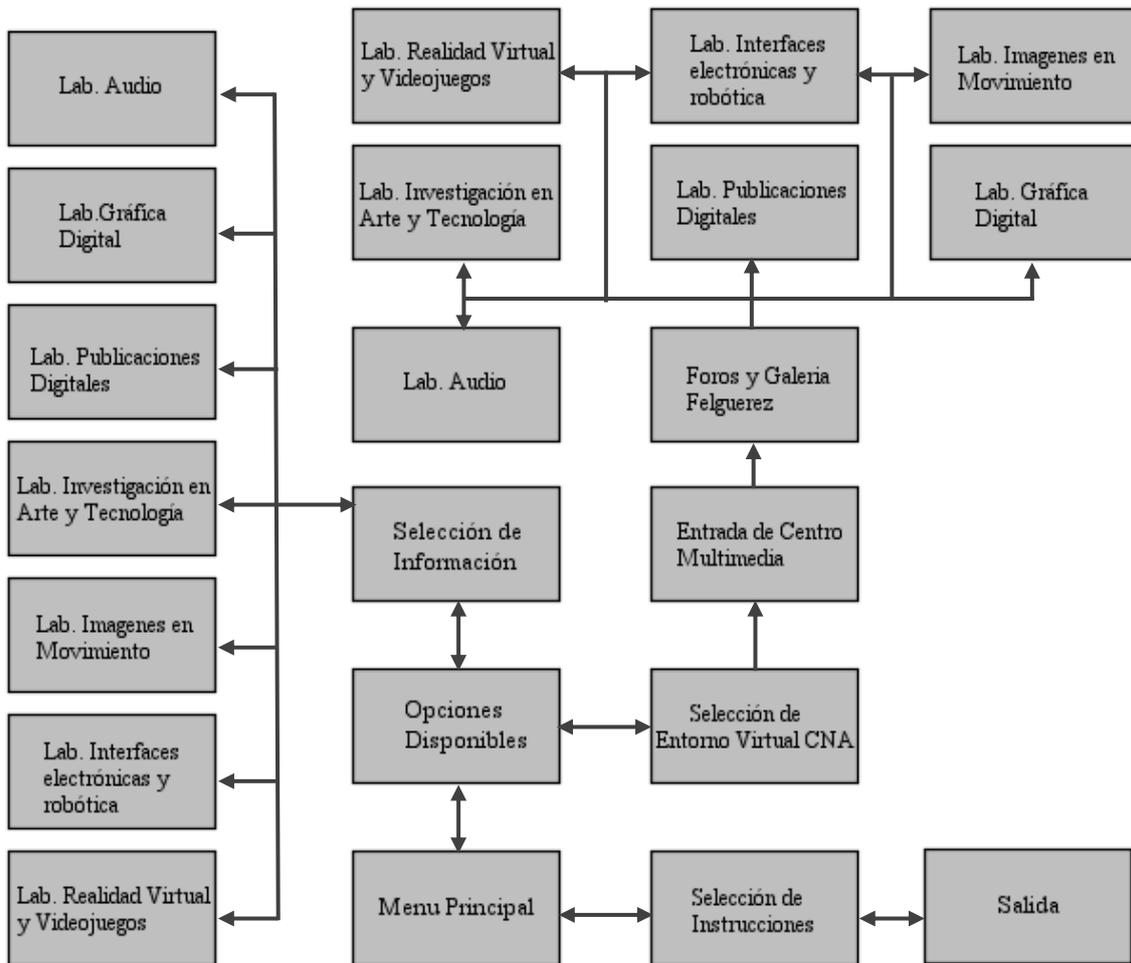


Fig. 4.11 Diagrama de transición de escenas de la aplicación. (Elaboración propia).

El entorno virtual desarrollado también puede fungir como una guía exacta para los conferencistas y expositores que deseen presentar su obra en alguno(s) de los foros situados en las instalaciones del *Centro Multimedia*, ya que la estructura modelada cuenta con medidas exactas del inmueble. Es por esta razón es que en el desarrollo del modelo 3D se hace especial reconocimiento a los detalles de la estructura más que de los detalles de los objetos encontrados dentro de sus laboratorios de trabajo. De igual manera resulta importante mencionar que por cuestiones de privacidad y seguridad, los objetos encontrados dentro del entorno son simples referencias hacia los principales atractivos de este centro de investigación.

4.2.2.4.2.3 Asignación de texturas y materiales

Tal como se menciona en el *Capítulo III*, los materiales y texturas aplicados en los objetos son muy importantes, sin embargo, estos a su vez deben gran parte de su apariencia a las luces utilizadas en el ambiente, ya que estas definirán cuán brillante, opaco y/o nítido es la apariencia de un objeto y eso debe tener concordancia con su entorno (día, noche, lluvia, etc).

La asignación de texturas y materiales para el modelo 3D, fue basado en imágenes capturadas de las instalaciones con retoque digital para crear concordancia visual entre objetos. Los materiales utilizados fueron creados a través de herramientas que ofrece el mismo programa Blender (software de desarrollo del modelo 3D), los cuales son materiales básicos, tales como vidrio, aluminio y material sólido de diversos colores. (Fig. 4.12)



Fig. 4.12 Texturas predominantes utilizadas en el modelo 3D. (Elaboración propia).

Los materiales utilizados en el modelado 3D, son creados a través de las herramientas que proporciona Blender, tales como la elección del color utilizado por texturas especiales para mejorar su aspecto, reflejos para simular materiales brillantes, difusión para atenuar la solidificación de un material, especular para crear brillantes, transparencia para crear restarle dureza al material, entre otros.

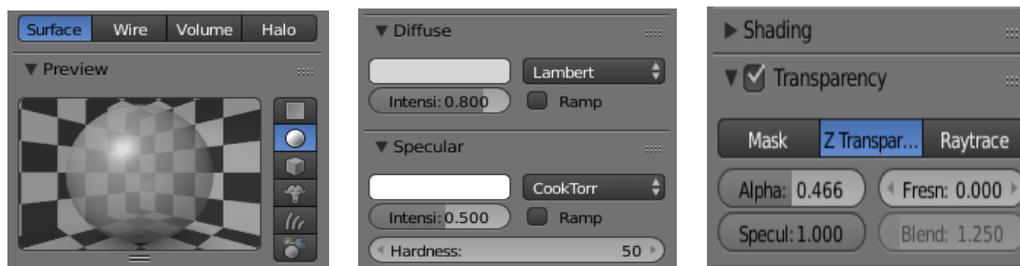


Fig. 4.13 Ejemplo de creación de un material. (Elaboración propia).

Así mismo, para el proceso para aplicar la textura o material, implica otras técnicas entre las cuales se escogió la técnica de *Mapeado UV*. La texturización viene a cubrir la falta de uniformidad que podría quedar utilizando solo materiales, por ejemplo, que no cubra por completo el color que se desea, la forma, o algún otro detalle de su aspecto final.

El *Mapeado UV*, es una técnica que se utiliza para mapear texturas “imagen” sobre modelos tridimensionales. Las texturas utilizadas fueron arregladas mediante Photoshop para hacer crear patrones concordantes en el mapeado. Esta técnica constituye una buena manera de texturizar un modelo pues se ajusta perfectamente a la imagen. Esto se logra gracias a que el mapeado logra asignar una imagen a cada polígono que conforma el modelo, pues cada vértice del polígono representa un par de coordenadas 2D que definen que parte de la imagen es mapeada. Precisamente, estas coordenadas son conocidas como UV's, las cuales son equiparables a las coordenadas XYZ en el plano 3D.

También, resulta sobresaliente mencionar que esta técnica es el estándar utilizado en las aplicaciones tridimensionales pues la operación para crear estos mapas UV es conocida también como “despliegue” (unwrap), la cual funciona como si fuera una malla desplegada sobre un plano 2D.

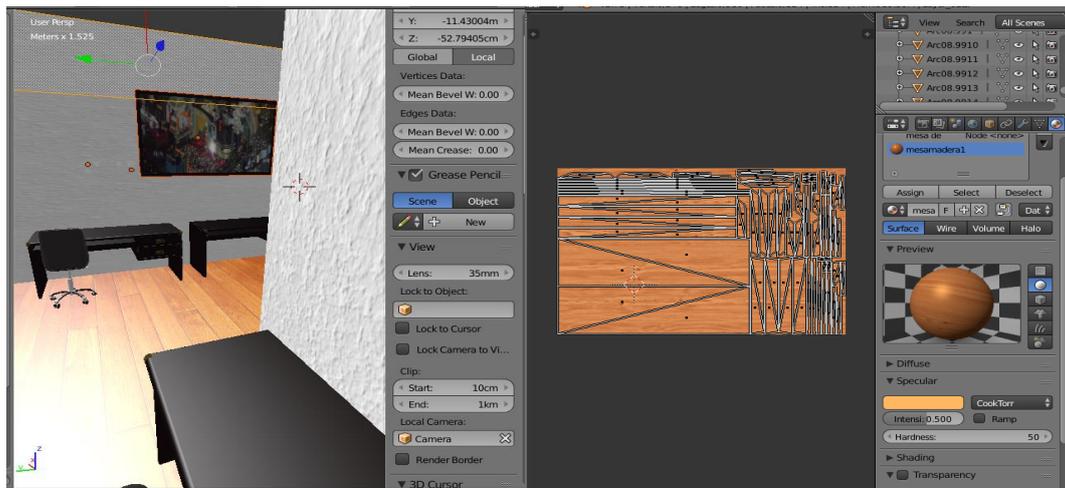


Fig. 4.14 Proceso de Texturización con UV Mapping en Blender. (Elaboración propia).

Cuando se aplica un *UV Mapping*, las coordenadas son medibles en una escala con rango de 0.0 – 0.1 (posiciones que van en los lados opuestos de la textura, respectivamente). Es decir, cuando la escala se indique en números más grandes que 1.0 la textura se repetirá para cubrir la superficie, por el contrario, si el número es menor la textura realizará otra operación para crear un efecto “espejo” con la textura aplicada. La mayor parte de las texturas utilizadas son aplicadas en la estructura del inmueble, esto para minorizar costos computacionales y tiempo de entrega, sin embargo, el resultado final tiene un nivel de realismo bastante aceptable en comparación con el ambiente real. (Juan Manuel Jiménez Gutiérrez, 2014)

4.2.2.4.2.4 Definición del Nivel de Física

Definir el nivel de física significa considerar las características físicas de los elementos que conforman el entorno virtual 3D del sistema, empezando por la cinemática¹⁵ de los objetos, hasta los fenómenos físicos implícitos, tales como gravedad, reacciones, colisiones, etc.

Para poder lograr un comportamiento físico convincente, cada objeto dentro del entorno debe acelerar y ser afectado por colisiones, gravedad, etc. En este caso en particular, estos aspectos fueron definidos en el software donde se generó el modelo original. Sin embargo, al ser exportado a Unity, se tuvieron que modificar ciertos parámetros para que se respetara la física del modelo en general. (Technologies, 2017)

Al controlar la física de los objetos desde la creación de scripts, se pueden programar dinámicas propias de una computadora, un vehículo, incluso una papel en movimiento. Algunas de las funciones fundamentales que fueron utilizadas en el entorno virtual fueron las siguientes:

Propiedad	Función / Box_Collider
is_trigger	Esta propiedad, es utilizado para eventos de triggering/desencadenante. El cuál será ignorado por el motor de física.
material	Se trata de “física del material” determina cuál es la forma en que este tipo de collider interactúa con otros objetos del ambiente.
center	La posición del collider en el espacio local del objeto.
size	El tamaño que abarca al objeto en las direcciones x,y y z.
Propiedad	Función / Capsule_Collider
is_trigger	Esta propiedad mantiene sus características de la función anterior, es utilizado para eventos desencadenantes y no interviene el motor de física.
material	Se trata de “física del material” determina cuál es la forma en que este tipo de collider interactúa con otros objetos del ambiente.
center	La posición del collider en el espacio local del objeto.
radius	Establece el radio del ancho local del collider en el objeto.
height	Altura del collider
direction	Se trata del eje de orientación longitudinal de la capsula en el espacio local del objeto.
Propiedad	Función / Character_Controller
slope_limit	Esta propiedad limita al collider para subir alguna pendiente que no represente una inclinación determinada (grados).
step_offset	El personaje va a identificar una escalera siempre y cuando se encuentre a una distancia oportuna del piso. Esta altura no debe ser mayor que la altura del character controller, de lo contrario se generaría un error.
skin_width	Cuando se encuentra con otro collider, se juntan tan profundamente como el ancho de su capa “piel”. Entre mayor sea el skin width se reduce la fase de fluctuación, si es mejor el personaje puede quedar atrapado en el collider de otro objeto.
min_move distance	Cuando el usuario trata de moverse por debajo del valor indicado, el movimiento quedara anulado. Lo cual, reduce la fluctuación de fase. Generalmente este valor debe ser varias de 0.
center	Esto va a desplazar el Capsule Collider en el espacio del mundo, y no va a afectar qué tanto el Personaje hace pivotes.
radius	Longitud del radio del Capsule Collider. Este es esencialmente el ancho del collider.
height	La altura de la Capsule Collider del personaje. Cambiar esto va a escalar el collider a lo largo del eje Y en ambas direcciones positivas y negativas.
Propiedad	Función / Mesh_Collider

¹⁵ Comprende una rama de la física que estudia el movimiento de los cuerpos en el espacio, independientemente de las causas que lo producen. (General, 2015)

is_trigger material	Collider utilizado para eventos de triggering, y será ignorado por el motor de física. Esta ligado directamente a la física del material, que determina en qué forma este Collider interactúa con otros.
mesh convex	Referencia al Mesh en ser utilizado para colisiones. Si está activado, este Mesh Collider va a colisionar con otros Mesh Colliders. Los Convex Mesh Colliders están limitados a 255 triángulos.

Tabla 4.6 Funciones de propiedades físicas implementadas. (Adaptación de Unity Technologies, 2017).

Estas funciones que determinan las propiedades físicas básicas de los objetos fueron aplicada debidamente en base al tipo de estructura que conforma el modelo al que se aplica, ya que pueden crear errores en la interacción con el usuario virtual.

4.2.2.4.3 Módulo Gráfico

El motor gráfico es el responsable del “dibujado tridimensional”. También, es el encargado de la creación de la ventana, utilizando las llamadas al sistema operativo subyacente.

Toda la información generada en el modelo 3D, se exporto al motor gráfico utilizado en el presente trabajo, el cual es Unity, mediante un archivo .fbx, que es el que tiene mayor compatibilidad con el software y el cual se ha comprobado que guarda de manera más completa por su jerarquización, la información creada del modelo (Capítulo III). El proceso de jerarquización, funciona mediante diferentes niveles que contienen objetos con sus propios parámetros, esta división de niveles se efectua de modo que solo haya pocos objetos por cada nivel para mayor control. La siguiente figura ilustra de manera general los niveles de jerarquización en una aplicación computacional con modelado 3D a nivel gráfico.

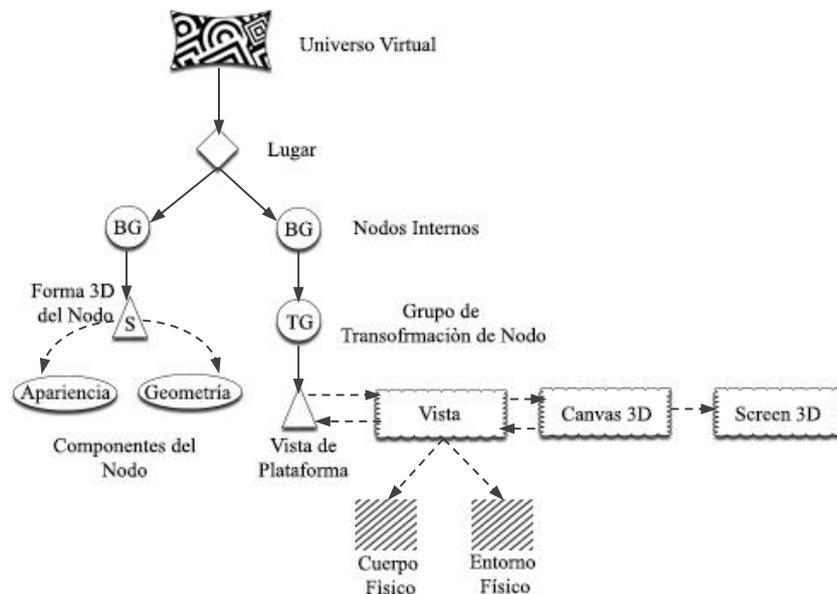


Fig. 4.15 Esquema de jerarquización computacional en una aplicación 3D. (Adaptación de Sivunen & Nordback, 2014).

El motor de juego de Unity, integra todas las instrucciones necesarias para el desarrollo de proyectos avanzados en el ambito tecnológico, pues incluye bastas

herramientas aplicables en imagen, interacción, sonidos, etc. Sin embargo, también permite adición de código propio, lo cual brinda muchas más posibilidades de desarrollo.

Una de las principales ventajas de Unity es la versatilidad para generar cualquier tipo de aplicación, ya que permite trabajar en 3D y 2D. Así mismo, ofrece librerías de programación como *Direct3D* y *OpenGL*. De esta manera, logra una fidelidad visual de alta gama, pues el poder de renderizado que ofrece a través de la programación de atributos propios del ambiente como: luces, cámaras, materiales, sombras, texturas, partículas y/o efectos especiales, logra crear ambientes hiperealistas apoyados, claro esta, de las ilustraciones, texturas, materiales que el autor posea para implementar.

Este motor gráfico posee clases principales que ayudan al funcionamiento básico de la aplicación, generalmente si se utilizada

- **Inicialización/termino.** El motor de unity da gran apertura a diseñar el inicio y el termino de la aplicación utilizando herramientas predefinidas, animaciones en distintos formatos, o comportamiento personalizado a través de código.
- **Bucle principal.** EL control principal de la aplicación, en gran medida, se lleva a cabo mediante el motor de Unity, específicamente la ejecución o “bucle principal”. En cada ejecución debe intervenir el motor gráfico, pues este dibuja los fotograma correspondientes, contemplando todas las entidades resgistradas.
- **Código específico.** Los objetos del entorno que cuentan con una representación gráfica, deberán invocar al mator gráfico para cargar los modelos, y para ello varia la forma de invocarlo pues dependiendo del material y texturas, sera el proceso más o menos engorroso.

El motor gráfico, también gran relevancia en toda la ejecución en bajo nivel, pues liga los procesos internos a los externos. Para ejemplificar esto, se hace referencia a la comunicación entre la interfaz física y la interfaz lógica, la cual puede estar representada por la cobinación mouse-cursor en el entorno virtual, la interfaz física es lo que nos permite recibir un estímulo, en este caso el mouse; por otro lado, la interfaz lógica corresponde al cursos que nos guia por el mapeado del recorrido. Asi mismo, en la interfaz de usuario 2D, dondé existe la concordancia entre la pulsación del mouse y la reacción ante este.

4.2.2.4.3.1 Procesamiento de la Mesh Geometry

La clase *mesh* (malla) permite acceder al script de la geometría de un objeto, permitiendo modificar o crear meshes en el tiempo de ejecución. Está técnica es especialmente util para efectos gráficos (estirar o aplastar un objeto), sin embargo también es utilizada para el diseño de nivel y optimización de los procesos internos. (Technologies, 2017)

Tal cómo se menciono el proceso de modelado a lo largo del documento, un mesh esta compuesto por triangulos organizados en un espacio 3D, para generar la impresión de un objeto solido. Un triángulo se define por tener tres puntos que corresponden a las esquinas (vértices), los cuales seran utilizados como parámetros en varios procesos de

interacción. Esta clase, almacena los vértices en una sola matriz y cada triángulo está especificado mediante tres enteros que son correspondientes a los índices de la matriz de vértices. Así mismo, los triángulos se agrupan en una sola matriz de enteros, lo cuales, se invocan en grupos de tres desde el inicio de la matriz, y por lo tanto los elementos 0,1 y 2 conforman el primer triángulo, posteriormente 3,4 y 5 el segundo, y así hasta el último elemento.

Estos arreglos no solo representan los vértices, triángulos, normales (vectores) y coordenadas de las texturas aplicadas, sino también, provee datos de otras propiedades útiles y funciones para colaborar en la creación de una malla. La información de una malla es adjuntada a un objeto mediante el componente *mesh filter* a parte del proceso de renderizado para hacer la geometría visible. Este componente se encuentra dentro de la función *GetComponent*

```
var mf: MeshFilter = GetComponent.<MeshFilter>();
// use mf.mesh to refer to the mesh itself.
```

Tabla 4.7 Función GetComponent de Unity para el manejo de mesh. (Unity Technologies, 2017).

El objeto mesh contiene propiedades para programar los vértices e información asociada como normales y coordenadas UV, que también funcionan para configurar el triángulo en general. Las UV's (técnica de texturizado) deben estar ordenadas para asignar a los vértices correspondientes el elemento asignado, recordando que los vértices están posicionados mediante vectores en el espacio local del objeto.

En cada vértice existe asociada una normal, donde generalmente hay dos coordenadas de textura, color y tangente. Toda esta información se almacena en matrices separadas del mismo tamaño, lo cual quiere decir que si la malla tiene 10 vértices, también se generarían matrices de 10 tamaños para normales y sus atributos. (Technologies, 2017)

```
using UnityEngine;
using System.Collections;
class ExampleClass : MonoBehaviour {
    void Update () {
        Mesh mesh = GetComponent <MeshFilter> (). mesh;
        Vector3 [] vertices = mesh.vertices;
        Vector3 [] normals = mesh.normals;
        int i = 0;
        while (i < vertices.length) {
            vertices [i] += normals [i] * Mathf.Sin (Time.time);
            i ++;
        }
        mesh.vertices = vertices;
    }
}
```

Tabla 4.8 Atributos modificables en la función mesh. (Unity Technologies, 2017).

En código mostrado en la *Tabla 4.8*, es un ejemplo de cómo se obtienen los vértices seleccionados de un objeto para ser modificados y asignarlos nuevamente a su posición dentro de la malla.

4.2.2.4.3.1 Rendimiento Gráfico

Dentro del procesamiento gráfico de una aplicación, se suele hablar del precio computacional que se divide en dos partes: el GPU y/o CPU. Uno no necesariamente excluye al otro, sin embargo para mejorar el rendimiento las técnicas aplicables son diferentes. Los siguientes puntos, son recomendaciones echas por la guía de optimización de *Unity Technologies, 2017*.

- El GPU suele ser limitado por el ancho de banda de memoria.
- Se debe cuidar la resolución de la pantalla, lo mejor es mantener una baja resolución antes de ejecutar la aplicación. Esto hará que la fluidez de sistema se acelere, por supuesto, bajo ciertas limitaciones propias del GPU.
- El CPU se encuentra limitado, en ocasiones, por el número de batches que necesitan ser renderizados.
- Es de utilidad marcar los “batches” en la ventana de “estadísticas de renderizado”. Entre mayor batches, conocidos también como *lotes*, estén siendo renderizados, mayor será el coste del CPU.

