



# ANEXOS



# CONTENIDO

---

<b>ANEXO A BASES TEORICAS.....</b>	<b>1</b>
A.1. VIDEOJUEGOS .....	1
A.2.1. <i>Historia de los Videojuegos</i> .....	2
A.2.2. <i>Tipos de videojuegos</i> .....	2
A.2.3. <i>Personajes</i> .....	5
A.3. AGENTES SOFTWARE.....	7
A.3.1. <i>¿Qué es un Agente?</i> .....	7
A.3.2. <i>Propiedades de los Agentes</i> .....	8
A.3.3. <i>Metodologías de Agentes</i> .....	10
A.3.4. <i>Teoría de los Actos de Habla</i> .....	20
A.4. MOTORES DE JUEGOS .....	25
A.4.1. <i>Fly3D</i> .....	26
A.4.2. <i>Unreal</i> .....	27
A.4.3. <i>Genesis3D</i> .....	27
A.4.4. <i>Cristal Space</i> .....	27
A.4.5. <i>XNA</i> .....	28
A.4.6. <i>Ogre</i> .....	29
<b>ANEXO B SELECCIÓN DE LA METODOLOGÍA PARA LA CONSTRUCCIÓN DE PERSONAJES .....</b>	<b>30</b>
B.1. DEFINICIÓN. ....	30
B.2. REVISIÓN DE METODOLOGÍAS PARA AGENTES. ....	31
<b>ANEXO C APLICACIÓN DE LA METODOLOGÍA PARA LA CONSTRUCCIÓN DE PERSONAJES EN UN VIDEOJUEGO DE AVENTURA EN 3D.....</b>	<b>49</b>
C.1. CONCEPTUALIZACIÓN .....	49
C.1.1. <i>StoryLine – Simon y la tuberculosis – Nivel 2</i> .....	49
C.1.2. <i>Análisis del argumento base</i> .....	51
C.1.3. <i>Definición de reglas</i> .....	67
C.1.4. <i>Definición del mundo</i> .....	72
C.1.5. <i>Identificación de Personajes</i> .....	74
C.1.6. <i>Descripción de los Personajes</i> .....	78
C.1.7. <i>Identificación de los casos de uso de los personajes</i> .....	82
C.1.8. <i>Descripción de los casos de uso</i> .....	85
C.2. MODELO DE COMPORTAMIENTO .....	90
C.2.1. <i>Notación grafica</i> .....	91
C.3. MODELO DE COORDINACIÓN.....	94



C.4.	MODELO DE ORGANIZACIÓN .....	119
C.5.	MODELO DE DISEÑO .....	120
C.5.1.	<i>Diseño De Agentes</i> .....	120
C.5.2.	<i>Diseño De Plataforma</i> .....	121
<b>BIBLIOGRAFIA.....</b>		<b>122</b>



## LISTA DE FIGURAS

---

<i>Figura 1 – Agente Zeus</i> .....	18
<i>Figura 2 – Notación de los personajes para el argumento base</i> .....	51
<i>Figura 3 – Controles de desplazamiento</i> .....	68
<i>Figura 4 – Botón de Esc</i> .....	68
<i>Figura 5 – Iconos Elementos del Juego</i> .....	69
<i>Figura 6 - Estructura del Juego</i> .....	71
<i>Figura 7 – Bocetos del terreno</i> .....	72
<i>Figura 8 – Ubicación Unidades Terreno</i> .....	73
<i>Figura 9 – Diagrama Casos de Usos 1</i> .....	89
<i>Figura 10 – Diagrama Casos de Uso 2</i> .....	89
<i>Figura 11 – Diagrama del comportamiento aguardar</i> .....	91
<i>Figura 12 – Diagrama del comportamiento Patrullar</i> .....	91
<i>Figura 13 – Diagrama del comportamiento Acompañar</i> .....	92
<i>Figura 14 – Diagrama del comportamiento Recibir</i> .....	93
<i>Figura 15 – Diagrama Conversación Saludo</i> .....	113
<i>Figura 16 – Diagrama Conversación guía</i> .....	114
<i>Figura 17 – Conversación Diagrama Conversación Soldado</i> .....	115
<i>Figura 18 – Conversación Repartir tapabocas</i> .....	116
<i>Figura 19 – Conversación Conspiración</i> .....	117
<i>Figura 20 - Conversación Don Segundo</i> .....	118
<i>Figura 21 – Modelo de Organización “Simón y La Tuberculosis”</i> .....	119



## LISTA DE TABLAS

---

<i>Tabla 1 – Tecnología de agentes</i> .....	22
<i>Tabla 2 – Aplicaciones de prueba</i> .....	22
<i>Tabla 3 – Evolucion del modelo de comunicaciones de agentes</i> .....	22
<i>Tabla 4 – Uso de entornos software existentes</i> .....	22
<i>Tabla 5 - Revisión de Metodologías</i> .....	41
<i>Tabla 6 – Análisis (Compleitud)</i> .....	42
<i>Tabla 7 – Análisis (Aplicabilidad)</i> .....	42
<i>Tabla 8 – Análisis (Complejidad)</i> .....	43
<i>Tabla 9 – Análisis (Validada)</i> .....	44
<i>Tabla 10 – Análisis (Argumentada)</i> .....	44
<i>Tabla 11 – Diseño (Compleitud)</i> .....	45
<i>Tabla 12 – Diseño (Complejidad)</i> .....	45
<i>Tabla 13 – Diseño (Validada)</i> .....	46
<i>Tabla 14 – Diseño (Argumentada)</i> .....	47
<i>Tabla 15 – Desarrollo (Compleitud)</i> .....	47
<i>Tabla 16 – Desarrollo (Complejidad)</i> .....	48
<i>Tabla 17 – ESCENA: Inicio</i> .....	54
<i>Tabla 18 – ESCENA: Encontrando a Don Segundo</i> .....	58
<i>Tabla 19 – ESCENA: Obteniendo el Cristal</i> .....	61
<i>Tabla 20 – ESCENA: Cuestionario</i> .....	63
<i>Tabla 21 – ESCENA: Desenlace</i> .....	65
<i>Tabla 22 – Identificación Simón</i> .....	74
<i>Tabla 23 – Identificación Oritah</i> .....	75
<i>Tabla 24 – Identificación Don Segundo</i> .....	76
<i>Tabla 25 – Identificación Soldados</i> .....	76
<i>Tabla 26 – Identificación Campesinos</i> .....	77
<i>Tabla 27 – Descripción Simón</i> .....	78
<i>Tabla 28 – Descripción Oritah</i> .....	79
<i>Tabla 29 – Descripción Don Segundo</i> .....	80
<i>Tabla 30 – Descripción Soldado</i> .....	81
<i>Tabla 31 – Descripción Campesino</i> .....	82
<i>Tabla 32 – Caso de Uso Aguardar</i> .....	82



<i>Tabla 33 – Caso de Uso Acompañar a Simón</i> .....	83
<i>Tabla 34 – Caso de Uso Patrullar</i> .....	83
<i>Tabla 35 – Caso de Uso Repartir Tapabocas</i> .....	83
<i>Tabla 36 – Caso de Uso Encontrar Oritah</i> .....	84
<i>Tabla 37 – Caso de Uso Encontrar Don Segundo</i> .....	84
<i>Tabla 38 – Caso de Uso Obtener Cristal</i> .....	84
<i>Tabla 39 – Descripción Caso de Uso Aguardar</i> .....	85
<i>Tabla 40 – Descripción Caso de Uso Acompañar a Simón</i> .....	86
<i>Tabla 41 – Descripción Caso de Uso Patrullar</i> .....	86
<i>Tabla 42 – Descripción Caso de Uso Repartir Tapabocas</i> .....	87
<i>Tabla 43 – Descripción Caso de Uso Encontrar Oritah</i> .....	87
<i>Tabla 44 – Descripción Caso de Uso Encontrar Don Segundo</i> .....	88
<i>Tabla 45 – Descripción Caso de Uso Obtener Cristal</i> .....	89
<i>Tabla 46 – Comportamiento Aguardar</i> .....	90
<i>Tabla 47 – Comportamiento Acompañar</i> .....	90
<i>Tabla 48 – Comportamiento Patrullar</i> .....	90
<i>Tabla 49 – Comportamiento Recibir</i> .....	90
<i>Tabla 50 – MSJ01</i> .....	94
<i>Tabla 51 – MSJ02</i> .....	94
<i>Tabla 52 – MSJ03</i> .....	95
<i>Tabla 53 – MSJ04</i> .....	95
<i>Tabla 54 – MSJ05</i> .....	95
<i>Tabla 55 – MSJ06</i> .....	96
<i>Tabla 56 – MSJ08</i> .....	96
<i>Tabla 57 – MSJ07</i> .....	97
<i>Tabla 58 – MSJ09</i> .....	97
<i>Tabla 59 – MSJ10</i> .....	97
<i>Tabla 60 – MSJ11</i> .....	98
<i>Tabla 61 – MSJ12</i> .....	98
<i>Tabla 62 – MSJ13</i> .....	98
<i>Tabla 63 – MSJ14</i> .....	99
<i>Tabla 64 – MSJ15</i> .....	99
<i>Tabla 65 – MSJ16</i> .....	100
<i>Tabla 66 – MSJ17</i> .....	100
<i>Tabla 67 – MSJ18</i> .....	101
<i>Tabla 68 – MSJ19</i> .....	101
<i>Tabla 69 – MSJ20</i> .....	102
<i>Tabla 70 – MSJ21</i> .....	102
<i>Tabla 71 – MSJ22</i> .....	102
<i>Tabla 72 – MSJ24</i> .....	103
<i>Tabla 73 – MSJ25</i> .....	103
<i>Tabla 74 – MSJ26</i> .....	104



<i>Tabla 75 – MSJ27</i> .....	104
<i>Tabla 76 – MSJ28</i> .....	105
<i>Tabla 77 – MSJ29</i> .....	105
<i>Tabla 78 – MSJ30</i> .....	105
<i>Tabla 79 – MSJ31</i> .....	106
<i>Tabla 80 – MSJ32</i> .....	106
<i>Tabla 81 – MSJ33</i> .....	107
<i>Tabla 82 – MSJ34</i> .....	107
<i>Tabla 83 – MSJ35</i> .....	108
<i>Tabla 84 – MSJ36</i> .....	108
<i>Tabla 85 – MSJ37</i> .....	109
<i>Tabla 86 – MSJ38</i> .....	109
<i>Tabla 87 – MSJ39</i> .....	110
<i>Tabla 88 – Conversación Saludo</i> .....	110
<i>Tabla 89 – Conversación guía Orah</i> .....	111
<i>Tabla 90 – Conversación Soldado</i> .....	111
<i>Tabla 91 – Conversación Conspiración</i> .....	111
<i>Tabla 92 – Conversación repartir tapabocas</i> .....	112
<i>Tabla 93 – Conversación Soldado</i> .....	112
<i>Tabla 94 – Arquitectura Orah</i> .....	120
<i>Tabla 95 – Arquitectura Don Segundo</i> .....	120
<i>Tabla 96 – Arquitectura Soldado</i> .....	120
<i>Tabla 97 – Arquitectura Campesino</i> .....	121
<i>Tabla 98 – Arquitectura de la plataforma</i> .....	121



# ANEXO A

## BASES TEORICAS

---

Este capítulo contiene las bases teóricas sobre las cuales se encuentra enmarcado el proyecto: videojuegos, agentes software, motores de juegos y la tuberculosis.

### **A.1. Videojuegos**

Los videojuegos son un medio de entretenimiento que ha ido abriendo camino en el mercado desde finales de los años 70 convirtiéndose en una de las más grandes y solventes industrias en el mundo del entretenimiento que ha crecido a la par con la evolución del software y los avances tecnológicos en materia del hardware [14]. Hoy en día existe una gran variedad de videojuegos, por lo que se vio la necesidad de crear una forma de clasificación para éstos de manera que pudieran ser agrupados a partir de distintos ejes como pueden ser: la temática, el argumento, la plataforma, entre otros [15]. Un género importante en el mundo de los videojuegos es el de aventura, ya que este puede ser visto como otro tipo de medio narrativo como los libros o las películas en los cuales el jugador adopta el papel de protagonista de una historia, viéndose obligado a sumergirse en el argumento de ésta, para lograr alcanzar los objetivos del juego [16]. En los juegos de aventura la historia forma parte de los elementos más importantes; sin embargo, para el filósofo Jung o Freud no podría existir una historia sino existen los personajes ya que una historia es una narración de los sucesos que ocurren a un determinado personaje en la búsqueda de una determinada solución simplificando un poco la definición [17]. Es por esto que para la creación de juegos de aventura debe existir un cuidado especial al crear los personajes sin dejar de lado los demás elementos indispensables para la creación de un videojuego como el diseño de los ambientes, la visualización de terrenos, los efectos de sonidos, el diseño de niveles entre otros. Con respecto al área de la programación es importante tener en cuenta el motor de juegos, considerado como el soporte del juego, cuando se hace referencia al tema de la codificación [18]. En un motor de juegos es posible encontrar una gran cantidad de código que puede ser reutilizado facilitando la definición de sonidos, gráficos y otras características del juego



## A.2.1. Historia de los Videojuegos

La prehistoria de los videojuegos se remonta a 1958. Bill Nighinbotthan presenta en una feria científica un aparato que permite, mediante unos potenciómetros, mover una pequeña raqueta en un tubo de rayos catódicos [15]. Basándose en este aparato, Nolan Bushnell (fundador de Atari) crea en 1972 un videojuego llamado Pong. Años más tarde, en 1976, un empleado de Atari llamado Steve Jobs, ayudado por Steve Wozniak, crean un videojuego llamado BreakOut. Dos años después, ambos dejarán Atari para crear Apple Computer y pasar a la historia de la informática [15]. Después en 1978, Taito lanza al mercado el famoso Space Invaders. Este juego era capaz de almacenar las puntuaciones máximas, convirtiéndose en todo un clásico. En los siguientes años, comienzan a aparecer en el mercado nuevos videojuegos de excelente calidad como Donkey Kong, Frogger, Galaga, Pac Man, etc [15].

En los tiempos del MS-DOS la programación de videojuegos era bastante compleja, hacían falta grandes conocimientos de lenguaje ensamblador y de la estructura interna de la máquina, así como de la tarjeta de video y de sonido. Toda la programación se hacía directamente sobre el hardware [15]. Es fácil imaginar los inconvenientes, había que preparar el juego para que funcionara con muchos modelos de tarjeta gráfica y también para los diferentes modelos de tarjetas de sonido (Adlib, Gravis Ultrasound (GUS), Sound Blaster). De esta forma hacer un juego, por simple que éste fuera, requería un esfuerzo muy alto, sin contar con que no siempre toda la información necesaria para acceder al hardware estaba al alcance de todo el mundo [15].

## A.2.2. Tipos de videojuegos

En la actualidad existe una gran variedad de videojuegos, dentro de los cuales muchos comparten características fundamentalmente relacionadas con la temática principal, el argumento, la estructura del juego, la plataforma, las habilidades lúdicas perseguidas [19]. Los videojuegos que combinan estas características, reciben el nombre de híbridos. A continuación se presenta una clasificación de los videojuegos con base en una propuesta hecha por el profesor Pere Marques Graells de la Universidad Autónoma de Barcelona en la cual se identifican seis tipos de videojuegos [20]:

- **Arcades:** Los videojuegos de este tipo son definidos como juegos de animación que requieren mucha rapidez de reacción. El personaje central debe ir superando los distintos escenarios en los cuales el grado de dificultad incrementa. Es conveniente plantear una subdivisión en dos tipos de arcade diferentes: los juegos de plataformas y los juegos de acción, estos últimos también pueden ser subdivididos en *First Person Shooter* (FPS) y juegos de acción en tercera persona [20].  
En los juegos de **plataformas** el personaje debe abrirse camino a través de los diversos escenarios atravesando obstáculos, saltando y eliminando o esquivando a aquellos “obstáculos” que impiden el progreso. A medida que el juego avanza el nivel de dificultad aumenta, lo cual suele ser reflejado en un aumento en la velocidad y en los obstáculos,



por lo que el jugador debe cada vez reaccionar de forma más rápida, por ejemplo Mario Bross, Prince Of Persia, etc. [20]

Los **FPS** son videojuegos de acción en primera persona en donde el usuario se integra con el personaje del juego. Lo que aparece en la pantalla o televisor es la mano del personaje portando el arma que lleve en ese momento. Estos juegos son de gran variedad temática y en la actualidad suelen ser desarrollados basados en un guión o historia cinematográfica. La base del desarrollo del juego es la supervivencia a lo largo de los diferentes niveles neutralizando a cualquier enemigo que se interponga en el camino. Una de las características que determinan la calidad de este tipo de videojuego es la interacción con los escenarios. Cuantos más elementos del escenario puedan ser manipulados o destruidos por el personaje, mejor y más completo es considerado el juego, por ejemplo DOOM, Quake, etc. [20]

Los juegos de **acción en tercera persona** se diferencian de los anteriores en la forma de integración entre el jugador y el personaje. En este caso el jugador no alcanza un grado muy alto de identificación con el personaje. Debido a que el usuario tiene una visión completa del personaje en la pantalla, lo que le permite crear una concepción ajena sobre el actor. Juegos en los que la acción se ve relegada por la estrategia y el sigilo. Resultan algo más lentos de jugar y con una historia o guión más elaborados. Contienen un alto componente de plataforma, como Tomb Raider, Metal Gear Solid, etc. [20]

- **Aventura y Rol [20]:** Aunque el profesor Marqués en esta clasificación los unifica dentro de la misma categoría, existen algunas diferencias entre estos dos tipos de juegos tanto en jugabilidad como en temática y la forma en se relaciona el jugador con el videojuego. En los juegos de aventura el jugador protagoniza una historia en la que deben superarse pruebas, adquirir habilidades y descubrir paso a paso nuevos elementos hasta llegar al final del argumento. Los juegos de aventura son un tipo de programas de entretenimiento para computador, los cuales se caracterizan por introducir al jugador en mundo donde la investigación es la base del juego, y para alcanzar el objetivo del juego deberá explorar en su entorno, resolver puzzles (rompecabezas), interactuar con los demás personajes del juego y sobre todo estar muy enfocado en la narrativa (historia) la cual refleja los desafíos del juego. Es de notar que el término Juegos de Aventura no tiene una relación directa con los *films* o las novelas de aventura, por lo que no es indicativo del tema o contenido del juego. La mayoría de los juegos de aventura son juegos de computador, y aunque existen juegos de aventura para consola, no es común oír de ellos. Los juegos de aventura se enfocan en una historia, lo que les permite estar basados en distintos medios narrativos, tales como la literatura y las películas. Los juegos de aventura abarcan una gran cantidad de géneros literarios, entre ellos la fantasía, ciencia ficción, misterio, horror y comedia. Dentro de los títulos más nombrados en este género se encuentran Zork, King's Quest, The Secret of Monkey Island, Gabriel Knight y Myst. Casi todos los videojuegos de aventura son diseñados para un jugador, dado que se hace tanto énfasis en la historia resulta realmente difícil diseñarlos de forma que se pueda jugarse en multi-jugador. Este género alcanzó su máxima popularidad a finales de los años 80 y comienzos de los 90, además considerado por muchos técnicamente una de las clasificaciones más avanzadas en materia de videojuegos.



En los juegos de Rol o RPG (Rol Player Game) las acciones y decisiones son tomadas en función de las propuestas del juego y de las acciones del resto de personajes. Se mantiene en todo momento un control exhaustivo de las características del personaje como las constantes vitales, habilidades, armas y poderes entre otras. Son juegos que por lo general manejan una temática mágica, y anacrónica. Su diseño propone una jugabilidad lenta. Una de sus principales diferencias con los juegos de acción es que sus luchas se desarrollan por turnos lo que exige al jugador dedicar más tiempo a la planificación de una buena estrategia de ataque que a la acción en sí, como Final Fantasy.

- **Simuladores** [20]: Estos juegos se caracterizan por su gran veracidad y realismo sobre todo al tratarse de entornos y situaciones reales. Esto obliga al usuario a aplicar en ocasiones conocimientos específicos. El grado de integración del jugador en el entorno es variable, ya que la visión puede ser en tercera o en primera persona. Es común encontrar de todo tipo de simuladores desde aviones comerciales, de combate, autos, motos y muchos más. En la actualidad muchos simuladores se usan como mecanismo de preparación para distintas situaciones reales.
- **Estrategia** [20]: En este tipo de juegos se promueve la toma de decisiones, cálculos, razonamiento lógico, establecimiento de hipótesis, para conseguir un objetivo. Suelen ser de tema histórico y épico donde se recrean batallas entre distintas poblaciones, El jugador debe, a partir de unos pocos elementos de construcción y personajes, ir construyendo las estructuras necesarias para crear sus ejércitos y de esta manera enfrentarse a otros ejércitos. Existen dos tipos claramente diferenciados de juegos de estrategia: en tiempo real y por turnos. En los juegos en tiempo real todos los jugadores realizan sus acciones al mismo tiempo, por ejemplo Age Of Empires, StarCraft, etc. Mientras que en los juegos por turnos los jugadores van realizando sus diferentes movimientos uno detrás de otro, como en Civilization y Populous.
- **Deportes** [20]: Los juegos de deportes siempre han tenido un gran éxito comercial. Estos videojuegos han evolucionado de basarse en el enfrentamiento entre dos equipos donde un partido decide el ganador, hasta los juegos actuales en donde el usuario puede gestionar todo un equipo como si fuera un presidente de un club deportivo donde tiene la posibilidad de tomar decisiones sobre inversiones, fichajes, traspasos y cesiones del mismo. Incluso algunos de estos juegos han llegado a limitar la posibilidad de que el jugador participe en los partidos, dejando al computador o a la consola que calcule el resultado de un partido en función de cómo lo ha planteado tácticamente el usuario. De esta forma los juegos de deportes han encontrado una dualidad en su jugabilidad. Un tipo se basa en la gestión y organización de un club deportivo siendo más tranquilo y lento de jugar, y otros se ha dirigido hacia la simulación de partidos en donde el jugador participa activamente en los encuentros. Algunos títulos de gran éxito son *PC Futbol*, *Pro Evolution Soccer*, *NBA*.
- **Puzzles y preguntas** [20]: Este tipo de juegos estuvieron en auge durante un largo tiempo ya que sus necesidades graficas eran bajas lo que permitía a un gran número de usuarios acceder a ellos. Además, el mercado de los videojuegos no tenía aún el potencial que tiene



actualmente. Pero con el paso del tiempo el público hacia el cual se enfocan estos videojuegos se ha ido reduciendo. Es preciso tener en cuenta, que estos videojuegos fueron los precursores de la industria de los videojuegos con títulos como *Tetris*, *Tenis For Two* o *Comecocos*, y aun siguen siendo utilizados en dispositivos de bajas capacidades graficas como dispositivos móviles o consolas de mano.

### A.2.3. Personajes

Autores como William Archer opinan [21], que la creación de personajes no puede ser regulada por recomendaciones teóricas. A pesar de que en la actualidad es muy común encontrar una gran cantidad de manuales y guías para escribir historias, la creatividad no debe ser atada a un conjunto de reglas o pasos. Y no es lo que pretende este proyecto, dado que la idea es dar un enfoque “mostrar un camino a seguir” lo que no quiere decir que esta sea la única forma, es solo una opción para crear personajes en un videojuego de aventura [21].

En este proyecto se trabajara con personajes en tres dimensiones (3D), y generalmente se asocia este concepto exclusivamente a la parte física del personaje en lo que se refiere a la altura, anchura y profundidad. El término de tres dimensiones también es empleado por algunos autores para ofrecer una descripción detallada del personaje determinando su carácter físico, psicológico y sociológico, esta definición es aplicada a todos los personajes de un juego ya sea o no el personaje jugador [17].

- **Fisiología del personaje:** En esta dimensión se cubren los aspectos físicos detallando la apariencia y estado de salud del personaje. La apariencia física del personaje puede hacerlo ver y sentir seguro o inseguro en sus actividades, una persona que se encuentre sana reacciona de forma distinta a como reacciona una persona enferma ante la presencia de situaciones similares. Su estado salud genera una diferencia en su actitud hacia la vida. Ambos, la apariencia y la salud afectan su desarrollo mental, lo que, suele crear un complejo de superioridad o inferioridad en el personaje [17][21].
- **Sociología del personaje:** En esta dimensión se trata al personaje analizando su entorno. El comportamiento de las personas varía dependiendo de las características en las que se desarrolle su vida. El entorno de un personaje significa la vida en el hogar, el estado civil de los padres, la capacidad de obtención de ingresos, si es divorciado, viudo, compatible o incompatible. ¿De qué manera el personaje afecta a sus amigos, y cómo le afectan a él?, ¿Cuál es su actitud y aptitud en la escuela, sus materias favoritas?, etc. [17][21]
- **Psicológica del personaje:** Esta dimensión es el resultado de las dos dimensiones anteriores. Aquí se da vida a las aspiraciones, frustraciones, temperamentos, actitudes, y complejos de los personajes. Para entender la acción de un personaje, debemos primero encontrar su motivación. Si se tiene un conocimiento íntimo en la formación en estos tres ámbitos puede obtenerse una idea de la motivación del personaje [17][21].



El grado de profundización que se debe alcanzar en estas dimensiones depende de la importancia del personaje en la historia, no es necesario ser tan detallista en un personaje que solo se presentará en una o dos escenas, mientras que el protagonista y actores principales deben tener toda la atención necesaria [17].

Existen diferentes tipos de personajes en un videojuego:

- **Personaje jugador:** Es el más importante del juego debido a que es el protagonista, es el más complicado y desafiante al momento de escribir la historia. Un personaje jugador o PJ (PC en inglés) es un personaje creado para ser controlado por el jugador, además, son los únicos con la capacidad de tomar decisiones según el criterio del jugador, en torno a ellos gira toda la acción del juego. Los PJ se crean a partir de los parámetros que los definen durante el transcurso del juego, estos van evolucionando, aprendiendo y obteniendo poderes (mientras que un personaje no jugador, al aparecer tan solo en un determinado momento del juego, no evoluciona a lo largo del tiempo). Los PJ tiene la posibilidad de tomar diferentes caminos que le llevarán hasta distintas metas, ésta es una de sus principales características, razón por la cual, los jugadores suelen acostumbrarse a cierto tipo de personaje, y al cambiar de juegos la mayoría tiende a elegir personajes de la misma raza o, al menos, de comportamientos o actitudes similares a las de aquellos con los que ya ha jugado [21].
- **Personaje no jugador:** Son los personajes controlados por el servidor o el computador. Conocidos comúnmente como NPC (Non Player Characters), los cuales tienen una misión específica dentro de la historia del juego, por ejemplo, es el mago que da respuestas para enigmas que se presentan en medio del juego, el rey que requiere de los servicios del protagonista, el aliado que conduce al protagonista por un gran misterio, y, por último puede ser el villano de la historia al que el protagonista tarde o temprano derrotará [21].
- **Personaje pivote:** Es el personaje que lanza la acción entre el protagonista y el antagonista, es quien pone a la historia en movimiento porque crea el conflicto y hace que el juego cobre vida [21].

Cada uno de los tipos de personajes anteriores puede representar un papel que será de gran importancia para el desarrollo de la historia. Entre estos papeles se encuentran:

- **El Héroe:** Es el personaje con el que más se identifica las personas cuando leen, ven o juegan una historia. Normalmente suele ser porque el héroe es el protagonista, el más importante de la historia debido a que, esta se centra en los sucesos que le ocurren a este personaje [22].
- **El Tutor del héroe:** Todo héroe, sin importar las condiciones sociales y culturales en las que se encuentre, siempre va a encontrar en su camino a una figura sabia, anciana y que le enseñará cómo superar las mayores dificultades [22].



- **La Princesa:** La princesa es aquella figura que complementa al héroe considerada también la fuerza que motiva al héroe a superar todas las pruebas que se le interpongan es común ligar a estos personajes en una relación sentimental, dejando una señal de romanticismo en la historia [22].
- **El Villano:** El MAL (con mayúsculas) es el comienzo de toda buena historia. El elemento maligno, el personaje contrario al héroe, que cambia el mundo del protagonista y tiene que luchar contra él [22].

## A.3. Agentes Software

### A.3.1. ¿Qué es un Agente?

Aunque no hay una definición unificada en cuanto a qué es un agente, un intento de unificar los esfuerzos para el desarrollo de esta tecnología puede encontrarse en **FIPA** (*Foundation for Intelligent Physical Agents*) [23] donde se define a los agentes como una entidad de software con un grupo de propiedades entre las que se destacan el ser capaz de actuar autónomamente en un ambiente, comunicarse directamente con otros agentes, estar condicionado por un conjunto de tendencias u objetivos, manejar recursos propios, ser capaz de percibir su ambiente y tomar de él una representación parcial, además, ser una entidad que posee habilidad y ofrece servicios, que puede reproducirse, etc. [23] De forma general, varios autores reconocen en los agentes diversas propiedades, entre las que se destacan el ser autónomos, reactivos, pro-activos y tener habilidad social [25].