Los problemas menos frecuentes en una aplicación desarrollada con las características que aquí se presentan, son:

- El GPU procesa demasiados vértices. Como se menciona en el apartado de Unity en el Marco Teórico, la cantidad de vértices aceptables lo determina el GPU y la complejidad de los *vertex shaders*¹⁶. Es recomendable no tener más de 100,000 vértices en móviles. Un PC puede soportar varios millones de vértices, no obstante, es una buena práctica mantener un número mínimo para optimizar. (Unity Technologies, 2017)



Fig. 4.16 Proceso gráfico de los shaders. (Ziff Davis, 2017).

- El CPU, también tiene muchos vértices que procesar. Los cuales pueden estar en *skinned meshes*, simulación de telas, partículas, u otros *game objects*. Por lo cual sigue siendo una buena práctica mantener el número de estos en lo más mínimo, sin sacrificar la calidad de la aplicación.
- Si al comprobar que el renderizado no es problema para el GPU o para el CPU, y sigue existiendo algún problema de procesamiento, este debe recidir en otros aspectos como los scripts o físicas del entorno.

¹⁶ Función programable en tarjetas gráficas que ofrece flexibilidad al programador para renderizar una imagen. Se utiliza para transformar los atributos de los vértices, como son: color, textura, posición y dirección desde el espacio de color original hasta el espacio de visualización. (Group, 2017)

- Para renderizar un objeto en la pantalla, el CPU realiza mucho del trabajo pues, se encarga de las luces que afectan un objeto en particular, configura el shader y sus parámetros, y enviar comandos de dibujo al driver gráfico, quien posteriormente prepara los comandos para que sea enviado a la tarjeta gráfica.

En la ventana de game, existe una opción denominada “stats”, la cuál muestra las estadísticas del tiempo real de la renderización, dicha información resulta útil para mejorar el rendimiento de la aplicación. Por la naturaleza de la aplicación y la técnica de modelado implementado, los vértices procesados sobrepasan los miles, sin embargo de mantienen debajo de lo sugerido.

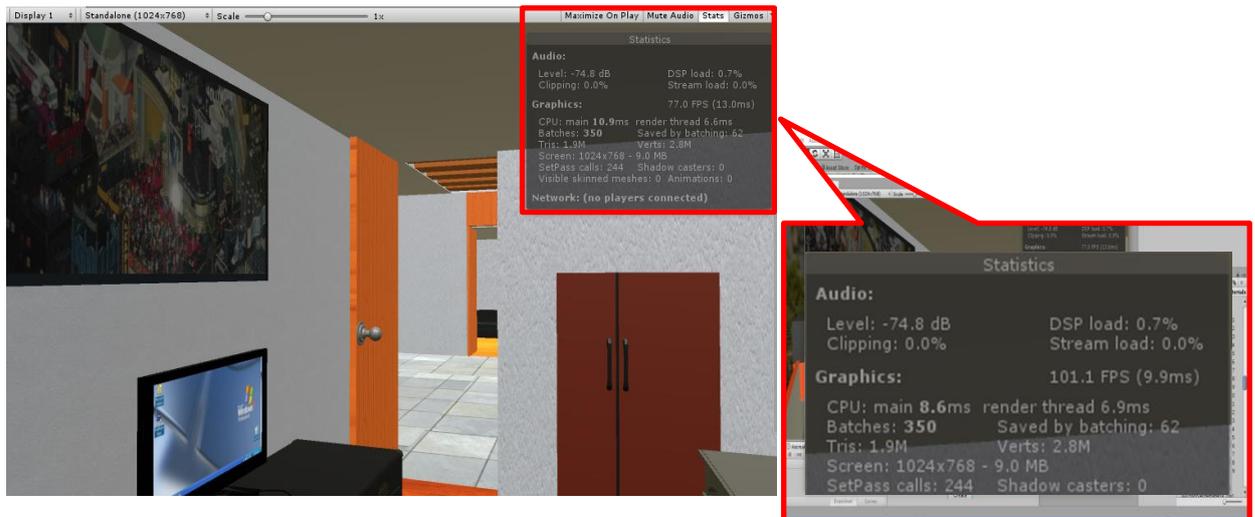


Fig. 4.17 Ventana game con las estadísticas de renderizado. (Elaboración propia).

Propiedad	Descripción
Time per frame and FPS	Es el tiempo que toma para procesar y renderizar un fotograma de una aplicación (fotogramas por segundo). Este número solo incluye el tiempo que toma para hacer la actualización del fotograma y renderizar la vista, lo cual quiere decir que no toma en cuenta el tiempo que toma el editor para dibujar la “vista de escena”, inspectores, y otros procesamientos del editor.
Batches	Se trata del número de “draw calls” inicialmente separadas que fueron agregadas a los batches. “Batching” es dondó se intenta combinar la renderización de varios objetos en un solo “draw call”, con el propósito de reducir la carga del CPU. Un buen batching, se crea a partir de compartir materiales entre los objetos, lo más posible.
Saved by batching	Número de lotes combinados del proceso de batching.
Tris and verts	Cantidad de triángulos y vértices dibujados.
Screen	Tamaño de la pantalla y uso de memoria.
SetPass	Cantidad de pases de representación. Cada paso enlaza un nuevo shader.
Visible Skinned Meshes	Cantidad de skinned meshes renderizados.
Animations	Número de animaciones reproduciéndose.

Fig. 4.18 Información de ventana para renderizar estadísticas. (Adaptación Unity Technologies, 2017).



Fig. 4.19 Ventana “Profiler” con el estudio de ejecución del entorno virtual. (Elaboración propia a partir de Unity 5.5.2).

La línea de tiempo presentada está conformada por varias áreas: El uso del CPU, Rendering y Memory, las cuales son principales por ofrecer datos más fundamentales en la optimización del rendimiento gráfico. Estas secciones pueden ser omitidas dando click en el botón de cerrar dentro del panel, y si es necesario, reincorporarlas usando el desplegable “add área” en la barra de los controles del perfilador.

En la parte superior de la ventana de profiler, se puede observar los datos del rendimiento en el tiempo, cuando se ejecuta la aplicación la información generada es guardada en cada cuadro y la historia de los cientos de cuadros finales se visualiza. Cuando se da click en un cuadro en específico, se mostrarán los detalles de este en el apartado inferior de la ventana, tal como se muestra en la Fig. 4.19; donde se eligió uno de los cuadros intermedios en cuanto en procesamiento del CPU.

Se debe tener presente que los cuadrados coloreados en el área de la etiqueta pueden controlar si se muestra la línea de tiempo asociada o no. Las muestras también se pueden manipular para que solo aparezca la información que se necesita según el objetivo de cada proyecto. Para ocultar una muestra de la pantalla se debe hacer click en la tecla de color. Lo cual provocará que la tecla se atenúe, y la información correspondiente será removida de la gráfica. Entre otras, esta opción ayuda a identificar la causa de los picos de la gráfica de la CPU.

En general, el resultado de la evaluación ha marcado niveles promedio en costos computacionales, sin embargo, ha marcado un procesamiento pesado en los scripts utilizados, por lo cual se evaluarán individualmente para poder ser modificados de ser necesario sin que esto comprometa el funcionamiento general del entorno, (para resumir información, no se presentarán todas las secciones mostradas en el estudio, no obstante, fueron debidamente analizadas).

4.2.2.4.4 Módulo de Movimiento

Este módulo está ligado directamente del desarrollo de la física dentro del entorno, sin embargo se alimenta de todo el comportamiento del sistema en general. Se basa en la definición de las características de los acontecimientos físicos, para poder determinar movimientos propios de un objeto o animaciones dentro del entorno. (Santiago, 2015)

En este módulo converge la cinemática de la física dentro del entorno y los movimientos específicos programados. Es decir, en este apartado se manejan características específicas de los movimientos, por ejemplo, el movimiento de un avatar ante cierto estímulo. (Santiago, 2015)

Tal como se puede apreciar a lo largo de la documentación presente, no existe un avatar definido para el usuario, por motivos de inmersividad, sin embargo no se descarta como entrega futura. En lo que respecta a lo desarrollado hasta ahora, se habla específicamente de la propuesta de un módulo encargado específicamente de proveer información acerca de avatares u objetos en específico, para crear un movimiento realista que aporte mucho más dinamismo al entorno. Ya que no solamente es importante que el entorno alrededor del usuario tenga similitud con la realidad, sino también cada elemento que lo conforma.

4.2.2.4.4.1 Propiedades

Hasta el momento se tiene propiedades básicas que conforman parte de la cinemática de la física del sistema de una manera más específica a la que fue definida en la geometría del modelo. Esta se basa en la información de la geometría y del comportamiento del usuario.

- Movimientos del usuario (eje “x”, “y” y “z”)
- Cinemática física del entorno (colisiones, aceleración, gravedad, etc)

Como todo módulo se rigen bajo ciertas propiedades que ayudan a procesar los datos para reconocer eventos programados.

- **Centro de gravedad.** Tal como su nombre sugiere, se trata del punto central de un objeto, el punto de origen definido en su creación. (Gonzalo Luzardo, 2014)
- **Masa.** Característica fundamental para el funcionamiento de cinemática a bajo nivel, se trata del peso de cada objeto. Gracias a esta propiedad los objetos reaccionan pertinentemente a colisiones y a campos de fuerza. (Gonzalo Luzardo, 2014)
- **Fricción.** A medida que la precisión de movimiento de los objetos aumenta, esta característica adquiere relevancia, pues será esta quien indique la resistencia del desplazamiento respecto a otro objeto. Esta característica suele ser utilizada sobre todo en las normales (caras) del objeto. (Gonzalo Luzardo, 2014)
- **Rebote.** Es una propiedad propia de la física de los objetos, sin embargo para aplicar mayor precisión, puede ser programada de manera particular, aunque no es muy común en el desarrollo de entornos de índole exploratoria, como lo es el

presente proyecto. Suele utilizarse, sobretodo, en videojuegos de diversas tematicas. (Gonzalo Luzardo, 2014)

4.2.2.4.4.1 Eventos

Los eventos son aquellos movimientos que son desencadenados a partir de la información recibida. Los eventos programados para hacerse visibles a partir del comportamiento del usuario son informativos, es decir, auxlian al usuario a lo largo de recorrido con información acerca del inmueble, por ejemplo:

- Dónde se encuentra.
- A dónde va a acceder.
- Características particulares sobre el espacio que visita.
- Proximidad de ojetos.

Hasta este momento, los eventos dentro del entorno son sobretodo informativos, para poder interactuar con el usuario y que este aprenda acerca del lugar que esta visitando de manera virtual, de esta menra la información leida en el menú principal se complementa con una experiencia mucho más vivida, sin limitarse a leer texto y ver imágenes planas.



Fig. 4.20 Algunos mensajes que aparecen a lo largo del recorrido en el entorno. (Elaboración propia).

Los mensajes son respuestas a los movimiento del usuario, conforme este vaya recorriendo el inmueble visualizara diferente información en referencia a su posición.

4.2.2.4.5 Módulo de Comportamiento

Es aquí donde se diseña el modelo de comportamiento que describe el comportamiento del sistema en términos generales, lo cual coverge perfectamente con el módulo de movimiento complementando la información necesaria para el algoritmo del agente inteligente. Este nivel se enfoca en el sistema semántico donde todos los elementos que conforman la aplicación deben ser definidos, y en caso de ser requerido se debe asignar caractesticas propios, clasificarlas y definir las en un contexto universal en base al motor de desarrollo. Dicho en otras palabras, es la conexión que hace posible el que un elemento del EV al sufrir un cambio en su comportamiento suare circundante note este movimiento.

Los modelos de comportamiento son aplicados para describir la conducta del sistema en su totalidad. Para ello existen diferentes modelos entre los que destacan:

- **Modelos de flujo de datos.** Estos modelan el procesamiento de los datos en el sistema. (M.en C. Pozas Cárdenas Mariano Javier, 2015)
- **Modelos de maquinas de estados.** Se refiere al modelado de las reacciones del sistema ante los eventos que susciten. (M.en C. Pozas Cárdenas Mariano Javier, 2015)

Estos modelos ofrecen la ventaja de poder usados individualmente o en conjunto, según las los objetivos del sistema desarrollado. Para el presente proyecto, el mayor enfoque se tiene en las reacciones del sistema ante las decisiones del usuario, por tal motivo se maneja *el modelo maquina de estados*, mediante el agente inteligente quien sera el encargado de gestionar comportamiento del entorno.

Como se aprecia en el esquema de arquitectura del entorno, al inicio de la presente sección (Fig. 4.8), este módulo sera quien dara los datos más relevantes al “agente usuario” para retroalimentarse y tomar decisiones.

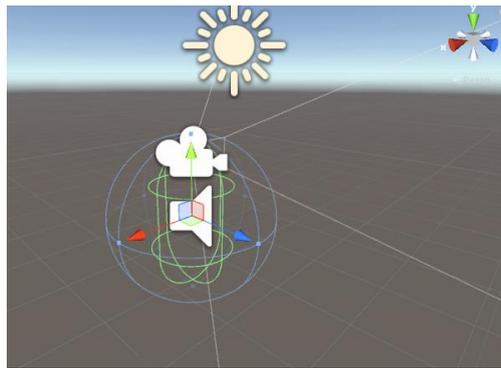


Fig. 4.21 Cámara y controlador en primera persona de Unity. (Unity Technologies, 2017).

```
int hrz;
int vert;
rightHandTracking : GameObject;
rightHandActive : GameObject;
horzRightScale : float;
vertRightScale : float;
initRightPos : Vector3;
runningRightPos : Vector3;
Inicialización
funcion Inicializar () {
    motor = GetComponent(CharacterMotor);
    rightHandTracking = GameObject.Find ("RightHandTracking");
    rightHandActive = GameObject.Find ("RightHandActive");
    rightHandActive.SetActive(true);
    vertical = 0;
    horizontal = 0;
Los valores de escala horizontal y vertical de abajo se relacionan con las “caja contenedora”
    horzRightScale = 10.0f;
```

```

    vertRightScale = 10.0f;
    initRightPos = rightHandTracking.transform.position;
    runningRightPos = rightHandTracking.transform.position;
}
El bloque siguiente hace que el FPC actualice la posición del controlador
funcion Update () {
    si(rightHandActive.activeSelf){
        runningRightPos = rightHandTracking.transform.position;
    }
    sino{
        rightHandTracking.transform.position = initRightPos;
        runningRightPos = initRightPos;
    } fin si
Obtener el vector de entrada del teclado (mando analógico)
Input.GetAxis("Vertical");
var directionVector = new Vector3(horizontal, 0, vertical);
si (directionVector != Vector3.zero) {
Se obtiene la longitud del vector de dirección
    var directionLength = directionVector.magnitude;
    directionVector = directionVector / directionLength;
Asegurarse de que la longitud no sea mayor que 1
    directionLength = Mathf.Min(1, directionLength);
Hacer el vector de entrada más sensible hacia los extremos y menos sensible en el medio de esta manera es más sencillo controlar velocidades bajas al usar mandos analógicos
    directionLength = directionLength * directionLength;
Multiplicar el vector de dirección normalizado por la longitud modificada
    directionVector = directionVector * directionLength;
} fin si
Aplicar la dirección a CharacterMotor
motor.inputMoveDirection = transform.rotation * directionVector;
motor.inputJump = Input.GetButton("Jump");
}

```

Tabla 4.9 Pseudocódigo de las funciones de controlador de personaje principal. Adaptación de trabajo Controlador en 1ra persona (Lynn Thompson, 2015).

El pseudocódigo descrito en la *Tabla 4.9* muestra las funciones principales que permiten controlar el comportamiento del usuario en primera persona, esta información es fundamental para que el “agente usuario” funcione debidamente, pues en base a este comportamiento el algoritmo del agente determinara cuál es el mejor camino dentro del inmueble, dando prioridad a las ordenes del controlador.

4.2.2.4.5.1 Objetivos

Los objetivos en este apartado completan al objetivo general definido en el agente inteligente “user”, solo que en este caso se centra en el área circundante, para controlar las reacciones del entorno respecto a las acciones inmediatas del usuario.

4.2.2.4.5.2 Creencias

Las creencia en el comportamiento del controlador principal, se basan en ciertos aspectos de interacción con el entorno. Lo cual ayuda a reconocer y crear conexiones en la interfaz lógica.

- **Simplicidad.** Se trata de identificar si existe ambigüedad en los objetos, es decir, si se percibe dos (o más) objetos que estén constituidos por los mismo elementos, el algoritmo pueda escoger ir por el más simple.
- **Proximidad.** Identificar objetos cercanos, los cuales se perciben como un cuerpo sólido con características definidas.
- **Similaridad.** Analiza elementos próximos similares (color, forma, etc).
- **Simetría.** Detecta las regiones rodeadas por contornos simétricos que mandan la noción de un objeto “coherente” a su entorno de interacción.

4.2.2.4.5.3 Acciones posibles

Las acciones posibles son en primer instancia, tomadas de manera directa por el usuario, es decir, bajo el procesamiento del algoritmo del agente, pero dándole prioridad a las instrucciones del usuario. La importancia del módulo comportamental, reside en que es el encargado de la integración de personajes autónomos, dado que el agente desarrollado da cierta libertad al usuario de tomar ciertas decisiones, el algoritmo es el que toma las decisiones importantes. Esto resulta como una ventaja en el momento en que se desarrollen más personajes dentro del entorno, pues habrá más oportunidad de interacción con agentes que completamente autónomos u otros jugadores en la misma escena.

Es importante recalcar esto, ya que no se trata de información local centrada en cada objeto o en el personaje principal (cómo en las secciones de física y cinemática), sino de elementos embebidos funcionando en conexión con todos su entorno.

Las acciones próximas posibles son definidas por el agente inteligente, las cuales se describen en la siguiente sección. Aunque la interacción dentro del entorno es de cierta manera limitada (objetos estáticos), y se reduce a la reacción del usuario, la finalidad del diseño está enfocada al comportamiento inteligente del entorno y el usuario en convergencia. La navegación a través del entorno, se realiza mediante el modelo geométrico, cuyos datos son utilizados por el algoritmo de búsqueda para poder tener la información necesaria actualizada y sustituirla según se determine las operaciones de optimización. (Gonzalo Luzardo, 2014)

Gracias a esta base de información manejada por matrices, el entorno en el que se desarrolla el agente reactivo deriva en una arquitectura con características cognitivas, ya se trabaja el comportamiento desde una perspectiva mucho más general. Dicho en otras palabras, la cuestión que se maneja es animar un conjunto de acciones consistentes, tomando en cuenta los objetivos definidos para el agente y el estado del entorno virtual. (Gonzalo Luzardo, 2014)

4.2.2.4.6 Técnica de IA Incorporada: Agente Reactivo Basado en Modelos

En este apartado se explicara la técnica de inteligencia artificial aplicada al comportamiento de la representación virtual del usuario y objetos del entorno, con el fin de garantizar una completa comunicación entre estos.

Para comenzar, se debe recordar que un agente tipo reactivo basado en modelos será la técnica de IA elegida para implementar en el entorno virtual desarrollado. Esta fue seleccionada mediante una previa investigación en el apartado de *Agentes Inteligentes* en el *Capítulo III*; en dicha sección, se analizaron diferentes algoritmos de los agentes inteligentes más utilizados en aplicaciones interactivas, por lo cual surgió la idea de implementar esta técnica con base en las características del comportamiento del mismo usuario con el entorno en donde interactúa. A partir de esta integración se desarrolla el modelo de conducta del agente.

Por otro lado, es importante resaltar que justamente al añadir este proceso al *recorrido/entorno virtual interactivo* (usuario puede recorrer el modelo 3D e interactuar con los objetos) el entorno adquiere la categoría de *entorno virtual inteligente*. Una de las principales motivaciones de implementar esta técnica es por las futuras líneas de desarrollo, pues los procesos funcionan mejor entre sí, sin necesidad de replicar métodos o clases. Lo cual incrementa la capacidad interactiva y comportamental del entorno en cuestión, mediante la integración de motores de razonamiento simbólico interpretativo, en este caso enfocadas al usuario.

El recorrido del usuario basado en el agente reactivo funciona bajo las siguientes especificaciones:

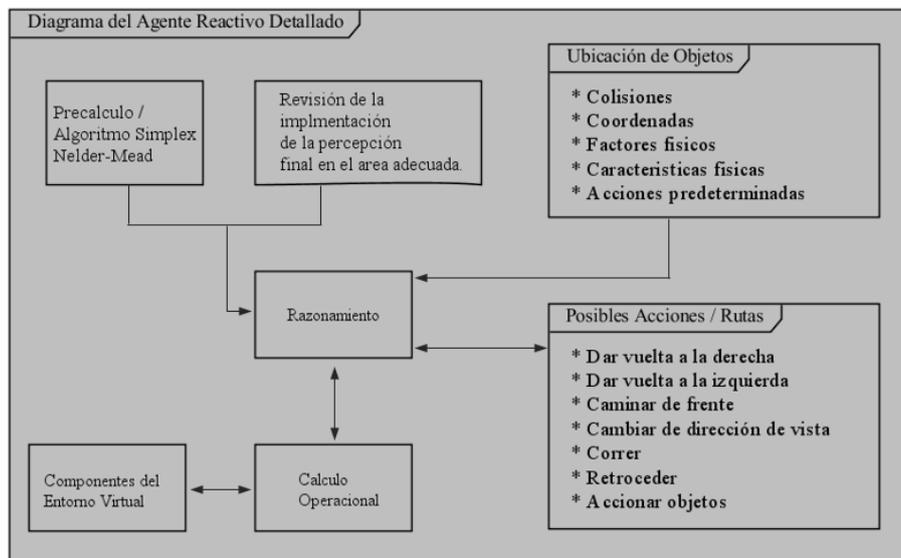


Fig. 4.23 Diagrama detallado del “agente usuario” basado en modelos. (Elaboración Propia).

4.2.2.4.6.1 Descripción Detallada del “Agente Usuario”

Para el diseño del diagrama del agente inteligente se baso en criterios fundamentales que se han ido describiendo a lo largo del *marco teórico*, pues para desarrollar un entorno

virtual aplicando una técnica de IA, se deben cumplir ciertos parámetros para llegar a una convergencia óptima.

- **Objetivos Específicos.** El entorno virtual diseñado tiene como objetivo particular dar visibilidad a uno de los centros de investigación y por ende, espacio de cultura, más sobresalientes de esta índole en la Ciudad de México, sin embargo, también ofrece ventajas empíricas en su aplicación. La primera es identificar el camino más corto para llegar a concluir el recorrido sin demorar en dobles visitas, y en base a ello tomar la acción pertinente de acuerdo a la percepción captada. La segunda es poder conocer el inmueble con información fidedigna, desde las medidas exactas de la arquitectura, como parte del material y objetos que se encuentran dentro de cada laboratorio. Así mismo, una localización geográfica fiel al inmueble original, lo cual significa una ventaja para organizar exposiciones dentro de las instalaciones y/o información espacial en algún siniestro.
- **Funcionalidades.** Las funcionalidades definidas son propias de la física de la interacción, como se mencionó anteriormente en los módulos comportamiento y movimiento, lo cual habla de identificar obstáculos en el recorrido para poder superarlos (sillones, sillas, paredes, computadoras, etc), y seguir la ruta posibilitada por el agente.
- **Posibles escenarios.** Los escenarios disponibles en este entorno virtual son básicos de un paseo por un inmueble cultural, ya que mayormente la actividad se limita a la apreciación de la arquitectura, muebles, espacios, herramientas, etc. Así como ciertos letreros que le brindan información al usuario de su posición actual.
- **Percepción.** Las percepciones que el entorno recibe son las seleccionadas por el usuario, ya que según estas el entorno reaccionará y su vez el usuario toma una decisión, es un ciclo continuo que determina la interacción del usuario con la aplicación (colisiones, coordenadas, vistas, etc).
- **Acciones.** Las acciones son las respuestas directas de las percepciones, las cuales resultan en el comportamiento de la representación virtual del usuario, el cual se maneja en primera persona, y puede, según el estímulo, retroceder, avanzar, brincar, correr, etc.
- **Protocolo.** El protocolo de interacción entorno virtual-usuario está establecido por la técnica de IA aplicada, en este caso en particular se escogió el desarrollo de un agente inteligente reactivo, el cual determinará la acción que realizará el usuario en su representación virtual y/o elegir la ruta más factible a seguir en base al objetivo principal del recorrido. Esto es logrado gracias a la información que maneja desde la actualización de la ubicación del usuario para que el entorno responda a corde (nivel gráfico), lo cual quiere decir que también define el cambio en el entorno (subir una escalera, mirar a través de una ventana).

Es fundamental tener claro que el usuario dentro del entorno, su representación virtual será el agente reactivo, el cual, recibirá información sobre la ubicación de los objetos del entorno para poder moverse con naturalidad. El objetivo es simple, recorrer todas las salas del inmueble e interactuar con los objetos disponibles ofreciendo la ruta

más óptima hacia dicho objetivo, sin limitar al usuario en sus decisiones. Dentro del recorrido el usuario puede seguir la ruta marcada o elegir por dónde desea comenzar, aunque signifique mayor coste, esto es posible gracias al previo cálculo de operaciones dentro del entorno, y el recalcado del algoritmo que provea la información de los datos necesarios.

4.2.2.4.6.2 Algoritmo Simplex Nelder-Mead

Para poder cumplir el objetivo en base a las interacciones mediante estímulos exteriores, se utilizara un algoritmo simplex. Se trata de un algoritmo de búsqueda directa, el cual tiene como principal atributo la optimización del tiempo de proceso para minimizar resultados de funciones multidimensionales. Esto brinda como principal ventaja el poder agregar datos y agentes para interactuar en el futuro desarrollo de la aplicación.

Varela (2015) menciona que este algoritmo es uno de los métodos más utilizados para la optimización no lineal sin restricciones pues se ajusta precisamente a lo que se requiere en esta aplicación. Se prioriza la libertad de decisión del usuario en el entorno, sin predefinir rutas lineales o actividades niveladas (cumplir cierta acción para pasar a la siguiente). Sin embargo, es importante recalcar que el hecho de “no tener restricciones” no significa que dentro del entorno no habra ciertas áreas bloqueadas que no pertenezcan al inmueble donde se desarrolla el recorrido o acciones no preestablecidas.

Este algoritmo funciona mediante tres estructuras para almacenar la información, una almacena los lugares que el usuario puede visitar, otra la información del lugar en donde se encuentra “temporalmente” y una última que funge como referencia para comparar el “el mejor camino”, así cada decisión que toma el usuario altera el entorno, además que la información recibida le ayudara al usuario a tomar una buena decisión; por ejemplo, si se dirige a un área que ya visito, o información que ya visualizo, de esta manera reducir el tiempo de recorrido. En el presente trabajo se tiene en cuenta solo un participante (usuario), que interactua con su ambiente en el cual hay una serie de posibilidades que se traducen a las iteraciones que hara la función de búsqueda. (Soto, 2012)

El hecho de que se busque que el usuario tome la mayor cantidad de decisiones, no inhibe ciertos límites que debe respetar el algoritmo, estos límites son definidos por la cantidad de “personajes” que interactuan en el entorno y la cantidad de posibles rutas u opciones disponibles. En resumen, el cálculo del agente, debe cumplir con superar la aleatoriedad en el menor número de repeticiones posibles para llegar a un respuesta óptima respecto a las circunstancias en tiempo real del usuario en la virtualidad.

Este algoritmo funciona mediante una exploración de las direcciones de búsqueda generadas por el proceso en cada iteración, esto hace que el algoritmo encuentre resultados satisfactorios en menos tiempo e iteraciones.

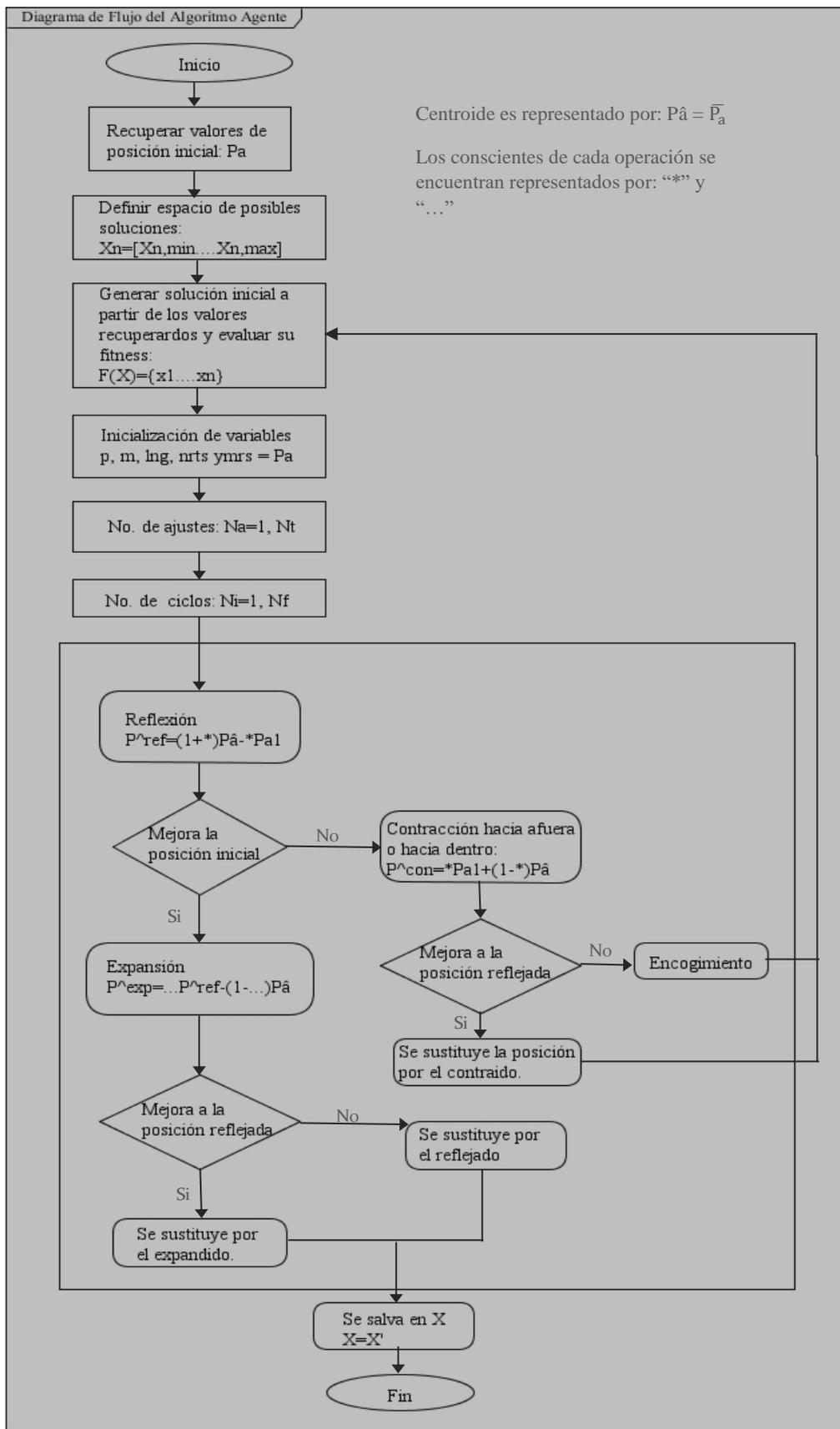


Fig. 4.24 Diagrama de flujo del Algoritmo NM. (Elaboración propia).

Como se menciona, anteriormente, este algoritmo parte de un modelo geométrico, el cual tiene ciertos parámetros que son evaluados para mejorar el rendimiento del recorrido tridimensional. Cada vértice del simplex representa una posible solución al problema en cuestión. En este caso, encontrar el mejor camino en el menor tiempo de respuesta. Su principal ventaja consiste en brindar precisión y calidad en cada uno de sus vértices (posiciones), esto se mide en base a una *función de coste o función fitness*. En este caso la transformación de posición inicial a la siguiente mejor posición según el objetivo, cada vértice de posición quedara representado por un vector.

$$P_a = (M_{x,y,z} F_{x,y,z}, \dots, M_{i,i,i} F_{h,h,h}) \quad (1)$$

Cuando se inicializa los puntos que conforman el simplex, el algoritmo procesa los vértices mediante una serie de operaciones con el objetivo de reemplazar a aquellos que representan las soluciones de menor calidad por vértices con más precisión. Después, en cada iteración del algoritmo efectuada se selecciona el mejor de los vértices P_a , y los peores P_i y P_h y se aplican en estos las operaciones de *reflexión*, *expansión* y *contracción*. El resultado de estas operaciones se traduce a nuevos puntos que sustituirán al peor de todos los vértices de la iteración en proceso. (López J. R., 2005)

La operación de reflexión genera nuevos vértices en una matriz F^{ref} , y un centroide \bar{P}_a a partir de la posición inicial P_a . Si este mejora respecto a los vértices iniciales, entonces se puede pensar en una expansión más allá del punto de reflexión, que pudiera mejorar aun más el resultado. (López J. R., 2005)

$$P^{ref} = (1 + \alpha) \bar{P}_a - \alpha P_{a1} \quad (2)$$

Posteriormente si el F^{exp} , mejora su valor en el fitness (al obtenido en la reflexión), se sustituyen estos valores en el simplex que contiene la posición actual del usuario y se repite el proceso. En el caso contrario que la operación de expansión resulte con valores poco favorables, el F^{exp} sustituirá el último vértice que se procesara P_{a1} . (López J. R., 2005)

$$P^{exp} = \dots P^{ref} + (1 - \dots) \bar{P}_a \quad (3)$$

Cuando la operación de reflexión no genera una mejora en los valores de turno, entonces se debe aplicar una contracción sobre el peor punto. La contracción está regida por la siguiente expresión, en este caso " α " representa el coeficiente de contracción. (López J. R., 2005)

$$P^{con} = \alpha P_{a1} + (1 - \alpha) \bar{P}_a \quad (4)$$

La función fitness dará como resultado el cálculo de la distancia entre los espacios disponibles, mientras el arreglo inicial se encargará del cálculo del recorrido del usuario. Mediante los niveles de representación de información, se manejan listas de sus propiedades centradas en los objetos con los que existe interacción. De esta manera se

van heredando jerarquias de clases del motor gráfico, de esta manera hay una conexión congruente, entre el agente y su entorno virtual.

En este contexto, la función fitness estará determinada por los siguientes parámetros.

$$\text{Función Fitness} = \left(\left(\text{Lng}(M(p)) + \text{Pincl}(P_a) + \text{Nrts}(M(p)) \right) * 0.5 - \text{Mrs}(M(p)) \right) \quad (5)$$

- **p** : posición actual
- **Pincl(P_a)**: posición inicial
- **M(p)**: áreas adyacentes a la posición actual.
- **Lng(M(p))**: longitud de superficie disponible respecto a la posición.
- **Nrts(M(p))**: posibles rutas respecto a la posición actual.
- **Mrs(M(p))**: número de paredes en las posiciones continuas a la actual.

La función fitness (5) representa la meta en el entorno virtual pues en cada iteración generada por el algoritmo buscará la mejor ruta respecto a la posición más actual mediante las operaciones descritas anteriormente (2,3 y 4). En este procedimiento, se califica positivamente a los resultados que den posiciones cercanas a la entrada principal y se descartarán las posiciones que conduzcan al encierro del representante virtual, que maneja el usuario. Los valores de superficies y paredes se multiplican por 0.5 porque es el valor menor que pueden tener estos objetos en el entorno. Todos los resultados dados por las operaciones de reflexión, harán sus cálculos respecto al valor fitness que se vaya obteniendo en cada nueva posición y va mejorándola. (Soto, 2012)

4.2.2.4.6.3 Sistema de Navegación.

Para lograr la correcta participación del agente inteligente, muchos de los datos implícitos en la navegación dentro del modelo 3D, son fundamentales. Pues en base a ellos el algoritmo de optimización de caminos logra cometer su objetivo. Unity brinda las clases necesarias para reconocer estos datos y utilizarlos en los métodos generados.

La principal estructura utilizada en el agente es *NavMesh*, quien aporta los datos descriptivos de las superficies que son “navegables” dentro del entorno virtual. Esta estructura está contruida a partir de la geometría que constituye el modelo. Este proceso colecciona los render meshes (mallas renderizadas) y terrenos de los game objects disponibles, estructurando las superficies para posibilitar su navegación. Utilizando esta herramienta también es posible crear personajes independientes trazándoles una ruta determinada a lo largo del recorrido, esta característica es de gran utilidad cuando se requiere de la participación de más personajes respecto a la historia y objetivos del entorno (p. ej. enemigos, compañeros, extras, etc). (Technologies, 2017)

Otro factor importante a mencionar, es la clase “base” en el desarrollo con Unity, se trata de *MonoBehaviour*, pues de esta clase heredan todos los scripts creados. Esta clase brinda métodos básicos predefinidos de Unity, el cual se puede complementar con la programación física y comportamental determinada en cada proyecto. (Technologies, 2017)

Los principales métodos utilizados se describen a continuación de manera resumida.

- **Start.** Este objeto es utilizado en objetos instanciados, cuando el objeto se encuentra activo. Dicho en otras palabras, cuando un objeto instanciado se encuentra en estado inactivo, no podrá llamar a *Start* hasta que sea activado. Solo se ejecuta una vez, según sea programado, específicamente al aparecer las animaciones que indican los nombres dentro del inmueble.
- **Awake.** Es un método invocado cuando el objeto al que pertenece un script es instanciado o cargado en memoria. Es una ventaja para cuando se requiere inicializar variables dentro del objeto, asegurando la disponibilidad de ciertos valores para utilizarlos en otros scripts; esto es muy útil en la organización de la información que describe el comportamiento del controlador del usuario.
- **Update.** Cuando el *GameObject* se encuentra activo, este método es de los más utilizados, específicamente, es llamado una vez por *frame* (unidad mínima, en forma de imagen estática, de descomposición de un video). Debido a su constante ejecución, se debe tener cierta restricción y selectividad al elegir las operaciones que contendrá este método. De tal manera, es utilizada para controlar los valores de entrada en el entorno 3D y entidad principal, los cuales se ven reflejados en los desplazamientos y acciones disponibles.
- **OnEnable.** Método ejecutable cada vez que se activa un objeto. Propiedad similar al método de *Start*, sin embargo, este no solo se ejecuta cuando el *GameObject* se activa después de su instancia, sino que se convoca cada vez que el objeto pasa de estar inactivo a activo, este método se utiliza para activar eventos definidos dentro del entorno (p. ej. señalizaciones, información importante, anuncios.).

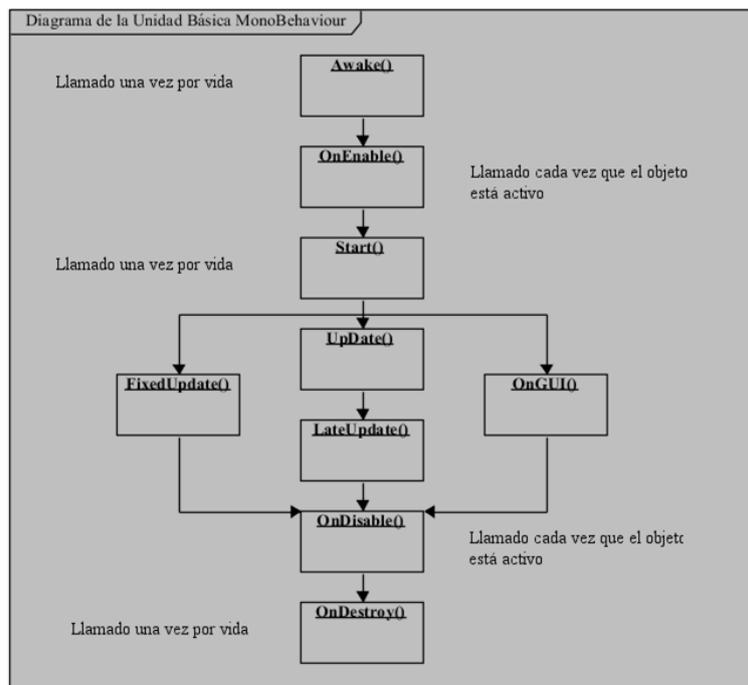


Fig. 4.25 Diagrama de la Clase de MonoBehaviour de Unity. (Adaptación Unity Videojuegos, 2014).

En el diagrama expuesto en la *Fig. 4.25* se puede observar el flujo de los métodos principales de esta clase a nivel local. También, se deben destacar las “corutinas”, estos metodos se pueden suspender durante algún tiempo determinado o dejarlo en espera de determinada sentencia.

4.2.2.4.7 GUI (Graphical User Interface)

La GUI (Interfaz Gráfica de Usuario), se encarga de comunicar al usuario información importante para poder utilizar un sistema, es el contacto directo aplicación-usuario. Lo cual logra mediante la presentación de elementos como botones, listas de opciones, leyendas, etc. Es utilizado por la aplicación en los menús, información durante el juego, HUDs, etc. (Cervigón, 2012)

La ejecución se debe diseñar de manera práctica para que el jugador pueda interactuar con el sistema de manera intuitiva. En el entorno virtual se contemplan leyendas que ofrecen información relevante para que el usuario pueda identificar los datos leídos, de manera previa, disponibles en la interfaz general que se describirá a continuación.

4.2.2.4.7.1 Desarrollo de la interfaz 2D para el sistema CNA RV

El desarrollo de una interfaz 2D se vuelve necesario en el proyecto, ya que además de un recorrido virtual se ofrece información acerca del inmueble y las actividades que ahí se imparten. Al ejecutarse el sistema, este inicia con una interfaz 2D, donde podrá elegir entre distintas opciones disponibles.



Fig. 4.26 Pantallas de Interfaz principal de la app CNA-RV. (Elaboración propia).

El desarrollo de la interfaz fue diseñada en base al tema de artes tecnológicas. La creación del fondo, botones e ilustraciones fueron mediante el software de diseño *Photoshop*. Por otro lado, para mantener la interactividad en su funcionamiento se crearon

una serie de animaciones que se ejecutan en cada transición de pantallas, selección de botones y utilización de barras de desplazamiento.

4.2.2.4.7.2 *Diseño de Interfaz*

Salvador Llobet (2015), habla de la interfaz gráfica como uno de los aspectos fundamentales en una aplicación. Ya que el usuario se encuentra en constante interacción con esta, es la comunicación directa entre usuario-app que permite seleccionar entre las opciones disponibles. También cita a Shneiderman (2005), para enlistar las normas que son necesarias cumplir para el diseño de una interfaz completa.

- **Diseño consistente y coherente.** Norma que se refiere a la similaridad de diseño entre los distintos menus (ventanas), lo cual quiere decir, aplicar una estructura común. (Llobet, 2015)
- **Accesos directos.** Estos están diseñados para los usuarios experimentados, y se refiere a evitar “pasos obligatorios” es decir, tener la posibilidad de interactuar con la interfaz de manera “rápida y fluida”. Cuando el usuario ya conoce el sistema puede volverse frustrante si este consiste en pasar por los mismos pasos, sin poder saltar lo que ya no considera necesario. (Llobet, 2015)
- **Retroalimentación.** A lo largo de la aplicación la retroalimentación es muy importante, ya que permite al usuario saber que opción ha seleccionado o si su petición se encuentra efectuándose o no. (Llobet, 2015)
- **Acciones secuenciales.** Cuando se realice una acción que requiera de muchos pasos, estos pasos deben ser señalados en el transcurso del avance. Este tipo de secuencias se deben componer por un inicio, intermedio y final, que deben estar visibles para el usuario todo el tiempo. (Llobet, 2015)
- **Gestión de errores.** Cuidar el diseño del sistema, para reducir el margen de error dentro de las opciones disponibles. En caso de que el usuario cometa algún error, se debe contar con un protocolo disponible para que el sistema pueda resolverlo. (Llobet, 2015)
- **Retroceso de acciones.** El usuario debe estar posibilitado a deshacer alguna acción elegida dentro del sistema. Dándole al usuario la confianza de que en caso de cometer un error al elegir alguna acción, esta pueda ser corregida. De esta manera, el usuario siente mayor libertad de navegar por la interfaz sin temor a provocar un error significativo. (Llobet, 2015)
- **Control.** Sin dejar de lado las limitaciones permitidas dentro de la interfaz, el usuario deberá tener el control total del manejo del sistema. Es importante crear la sensación de que el sistema responde a las acciones del usuario, y no viceversa. (Llobet, 2015)
- **Carga de memoria a corto plazo.** Las pantallas deben tener tal simpleza que los cambios de ventanas no sean tan frecuentes, y brindando un periodo de aprendizaje adecuado. Este factor, aunado a las acciones secuenciadas, evitará al usuario memorizar algún punto en el proceso iniciado. (Llobet, 2015)

La interfaz se compone de secciones de información respecto a los laboratorios que conforman el entorno virtual generado a partir del inmueble Cenart. De esta manera, se consigue que el usuario tenga un acercamiento mediante la lectura de objetivos, proyectos, e información relevante acerca del trabajo en los laboratorios que posteriormente visitara gracias al modelo 3D.

Para la conexión de la interfaz 2D con el recorrido 3D, se programo un controlador que se maneja a través de clases definidas para la construcción de la interfaz principal. Los eventos programados para la dinámica la interfaz se describen en la *Fig 4:25*.

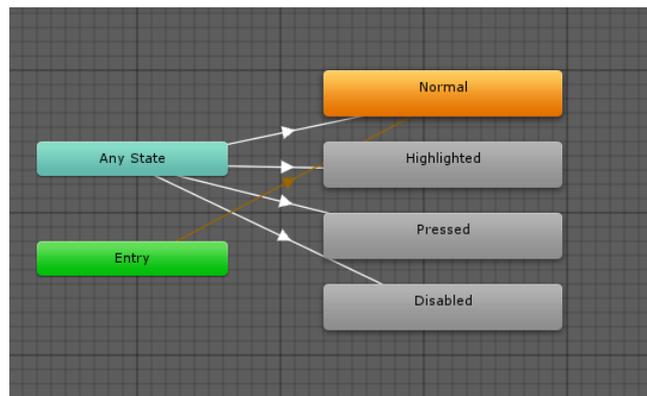


Fig. 4.27 Conectividad de eventos entre pantallas de la interfaz principal. (Elaboración propia).