**El Agente AIMA** “Un agente es cualquier cosa que pueda ser vista percibiendo su entorno a través de sensores y actuado hacia el entorno a través de unos efectores.”

**El Agente Maes** “Los agentes autónomos son sistemas computacionales que habitan en entornos dinámicos complejos, percibiendo y actuando autónomamente en ese entorno, y realizan un conjunto de metas o tareas para las que han sido diseñados.”

**El Agente KidSim** “Definiremos un agente como una entidad software persistente dedicada a un propósito específico. La ‘persistencia’ distingue a los agentes de las subrutinas los agentes tienen sus propias ideas sobre cómo ejecutar tareas, sobre sus agendas. Con ‘propósito específico’ se distinguen los agentes de las aplicaciones multifunción, que son típicamente más pequeños.”

**El Agente Hayes-Roth** “Los agentes inteligentes realizan continuamente tres funciones: percibir condiciones dinámicas en el entorno, actuar afectando a las condiciones del entorno, y razonar para interpretar lo percibido; resuelven problemas, muestran interfaces y determinan acciones.”

**El Agente IBM** “Los agentes inteligentes son entidades software que llevan a cabo un conjunto de operaciones en beneficio de un usuario u otro programa con algún grado de independencia o



autonomía, y haciendo esto, emplean algún conocimiento o representación de las metas y deseos del usuario.”

**El Agente Wooldridge y Jennings** “Es un hardware o más comúnmente un sistema software basado en computador que disfruta de las siguientes propiedades:

**Autonomía:** los agentes operan sin la intervención directa de personas u otros, y tienen algún tipo de control sobre sus actuaciones y estado interno.

**Habilidad social:** los agentes interactúan con otros agentes (posiblemente humanos) vía algún tipo de lenguaje de comunicación de agentes [24].

**Reactividad:** los agentes perciben el entorno o ambiente, (lo que representa la palabra físicamente, un usuario vía una interfaz de usuario, una colección de otros agentes, Internet, o quizás todos ellos combinados), y responde rápidamente a cambios que ocurren en dicho entorno. [24]

**Pro-actividad:** los agentes no actúan simplemente en respuesta a su entorno, sino que son capaces de exhibir ‘comportamiento dirigido hacia el objetivo’, tomando la iniciativa” con ligeras modificaciones de enfoques, otros autores también reconocen estas propiedades [24].

Franklin, después de estudiar doce definiciones, llegó a la conclusión que los agentes tienen entre sus propiedades principales la autonomía, estar orientados a metas, ser colaborativos, flexibles, autoiniciables, con continuidad temporal, carácter, comunicativos, adaptativos y móviles [25]. Agrega que un agente autónomo es un sistema situado en un ambiente que percibe el ambiente y actúa sobre él, en el tiempo, según su agenda propia y de esta manera produce efectos en lo que él mismo podrá sentir en el futuro [25].

Russel y Norvig, tienen una visión más flexible de los agentes, donde podrían considerarse herramienta para analizar sistemas y no como una característica abstracta que divida al mundo en agentes y no-agentes [24].

### A.3.2. Propiedades de los Agentes

- **Inteligencia:** Es difícil de definir con precisión. Cuando se aplica a los agentes, se podría definir la inteligencia como algunas capacidades de alto nivel, es decir, la capacidad del agente para razonar sobre su conocimiento, aprender e inferir nuevos conocimientos y planificar las acciones consecuentes para alcanzar sus objetivos. En este caso, la inteligencia tiene el mismo sentido que el término "inteligencia artificial" [24] [26].
- **Autonomía:** Existen diferentes formas de entender el término "autonomía". En el sentido más común (absoluto), la autonomía se refiere a lo impredecible del comportamiento del agente: cuanto más impredecible sea, más autonomía aparecerá [26]. Sin embargo, la autonomía absoluta no se aplica muy bien a los agentes ya que, de acuerdo con la mayoría



de definiciones, un agente debe servir para algunos propósitos que lo limitan. Un tipo más útil de autonomía es la "autonomía social", que se refiere a la autonomía del agente cuando es analizado en una sociedad de agentes. No obstante, las relaciones sociales más comunes como la cooperación, coordinación y el compromiso, limitan al agente. Finalmente, la ejecución de la autonomía se refiere a la capacidad del agente para elegir y llevar a cabo la acción adecuada en el momento oportuno (técnicas de planificación). De hecho, al diseñar un agente, se tiene que encontrar un fino equilibrio entre autonomía y servidumbre. Un nivel inferior puede consistir en un agente que represente a un usuario, como el bien conocido PDA (Asistente Personal Digital) [24] [26].

- **Racionalidad:** Se aplica esencialmente a los agentes autónomos e inteligentes. Los agentes racionales tienden a elegir y realizar acciones que maximicen su esperada utilidad en función de sus percepciones y de nuestro propio conocimiento. La racionalidad también implica que la acción elegida sea consistente con los deseos y creencias del agente [24] [26].
- **Movilidad:** Es la capacidad que tiene un agente para iniciar su ejecución, y desplazarse a otro entorno (llevando consigo datos y códigos) para continuar su ejecución[26]. El término "movilidad" dividió a la comunidad de agentes en dos escuelas: la de los que argumentan que la movilidad no es un aspecto esencial (la comunidad de multi-agentes DAI, esencialmente académicos) y la de los que afirman que los agentes móviles son el futuro de los agentes (comunidad de programadores orientados a objetos) [26]. La programación basada en agentes móviles genera un nuevo paradigma que realza la flexibilidad y eficacia del diseño y ejecución de las aplicaciones distribuidas, y reduce el ancho de banda de red requerido [24].
- **Comunicación:** Es un aspecto crucial para un agente, que, por definición, tiene que comunicarse durante su vida con la entidad en nombre de la que actúa, y con los otros agentes con los cuales necesita colaborar[26]. El nivel de interacción/comunicación depende en gran medida del nivel de conocimiento del agente, que se desenvuelve entre "datos", "información" y "conocimiento". Los datos se corresponden con el nivel más básico, que puede ser el contenido de una variable, un archivo, etc. La información está mucho más estructurada y puede consistir en la descripción de parte de un equipo de telecomunicación, un documento XML (eXtensible Markup Language) electrónico), o un correo electrónico (con datos asociados) [26]. El conocimiento consiste en información estructurada y reglas lógicas. Se pueden asociar con cada nivel de información los lenguajes de comunicación y protocolos adecuados. Por ejemplo, un agente puede usar FTP (Protocolo de Transferencia de Archivos) para enviar un archivo, o llamar al método de otro agente para obtener o transmitir información. Igualmente, el HTTP (HyperText Transfer Protocol) se puede utilizar para transmitir descripciones XML [24].
- **Sociabilidad:** los agentes son capaces de interactuar con otros agentes (humanos o no) a través de un lenguaje de comunicación entre agentes [26].
- **Reactividad:** los agentes son capaces de percibir estímulos de su entorno y reaccionar a dichos estímulos [26].



- **Proactividad, iniciativa:** los agentes no son sólo entidades que reaccionan a un estímulo, sino que tienen un carácter emprendedor, y pueden actuar guiados por sus objetivos[26].
- **Veracidad:** asunción de que un agente no comunica información falsa a propósito [26].
- **Benevolencia:** asunción de que un agente está dispuesto a ayudar a otros agentes si esto no entra en conflicto con sus propios objetivos [26].

Finalmente, se puede usar lenguajes para intercambiar información entre agentes que se encuentren basados en la Speech Acts Theory, como KQML/KIF (Knowledge Query and Manipulation Language/Knowledge Interchange Format) [KQML, KIF], o el lenguaje de comunicación de agentes FIPA (Formation for Intelligent Physical Agents) pueden ser usados para intercambiar información[26].

### A.3.3. Metodologías de Agentes

Las técnicas convencionales de ingeniería de software (Proceso Unificado), no tienen en cuenta las necesidades de especificación de los Sistemas Multi-agente, como la especificación de planificación de tareas, intercambio de información con lenguajes de comunicación orientados a agentes, movilidad del código o motivación de los componentes del sistema. Por esta razón, se plantean nuevas metodologías basadas en agentes tales como: BDI, Vowel Engineering, MAS-CommonKADS, GAIA. Estas metodologías parten de un modelo informal en la mayoría de casos, de cómo debe ser un Sistema Multi-agente y dan guías para su construcción. Las primeras metodologías, consistían en una lista breve de pasos a seguir. En la actualidad, aunque es posible encontrar varios progresos en el intento de integrar metodologías orientadas a agentes con la ingeniería del software clásica, aún no se alcanza la madurez que se puede encontrar en metodologías convencionales como el Proceso Unificado.

El motivo principal para esta falencia radica en que todavía no existen herramientas genéricas de soporte y un lenguaje para la especificación de los Sistemas Multi-Agentes que permitan trabajar de forma similar a como se trabaja en Rational Rose o Power Designer. Algunas de las metodologías más usadas actualmente son:

#### A.3.3.1. *MaSE*

(Multi-agent systems Software Engineering) es una metodología basada en el ciclo de vida clásico del software, con un entorno propio de desarrollo (agentTool) el cual permite el análisis, diseño y construcción sistemas multi-agente heterogéneos[28]. Se concibe como una abstracción del paradigma orientado a objetos donde los agentes son especializaciones de objetos, en lugar de simples objetos, con métodos que pueden invocarse desde otros objetos, los agentes se coordinan unos con otros vía conversaciones, pero no son considerados entes autónomos y sociales, sino simples procesos que se comunican para alcanzar metas individuales y del sistema. En MaSE los agentes son simplemente una abstracción conveniente, que puede o no poseer inteligencia. En este sentido, los componentes inteligentes y no inteligentes se gestionan igualmente dentro de la



misma estructura[28]. Dado el enfoque inicial, los agentes se ven como especializaciones de objetos. MaSE se divide en dos fases principales las que a su vez se dividen en diferentes pasos los cuales son necesarios para su elaboración.

- **Análisis:**

*Capturar los objetivos:* En este paso se busca como tareas principales la identificación y estructuración de los objetivos principales del sistema, dejado de lado por el momento las actividades o desarrollos que permiten alcanzar dichos objetivos centrándose en su análisis y estructuración según su importancia[28].

*Aplicar los casos de uso:* En éste paso se hace necesario definir los casos de uso para explicar el comportamiento deseado del sistema. Luego se procede a, estructurar los casos de uso en diagramas de secuencia mostrando la secuencia de eventos entre roles y, como resultado, define la mínima comunicación necesaria entre ellos[28].

*Refinar roles:* Es necesario asegurarse de que todos los objetivos son tenidos en cuenta asignando un rol a cada uno, siendo desempeñado este último por al menos un agente en el diseño final. Pese a que, generalmente, cada objetivo es mapeado a un rol individual, existen situaciones en las que conviene asignar más de un objetivo a un mismo rol, por motivos de eficiencia. Este tipo de decisiones son basadas en conceptos clásicos de ingeniería del software como la cohesión funcional, comunicacional, procedural o temporal. También podría ser por distribución de recursos o por cuestiones especiales de las interfaces. Finalmente, los roles son captados en el modelo de roles. Una vez obtenidos los roles, se procede a convertirlos en clases de agentes con su comportamiento interno y sus conversaciones entre ellos, este paso resulta ser uno de los más complejos de esta metodología. Para conseguir esto, se deben definir tareas de alto nivel que puedan ser transformadas en funcionalidades específicas de agentes. Estas funcionalidades nos ayudan a definir los componentes internos de los agentes y los detalles de las conversaciones en que participan. Todo esto queda reflejado en el modelo de roles detallado[28].

- **Diseño**

*Crear clases de agentes:* Las clases de agentes son identificadas a partir de los roles y descritas en el diagrama de clases de agentes. Nuevamente, lo más común es establecer una correspondencia uno a uno entre roles y clases de agente. Sin embargo, el diseñador puede decidir unificar varios roles en una clase o crear varias clases de un mismo rol[28]. Una vez creado el diagrama de clases, la organización del sistema queda definida. En este punto es bueno fusionar roles que tengan en común un gran volumen de tráfico de mensajes para optimizar el sistema[28].

*Construir conversaciones:* Este paso se encuentra íntimamente ligado con el siguiente (Ensamblaje), específicamente una conversación consiste en dos diagramas de clases de comunicación, uno para el emisor y otro para el receptor. Un diagrama de clases de



comunicación es un par de máquinas de estados finitos que definen una conversación entre dos clases agente participantes[28].

*Ensamblar clases de agentes:* En este paso se crea el interior de los agentes. Este proceso se simplifica usando un lenguaje de modelado arquitectónico que combina la naturaleza abstracta de los lenguajes tradicionales de descripción arquitectónica con el lenguaje de restricción de objetos, que permite al diseñador especificar detalles de bajo nivel[28].

*Diseño del sistema:* En esta fase se decide la configuración final del sistema a ser implementado. Hasta ahora en MaSE sólo se consideran sistemas estáticos, los móviles aun no es posible modelarlos. En MaSE se define la arquitectura global del sistema mediante diagramas de despliegue para mostrar el número, tipo y localización de los agentes. En este momento también se toman las decisiones de implementación que no se habían tomado antes, como el lenguaje de programación a usar o el “framework” para las comunicaciones[28].

La herramienta agentTool es el software de soporte que actualmente implementa los siete pasos del proceso MaSE con soporte para transformar modelos de análisis en modelos de diseño. Permite crear de modo visual todos los diagramas del proceso de desarrollo, pudiendo realizar cualquiera de los 7 pasos en el orden que se prefiera (siempre respetando las dependencias entre unos y otros). Además, dispone de una base de conocimiento persistente, un verificador de conversaciones y generación automática de código.

Los estudios recientes sobre MaSE y agentTool se concentran en dar soporte a agentes móviles. El paso inicial para dar este soporte es el de incluir una actividad de movimiento que, básicamente, pide que el agente cambie de lugar. El proceso retorna verdadero o falso, que indica si el agente se ha movido o no. Este añadido a la fase de análisis permite al analista especificar cuándo debe ocurrir un movimiento, su lugar de destino, y saber si se ha podido realizar la operación[28].

### **A.3.3.2. GAIA**

GAIA es una metodología para el diseño de sistemas basados en agentes cuyo objetivo es obtener un sistema que maximice alguna medida de calidad global (no se llega a detallar cuál). GAIA pretende ayudar al analista a ir sistemáticamente desde unos requisitos iniciales a un diseño que esté lo suficientemente detallado como para ser implementado directamente[29]. En GAIA se entiende que el objetivo del análisis es conseguir comprender el sistema y su estructura sin referenciar ningún aspecto de implementación. Esto se consigue a través de la idea de organización. Una organización en GAIA es una colección de roles, los cuales mantienen ciertas relaciones con otros y toman parte en patrones institucionalizados de interacción con otros roles. Los roles agrupan cuatro aspectos: responsabilidades del agente, los recursos que se le permite utilizar, las tareas asociadas e interacciones[29].



- **Análisis**

GAIA propone trabajar inicialmente con un análisis a alto nivel. El objetivo de este análisis consiste en definir los elementos abstractos que conformarán el sistema, es decir, entender el sistema y su estructura sin entrar en detalles de implementación. En esta parte se obtendrá la visión de la organización como un conjunto de roles que tienen relación unos con otros, así como del entorno en el que esta embebido el sistema[29]. En este análisis se usan dos modelos:

*Modelo de roles:* Se utiliza para identificar los roles clave en el sistema junto con sus propiedades definitorias.

*Modelo de interacciones:* Define las interacciones mediante una referencia a un modelo institucionalizado de intercambio de mensajes, como el FIPA-Request.

- **Diseño**

El diseño se puede subdividir en dos subfases: diseño de la arquitectura, que en gran medida corresponde a la fase abstracta del método y una diseño detallado, donde pasaremos de la abstracción a la concreción, obteniendo modelos de bajo nivel fácilmente implementables como[29]:

*Modelo de agentes:* define los tipos de agente que existen, cuántas instancias de cada tipo y qué papeles juega cada agente.

*Modelo de servicios:* identifica los servicios (funciones del agente) asociados a cada rol.

*Modelo de conocidos:* define los enlaces de comunicaciones que existen entre los agentes.

A partir de aquí, los autores de GAIA proponen aplicar técnicas clásicas de diseño orientado a objetos. Sin embargo, GAIA declara que queda fuera de su ámbito. Esta metodología sólo busca especificar cómo una sociedad de agentes colabora para alcanzar los objetivos del sistema, y qué se Requiere de cada uno para lograr esto último[29].

### **A.3.3.3. MESSAGE**

MESSAGE trata de integrar resultados de las anteriores metodologías. Propone el análisis y diseño del SMA desde cinco puntos de vista para capturar los diferentes aspectos de un SMA[32]:

*Organización:* que captura la estructura global del sistema.

*Tareas/Objetivos:* que determina qué hace el SMA y sus agentes constituyentes en términos de los objetivos que persiguen y las tareas implicadas en el proceso.

*Agente:* que contiene una descripción detallada y extensa de cada agente y rol dentro del SMA.



*Dominio*: que actúa como repositorio de información (para entidades y relaciones) concernientes al dominio del problema; y el de Interacción, que trata las interacciones a distintos niveles de abstracción.

Estos elementos están presentes en los dos modelos fundamentales que propone MESSAGE: el modelo de análisis y el modelo de diseño. El modelo de análisis se limita a generar modelos a partir de los meta-modelos. El modelo de diseño de esta metodología no llegó a concretarse completamente. Se decidió que el propósito del diseño sería producir entidades computacionales que representen el SMA descrito en el análisis. Por ello, cada artefacto producido en el análisis debería transformarse en una entidad computacional o varias cuyo comportamiento fuera el que se esperaba en el análisis. Esto significa que las entidades del análisis se deberían traducir a subsistemas, interfaces, clases, signaturas de operaciones, algoritmos, objetos, diagramas de objetos y otros[32].

MESSAGE aporta mejoras en cuanto a conceptos de ingeniería respecto de las alternativas existentes, entre ellas el desarrollo dentro de un paradigma de ingeniería del software (el Proceso Unificado), aportación de métodos para la traducción de entidades de análisis a entidades de diseño y guías para la generación de los modelos. Sin embargo, los objetivos de MESSAGE no se completaron totalmente. La integración con el Proceso Unificado no fue total, ya que las actividades definidas no se adecuaban a las necesidades reales y no se indicó cómo encajaban dentro de este proceso. Además, faltó trabajo en el estudio de las interdependencias entre los distintos modelos propuestos[32].

#### **A.3.3.4. INGENIAS**

Es una metodología creada a partir de la evolución de las ideas de MESSAGE. INGENIAS profundiza en los elementos mostrados en el método de especificación, en el proceso de desarrollo, además de incorporar nuevas herramientas de soporte y ejemplos de desarrollo[30]. INGENIAS, como MESSAGE, define un conjunto de meta-modelos (una descripción de alto nivel de qué elementos tiene un modelo) con los que se debe describir el sistema. Los meta-modelos indican qué falta por describir: agentes aislados, organizaciones de agentes, el entorno, interacciones entre agentes o roles, tareas y objetivos. Estos metamodelos se construyen mediante un lenguaje de meta-modelado, el GOPRR (Graph, Object, Property, Relationship, and Role). En la construcción de estos meta-modelos se integran resultados de investigación en forma de entidades y relaciones entre entidades[30].

La instanciación de estos meta-modelos produce diagramas, los modelos, similares a los que se usa en UML, con la diferencia de que estos diagramas se han creado exclusivamente para definir el sistema multi-agente. El proceso de instanciación de los meta-modelos no es trivial[30]. Existen muchas entidades y relaciones a identificar, además de dependencias entre distintos modelos. Por ello, INGENIAS define un conjunto de actividades cuya ejecución termina en un conjunto de modelos. Estas actividades a su vez se organizan siguiendo un paradigma de ingeniería del software, el Proceso Unificado[30]. La ejecución de actividades para producir modelos se basa en la herramienta INGENIAS IDE, una herramienta para modelado visual. Esta herramienta almacena



la especificación del sistema utilizando XML. Desde la especificación en XML, se plantea el procesarla[30]:

- Para generar código. Genera código relleno de plantillas de código con información de la especificación.
- Para generar la documentación. De una forma similar a la generación de código, se parte de plantillas de documentación que se completan utilizando información de los modelos. Tanto la herramienta de análisis/diseño como el editor se pueden encontrar en <http://ingenias.sourceforge.net>. Desde esta web también se tiene acceso a ejemplos de especificación siguiendo la metodología.

#### **A.3.3.5. MAS-CommonKADS**

La metodología MAS-CommonKADS está basada en CommonKADS, aportando una serie de modelos para desarrollar las fases de análisis y de diseño de sistemas multiagente. La principal característica es la incorporación de técnicas orientadas a objetos a CommonKADS, la cual es tomada como eje fundamental a lo largo de todo el proceso[33]. En MAS-CommonKADS se proponen diferentes puntos de vista del sistema a analizar, lo cual se traduce en diferentes fases, que a continuación se comentan brevemente.

##### **Conceptuación**

En la fase de conceptualización se analiza el sistema desde el punto de vista del usuario del sistema. Es una fase un tanto informal, donde se propone un análisis centrado en el usuario para la captura de requisitos. El empleo de los casos de uso es la técnica por la que se apuesta para la realización de esta fase.

##### **Organización**

El modelo de organización analiza, por una parte, la empresa (organización) en que va a ser introducido el sistema, y la posible repercusión del sistema en la empresa. Por otra parte, el modelo de organización sirve para describir las relaciones de la sociedad multiagente entre sí y para con su entorno.

##### **Agente**

El modelo de agente describe las propiedades y características de cada agente. Es una de las fases más importantes, consta de una subfase de identificación de agentes (con diferentes opciones) y una subfase de descripción detallada de la que debe realizar cada agente.

##### **Tareas**

El modelo de tareas describe las funciones (o tareas) que desarrolla el sistema, en realidad son actividades relacionadas para alcanzar un objetivo. Este modelo se refina según la naturaleza de las tareas (cognitivas, comunicación hombre-máquina, comunicación entre agentes) en los modelos de experiencia, comunicación y coordinación, respectivamente a realizar posteriormente.



### **Experiencia**

En este modelo se describe como se llevan a cabo tareas cognitivas. Es decir, aquellas tareas que requieren conocimiento para ser llevadas a cabo y que permitirán caracterizar al agente que desarrolle esa tarea como un sistema basado en el conocimiento. Para ello se realiza un modelado con KADS actualizado. Es el modelo que mas diferencia esta metodología de otras, pues es claramente un elemento de la llamada ingeniería del conocimiento.

### **Coordinación**

El modelo de coordinación desarrolla y describe las interacciones entre los agentes de un sistema multiagente. La idea del modelo es que los agentes ofrecen servicios para realizar tareas a otros agentes. El elemento principal del modelo es la *conversación*, de la cual derivan el resto de elementos como: protocolo, intervención, servicio, mensaje, etc. Este modelo está íntimamente ligado con el modelo de agente ya que existen interdependencias entre ambos.

### **Diseño**

Describe las decisiones que deben tomarse para que el modelo definido en los modelos anteriores pueda ser implementado mediante un lenguaje de programación. En este modelo se desarrolla un diseño de la red, un diseño de los agentes y un diseño de la plataforma. El modelo de red proporciona a los agentes una visión uniforme de la red, donde entre otras cosas se determina la presencia o no de agentes auxiliares.

El diseño de los agentes describe fundamentalmente la arquitectura seleccionada para cada agente. Finalmente, el diseño de la plataforma permite seleccionar el sistema operativo y la plataforma tanto software como hardware.

Esta secuencia de fases resulta bastante interesante al reunir técnicas típicas de ingeniería del software con técnicas de la ingeniería del conocimiento. Uno de los problemas que observamos en esta aproximación es la dificultad de su aplicación al igual que ocurre con CommonKADS.

### **A.3.3.6. ZEUS**

ZEUS es una herramienta para construir aplicaciones multi-agente colaborativas. Que provee un entorno integrado para el desarrollo rápido de sistemas[34]. ZEUS define una metodología de diseño de sistemas multi-agente y lo soporta mediante un entorno visual para capturar las especificaciones de los agentes. Estas especificaciones son luego utilizadas para generar el código fuente en Java[35].

El objetivo del proyecto ZEUS es facilitar el desarrollo rápido de nuevas aplicaciones multi-agente mediante la abstracción de los principios y componentes más comunes a una herramienta. Su objetivo consiste en proporcionar una herramienta de propósito general y personalizable, que permita la creación de agentes colaborativos y que pueda ser usada por ingenieros de software con poca experiencia en tecnología de agentes para crear sistemas multi-agente[34].



La herramienta ZEUS consiste de un conjunto de componentes, escritas en el lenguaje de programación Java, que puede ser categorizada en tres grupos funcionales o librerías: una librería de componentes de agentes, una herramienta de construcción de agentes y un conjunto de agentes utilitarios entre los cuales podemos encontrar servidores de nombres, facilitadores y agentes visualizadores. A continuación se describe cada una de ellas[35].

- **Librería de componentes de agentes:** es una colección de clases que forman los bloques de construcción de los agentes individuales. El contenido de esta librería muestra los puntos identificados antes: entre ellos la comunicación, ontologías, coordinación, etc. Para la comunicación, la librería de componentes de agentes provee un lenguaje de comunicación entre agentes basado en actos del habla y en performatives, un sistema de pasaje de mensajes asincrónico basado en sockets, un editor para describir ontologías de dominio específico y un lenguaje de representación de conocimiento basado en frames para representar los conceptos del dominio. Para el razonamiento y la coordinación de múltiples agentes, la librería de componentes de agentes provee[35]:
  - Un planificador de propósito general y un sistema de planificación preparado para dominios típicos de aplicaciones orientadas a tareas, y un mecanismo cooperativo de resolución de problemas de estas aplicaciones.
  - Un motor de coordinación que controla el comportamiento social de un agente.
- **Herramienta de construcción de agentes:** esta herramienta está diseñada para proveer un desarrollo rápido de agentes a alto nivel, ocultando la complejidad de la librería de componentes de agentes. Está conformada por un conjunto de editores diseñados para permitir a los usuarios interactivamente crear agentes mediante una especificación visual de sus atributos. El conjunto actual de editores incluye[35]:
  - Un editor de Ontologías para definir los ítems de la ontología en un dominio.
  - Un editor de hechos/variables para describir instancias específicas de hechos y variables, utilizando los templates creados usando el editor de ontologías.
  - Un editor de definiciones de agentes para describir los agentes lógicamente. Esto involucra la especificación de las tareas de cada agente, sus recursos iniciales y las dimensiones de su plan
  - Un editor de descripción de tareas para especificar los atributos de las tareas y para resúmenes gráficos de las tareas.
  - Un editor de organización para definir las relaciones organizacionales entre los agentes, y las creencias de los agentes acerca de las habilidades de otros agentes.
  - Un editor de coordinación para seleccionar el conjunto de protocolos de coordinación con los cuales cada agente estará equipado.

- **Agentes utilitarios ZEUS:** consiste de un servidor de nombres, un facilitador para el descubrimiento de información, y un agente para visualizar o realizar un debugging de sociedades de agentes. Una sociedad de agentes en ZEUS puede contener cualquier número de agentes utilitarios, con al menos un servidor de nombres. Todos los agentes utilitarios son construidos utilizando las componentes básicas de la librería de componentes de agentes, y son en realidad simplificaciones del agente genérico ZEUS. El proceso de la realización del agente sigue la etapa del diseño de la metodología del desarrollo del agente de ZEUS. Durante este capítulo será descrito metódicamente como serie de etapas del desarrollo, que consiste en las actividades individuales que ponen aspectos en ejecución particulares de un agente. Un requisito previo para usar este documento es un diseño para el uso que es observado. Además el revelador debe saber cuándo y cómo realizar las actividades necesarias. Pero antes de que consideremos cómo poner un agente en ejecución, necesitamos considerar la base conceptual de los agentes creados con la caja de herramientas de ZEUS[35].

### El agente conceptual de Zeus

En el del más alto nivel de la abstracción, el acercamiento del diseño del agente de ZEUS requiere a reveladores ver un agente como entidad cinco-acodada, según lo ilustrado en el cuadro de abajo. La **Figura 1 – Agente Zeus** muestra las capas de un nivel inferior a un nivel superior, [35]:



Figura 1 – Agente Zeus

**La capa del *interfaz*** recibe la entrada vía sus sensores y puede cambiar el mundo exterior con sus efectores. Por ejemplo, un sensor podía recibir instrucciones de un GUI o de un acto modificando una base de datos, así proporcionando las realizaciones físicas de los recursos y de las habilidades de los agentes. Pues los sensores y los efectores son externos al agente, la configuración de la capa del interfaz consiste en el especificar de interfaces entre ellos y el agente[35].