Unity, con el propósito de minimizar los tiempos de creación de la interfaz, incorporo un sistema gráfico de diseño. Sin embargo, este se debe complementar con scripts con instrucciones específicas, en este caso, como ya se menciono, se genero un controlador para la interfaz principal llamado *ControladorMainMenu*. Se creo con la finalidad de tener mayor organización en el sistema general. De igual forma, en lo referente al entorno virtual 3D, fue necesario un “gestor de eventos” por lo que se creo otro controlador para lograr mejor comunicación con las clases, métodos utilizados y entre estos mismos.

Los scripts utilizados estan centrados en un concepto muy propio del desarrollo de videojuegos, se trata de “la maquina de estado finito”. Este termino se refiere a una maquina de estados finitos que en esencia define un sistema de interacción condicional, cuyas condiciones representan los *estados*. (Zaragoza., 2012)

Los *estados* pueden representar casi cualquier evento, desde si un objeto se debe renderizar hasta si debe estar vinculado con alguna ley física (encender o apagar una sombra, si un objeto debe rebotar, posición en pantalla, etc). Quien se encarga de agregar la interacción y administración de *estados* específicos utilizados, son los scripts que a su vez estan relacionados con diversos *GameObjects*. La comunicación entre estos scripts se implementa mediante el envio de mensajes y el llamado de ciertas funciones necesarias, es por ello que es fundamental saber como funciona esta estructura que a continuación se ilustra en la *Fig 4.28*.

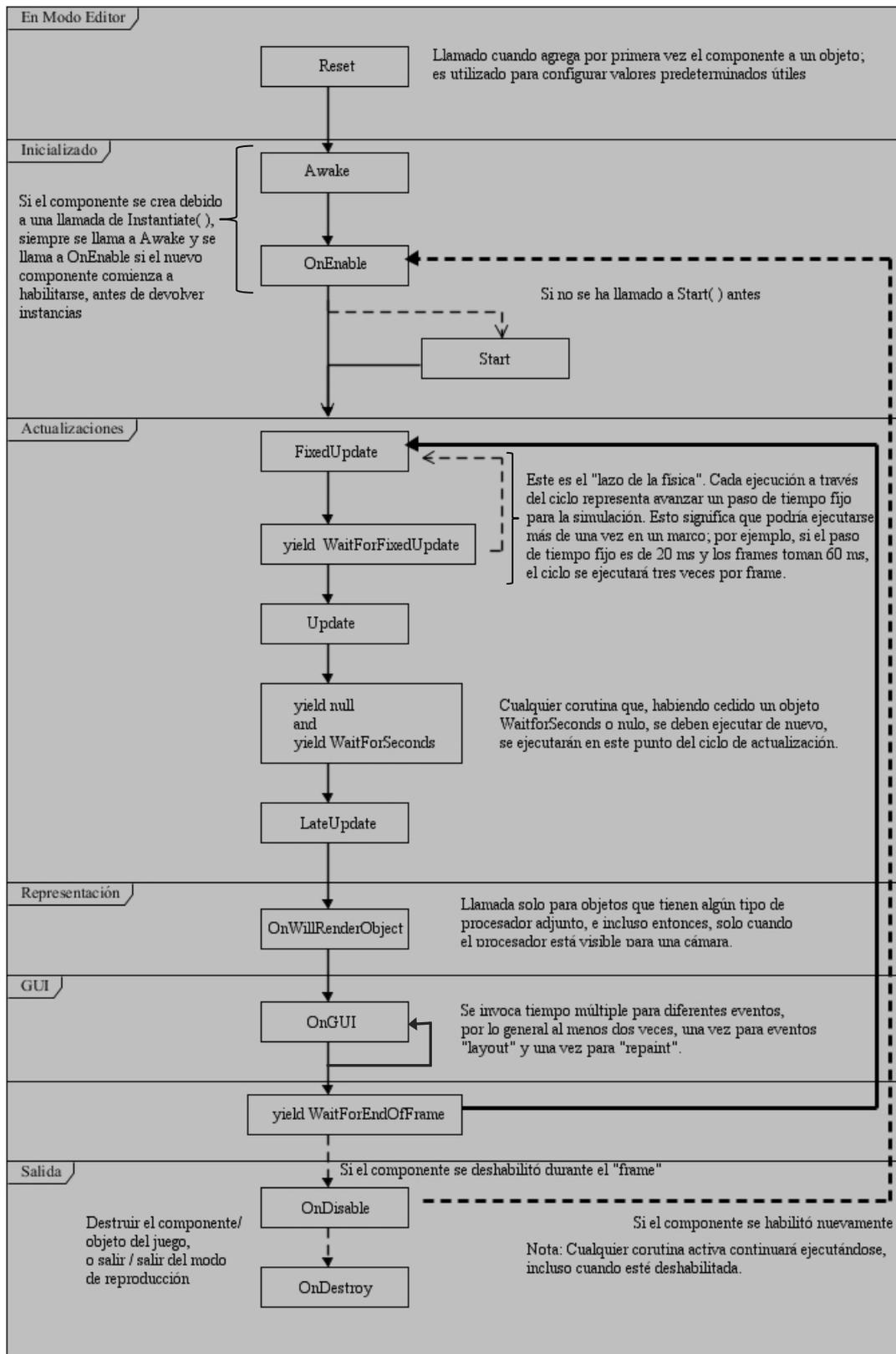


Fig. 4.28 Diagrama Detallado de MonoBehaviour y GUI. Adaptación de (Richard Fine, 2012).

4.2.2.4.7.3 Interfaz orientadas a la Realidad Virtual

Uno de las principales metas de una interfaz bien desarrollada es proveer efectividad y calidad a través de factores tecnológicos, psicológicos, físicos, etc. Para ello las herramientas para su creación se han vuelto más desarrolladas, más reducidas y con mejores prestaciones en lo que se refiere al consumo de recursos. Existen varios tipos de interfaces que están clasificadas respecto al objetivo que cada una tiene.

Se debe recordar que el presente proyecto fue diseñado bajo una arquitectura orientada a videojuegos, por lo cual la parte interactiva sobresale, no solamente en el recorrido 3D, sino también en la interfaz general del sistema “CNA RV”. Botones, menú, barras y todos los elementos que conforman la interfaz están orientados a un público interesado en la tecnología conjuntamente con el arte, a manera de hacer más atractiva la experiencia del usuario consumiendo información de una manera novedosa, tal como menciona Martínez Díaz (2010) en su trabajo “La importancia del diseño en los videojuegos. Interacción e interrelación”.

La interfaz orientada a la realidad virtual, no siempre está basada en un arquetipo inteligente, sin embargo resulta conveniente utilizar alguna técnica de IA. Esto permitirá una mejor comunicación usuario-máquina, tal como se ha explicado a lo largo de la arquitectura del entorno virtual. Específicamente, en el presente trabajo, solo se ha contemplado la utilización de un agente inteligente para controlar el comportamiento del recorrido 3D, este algoritmo no interviene en la ejecución del interfaz 2D. Por lo tanto, la posibilidad de desarrollo de un *agente de interfaz*, no es solo una opción, sino que dado el propósito del sistema, se convierte en una obligación, ya que se puede manejar la información del usuario para determinar sus gustos y en base a ello poder ofrecer mejores opciones de educación impartidas en el inmueble.

La interacción siendo tan importante en el paradigma de interfaces enfocadas a la realidad virtual, según Isidro Ramos (2010), está modelada según la siguiente expresión:

$$Interacción_i(usuari_o_j, tarea_k, contexto_l)$$

De esta manera, el sistema determina los servicios disponibles según el *contexto_l*, para que el *usuari_o_j* pueda realizar la *tarea_k*. El cambio de contexto hace que el sistema verifique los cambios en los servicios disponibles. Esto podría afectar a la actividad iniciada por el usuario, de tres modos: puede continuar igualmente, hacer necesario cancelarla o delegar su ejecución a otros servicios. Si es necesario utilizar otros servicios para continuar con la tarea, se realizar los cambios necesarios sin que el usuario tenga que configurar o hacer algo extra a su actividad. (Salavert, 2010)

EL diseño de una interfaz siempre requerirá de una evaluación por parte de una muestra de la población de usuarios al que va dirigido, implementando técnicas rigurosas y fiables. Los resultados de tales pruebas servirán para realizar mejoras de usabilidad en el sistema, ya que se realizó en una etapa avanzada del desarrollo, esto se explicará de manera más clara en el *Capítulo V: Pruebas y Resultados*.

C

APÍTULO V. PRUEBAS Y RESULTADOS

La ingeniería o la tecnología consiste en utilizar el poder de la ciencia para mejorar la vida de las personas, reducir los costos, mejorar el confort, mejorar la productividad, etc.

- N. R. Narayana Murthy

5.1 CRITERIOS DE ACEPTACIÓN DE UN PRODUCTO SOFTWARE

Para la evaluación de cualquier producto en el mercado siempre habrá estándares definidos para su valoración, en el caso del desarrollo de software este no representa una excepción. Pereira (2002), citando a Nielsen, (1993) menciona que las aplicaciones desarrolladas para uso educativo son útiles si los usuarios a los que van dirigidas aprenden de manera óptima a través de ellas, el mismo criterio se asume para diferentes tipos de aplicaciones con fines de entretenimiento, en el caso de un videojuego se debe lograr alguna utilidad, es decir, el objetivo para el cual fue creado.

Por lo tanto, el grado de aceptación de un producto software en su evaluación está directamente ligado a la usabilidad de este. A continuación, en la *Fig. 5.1*, se muestran los atributos del “grado de aceptación” según lo definido por Nielsen, para la valoración de un producto.

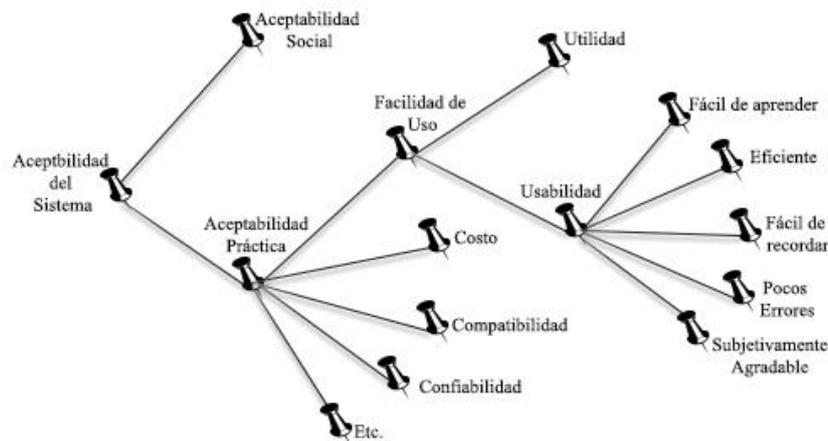


Fig. 5.1 Esquema de grado de aceptación de un producto. (Adaptación de Pereira, 2002).

Dado la naturaleza de la aplicación aquí presentada, la prueba del software se enfocará en el requerimiento de usabilidad que se menciona dentro del esquema anterior, tomando en cuenta los parámetros oficialmente establecidos para ejecutar una valoración consistente.

La información requerida dentro de una prueba de usabilidad deberá ser debidamente solicitada a lo largo de la generación de este. Esto se debe desarrollar con la finalidad de lograr ciertos requerimientos derivados de la usabilidad, los cuales son generalmente criterios, definidos por el usuario, que permiten marcar una pauta de mejora entre diferentes versiones del producto involucrado. Entre estos se puede encontrar un apartado que mida la experiencia del usuario, el cantidad de información y el funcionamiento general de la aplicación. Las características mencionadas hasta el momento permitirán ofrecer una experiencia mucho más completa para garantizar el aprendizaje cognitivo en diversos niveles del usuario. (PEREIRA, 2002)

5.2 MÉTODOS DE USABILIDAD DE SOFTWARE

Para validar el funcionamiento del sistema CNA RV, se someterá a un test de usabilidad desarrollado en base a criterios fundamentados a lo largo del *Capítulo V*, el cual será valorado por participantes de diferentes edades para conocer su aceptación en diferentes generaciones, tal como se estipula en los criterios vistos anteriormente de este proceso.

Existen varios métodos tradicionales de evaluación de usabilidad para el desarrollo de aplicaciones software, a continuación en la *Tabla 5.1* se muestran algunas categorías importantes en las que se han dividido estos métodos y así poder reconocer cuáles son aplicables para un EVI, según el “COST294-MAUSE”¹⁷. (Zumaquero, 2010)

DGMM	Data Gathering & Modelling Methods, por sus siglas en inglés, es un método utilizado para adquirir información sobre el usuario y sus actividades, lo cual es utilizada comúnmente para la comparación de productos. Esta se divide en las siguientes categorías:	DGM. Data Gathering Methods, este tipo de métodos se basan en recabar conocimiento sobre las características relevantes de los usuarios, las tareas y contexto en que se implementan los sistemas interactivos (p. ej. entrevistas, cuestionarios y/o test).
		MM. Modeling Method, su principal objetivo proveer información detallada de los usuarios y sus actividades. Estos métodos están generalmente relacionados a métodos de recolección de datos y/o sus combinaciones (p. ej. GOMS, HTA y CTTE).
UIEM	User Interactions Evaluation Methods, estos métodos se centran en la evaluación de las interacciones del usuario. Estos métodos se dividen en las siguientes categorías:	KMBM. Knowledge-based and Model-based Methods, están basados en los conocimientos que se encuentran disponibles y no requieren el acceso directo a los usuarios. Por lo cual, se suelen utilizar cuando no es posible recabar datos de los usuarios. A su vez, estos métodos están subdivididos en las categorías: evaluación por expertos, por documentos, y basada en modelos.
		EM. Empirical Methods, estos se basan en pruebas de usuario.

CM. Collaborative Methods, estos métodos se utilizan para el intercambio de ideas entre diferentes grupos de usuarios desde un punto de vista de usabilidad siendo valorada. Se necesita la participación directa del usuario (p. ej. lluvia de ideas)

Mixed Methodologies. Tal como su nombre lo sugiere, esta categoría se creó con el propósito de englobar los métodos que usan conjuntamente varios métodos para evaluar.

Tabla 5.1 Métodos de Usabilidad para el Desarrollo de Software. (Elaboración propia).

¹⁷ Comunidad de investigadores en el ámbito de usabilidad, cuya labor pretende incentivar la evaluación de la usabilidad mediante actividades científicas. (Zumaquero, 2010)

Los métodos clásicos de evaluación de software son aplicables al nuevo paradigma de interacción usuario-software, es decir, a las nuevas tecnologías de desarrollo de software como: la RV y/o RA.

5.2.1 EFICIENCIA DE LOS MÉTODOS CLÁSICOS DE EVALUACIÓN APLICABLES

Tal como menciona Zumaquero en su artículo, los métodos descritos por la *Cost294-Mause* se basan en criterios marcados en la definición de “usabilidad” propuesta por la *ISO 9241-11*¹⁸, los cuales son: **efectividad, eficiencia y satisfacción**. Sin embargo, algunos métodos están enfocados a evaluar aspectos específicos que no siempre engloban los tres criterios, por lo cual en algunas ocasiones se mezclan según los objetivos de la evaluación. A continuación se muestra la *Fig. 5.1*, donde se mide la eficiencia de los métodos de usabilidad.

	DGMM		UIEM		CM	Mixed Methods
	DG	MM	KMbM	EM		
Realidad Virtual		X	X			X
Realidad Aumentada		X	X		X	X
Sistemas Ubicuos	X			X		X

Tabla 5.2 Eficiencia de los Métodos aplicables según los Criterios de Usabilidad. (Zumaquero, 2010)

Estos métodos, en cualquiera de sus categorías son aplicables en el desarrollo de software en RV, dado a lo estipulado en la norma *ISO 9241-11*. Pues como cualquier producto software, un sistema en RV, para ser considerado útil necesita cumplir con algunos requerimientos tales como ser intuitivos, es decir, de fácil manejo, también debe contar con una interfaz atractiva y amigable, y además, debe permitir al usuario alcanzar su objetivo de manera rápida y clara. Todo esto debe ser cumplido consumiendo la menor cantidad de recursos computacionales.

En la evaluación, la simpleza del uso general del sistema está directamente relacionado con la satisfacción del usuario, de la misma manera todo sistema debe ofrecer una interfaz llamativa. La secuencia rápida de las imágenes y/u objetos dentro del entorno, es un aspecto importante que determina la efectividad, y el consumo bajo de recursos en la ejecución del sistema establecerá su eficiencia. (Zumaquero, 2010)

La aplicación de los métodos de evaluación de usabilidad en sistemas de RV no se diferencia en gran margen a la evaluación de los sistemas tradicionales. Sin embargo, tratándose de un nuevo paradigma de desarrollo, donde intervienen herramientas novedosas que proveen de ciertos comportamientos a las aplicaciones, se debe tomar en cuenta en la evaluación estas nuevas características para ser valoradas.

¹⁸ International Standar Organization. Normas que define la usabilidad y explica cómo identificar la información necesaria a considerar cuando se evalúa la usabilidad de una plataforma visual (hardware, software y/o servicio) en términos de medidas de rendimiento de usuario y satisfacción. (ISO, 1998)

Zumaquero, concluye en su artículo, que no todos los métodos tenían la misma eficacia evaluando las características de estos nuevos sistemas.

Por esta razón, se seguirán los métodos propuestos por este autor, añadiendo algunos aspectos importantes para cumplir con los objetivos planteados. Para evaluar el entorno físico característico de un entorno virtual se utilizara el método de *Interviews/Test* (perteneciente a la categoría DGM, anteriormente descrita), las sensaciones de usuario son fundamentales en la evaluación de este sistema, por lo cual los cuestionarios serán de gran ayuda para obtener resultados sustentables en este factor del sistema. Así mismo, la aplicación de métodos del grupo EM, donde es imprescindible la participación del usuario, brindara información acerca de la satisfacción del usuario utilizando el sistema.

	DGMM		UIEM		CM	Mixed Methods
	DG	MM	KMbM	EM		
Realidad Virtual	X			X	X	X
Realidad Aumentada	X			X	X	X
Sistemas Ubicuos	X			X	X	X

Tabla 5.3 Métodos de Evaluación de Usabilidad y su Aplicación a Diferentes Tipos de Software. (Zumaquero, 2010)

5.3 CARACTERÍSTICAS RELEVANTES EN LA EVALUACIÓN DE UN SISTEMA EN RV

Para poder desarrollar una evaluación óptima para el sistema desarrollado en el presente documento, se plantearon las siguientes características esenciales que deben estar presentes en un espacio virtual: (M.en C. Pozas Cárdenas Mariano Javier, 2015)

- **Sensación presencial.** Se refiere a la sensación realista de ser parte del mundo virtual creado.
- **Telepresencia.** Tiene que ver con que personas geográficamente distribuidas tengan presencia simultánea en el mundo.
- **Niveles de inmersión.** Se relaciona con que el espectador se sienta rodeado del entorno.
- **Amplio ancho de banda visual.** Es contar con despliegues panorámicos, que permitan colocar gran cantidad de información visual en el escenario.
- **Estereografía.** Contempla la percepción de profundidad y la ubicación en el espacio tridimensional de los objetos.
- **Interactividad.** Incluye la realimentación que se tiene con otros espectadores a través del medio de la realidad virtual, así como la respuesta en tiempo real del sistema computacional (y el mundo virtual) a las acciones tomadas por los usuarios.

En un sistema de RV, si se cumplen las expectativas de usuario y su experiencia (sensaciones) manipulando el sistema es buena, el nivel de satisfacción será elevado y el

sistema podrá considerarse válido, este resultado deberá llevar implícito el correcto funcionamiento del sistema, de lo contrario se anula (Zumaquero, 2010).

Lo cual nos habla de un alto valor en el criterio de satisfacción del usuario en este tipo de sistemas, por lo cual los métodos elegidos anteriormente, deberán cumplir con los requerimientos de evaluación. Como base para desarrollar el test y cuestionario de usabilidad requerido en el método DGM, se utilizara un cuestionario desarrollado para evaluar la usabilidad de sistemas multimedia implementados en la educación por Borges de Barros Pereira, Hernane (2002).

Generalmente, cuando se habla de usabilidad se asocian aspectos importantes como la definición del objetivo en el cual se enfocara, e identificar el tipo de usuarios cuya participación brindara información importante para la evaluación del sistema. Pereira, también menciona, citando a Velte (1989), que los objetivos de la usabilidad son comúnmente establecidos como premisas que permitirán la mejora del producto software a través de las versiones que genere su evolución. (Pereira, 2002)

También menciona la creciente evolución en desarrollo de software multimedia y su implementación, sobretodo, en la educación. Es importante tener en cuenta estos datos ya que en las mejoras futuras del presente sistema, se pretende incorporar algunas actividades pedagógicas para distintas disciplinas. Sin embargo, no se puede pasar por alto que las opciones ya incluidas en el desarrollo del *CNA-RV*, contienen un nivel básico de pedagogía, pues a través de la información presentada en el sistema, el usuario tiene la oportunidad de informarse y aprender acerca de un lugar determinado.

Los aspectos ergonómicos, establecidos por estándares de la ISO y centrados en la usabilidad y diseño de sistemas informáticos, toman como prioridad el punto de vista de los usuarios a los que van dirigidos, independientemente de las tecnologías de desarrollo de estos sistemas. A continuación se muestra la tabla con los estándares más importantes a tomar en cuenta en la evaluación de sistemas informáticos.

Categoría	Estándar Internacional	Descripción/Partes
Estándar Orientada a Proceso	ISO 6385 (1981)	Principios ergonómicos en el diseño de sistemas de trabajo.
	ISO 13407 (1999)	Procesos de diseño centrado en el hombre para sistemas interactivos.
	ISO 9241	Requerimientos ergonómicos para trabajos de oficina con terminales de visualización. Parte 1: Introducción General (1997) Parte 2: Guía sobre requerimientos de tarea (1992) Parte 11: Guía sobre usabilidad (1998)
	ISO 10075 (1991)	Principios ergonómicos relacionados con la carga de trabajo mental – Términos generales y definiciones.
	ISO/IEC 14598	Tecnología de la información – Evaluación de producto de software. Parte 1: Visión General (1999)

Estándar Orientado a Producto	ISO 9241	Requerimientos ergonómicos para trabajos de oficina con terminales de visualización. Parte 3: Requerimientos para la visualización en monitores (1992) Parte 4: Requerimientos para teclado (1998). Parte 5: Requerimientos de postura y “layout” para estaciones de trabajo (1998) Parte 6: Guía sobre el entorno de trabajo (1999). Parte 7: Requerimientos para el tratamiento de reflejo en monitores (1998). Parte 8: Requerimientos para el uso de colores en monitores (1997). Parte 9: Requerimientos para dispositivos de entrada sin teclado (2000). Parte 10: Principios de diálogo (1996). Parte 12: Presentación de información (1998). Parte 13: Guía de usuario (1998). Parte 14: Diálogos de menús (1997). Parte 15: Diálogos de comandos (1997). Parte 16: Diálogos de manipulación directa (1999). Parte 17: Diálogos para rellenar formularios (1998).
	ISO 11581	Tecnología de la información – Interfaces y símbolos de sistemas de usuario – Símbolos y funciones de iconos. Parte 1: Iconos – General (2000). Parte 2: Iconos de Objetos (2000). Parte 3: Iconos de punteros (2000). Parte 6: Iconos de acción (1999).

Tabla 5.4 Estándares de usabilidad de sistemas informáticos establecidos por ISO, 2da Parte. (Pereira, 2002)

Para cumplir satisfactoriamente los requerimientos señalados anteriormente, los cuales cumplen con un producto software óptimo. Pereira (2002) hace mención, citando a Gould y Lewis (1985), de los tres principios del diseño los cuales son: diseño iterativo, atención dirigida al usuario y a sus tareas, finalizando con las mediciones empíricas. Por lo cual, es importante mencionar que en el método formal de test de usabilidad, se debe cumplir con datos fiables y validos en los procedimientos, para satisfacer tales aspectos, el test y cuestionario será aplicado a un grupo de personas de diferentes perfiles. En este punto, lo recomendable, según diversos autores como Nielsen (1993), Rubin (1994), y Mayhew (1999), es considerar una población de muestra dentro de un rango de 4-10 participantes, dato que se tomara en cuenta para las pruebas del presente trabajo.

5.4 DESARROLLO DE LA EVALUACIÓN DE USABILIDAD

El cuestionario elaborado por Pereira (2002), está basado en el estudio de los alcances que conllevan las aplicaciones multimedia dirigidas al sector de educación, mayormente a plataformas multiusuarios donde se auxilia el aprendizaje de forma digital, sin embargo, estas bases concuerdan perfectamente con los criterios de usabilidad y aspectos requeridos en los métodos seleccionados para la evaluación. Por esta razón, se utilizara el test de usabilidad propuesto por este autor, que además aporta un cuestionario para recabar la información necesaria del usuario. Haciendo las modificaciones permitidas según el método elegido para evaluar el presente sistema añadiendo algunos aspectos importantes que caracterizan los entornos virtuales, se obtendrá las métricas necesarias para una óptima evaluación.

5.4.1 ASPECTOS MEDIBLES EN LA EVALUACIÓN DE USABILIDAD.

Al principio del *Capítulo V* se presentó información respecto a lo que se refiere una evaluación de usabilidad, los métodos aplicables y los estándares de evaluación. Sin embargo, no se conoce aún como se administran los datos obtenidos. A continuación se muestran de manera general la clasificación de datos que pueden derivar de estas evaluaciones y su manejo. En la evaluación de usabilidad se mide la manera en que el usuario realizara una tarea específica, número de clics, tiempo que se lleva en terminar, y los inconvenientes que se le presenten a lo largo de la ejecución del sistema. (S.L.)

- **Datos empíricos del funcionamiento de la aplicación.** Son el tipo de datos que muestran el tiempo en que los usuarios realizan cada una de las tareas, lo cual equivale a uno de los criterios que son utilizados para medir la eficiencia de la interfaz de usuario.
- **Impresiones de los usuarios.** Los usuarios dan puntos de vista subjetivos después de la prueba, mediante el método elegido surgirán los datos más significativos a cerca de su experiencia utilizando la aplicación, tales como complejidad en la misma, esfuerzo requerido, etc.
- **Observaciones por parte de expertos.** En caso de que se haya contemplado, existirá un equipo de expertos multidisciplinar que examinara las pruebas en tiempo real y/o utilizando recursos como grabaciones. Esta opción es utilizada cuando se requiere una valoración de mayor detalle con métodos heurísticos.

Como cualquier producto software, un sistema realizado con las características que se plantean en el presente proyecto, para ser de utilidad, tiene que cumplir ciertos factores formalizados como una manipulación sencilla, ofrecer una interfaz atractiva (diseño y funcionalidad) y sobre todo, debe tener un objetivo claro y fácil de cumplir. Todo esto procurando optimizar recursos.

En base a descrito, se puede identificar que los datos obtenidos, mediante el método seleccionado para evaluar el entorno virtual desarrollado CNA RV, pertenecen a la categoría de datos empíricos, esto quiere decir que fueron obtenidos mediante la experiencia del usuario. Este tipo de prueba se realiza al finalizar el desarrollo, cuando se obtiene la versión final de la aplicación para poder ratificar que se alcanzó el o los objetivo(s) planteados a través de los aspectos evaluados.

Todo producto software usable, como menciona Zumaquero, debe incorporar en su diseño atributos para beneficiar al usuario en un determinado contexto de uso. Lo cual no asegurara el correcto funcionamiento de la aplicación por sí sola, sino también se debe prestar especial atención en las estructuras implementadas para el desarrollo (arquitectura y metodología). Por otra parte, la técnica de evaluación utilizada va dirigida a medir cuan satisfactoria fue la experiencia de los usuarios al trabajar con el sistema desde diferentes perspectivas.

A continuación se detalla el propósito de las pruebas aplicadas, tipo de prueba, herramienta de medición, método, participantes, forma de recolección y análisis de la información obtenida. En base a esto, y después de una breve introducción y observación de la literatura que existe al respecto de este tipo de pruebas, se han definido los aspectos a evaluar que se presentan a continuación:

Tipo	Aspecto a Evaluar		Método
User	1. Información del Usuario	1.1. Escolaridad 1.2. Edad 1.3. Interacción con computadora (Hrs) 1.4. Actividad Predilecta 1.5. Utilización de App's	Cuestionario
Int	2. Estructura de la Aplicación	2.1. Organización Estructural 2.2. Densidad Estructural 2.3. Consistencia de la Estructura	Test
Int	3. Operación de la Aplicación	3.1. Navegabilidad 3.2. Interactividad 3.3. Accesibilidad 3.4. Sistema de Indicación 3.5. Desempeño del Sistema 3.6. Fiabilidad del Sistema 3.7. Consistencia de la Operación	Test
Int y RV	4. Información para el Usuario	4.1. Sistema de Ayuda 4.2. Retroalimentación 4.3. Búsqueda de Información	Test
Int y RV	5. Apariencia	5.1. La presentación del Contenido	Test
RV	6. Intuición y Nivel de Inmersión	6.1. Procedimiento de Navegación 6.2. Sensación Inmersiva	Test
Int	7. Contenido	7.1. Organización del Contenido 7.2. Densidad del Contenido 7.3. Fiabilidad de Contenido 7.4. Comprensión del Contenido	Test
Int y RV	8. Opinión General de la App	8.1. Calificación de opinión subjetiva del participante	Test

Tabla 5.5 Aspectos a Evaluar del Sistema CNA RV. (Elaboración Propia).

La columna llamada “Tipo” en la Tabla 5.5, clasifica los aspectos evaluados en el ámbito de *Interfaz (Int)* y *Realidad Virtual (RV)* y la técnica de comprobación que en ambos casos se aplicó fue mediante un Test. La información respecto al tipo Usuario (*User*), dado la naturaleza de los datos, se debe tratar con otra técnica o método, el cuál

será explicado de manera más detallada en *las especificaciones del cuestionario*. Los recursos Cuestionario y de Test, pertenecen a diferentes sistema de evaluación, aunque pueden parecer iguales, la información que recaban ambos formatos tiene diferentes enfoques. (Dra. Lilliam Perurena Cancio, 2013)

A continuación se muestran la categorización asociada a los aspectos a evaluar, estas categorías son:

- **Interfaz.** Elementos con los cuales interactúa el usuario en primera instancia. Es la conexión directa entre usuario-maquina, gracias a esto el usuario se mantiene comunicado con la aplicación. Su evaluación es a través de aspectos ergonómicos del panel principal de la aplicación. (Pereira, 2002)
- **Realidad Virtual.** El principal objetivo de la evaluación de los aspectos marcados en esta categoría, es medir el nivel de interacción y dinámicas fluidas que son posibles mediante la combinación de programación y animaciones. El entorno virtual debe tener la capacidad de crear la sensación de presencia al usuario participante, esto se refiere al nivel de integración a un mundo virtual aceptable en una aplicación de RV de escritorio. (Pereira, 2002)
- **Usuario:** Se busca tener información básica del usuario, de esta manera se puede medir factores que intervienen en la experiencia del usuario directamente con la aplicación. Es decir, factores como conocimiento previos de uso, experiencia con app's a fines, intereses, etc. influyen significativamente en la calidad de interacción con el sistema. (Dra. Lilliam Perurena Cancio, 2013)

Cada uno de los aspectos evaluados son explicados de manera detallada en el test aplicado, para que el usuario no tenga ninguna duda (material que puede ser consultado en el *Anexo A*).

5.4.2 JUSTIFICACIÓN DE LOS MÉTODOS IMPLEMENTADOS

Los sistemas de evaluación establecen la técnica específica que será implementada para evaluar la usabilidad de un sistema o servicio. Esto quiere decir que los métodos de evaluación también pueden ser clasificados por técnicas, de las cuales “métodos de inspección” y “métodos de indagación”, son la más utilizadas para medir la usabilidad. (Dra. Lilliam Perurena Cancio, 2013)

Estos métodos son considerados en la tabla de categorías expuesta por Zumaquero (2010), sin embargo, no se muestran las técnicas de evaluación a los que pertenecen.

Los métodos seleccionados se encargan de evaluar la experiencia de los participantes más que cualquier otro factor, ya que la finalidad de esta evaluación es medir ciertos aspectos de la experiencia del usuario con el sistema. Resumiendo, esta evaluación se realizó bajo las métricas de *eficacia*, *satisfacción* y *efectividad* definidas por los estándares de Usabilidad (ISO, 1998), que serán medibles de acuerdo a los métodos cuyas especificaciones serán aplicadas a través de las herramientas: cuestionario y test.

5.4.2.1 ESPECIFICACIONES DEL TEST

Dentro de las diversas metodologías de test que existen, Pereira (2002) cita en su estudio en el apartado de “Usabilidad”, a Rubin (1994), quien indica cuatro tipos de test predominantes, los cuales se encuentran asociados a diferentes fases del ciclo de vida de desarrollo de un producto software. Esto permite un reconocimiento mayor de la propuesta de cada test. Esta propuesta sugiere los siguientes tipos: (Pereira, 2002)

- Exploratorio.
- Evaluación de operaciones y aspectos del producto o servicio.
- Validación.
- Comparación.

En este contexto, el tipo de test implementado, arroja datos de *Validación*, factor ligado a la satisfacción del usuario. Este tipo de Test es aplicable a la versión final del sistema o servicio a evaluar y tiene las siguientes características:

Tipo de Test	Fases del ciclo de desarrollo de sistema			Objetivos	Implicaciones	Características Generales
	Inicio	Inter	Final			
Validación			✓	<p>Evalúa el sistema asegurando si satisface los requerimientos predeterminados de usabilidad. También, evalúa la interrelación entre los componentes del producto y su integración.</p>	<p>Existe el riesgo de implementación de un nuevo producto con problemas y consecuentemente la necesidad de retocarlo. Posibilidad de anticiparse a las posibles deficiencias del sistema.</p>	<p>Certificar la usabilidad del sistema. Aplicación de la versión avanzada. Considera los objetivos de usabilidad expresados en términos de criterios de desempeño del sistema. Con poca o nula interacción entre el “testeador” y el participante. Enfoque en la colecta de datos cuantitativos¹⁹.</p>

Tabla 5.6 Especificaciones del Test de Validación. (Adaptación de Pereira, 2002).

El método formal de un Test de Usabilidad según Pereira (2002),

“En general. Los Test de Usabilidad (i.e Test de aspectos ergonómicos) son los procedimientos de análisis aplicados a los usuarios destinos de un producto, en

¹⁹ Cuantitativo o cuantitativa es un adjetivo que hace referencia a la naturaleza numérica de datos, métodos, investigaciones y/o resultados. (Significados.com, 2017)

los cuales se verifica si dicho producto ha sido desarrollado de acuerdo con los requerimientos predeterminados de usabilidad.”

Métrica	Reactivo	Características
Sat	a) Organización estructural	La distribución de los elementos estructurales de la aplicación es buena (barras de desplazamiento, zonas de contenido, botones, etc.).
Sat, Efi	b) Densidad estructural	La cantidad de elementos estructurales utilizados en la aplicación es excesiva.
Sat, Efi, Efe	c) Consistencia de la estructura	La distribución de los elementos estructurales se mantiene constante a lo largo de la aplicación.
Sat, Efe	d) Navegabilidad	El recorrido establecido por la aplicación es fácil de seguir.
Sat	e) Interactividad	La relación mutua entre el usuario y la aplicación es buena.
Sat, Efi, Efe	f) Accesibilidad	Las acciones que solicita la aplicación son fáciles de ejecutar.
Sat, Efe	g) Sistema de indicación	Se identifican fácilmente las figuras, las tablas, los hipertextos, las zonas activas y el tipo de acción que se debe ejecutar.
Sat, Efi, Efe	h) Desempeño del sistema	La velocidad de funcionamiento de la aplicación, tomando en cuenta el tipo de tarea que exige, es buena.
Sat, Efi	i) Fiabilidad del sistema	Hay demasiados errores durante la operación de la aplicación.
Efi, Efe	j) Consistencia de la operación	La ejecución de tareas (navegar por la app, hacer clic en botones, seleccionar opciones) sigue un estándar a lo largo de la aplicación.
Sat, Efi	k) Sistema de ayuda	Las dudas del usuario se resuelven fácilmente.
Sat, Efe	l) Retroalimentación	La aplicación mantiene al usuario informado sobre las tareas a ejecutar.
Sat, Efe	m) Búsqueda de información	Los datos que busca el usuario son fáciles de encontrar.
Sat, Efe	n) La presentación del contenido	Tipo y tamaño de fuente, uso del color, disposición de los elementos según su significado es buena.
Sat, Efe	o) Intuición	Los procedimientos de navegación por la aplicación o ejecución de tareas asignadas se aprenden de forma prácticamente inmediata.
Sat, Efe	p) Nivel de inmersión	La sensación de presencia dentro del entorno virtual es buena.
Sat, Efe	q) Organización del contenido	La distribución del contenido de la aplicación es buena (textos, botones, imágenes).
Sat	r) Densidad del contenido	La información que se presenta en la aplicación es demasiado extensa.

Sat, Efi	s) Fiabilidad del contenido	No hay errores en la información que se presenta.
Sat	t) Comprensión del contenido	La información presentada es fácil de entender.
Sat, Efi, Efe	u) Opinión General	Calificación general de la aplicación.

Tabla 5.7 Reactivos del Test propicios para el presente sistema “CNA RV”. (Elaboración propia).

Efectivamente, como se menciona en la *Tabla 5.6*, este tipo de test se realiza a partir de la versión final del sistema, de esta manera las estimaciones generadas en situaciones reales de usabilidad son más precisas. Así mismo, la primera columna indica también las métricas que están siendo contempladas en el desarrollo de los reactivos del test.

5.4.2.2 ESPECIFICACIONES DEL CUESTIONARIO

Esta es una de las formas de evaluación de *Usabilidad* más utilizada. Principalmente se debe a la facilidad de aplicación, se basa en una colección de preguntas que deben responderse entre determinadas posibles respuestas. Generalmente su procesamiento es mediante estudios estadísticos. (Dra. Lilliam Perurena Cancio, 2013)

Se tomó la decisión de manejar la información obtenida, en base a su naturaleza, a manera de cuestionario ya que la medición de los datos arrojados precisa de diferente técnica. El objetivo del cuestionario aplicado (Pereira, 2002), es recabar información personal del usuario tales como gustos de contenido, experiencia con el dispositivo y app similares (*Tabla 5.5*).

Característica	Pregunta
a) Escolaridad	Nivel educativo actual.
b) Horas dedicadas a utilizar la computadora	Opciones definidas por un rango establecido (1-8 o más).
c) Actividad prevalente	Opciones disponibles: trabajar, estudiar, entretenimiento o no utilizo.
d) Tipo de software utilizado en los últimos 5 meses	El usuario puede elegir la categoría que más se ajuste a su experiencia, entre: nula, principiante, intermedio y avanzado.

Tabla 5.8 Reactivos del Cuestionario. (Elaboración propia).

Este cuestionario no ahonda demasiado en la información del usuario ya que los datos recopilados están orientados a la agrupación de los participantes a partir de determinadas características. Estos datos son denominados como cualitativos, pues los reactivos se centran en la calidad de un producto. A diferencia del cuestionario, el Test se mide mediante una calificación que evalúa diversas propiedades (*Tabla 5.5*). (Definiciones, 2017)

Los cuestionarios manejan un prejuicio objetivo y subjetivo, esto representa una ventaja, por su propiedad subjetiva, no obstante gran parte de las muestras conllevan una visión objetiva de la aplicación. (Gómez, 2006)

5.4.3 CONDICIONES DE APLICACIÓN

Las condiciones de aplicación de un test de usabilidad, es un procedimiento muy importante que en altos niveles requiere de gran inversión y organización, pues generalmente en el desarrollo de un software con las características presentes participa un equipo de varios profesionistas con diferente labor (p. ej. directivos, desarrolladores, programadores, editores, diseñadores, testadores, etc.), dadas la condiciones de desarrollo del sistema “CNA RV”, la prueba fue efectuada por la autora del sistema. (Pereira, 2002)

El desarrollo de la industria y de la propia academia, han fomentado el desarrollo de investigaciones que derivan en nuevos planteamientos para mejorar la calidad de procesos y/o productos mediante la aplicación de *test de usabilidad*, el cual será aplicado para evaluar el presente sistema. En este contexto, la configuración de los “laboratorios de usabilidad” es de gran importancia para lograr la consolidación de la aplicación del *test de usabilidad*. (Pereira, 2002). Dado las circunstancias y limitaciones presentadas para evaluar la presente aplicación mediante un *test de usabilidad*, se concluyó que cumple con la siguiente configuración del entorno físico para su realización. Configuración propuesta por Rubin (1994).

- **Laboratorio móvil.** El laboratorio móvil consiste en una configuración alternativa a las configuraciones físicas usuales, ya que este carece de un espacio físico predeterminado para la aplicación de la prueba, lo cual describe perfectamente el escenario que se presentó para la aplicación de los métodos seleccionados de evaluación del sistema *CNA RV*. De esta manera, se utilizará la infraestructura tecnología como mayor soporte para realizar las pruebas, las herramientas utilizadas son: lentes especiales VR y computadora portátil; esto instrumentos se eligieron para poder sobresaltar las características propia de una interfaz realizada en 2D y un modelo 3D.

Las ventajas que se encontraron con estas condiciones de aplicación fueron:

- **Costos.** Este tipo de configuración no demanda demasiada inversión, por lo cual la relación coste-eficiencia resulta óptima.
- **Portabilidad.** Tal como menciona Pereira, en el análisis de este tipo de laboratorio, permite el contacto directo del laboratorio con el participante y por consecuente con el Testeador. Lo cual, evidentemente, se logra gracias a una infraestructura tecnológica móvil.
- **Tiempo.** El tiempo significo una desventaja en el momento de la aplicación del test, por lo cual facilito y disminuyo de manera considerable este recurso, gracias a la facilidad que implico montar todo lo necesario para la evaluación.

El espacio requerido y las herramientas utilizadas para implementar este tipo de laboratorio se puede ilustrar de la siguiente manera:

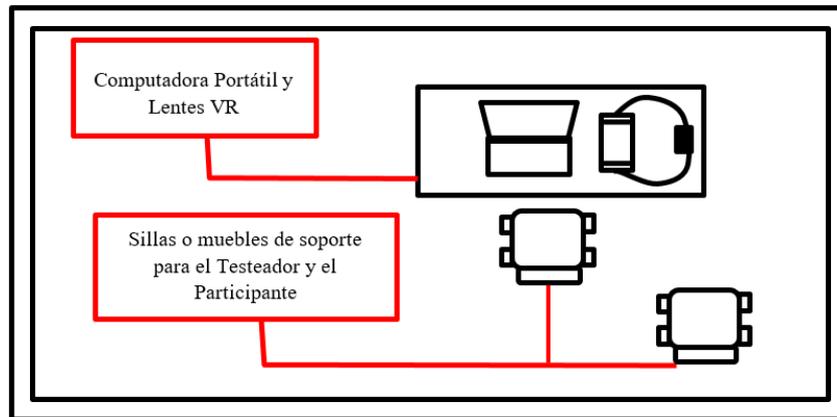


Fig. 5.2 Descripción gráfica de configuración del entorno físico de aplicación. (Elaboración propia).

Computadora portátil: Cuenta con las especificaciones definidas en la arquitectura del sistema, y cuenta con la aplicación debidamente instalada.

Lente VR: La utilización de los lentes son un elemento extra en cada evaluación, ya que los lentes solo están equipado con un video que muestra el funcionamiento dentro del entorno 3D. Por lo cual, se les indico a los participantes calificar de acuerdo a su experiencia utilizando la computadora portátil.



Fig. 5.3 Dispositivos implementados en la aplicación del Test de Usabilidad.

5.4.4 PARTICIPANTES

A manera de recordatorio, el sistema de evaluación elegido (método(s) de evaluación), fue aplicado a una población conformada por 32 participantes (8 nivel primaria, 8 nivel secundaria, 8 nivel bachillerato y 8 nivel universidad), tal como se planteó al principio y bajo las recomendaciones de Nielsen (1993), Rubin (1994), y Mayhew (1999).

Los participantes son personas que usan cotidianamente el internet como herramienta de trabajo, por lo tanto cuentan con los conocimientos suficientes para utilizar los dispositivos necesarias para probar la aplicación, también, se habla de un rango de edad que oscila entre los 8-27 años, con nivel escolar mínimo de primaria. Estos datos pueden ser constatados gracias al cuestionario de usuario aplicado. La ficha técnica correspondiente se muestra a continuación (Tabla 5.9).

Nombre de Encuesta	Test de Evaluación de Usabilidad para Sistemas en Realidad Virtual - Aplicación CNA-RV
Universo	Población General
No. De Participantes	32
Fecha de inicio	25/09/17
Fecha de finalización	29/09/17

Tabla 5.9 Ficha de Detalles Técnicos de Muestreo. (Elaboración propia).

Esta muestra fue dividida en 4 grupos, para hacer más evidente la brecha generacional y los conocimientos previos de los participantes, de esta manera poder relacionar la calidad de la valoración del software con estas características.

Grupo 1: Primaria	8-11
Grupo 2: Secundaria	12-15
Grupo 3: Bachillerato	15-21
Grupo 4: Universidad	20-26

Tabla 5.10 Grupos creados a partir de la muestra general. (Elaboración Propia).