En **la capa de la *definición*** el agente se percibe como una entidad racional autónoma, en términos de sus capacidades del razonamiento, metas, recursos, habilidades, creencia y preferencias. Como



esta capa es observada físicamente por las clases de la biblioteca componente del agente de ZEUS, (descrita en el manual técnico), esta etapa de la metodología implica la configuración de estos componentes[35].

En la **capa de la organización** el agente se percibe en términos de sus relaciones con otros agentes. Esto introduce el concepto de una agencia - grupo de agentes relacionados[35].

Las agencias pueden ser verdaderas, en que son relacionadas en virtud de cualidad común, tal como ser parte de la misma compañía. O las agencias pueden ser virtuales, en que sus componentes comparten un acuerdo de la cooperación. Así esta etapa metodología implicar configurar agente en términos de agencia él pertenecer, qué papel él jugar con agencia, qué otro agente él ser enterado, y qué capacidad él saber, otro poseer[35].

En la **capa de la coordinación** el agente se considera ser una entidad social que, por lo tanto esta etapa implica el configurar de ella para las formas deseadas de comportamiento social. Esto es alcanzado equipándola de los protocolos y de las estrategias apropiadas de la negociación[35].

La capa de la **comunicación** proporciona un protocolo y una lengua del transporte permitiendo a agentes enviar mensajes el uno al otro. Para la compatibilidad esta capa debe ser igual en cada agente, y tan allí ninguna necesidad para que los reveladores configuren esta capa.

#### **A.3.3.7. MASINA**

La metodología MASINA (Multi Agent Systems in Automation) está fundamentada en MAS-CommonKADS, usando un ciclo de desarrollo muy similar, el cual solo se utilizan cinco modelos, para la descripción de las características básicas de los SMA[38]. Los modelos definidos en MASINA son: **Modelo de Agente**, que describe las características de cada agente, con dos extensiones particulares, una que permite el uso de modelos de referencia para especificar un agente dato, y otra para modelar los niveles de abstracción del Sistema Multi-agentes; **Modelo de Tareas** representa las actividades realizadas por los agentes para alcanzar sus objetivos, es posible especificar tareas que requieren el uso de técnicas inteligentes, así como describir el macro algoritmo de la tarea; **Modelo de Inteligencia** que describe los conocimientos, mecanismo de aprendizaje y métodos de razonamiento necesarios para que los agentes cumplan sus objetivos; **Modelo de Coordinación** que agrupa las interacciones en conversaciones y define los mecanismos para que esas interacciones se lleven a cabo coherentemente y **Modelo de Comunicación** que describe los actos de habla (interacciones) [38].

MASINA permite plasmar elementos fundamentales del área de SMA, como representar la noción de inteligencia en un agente (modelado de agentes inteligentes), a nivel colectivo, los mecanismos de coordinación entre agentes (planificación emergente, resolución de conflictos, etc.), la comunicación directa o indirecta, entre otras cosas. Además, es posible representar aspectos como el uso de técnicas inteligentes para realizar tareas, o modelos de referencia para especificar agentes. Finalmente, MASINA permite describir las conversaciones entre los agentes como los actos de habla (interacciones) presentes en las mismas[38].

### A.3.4. Teoría de los Actos de Habla

La teoría de los actos de habla (*speech acts*) es un marco teórico de alto nivel para el estudio del lenguaje desarrollado por Austin, Grice y Searle, entre otros. Esta teoría surgió para evitar algunos de los problemas de las teorías semánticas desarrolladas hasta 1960[33]. Hasta entonces, se consideraba que el contenido semántico de una frase coincidía con su valor de verdad. Sin embargo, esta interpretación sólo era válida para aserciones, mientras que las frases que contenían deseos o promesas no tenían una semántica asociada, pues no pueden ser simplemente ciertas o falsas, sino que se puede considerar que la semántica de una promesa o una petición depende de que sean sinceras o no, o que se hayan realizado con éxito o no[33].

La principal contribución de Austin consiste en considerar que el principal papel del lenguaje, de la comunicación, es un tipo de *acción*. Según Austin podemos distinguir tres tipos de acciones que pueden ser realizadas en un acto comunicativo[33]:

- **Una locución:** que es la acción física asociada a pronunciar la frase.
- **Una in-locución:** que son las acciones que el hablante intenta realizar con el acto comunicativo.
- **Las per-locuciones:** que son las acciones que ocurren como resultado de las inlocuciones.

Por ejemplo, en la frase “Dobla la esquina”, el sonido resultante de su pronunciación es una locución; la orden que intenta el hablante es la inlocución, y la perlocución puede ser que el oyente doble (o no) la esquina[33]. Esta frase podría tener un verbo realizativo explícito: “Te ordeno que dobles la esquina”. De acuerdo con esta teoría, todas las frases se pueden escribir con un verbo realizativo (*performative verb*) asociado, que se identifica con el acto comunicativo, y actúa como verbo principal de la frase. La teoría de los actos de habla identifica varios tipos de actos de habla, ofreciendo diversas clasificaciones[33], como:

- **Asertivos:** que son declaración de hechos.
- **Directivos:** que son órdenes, peticiones o consejos
- **Compromisitos:** que son promesas;
- **Expresivos:** que expresan sentimientos y actitudes
- **Declarativos:** verbos que son una acción en sí mismos, p.ej. “Declaro la guerra”, “Llamo a la máquina bosco”.

La teoría de los actos de habla está siendo aplicada en áreas muy diferentes, entre las cuales se destacan:

- **Comprensión y generación de lenguaje natural:** el proceso de comprensión a menudo se estudia como un proceso de reconocimiento de la planificación que tiene como misión reconocer cuál es la intención del hablante cuando expresa una frase[33]. De la misma forma, el proceso de generación intenta, mediante un proceso de planificación, buscar las frases y estructuras que pueden conseguir transmitir mejor unos objetivos comunicativos.



Para ambos procesos, se ha realizado una extensión de la teoría de los actos de habla para contemplar sentencias relacionadas en vez de sentencias aisladas, denominada *teoría del discurso*[33].

- *Inteligencia artificial distribuida*: los mensajes intercambiados pueden considerarse actos de habla.

Se han seguido diversos enfoques para determinar la semántica de los actos de habla. El primer enfoque considera que la semántica de los actos de habla consiste en determinar bajo qué condiciones se puede asegurar que un acto comunicativo ha ocurrido, es decir, cómo identificar el tipo de acto comunicativo (asertivo, directivo, etc.) que ha tenido lugar[33]. El segundo enfoque consiste en centrarse en el estudio de los aspectos perlocutivos del acto comunicativo, esto es, qué efectos tiene el acto comunicativo en los estados cognitivos de oyente y hablante. Este enfoque podría clasificarse más como pragmático que como semántico. El tercer enfoque considera que la semántica de un acto comunicativo dado son las condiciones bajo las cuales podemos afirmar que ese acto comunicativo ha sido satisfecho[33].

El conocimiento de la semántica de los actos de habla es relevante en el diseño de sistemas multiagente, ya que el diseñador de un sistema multiagente debe intentar asegurar que las intenciones y el saber-cómo de los agentes sean tales que sólo se den escenarios correctos (entre todos los escenarios posibles).

#### **A.3.4.1. FIPA - Foundation for intelligent and physical agents**

FIPA es un consorcio industrial fundado en 1996 por varias compañías de telecomunicaciones e informática. Es un grupo multi-disciplinar que persigue la estandarización de las tecnologías de agentes[39]. Esta organización ha puesto a disposición una serie de especificaciones que soporten la interoperabilidad entre agentes y aplicaciones basadas en agentes para dirigir el desarrollo de sistemas multiagentes. Cubre todos los aspectos de un entorno de agentes[39]:

- Aplicaciones
- Arquitectura
- Comunicación
- Protocolos de Interacción, Actos comunicativos y Lenguajes de Contenidos
- Gestión de Agentes
- Transporte de mensajes

Especificaciones de FIPA

Hasta la fecha ha publicado 3 conjuntos de especificaciones FIPA 97, FIPA 98 y FIPA 2000



### FIPA 97 y 98

<b>Tecnología de agentes</b>
Gestión de agentes.
Lenguaje de comunicación entre agentes.
Interacción agente-humano.
Interacción de agentes con sistemas software.
Movilidad de agentes.
Seguridad de agentes.

Tabla 1 – Tecnología de agentes

<b>Aplicaciones de prueba.</b>
Agente personal de viaje.
Asistente personal.
Difusión y entretenimiento audio-visual.
Gestión de redes.

Tabla 2 – Aplicaciones de prueba

### FIPA 2000

<b>Evolución del modelo de comunicaciones de agentes</b>
Transporte de los mensajes.
Representación de los mensajes (como strings, objetos o XML).
Atributos opcionales en los mensajes (p.ej. autenticación, encriptado).

Tabla 3 – Evolucion del modelo de comunicaciones de agentes

<b>Uso de entornos software existentes</b>
Plataformas y lenguajes de programación para computación distribuida.
Plataformas de mensajería.
Servicios de seguridad.
Servicios de directorio.
Tecnologías de conectividad intermitente (móviles).

Tabla 4 – Uso de entornos software existentes



### A.3.4.2. FIPA ACL

Basado en la teoría de los actos de habla, esta compuesto por 22 actos comunicativos distintos que permiten representar las intenciones (ilocuciones) de la comunicación (Pedir, proponer, informar, preguntar, rehusar, etc) [40]. FIPA define una serie de patrones de creconversación para las comunicaciones más frecuentes: protocolos de interacción. Además de la ilocución, los mensajes incluyen:

- Información sobre el emisor y el receptor.
- Referencia a la conversación de la que forma parte.
- Información temporal incluyendo tiempos de espera máximos.
- Propiedades varias del mensaje: codificación, lenguaje utilizado
- El contenido del mensaje propiamente dicho.

Fipa acl no define de manera restrictiva que es un agente o como implementarlo, no obstante este asume que el agente tiene las siguientes propiedades[40]:

#### Actitudes mentales

- **Creencias:** el conjunto de propiedades que el agente acepta como verdaderas.
- **Incertidumbre:** el conjunto de propiedades que el agente acepta que probablemente son mas verdaderas que falsas.
- **Intención:** el conjunto de propiedades que el agente desea que sean verdaderas pero que no se podrían considerar de esta forma.

#### Actos de habla

El agente ejecuta algunas acciones llamadas actos de habla que son ejecutados enviando mensajes codificando el acto[40].

FIPA ACL minimiza requerimientos al servicio de entrega de mensajes a fin de que los agentes puedan utilizar diferentes mecanismos de transporte (TCP/IP, HTTP, SMTP, GSM, etc). Este impone un conjunto de requerimientos sobre el servicio de envío de mensajes[40]:

- El servicio de envío de mensajes está disponible para entregar mensajes como una secuencia de bytes.
- El servicio de mensajes es confiable, exacto y ordenado a menos que sea informado de manera errónea.
- El agente puede escoger si es un proceso asíncrono o síncrono.
- Los parámetros del acto de entrega de mensajes no son codificados en el mensaje pero el servicio de mensajes esta disponible para manejar esto.
- El servicio de entrega de mensajes informa al agente de errores ocurridos.
- Cada agente tiene un nombre que permite al servicio de envío de mensajes conocer donde se encuentra.



### A.3.4.3. **Protocolos de interacción**

Para poder establecer una conversación entre agentes es necesario definir previamente el protocolo que van a seguir durante la conversación[41]. Un protocolo de interacción es una descripción detallada del tipo y orden de los mensajes involucrados en una conversación entre agentes, un agente puede participar simultáneamente en múltiples diálogos con diferentes agentes y con diferentes protocolos de interacción. Dentro del estándar FIPA (utilizando AUML) se encuentran definidos algunos de los protocolos más extendidos. Para utilizar un protocolo en una conversación, los agentes deben escribir el nombre del protocolo a utilizar en el parámetro: **protocol**, un protocolo termina cuando se alcanza el último mensaje del protocolo o se elimina el nombre del protocolo del parámetro: `protocol`. Cuando un agente no conoce un determinado protocolo tiene que devolver un mensaje *refuse* explicando el motivo por el que rechaza la comunicación, si durante el seguimiento de un protocolo, un agente recibe un mensaje no contemplado en el mismo tiene que devolver un mensaje *not-understood*, está prohibido responder un *not-understood* con otro *not-understood* para no caer en bucles infinitos[41]. Algunos protocolos son:

- **FIPA QUERY:**

Este protocolo posibilita la petición de información Se inicia con uno de las siguientes actos comunicativos[41]:

- `query-if` (verdad o falsedad de una proposición)
- `query-ref` (información en referencia a un objeto)

El participante puede responder con[41]:

- `inform` (la información solicitada)
- `refuse` (rechazar la petición)
- `failure` (error)
- `not-understood` (no entiende la petición)

- **FIPA CONTRACT NET**

Es un protocolo de Negociación que se utiliza para permitir que un agente iniciador solicite la ejecución de una tarea a uno o más agentes[41]:

- El iniciador hace una llamada para ver qué agentes pueden realizar la tarea
- Los agentes receptores pueden rechazar el mensaje, no entenderlo o devolver al iniciador una propuesta con sus precondiciones para realizar la tarea
- Cuando el iniciador recibe las propuestas, las evalúa, escoge a un agente (o a varios) para realizar la tarea y envía una aceptación al agente (o agentes) elegido y un rechazo al resto
- Cuando los agentes contratados realicen la tarea informan al iniciador. Estos agentes pueden subcontratar la tarea a otros



- **FIPA ENGLISH AUCTION**

Protocolo de Subasta, el subastador intenta encontrar el precio de venta de un objeto proponiendo inicialmente un precio por debajo del valor del mercado para después ir incrementándolo gradualmente, una vez anunciado el precio, el subastador espera un tiempo para ver si algún comprador expresa su intención de pagar dicha cantidad[41]:

- Si un agente acepta el precio, el subastador incrementa el precio y lo anuncia al resto esperando nuevas propuestas
- La subasta termina cuando no hay pujas para el último precio propuesto
- Se vende si en la última puja el precio supera un determinado precio de reserva (privado para el subastador)
- No se vende si en la última puja el precio no supera el precio de reserva

- **FIPA DUTCH AUCTION**

Protocolo de Subasta en el cual un subastador intentar encontrar el precio de venta de un objeto proponiendo inicialmente un precio por encima del valor del mercado para después reducirlo gradualmente, al anunciar el precio, el subastador espera un tiempo para ver si algún comprador expresa su intención de pagar la cantidad si ningún agente acepta el precio, el subastador disminuye el precio y lo anuncia al resto esperando nuevas propuestas. La subasta termina cuando[41]:

- hay algún comprador para el precio propuesto (se vende)
- Se ha bajado el precio hasta el mínimo permitido (no se vende)
- Inspirado en las subastas de los mercados de flores holandeses.

## ***A.4. Motores de Juegos***

Un motor de juegos es una Interfaz de Programación de Aplicaciones (API por sus siglas en inglés) o librería de funciones, que se encarga del manejo de los gráficos, de interpretar las acciones del usuario en el teclado, mouse o joystick, comunicándole al juego que el jugador quiere que su personaje salte, camine, pelee, etc. En otras palabras, el motor de juegos es la tecnología que mueve al juego [18]. Esta es la razón por la cual el desarrollo de un juego debe estar soportado en la selección de un buen motor de juegos, que considere el aspecto gráfico, la comunicación humano-computador, la dinámica y la flexibilidad del juego en tiempo real.

A continuación se muestran los elementos básicos comunes de un Motor de Juegos [18]:

- **Motor de renderizado y Animación:** Proporciona funciones de renderizado (2D - 3D – Sprites 3D), para realizar esto se apoya en librerías y API graficas como OpenGL y DirectGraphics. Este motor se encarga de la definición de una capa de abstracción entre Hardware, Software y el



juego en sí, además de visibilidad, mapeado, gestión de mallas 3D, etc. Todas estas funciones son las que permiten tener la impresión de ver figuras en 3D en la pantalla (en dos dimensiones) de un monitor, que no está diseñado para proyectar imágenes en 3D.

- **Grafo de escena:** Esta es una parte muy importante de un motor de juegos, y se basa en una estructura de datos muy utilizada para el desarrollo de gráficos basados en vectores, además de la construcción de videojuegos. Es la encargada de ordenar la representación lógica y espacial de una escena gráfica.
- **Motor de física:** Este software es el encargado de simular los diferentes modelos de física newtoniana utilizando variables del tipo velocidad, masa, etc. Su objetivo principal consiste en simular y predecir los efectos bajo diversas situaciones de lo que ocurre en la vida real o en un mundo de fantasía.
- **Motor de Sonidos:** Encargado de la reproducción de la banda sonora, los efectos de sonido (disparos, explosiones, etc.), de una forma sincrónica con las acciones dentro del videojuego.
- **Gestión de Redes:** Permiten a los juegos utilizar la componente de red como: los juegos en línea, los juegos multijugador a través de diferentes protocolos de comunicación, etc.

De todos los componentes anteriores, solamente el motor de renderizado y el motor escena aparecen en la mayoría de motores de juegos, los demás no son tan imprescindibles al momento de escoger el motor, más bien dependen de lo que se va a crear. Si se tiene en cuenta que al utilizar un motor que vincule la mayor cantidad de elementos necesarios para el desarrollo de un videojuego, se economiza significativamente el tiempo de creación debido a que no se reinvierte esfuerzos en el desarrollo de partes que se encuentren implementadas y probadas.

A continuación son enumerados algunos de los motores utilizados en el desarrollo de videojuegos en la actualidad.

### A.4.1. Fly3D

Fly3D SDK es un motor de videojuegos y de desarrollo de aplicaciones 3D en tiempo real. Incluye un potente motor 3D de simulación orientado a objetos que se encarga de tareas como visualización en tiempo real, captura de eventos de entrada o simulación física de objetos 3D. Fly3D usa el modelo de agregados plugins<sup>1</sup>. Los plugins están completamente separados del núcleo de Fly3D, y todas las aplicaciones tienen un front-end<sup>2</sup> común con la interfaz de usuario y el control principal del videojuego[42]. Una aplicación Fly3D es un conjunto de objetos que heredan

<sup>1</sup> Plugin: es una aplicación informática que interactúa con otra aplicación para aportarle una función o utilidad específica, generalmente muy específica, como por ejemplo servir como controlador en una aplicación, para hacer así funcionar un dispositivo en otro programa.

<sup>2</sup> Front-end: Es la parte del software que interactúa con el usuario.



del objeto básico de Fly3D. Cada objeto redefine un conjunto de funciones virtuales que se ponen en ejecución en determinados momentos del videojuego. Entre estas funciones están la función de simulación del objeto y la de visualización. El programador define el comportamiento del objeto usando estas funciones. El proceso de simulación recorre todos los objetos del videojuego ejecutando la función de simulación de cada objeto tomando en consideración el tiempo desde la última simulación. La visualización sigue un proceso similar. Entre otras características hay que destacar que su licencia es libre sin coste y se tiene acceso a todo el código fuente[42].

#### **A.4.2. Unreal**

Es el Motor gráfico del juego Unreal desarrollado por la empresa Epic MegaGames basado en una extensión del renderizado en portales conocido como *Dynamic Scene Graph Technology (DSG)*, BSP y radiosidad<sup>3</sup>. Su principal característica es la gran escalabilidad, es decir, es muy modular. Se encuentra disponible para las plataformas Linux, Windows, Macintosh, Playstaion 2 y Xbox. Para desarrollar con este motor se tiene que adquirir una licencia (USD 250.000 - USD 500.000) que da acceso a todo el código fuente, a las herramientas y juegos[42].

#### **A.4.3. Genesis3D**

Desarrollado por Eclipse Entertainment basado en renderizado en portales, BSP y radiosidad. La única plataforma soportada es Windows. Bajo licencia libre y derecho a modificar el código fuente, a cambio de tener que mostrar el logotipo del motor en las aplicaciones que lo utilicen o bien bajo licencia comercial de coste USD 10,000 por título, para que no salga el logotipo de genesis 3D y código abierto[42].

#### **A.4.4. Cristal Space**

Desarrollado por Jorrit Tyberghein, basado en renderizado en portales, BSP, ZBuffering y radiosidad. Entre todas sus principales características destacamos la gran portabilidad y la gran escalabilidad proporcionada por el sistema de plugins y, además, bajo licencia LGPL. Las plataformas soportadas son UNIX (Linux, y Solaris), DOS, Macintosh, Amiga, Windows, BeOS, NextStep, Rhapsody y ports OpenStep[42].

---

<sup>3</sup> Radiosidad: Técnica para el cálculo de la iluminación global de un ambiente cerrado.



### A.4.5.XNA

XNA fue desarrollado por Microsoft y lanzado hace pocos años. Xna no es solo un framework como DirectX<sup>4</sup>, sino que también contiene una gran cantidad de herramientas e incluso un entorno de desarrollo integrado (IDE) derivado del visual Studio para facilitar la programación de videojuegos[43]. XNA Game Studio Express es una nueva solución de desarrollo de juegos para Microsoft Windows y Xbox 360 dirigida principalmente a estudiantes, aficionados y desarrolladores de juegos independientes, está basado en Visual C# 2005, XNA Game Studio Express está integrado en XNA Framework, una biblioteca de clases de código administrado diseñada para el desarrollo del juegos para varias plataformas, tanto para Windows como para Xbox 360. XNA Framework simplifica drásticamente no sólo los gráficos, el sonido, la entrada y el almacenamiento de datos, sino también los bucles de temporización y de dibujo utilizados en todos los juegos[43].

Antes de lanzar XNA Game Studio Express, los desarrolladores de juegos usaban tecnologías como DirectX<sup>®</sup> y OpenGL. Estas API permiten obtener acceso básico al hardware, pero no son intuitivas[43]. Debido a ello, muchas veces los desarrolladores ensamblaban componentes básicos como gráficos, audio y datos de entrada en un marco, denominado motor y luego establecían una lógica específica del juego por niveles sobre este marco. Esta abstracción a menudo es costosa y ocupa mucho tiempo de creación y utilización, además, la administración de la memoria y la lógica en C++ (el lenguaje elegido para el desarrollo de juegos) es muy exigente técnicamente, ya que administra todas las piezas de contenido artístico presentes en un juego. No es de extrañar que el desarrollo de juegos haya sido tan complicado para los desarrolladores aficionados. Igual de intimidantes son los complejos requisitos técnicos y empresariales para el desarrollo de consolas de videojuegos de última generación, como la Xbox 360[43].

XNA Game Studio Express también incluye el XNA Framework Content Pipeline, integrado en el entorno de generación de Visual C# 2005. Mediante Content Pipeline, los desarrolladores de juegos pueden agregar su contenido artístico (modelos 3D, texturas gráficas, audio) en su solución del IDE, de forma que se integre en objetos de tiempo de ejecución a los que se puede obtener acceso con un código[43]. Es una forma de reducir la posibilidad de sobrecarga que se asocia con el contenido artístico.

Las soluciones de desarrollo de juegos anteriores solían empezar de cero: con bucles de temporización y de representación escritos manualmente que administraban cada secuencia y garantizaban el control correcto de la acción del juego[43]. XNA Game Studio Express crea los bucles de temporización y de representación automáticamente. Se dispone de una clase de juego que controla su propia temporización y representación, que expone un método de actualización donde se pueden actualizar los marcos y un método de representación donde se pueden agregar objetos para representarlos en la pantalla, además el marco que se suministra con el juego, denominado modelo de la aplicación, establece automáticamente el dispositivo de gráficos

---

<sup>4</sup> DirectX: Programa creado por Microsoft para crear gráficos y efectos el cual permite la integración a aplicaciones como juegos o sitios de internet.



apropiado sin ningún código complicado de enumeración de dispositivo. Como el código ya está establecido, se puede compilar y ejecutar un proyecto de XNA Game Studio Express con un bucle de temporización y representación completamente funcional que se ejecutará desde el momento de su creación[43].

#### **A.4.6. Ogre**

Ogre (Object Oriented Graphics Engine) es un motor de gráficos en tres dimensiones multiplataforma. La principal ventaja de Ogre sobre otros engines 3D es que es un proyecto open source bajo licencia LGPL. Esto significa que su uso es gratuito y solo existen unas pocas exigencias para su uso[46]. Ogre está diseñado desde el principio con la idea de la orientación a objetos, por lo que su interfaz es clara, intuitiva y fácil de usar. Con respecto a sus características, la última versión de Ogre, la 1.2 RC2 que salió hace apenas el 17 de abril, no tiene nada que envidiarle a los motores de juegos más modernos disponibles en el mercado. Entre estas características está el soporte para PS y VS 3.0 y exportadores para las herramientas de modelación tridimensional más usadas en el mercado[46]. Esto significa que con Ogre se pueden desarrollar juegos o cualquier tipo de aplicación que requiera gráficos tridimensionales que no tengan nada que envidiarle a lo más moderno del mercado. Algo que hay que recordar es que Ogre no es un motor diseñado solo con los juegos en mente, es un engine de gráficos 3D general. Debido a esto, Ogre no trae soporte nativo para sonido ni física, lo que no es un problema, ya que gracias a la enorme comunidad existente en Internet, existen módulos especialmente diseñados para Ogre, que permiten tener estas facilidades[46].



# ANEXO B

## SELECCIÓN DE LA METODOLOGÍA PARA LA CONSTRUCCIÓN DE PERSONAJES

Este capítulo presenta el resultado de un estudio con varios de los modelos para el desarrollo de agentes software y presenta una propuesta metodológica denominada GAME-MAS-CKADS para la construcción de personajes en videojuegos de aventura 3D.

### ***B.1. Definición.***

Antes de presentar la metodología resultante de la investigación se debe definir el término **metodología** para clarificar la terminología adoptada en el proyecto. Metodología puede definirse, en un sentido amplio, como un conjunto de métodos o técnicas que ayudan en el desarrollo de un producto software, tal como señala Rumbaugh [47]:

*“Una metodología de ingeniería software es un proceso para la producción organizada del software, empleando una colección de técnicas predefinidas y convenciones en las notaciones. Una metodología se presenta normalmente como una serie de pasos, con técnicas y notaciones asociadas a cada paso... Los pasos de la producción del software se organizan normalmente en un ciclo de vida consistente en varias fases de desarrollo.”*

De acuerdo con Pressman [49], las principales actividades de una metodología son:

- La definición y descripción del problema que se desea resolver.
- El diseño y la descripción de una solución que se ajuste a las necesidades del usuario.
- La construcción de la solución.
- La prueba de la solución implementada.

Entre los requisitos que debe cumplir una metodología se pueden citar los siguientes [28]:

- La metodología está *documentada*: el procedimiento de uso de la metodología está contenido en un documento o manual de usuario.



- La metodología es *repetible*: cada aplicación de la metodología es la misma.
- La metodología es *enseñable*: los procedimientos descritos tienen un nivel suficientemente detallado y existen ejemplos para que personal cualificado pueda ser instruido en la metodología.
- La metodología está basada en *técnicas probadas*: la metodología implementa procedimientos fundamentales probados u otras metodologías más simples.
- La metodología ha sido *validada*: la metodología ha funcionado correctamente en un gran número de aplicaciones.
- La metodología es *apropiada* al problema que quiere resolverse.

## ***B.2. Revisión de metodologías para agentes.***

El primer paso para la adaptación de una metodología que permita la construcción de personajes fue la revisión de cada uno de los modelos de agentes, analizando sus ventajas y desventajas, encontrando la metodología que más se adapte a las características propias de los personajes en videojuegos de aventura.

A continuación se presentan las metodologías que fueron estudiadas para la investigación y se muestran los diferentes modelos que cada una de ellas posee:

<b>METODOLOGIA</b>	<b>DESCRIPCION</b>	<b>MODELOS</b>
<b>MASE [28]</b>	Se concibe como una abstracción del paradigma orientado a objetos donde los agentes son especializaciones de objetos. En lugar de simples objetos con métodos que pueden invocarse desde otros objetos, los agentes se coordinan unos con otros vía conversaciones y actúan pro activamente para alcanzar metas individuales y del sistema.	<ul style="list-style-type: none"><li>➤ <i>Capturar los objetivos</i>: Es la primera fase de la metodología. Toma las especificaciones iniciales del sistema y las transforma en un conjunto de metas del sistema.</li><li>➤ <i>Transformar metas en roles</i>: En este paso las metas anteriormente estructuradas se transforman en una forma más útil para la construcción de sistemas MultiAgentes: los roles y sus tareas asociadas.</li><li>➤ <i>Aplicar casos de uso</i>: Tras la creación de los roles, se asocian las tareas para cada rol. Cada meta asociada a un rol puede</li></ul>



METODOLOGIA	DESCRIPCION	MODELOS
		<p>tener tareas que definan los detalles de cómo se logra.</p> <ul style="list-style-type: none"><li>➤ <i>Crear clases de Agentes:</i> Llegada esta fase, las clases agente se identifican a partir de los roles. El resultado es el Diagrama de Clases Agente, que muestra las clases agente junto con las conversaciones que mantienen dichos agentes.</li><li>➤ <i>Construir conversaciones:</i> Una conversación bajo MASE define el protocolo coordinado entre dos agentes. La metodología define específicamente una conversación como dos Diagramas de Comunicación de Clases, uno para el iniciador de la conversación y otro para el que responde.</li><li>➤ <i>Ensamblar clases de agentes:</i> En esta etapa de MASE se construye la parte interna de una clase agente.</li><li>➤ <i>Diseño del sistema:</i> Es la fase final de la metodología MASE. En esta fase es donde se instancia las clases agente. Se utiliza un diagrama de desarrollo para mostrar el número, tipo y localidad de los agentes dentro del sistema.</li></ul>
<b>GAIA [50]</b>	GAIA es una metodología para el diseño de sistemas basados en agentes cuyo objetivo es obtener un sistema que maximice alguna medida de calidad global (no se llega a detallar cuál). GAIA pretende ayudar al analista a ir sistemáticamente desde unos requisitos iniciales a un diseño que, según los autores, esté lo suficientemente detallado como para ser implementado directamente.	<ul style="list-style-type: none"><li>➤ <i>Roles:</i> Los modelos de roles organizacionales describen de manera precisa todos los roles que constituyen la organización computacional, en términos de sus funcionalidades, actividades y responsabilidades, y los términos de sus protocolos y patrones de interacción.</li><li>➤ <i>Interacciones:</i> Corresponde a todos los principios, procesos, protocolos, estrategias de</li></ul>



METODOLOGIA	DESCRIPCION	MODELOS
		<p>comunicación, políticas sociales y estructuras que permiten a los agentes transportar información</p> <ul style="list-style-type: none"><li>➤ <i>Agentes</i>: El modelo de agentes define el tipo de agentes que creará el sistema, tanto durante su desarrollo como durante la ejecución.</li><li>➤ <i>Servicios</i>: Identifica los servicios que son necesarios para llevar a cabo el rol del agente, como su nombre indica.</li></ul>
<b>MESSAGE [32]</b>	<p>Propone el análisis y diseño del SMA desde cinco puntos de vista para capturar los diferentes aspectos de un SMA: el de Organización, el de Tareas/Objetivos, Los objetivos que persiguen y las tareas implicadas en el proceso; el de Agente, que contiene una descripción detallada y extensa de cada agente y rol dentro del SMA.</p>	<ul style="list-style-type: none"><li>➤ <i>Organización</i>: captura la estructura global del sistema.</li><li>➤ <i>Objetivos/Tareas</i>: determina qué hace el SMA y sus agentes constituyentes en términos de los objetivos que persiguen y las tareas implicadas en el proceso.</li><li>➤ <i>Agentes/Roles</i>: contiene una descripción detallada y extensa de cada agente y rol dentro del SMA</li><li>➤ <i>Interacción</i>: trata las interacciones a distintos niveles de abstracción.</li><li>➤ <i>Dominio</i>: actúa como repositorio de información (para entidades y relaciones) concernientes al dominio del problema.</li></ul>
<b>INGENIAS [30]</b>	<p>INGENIAS como evolución de las ideas de MESSAGE. INGENIAS profundiza en los elementos mostrados en el método de especificación, en el proceso de desarrollo, además de incorporar nuevas herramientas de soporte y ejemplos de desarrollo.</p>	<ul style="list-style-type: none"><li>➤ <i>Agentes</i>: Se usa para describir agentes particulares excluyendo las interacciones con otros agentes. Se centra en la funcionalidad del agente y en el diseño de su control.</li><li>➤ <i>Tarea</i>: Se basa en encontrar las consecuencias que tiene la ejecución de tareas y por qué se deberían ejecutar. Justifica la ejecución de tareas basándose en objetivos.</li></ul>



METODOLOGIA	DESCRIPCION	MODELOS
		<ul style="list-style-type: none"> <li>➤ Interacciones: Define las interacciones entre los agentes o entre agentes y humanos, se definen a alto nivel, en diseño se detalla el protocolo de interacción.</li> <li>➤ Entorno: Define las entidades del entorno del SMA con las que vaya a interactuar.</li> </ul>
<b>ZEUS [34][35]</b>	<p>La metodología ZEUS propone un desarrollo en cuatro etapas: el análisis del dominio, el diseño de los agentes, la realización de los agentes y el soporte en tiempo de ejecución. Las etapas soportadas por la herramienta son la de <i>realización de los agentes</i> y la de <i>soporte en tiempo de ejecución</i>. Las etapas anteriores se basan en el uso de roles para analizar el dominio y en su asignación a agentes.</p>	<ul style="list-style-type: none"> <li>➤ Roles: (Kendall) Cada rol describe una posición y un conjunto de responsabilidades en un cierto contexto. Los roles deben:               <ul style="list-style-type: none"> <li>- Ser modulares</li> <li>- Alta cohesión</li> <li>- No disponer de responsabilidades extrañas</li> <li>- Ser completo (no triviales)</li> <li>- Bajo acoplamiento.</li> </ul> </li> </ul>
<b>MAS – CommonKADS [33]</b>	<p>Esta metodología extiende CommonKADS aplicando ideas de metodologías orientadas a objetos para su aplicación a la producción de SMA. La metodología CommonKADS gira alrededor del modelo de experiencia y está pensada para desarrollar sistemas expertos que interactúen con el usuario. De hecho considera sólo dos agentes básicos: el usuario y el sistema. MAS-CommonKADS extiende los modelos de CommonKADS para tener en cuenta la posibilidad de que dos o más componentes del sistema interactúen.</p>	<ul style="list-style-type: none"> <li>➤ Conceptuación: Permite obtener una primera descripción del problema y la determinación de los casos de uso que pueden ayudar a entender los requisitos informales y probar el sistema.</li> <li>➤ Agentes: Especifica las características de un agente: sus capacidades de razonamiento, habilidades, servicios, sensores, efectores, grupos de agentes a los que pertenece y clase de agente. Un agente puede ser un agente humano, software, o cualquier entidad capaz de emplear un lenguaje de comunicación de agentes.</li> <li>➤ Tarea: Describe las tareas que los agentes pueden realizar: los objetivos de cada tarea, su descomposición, los</li> </ul>



METODOLOGIA	DESCRIPCION	MODELOS
		<p>ingredientes y los métodos de resolución de problemas para resolver cada objetivo.</p> <ul style="list-style-type: none"> <li>➤ Experiencia: Describe el conocimiento necesitado por los agentes para alcanzar sus objetivos. Sigue la descomposición de <i>CommonKADS</i> y reutiliza las bibliotecas de tareas genéricas.</li> <li>➤ Organización: es una herramienta para analizar la organización humana en que el sistema multiagente va a ser introducido y para describir la organización de los agentes software y su relación con el entorno.</li> <li>➤ Diseño: Mientras que los otros cinco modelos tratan del análisis del sistema multiagente, este modelo se utiliza para describir la arquitectura y el diseño del sistema multiagente como paso previo a su implementación.</li> </ul>
<p><b>MASINA[38]</b></p>	<p>(Multi Agent Systems in Automation). Esta metodología usa como base a MAS-CommonKADS, proponiendo algunas extensiones, modificaciones y sustituciones de los modelos definidos en MAS-CommonKADS. MASINA permite plasmar elementos fundamentales del área de SMA, como representar la noción de inteligencia en un agente (modelado de agentes inteligentes), a nivel colectivo, los mecanismos de coordinación entre agentes (planificación emergente, resolución de conflictos, etc.), la comunicación</p>	<ul style="list-style-type: none"> <li>➤ Agentes: describe las características de cada agente, con dos extensiones particulares, una que permitir el uso de modelos de referencia para especificar un agente dato, y otra para modelar los niveles de abstracción del Sistema Multi-agentes</li> <li>➤ Tareas: representa las actividades realizadas por los agentes para alcanzar sus objetivos, en nuestra extensión es posible especificar tareas que requieren el uso de técnicas inteligentes, así como describir el macro algoritmo de</li> </ul>



METODOLOGIA	DESCRIPCION	MODELOS
	directa o indirecta, entre otras cosas.	la tarea ➤ Inteligencia: describe los conocimientos, mecanismo de aprendizaje y métodos de razonamiento necesarios para que los agentes cumplan sus objetivos ➤ Comunicación: agrupa las interacciones en conversaciones y define los mecanismos para que esas interacciones se lleven a cabo coherentemente ➤ Coordinación: describe los actos de habla (interacciones).
METODOLOGIA	DESCRIPCION	MODELOS
<b>MASE [28]</b>	Se concibe como una abstracción del paradigma orientado a objetos donde los agentes son especializaciones de objetos. En lugar de simples objetos, con métodos que pueden invocarse desde otros objetos, los agentes se coordinan unos con otros vía conversaciones y actúan pro activamente para alcanzar metas individuales y del sistema.	➤ <i>Capturar los objetivos:</i> Es la primera fase de la metodología. Toma las especificaciones iniciales del sistema y las transforma en un conjunto de metas del sistema. ➤ <i>Transformar metas en roles:</i> En este paso las metas anteriormente estructuradas se transforman en una forma más útil para la construcción de sistemas MultiAgentes: los roles y sus tareas asociadas. ➤ <i>Aplicar casos de uso:</i> Tras la creación de los roles, se asocian las tareas para cada rol. Cada meta asociada a un rol puede tener tareas que definan los detalles de cómo se logra. ➤ <i>Crear clases de Agentes:</i> Llegada esta fase, las clases agente se identifican a partir de los roles. El resultado es el Diagrama de Clases Agente, que muestra las clases agente junto con las conversaciones que mantienen dichos agentes. ➤ <i>Construir conversaciones:</i> Una conversación bajo MASE define



METODOLOGIA	DESCRIPCION	MODELOS
		<p>el protocolo coordinado entre dos agentes. La metodología define específicamente una conversación como dos Diagramas de Comunicación de Clases, uno para el iniciador de la conversación y otro para el que responde.</p> <ul style="list-style-type: none"><li>➤ <i>Ensamblar clases de agentes:</i> En esta etapa de MASE se construye la parte interna de una clase agente.</li><li>➤ <i>Diseño del sistema:</i> Es la fase final de la metodología MASE. En esta fase es donde se instancia las clases agente. Se utiliza un diagrama de desarrollo para mostrar el número, tipo y localidad de los agentes dentro del sistema.</li></ul>
<b>GAIA [50]</b>	<p>GAIA es una metodología para el diseño de sistemas basados en agentes cuyo objetivo es obtener un sistema que maximice alguna medida de calidad global (no se llega a detallar cuál). GAIA pretende ayudar al analista a ir sistemáticamente desde unos requisitos iniciales a un diseño que, según los autores, esté lo suficientemente detallado como para ser implementado directamente.</p>	<ul style="list-style-type: none"><li>➤ <i>Roles:</i> Los modelos de roles organizacionales describen de manera precisa todos los roles que constituyen la organización computacional, en términos de sus funcionalidades, actividades y responsabilidades, y los términos de sus protocolos y patrones de interacción.</li><li>➤ <i>Interacciones:</i> Corresponde a todos los principios, procesos, protocolos, estrategias de comunicación, políticas sociales y estructuras que permiten a los agentes transportar información</li><li>➤ <i>Agentes:</i> El modelo de agentes define el tipo de agentes que creará el sistema, tanto durante su desarrollo como durante la ejecución.</li><li>➤ <i>Servicios:</i> Identifica los servicios que son necesarios para llevar a cabo el rol del agente, como</li></ul>



METODOLOGIA	DESCRIPCION	MODELOS
<b>MESSAGE [32]</b>	<p>Propone el análisis y diseño del SMA desde cinco puntos de vista para capturar los diferentes aspectos de un SMA: el de Organización, el de Tareas/Objetivos, Los objetivos que persiguen y las tareas implicadas en el proceso; el de Agente, que contiene una descripción detallada y extensa de cada agente y rol dentro del SMA.</p>	<p>su nombre indica.</p> <ul style="list-style-type: none"> <li>➤ <i>Organización:</i> captura la estructura global del sistema.</li> <li>➤ <i>Objetivos/Tareas:</i> determina qué hace el SMA y sus agentes constituyentes en términos de los objetivos que persiguen y las tareas implicadas en el proceso.</li> <li>➤ <i>Agentes/Roles:</i> contiene una descripción detallada y extensa de cada agente y rol dentro del SMA</li> <li>➤ <i>Interacción:</i> trata las interacciones a distintos niveles de abstracción.</li> <li>➤ <i>Dominio:</i> actúa como repositorio de información (para entidades y relaciones) concernientes al dominio del problema.</li> </ul>
<b>INGENIAS [30]</b>	<p>INGENIAS como evolución de las ideas de MESSAGE. INGENIAS profundiza en los elementos mostrados en el método de especificación, en el proceso de desarrollo, además de incorporar nuevas herramientas de soporte y ejemplos de desarrollo.</p>	<ul style="list-style-type: none"> <li>➤ <i>Agentes:</i> Se usa para describir agentes particulares excluyendo las interacciones con otros agentes. Se centra en la funcionalidad del agente y en el diseño de su control.</li> <li>➤ <i>Tarea:</i> Se basa en encontrar las consecuencias que tiene la ejecución de tareas y por qué se deberían ejecutar. Justifica la ejecución de tareas basándose en objetivos.</li> <li>➤ <i>Interacciones:</i> Define las interacciones entre los agentes o entre agentes y humanos, se definen a alto nivel, en diseño se detalla el protocolo de interacción.</li> <li>➤ <i>Entorno:</i> Define las entidades del entorno del SMA con las que vaya a interactuar.</li> </ul>
<b>ZEUS [34][35]</b>	<p>La metodología ZEUS propone un desarrollo en cuatro etapas: el</p>	<ul style="list-style-type: none"> <li>➤ <i>Roles: (Kendall)</i> Cada rol describe una posición y un</li> </ul>



METODOLOGIA	DESCRIPCION	MODELOS
	<p>análisis del dominio, el diseño de los agentes, la realización de los agentes y el soporte en tiempo de ejecución. Las etapas soportadas por la herramienta son la de <i>realización de los agentes</i> y la de <i>soporte en tiempo de ejecución</i>. Las etapas anteriores se basan en el uso de roles para analizar el dominio y en su asignación a agentes.</p>	<p>conjunto de responsabilidades en un cierto contexto. Los roles deben:</p> <ul style="list-style-type: none"><li>- Ser modulares</li><li>- Alta cohesión</li><li>- No disponer de responsabilidades extrañas</li><li>- Ser completo (no triviales)</li><li>- Bajo acoplamiento.</li></ul>
<p><b>MAS – CommonKADS [33]</b></p>	<p>Esta metodología extiende CommonKADS aplicando ideas de metodologías orientadas a objetos para su aplicación a la producción de SMA. La metodología CommonKADS gira alrededor del modelo de experiencia y está pensada para desarrollar sistemas expertos que interactúen con el usuario. De hecho considera sólo dos agentes básicos: el usuario y el sistema. MAS-CommonKADS extiende los modelos de CommonKADS para tener en cuenta la posibilidad de que dos o más componentes del sistema interactúen.</p>	<ul style="list-style-type: none"><li>➤ <b>Conceptuación:</b> Permite obtener una primera descripción del problema y la determinación de los casos de uso que pueden ayudar a entender los requisitos informales y probar el sistema.</li><li>➤ <b>Agentes:</b> Especifica las características de un agente: sus capacidades de razonamiento, habilidades, servicios, sensores, efectores, grupos de agentes a los que pertenece y clase de agente. Un agente puede ser un agente humano, software, o cualquier entidad capaz de emplear un lenguaje de comunicación de agentes.</li><li>➤ <b>Tarea:</b> Describe las tareas que los agentes pueden realizar: los objetivos de cada tarea, su descomposición, los ingredientes y los métodos de resolución de problemas para resolver cada objetivo.</li><li>➤ <b>Experiencia:</b> Describe el conocimiento necesitado por los agentes para alcanzar sus objetivos. Sigue la descomposición de <i>CommonKADS</i> y reutiliza las bibliotecas de tareas genéricas.</li><li>➤ <b>Organización:</b> es una</li></ul>



METODOLOGIA	DESCRIPCION	MODELOS
		<p>herramienta para analizar la organización humana en que el sistema multiagente va a ser introducido y para describir la organización de los agentes software y su relación con el entorno.</p> <ul style="list-style-type: none"><li>➤ Diseño: Mientras que los otros cinco modelos tratan del análisis del sistema multiagente, este modelo se utiliza para describir la arquitectura y el diseño del sistema multiagente como paso previo a su implementación.</li></ul>
<b>MASINA[38]</b>	<p>(Multi Agent Systems in Automation). Esta metodología usa como base a MAS-CommonKADS, proponiendo algunas extensiones, modificaciones y sustituciones de los modelos definidos en MAS-CommonKADS. MASINA permite plasmar elementos fundamentales del área de SMA, como representar la noción de inteligencia en un agente (modelado de agentes inteligentes), a nivel colectivo, los mecanismos de coordinación entre agentes (planificación emergente, resolución de conflictos, etc.), la comunicación directa o indirecta, entre otras cosas.</p>	<ul style="list-style-type: none"><li>➤ Agentes: describe las características de cada agente, con dos extensiones particulares, una que permitir el uso de modelos de referencia para especificar un agente dato, y otra para modelar los niveles de abstracción del Sistema Multi-agentes</li><li>➤ Tareas: representa las actividades realizadas por los agentes para alcanzar sus objetivos, en nuestra extensión es posible especificar tareas que requieren el uso de técnicas inteligentes, así como describir el macro algoritmo de la tarea</li><li>➤ Inteligencia: describe los conocimientos, mecanismo de aprendizaje y métodos de razonamiento necesarios para que los agentes cumplan sus objetivos</li><li>➤ Comunicación: agrupa las interacciones en conversaciones y define los mecanismos para que esas</li></ul>



METODOLOGIA	DESCRIPCION	MODELOS
		interacciones se lleven a cabo coherentemente ➤ Coordinación: describe los actos de habla (interacciones).

Tabla 5 - Revisión de Metodologías

Después del estudio realizado a cada metodología se aplicaron los requisitos que debe cumplir cada una de las fases de todas las metodologías, obteniendo como resultado lo siguiente:

### Análisis

Los criterios del Análisis incluyen:

- Completitud: ¿el método de análisis es útil, y está bien documentado?

METODOLOGIA	COMPLETITUD
<b>MASE</b>	Esta metodología divide cada uno de los pasos del análisis en 3 etapas más en donde cada uno de los resultados de una etapa es la entrada de la siguiente, facilitando de esta manera definir cada una de las funcionalidades del sistema, se encuentra bien documentada y se explica con un ejemplo cada etapa[27] [60] .
<b>GAIA</b>	Maneja el concepto de roles con el fin de entender el sistema sin necesidad de hablar de la implementación. Se trabaja con la idea de <i>organización</i> . Pero le falta detalle al momento de definir cada proceso. [50]
<b>MESSAGE</b>	El sistema es visto como un conjunto de organizaciones que interactúan con recursos, actores y otras organizaciones. La idea es ir aumentando el nivel de detalle a medida que se conoce más del sistema a modelar. Se encuentra poco documentado. [32]
<b>INGENIAS</b>	INGENIAS dispone de una cantidad enorme de entidades y relaciones. Su uso mediante la herramienta de soporte INGENIAS IDE, se encuentra bien documentada por el equipo GRASIA [30]
<b>ZEUS</b>	El análisis se debe a la voluntad del desarrollador Basado en el modelado de roles, utilizando diagramas de clase UML y patrones. [34]
<b>MAS-CommonKADS</b>	Esta metodología está bien documentada [33][36][57][59], es la primera en incorporar la idea de proceso de ingeniería con el enfoque de Pressman [49] y describe con detalle cómo se debe definir el sistema teniendo en cuenta las dependencias entre los modelos. En la fase de análisis de MAS-COMMONKADS propone seis modelos: Conceptuación, Agentes, Tareas, Coordinación, Experiencia, Organización [33].
<b>MASINA</b>	La fase de análisis de MASINA se basa en los cinco modelos propuestos en MAS-CommonKADS, los cuales se consideran necesarios para describir las características básicas de los SMA.

METODOLOGIA	COMPLETITUD
	MASINA hace uso de los modelos de agente, tareas, comunicación, coordinación y sustituye el modelo de experiencia de MAS-CommonKADS por el modelo de inteligencia.[35]

Tabla 6 – Análisis (Completitud)

- Aplicabilidad: ¿a qué dominios y problemas se aplica este método?

METODOLOGIA	APLICABILIDAD
<b>MASE</b>	Esta metodología fue diseñada específicamente para modelar Sistemas Multiagente heterogéneos[27]. En ella se contempla a los agentes como entidades no necesariamente deben tener inteligencia que surgen como una especialización de los objetos, por lo cual, las técnicas que utiliza MaSE Proviene del paradigma orientado a objetos. [28]
<b>GAIA</b>	Permite el modelado de sistemas de agentes heterogéneos y ayudar a pasar desde unos requisitos iniciales a un diseño que es lo suficientemente detallado como para ser implementado directamente. [50]
<b>MESSAGE</b>	Se utiliza para construcción de SMA, y con una variación permite el manejo de sistemas en tiempo real. [32]
<b>INGENIAS</b>	Se aplica a una gran variedad de problemas, puesto que combina el esfuerzo metodológico con el desarrollo de la herramienta [30].
<b>ZEUS</b>	Es una herramienta para construir aplicaciones multi-agente colaborativas, Provee un entorno integrado para el desarrollo rápido de sistemas. [34]
<b>MAS-CommonKADS</b>	Al igual que CommonKADS fue desarrollada para sistemas expertos que interactúen con el usuario, la extensión MAS-CommonKADS permite tener en cuenta la posibilidad de que dos o más componentes del sistema interactúen entre sí formando un sistema multiagente. [33]
<b>MASINA</b>	Una Metodología para Especificar Sistemas Multi-agentes, donde se utilizan técnicas inteligentes para la realización de tareas [38].

Tabla 7 – Análisis (Aplicabilidad)

- Complejidad: ¿es el método de análisis fácil de comprender y de aplicar?

METODOLOGIA	COMPLEJIDAD
<b>MASE</b>	Es fácil de entender debido a que maneja conceptos de rol y etapas similares a RUP, como en la creación de los casos de uso, también utiliza diagramas de interacción de UML aunque menos detallados [28].
<b>GAIA</b>	Maneja el paradigma orientado a objetos y el concepto de rol. [50]
<b>MESSAGE</b>	Los modelos son fáciles de entender ya que están basados en UML y están hechos de una forma visual. [32]
<b>INGENIAS</b>	Es un esfuerzo arduo, para poder entenderlo, para aplicarlo pero



METODOLOGIA	COMPLEJIDAD
	después de comprenderlo, resulta muy completo y eficiente. [30]
<b>ZEUS</b>	El método es arbitrario, y escogido por el desarrollador. [34]
<b>MAS-CommonKADS</b>	La información que hay que obtener se expresa con lenguaje natural, por lo que permite modelar diversos tipos de aplicaciones, pero le falta incluir formalismos que logren una definición más precisa y menos ambigua del SMA [33].
<b>MASINA</b>	Al igual que MAS-CommonKADS, es muy flexible y se puede adecuar al problema, pero presentan el problema de la ambigüedad de sus especificaciones que son realizadas netamente en lenguaje natural [38].

Tabla 8 – Análisis (Complejidad)

- Validada: ¿A sido utilizada en diferentes aplicaciones y funciona correctamente?

METODOLOGIA	VALIDADA
<b>MASE</b>	MaSE es una metodología académica que cuenta con el soporte de la Fuerza Aérea de los Estados Unidos. Donde una de sus aplicaciones fue la realización del proyecto “El Escondite[56]” [27].
<b>GAIA</b>	GAIA cuenta con un grupo de investigación (Grupo de Ambientes Inteligentes Adaptativos) dedicado a ampliar la experiencia de esta metodología a partir de la realización de proyectos de investigación basados en esta metodología [50].
<b>MESSAGE</b>	Esta metodología ha sido utilizada para el desarrollo de aplicaciones de telecomunicaciones en proyectos como: KIMSAC[51], AMASE[52], ABROSE[53], entre, otros [32].
<b>INGENIAS</b>	En el sitio web del GRASIA[31] se encuentran listados varios proyectos en los que se trabaja con esta metodología, pero no se han publicado resultados. [30]
<b>ZEUS</b>	En la actualidad existen grupos que trabajan con esta metodología e en sus proyectos como Citeseer[54]. Donde se encontraron los proyectos bien definidos pero no se encontraron resultados concretos [35].
<b>MAS-CommonKADS</b>	Esta metodóloga ha sido aplicada con éxito en varios proyectos de investigación, en diferentes campos, tales como la gestión inteligente de red (proyecto CICYT TIC94-0139 PROTEGER, Sistema Multiagente para Gestión de Red y Servicios)[57] y el desarrollo de aplicaciones con sistemas híbridos (proyecto ESPRIT-9119 MIX, Integración Modular de Sistemas Basados en Conocimiento Simbólicos y Conexionistas)[57]. Además, otros proyectos que se han llevado acabos en la Universidad Politécnica de Madrid.[33]
<b>MASINA</b>	Entre los proyectos en los cuales se ha usado esta metodología es importante referenciar Definition of a Verification Method for the MASINA Methodology[45], pero no se encontraron resultados relacionados con dichos proyectos. [38]

Tabla 9 – Análisis (Validada)

- Argumentada: ¿Las bases teóricas provienen de fundamentos probados o metodologías más simples?

METODOLOGIA	ARGUMENTADA
<b>MASE</b>	Cuenta con el soporte de la fuerza aérea de los Estados Unidos, diseñada para modelar sistemas multiagente heterogéneos. [27] Las técnicas utilizadas se basan en el paradigma orientado a objetos[28].
<b>GAIA</b>	Se basa en el paradigma orientado a objetos. [50]
<b>MESSAGE</b>	Es una metodología resiente pero no se basa en ninguna metodología anterior. [32]
<b>INGENIAS</b>	Esta metodología se encuentra fundamentada en MESSAGE. [30]
<b>ZEUS</b>	Toma como base algunos de los conceptos de UML. Además se basa en conceptos muy comunes para el software como modularidad, alta cohesión, bajo acoplamiento, entre otros. [34]
<b>MAS-CommonKADS</b>	La metodología MAS-CommonKADS es una extensión de la metodología CommonKADS para modelar SMA, agregando aspectos de las metodologías orientadas a objetos como OMT, Ingeniería de Software Orientada a Objetos (OOSE) y Responsibility Driving Design (RDD). [33]
<b>MASINA</b>	Esta metodología se encuentra fundamentada Mas-CommonKads. [38]

Tabla 10 – Análisis (Argumentada)

### Diseño

Los criterios de Diseño incluyen:

- Completitud: ¿el método de diseño es útil y está bien documentado? ¿Existen herramientas para apoyar el proceso de diseño?

METODOLOGIA	COMPLETITUD
<b>MASE</b>	Existen diferentes artículos en donde se realizan todas las etapas del diseño para resolver un problema en particular. Cuenta con una herramienta de soporte llamada AgentTool. [27]
<b>GAIA</b>	Falta detalle al momento de definir cada proceso. Aunque el impacto de esta metodología es notable cabe mencionar la ausencia de una herramienta de soporte para ella. [50]
<b>MESSAGE</b>	Falta estereotipar el diseño. No posee una herramienta de soporte. [32]
<b>INGENIAS</b>	Muy documentado y respaldado por el grupo GRASIA[31]. La herramienta INGENIAS IDE. [30]
<b>ZEUS</b>	El método es arbitrario, y escogido por el desarrollador. Se encuentra soportada en el entorno de desarrollo con su mismo nombre (ZEUS) [34]
<b>MAS-</b>	Muy documentado en la tesis doctoral de Carlos Ángel Iglesias



METODOLOGIA	COMPLETITUD
<b>CommonKADS</b>	Fernández, y posterior mente por otros estudios, Según la web de MAS-CommonKADS existen herramientas de soporte desarrolladas dentro del proyecto. Estas herramientas aparecen bajo el epígrafe MAST (MultiAgent Systems Tool). [33] Sin embargo, la única que se puede descargar no es una herramienta para apoyo al análisis y diseño, sino un conjunto de arquitecturas y frameworks de agentes. [33]
<b>MASINA</b>	Poco documentada, se espera la liberación de la metodología por completo. Hasta el momento no cuenta con una herramienta definida. [38]

Tabla 11 – Diseño (Complejidad)

- Complejidad: ¿el método de diseño es fácil de comprender y de aplicar?