La población encuestada puede o no contar con una computadora en casa, sin embargo utiliza esta tecnología constantemente. Datos de la encuesta más recientemente realizada “Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en los Hogares” (ENDUTIH), presenta una relación entre el nivel académico y el uso de tecnologías de la información (TIC’s). También, cabe mencionar que el Estado de México (entidad de origen de esta prueba) se encuentra en el 1er puesto entre los estados con más usuarios de computadora del país, por su parte Colima ocupa el último puesto. Otro dato significativo que vale la pena contemplar respecto al uso de nuevas tecnologías, es el uso de internet, donde los datos obtenidos muestran el consumo de esta tecnología proporcional con la edad y escolaridad de la población, ya que entre mayor edad tiene el individuo menor es el uso de esta tecnología, así mismo, mientras el nivel escolar avanza el consumo aumenta. La edad que impera entre los usuarios de las TIC’s se encuentra en un rango de 12-34 años. Este dato es importante a futuro, ya que si la aceptación del sistema es exitosa, se puede contemplar una disponibilidad online, es decir, que pueda ser vista desde la página oficial del inmueble. (INEGI, 2016)

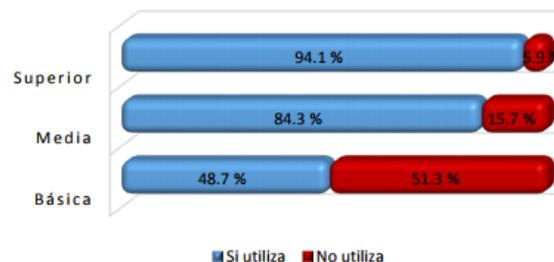


Fig. 5.4 Gráfica acerca del uso de internet respecto al nivel escolar. (INEGI, 2016).

Tal como se mencionó, una especificación fundamental para la utilización del app es contar con una computadora, ya que se trata de un sistema de escritorio. En este sentido, los usuarios de computadora, según ENDUTIH, clasificados por nivel de escolaridad son representados por las siguientes cifras (*Fig. 5.5*).

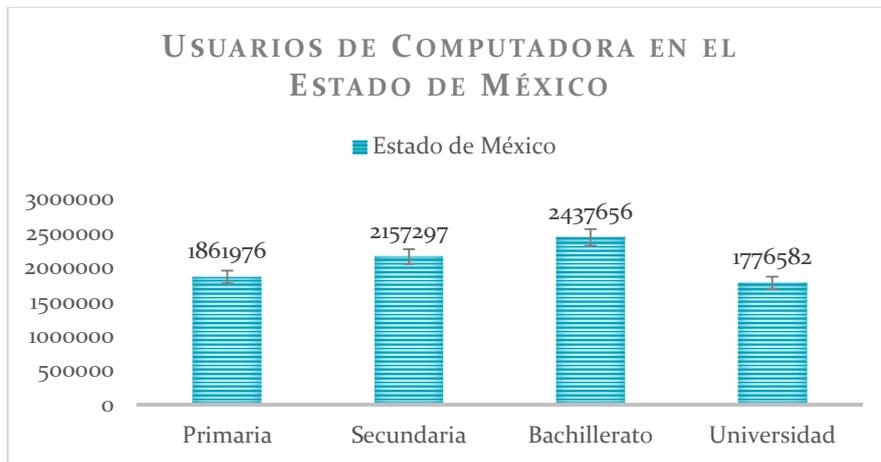


Fig. 5.5 Usuario de computadora en el Edo. De México, por nivel escolar. (INEGI, 2016)

Se puede observar un comportamiento continuo, donde el crecimiento es exponencial en los niveles que corresponden a la educación básica y medio superior en México, sin embargo, los números sufren un decremento considerable con los usuarios que se encuentran en nivel profesional, ergo, los usuarios de nivel superior prefieren utilizar otro tipo de dispositivos electrónicos.

En la *Fig. 5.6* se muestran los usuarios de computadora en los estados circundantes a la CDMX, ya que es aquí donde se encuentra el inmueble en el cual fue basado el modelo 3D del presente sistema. Entre estos estados se encuentran: Morelos, Tlaxcala, Hidalgo y Puebla, los cuales resultan significativos pues mucha de la población podría estar interesada en las visitas virtuales a recintos culturales debido a su relativa cercanía con estos, para conocer el inmueble previo a una visita presencial. La potencial población consumidora corresponde a los siguientes datos.

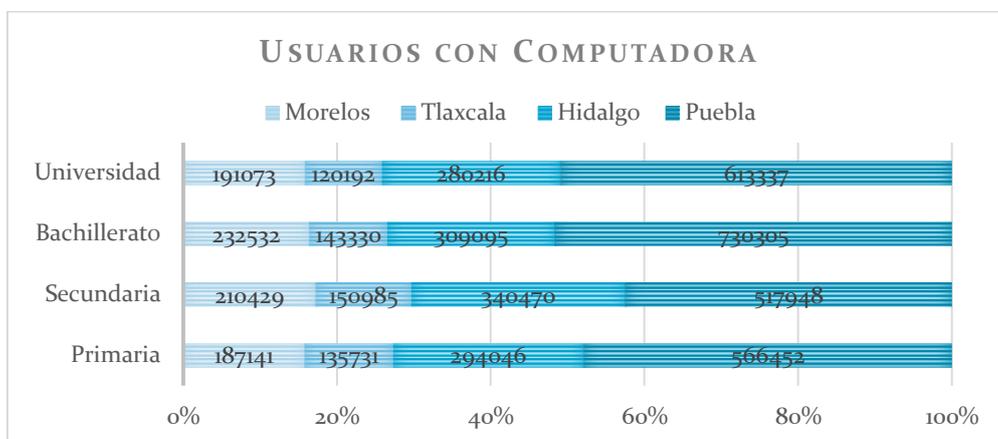


Fig. 5.6 Usuarios de computadora por entidad federativa seleccionada, clasificados por nivel de escolaridad. (INEGI, 2015)

5.4.5 PERFIL DE LOS PARTICIPANTES

El *cuestionario*, como se mencionó en las especificaciones, tiene como objetivo recabar información notable acerca del participante. Entre esta información se ha recuperado los datos necesarios para clasificar a los participantes entre los siguientes perfiles en base a su experiencia con aplicaciones relacionadas al ámbito de desarrollo del presente proyecto. (Vega, 2010)

Perfil	Sexo	Experiencia en Entornos Virtuales o Recorridos 3D			
		Primaria	Secundaria	Bachillerato	Universidad
Exp. Nula	Femenino	no	no	no	juegos
	Masculino	no	no	no	no
Principiante	Femenino	no	juegos	no	no
				juegos	
	Masculino	no	juegos	juegos	juegos
Intermedio	Femenino	Aplicaciones educativas	juegos	juegos online	recorrido museo 3d
		no	no	no	no
	Masculino	Aplicaciones educativas	juegos	juegos	juegos
		no			no
Avanzado	Femenino	no	no	no	no
	Masculino	no	juegos	no	no
no					

Tabla 5.11 Perfiles de Participantes basados en la experiencia con app. (Elaboración Propia).

Tal como se mencionó, los datos fueron obtenidos antes de la aplicación del *cuestionario* y *test de usabilidad*, utilizando el método pensando alto (*thinking aloud*) que es parte del mismo sistema de evaluación que el test, consiste en solicitar al participante expresar sus comentarios generales o específicos (según se requiera) respecto al sistema por parte del testeador. Por este motivo, se decidió tomar en cuenta este factor para tener una idea de cuán involucrados están los participantes con aplicaciones desarrolladas con tecnologías como realidad virtual y/o herramientas multimedia. (Dra. Lilliam Perurena Cancio, 2013)

5.5 RESULTADOS OBTENIDOS

Dentro de las herramientas software utilizadas para promover el conocimiento cultural entre la población de México, existen sobretodo recorridos virtuales que pretender preservar recintos prehispánicos para enriquecer la cultura general y llegar a más personas en diferentes zonas del país, incluso del mundo. Por este motivo, es prioridad ofrecer un producto de buena calidad que represente una aportación relevante. En este sentido los resultados obtenidos por la evaluación de usabilidad del presente proyecto, se presentan y analizan mediante cálculos probabilísticos, que proponen un panorama más claro acerca de la calidad de la aplicación desarrollada.

5.5.1 ANÁLISIS DE DATOS: CUESTIONARIO

Dentro de los datos que aportó la aplicación del cuestionario a los participantes, existe una sección que recolecta su experiencia en el uso de diferentes tipos de software utilizando una computadora de escritorio.

El *cuestionario* es resuelto por los participantes antes de realizar el *test* cuyos reactivos son completamente destinados para calificar la aplicación desarrollada. La *Tabla 5.12* muestra los resultados en porcentaje de la experiencia con la que cuentan los participantes, donde 15 individuos son de sexo femenino y 17 de sexo masculino. Se contemplaron tipos de software's relevantes de la sección del cuestionario que recaba la experiencia del usuario con diferentes TIC's (tomando en cuenta sus comentarios presentados en la *Tabla 5.11*), a propósito de las habilidades cognitivas previas que ayudan al usuario a llevar a cabo una mejor participación en la prueba. *Experiencia en App Multimedia, S.O. Windows y Juegos*, fueron las elegidas para representar la experiencia del usuario en los perfiles planteados anteriormente (*Tabla 5.11*).

Perfil	Sexo	Experiencia en App Multimedia				Sistema Operativo Windows				Juegos				Prom por Sexo	Prom por Perfil
		Prim	Sec	Bach	Uni	Prim	Sec	Bach	Uni	Prim	Sec	Bach	Uni		
Exp. Nula	Femenino	0.0	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.041	0.070
	Masculino	0.4	0.0	0.0	0.0	0.4	0.0	0.0	0.0	0.4	0.0	0.0	0.0	0.1	
Principiante	Femenino	0.75	0.0	0.25	0.0	0.5	0.0	0.25	0.0	0.5	0.25	0.25	0.0	0.229	0.204
	Masculino	0.4	0.25	0.25	0.0	0.4	0.0	0.00	0.2	0.2	0.25	0.0	0.2	0.179	
Intermedio	Femenino	0.0	0.75	0.5	0.33	0.0	0.5	0.25	0.33	0.0	0.5	0.5	0.33	0.332	0.343
	Masculino	0.2	0.75	0.25	0.6	0.2	0.25	0.25	0.2	0.4	0.25	0.5	0.4	0.354	
Avanzado	Femenino	0.25	0.25	0.25	0.66	0.0	0.5	0.5	0.66	0.0	0.25	0.25	0.66	0.352	0.359
	Masculino	0.0	0.0	0.5	0.4	0.0	0.75	0.75	0.6	0.0	0.5	0.5	0.4	0.366	
Promedio por Nivel de Escolaridad		0.25	0.25	0.18 7	0.24 8	0.25	0.25	0.25	0.24 8	0.25	0.25	0.25	0.24 8		
Promedio Total		0.25				0.25				0.229				0.248	

Tabla 5.12 Perfil de los participantes de prueba. Elaboración propia.

En la *Fig. 5.7*, se muestra una gráfica que permite apreciar mejor los resultados entre los perfiles de los participantes, resultados que muestran una relación en el nivel de preparación que tienen para interactuar con aplicaciones de esta índole y la calidad de su evaluación derivada del test.

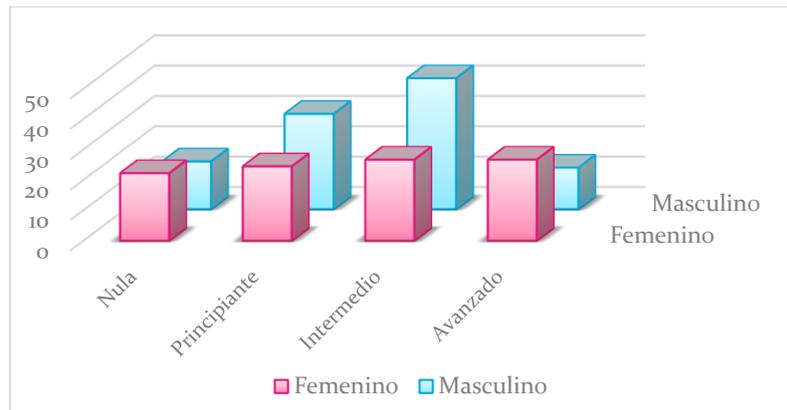


Fig. 5.7 Promedio de los porcentajes obtenidos por nivel escolar y sexo. (Elaboración propia).

Retomando, los datos presentados (Tabla 5.12) muestran y confirman (Fig. 5.4) una relación creciente entre el nivel de escolaridad de los participantes con los perfiles sugeridos ergo su experiencia con software’s específicos. Perfiles en los que ellos mismos se catalogaron mediante los comentarios realizados previamente a la prueba (Tabla 5.11), tal información posteriormente se infiere gracias a los resultados obtenidos y presentados.

Lo siguientes datos muestran información específica acerca de los hábitos de los involucrados respecto al tiempo que invierten en el uso del dispositivo electrónico de interés para el presente estudio, el cual es la computadora, pues es mediante esta que el usuario podrá interactuar con el sistema desarrollado en la presente. Es importante mencionar que en la sección de “Actividad en la Computadora” se le permitió al participante elegir una o más opciones disponibles, de manera que se identifique lo más posibles con su realidad (Tabla 5.13).

Utilización de Computadora (Hrs)	Primaria	Secundaria	Bachillerato	Universidad
Menos 1	1	1	2	0
Entre 1-2	3	1	3	1
Entre 2-4	3	4	2	0
Entre 4-8	0	2	0	6
Más de 8	0	0	0	1
No utilizo	1	0	1	0
Actividad en la Computadora	Primaria	Secundaria	Bachillerato	Universidad
Trabajar	0	1	1	8
Estudiar	7	6	2	8
Entretenimiento	5	6	4	5
No utilizo	2	1	1	0

Tabla 5.13 Datos Obtenidos mediante la Técnica de Cuestionario. Elaboración Propia.

En tal contexto, se puede deducir que entre el nivel de escolaridad avanza mayor es el número de participantes con la experiencia necesaria para la utilización de aplicaciones con contenido multimedia y tecnologías relacionadas, en este caso, RV.

5.5.2 ANÁLISIS DE DATOS: TEST DE VALIDACIÓN

Los resultados obtenidos a través del test que está enfocado en la evaluación del sistema a partir de las métricas definidas (interfaz, realidad virtual, usuario). La siguiente tabla muestra los datos de evaluación emitidos por los participantes de la presente prueba de usabilidad, a partir de diversos reactivos especificados para corroborar el correcto funcionamiento de un sistema con RV. Se presentan cuatro tablas señalando los niveles de escolaridad contemplados que manifiesta una correlación con diferentes perfiles pronunciados (*Tabla 5.12*).

El sistema de calificación está representado por una línea con valores correspondientes a una valoración del reactivo dentro de una escala del 1-10, en la cual el participante elige subjetivamente (percepción personal) el valor que represente de manera más precisa su experiencia con el sistema.

En Desacuerdo	2	4	6	8	10	De acuerdo
---------------	---	---	---	---	----	------------

Nivel de Escolaridad Primaria – Test de Validación												
Tipo	Aspecto a Evaluar		Calificaciones Asignadas al Sistema CNA RV								Prom. General	
			P1	P2	P3	P4	P5	P6	P7	P8		
Int	Estructura de la Aplicación	Organización Estructural	8	10	10	10	10	10	10	10	10	9.75
		Densidad Estructural	4	8	8	10	8	6	4	10	10	7.25
		Consistencia de la Estructura	6	10	10	10	8	10	10	10	10	9.25
	Operación de la Aplicación	Navegabilidad	2	10	10	10	8	10	10	10	10	8.75
		Interactividad	4	10	10	10	8	10	10	10	10	9.00
		Accesibilidad	6	8	10	10	8	10	10	10	10	9.00
		Sistema de Indicación	4	8	10	10	10	10	10	10	10	9.00
		Desempeño del Sistema	8	8	10	10	10	8	10	10	10	9.25
		Fiabilidad del Sistema	2	6	4	8	6	8	6	8	8	6.00
		Consistencia de la Operación	4	8	10	10	10	10	10	10	10	9.00
Int y RV	Información para el Usuario	Sistema de Ayuda	8	10	10	10	10	10	10	8	8	9.50
		Retroalimentación	6	8	10	10	10	10	10	10	10	9.25
		Búsqueda de Información	2	8	10	10	10	10	10	8	8	8.50
Int y RV	Apariencia	Presentación del Contenido	6	10	10	10	10	10	8	8	9.00	
RV	Intuición y Nivel de Inmersión	Procedimiento de Navegación	4	10	10	10	10	10	10	10	10	9.25
		Sensación Inmersiva	6	10	10	10	10	6	10	10	10	9.00
Int	Contenido	Organización	8	10	10	10	10	10	8	10	10	9.50
		Densidad	4	10	4	10	10	10	2	10	10	7.50
		Fiabilidad	2	10	10	10	10	4	10	10	10	8.25
		Comprensión	4	6	10	10	8	10	10	10	10	8.50
Int y RV	Opinión General de la App	Calificación subjetiva del participante	4	8	10	10	8	8	10	10	8.50	
Promedio Total de Test											8.71	

Tabla 5.14 Calificaciones y promedio de la evaluación obtenida mediante el Test de Usabilidad (Nivel Primaria). Elaboración Propia.

Tal como se mencionó anteriormente, los resultados de las *Tablas 5.14, 15, 16 y 17* presentan la calificación asignada de los participantes que intervinieron en la prueba respecto a su propia experiencia. Para conocer la media de estas calificaciones, se determinó el promedio por cada reactivo evaluado, esto permite conocer la proximidad a un valor esperado, en este caso, la mayor valoración corresponde al 100%.

Nivel de Escolaridad Secundaria – Test de Validación											
Tipo	Aspecto a Evaluar		Calificaciones Asignadas al Sistema CNA RV								Prom. General
			P1	P2	P3	P4	P5	P6	P7	P8	
Int	Estructura de la Aplicación	Organización Estructural	6	10	10	10	10	10	10	10	9.50
		Densidad Estructural	4	10	10	8	10	10	10	4	8.25
		Consistencia de la Estructura	10	10	8	10	10	10	10	10	9.75
	Operación de la Aplicación	Navegabilidad	10	8	8	10	8	10	10	8	9.00
		Interactividad	4	8	8	10	10	10	10	8	8.50
		Accesibilidad	8	10	6	10	8	10	10	10	9.00
		Sistema de Indicación	8	10	4	10	8	10	10	10	8.75
		Desempeño del Sistema	8	8	4	10	8	10	10	6	8.00
Fiabilidad del Sistema	2	10	6	8	4	10	6	8	6.75		
Consistencia de la Operación	10	10	6	10	10	4	10	10	8.75		
Int y RV	Información para el Usuario	Sistema de Ayuda	10	8	8	10	8	8	10	10	9.00
		Retroalimentación	8	10	6	8	10	10	10	8	8.75
		Búsqueda de Información	8	10	8	8	10	8	10	8	8.75
Int y RV	Apariencia	Presentación del Contenido	6	10	6	10	10	10	10	9.00	
RV	Intuición y Nivel de Inmersión	Procedimiento de Navegación	8	8	2	10	10	10	10	10	8.50
		Sensación Inmersiva	4	10	4	10	10	10	8	10	8.25
Int	Contenido	Organización	6	10	8	10	10	10	10	10	8.00
		Densidad	8	8	4	6	6	10	10	2	6.75
		Fiabilidad	10	10	8	10	10	6	6	8	8.50
		Comprensión	8	10	8	10	10	10	10	10	9.50
Int y RV	Opinión General de la App	Calificación subjetiva del participante	8	10	10	10	10	10	10	10	9.75
Promedio Total de Test										8.61	

Tabla 5.15 Calificaciones y promedio de la evaluación obtenida mediante el Test de Usabilidad (Nivel Secundaria). Elaboración Propia.

Posteriormente a determinar el promedio de calificaciones por cada aspecto que se evaluó, se calculó el promedio total que evalúa el sistema completo por cada nivel escolarizado, esto con la finalidad de conocer en términos generales la aceptación del sistema mediante los criterios anteriormente especificados (efectividad, eficiencia y satisfacción).

Un dato importante que se ha manifestado hasta el momento reside en las calificaciones emitidas por los participantes de nivel primaria y secundaria, pues tomando en cuenta la relación establecida previamente entre perfiles y nivel de escolaridad, se pueden observar mejores calificaciones por parte del grupo de participantes con menor nivel de experiencia.

En este contexto, los datos establecen un decremento en la evaluación del sistema por parte de los participantes proporcional al avance del nivel escolar. Situación prevista, tal como se mencionó anteriormente, desde la evaluación de los niveles primaria y secundaria, posteriormente confirmada por la evaluación de los participantes del nivel bachillerato (*Tabla 5.16*). Esto nos lleva a la conclusión de que los participantes con mayor experiencia encuentran deficiencias en el sistema con mayor rigurosidad que aquellos que demostraron tener menos nociones de la utilización de este tipo de aplicaciones.

Por otro lado, se debe tener presente que el público al que va dirigido el sistema es general, por lo cual puede que a ciertos usuarios les motive más interés temas específicos de otro ámbito. Sin embargo, este punto se va aunado a otros factores intrínsecos de cada participante pues los datos son analizados dentro de la subjetividad que proporciona la percepción de usuarios reales que funcionan como participantes para esta prueba. Lo cual también representa una ventaja, ya que el sistema se está aplicando en una muestra de la población real a la que va dirigido.

Nivel de Escolaridad Bachillerato – Test de Validación											
Tipo	Aspecto a Evaluar		Calificaciones Asignadas al Sistema CNA RV								Prom. General
			P1	P2	P3	P4	P5	P6	P7	P8	
Int	Estructura de la Aplicación	Organización Estructural	8	8	8	8	8	10	8	10	8.50
		Densidad Estructural	8	2	2	8	4	2	6	6	4.75
		Consistencia de la Estructura	8	6	8	8	6	10	8	8	7.75
	Operación de la Aplicación	Navegabilidad	8	8	10	10	8	2	10	2	7.25
		Interactividad	8	10	10	10	8	10	10	10	9.50
		Accesibilidad	10	8	10	10	8	10	10	4	8.75
		Sistema de Indicación	10	2	8	10	10	10	8	6	8.00
		Desempeño del Sistema	10	6	8	10	8	10	10	10	9.00
		Fiabilidad del Sistema	2	2	6	8	8	2	10	8	5.75
		Consistencia de la Operación	6	8	10	8	8	10	10	6	8.25
Int y RV	Información para el Usuario	Sistema de Ayuda	8	8	10	8	6	10	8	8	8.25
		Retroalimentación	8	8	10	8	8	10	10	10	9.00
		Búsqueda de Información	8	10	8	10	8	10	10	6	8.75
Int y RV	Apariencia	Presentación del Contenido	2	8	10	10	8	10	8	4	7.50
RV	Intuición y Nivel de Inmersión	Procedimiento de Navegación	8	4	8	8	6	10	10	8	7.75
		Sensación Inmersiva	8	6	10	10	8	10	10	2	8.00
Int	Contenido	Organización	10	8	6	10	8	10	10	6	8.50
		Densidad	10	2	10	6	8	6	8	8	7.25
		Fiabilidad	10	10	8	6	2	10	10	8	8.00
		Comprensión	10	10	6	8	8	10	8	10	8.75
Int y RV	Opinión General de la App	Calificación subjetiva del participante	10	10	10	10	8	10	10	8	9.50
Promedio Total de Test										8.03	

Tabla 5.16 Calificaciones y promedio de la evaluación obtenida mediante el Test de Usabilidad (Nivel Bachillerato). Elaboración Propia.