METODOLOGIA	COMPLEJIDAD
<b>MASE</b>	Es fácil de comprender porque se basa en los elementos descritos como agentes y sigue utilizando diagramas del UML como el diagrama de despliegue donde se representan los diferentes agentes, sus comunicaciones y su distribución física. [28]
<b>GAIA</b>	Esta metodología se basa en la identificación de los roles principales que aparecen en el sistema, entendiendo rol como una descripción abstracta de la función esperada de una determinada entidad.[50]
<b>MESSAGE</b>	Aunque esta metodología se basa en UML resulta algo complejo dado que no se ha terminado de concretar. [32]
<b>INGENIAS [30]</b>	El método está bien guiado para su entendimiento y tiene excelente enfoque para su aplicabilidad [30]
<b>ZEUS</b>	El modelado de roles, propuesto no profundiza en la aplicación de la herramienta dentro del proceso de desarrollo. El ámbito de la metodología se limita a estudiar cómo agrupar la funcionalidad del sistema dentro de cada rol, dejando aparte consideraciones acerca de cómo organizar las tareas, definir las ontologías y las dependencias sociales, aspectos que son modelables dentro de la herramienta. [34]
<b>MAS-CommonKADS</b>	La información obtenida es expresada con lenguaje natural, lo que facilita en gran medida su comprensión y aplicación. Pero dada la magnitud de sus fases puede resultar realmente exhaustiva. [33]
<b>MASINA</b>	Comparte con Mas-CommonKADS sus ventajas y desventajas al momento de ser aplicada en la fase de diseño.[38]

Tabla 12 – Diseño (Complejidad)



- Validada: ¿A sido utilizada en diferentes aplicaciones y funcionó correctamente?

METODOLOGIA	VALIDADA
<b>MASE</b>	Es una metodología académica que cuenta con el soporte de la Fuerza Aérea de los Estados Unidos. Donde una de sus aplicaciones fue la realización del proyecto “El Escondite[56]” [27].
<b>GAIA</b>	GAIA cuenta con un grupo de investigación (Grupo de Ambientes Inteligentes Adaptativos [54]) dedicado a ampliar la experiencia de esta metodología a partir de la realización de proyectos de investigación basados en esta metodología [50].
<b>MESSAGE</b>	Esta metodología ha sido utilizada para el desarrollo de aplicaciones de telecomunicaciones en proyectos como: KIMSAC[51], AMASE[52], ABROSE[53], entre, otros [32].
<b>INGENIAS</b>	En el sitio web del GRASIA[31] se encuentran listados varios proyectos en los que se trabaja con esta metodología, pero no se han publicado resultados. [30]
<b>ZEUS</b>	En la actualidad existen grupos que trabajan con esta metodología e en sus proyectos como Citeseer[54]. Donde se encontraron los proyectos bien definidos pero no se encontraron resultados concretos [35].
<b>MAS-CommonKADS</b>	Esta metodóloga ha sido aplicada con éxito en varios proyectos de investigación, en diferentes campos, tales como la gestión inteligente de red (proyecto CICYT TIC94-0139 PROTEGER, Sistema Multiagente para Gestión de Red y Servicios)[57] y el desarrollo de aplicaciones con sistemas híbridos (proyecto ESPRIT-9119 MIX, Integración Modular de Sistemas Basados en Conocimiento Simbólicos y Conexionistas)[57]. Además, otros proyectos que se han llevado acabos en la Universidad Politécnica de Madrid.[33]
<b>MASINA</b>	Entre los proyectos en los cuales se ha usado esta metodología es importante referenciar Definition of a Verification Method for the MASINA Methodology[45], pero no se encontraron resultados relacionados con dichos proyectos. [38]

Tabla 13 – Diseño (Validada)

- Argumentada: ¿Las bases teóricas provienen de fundamentos probados o metodologías más simples?

METODOLOGIA	ARGUMENTADA
<b>MASE</b>	En la etapa de diseño de MASE se apoya en varios de los diagramas de definidos en UML[27].
<b>GAIA</b>	Se basa en las técnicas y metodologías del diseño orientado a objetos [50]
<b>MESSAGE</b>	Los modelos son fáciles de entender ya que están basados en UML y están hechos de una forma visual, aunque aun es considera incompleta. [32]



METODOLOGIA	ARGUMENTADA
<b>INGENIAS</b>	Se encontraron listados varios proyectos desarrollados por el grupo GRASIA en los que se trabaja con esta metodología, pero no se han publicado resultados. [30]
<b>ZEUS</b>	Las notaciones del modelos de roles son originarias de la notación de diagramas de clases de UML. Se basa en conceptos muy comunes para e [35] software como modularidad, alta cohesión, bajo acoplamiento, entre otros.
<b>MAS-CommonKADS</b>	Esta metodología es una extensión de Common-KADS, por los que sus bases se encuentran fundamentadas en esta metodología.
<b>MASINA</b>	Esta metodología es una extensión[33] de MAS-CommonKADS por lo que la mayor parte de fundamentación proviene de ésta. [38]

Tabla 14 – Diseño (Argumentada)

### Desarrollo

Los criterios de Desarrollo incluyen:

- Completitud: ¿cuán útiles son las herramientas de desarrollo proporcionadas?

METODOLOGIA	COMPLETITUD
<b>MASE</b>	Proporciona una herramienta llamada AgentTool. En la cual se puede trabajar las fases de análisis y diseño[28].
<b>GAIA</b>	No cuenta con una herramienta propia para desarrollo. [50]
<b>MESSAGE</b>	No cuenta con una herramienta propia para desarrollo. [30]
<b>INGENIAS</b>	INGENIAS IDE, permite el modelado como la exportación para generar código. [30]
<b>ZEUS</b>	Herramientas en desarrollo no se ha liberado. [35]
<b>MAS-CommonKADS</b>	La metodología no cuenta con una herramienta de desarrollo liberada, pero se han definido un conjunto de un conjunto de arquitecturas y frameworks de agentes, que sirven de soporte a su desarrollo. [33]
<b>MASINA</b>	Herramientas en desarrollo no se ha liberado. [38]

Tabla 15 – Desarrollo (Completitud)

- Complejidad: ¿son las herramientas de desarrollo y los lenguajes fáciles de utilizar? ¿-Es popular el lenguaje utilizado?

METODOLOGIA	COMPLEJIDAD
<b>MASE</b>	Dispone de un lenguaje de especificación basado en UML+OCL (Object Constraint Language), lo que evidencia mucho acercamiento a los conceptos orientados a objetos, asumiendo al agente como un "objeto mejorado" el cual puede contar con inteligencia o no [27]



<b>GAIA</b>	No tiene herramienta para desarrollo. [34]
<b>MESSAGE</b>	No tiene herramienta para desarrollo. [32]
<b>INGENIAS</b>	Posee un manual el cual es muy completo pero extremadamente extenso, esta herramienta es promovida por el grupo GRASIA[31], el cual actualiza constantemente la información sobre la herramienta. [30]
<b>ZEUS</b>	El uso de esta herramienta es bastante complejo dado que necesita un gran conocimiento de los desarrolladores en cuanto al tema de agentes se refiere. [34]
<b>MAS-CommonKADS</b>	Herramientas en desarrollo no se ha liberado [33].
<b>MASINA</b>	Herramientas en desarrollo no se ha liberado. [38]

Tabla 16 – Desarrollo (Complejidad)



# ANEXO C

## APLICACIÓN DE LA METODOLOGÍA PARA LA CONSTRUCCIÓN DE PERSONAJES EN UN VIDEOJUEGO DE AVENTURA EN 3D

---

Este capítulo contiene el desarrollo y documentación de las actividades definidas en las etapas de conceptualización, análisis, modelo de diseño, diseño, codificación, prueba de cada agente, integración, operación y mantenimiento.

### ***C.1. Conceptualización***

#### **C.1.1. StoryLine – Simon y la tuberculosis – Nivel 2**

Todo comienza en un pequeño pueblo, San José de las piedras, En donde vive Simón nuestro protagonista; un niño muy especial con un destino increíble que aun no conoce. San José de las piedras ha sido escenario de importantes eventos pero el que nos trae aquí es uno en especial, y tiene que ver, como la magia dentro del corazón de un niño y su pericia sin igual, tienen la capacidad de cambiar el rumbo de todo su e indiscutiblemente muchos más en un futuro.

Hace 30 años hubo una gran batalla, entre un invasor desconocido y todos los pueblos de la región, siendo San José del último bastión de los habitantes de estos pueblos y en el único lugar donde lograron quedar algunos sobrevivientes ahora esclavos, bajo el reinado de aquel desconocido tirano.

Las fuerzas conjuntas de toda la región lucharon por largos 20 años hasta quedar totalmente reducidos, quedando pueblos más pequeños alrededor totalmente destruidos reducidos a cenizas. Todo comenzó por la ambición y maldad de un extraño rey invasor, el cual usaba unas técnicas que siempre fueron muy peculiares y extrañas, inexplicablemente a su paso imponía ciertos cambios en el espacio (aquí causas de higiene publico), y mucha gente caía enferma y en su mayoría, obligados a persistir en prácticas que empeoraban su salud hasta la muerte.

Así han pasado 10 años, desde que el invasor ganó la guerra y se autoproclamó rey. La única explicación de que no termino con la vida de todas las personas es que de alguna manera también necesita de ellas y para él es divertido verlas enfermas. En este pueblo, el único reducto de



resistencia queda en una casita con un gran sótano, donde una humilde familia intenta sin muy buenos resultados salvar a enfermos graves de la extraña enfermedad. Simón es el menor de la casa y junto a su mamá y a sus abuelitos ayuda en esta laboriosa misión, Un día su mamá, por un descuido se vio infectada por el peligroso virus y todos en su casa se asustaron mucho porque no había mayor modo de ayudarla y sin su ayuda tampoco podría ayudarse a otro montón de personas. La situación empeora cuando Simón por querer ayudar ingenuamente le dijo a alguien extraño que necesitaban ayuda y así quedaron al descubierto, el hombre resulto ser un espía del rey invasor y al descubrirlos este último se puso furioso.

El rey invasor determino desalojar a todos los enfermos de la casa, frente a tanto dolor Simón no supo más que hacer, así que salió corriendo a la tumba de su padre y pedir desesperadamente ayuda sin saber a quién, Cuando de pronto, apareció un gracioso ser mágico que al escuchar sus potentes suplicas, el destino le dio permiso para intervenir; Simón atónito se quedó admirando a aquel ser mientras este le decía:

“Simón, he venido a ayudarte porque así se me ha permitido, mi nombre es Orah y mi misión eterna es mantener el tiempo en orden, en este momento no es mucho lo que puedes hacer por la personas pero existe una oportunidad, solo una y si así deseas no importan los riesgos, está en tus manos...”

Simón al escuchar esas palabras se llenó de alegría y aceptó sin vacilar su misión, al cual el ser mágico respondió:

“Muy bien valiente jovencito, lo que podemos hacer es viajar en el tiempo y regresar al momento crucial en donde tú puedes ayudar, ni un segundo antes ni un segundo después, cierra los ojos... antes de irnos quiero que sepas algo; vas a encontrar pruebas difíciles en tu camino por cada prueba que resuelvas con éxito estaremos a un paso de descubrir la manera de evitar que todas las personas enfermen y poder liberarnos de ese maléfico mal que nos aqueja, en esta misión no estarás solo, yo te acompañare estando siempre a tu lado, culto bajo un disfraz, de un (animalito a escoger).

Así Simón y Orah emprenden su mágico viaje al pasado. Orah guiara por medio de pistas a Simón y le ayudará a interpretar ciertas cosas que él no entienda. Simón se enfrentará por miedo de retos con los capitanes del ejército invasor y a medida que se enfrenta a estos recoge unos cristales que todos unidos desaparecen al malo final, así Simón vuelve a su tiempo y se llena de felicidad al ver todo bonito y nadie con la enfermedad. Su felicidad es tal que todo queda en secreto y se dedica a disfrutar cada día de su vida.

Así, descubren completamente la tuberculosis y las personas pueden lograr un mundo mejor<sup>5</sup>.

---

<sup>5</sup> StoryLine desarrollado por el estudiante de diseño gráfico Alvaro Felipe Bacca Maya.

## C.1.2. Análisis del argumento base

En el caso de los mapas las zonas amarillas son las zonas donde se desarrolla la *acción*. Y en el caso de los enemigos y amigos los distinguimos por los siguientes íconos:



Figura 2 – Notación de los personajes para el argumento base

Los elementos como las piedras y las cercas son obstructores de los recorridos.

ESCENA: Inicio	
<b>CASO:</b>	<b>1</b>
<b>CONTROLES:</b>	Funcionan completamente (Con excepción 1.1.C)
<b>PANTALLA:</b>	Simón en el escenario de juego.
<b>NOTAS:</b>	<ul style="list-style-type: none"><li>- Los soldados no atacan a Simón en los conectores argumentales, por lo tanto no hay gameover.</li><li>- Sdi.3 se encuentra de espaldas a la zona donde Simón se puede mover.</li></ul>
<b>ACCIONES:</b>	- Simón debe ir hasta su casa para encontrarse con Orah cachorrita.
<b>HITS DE CASO:</b>	<p><b>1.1.A.</b> Orah le envía un mensaje “mágico” a Simón, donde le dice que se dirija hasta su casa.</p> <p><b>1.1.B.</b> Simón debe ir a su casa enviado por el mensaje de Orah.</p> <p>* Dado el caso el jugador puede dar una vuelta por el territorio permitido antes de iniciar el diálogo.</p> <p><b>1.1.C.</b> Simón se encuentra con Orah cachorrita.</p>

ESCENA: Inicio				
<b>RESTRICCIÓNES Y POSICIONES DEL MAPA:</b>				
<b>PERSONAJES DEL MODULO:</b>	- Simón - Orah	-Cno.1 -Cno.2	- Sdi.1 - Sdi.2	- Sdi.3 - Sdi.4
<b>DIALOGOS FIJOS:</b>	<ul style="list-style-type: none"> <li>• <b>Caso1.1.A:</b> <b>Antes de iniciar el recorrido y sin necesidad de que El jugador presione click sobre ella:</b> <ol style="list-style-type: none"> <li>1- <b>Orah:</b> Simón!, lo lograste!. Has viajado 25 años en el tiempo hasta la época en que todo comenzó. Necesitas hacer algo antes de continuar: Ve a la casa donde vivirás en un futuro, allí te encontrarás con la amiga que te comenté hace un momento, ella te guiará en tu misión.</li> <li>2- <b>Orah:</b> Mucha suerte pequeño Simón, recuerda que el futuro de tu gente está en tus manos...</li> <li>3- <b>Simón:</b> Pero... Orah!? Te volveré a ver?</li> <li>4- <b>Orah:</b> Más pronto de lo que te imaginas... adiós.</li> <li>5- <b>Simón:</b> Orah?... mejor me doy prisa.</li> </ol> </li> <li>• <b>Caso1.1.B:</b> <b>Comentarios-pistas que dan los ciudadanos (diálogos libres). De hecho uno de estos debe darle a Simón una nota de inventario.</b></li> <li>• <b>Caso1.1.C:</b> <ol style="list-style-type: none"> <li>6- <b>Orah pequeña:</b> Hola holita hola!!!.... Oh por todos los dioses has llegado!! Tú debes ser... Simón?</li> <li>7- <b>Simón:</b> Pero... quien eres tú!?! Acaso....</li> <li>8- <b>Orah pequeña:</b> Ji ji ji, si si si !!! Me presento: Mi nombre es Orah y soy nueva por aquí... no te cuento más de mi porque estoy en una</li> </ol> </li> </ul>			



ESCENA: Inicio		
	<p>misión secreta ya sabes... de todas maneras creo que debes saber más de mí de lo que yo misma sé</p> <p><b>9- Simón: Ah...</b> “Vieja amiga”... ahora veo.. en realidad no “tan” vieja!</p> <p><b>10- Orah pequeña:</b> De que hablas Simón?.</p> <p><b>11- Simón:</b> Oh! No no de nada, es sólo que alguien me dijo que tú eras una buena amiga...</p> <p><b>12- Orah pequeña:</b> Aaah!.. quizás estés hablando de aquella perrita parecida a mí que se me apareció anoche!! No te imaginas el susto que tuve... sin embargo no sé... hay algo en ella que me causo mucha confianza, además si es un ser mágico debe ser igual a mi...</p> <p><b>13- Orah pequeña:</b> Aunque... no es por nada, pero yo soy más linda y joven! Ji ji ji... aquella estaba viejita y despeinada! Espero nunca ser así cuando grande!!!</p> <p><b>14- Simón:</b> Eh?! ... mmmm.. no sabes nada... Oye Oritah.. te puedo decir Oritah?</p> <p><b>15- Orah pequeña:</b> Si, de ahora en adelante dime Oritah... no me molesta en lo absoluto... que me querías decir?</p> <p><b>16- Simón:</b> Mira, necesito tu ayuda, he viajado hasta aquí para intentar detener algo muy malo que va a suceder.</p> <p><b>17- Oritah:</b> Aaaa.. SI! Algo me dijo aquella perra señora, no te preocupes creo que me dió todas las indicaciones necesarias así cuenta conmigo!!!</p> <p><b>18- Oritah:</b> Para empezar, creo que debemos ir a visitar a Don Segundo (DS)... escuché cuando le decía a un vecino que algo extraño le está sucediendo y que al parecer los guardias de aquel brujo curandero no han podido hacer nada para detener la peste que cada día es más poderosa!</p> <p><b>19- Oritah:</b> Don Segundo vive aquí arriba por el camino de la izquierda, sígueme.</p> <ul style="list-style-type: none"> <li>• En este momento Oritah sigue el camino y Simón la sigue automáticamente.</li> </ul>	
<p><b>DIALOGOS LIBRES:</b></p>	<p>Los diálogos “1A” se refieren a cuando simón se ha acercado por primera vez al personaje no jugador. Y los diálogos “1B” es cuando se acercan por segunda vez a estos mismos.(1.1.B)</p> <p><b>1- 1A- Cno.1:</b> Nunca te había visto por aquí pequeño!, vienes del norte!? Porque si es así llamaré a los guardias del curandero!</p> <p><b>1- 1B- Cno.1:</b> Por tu bien no</p>	<p><u><b>Sdi.2 // Sdi.3 // Sdi.4</b></u></p> <p><b>Sí Simón se acerca a cualquier Sdi.* por primera vez o varias veces seguidas con el mismo. (1.1.B):</b></p> <p><b>1- Sdi.*:</b> ¿eeh... Hola niño!.. bonito perrito je, je... Mmm... Lo siento pero el paso por este lugar no está permitido... por lo de la peste tu sabes...</p> <p><b>Sí Simón se acerca a un Sdi.* Inmediatamente después de haber</b></p>



<b>ESCENA: Inicio</b>	
	<p>dudes del alcalde y de aquel magnífico brujo curandero, están haciendo lo mejor para salvarnos de la peste del norte!</p> <p><b>2- 1A- Cno.2:</b> Deberías de estar con los otros hombres trabajando en la granja! Ya estás en edad y realmente necesitan más ayuda... como verás ya soy un poco viejo y ya no tengo fuerzas para hacerlo.</p> <p><b>3- 1B- Cno.2:</b> Sabes? El alcalde da algunos permisos firmados para que los forasteros como tu puedan moverse por la ciudad... Tengo uno. Tómalo, de pronto te sirve niño forastero y toma esta nota que encontré, quizás también te sirva.</p> <p><b>(1.1.B) Al entregarle las notas a Simón el personaje Cno.2 ya no vuelve a decir el diálogo 1B.</b></p>
	<p><b>hablado con otro(1.1.B):</b></p> <p><b>2- Sdi.*:</b> Querido pequeñín... como te dijo antes mi amigo, no puedes pasar... Es por efectos del cuidado que debemos tener según las indicaciones del maestro brujo... tu entiendes cierto?</p> <p><b>(1.1.B) Sí Simón recoge el permiso que le da el Cno.2 en este caso (el permiso no sirve hasta el caso 1.2).</b></p> <p><b>3- Sdi.*:</b> Lo siento pequeñín pero ese pase que tienes no sirve para seguir por ésta sección del pueblo, muy probablemente te funcione en otra parte.</p> <p><b>Sdi.1 No se dirige a Simón en ningún momento pues se encuentra de espaldas a él siempre en este caso (1.1)</b></p>

Tabla 17 – ESCENA: Inicio

ESCENA: Encontrando a Don Segundo				
<b>CASO:</b>	<b>1.2</b>			
<b>CONTROLES:</b>	Funcionan completamente.			
<b>PANTALLA:</b>	Simón en el escenario de juego.			
<b>NOTAS:</b>	Los soldados no atacan a Simón en los conectores argumentales, por lo tanto no hay gameover.			
<b>ACCIONES:</b>	- Simón y Oritah deben encontrarse con Don Segundo (DS)			
<b>HITS DE CASO:</b>	<p><b>1.2.A.</b> Orah y Simón hablan con <i>Sdi.1</i> para poder pasar hasta donde Don Segundo.</p> <p><b>1.2.B.</b> En el caso de que el jugador no tenga el permiso del alcalde que le debió entregar <i>Cno.2</i> en el punto <b>1.1 (caso anterior)</b>, debe entonces ir a buscarlo</p> <p><b>1.2.C.</b> Simón habla con Don Segundo y este lo invita a que hablen del tema en el consejo del pueblo.</p> <p><b>1.2.D.</b> Mini prueba en la que Don segundo a Simón para ayudarlo.</p> <p><b>1.2.E.</b> <i>Sdi.1</i> los descubre cuchicheando</p> <p><b>1.2.F.</b> <i>Sdi.1</i> Explica la prueba</p>			
<b>RESTRICCIONES Y POSICIONES DEL MAPA:</b>				
<b>PERSONAJES DEL MODULO:</b>	- Simón - Orah - Don Segundo	-Cno.1 -Cno.2	- <i>Sdi.1</i> - <i>Sdi.2</i>	- <i>Sdi.3</i> - <i>Sdi.4</i>
<b>DIALOGOS FIJOS:</b>	<ul style="list-style-type: none"> <li><b>Caso1.2.A:</b> Antes de iniciar el recorrido y sin necesidad de que El jugador presione click sobre ellos: 1- <b>Oritah:</b> Guau guau !!!</li> </ul>			



### ESCENA: Encontrando a Don Segundo

Cuando Oritah ladra hace que Sdi.1 se voltee y les da la cara.

- 2- **Sdi.1:** Oh que perrita tan linda tienes pequeñin!
- 3- **Simón:** ...Señor... Voy de visita a donde Don Segundo, sería tan amable de dejarme seguir?.
- 4- **Sdi.1:** Debes tener una autorización expresa del alcalde para poder seguir... Tengo ordenes niño!
- 5- **Oritah:** <Psss Simón... mira en tu inventario y haz click sobre el permiso, me da la impresión que ya lo tienes... Si no, creo que tendremos que ir a buscar uno... >

**(En el caso de haber recogido las notas)**

- 6- **Simón:** Aquí tiene señor...
- 7- **Sdi.1:** eh!?!? Pero com...?!?... eeh je je.. sigue sigue, disfruta tu visita!
- 8- **Oritah:** < Vamos Simón apurémonos tenemos que hablar con el antes de la reunión semanal del consejo y ya casi es la hora! >

• **Caso11.2.B:**

**(En el caso de NO haber recogido la nota)**

- 9- **Sdi.1:** Emm... Lo siento niño sin la autorización del alcalde no te puedo dejar seguir...
- 10- **Simón:** Autorización?!?!?.. pero... cual!?
- 11- **Oritah:** < shisstt... calmate simón! Vámos rápido a buscar ese permiso, el alcalde no se encuentra... Quizás algún ciudadano tenga uno y con suerte nos lo dé ! >

• **Caso1.2.C:**

**Al acercarse a Don Segundo**

- 12- **DS:** ¿?... Oye niño no te conozco... que quieres hablar conmigo?
- 13- **DS:** -- QUE?!?!? Que es lo que dices!? Que la pest...
- 14- **Simón:** Shisstt Don Segundo por favor! Sea precavido que no queremos que nadie que no debe se entere! Recuerde que el guardia está muy cerca y podría escucharnos.
- 15- **DS:** QUE DICES!?!... es algo inaudito!... no lo puedo creer!... sin embargo, he tenido mis dudas sobre todo esto... y lo que me dices tiene totalmente sentido... hay que hacer algo antes de que sea tarde!

• **Caso1.2.D:**

**Antes de la miniprueba.**

- 16- **DS:** por cierto... Espera! Hace unos días descubrí una nota rota en pedazos, sin embargo después de lo que me dices recordé ver algo que tenía sentido, Si me ayudas a armarla puede ser una pista!

***-(La prueba consta de armar la nota a partir de fragmentos. Similar a los juegos de piñata de armar imágenes o secuencias numéricas corriendo fichas en un tablerito)-***



### ESCENA: Encontrando a Don Segundo

**- LA NOTA DICE:**

*“El tratamiento antituberculoso tiene que cumplir una serie de requisitos imprescindibles. En primer lugar, dado que existe cierto riesgo de que la tuberculosis se haga espontáneamente resistente a cualquiera de los fármacos utilizados, será necesaria la combinación de al menos dos fármacos para reducir este riesgo.”*

**Después de la mini prueba.**

**17- DS:** Excelente! Con la nueva información que tenemos podremos ayudar a los demás...!

• **Caso1.2.E:**

**Sdi.1 se acerca**

**18- Sdi.1:** Ajá!!! Los descubrí conspirando contra nuestro amo! Eso debe ser castigado!

**19- DS:** Pe...pe..pero señor guardia no!...

**20- Sdi.1:** Silencio!, no permitiré que dejen al descubierto el plan de mi señor Kash!

**21- DS:** Oh.. nos descubrieron y ahora que haremos!?

**22- Oritah:** Simón, esos guardias me robaron los cristales del tiempo que me regalo mi padre! Si los desafías a una prueba quizás es la única manera en que las puedas recuperar y con la ayuda de ellas salvarnos del malvado brujo!

**23- Simón:** No dejaré que le hagas más daño a mi gente guardia apestoso!

• **Caso1.2.F:**

**24- Sdi.1:** Ya es tarde enano!, he ordenado a los demás guardias rondar cerca de los debiluchos que quedan en este pueblucho! Es solo cuestión de minutos para que todos se empiecen a enfermar!!!! JUA, JUA, JUA

**25- Simón:** Don Segundo hay que actuar!, necesito algo para evitar que esos guardias contagien a los ciudadanos que se encuentran en el pueblo... los tapabocas!

**26- DS:** El tapabocas que uso cuando fumigo las plantas! Coincidentalmente lo tengo en el bolsillo, tómallo!, llévaselo a alguien y mientras vuelves buscaré más en mi casa... DATE PRISA!

**27- Oritah:** Simón!, tienes que encontrar a los 5 ciudadanos fuera de Don Segundo que hay ahora en el pueblo antes de que sea tarde!

**28- Sdi.1:** Ja,ja,ja que enano tan osado!... Hagamos un trato: Si en 60 segundos (*tiempo tentativo*) No has salvado a todos tus enclenques amiguitos y estás de vuelta frente a mí, daré la orden para que esparzan el virus de mi amo por todo el pueblo!!! JUA, JUA, JUA

**29- Oritah:** Simón recuerda los cristales!!

**30- Simón:** Y si lo logro deberás darme el cristal que llevas en tu bolsillo!

**31- Sdi.1:** QUE!?!? Pero como lo supist...Enano brujo!.. ehh... Pago por



ESCENA: Encontrando a Don Segundo	
	ver! ... TRATO HECHO!
<b>DIALOGOS LIBRES:</b>	<p>Los diálogos "1A" se refieren a cuando simón se ha acercado por primera vez al personaje no jugador. Y los diálogos "1B" es cuando se acercan por segunda vez a estos mismos.(1.2.B)</p> <p>1- <b>1A- Cno.1:</b> Nota de qué?... mmm... El alcalde da algunos permisos firmados para que los forasteros como tu puedan moverse por la ciudad... pero yo no tengo ninguno.</p> <p>1- <b>1B- Cno.1:</b> ... quizás el vecino lo sepa.</p> <p>2- <b>1A- Cno.2:</b> Permiso del alcalde?... no no no... yo no puedo dártelo... es el único que tengo!</p> <p>2- <b>1B- Cno.2:</b> Está bien... toma esta nota es el permiso. No le digas a nadie que yo te lo di!... Sólo lo hago porque algo extraño pasa con esto de la peste y de pronto puedas ayudarnos a descubrirlo.</p> <p><b>(1.2.B) Al entregarle la nota a Simón el personaje Cno.2 ya no vuelve a decir el diálogo 1B.</b></p> <p><u>Sdi.2 // Sdi.3 // Sdi.4</u></p> <p><b>(1.2.B)</b> Sí Simón aún no tiene la nota de autorización, los guardias hacen los mismos comentarios que en el caso: 1.1.B (En el caso 1.1 - cuadro anterior )</p> <p><b>Sí Simón ya tiene la nota de autorización entonces el diálogo de los guardias es:</b></p> <p>1- <b>Sdi.*:</b> Lo siento pequeñín pero ese pase que tienes no sirve para seguir por ésta sección del pueblo, muy probablemente te funcione en otra parte.</p>

Tabla 18 – ESCENA: Encontrando a Don Segundo

ESCENA: Obteniendo el Cristal	
<b>CASO:</b>	<b>2.1</b>
<b>CONTROLES:</b>	Funcionan completamente.
<b>PANTALLA:</b>	<p>Interfaz de juego con elementos adicionales necesarios para el desarrollo de la prueba: Cuenta regresiva (<i>parte media-superior de la pantalla</i>) y un ícono indicador de el avance de la prueba (<i>parte inferior-derecha; icono de un ciudadano con tapabocas</i>). El número de la izquierda es la cuenta de la prueba y el número de la derecha es el total para superar la prueba).</p>  <p>El ícono de "tapabocas" (<i>mitad-inferior</i>) solo sale cuando Simón carga un tapabocas, de lo contrario no sale nada (Ejemplo: <i>cuando ya lo ha entregado al ciudadano</i>)</p>
<b>NOTAS:</b>	En este punto Simón es vulnerable a los guardias.
<b>ACCIONES:</b>	EL jugador debe llevar tapabocas a los ciudadanos
<b>HITS DE CASO:</b>	<p><b>2.1.A.</b> Inicia la cuenta regresiva y El jugador inicia la prueba. Simón debe ir cada vez que entrega un tapabocas de nuevo hasta donde Don Segundo para que este le entregue otro.</p> <p><b>2.1.B.</b> Oritah habla con Simón</p> <p><b>2.1.C.</b></p> <p>-GAME OVER (<i>Ilustración</i>. Cuando Simón es alcanzado por un Sdi.* ó porque termino la cuenta regresiva.&lt;sale opción de repetir la prueba o Salir del juego con la misma gráfica del menú de opciones ingame&gt;)</p> <p>-PANTALLAZO DE VICTORIA (<i>Ilustración</i>. Cuando el jugador supera la prueba)</p> <p><b>2.1.D.</b> Apenas Simón sale del inventario y de nuevo está en la interfaz de</p>

ESCENA: Obteniendo el Cristal			
juego, sale un diálogo automático de Orah de opción múltiple..			
<b>RESTRICCIONES Y POSICIONES DEL MAPA:</b>	 <p>Las Franjas rojas son el desplazamiento de guardia que hacen los Sdi. Menos el Sdi.3 que tiene posición estática mirando hacia la izquierda (dándole la espalda al camino de la derecha).</p> <p><b>NOTA:</b> los recorridos son irregulares: Ejemplo: Mientras Sdi.1 está en el límite superior izquierdo de su recorrido el Sdi.5 Está en su límite inferior, mientras Sdi.5 se encuentra en la mitad de su recorrido.</p>		
<b>PERSONAJES DEL MODULO:</b>	<table border="0" style="width: 100%;"> <tr> <td style="width: 50%; vertical-align: top;"> <ul style="list-style-type: none"> <li>- Simón</li> <li>- Oritah</li> <li>- Don Segundo</li> <li>- Don Nicolás</li> <li>- Cno.1</li> <li>- Cno.2</li> <li>- Cno.3</li> </ul> </td> <td style="width: 50%; vertical-align: top;"> <ul style="list-style-type: none"> <li>- Cno.4</li> <li>- Sdi.1</li> <li>- Sdi.2</li> <li>- Sdi.3</li> <li>- Sdi.4</li> <li>- Sdi.5</li> </ul> </td> </tr> </table>	<ul style="list-style-type: none"> <li>- Simón</li> <li>- Oritah</li> <li>- Don Segundo</li> <li>- Don Nicolás</li> <li>- Cno.1</li> <li>- Cno.2</li> <li>- Cno.3</li> </ul>	<ul style="list-style-type: none"> <li>- Cno.4</li> <li>- Sdi.1</li> <li>- Sdi.2</li> <li>- Sdi.3</li> <li>- Sdi.4</li> <li>- Sdi.5</li> </ul>
<ul style="list-style-type: none"> <li>- Simón</li> <li>- Oritah</li> <li>- Don Segundo</li> <li>- Don Nicolás</li> <li>- Cno.1</li> <li>- Cno.2</li> <li>- Cno.3</li> </ul>	<ul style="list-style-type: none"> <li>- Cno.4</li> <li>- Sdi.1</li> <li>- Sdi.2</li> <li>- Sdi.3</li> <li>- Sdi.4</li> <li>- Sdi.5</li> </ul>		
<b>DIALOGOS FIJOS:</b>	<ul style="list-style-type: none"> <li>• <b>Caso 2.1.A:</b> Diálogos en Orden de cada una de las veces que Simón se acerca a DS para recoger un tapabocas (diálogos automáticos que se cierran después de unos segundos sin necesidad de hacer click del mouse o hundir una tecla adicional fuera de los controles de juego): (Segundo tapabocas) 1- <b>DS:</b> Toma este otro Simón, Voy a seguir buscando! (Tercer tapabocas) 2- <b>DS:</b> Creí que no tenía más pero aquí va un tercero! (cuarto tapabocas) 3- <b>DS:</b> Otro más falta sólo uno... Perrita parlante ayúdame a buscar!</li> </ul>		

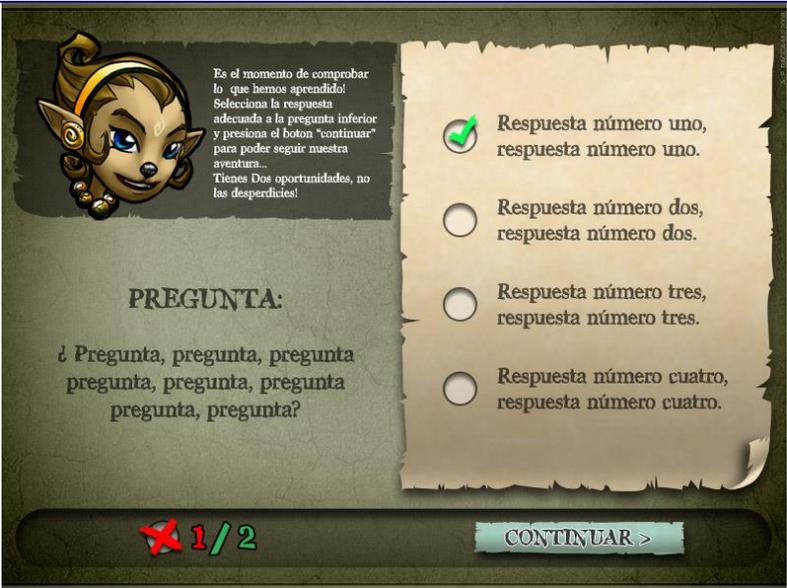


<b>ESCENA: Obteniendo el Cristal</b>	
	<p>(Quinto tapabocas)</p> <p>4- <b>DS:</b> Bendito seas muchachito tenía un último en el armario... apúrate!!</p> <p>5- <b>Oritah:</b> Simón ya casi!!! GUAUUU!!</p> <ul style="list-style-type: none"><li>• <b>Caso 2.1.B:</b> <b>Simón llega a donde Oritah y Don Segundo.</b></li></ul> <p>6- <b>DS:</b> Felicitaciones muchacho!! Lo lograste!. Ahora estamos Salvados</p> <p>7- <b>Sdi.1:</b> Noooo! Enlenque no tan enlenque me has ganado!!.. mi cristal!!... mi cristal!!... tómallo antes de que me arrepienta y te mande a la luna!</p> <ul style="list-style-type: none"><li>• <b>Caso 2.1.C:</b> PANTALLAZO CORRESPONDIENTE.</li><li>• <b>Caso 2.1.D:</b> <b>10- Oritah:</b> GUAUU GUAU!! Que bien Simón, deberías revisar en tu inventario la información contenida en la gema! (se abre automáticamente el inventario).</li></ul> <p><b>-INFORMACION DE LA GEMA DOS:</b> <i>“Debido a que tuberculosis requiere mucho tiempo para multiplicarse, con largos períodos de inactividad metabólica intracelular, el tratamiento antibiótico deberá ser siempre muy prolongado. Debe considerarse que la respuesta inmunitaria del huésped es esencial en el control de la infección, por lo que el tratamiento debería modificarse según la enfermedad de base del paciente.”</i></p> <p><b>Al salir del inventario:</b></p> <p>1- <b>Oritah:</b> Es hora de probar lo que hemos aprendido!, Estás listo para continuar?</p> <ul style="list-style-type: none"><li>- <b>Op1:</b> - No, aún no, déjame ver mi inventario de nuevo un momento (si presiona aquí entra automáticamente al inventario y al salir se hace automáticamente esta pregunta de nuevo)</li><li>- <b>Op2:</b> - Ya estoy listo Orah, Continuemos!</li></ul>

Tabla 19 – ESCENA: Obteniendo el Cristal

**10. Cuestionario de segundo nivel:**

- En este momento aparece en pantalla la interfaz de cuestionario.
- Los popups de los comentarios de Oritah salen sobre la interfaz de cuestionario.

ESCENA: Cuestionario	
<b>CASO:</b>	<b>3.1</b>
<b>CONTROLES:</b>	Funcionan completamente.
<b>PANTALLA:</b>	Interfaz de cuestionario
<b>NOTAS:</b>	No salen los personajes.
<b>ACCIONES:</b>	<p>El jugador debe seleccionar la opción correcta de entre las respuestas posibles. Si existen 5 ó 6 respuestas posibles entonces el jugador tiene 3 créditos. En tal caso debe haber un contrarreloj de mínimo 3 a 4 minutos para contestar.</p> <p><b>3.1.A.</b> Orah da un mensaje de bienvenida y explica cómo se usa la interfaz.  <b>3.1.B.</b> Se hace visible la pregunta y salen las posibles respuestas.  <b>3.1.C.a.</b> En caso de error Orah le dice algo en particular al jugador.  <b>3.1.C.b.</b> En caso de victoria, Orah le dice algo particular al jugador.  <b>3.1.D.a.</b> En caso de game over la prueba se reinicia.  <b>3.1.D.b.</b> En caso de victoria Orah da una nota (visible desde el inventario) a simón con información más detallada, sin embargo al terminar la prueba sale por primera vez en la ventanita de diálogo de Orah.</p>
<b>HITS DE CASO:</b>	
<b>RESTRICCIONES Y POSICIONES DEL MAPA:</b>	
<b>PERSONAJES DEL MODULO:</b>	- Simón
	- Orah
<b>CASOS:</b>	<p><b>Caso 3.1.A.</b></p> <p>- <b>Orah:</b> “Es el momento de comprobar lo que hemos aprendido!”</p>



ESCENA: Cuestionario	
	<p>Selecciona la respuesta más adecuada a la pregunta inferior y presiona el botón continuar para poder seguir nuestra aventura... Tienes sólo Dos oportunidades, no las desperdicies!</p> <p><b>Caso 3.1.B.</b></p> <ul style="list-style-type: none"><li>- “Pregunta: Una de las siguientes opciones es la manera más efectiva de evitar contagiarnos por la tuberculosis pulmonar:”</li></ul> <p><b>Posibles respuestas:</b></p> <ul style="list-style-type: none"><li>- 1- Dejar de cantar con mis amigos.</li><li>- 2- Usar tapabocas para evitar el contagio por el aire. <i>(esta es la respuesta)</i></li><li>- 3- No respirar frente a personas con tuberculosis.</li><li>- 4- Evitar tener heridas.</li></ul> <p><b>Caso 3.1.C.a.</b></p> <ul style="list-style-type: none"><li>- <b>Orah:</b> Pequeño Simón! Recuerda lo que descubrimos en nuestro camino! Creo que deberías intentar otra respuesta!.</li></ul> <p><b>Caso 3.1.C.b.</b></p> <ul style="list-style-type: none"><li>- <b>Orah:</b> Lo hiciste perfectamente! Definitivamente creo que esta aventura nos va a dejar una gran experiencia!.</li></ul> <p><b>Caso 3.1.D.a.</b></p> <ul style="list-style-type: none"><li>- <b>Orah:</b> Nooo! Simón ahora que vamos a hacer! Eras la única esperanza que teníamos!</li><li>- <b>Game over.</b></li></ul> <p><b>Caso 3.1.D.b.</b></p> <ul style="list-style-type: none"><li>- <b>Orah:</b> Que bien!, mientras tu respondías el cuestionario encontré esta nota! posteriormente podrás leerla en el inventario de nuevo.</li><li>- <b>Orah:</b> “.....”</li><li>- <b>Orah:</b> Perfecto!, ahora... es momento de continuar nuestro camino.</li></ul>
<b>CUESTIONARIOS:</b>	

Tabla 20 – ESCENA: Cuestionario



ESCENA: Desenlace									
<b>CASO:</b>	<b>4.1</b>								
<b>CONTROLES:</b>	No funcionan. Sólo para avanzar los diálogos								
<b>PANTALLA:</b>	Escenario de juego. Junto al lugar donde se desarrollo la prueba.								
<b>NOTAS:</b>	Todos los ciudadanos se encuentran alrededor de Simón en frente de la Casa Mayor.								
<b>ACCIONES:</b>	Los ciudadanos le agradecen a Simón lo que hizo. Ahora es tiempo de que Simón vuelva a su tiempo real.								
<b>HITS DE CASO:</b>	<b>4.1.A.</b> Los aldeanos que Simón acaba de ayudar lo rodean en círculo y lo felicitan.								
<b>RESTRICCIONES Y POSICIONES DEL MAPA:</b>									
<b>PERSONAJES DEL MODULO:</b>	<table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none;">- Simón</td> <td style="width: 50%; border: none;">-Cno.1</td> </tr> <tr> <td style="border: none;">-Oritah</td> <td style="border: none;">-Cno.2</td> </tr> <tr> <td style="border: none;">- Don Segundo</td> <td style="border: none;">-Cno.3</td> </tr> <tr> <td style="border: none;">-Don Nicolás</td> <td style="border: none;">-Cno.4</td> </tr> </table>	- Simón	-Cno.1	-Oritah	-Cno.2	- Don Segundo	-Cno.3	-Don Nicolás	-Cno.4
- Simón	-Cno.1								
-Oritah	-Cno.2								
- Don Segundo	-Cno.3								
-Don Nicolás	-Cno.4								
<b>DIALOGOS FIJOS:</b>	<ul style="list-style-type: none"> <li>• <b>Caso 4.1.A:</b> <b>Automáticamente termine la prueba:</b></li> <li>1- <b>Oritah:</b> Felicitaciones Simón! Lo hiciste!... estuviste genial... fantástico!! GUAUU GUAUU!!</li> <li>2- <b>DS:</b> Todos un Hurra por Simón!!</li> <li>3- <b>Cno.1/Cno.2/Cno.3/Cno.4/DN:</b> HURRA!! HURRA!! HURRA!!</li> <li>4- <b>DS:</b> Has hecho por nosotros más de lo que cualquier otra persona ha podido hacer. Haz desenmascarado al malvado brujo y descubierto la enfermedad!.</li> <li>5- <b>DN:</b> Ahora sabemos que se trata de la tuberculosis y gracias a ti tenemos como protegernos!</li> <li>6- <b>Oritah:</b> Simón, yo creo que ya es tiempo de que vuelvas. Los demás están a punto de llegar.</li> <li>7- <b>Simón:</b> Pero... quiero ver a mi madre y ... conocer a mi pap...</li> <li>8- <b>Oritah:</b> Lo siento Simón. Pero una de las instrucciones que me dió la perrita viejita fue que no debías esperar. Que las cosas tienen su tiempo y así desearas lo que desearas... debes hacer lo que debes hacer.</li> <li>9- <b>DS:</b> Pequeño... El destino es sabio, quizás esta perrita parlante tenga razón. Las cosas deben ser como deben ser... No se que más decir. En nombre de todos te deseamos muchísima suerte a donde quiera que vayas.</li> <li>10- <b>Orah:</b> Simón. Trae los cristales.</li> </ul>								
<b>DIALOGOS</b>									



ESCENA: Desenlace	
<b>LIBRES:</b>	

Tabla 21 – ESCENA: Desenlace

### **ESCENA 5:**

**Caso: 5.1. PANTALLA:** *Secuencia de imágenes con textos.*

**IMAGEN 1:** Simón pone los cristales sobre un círculo mágico que Oritah está terminando de dibujar. Están rodeados por los aldeanos cogidos de la mano.

**Oritah:** Ya está el círculo mágico. Simón pon las pruebas en el suelo...

**IMAGEN 2:** Simón empieza a elevarse del suelo flotando en “materia” de luz con vapor, está por encima de las cabezas de los ciudadanos que hacen el círculo.

**Simón:** Oritah!... los voy a extrañar!

**DS:** Seguro, nosotros también! Te estaremos agradecidos para toda la vida!

**IMAGEN 3:** Simón se percata a lo lejos que viene su madre con su padre abrazados y los demás aldeanos de trabajar.

**Simón:** papá!....

**IMAGEN 4:** Pantalla en blanco

**Orah:** Simón. Tu destino empieza a escribirse desde este momento... Eres todo un Héroe. Tanto en el pasado como en tu presente y en tu futuro estaremos agradecidos contigo.

**Simón:** Orah?... y.. que me espera?

**Orah:** ten calma pequeño... Mi padre ha visto lo que has hecho y está muy complacido. Ve tranquilo a tu casa que te espera una sorpresa.

### **ESCENA 6: ENDING(Animación 2D ó 3D):**

#### **ESCENA 6.1: exterior- casa de Simón.**

Simón aparece en su ciudad... frente a la puerta de su casa. El aire es puro; hay flores silvestres por todos lados alrededor de su casa. Revuelan pájaros por encima de su cabeza en el cielo lejano. Simón mira alegremente todo lo que sucede con ojos iluminados y una gran sonrisa en su cara. Hasta que su mamá lo llama desde dentro de la casa (se sigue viendo desde dentro)

- **Esperanza:** *Simón la cena está lista!*

Simón se emociona y corre hasta la puerta de su casa.

#### **ESCENA 6.2: interior- casa de Simón. Contraluz**

**Toma 1:**

La puerta se abre y en un dolly vertical de abajo hacia arriba se alcanza ver que allí está la mamá de Simón colocando un plato sobre la mesa (todo es blanco y negro hasta aquí por el fuerte contraluz).

**Toma2:**

Close up de Simón con cara de felicidad extrema.

- **Simón:** *Mamá... estas aquí y estás bien!!!*

#### **ESCENA 6.3: Exterior- casa de Simón.**



Un plano completo de la casa de Simón a ras de piso hasta que unos pies se posan frente a la cámara dándole la espalda.

- **Papa de Simón:** *Hijo! Ven y dale un abrazo a tu padre.*

#### **ESCENA 6.4: Interior\_casa de simón- closeup extremo.**

Los ojos de Simón tiemblan. El contraste no es tan fuerte sin embargo la imagen sufre una transición desde el blanco enceguecedor que entra por el marco de la puerta y un fondo donde sale muy atrás la imagen de su padre parado con las manos entre abiertas esperando a que Simón llegue.

Después de eso hay un fade out hasta blanco lentamente.

- **Simón:** *PAPÁ!!!!*

#### **ESCENA 6.5: Exterior-Casa de Simón.**

Al inicio hay un fade in desde blanco a opaco lentamente.

Simón está sentado en el techo de su casa junto a su padre y a su madre (de espaldas a la pantalla) viendo el atardecer.

- **Esperanza:** *Simón... A dónde has estado todo este tiempo? Tu abuelo estuvo buscándote toda la tarde y como no te encontró por ningún lado se fue a donde sus amigos.*
- **Simón:** *Es una muy larga historia...*
- **Papá de Simón:** *No importa muchacho! Tendremos toda la noche para que nos cuentes! Tu abuelo vendrá después de la partida de cartas que organizaron Don Nicolás y Don Segundo.*
- **Simón:** *No se si quiera hablar de eso... es algo triste.*
- **Esperanza:** *Ah bueno... Si no quieres entrar en detalles está bien. Sin embargo necesitamos saber en dónde estabas! Somos tus padres nos preocupamos por ti!*
- **Papá de Simón:** *Hijo, alguna vez mi padre de pequeño me recitó un verso muy bonito que escribió una vez que estaba muy triste... quizás este sea un buen momento para traerlo de vuelta:*

Extraños agravios inundan mis noches, entre sueños y llantos mi alma se corrompe...  
Pero... qué es eso!?, si los agravios son malos! el dolor injusto así el sufrir sea de humanos!

Alma querida solo me queda decirte que de quebrantos y tropiezos el hombre se forja

Te dedico el atardecer cuando el sol nos deja oculto tras las montañas,  
mientras sonrío aquel gran ojo rojo creyendo que te engaña... Pero no!...  
no te engaña!!

No sufras ni derrames tus lágrimas por el sol que nos deja.  
No creas que la noche es eterna y el alba una utopía lejana.  
No dejes que el miedo te enceguezca y se vuelva una trampa.

Una trampa para olvidar que anhelas ver el sol de nuevo.  
Cuando Realmente es inevitable que bello y grande salga  
de nuevo cada mañana.

- **Simón:** *Suena muy bonito pero... que quiere decir?*



- **Papá de Simón:** *Ja, ja,ja!!... Es un poco complicado de entender cierto?... por muchos años no entendí lo que decía. Pero hoy creo que lo he comprendido completamente: Después de la noche siempre llega el día. Las tristezas no son para siempre pues en el momento menos esperado la alegría intensa llenará tu vida.*
- **Simón:** *aah... claro que sí.. no te imaginas lo feliz que soy.*
- ...
- **Simón:** *Soy el niño más feliz del mundo!*

### C.1.3. Definición de reglas

El videojuego tiene un carácter progresivo; el sistema de reglas puede ser planteado de tal manera que el usuario pueda fácilmente evolucionar las pruebas. En un caso podría proponer que se juegue con un modelo de “desbloqueo” de pruebas. Es decir: que al resolver la prueba número uno, se habilite según un menú la posibilidad de que los jugadores puedan jugar las pruebas tanto dos, tres y repetir la uno sin un orden predeterminado, y al superar estas tres puedan escoger entre la cuatro, cinco y seis del mismo nivel(similar a los juegos mentales ‘*braingames*’ ). Un modelo de este tipo puede ser práctico para el usuario en el momento en que el deseo simplemente navegar, pero siguiendo un modelo argumental como tenemos puede ser realmente imposible ya que quebraría la historia. Puede bien estudiarse una manera mucho más original de plantear este sistema, sin embargo no creo prudente que entre en consideración ya que nuestro proyecto se ciñe más a un modelo tradicional *de videojuego comercial* que a un producto experimental que requeriría una investigación adicional. Lo que sí me parece realmente considerable es el planteamiento de tal manera que cada nivel pueda ser repetido indefinidamente. Recurrir a pruebas que inviten al jugador a tener el mayor puntaje puede hacer que el usuario adquiera un carácter menos crítico frente a las reglas y más propenso a aceptarlas (*Jesper Juul*), esto además le daría más “vida” y justificaría la repetición de las pruebas que le gusten al jugador (si ignoramos momentáneamente el proceso de comunicación).

#### C.1.3.1. Navegación

La Interfaz de juego debe tener elementos como un Mini-mapa, indicador de tiempo, indicador de vida (contrarreloj de pruebas), indicador de puntajes(que puede ser además un identificador gráfico referente a los objetos que se requieren para superar el nivel) Y el Inventario (Donde se guardan y pueden visualizar estos objetos recogidos).

Los textos de soporte y los diálogos deben ser desplegados en popups, Que son tanto a pantalla completa como a modo de etiquetas (juegos RPG) respectivamente.

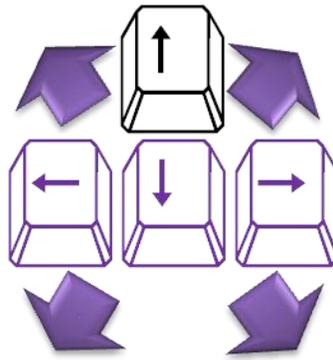
Al inicio de cada nivel debe salir un texto que contextualice al jugador, así mismo cada prueba debe tener un texto a pantalla completa que contextualice al jugador; a modo de “objetivos de la misión” (como en los juegos de estrategia en modo “historia”) antes de dar inicio tanto al nuevo nivel como a la nueva prueba.

La navegación general debe ser simple, con pantallas automáticas. Los textos de soporte no pueden ser obviados, es obligatorio verlos, mientras los diálogos y las secuencias de imágenes se secuencian por medio del clic izquierdo del mouse ó por una tecla especial.

### C.1.3.2. Controles

Las entradas deben ser totalmente controlables sólo con el teclado. Ya que no amerita el uso de más entradas puesto que la complejidad en el manejo es relativamente sencilla. Inicialmente, las teclas que se han configurado para el control del personaje de Simón son:

Los controles de desplazamiento: La Figura\_3 – Controles de desplazamiento muestra los botones del teclado que permiten controlar los movimientos del Simón sobre el terreno.



Figura\_3 – Controles de desplazamiento

- La tecla Esc: Despliega una ventana de menú, y además mantiene el juego pausado.



Figura\_4 – Botón de Esc

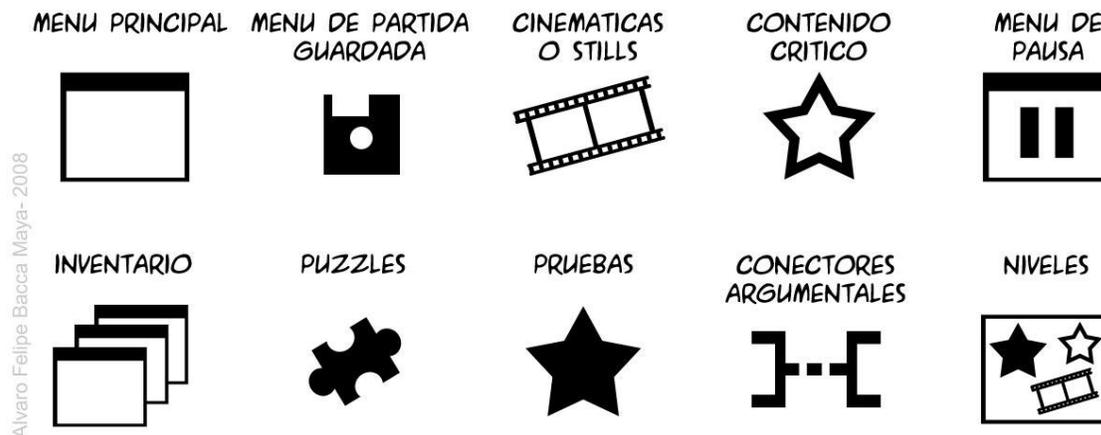
Simón no requiere saltar, por tal motivo debe hacer los recorridos evitando obstrucciones en el camino como piedras y/o cercos.

### C.1.3.3. Estados Del Personaje

- El único personaje jugador es Simón, Su movimiento en todas las direcciones de la pantalla. Tanto horizontal, vertical como diagonal (por ejemplo: *superior + izquierda*).
- El movimiento de los diálogos se muestra mientras los diálogos son dispuestos en la pantalla (sin importar el personaje que está hablando) acompañados de una cara ilustrada del personaje que habla.
- El movimiento de “acción” puede ser particularmente útil en el caso de que deseemos que al recoger un objeto o presionar sobre algún personaje antes de hablar con él y cualquier evento relacionado directamente con *accionar*.
- Simón no requiere saltar, por tal motivo debe hacer los recorridos evitando obstrucciones en el camino como piedras y cercos.

### C.1.3.4. Elementos de Interfaz Gráfica

A continuación se describen los elementos que componen el videojuego (Figura\_5 – Iconos Elementos del Juego.). Tanto pantallazos únicos como contenedores que se refieren a las situaciones generales para posteriormente plantear un diagrama de funcionamiento (guión técnico), identificándolos con un ícono particular.



Figura\_5 – Iconos Elementos del Juego.