En este punto, la observación descrita previamente por las calificaciones de los participantes de los niveles de primaria, secundaria y bachillerato que presentaba una decreciente valoración, da un giro relativamente drástico en la evaluación del sistema. Los datos derivados de la evaluación a nivel universidad (*Tabla 5.17*), muestran una mejoría de calificaciones, ergo, del promedio general.

De esta manera, es posible deducir que pese a que existe, de manera inicial, una caída exponencial en la valoración del sistema en correlación con los perfiles de los participantes, la mejor valoración, propia del perfil de “avanzado”, corresponde al nivel universitario, por lo cual la evaluación emitida por este grupo de participantes se interpreta como la más objetiva, puesto a que tienen los conocimientos básicos sobre el manejo de aplicaciones de la misma índole. Cabe destacar que las calificaciones dadas en los reactivos de manera individual, en varias ocasiones, indistintamente del nivel de escolaridad, no se observaba congruencia alguna con la calificación asignada al sistema de manera general, respectivamente.

Nivel de Escolaridad Universidad – Test de Validación											
Tipo	Aspecto a Evaluar		Calificaciones Asignadas al Sistema CNA RV								Prom. General
			P1	P2	P3	P4	P5	P6	P7	P8	
Int	Estructura de la Aplicación	Organización Estructural	8	10	8	8	10	10	8	10	9.00
		Densidad Estructural	6	10	8	8	10	8	10	8	8.50
		Consistencia de la Estructura	8	10	8	8	10	10	8	8	8.75
	Operación de la Aplicación	Navegabilidad	6	10	6	10	10	10	10	8	8.75
		Interactividad	8	10	6	10	10	10	10	8	9.00
		Accesibilidad	8	10	8	10	10	10	8	10	9.25
		Sistema de Indicación	6	10	8	10	10	10	8	10	9.00
		Desempeño del Sistema	8	10	8	10	10	10	6	10	9.00
Fiabilidad del Sistema	6	4	4	4	6	2	6	6	4.75		
Consistencia de la Operación	10	10	8	8	10	8	8	10	9.00		
Int y RV	Información para el Usuario	Sistema de Ayuda	6	10	8	8	10	10	10	10	9.00
		Retroalimentación	6	10	8	10	10	10	10	10	9.25
		Búsqueda de Información	10	10	8	10	10	10	10	10	9.75
Int y RV	Apariencia	Presentación del Contenido	6	10	10	10	10	8	10	10	9.25
RV	Intuición y Nivel de Inmersión	Procedimiento de Navegación	8	8	6	8	10	10	10	6	8.25
		Sensación Inmersiva	8	10	8	10	10	10	10	10	9.50
Int	Contenido	Organización	6	10	8	10	10	10	10	10	9.25
		Densidad	8	4	6	10	4	8	10	10	7.50
		Fiabilidad	10	8	8	10	10	10	10	10	9.50
		Comprensión	10	8	10	10	10	10	10	10	9.75
Int y RV	Opinión General de la App	Calificación subjetiva del participante	10	10	8	8	10	10	10	10	9.50
Promedio Total de Test										8.83	

Tabla 5.17 Calificaciones y promedio de la evaluación obtenida mediante el Test de Usabilidad (Nivel Universidad). Elaboración Propia.

A continuación se muestran los promedios generales, respecto a su nivel escolar y por ende a su perfil propuesto mediante el cuestionario inicial (*Tabla 5.18*), esto de manera general para poder apreciar mejor el contraste de evaluación entre las categorías de participantes. La presente información recabada permite reconocer importantes puntos dentro del test de validación, mediante conclusiones precisas, tales como

- El aspecto evaluado que generó mayor promedio fue el correspondiente a: **información para el usuario**. Lo cual es importante, ya que el agente se implementó para proveer al usuario de la información necesaria para cumplir con el objetivo.
- Los reactivos mejor valorados: **organización estructura, interactividad, retroalimentación, presentación del contenido, sensación inmersiva, comprensión**. Se ajustan perfectamente a los ideales del desarrollo de la aplicación, en base a los objetivos planteados. Sin embargo, el porcentaje obtenido en promedio dista de ser el óptimo.
- Los reactivos peor valorados: **consistencia de la estructura, desempeño del sistema, búsqueda de la información, procedimiento de navegación, fiabilidad**. Estos reactivos valorados de manera deficiente son los puntos que se necesitan mejorar y también implican una recaída importante en la evaluación.

De esta manera, partiendo de la comparación de los promedios generados por cada nivel de escolaridad implicado en la prueba, se calculó el **promedio global** del sistema, el cual quedó definido por el porcentaje de: **85%**, en una escala del 1 al 100.

Resultados Generales Test de Validación									
Tipo	Aspecto a Evaluar		Calificaciones Asignadas al Sistema CNA RV				Prom. General	Desv. Estándar	
			Prim	Sec	Bach	Uni			
Int	Estructura de la Aplicación	Organización Estructural	9.75	9.50	8.50	9.00	9.18	0.231	
		Densidad Estructural	7.25	8.25	4.75	8.50	7.18	2.199	
		Consistencia de la Estructura	9.25	9.75	7.75	8.75	8.87	0.547	
	Promedio por aspecto evaluado			8.91	9.16	7.00	8.75	8.41	0.729
	Operación de la Aplicación	Navegabilidad	8.75	9.00	7.25	8.75	8.43	0.481	
		Interactividad	9.00	8.50	9.50	9.00	9.00	0.125	
		Accesibilidad	9.00	9.00	8.75	9.25	9.00	0.031	
		Sistema de Indicación	9.00	8.75	8.00	9.00	d	0.168	
		Desempeño del Sistema	9.25	8.00	9.00	9.00	4.40	19.701	
	Fiabilidad del Sistema	6.00	6.75	5.75	4.75	5.81	0.512		
Consistencia de la Operación	9.00	8.75	8.25	9.00	8.75	0.094			
Promedio por aspecto evaluado			8.57	7.53	8.07	8.39	7.72	0.333	
Int y RV	Información para el Usuario	Sistema de Ayuda	9.50	9.00	8.25	9.00	8.93	0.199	
		Retroalimentación	9.25	8.75	9.00	9.25	9.06	0.043	
		Búsqueda de Información	8.50	8.75	8.75	9.75	8.93	0.231	
Promedio por aspecto evaluado			9.08	8.83	8.66	9.33	8.97	0.064	
Int y RV	Apariencia	Presentación del Contenido	9.00	9.00	7.50	9.25	8.68	0.481	
		Promedio por aspecto evaluado			9.00	9.00	7.50	9.25	8.68

RV	Intuición y Nivel de Inmersión	Procedimiento de Navegación	9.25	8.50	7.75	8.25	8.43	0.293
		Sensación Inmersiva	9.00	8.25	8.00	9.50	8.68	0.356
Promedio por aspecto evaluado			9.12	8.37	7.87	8.87	8.55	0.231
Int	Contenido	Organización	9.50	8.00	8.50	9.25	8.81	0.355
		Densidad	7.50	6.75	7.25	7.50	7.25	0.094
		Fiabilidad	8.25	8.50	8.00	9.50	8.56	0.324
		Comprensión	8.50	9.50	8.75	9.75	9.1	0.266
Promedio por aspecto evaluado			8.43	8.18	8.12	9.00	8.43	0.121
Int y RV	Opinión General de la App	Calificación subjetiva del participante	8.50	9.75	9.50	9.50	9.31	0.230
Promedios Totales y Global			8.71	8.61	8.03	8.83	8.54	1.381

Reactivo mejor calificado

Reactivo peor calificado

Reactivo con evaluación inversa

Tabla 5.18 Análisis comparativo de promedios generales por aspectos evaluados y sus reactivos. Elaboración propia.

Partiendo de la tabla general, en la siguiente gráfica (Fig. 5.8) se puede constatar la diferencia de promedio por aspecto evaluado mediante el test, lo que permite observar que grupo de participantes califico mejor y peor los diferentes aspectos planteados.

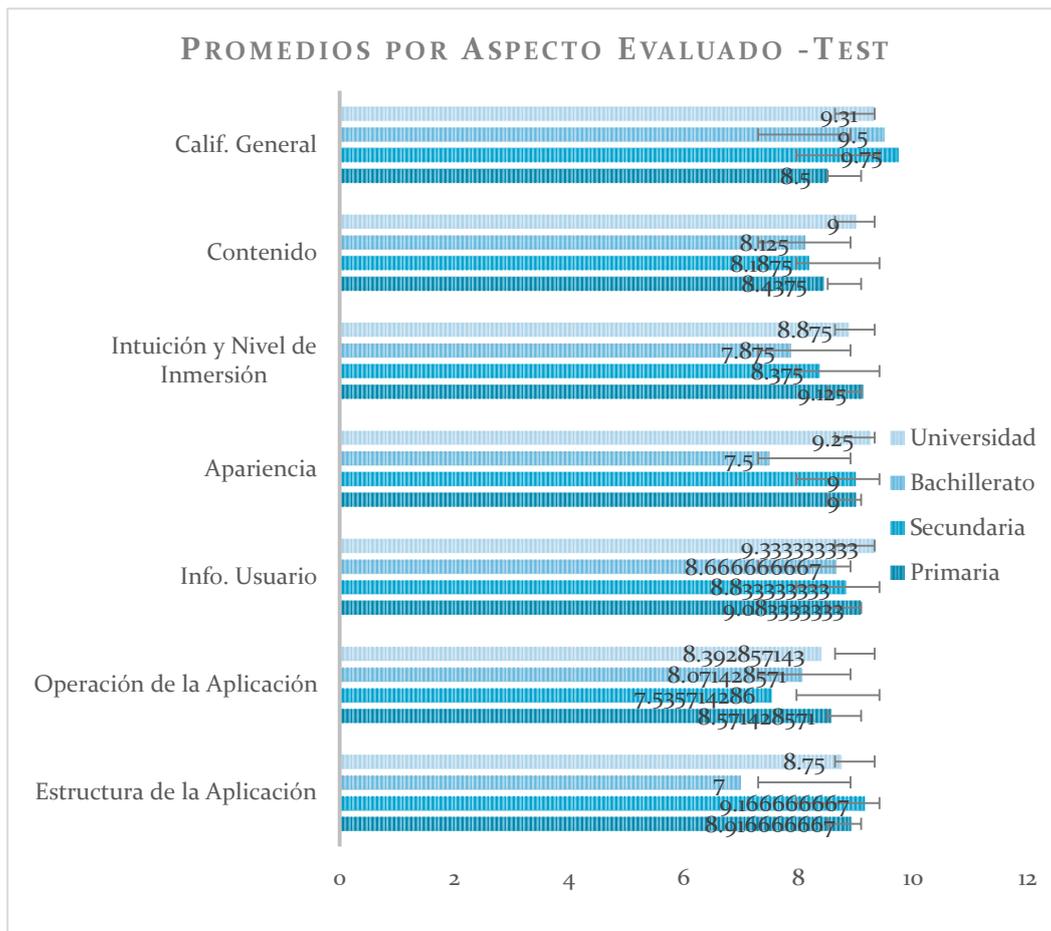


Fig. 5.8 Promedios por aspectos evaluados clasificados por nivel de escolaridad. Elaboración propia.

Cabe mencionar algunas observaciones sobresalientes en la gráfica “Promedio por aspecto evaluado” (Fig. 5.8), entre las cuales se encuentra “operación de la aplicación”

como el aspecto evaluado con menor promedio, recibiendo las calificaciones más bajas por parte del grupo secundaria cuyos participantes (en promedio) tienen un perfil “principiante”. Así mismo, se puede rescatar, gracias a la desviación estándar, que en un porcentaje de 71.4% de los aspectos evaluados, existe la tendencia a una evaluación positiva.

En este sentido, el cálculo de la desviación estándar, provee información interesante de probabilidad del comportamiento de las evaluaciones, así como un intervalo de confianza (límite máximo y mínimo) que establece un rango de calificaciones predominante (Fig. 5.9). El cálculo de estos parámetros se realizó con la finalidad de presentar información acerca de las inclinaciones presentes en las valoraciones emitidas por los participantes. La siguiente gráfica permite visualizar el rango de calificaciones que manejaron el mayor porcentaje de participantes, el cual oscila entre: 7 y 10, dato significativo y concluyente para la valoración del sistema a prueba.

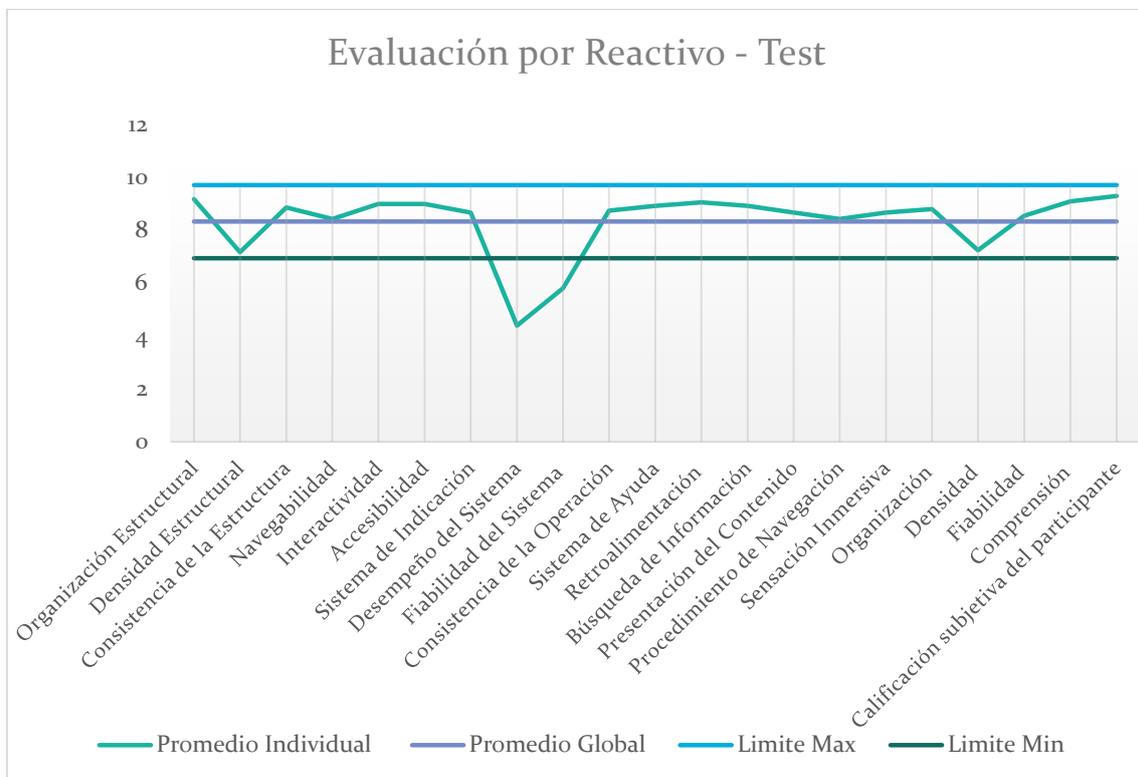


Fig. 5.9 Análisis de la desviación estándar y calificaciones generales por reactivo. Elaboración propia.

De igual manera, la gráfica “Evaluación por reactivo”, permite observar una súbita disminución de valores, la cual alcanza su punto más bajo en la valoración de “desempeño del sistema”, es decir, la velocidad de funcionamiento del sistema; esto no representa gran desventaja ya que solo comprende un pequeño porcentaje de la muestra total. Por otro lado, existe un comportamiento estable que se mantiene en gran porcentaje de los reactivos evaluados en un rango aproximado de: 8.8 y 9.2.

5.5.3 ASPECTOS LIMITANTES

Como se mencionó al principio de este *Capítulo V*, la finalidad del test de usabilidad es evaluar si el sistema ha sido desarrollado cumpliendo los requerimientos propios de la usabilidad. Sin embargo es importante mencionar que este test no garantizara el éxito del producto evaluado o si este será usado de manera adecuada por el público en general, esto se debe a varios aspectos que escapan del control del aplicador del test y (Rubin, 1994), explica el porqué de estos contratiempos.

- **Colecta de Datos.** Se considera que la generalización en los resultados puede derivar en la carencia de control sobre ciertos factores no previstos. Guillemette, 1989 por (Pereira, 2002)
- **Resultados.** Los resultados que arroje la evaluación serán de gran relevancia, sin embargo no se puede asegurar que el producto sea utilizado de la manera correcta. Rubin, 1994 por (Pereira, 2002)
- **Mala ejecución.** En base a las preguntas formuladas en el test de usabilidad, la técnica de implementación puede no ser la correcta por parte de los usuarios, lo que conlleva a una ejecución poco favorable para la óptima apreciación de características fundamentales a la hora de dar una calificación.

Todas las limitaciones planteadas fueron resueltas a manera de minorizar el rango de error en los datos recabados, sin embargo, algunos aspectos por su ambigüedad se siguen considerando como un pequeño porcentaje de circunstancias que pudiesen alterar de alguna manera la calificación emitida por el participantes.

C

APÍTULO VI. CONCLUSIONES Y FUTURAS LÍNEAS

DE INVESTIGACIÓN

Una máquina puede hacer el trabajo de cincuenta hombres ordinarios.

Ninguna máquina puede hacer el trabajo de un hombre extraordinario.

- Elbert Hubbard

La demanda de sistemas que simulan un entorno virtual, tal como se presenta en el *Capítulo I*, va en ascenso así como el desarrollo de herramientas para el desarrollo de la Realidad Virtual. Por ello, cada vez es más común incorporar técnicas de inteligencia artificial a este tipo de sistemas de cualquier índole (juegos, entornos virtuales, sistemas de simulación, etc.). Específicamente, para el presente sistema se utilizó la implementación de un agente inteligente reactivo, ya que por la naturaleza cambiante del entorno, se ajusta a las características de procesamiento de un agente.

La metodología orientada a prototipos, implementada permitió desarrollar el sistema mediante su tratamiento en dos fases, según está estipulado, lo cual implica generar un prototipo previo para establecer un comportamiento según los objetivos finales. Esto implica una ventaja principalmente para visualizar el sistema en términos generales, para poder determinar procesos y comportamientos internos. En la primera fase se desarrolla la interfaz de usuario, y en la segunda fase se desarrollan los procesos internos.

Por otro lado, en el diseño de la arquitectura se trató de resaltar ciertos factores dentro del desarrollo de este tipo de sistemas, por lo cual los diferentes módulos que se contemplaron llevan un orden de ejecución que contemplan lo más general a lo más específico del funcionamiento del sistema. Para definir los requerimientos generales se utilizó una arquitectura formal para sistemas de RV, así mismo, para desarrollar los procesos internos se utilizó una arquitectura más detallada, la cual fue adaptada a partir de una arquitectura de videojuegos (*Capítulo IV*) para cumplir con las características del sistema, ya que se implementó un agente inteligente reactivo. Quien fue determinante desde su concepto de encargarse de una tarea específica dentro de un *sistema* que, con su intervención, adquiere la propiedad de *inteligente*.

Por lo cual se tenía que generar relaciones con los procesos embebidos de un entorno virtual 3D. Esta estrategia fue de gran ayuda al momento de incorporar el algoritmo del “agente user”, pues facilitó la organización del comportamiento de este con cada módulo. Finalmente, el procesamiento del algoritmo resultó algo engorroso, es decir, la manera de manejar la información requerida para el funcionamiento del agente fue mediante arreglos y matrices que probablemente no fueron ejecutados de la mejor manera, por lo tanto, la velocidad del funcionamiento se vio afectada considerablemente.

Otro aspecto del funcionamiento del sistema esencial es la técnica de modelado tridimensional, la cual fue por polígonos que beneficio la geometría del modelo. Sin embargo, algunas de las normales fueron tratadas mediante la operación de fusión en algunos vértices, operación que influyó de manera perjudicial en el procesamiento gráfico, por lo cual el flujo de imágenes generadas se podía percibir lento y en algunos espacios no se apreciaba las propiedades definidas para cada objeto.

En cuanto al desarrollo de la GUI, no se presentaron inconvenientes significativos, el motor de Unity, incorpora herramientas muy potentes para realizar una interfaz interactiva muy agradable para el usuario, así mismo, el diseño del background, botones, paneles, barras de desplazamiento, títulos, etc., fueron generados mediante el programa

de diseño Photoshop, el cual demostró la compatibilidad necesaria con el software de desarrollo del sistema, Unity.

En cuanto al software de desarrollo Unity, se propone, mediante la presente Tesis, una visión particular en cuanto al desarrollo libre que ofrece el software. Esta característica es una de las más sobresalientes en cuanto al uso de este motor de videojuegos, ya que ofrece herramientas muy avanzadas de diseño de interfaces y procesos internos, sin embargo, con la utilización de estas técnicas muchas veces no se satisfacen los objetivos de un proyecto. Dicha situación se presentó en el desarrollo del agente inteligente incorporado en el comportamiento del usuario en el entorno virtual, no obstante, el sistema de navegación que provee Unity, maneja esta técnica de Inteligencia Artificial (agentes inteligentes) para personajes virtuales autónomos, en la definición de comportamientos predefinidos en un nivel básico.

A propósito del agente inteligente desarrollado en el presente documento, este debía fungir como “conciencia” en las decisiones del usuario dentro del entorno, lo cual se cumplió de manera limitada debido al algoritmo de búsqueda seleccionado, pues un óptimo aprendizaje requiere varios cálculos que a su vez dependen de un buen manejo de la información que retroalimenta estos procesos. Se consideró el algoritmo simplex de optimización Nelder-Mead, con el propósito de implementar un algoritmo no convencional en sistemas de esta índole, de esta manera fueron aplicadas sus operaciones destinadas a la optimización de procesos, a calcular las futuras posiciones del usuario en base a la posición inicial y posición objetivo.

Cabe mencionar que existen algoritmos más simples para el propósito de la intervención planteada en el presente documento, entre los ya comprobados en diversos proyectos existe, por ejemplo, el algoritmo de la colonia de hormigas, el cual ha tenido muy buena aceptación en sistemas de optimización de búsqueda. También es preciso señalar que todas las intervenciones programables fueron generadas a partir del lenguaje de programación C#, y no se presentaron inconvenientes de ejecución o desarrollo en Unity. Existe la información necesaria y se trabajó de manera cómoda con dicho lenguaje.

Las limitaciones que se pudieron observar están relacionadas con la aplicación del algoritmo del agente inteligente, sin embargo, logra satisfacer los requerimientos funcionales del sistema. Dentro de la literatura existen muchos proyectos en diferentes ámbitos de aplicación de esta técnica de IA, sin embargo, una de las principales aportaciones del presente proyecto es el enfoque planteado para el agente, ya que no se había trabajado de esta manera. También, la adaptación de la arquitectura de videojuegos para un programa de recorrido interactivo intrínseco, fue sumamente beneficiosa para el correcto funcionamiento de cada uno de los módulos.