- Menú Principal ( “partida nueva”, “partida guardada”, “opciones” y “salir”): Primer contacto del Jugador con la gráfica y la estética del juego, además que es el menú de selección de modo, tanto para iniciar como para seguir una partida o salir del juego. El menú de opciones contiene la opción “controles” para que el usuario escoja como desea jugar: si con mouse o teclado. Además de “instrucciones” para tener acceso a una infografía ó a un video tutorial del manejo de los controles del juego y las convenciones tanto de íconos como de objetos. (Otra manera puede ser ver las instrucciones al inicio del nuevo juego como un elemento arcade añadido), que también sea accesible desde el menú de opciones en medio de la partida.



- Menú De Partida Guardada: Menú sencillo por filas donde se puede escoger una partida guardada entre una cantidad determinada.
- Cinemáticas y/o Stills (Secuencia De Imágenes) Sobre La Historia: Espacio introductorio e intermedio de contextualización del usuario con la ficción. Argumentadores y motivadores para que el jugador se interese a explorar y comprenda la situación en la que se ve envuelto, correlacionando los eventos surgidos a través de todo el juego.
- Contenido Crítico: Cortinillas ó popups a pantalla completa de contenido textual explicativo e inductivo referente por completo a la tuberculosis (textos referidos al personaje tutor, en este caso a Orah: la amiga de Simón).
- Menú In-Game Ó Menú De Pausa (menú de opciones en medio de la partida): El menú In-game usualmente contiene opciones de configuración del videojuego, en nuestro caso contiene las mismas opciones que el menú principal. Se ejecuta preferiblemente con la tecla “return (enter)”.
- Inventario: El personaje debe coleccionar cierta cantidad de elementos a través del videojuego: objetos y conocimientos; este inventario contiene las piezas necesarias para llegar al próximo nivel y permite que el jugador tenga acceso a ellas cuando lo desee. El inventario no sólo debe tener casillas para almacenar dichos objetos, sino también la opción de presionar sobre ellos y poder visualizar su contenido cuantas veces lo quieran. Así como ver fichas de los personajes y aprender de ellos (quizás sean ítems extras que se encuentren en el camino y les dé una descripción de cada personaje ó tipo de personaje). También podría ser que cada vez que le jugador se encuentra una prueba, se le dé una ficha con las descripciones (fichas técnicas del caso) de cada personaje ó de los personajes de dicha prueba para que en un futuro el pueda más fácil saber cómo enfrentarse a los síntomas (segundo nivel del juego). Así entonces son dos tipos de ítems: los “cristales de conocimiento” y las “fichas de personaje”
- Puzzles: Es la parte mínima donde se ejecuta el ejercicio del juego como tal. Ejercicios rápidos, de lógica y contrarreloj.
- Pruebas: Las pruebas son el evento principal de todo el videojuego; y se interconectan entre unas y otras con algunos movimientos en el escenario que denomino “conectores argumentales”.
- Conectores Argumentales: Son los espacios argumentales que interconectan las pruebas; son aquellos diálogos ó movimientos que se usan para lograr continuidad en el argumento, además de ser una herramienta de distensión entre pruebas.
- Niveles: Los niveles son los grandes contenedores del videojuego. En este caso particular dividido en dos que son sobre el “tratamiento” y la “prevención”. Dentro de ellos se combinan una serie de pruebas, stills y contenidos críticos.

### C.1.3.5. Diagramas de Composición de “Simón y la Tuberculosis”

#### C.1.3.5.1. Estructura Del Juego (general)



Figura 6 - Estructura del Juego

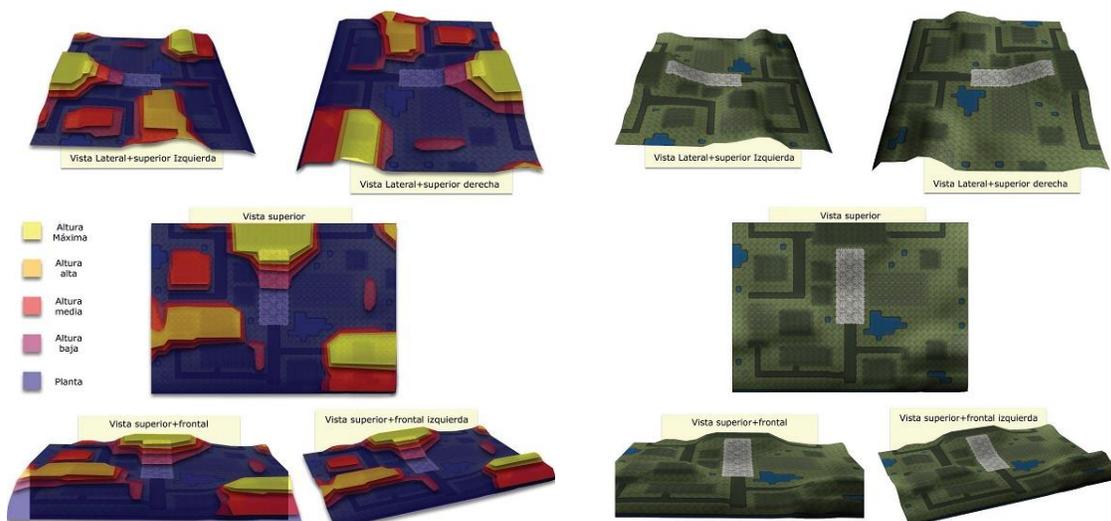
Es indispensable ya aclarados los diagramas de composición, definir el mapa de juego: definir las pruebas y desarrollar descriptivamente la secuencia jugable. Sabemos previamente el tema de cada prueba, sin embargo en este punto voy a estructurar la secuencia de juego mientras las pruebas serán definidas en un próximo documento:

*NOTA: si el jugador pierde el contrarreloj de las pruebas ó pierde 2 ó 3 veces consecutivas en el cuestionario, la prueba vuelve a iniciar infinitamente sin gameover (sin embargo podemos analizar si realmente vale la pena hacer gameover, en tal caso lo ponemos).*

- *Segunda prueba:* Simón desarrolla la segunda prueba que tiene que ver con: “la forma más sencilla de prevenir la transmisión de la tuberculosis pulmonar”.
- *Cuestionario de segundo nivel:* Se continúa la misma lógica del cuestionario del nivel anterior pero aplicado a la prueba del segundo nivel.
- *Desenlace argumental:* Simón ha superado las pruebas, tiene que volver a su presente.
- *Stills final de juego:* Simón vuelve a su presente y se encuentra con un lugar maravilloso lleno de flores y gente sana, regresa a su casa y descubre que todo está perfecto, que su padre no ha muerto, en ese momento el lo invita a un cerro y mirando la luna le recita un viejo verso (sobre un nuevo amanecer), allí los dos sentados en un plano panorámico le dan fin a esta historia.

### C.1.4. Definición del mundo

Para el videojuego “Simón y la Tuberculosis” – nivel 2, se define un terreno abierto porque los sucesos definidos tienen lugar en el pueblo y todos los escenarios planteados se llevan a cabo al aire libre en el pueblo San José de las Piedras. La (Figura\_7 – Bocetos del terreno) es un boceto del terreno en el cual se enfatiza su sistema montañoso (alturas), y los diferentes caminos del pueblo.



Figura\_7 – Bocetos del terreno

El terreno cuenta con los siguientes objetos que son considerados para el jugador como obstáculos al momento de desplazarse, entre los cuales se tiene:

- Las Casa de campesinos
- El Castillo del alcalde
- Aljibes
- Cercas
- Piedras
- Arboles

Los cuales se encuentran distribuidos de la siguiente manera sobre el terreno (Figura\_8 – Ubicación Unidades Terreno).



Figura\_8 – Ubicación Unidades Terreno

### C.1.5. Identificación de Personajes

A partir de la comprensión del storyline “Simón & La Tuberculosis”, se encontraron los siguientes personajes:

IDENTIFICACION PERSONAJES/AGENTES	
<b>Id</b>	IDPP01
<b>Nombre</b>	SIMON
<b>Bocetos</b>	
<b>Personajes con los que interactúa:</b>	<ul style="list-style-type: none"> <li>- ORITAH</li> <li>- CAMPESINOS</li> <li>- SOLDADOS</li> <li>- DON SEGUNDO</li> </ul>
<b>Descripción:</b>	<p>Simón es el personaje-jugador, esto lo convierte en el personaje principal de la historia, este personaje es la representación del usuario en el juego (es el personaje que el Usuario controla). Este personaje realiza un viaje a través del tiempo, para regresar al pasado con la tarea de prevenir una enfermedad que en el futuro está acabando con su familia y los campesinos que habitan su pueblo. Para ello cuenta con la guía de un personaje acompañante el cual está representada en una cachorrita llamada Oritah.</p>
<b>Tipo:</b>	Principal
<b>Referencias cruzadas</b>	Análisis del argumento base

Tabla 22 – Identificación Simón

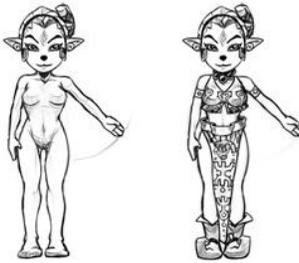
IDENTIFICACION PERSONAJES/AGENTES	
<b>Id</b>	IDPS02
<b>Nombre</b>	ORITAH
<b>Bocetos</b>	<p>Orah.</p> <p>(Version perrito/labrador chocolate)</p>  
<b>Personajes con los que interactúa:</b>	<ul style="list-style-type: none"> <li>- SIMON</li> <li>- SOLDADOS</li> </ul>
<b>Descripción:</b>	Oritah es un personaje no jugador la mayoría del tiempo se encuentra junto a Simón ya que es su personaje acompañante, es la encargada de guiarlo en su aventura proveyéndole pista que le ayuden en su odisea, este personaje puede ser un personaje jugador al hacer clic sobre ella. Si esta cerca de un soldado ladra.
<b>Tipo:</b>	Secundario
<b>Referencias cruzadas</b>	Análisis del argumento base

Tabla 23 – Identificación Oritah

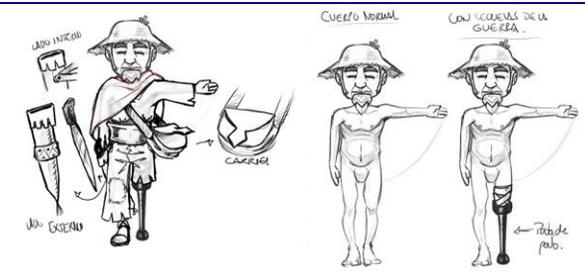
IDENTIFICACION PERSONAJES/AGENTES	
<b>Id</b>	IDPS03
<b>Nombre</b>	DON SEGUNDO
<b>Bocetos</b>	
<b>Personajes con los que interactúa:</b>	<ul style="list-style-type: none"> <li>- SIMON</li> <li>- SOLDADO</li> </ul>
<b>Descripción:</b>	Don Segundo es un personaje no jugador, que sirve de tutor a Simón en determinados puntos de la historia proporcionándole las pistas que le guiaran a alcanzar sus objetivos.
<b>Tipo:</b>	Secundario
<b>Referencias cruzadas</b>	Análisis del argumento base

Tabla 24 – Identificación Don Segundo

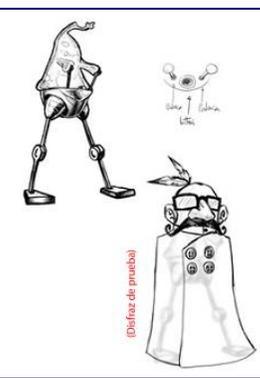
IDENTIFICACION PERSONAJES/AGENTES	
<b>Id</b>	IDPE04
<b>Nombre</b>	SOLDADOS
<b>Bocetos</b>	
<b>Personajes con los que interactúa:</b>	<ul style="list-style-type: none"> <li>- ORITAH</li> <li>- SOLDADO</li> </ul>
<b>Descripción:</b>	Los soldados son personaje no jugador, son súbditos del emperador Kash y siguen todas sus órdenes, entre las cuales están, mantener al pueblo en las condiciones propicias para que la enfermedad pueda propagarse. Uno de sus principales objetivos es impedir que Simón cumpla con sus objetivos.
<b>Tipo:</b>	Extra
<b>Referencias cruzadas</b>	Análisis del argumento base

Tabla 25 – Identificación Soldados

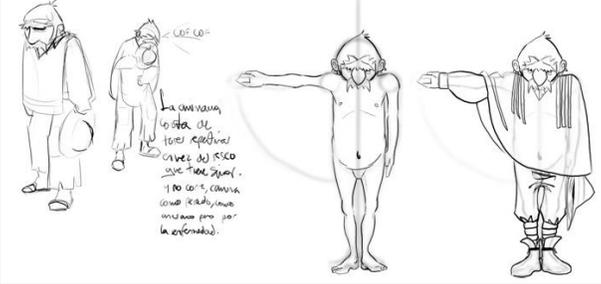
IDENTIFICACION PERSONAJES/AGENTES	
<b>Id</b>	IDPE05
<b>Nombre</b>	CAMPESINOS
<b>Bocetos</b>	 <p>La animación está de por repetir cuando el chico está mal. Simón. No es, cuando está mal, como está mal por la enfermedad.</p>
<b>Personajes con los que interactúa:</b>	- SIMÓN
<b>Descripción:</b>	Los campesinos son personajes no jugador, la apariencia de los campesinos cambia si su estado es sano o si están enfermos, esto influye en que un campesino enfermo no puede mantener conversaciones con Simón dada su enfermedad mientras que un campesino sano en ocasiones podrá brindar alguna información a Simón y en determinados momentos podrá obsequiarle ítems que ayuden a Simón.
<b>Tipo:</b>	Extra
<b>Referencias cruzadas</b>	Análisis del argumento base

Tabla 26 – Identificación Campesinos

### C.1.6.Descripción de los Personajes

A continuación es necesario profundizar en los comportamientos y funciones de cada uno de los personajes.

DESCRIPCION PERSONAJES/AGENTES	
<b>ID:</b>	DSPP01
<b>Nombre:</b>	SIMON
<b>Boceto:</b>	
<b>Personajes con los que interactúa:</b>	<ul style="list-style-type: none"> <li>- ORITAH</li> <li>- CAMPESINOS</li> <li>- SOLDADOS</li> <li>- DON SEGUNDO</li> </ul>
<b>Comportamiento:</b>	<ul style="list-style-type: none"> <li>- Encontrar a Oritah.</li> <li>- Encontrar a Don Segundo.</li> <li>- Obtener la gema.</li> </ul>
<b>Dimensiones Física – Social - Psicológica:</b>	<p>Simón es un niño tímido, que aspira poder ayudar a su comunidad en la batalla contra la peste que aqueja su mundo. Así como lo fue su padre, Simón quiere ser un héroe, sin embargo su corta edad no le permite ser gran protagonista, si de guerra física se tratara, su gran ventaja es su inteligencia e ímpetu. Es de pocos amigos pero a través de la historia, descubrirá que es tan valioso en si mismo que dejara atrás la timidez y será un gran líder. Sin embargo ahora su gran curiosidad, pasión e imaginación, sumado con las inmensas ganas de ayudar a su gente, son sus únicas pero valiosas armas para salvar a su pueblo.</p>
<b>Tipo:</b>	Principal
<b>Referencias cruzadas:</b>	<p>StoryLine – Simon y la tuberculosis – Nivel 2 Tabla 22 – Identificación Simón</p>

Tabla 27 – Descripción Simón

DESCRIPCION PERSONAJES/AGENTES	
<b>ID:</b>	DSPS02
<b>Nombre:</b>	ORITAH
<b>Boceto:</b>	
<b>Personajes con los que interactúa:</b>	<ul style="list-style-type: none"> <li>- SIMON</li> <li>- SOLDADOS</li> </ul>
<b>Comportamiento:</b>	- Encaminar la aventura de Simón.
<b>Dimensiones Física – Social - Psicológica:</b>	Oritah es una criatura mágica, la cual suele vivir entre las personas debido a que tiene un gusto especial por las situaciones que se presentan diariamente ya que las encuentra divertidas. En esta ocasión ha decidido servir de guía a un niño que está destinado a algo importante para su mundo; Vive bajo la forma de un labrador chocolate y es la mascota de Simón.
<b>Tipo:</b>	Secundario
<b>Referencias cruzadas:</b>	StoryLine – Simon y la tuberculosis – Nivel 2 Tabla 23 – Identificación Oritah

Tabla 28 – Descripción Oritah

DESCRIPCION PERSONAJES/AGENTES	
<b>ID:</b>	DSPS03
<b>Nombre:</b>	DON SEGUNDO
<b>Boceto:</b>	
<b>Personajes con los que interactúa:</b>	<ul style="list-style-type: none"> <li>- SIMON</li> <li>- SOLDADO</li> </ul>
<b>Comportamiento:</b>	- Encaminar la aventura de Simón.
<b>Dimensiones Física – Social - Psicológica:</b>	Don Segundo es un campesino arraigado de naturaleza, Comandante de tropa en época de guerra, uno de los pocos sobrevivientes de la batalla donde el papa de Simón perdió la vida. De temple y carácter fuerte.
<b>Tipo:</b>	Secundario
<b>Referencias cruzadas:</b>	StoryLine – Simon y la tuberculosis – Nivel 2 Tabla 25 – Identificación S Tabla 24 – Identificación Don Segundo

Tabla 29 – Descripción Don Segundo

DESCRIPCION PERSONAJES/AGENTES	
<b>ID:</b>	DSPE04
<b>Nombre:</b>	SOLDADOS
<b>Boceto:</b>	
<b>Personajes con los que interactúa:</b>	<ul style="list-style-type: none"> <li>- ORITAH</li> <li>- SOLDADO</li> </ul>
<b>Comportamiento:</b>	- Mantener a la gente del pueblo enferma.
<b>Dimensiones Física – Social - Psicológica:</b>	<p>La milicia completa del emperador Kash, está formada por bacterias desprendidas de su mismo cuerpo, algo así como sus hijos.</p> <p>Son los encargados de mantener los niveles de salubridad bajos y de vigilar constantemente a la población, además de ser sus propios soldados. Prácticamente todos son exactamente iguales y andan cubiertos con un disfraz que los hace parecer una persona común.</p>
<b>Tipo:</b>	Extra
<b>Referencias cruzadas:</b>	<p>StoryLine – Simon y la tuberculosis – Nivel 2</p> <p>Tabla 25 – Identificación Soldados</p>

Tabla 30 – Descripción Soldado

DESCRIPCION PERSONAJES/AGENTES	
<b>ID:</b>	DSPE05
<b>Nombre:</b>	CAMPESINOS
<b>Boceto:</b>	
<b>Personajes con los que interactúa:</b>	- SIMON
<b>Comportamiento:</b>	
<b>Dimensiones Física – Social - Psicológica:</b>	Los campesinos son todas las personas natales del pueblo de Simón, muchos de los cuales se encuentran bajo los síntomas de la enfermedad que agobia el pueblo, son de carácter muy humilde.
<b>Tipo:</b>	Extra
<b>Referencias cruzadas:</b>	StoryLine – Simon y la tuberculosis – Nivel 2 Tabla 26 – Identificación Campesinos

Tabla 31 – Descripción Campesino

### C.1.7. Identificación de los casos de uso de los personajes

En este ítem se realizara una breve descripción de las actividades realizadas por los personajes.

CASOS DE USO	
<b>Id:</b>	CU01
<b>Nombre de la Actividad:</b>	Aguardar
<b>Personajes:</b>	Oritah – Aldeanos – Don Segundo - Soldados
<b>Resumen:</b>	Los personajes no jugador mantienen en espera de que Simón se acerque a ellos para ejecutar alguna acción. La acción ejecutada depende del personaje al que se acerque Simón.
<b>Referencias Cruzadas:</b>	Tabla 17 – ESCENA: Inicio Tabla 18 – ESCENA: Encontrando a Don Segundo

Tabla 32 – Caso de Uso Aguardar



CASOS DE USO	
<b>Id:</b>	CU02
<b>Nombre de la Actividad:</b>	Acompañar a Simón
<b>Personajes:</b>	Oritah – Don Segundo
<b>Resumen:</b>	Una vez Simón se encuentra con alguno de estos personajes, estos deben seguirlo, además de facilitarle información acerca de las formas de prevención de la tuberculosis.
<b>Referencias Cruzadas:</b>	Tabla 17 – ESCENA: Inicio Tabla 18 – ESCENA: Encontrando a Don Segundo

Tabla 33 – Caso de Uso Acompañar a Simón

CASOS DE USO	
<b>Id:</b>	CU03
<b>Nombre de la Actividad:</b>	Patrullar
<b>Personajes:</b>	Soldados
<b>Resumen:</b>	Cuando Simón se acerca a algún aldeano que se encuentre a corta distancia del soldado este empezara a patrullar, realizando un desplazamiento que impida el paso de Simón.
<b>Referencias Cruzadas:</b>	Tabla 18 – ESCENA: Encontrando a Don Segundo Tabla 19 – ESCENA: Obteniendo el Cristal

Tabla 34 – Caso de Uso Patrullar

CASOS DE USO	
<b>Id:</b>	CU04
<b>Nombre de la Actividad:</b>	Repartir Tapabocas
<b>Personajes:</b>	Aldeanos – Don Segundo
<b>Resumen:</b>	Después de que Simón se encuentra con Don Segundo debe retar a un soldado para que este le dé un tiempo límite para entregar tapabocas a los aldeanos. Los aldeanos que se encuentren con Simón deben recibir el tapabocas.
<b>Referencias Cruzadas:</b>	Tabla 19 – ESCENA: Obteniendo el Cristal

Tabla 35 – Caso de Uso Repartir Tapabocas



CASOS DE USO	
<b>Id:</b>	CU05
<b>Nombre de la Actividad:</b>	Encontrar a Oritah
<b>Personajes:</b>	Simón – Oritah – Aldeanos – Soldados
<b>Resumen:</b>	Simón debe desplazarse por el terreno hasta encontrar su casa, de esta forma tendrá su primer encuentro con Oritah que es el personaje que lo guiara en su aventura, en su camino a casa puede tropezarse con Aldeanos o Soldados.
<b>Referencias Cruzadas:</b>	Tabla 17 – ESCENA: Inicio

Tabla 36 – Caso de Uso Encontrar Oritah

CASOS DE USO	
<b>Id:</b>	CU06
<b>Nombre de la Actividad:</b>	Encontrar a Don Segundo
<b>Personajes:</b>	Simón – Oritah – Don Segundo – Aldeanos – Soldados
<b>Resumen:</b>	Simón acompañado de Oritah debe dirigirse hasta donde se encuentra Don Segundo personaje del cual obtendrá información importante para ayudar a su pueblo, en el camino se encontrara con aldeanos y soldados.
<b>Referencias Cruzadas:</b>	Tabla 18 – ESCENA: Encontrando a Don Segundo

Tabla 37 – Caso de Uso Encontrar Don Segundo

CASOS DE USO	
<b>Id:</b>	CU07
<b>Nombre de la Actividad:</b>	Obtener la cristal del tiempo
<b>Personajes:</b>	Simón – Oritah – Aldeanos – Don Segundo - Soldados
<b>Resumen:</b>	Para obtener el cristal del tiempo, Simón después de ser retado por los guardias, debe llevar un tapabocas a un número determinado de ciudadanos en un tiempo determinado, acompañado por Oritah y Don Segundo.
<b>Referencias Cruzadas:</b>	Tabla 19 – ESCENA: Obteniendo el Cristal

Tabla 38 – Caso de Uso Obtener Cristal



### C.1.8.Descripción de los casos de uso

Se describen a fondo cada uno de los atributos de las actividades desarrolladas en los casos de uso:

DESCRIPCION CASOS DE USO	
<b>Id:</b>	DCU01
<b>Nombre de la Actividad:</b>	Aguardar
<b>Precondiciones:</b>	Simón debe estar fuera de un perímetro definido a partir del personaje no jugador.
<b>Personajes:</b>	Don Segundo – Oritah – Soldado – Aldeano
<b>Resumen:</b>	Todos los personajes no jugador tienen este comportamiento mientras y es el que les permite estar alerta de cuando se encuentren con Simón.
<b>Tipo:</b>	Primario
<b>Curso normal de los eventos:</b>	1. Al iniciar el juego todos los personajes no jugador deben esperar a encontrarse con Simón para ejecutar sus comportamientos pro-activos.
<b>Excepciones:</b>	
<b>Post-condiciones:</b>	Permite que los personajes no jugador definan su comportamiento reactivo antes de activar los comportamientos reactivos.
<b>Referencias Cruzadas:</b>	Tabla 32 – Caso de Uso Aguardar

Tabla 39 – Descripción Caso de Uso Aguardar

DESCRIPCION CASOS DE USO	
<b>Id:</b>	DCU02
<b>Nombre de la Actividad:</b>	Acompañar a Simón
<b>Precondiciones:</b>	Simón debe estar dentro de un perímetro determinado a partir del personaje no jugador (Oritah – Don Segundo)
<b>Personajes:</b>	Oritah – Don Segundo
<b>Resumen:</b>	Cuando Simón se acerque a alguno de los personajes (Oritah – Don Segundo) estos mantienen una conversación con Simón y después de esa conversación estos los personajes Oritah – Don Segundo deben seguir a Simón durante todo su recorrido brindándole pistas para obtener información acerca de los medios de prevención de la tuberculosis.
<b>Tipo:</b>	Primario
<b>Curso normal de los eventos:</b>	1. Simón se acerca a Oritah. 2. Oritah brinda información referente a la actividad que debe realizar Simón y decide acompañarlo (Oritah debe seguir a Simón a



	donde se dirija).
<b>Excepciones:</b>	
<b>Post-condiciones:</b>	Simón se encuentra preparado para la prueba de los tapabocas.
<b>Referencias Cruzadas:</b>	Tabla 33 – Caso de Uso Acompañar a Simón

Tabla 40 – Descripción Caso de Uso Acompañar a Simón

DESCRIPCION CASOS DE USO	
<b>Id:</b>	DCU03
<b>Nombre de la Actividad:</b>	Patrullar
<b>Precondiciones:</b>	Simón debe estar dentro de un perímetro determinado a partir del personaje Soldado
<b>Personajes:</b>	Soldado
<b>Resumen:</b>	Al Simón sobre pasar el perímetro definido hacia el personaje Soldado, este deberá sostener una conversación con Simón y a partir de la cual decidirá si permite o no el paso.
<b>Tipo:</b>	Primario
<b>Curso normal de los eventos:</b>	<ol style="list-style-type: none"><li>1. Simón irrumpe en el perímetro del soldado.</li><li>2. El soldado avanza hacia Simón y le pide un documento.<ol style="list-style-type: none"><li>2.1. Si. El soldado permite el paso de Simón.</li><li>2.2. No. El soldado va impedir el paso de Simón hasta que consiga el papel, realizando un movimiento repetitivo en la única entrada.</li></ol></li></ol>
<b>Excepciones:</b>	
<b>Poscondiciones:</b>	El soldado permite el paso de Simón por ese camino en el pueblo.
<b>Referencias Cruzadas:</b>	Tabla 34 – Caso de Uso Patrullar

Tabla 41 – Descripción Caso de Uso Patrullar

DESCRIPCION CASOS DE USO	
<b>Id:</b>	DCU04
<b>Nombre de la Actividad:</b>	Repartir Tapabocas
<b>Precondiciones:</b>	Simón no debe estar cerca del agente en cuestión.
<b>Personajes:</b>	Don Segundo – Oritah – Soldado – Aldeano
<b>Resumen:</b>	Todos los personajes no jugador tienen este comportamiento mientras no se encuentren con Simón no ejecutan ninguno de sus otros comportamientos.
<b>Tipo:</b>	Primario
<b>Curso normal de los eventos:</b>	<ol style="list-style-type: none"><li>1. Orah le envía un mensaje “mágico” a Simón, donde le dice que se dirija hasta su casa.</li><li>2. Simón debe desplazarse por el terreno hasta encontrar su casa.</li></ol>



	3. Simón se encuentra con Oritah e inician un dialogo, donde se define la siguiente actividad a realizar por Simón.
<b>Excepciones:</b>	
<b>Poscondiciones:</b>	Simón estará en capacidad emprender la búsqueda de Don Segundo.
<b>Referencias Cruzadas:</b>	Tabla 35 – Caso de Uso Repartir Tapabocas Tabla 39 – Descripción Caso de Uso Aguardar