Finalmente, gracias a los datos recabados en la prueba de usabilidad del sistema desarrollado, se puede concluir que, aunque el sistema presenta deficiencias, estas no son significativas para el objetivo final de la aplicación. Es decir, se cuenta con un sistema

funcional y aceptado por una muestra de 32 personas de la población considerada como objetivo, con una media de 85% en una evaluación formal con distintos métodos.

Además, aspectos considerados sobresalientes en una aplicación de RV de escritorio, tales como: interactividad, retroalimentación, presentación del contenido, sensación inmersiva, comprensión del contenido, obtuvieron calificaciones sobresalientes en cada una de las divisiones que se presentaron a lo largo de la aplicación de las pruebas de usabilidad, además de ser variables determinantes de acuerdo al cálculo de la desviación estándar. Lo cual indica la importancia de las variables en el análisis pues según el nivel de acercamiento con un promedio general adquieren un valor diferente en el estudio. Particularmente,

6.1 FUTURAS LINEAS DE INVESTIGACIÓN

El desarrollo de la aplicación se enfocó en el agente inteligente y su comportamiento con el entorno, sin embargo en el presente trabajo el funcionamiento de la interfaz de usuario fue de cierta manera relegada a un segundo plano en lo que respecta a la aplicación de dicha técnica. Por ello se tiene contemplado, tal como se menciona en el *Capítulo IV*, programar un agente de interfaz para encargarse de manejar información específica del usuario, mediante la creación de una cuenta/perfil para los usuarios que deseen tener una experiencia personalizada en lo que respecta a la oferta educativa de este inmueble; incluso, se piensa en generar un modelo adaptable a diferentes espacios culturales.

En cuanto al entorno virtual se tiene contemplado desarrollar un modelo más sofisticado del espacio y añadir la posibilidad de elegir un avatar con gesticulaciones, movimiento de brazos piernas, y mayor detalle, con la finalidad de que el usuario pueda visualizar una representación propia en el mundo virtualizado, y con ello tener acceso a vistas de diferentes perspectivas (3ra persona, 1ra persona, etc.).

Al no existir una metodología y arquitectura formalizada para este tipo de sistemas se podría desarrollar una en base a los resultados obtenidos por las adaptaciones que se realizar para el desarrollo del presente sistema.

BIBLIOGRAFÍA

- Alejandro Cerón Cardona, P. A. (2014). Manual Basico de Unity 3D como Apoyo al Desarrollo Turístico Nacional. Pereira .
- Andrea Bonsch, T. V. (2017). Turning Anonymous Members of a Multiagent System into Individuals. Aquisgrán. Visual Computing Institute, RWTH Aachen University. USC Institute for Creative Technologies, USA. , Alemania .
- Anu Sivunen, E. N. (2014). Social Presence as a Multi-Dimensional Group Construct in 3D Virtual Environments. *Journal of Computer-Mediated Communication*, 20: 19–36.
- Arango, J. A. (Julio de 2014). Desarrollo de Entornos Virtuales Inteligentes Basados en el Meta-Modelo MAM5 . Valencia, Universidad Politecnica de Valencia. Departamento de Sistemas Informáticos y Computación Valencia, España.
- Arsalan Heydarian, J. P.-G. (2015). Immersive virtual environments versus physical built environments: A benchmarking study for building design and user-built environment explorations. *Automation in Construction*, Volumen 54, Pages 116-126.
- Artes, C. N. (2017). *CENART*. Obtenido de <http://www.cenart.gob.mx>
- Autodesk, C. C. (2014). Novedades de Revit.
- Blanco, A. (2010). Computer Science Engineer Web Developer - Multimedia Project Manager . España.
- Blogs, U. (3 de September de 2014). *Documentation, Unity scripting languages and you*. Obtenido de blogs.unity3d.com
- Borrell, G. (26 de Oct de 2016). *GitHub, API_Market*. Obtenido de <https://github.com>
- Brunet, P. (2012). Sistemas Graficos Interactivos: Arquitecturas de Realidad Virtual . Barcelona. Departamente de Software, Universidad Politecnica de Cataluña, España.
- Cardozo, H. J. (2004). *Realidad Virtual* . TAI 2 – 2004 - UCA .
- Carlos González Morcillo, D. V. (2012). *Realidad Aumentada: Un enfoque Práctico con ARToolkit y BLENDER* . Bubok Publishing, IdenTIC .
- Carmen Ortigueira España, M. R. (2009). Realidad Virtual . A Coruña, España .
- Castellanos, L. R. (2012). Desarrollo de Sistemas de Información . Maracaibo , Venezuela .
- Cervigón, C. (2012). Interfaces gráficas de usuario. . Madrid. Universidad Complutense Madrid, España.
- Coca Bergolla, Y. (2009). Agentes inteligentes. Aplicación a la realidad virtual . *Revista Cubana de Ciencias Informáticas*, Vol. 3, No. 1-2.
- Cultura, S. d. (2015). Encuesta de Consumo Cultural de la Ciudad de México. CDMX, México.
- Cultural, R. N. (Octubre de 2016). *Coordinación Nacional de Desarrollo Instirucional/SIC*. Obtenido de sic.cultura.gob.mx

- Das, K. (Julio de 2014). Multi-Agent Positional Consensus Under Various Information Paradigms. Short article . Bangalore. Indian Institute of Science. , India.
- Dra. Lilliam Perurena Cancio, I. M. (2013). Usability of Web sites, Methods and Evaluation Techniques. *Revista Cuabana de Información en Ciencias de la Salud* , Vol. 24, Núm 2.
- Felipe Díaz del Castillo Rodríguez, E. d. (2015). Manual de Uso y Aplicación del Programa Autocad. Edo. de México, México.
- Fernández, M. (Abril de 2003). LOS MUSEOS ESPACIOS DE CULTURA, ESPACIOS DE APRENDIZAJE. Barcelona.
- FormacionProfesional.info. (2016). 3D Autodesk Maya.
- Fredy Martínez, F. M. (2016-2017). Visual identification and similarity measures used for on-line motion planning of autonomous robots in unknown environments. *Eighth International Conference on Graphic and Image Processing (ICGIP 2016)*. Tokio .
- Gallardo, A. (2001). *3D LIGHTING: History, Concepts, and Techniques*. EE.UU.: Charles River Media.
- Garduño, E. A. (Julio de 2012). Análisis y Diseño de la Interfaz para un Sistema de Aprendizaje Colaborativo Apoyada por un Agente Tutor Inteligente (Tesis Doctoral). México, Universidad Autónoma Metropolitana. División de Ciencias y Artes para el Diseño. , México.
- Geig, M. (19 de December de 2013). Working with Models, Materials, and Textures in Unity Game Development. EE. UU. .
- General, C. (6 de Mayo de 2015). Definiciones de Física Fundamental. . España.
- Giuseppe De Giacomo, V. N. (2014). Agent Behavior Composition in Virtual Environments Realized Using Game Engines. *21rd European Conference on Artificial Intelligence (ECAI) 2014*. Prague.
- Gómez, I. D. (Junio de 2006). Técnicas y Métricas para la Evaluación de la Usabilidad. Popayán. Universidad de Cauca. Facultad de Ingeniería Electrónica y Telecomunicaciones, Colombia.
- González Morcillo Carlos, V. F. (2009). Fundamentos de Síntesis de Imagen 3D. . Castilla- La Mancha. Centro de excelencia de software libre. , España.
- Gonzalo Luzardo, J. H. (5 de Agosto de 2014). Inteligencia Artificial en Ambientes Virtuales: Humanos Virtuales Autónomos (HVA) como Agentes Virtuales Inteligentes (3DIVA) . Madrid, España.
- Group, Z. D. (2017). *PC / Encyclopedia*. Obtenido de www.pcmag.com
- Hernández, L. R. (2017). Inteligencia Artificial: Agentes inteligentes y Ambientes. Ciudad de México. Universidad Nacional Autónoma de México , México.
- Hollinger, D. (2016-2017). Artificial Intelligence and Machine Learning. Breslavia, Wroclaw University of Technology. Faculty of Electronics. Department of Cybernetics and Robotics, Polonia .
- Ibañez, M. L. (2004). Animación Comportamental de Personajes Inteligentes 3D Basada en MiniMin-HSP (Heuristic Search Planning). Valencia. Universidad de Valencia. , España.
- INEGI. (2012). *Encuesta Nacional de Consumo Cultural en México* . México.
- INEGI. (2016). *Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en los Hogares 2016*.

- InTec, D. O. (26 de Diciembre de 2014). Tutorial de Sketchup (Crear, compartir y presentar modelos 3D). Buenos Aires.
- INTEF, I. N. (2012). Blender 3D en la educación . España.
- Irma Lucía Franco Sepúlveda, F. (2007). Los simuladores, estrategia formativa en ambientes virtuales de aprendizaje . *Revista Virtual Universidad Católica del Norte*.
- ISO, I. O. (1998). Ergonomic requirements for office work with visual display terminals (VDTs). EUA.
- Jahanzeb Hafeez, S. L. (2017). Image Based 3D Reconstruction of Texture-less Objects for VR Contents. *International Journal of Advanced Smart Convergence*, Vol.6 No.1 9-17.
- Jesus Alberto Flores Cruz, P. C. (2014). La realidad virtual, una tecnología innovadora aplicable al proceso de enseñanza de los estudiantes de ingeniería . *UDGVirtual*, Vol 6, No 2 .
- José T. Palma Méndez, R. M. (2008). *Inteligencia Artificial* . Madrid : McGRAW-HILL.
- Josu Ahedo Ruiz, I. D. (2013). Las nuevas tecnologías como herramientas que facilitan la educación formativa en la educación. España.
- Juan Manuel Jiménez Gutiérrez, L. F. (2014). Animación y Reconstrucción sobre la Zona Prehispánica de Monte Albán, Oaxaca. México .
- Kelly S. Hale, K. M. (2015). *Handbook Of Virtual Environments: Design, Implementation, and Applications*. Boca Raton, Florida: CRC Press, Taylor & Francis Group.
- Llobet, I. S. (2015). Babel: desarrollo de un videojuego de plataformas en Unity. Valencia. Universidad Politécnica de Valencia. , España.
- López, C. (17 de Septiembre de 2008). Arte y arquitectura digital netart y universos virtuales.
- López, J. R. (Octubre de 2005). Contribución a los métodos de optimización basados en procesos naturales y su aplicación a la medida de antenas en campo próximo. Cantabria. Universidad de Cantabria. Departamento de Ingeniería de Comunicaciones, España.
- Lujak, M. (2015). *Inteligencia Artificial Distribuida*. España.
- Luna, J. N. (Septiembre de 2012). Realidad Virtual, Estado del arte y análisis crítico (Máster en Desarrollo de Software). Granada. Universidad de Granada. , España.
- Luz, S. (October de 2014). Reactive Agents & Simulation. Dublín. Interactive Entertainment Technology. School of Computer Science and Statistics, Irlanda.
- M.en C. Pozas Cárdenas Mariano Javier, M. e. (19 de 05 de 2015). *Sistemas de Realidad Virtual* . Hidalgo, México.
- Mabanza, N. (July de 2016). Determining the Usability of Pedagogical Interface Agents in the Context of Adult Computer Literacy Training: A South African Perspective. Bloemfontein. University of the Free State. Department of Computer Science and Informatics, Sudáfrica .
- María del C. Ramos, J. L. (2007). Creación de ambientes virtuales inmersivos con software libre. *Revista Digital Universitaria UNAM*, Vol 8. No. 6. .
- Mario Luis Arce, J. G. (2012). *3D Max Studio* . Madrid, España: Prensa Técnica .
- Martínez, F. J. (2011). Presente y Futuro de la Tecnología de. *Creatividad y Sociedad*, 39.
- Martínez, J. R. (2015). *Manual 3D Studio Max (Conviértete en experto)*. México.

- Mathieu Aubry, B. C. (March de 2014). Painting-to-3D model alignment via discriminative visual elements. *ACM Transactions on Graphics (TOG)*, Article No. 14.
- Millán, T. (2005). Los referentes virtuales en la construcción de la realidad. *Global Media Journal*, vol 2, núm 3.
- Ministerio de Ciencia, T. e. (2013). Fotogrametría . Argentina, Secretaría de Articulación Científico Tecnológica.
- Moriel, A. G. (26 de Febrero de 2015). *Applenology*. Obtenido de www.applenology.com
- Multimedia, C. (s.f.). *Centro Multimedia. Centro Nacional de las Artes* . Obtenido de cmm.cenart.gob.mx
- Murray, J. W. (2014). *C# Game Programming Cookbook for Unity 3D*. Boca Raton , Florida: CRC Press. Taylor & Francis Group .
- Naimark, M. (2006). Aspen the Verb: Musings on Heritage and Virtuality. *Presence journal special issue on Virtual Heritage, MIT Press*.
- Olguin Carbajal, M., Rivera Zárate, I., & Hernández Montañez, E. (2006). Introducción a la Realidad Virtual. *Polibits*, 11-15.
- Pereira, H. B. (Septiembre de 2002). Análisis experimental de los criterios de evaluación de usabilidad de aplicaciones multimedia en entornos de educación y formación a distancia. . Universidad Politecnica de Cataluña. Departamento de Expresión Gráfica e Ingeniería., España.
- Pozo, G. M. (Sep. de 2008). Una Arquitectura Software Basada en Agentes y Recomendaciones Metodológicas para el Desarrollo de Entornos Virtuales de Entrenamiento con Tutoría Inteligente. Madrid , España .
- Qing-yun Dai, J.-d. W.-T.-h.-y. (2013). Inverter Power Monitoring System Based on VRML Virtual Reality Technology . *International Journal of Smart Home* , (págs. 187-196).
- R. Mora, H. A. (April de 2014). Looking ahead: a vision-based software for predicting pedestrian movement. *Engineering and Research*, Vol. 34, No. 1, pp. 79 – 82.
- Robles, J. M. (20 de Enero de 2017). Las tecnologías más relevantes para el futuro de la humanidad. Madrid, España.
- Rodríguez, M. Á. (12 de 2003). Desarrollo de Mundos Virtuales en el Área de Más Sobre Ciencia del Museo Virtual 3D el Rehilete . Pachuca de Soto , Hidalgo, México .
- Rodríguez, M. V. (21 de Septiembre de 2015). Realidad Virtual Aplicada a los Viajes . España .
- S. Quero, C. B.-P. (2012). La realidad virtual para el tratamiento de los trastornos emocionales: una revisión. *Anuario de Psicología Clínica y de la Salud*, 7-21.
- S.L., D. S. (s.f.). Ergoestudio, Sobre las pruebas de usuario. . Bilbao, España. Obtenido de <http://www.ergoestudio.com>
- Salavert, I. R. (2010). Tendencias Actuales en la Interacción Persona-Ordenador. Accesibilidad, Adaptabilidad y Nuevos Paradigmas. España.
- Salcedo, M. Á. (2010). Sistema de Rehabilitación Cardíaca Aumentado por Realidad Virtual (Tesis Maestría). San Sebastián. Universidad del País Vasco. Departamento de Ciencia de la Computación e Inteligencia Artificial, País Vasco.
- Santiago, S. P. (2015). Modelo de un Entorno Virtual Inteligente Basado en la Percepción y el Razonamiento de sus Elementos con un Personaje para la Generación de Realismo (Tesis

- Doctoral). Medellín, Universidad Nacional de Colombia. Facultad de Minas, Departamento de Ciencias de la Computación, Colombia.
- Serkan Dincer, A. D. (2015). The Impact of Pedagogical Agent on Learners' Motivation and Academic Success. *De Gruyter Open. Practice and Theory in Systems of Education*, Volumen 10, number 4.
- Significados.com. (9 de Octubre de 2017). *Significados: descubrir lo que significa, conceptos y definiciones*. Obtenido de www.significados.com
- Slick, J. (October de 2016). How 3D Modeling Is Used?. Hanover, Nuevo Hampshire, EE.UU.
- Soto, A. V. (12 de Mayo de 2012). Técnicas de programación Binivel para el diseño de redes. Cholula. Escuela de Ingeniería, Universidad de las Américas Puebla. Departamento de Ingeniería Industrial y Mecánica. , Puebla, México.
- Stuart J. Russell, P. N. (2004). *Inteligencia Artificial un Enfoque Moderno*. Madrid, España: Pearson, Prentice Hall.
- Technologies, U. (2017). Unity Documentation. Publication 5.3-Q. EUA.
- Thanos G. Stavropoulos, E. K. (2014). An Applied Energy Management Approach in Intelligent Environments based on a Hybrid Agent Architecture. *21rd European Conference on Artificial Intelligence (ECAI) 2014*. Praga.
- Thompson, M. (2017). *Virtual Worlds: Art Studio Extension*. New York: ProQuest.
- Ton Roosendaal, S. S. (Junio de 2014). El Oficial Blender: Suit Abierta de Creación 3D (Edición especial traducida).
- V. Juřík, L. H. (July 2016). Cognitive Aspects of Collaboration in 3D Virtual Environments. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, (págs. Volume XLI-B2, 2016 XXIII ISPRS Congress, 12-19).
- Valdez, K. V. (2011). Estudio Comparativo de herramientas de software libre y propietario para modelado 3D. Caso práctico Modelado de Rostros Humanos. Universidad de Murcia. Centro de Estudios de Arqueología Virtual. , España.
- Varela, Á. M. (2015). An application of interactive multi-criteria optimization to air pollution control / Anex01. Universidad de Vigo, Depto. de Matemática Aplicada, Vigo, España, España.
- Vargas, J. R. (2012). Computación Gráfica. Bogota. Fundación Universitaria San Martin Zipaquirá. Ingeniería en Sistemas, Colombia.
- Vega, K. F. (Febrero de 2010). TREG: Un juego de entrenamiento en Ingeniería de Requisitos. Rio de Janeiro. , Brasil.
- Xin Bai, J. L. (Octubre de 2013). Enhance Learning in a Virtual Professional. *GSTF International Journal of Nursing and Health Care (JNHC)*, Vol.1 No.1.
- Yang Kuang, J. J. (2015). Research on Key Problems of Making 3D Scene Model Based on VRML. *International Industrial Informatics and Computer Engineering Conference (IIIEEC 2015)* (págs. 1283-1286). The authors - Published by Atlantis Press.
- Zaragoza., A. d. (10 de 2012). Gestión de Proyectos en Videojuegos de Última Generación / Integración de un proyecto de Videojuego. . Zaragoza, España.
- Zumaquero, S. F. (8 de Febrero de 2010). Métodos de evaluación de la usabilidad para entornos de RV, RA y Sistemas Ubícuos. Albacete, España.

APÉNDICE A: TEST DE USABILIDAD

**Test de Evaluación de Usabilidad para Sistemas en Realidad Virtual
Aplicación en Realidad Virtual - CNA**

Parte 1: Datos del participante	Tiempo de aplicación:	<input style="width: 80%;" type="text"/>
Nombre: <input style="width: 80%;" type="text"/>	Sexo: F	<input type="checkbox"/>
Edad: <input style="width: 80%;" type="text"/>	M	<input type="checkbox"/>

Escolaridad	Primaria <input style="width: 50%;" type="text"/>	Universidad <input style="width: 50%;" type="text"/>
	Secundaria <input style="width: 50%;" type="text"/>	Maestría <input style="width: 50%;" type="text"/>
	Bachillerato <input style="width: 50%;" type="text"/>	Doctorado <input style="width: 50%;" type="text"/>
	<input style="width: 50%;" type="text"/>	<input style="width: 50%;" type="text"/>

¿Cuántas horas diarias dedicas a trabajos con una computadora?

Menos de 1 Entre 1 y 2 Entre 2 y 4 Entre 4 y 8 Más de 8

No utilizo

¿Para cuál actividad utilizas una computadora?

Trabajar Estudiar Entrenimiento No

En los últimos 5 meses, ¿qué tipo de software ha utilizado? (Seleccione las casillas que estén de acuerdo con su experiencia)

	Nula	Principiante	Intermedio	Avanzado
Sistema Operativo UNIX (Linux)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sistema Operativo Windows	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sistema Operativo Macintosh	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Lenguajes de programación	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Procesadores de texto (Word, notas)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Hojas de Cálculo (Excel)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sistemas de Bases de Datos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sistemas corporativos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Edición gráfica (Photoshop, Paint)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
CAD	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Aplicaciones Multimedia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Navegadores y buscadores de Internet	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Correo Electrónico	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Chats	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Juegos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Películas en DVD	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Música	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Otras aplicaciones	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Parte II: Características de las Aplicaciones Multimedia

Nombre de la Aplicación: CNA - RV

Marque las casillas que mejor represente su opinión sobre las características de la aplicación en Realidad Virtual que ha utilizado.

1. Estructura de la aplicación

1.1 Organización estructural: La distribución de los elementos estructurales de la aplicación es buena (barras de desplazamiento, zonas de contenido, botones, etc.).

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

1.2 Densidad estructural: La cantidad de elementos estructurales que se utilizan en la aplicación es excesiva.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

1.3 Consistencia de la estructura: La distribución de los elementos estructurales se mantiene constante a lo largo de la aplicación.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

2. Operación de la aplicación

2.1 Navegabilidad: El recorrido que se hace por el contenido de la aplicación es fácil.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

2.2 Interactividad: La relación mutua entre el usuario y la aplicación es buena.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

2.3 Accesibilidad: Las acciones que solicita la aplicación son fáciles de ejecutar.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

2.4 Sistema de indicación: Se identifican fácilmente las figuras, las tablas, los hipertextos, las zonas activas y el tipo de acción que se debe ejecutar.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

2.5 Desempeño del sistema: La velocidad de funcionamiento de la aplicación, considerando el tipo de tarea que exige, es buena.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

2.6 Fiabilidad del sistema: Hay demasiados errores durante la operación de la aplicación.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

2.7 Consistencia de la operación: La ejecución de tareas (navegar por la app, hacer clic en botones, seleccionar opciones) sigue un estándar a lo largo de la aplicación.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

3. Información del Usuario

3.1 Sistema de ayuda: Las dudas del usuario se resuelven fácilmente.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

3.2 Retroalimentación: La aplicación mantiene el usuario informado sobre las tareas a ejecutar.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

3.3 Búsqueda de información: Los datos que busca el usuario son fáciles de encontrar.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

4. Apariencia

4.1 La presentación del contenido: Tipo y tamaño de fuente, uso del color, disposición de los elementos según su significado es buena.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

5. Intuición y Nivel de Inmersión

5.1 Los procedimientos de navegación por la aplicación o ejecución de tareas asignadas se aprenden de forma prácticamente inmediata.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

5.2 La sensación de presencia dentro del entorno es buena.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

6. Contenido

6.1 Organización del contenido: La distribución del contenido de la aplicación es buena (textos, botones, imágenes).

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

6.2 Densidad del contenido: La información que se presenta en la aplicación es demasiado extensa.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

6.3 Fiabilidad del contenido: No hay errores en la información que se presenta.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

6.4 Comprensión del contenido: La información presentada es fácil de entender.

En										De
Desacuerdo	2	4	6	8	10					Acuerdo

7. Opinión de la aplicación en general

Mala										Buena
	2	4	6	8	10					