Tabla 42 – Descripción Caso de Uso Repartir Tapabocas

DESCRIPCION CASOS DE USO	
<b>Id:</b>	DCU05
<b>Nombre de la Actividad:</b>	Encontrar a Oritah
<b>Precondiciones:</b>	Simón debe haber leído el mensaje de Orah donde le decía que debía ir hasta su casa.
<b>Personajes:</b>	Simón – Oritah
<b>Resumen:</b>	Simón debe desplazarse por el terreno hasta encontrar su casa, de esta forma tendrá su primer encuentro con Oritah que es el personaje que lo guiara en su aventura, en su camino a casa puede tropezarse con Aldeanos o Soldados.
<b>Tipo:</b>	Primario
<b>Curso normal de los eventos:</b>	<ol style="list-style-type: none"> <li>1. Orah le envía un mensaje “mágico” a Simón, donde le dice que se dirija hasta su casa.</li> <li>2. Simón debe desplazarse por el terreno hasta encontrar su casa.</li> <li>3. Simón se encuentra con Oritah e inician un dialogo, donde se define la siguiente actividad a realizar por Simón.</li> </ol>
<b>Excepciones:</b>	
<b>Post-condiciones:</b>	Simón estará en capacidad emprender la búsqueda de Don Segundo.
<b>Referencias Cruzadas:</b>	Tabla 36 – Caso de Uso Encontrar Oritah Tabla 32 – Caso de Uso Aguardar

Tabla 43 – Descripción Caso de Uso Encontrar Oritah

DESCRIPCION CASOS DE USO	
<b>Id:</b>	DCU06
<b>Nombre de la Actividad:</b>	Encontrar a Don Segundo
<b>Precondiciones:</b>	Simón debe haberse encontrado antes con Oritah.
<b>Personajes:</b>	Simón – Oritah – Aldeanos – Soldados
<b>Resumen:</b>	Simón acompañado de Oritah debe dirigirse hasta donde se encuentra Don Segundo personaje del cual obtendrá información importante para ayudar a su



DESCRIPCION CASOS DE USO	
	pueblo, en el camino se encontrara con aldeanos y soldados.
<b>Tipo:</b>	Primario
<b>Curso normal de los eventos:</b>	<p>A. Simón se encuentran con un aldeano quien después de mantener un dialogo con Simón y Oritah les entrega un permiso firmado por el alcalde para forasteros.</p> <p>B. Al encontrarse con los Soldados estos le solicitan el permiso para seguir, de lo contrario no permiten que Simón llegue hasta Don Segundo.</p> <p>C. Llegan al lugar donde se encuentra Don Segundo Y mantienen un dialogo con él.</p>
<b>Excepciones:</b>	
<b>Post-condiciones:</b>	Don Segundo se une temporalmente a la odisea de Simón
<b>Referencias Cruzadas:</b>	Tabla 32 – Caso de Uso Aguardar Tabla 37 – Caso de Uso Encontrar Don Segundo

Tabla 44 – Descripción Caso de Uso Encontrar Don Segundo

DESCRIPCION CASOS DE USO	
<b>Id:</b>	DCU07
<b>Nombre de la Actividad:</b>	Obtener cristal
<b>Precondiciones:</b>	Simón debe encontrarse con Don Segundo
<b>Personajes:</b>	Simón – Oritah – Aldeanos – Soldados
<b>Resumen:</b>	Para obtener la gema Simón después de ser retado por los guardias Simón debe llevar un tapabocas a un número determinado de ciudadanos en un tiempo determinado, acompañado por Oritah y Don Segundo.
<b>Tipo:</b>	Primario
<b>Curso normal de los eventos:</b>	<ol style="list-style-type: none"> <li>1. Don Segundo y Simón son sorprendidos por un soldado hablando acerca del daño que la enfermedad está haciendo al pueblo.</li> <li>2. Simón desafía al soldado en busca de recuperar su cristal.</li> <li>3. Conseguir los tapabocas que Don Segundo usa para fumigar y entregarle uno a cada aldeano en un tiempo determinado.</li> <li>4. Pantallazo de victoria</li> </ol>
<b>Excepciones:</b>	Si el tiempo se acaba y Simón no ha entregado todos los tapabocas sale el pantallazo de fin de juego.
<b>Post-condiciones:</b>	
<b>Referencias Cruzadas:</b>	Tabla 19 – ESCENA: Obteniendo el Cristal Tabla 32 – Caso de Uso Aguardar

	Tabla 33 – Caso de Uso Acompañar a Simón Tabla 34 – Caso de Uso Patrullar
--	--

Tabla 45 – Descripción Caso de Uso Obtener Cristal

**C.1.8.1. Representación gráfica de los casos de uso**

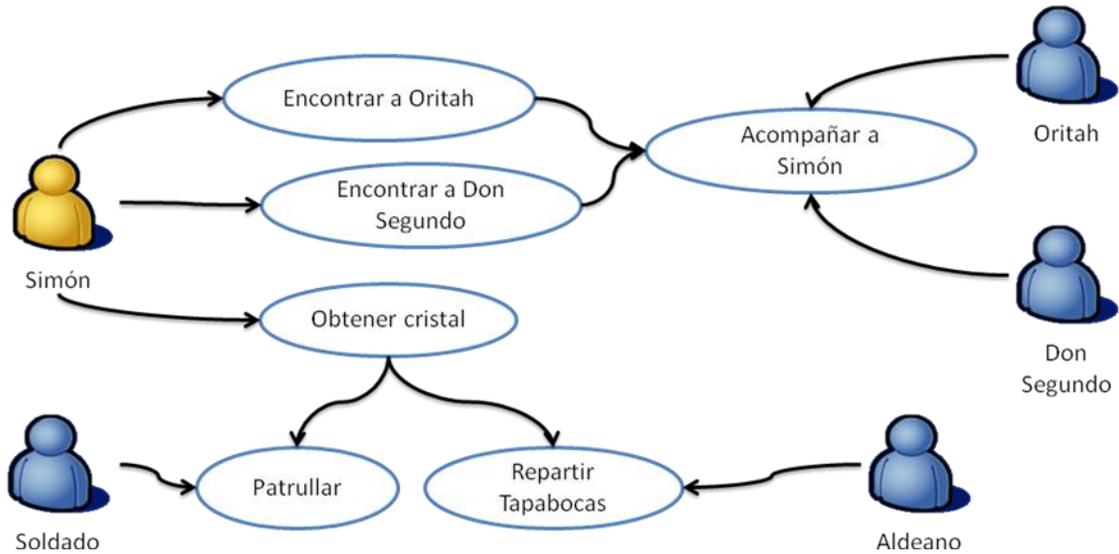


Figura 9 – Diagrama Casos de Usos 1

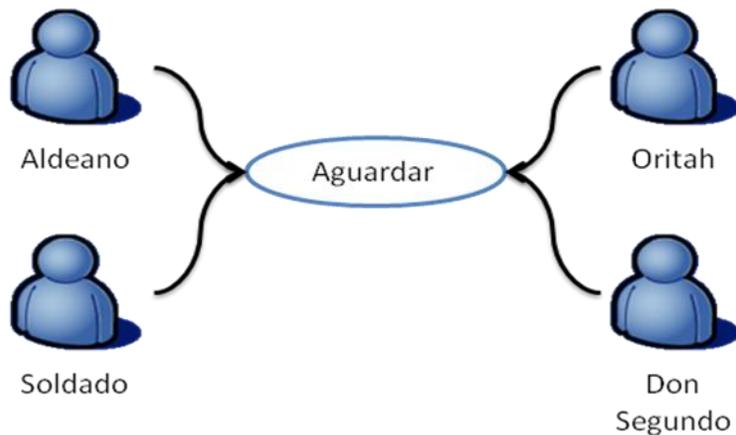


Figura 10 – Diagrama Casos de Uso 2



## C.2. Modelo de Comportamiento

COMPORTAMIENTO	
<b>Id:</b>	CMP01
<b>Comportamiento:</b>	Aguardar
<b>Caso de uso:</b>	CASO DE USO AGUARDAR
<b>Tipo:</b>	Proactivo
<b>Referencias Cruzadas:</b>	Tabla 39 – Descripción Caso de Uso Aguardar

Tabla 46 – Comportamiento Aguardar

COMPORTAMIENTO	
<b>Id:</b>	CMP03
<b>Comportamiento:</b>	Acompañar
<b>Caso de uso:</b>	Caso de Uso Acompañar a Simón
<b>Tipo:</b>	Proactivo
<b>Referencias Cruzadas:</b>	Tabla 40 – Descripción Caso de Uso Acompañar a Simón

Tabla 47 – Comportamiento Acompañar

COMPORTAMIENTO	
<b>Id:</b>	CMP02
<b>Comportamiento:</b>	Patrullar
<b>Caso de uso:</b>	Caso de Uso Patrullar
<b>Tipo:</b>	Reactivo
<b>Acción que lo dispara:</b>	Conversación con Simón
<b>Referencias Cruzadas:</b>	Tabla 41 – Descripción Caso de Uso Patrullar

Tabla 48 – Comportamiento Patrullar

COMPORTAMIENTO	
<b>Id:</b>	CMP04
<b>Comportamiento:</b>	Recibir
<b>Caso de uso:</b>	Caso De Uso Repartir Tapabocas
<b>Tipo:</b>	Proactivo
<b>Referencias Cruzadas:</b>	Tabla 42 – Descripción Caso de Uso Repartir Tapabocas

Tabla 49 – Comportamiento Recibir

### C.2.1. Notación gráfica

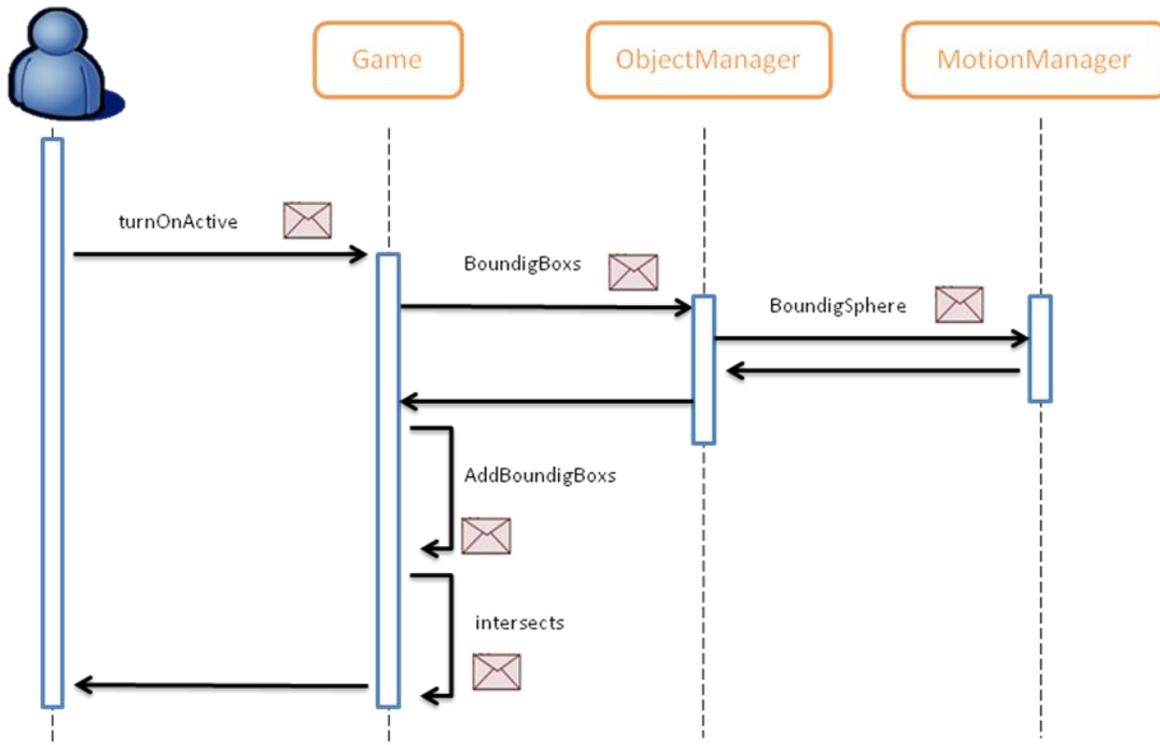


Figura 11 – Diagrama del comportamiento aguardar

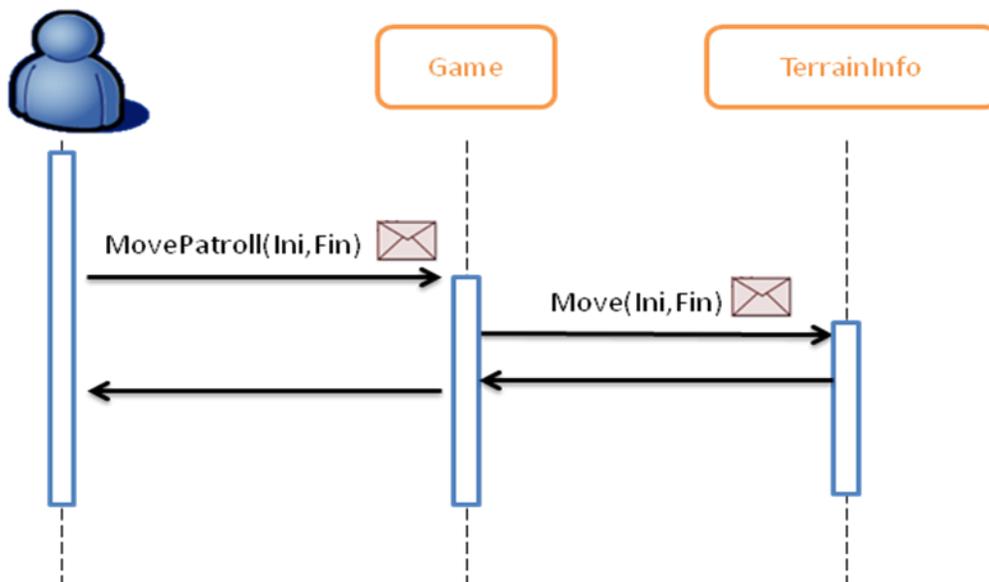


Figura 12 – Diagrama del comportamiento Patrullar

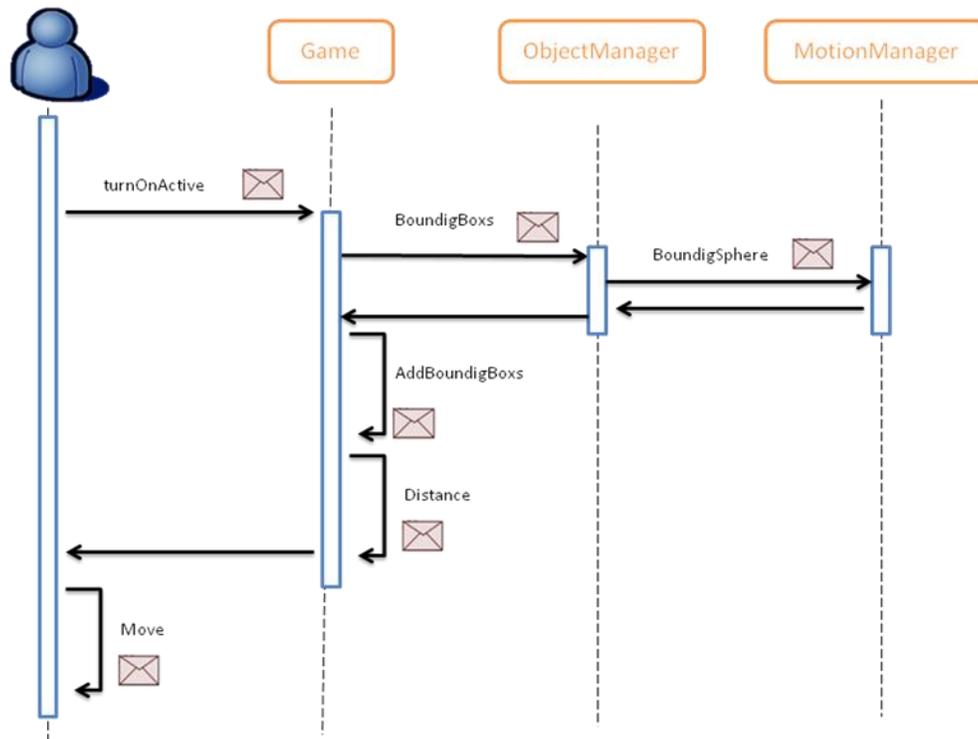


Figura 13 – Diagrama del comportamiento Acompañar

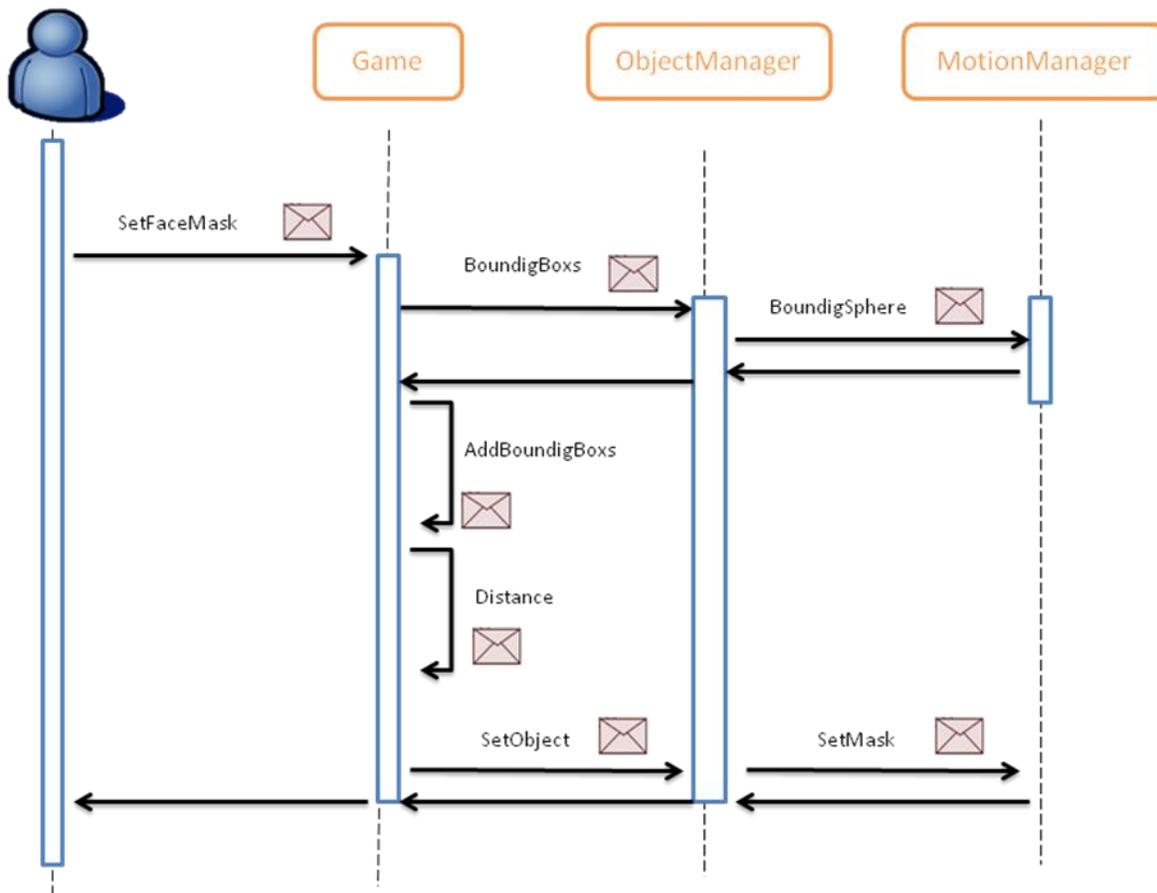


Figura 14 – Diagrama del comportamiento Recibir



### C.3. Modelo de Coordinación

MENSAJE	
ID Mensaje	MSJ01
Per-formativa	Accept_Proposal
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	Simón!, lo lograste!. Has viajado 25 años en el tiempo hasta la época en que todo comenzó. Necesitas hacer algo antes de continuar. Ve a la casa donde vivirás en un futuro, allí te encontrarás con la amiga que te comenté hace un momento, ella te guiará en tu misión.
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 50 – MSJ01

MENSAJE	
ID Mensaje	MSJ02
Per-formativa	Inform
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	Mucha suerte pequeño Simón, recuerda que el futuro de tu gente está en tus manos...
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 51 – MSJ02



MENSAJE	
ID Mensaje	MSJ03
Per-formativa	Inform>If
Agente Emisor	Simón
Agente Receptor	Orah
Agente a quien se enviaría Respuesta	Simón
Contenido: texto a enviar	Pero... Orah!? Te volveré a ver?
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 52 – MSJ03

MENSAJE	
ID Mensaje	MSJ04
Per-formativa	Inform
Agente Emisor	Simón
Agente Receptor	Orah
Agente a quien se enviaría Respuesta	Simón
Contenido: texto a enviar	Orah?... mejor me doy prisa.
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 53 – MSJ04

MENSAJE	
ID Mensaje	MSJ05
Per-formativa	Inform
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	Más pronto de lo que te imaginas... adiós.
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 54 – MSJ05



MENSAJE	
ID Mensaje	MSJ06
Per-formativa	Accept_Proposal
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	Hola holita hola!!!... Oh por todos los dioses has llegado!! Tú debes ser... Simón?
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 55 – MSJ06

MENSAJE	
ID Mensaje	MSJ08
Per-formativa	Inform
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	Ji ji ji, si si si !!! Me presento: Mi nombre es Orah y soy nueva por aquí... no te cuento más de mi porque estoy en una misión secreta ya sabes... de todas maneras creo que debes saber más de mí de lo que yo misma sé
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 56 – MSJ08

MENSAJE	
ID Mensaje	MSJ07
Per-formativa	Inform
Agente Emisor	Simón
Agente Receptor	Orah
Agente a quien se enviaría Respuesta	Simón
Contenido: texto a enviar	Oh! No no de nada, es sólo que



MENSAJE	
	alguien me dijo que tú eras una buena amiga...
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 57 – MSJ07

MENSAJE	
ID Mensaje	MSJ09
Per-formativa	Inform
Agente Emisor	Simón
Agente Receptor	Orah
Agente a quien se enviaría Respuesta	Simón
Contenido: texto a enviar	Ah... “Vieja amiga”... ahora veo.. en realidad no “tan” vieja!
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 58 – MSJ09

MENSAJE	
ID Mensaje	MSJ10
Per-formativa	If_Inform
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	De que hablas Simón?
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 59 – MSJ10

MENSAJE	
ID Mensaje	MSJ11
Per-formativa	Inform
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría	Orah



MENSAJE	
<b>Respuesta</b>	
<b>Contenido: texto a enviar</b>	Aunque... no es por nada, pero yo soy más linda y joven! Ji ji ji... aquella estaba viejita y despeinada! Espero nunca ser así cuando grande!!!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 60 – MSJ11

MENSAJE	
<b>ID Mensaje</b>	MSJ12
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Simón
<b>Agente Receptor</b>	Orah
<b>Agente a quien se enviaría Respuesta</b>	Simón
<b>Contenido: texto a enviar</b>	Eh?! ... mmmm.. no sabes nada... Oye Oritah.. te puedo decir Oritah?
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 61 – MSJ12

MENSAJE	
<b>ID Mensaje</b>	MSJ13
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Orah
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Orah
<b>Contenido: texto a enviar</b>	Si, de ahora en adelante dime Oritah... no me molesta en lo absoluto... que me querías decir?
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 62 – MSJ13



MENSAJE	
<b>ID Mensaje</b>	MSJ14
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Simón
<b>Agente Receptor</b>	Orah
<b>Agente a quien se enviaría Respuesta</b>	Simón
<b>Contenido: texto a enviar</b>	Mira, necesito tu ayuda, he viajado hasta aquí para intentar detener algo muy malo que va a suceder.
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 63 – MSJ14

MENSAJE	
<b>ID Mensaje</b>	MSJ15
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Orah
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Orah
<b>Contenido: texto a enviar</b>	Aaaa.. SI! Algo me dijo aquella perra señora, no te preocupes creo que me dió todas las indicaciones necesarias así cuenta conmigo!!!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 64 – MSJ15



MENSAJE	
<b>ID Mensaje</b>	MSJ16
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Orah
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Orah
<b>Contenido: texto a enviar</b>	Para empezar, creo que debemos ir a visitar a Don Segundo ( <i>DS</i> )... escuché cuando le decía a un vecino que algo extraño le está sucediendo y que al parecer los guardias de aquel brujo curandero no han podido hacer nada para detener la peste que cada día es más poderosa!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 65 – MSJ16

MENSAJE	
<b>ID Mensaje</b>	MSJ17
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Orah
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Orah
<b>Contenido: texto a enviar</b>	Don Segundo vive aquí arriba por el camino de la izquierda, sígueme.
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 66 – MSJ17



MENSAJE	
ID Mensaje	MSJ18
Per-formativa	Query_If
Agente Emisor	Soldado
Agente Receptor	Orah
Agente a quien se enviaría Respuesta	Soldado
Contenido: texto a enviar	Debes tener una autorización expresa del alcalde para poder seguir... Tengo ordenes niño!
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 67 – MSJ18

MENSAJE	
ID Mensaje	MSJ19
Per-formativa	Call_for_Proposal
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	Psss Simón... mira en tu inventario y haz click sobre el permiso, me da la impresión que ya lo tienes... Si no, creo que tendremos que ir a buscar uno...
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 68 – MSJ19



MENSAJE	
ID Mensaje	MSJ20
Per-formativa	Agree
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	Aquí tiene señor
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 69 – MSJ20

MENSAJE	
ID Mensaje	MSJ21
Per-formativa	Query_If
Agente Emisor	Soldado
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Soldado
Contenido: texto a enviar	eh!?!? Pero com...?!... eeh je je.. sigue sigue, disfruta tu visita!
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 70 – MSJ21

MENSAJE	
ID Mensaje	MSJ22
Per-formativa	Confirm
Agente Emisor	Orah
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Orah
Contenido: texto a enviar	Vamos Simón apurémonos tenemos que hablar con el antes de la reunión semanal del consejo y ya casi es la hora!
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 71 – MSJ22



MENSAJE	
ID Mensaje	MSJ24
Per-formativa	Inform_if
Agente Emisor	Don Segundo
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Don Segundo
Contenido: texto a enviar	¿?... Oye niño no te conozco... que quieres hablar conmigo?
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 72 – MSJ24

MENSAJE	
ID Mensaje	MSJ25
Per-formativa	Inform_if
Agente Emisor	Don Segundo
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Don Segundo
Contenido: texto a enviar	QUE?!?? Que es lo que dices!? Que la pest...
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 73 – MSJ25



MENSAJE	
<b>ID Mensaje</b>	MSJ26
<b>Per-formativa</b>	Call_for_Proposal
<b>Agente Emisor</b>	Simón
<b>Agente Receptor</b>	Don Segundo
<b>Agente a quien se enviaría Respuesta</b>	Simón
<b>Contenido: texto a enviar</b>	Shisstt Don Segundo por favor! Sea precavido que no queremos que nadie que no debe se entere! Recuerde que el guardia está muy cerca y podría escucharnos.
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 74 – MSJ26

MENSAJE	
<b>ID Mensaje</b>	MSJ27
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Don Segundo
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Don Segundo
<b>Contenido: texto a enviar</b>	Por cierto... Espera! Hace unos días descubrí una nota rota en pedazos, sin embargo después de lo que me dices recordé ver algo que tenía sentido, Si me ayudas a armarla puede ser una pista!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 75 – MSJ27



MENSAJE	
ID Mensaje	MSJ28
Per-formativa	Inform
Agente Emisor	Don Segundo
Agente Receptor	Simón
Agente a quien se enviaría Respuesta	Don Segundo
Contenido: texto a enviar	Excelente! Con la nueva información que tenemos podremos ayudar a los demás...!
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 76 – MSJ28

MENSAJE	
ID Mensaje	MSJ29
Per-formativa	Accept_Proposal
Agente Emisor	Soldado
Agente Receptor	Don Segundo
Agente a quien se enviaría Respuesta	Soldado
Contenido: texto a enviar	Ajá!!! Los descubrí conspirando contra nuestro amo! Eso debe ser castigado!
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 77 – MSJ29

CONVERSACION	
ID Mensaje	MSJ30
Per-formativa	Inform
Agente Emisor	Don Segundo
Agente Receptor	Soldado
Agente a quien se enviaría Respuesta	Don Segundo
Contenido: texto a enviar	Pe...pe..pero señor guardia no!...
Lenguaje	Español
Codificación	lenguaje natural
Ontología	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 78 – MSJ30



<b>MENSAJE</b>	
<b>ID Mensaje</b>	MSJ31
<b>Per-formativa</b>	Call_for_Proposal
<b>Agente Emisor</b>	Soldado
<b>Agente Receptor</b>	Don Segundo
<b>Agente a quien se enviaría Respuesta</b>	Soldado
<b>Contenido: texto a enviar</b>	Silencio!, no permitiré que dejen al descubierto el plan de mi señor Kash!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 79 – MSJ31

<b>MENSAJE</b>	
<b>ID Mensaje</b>	MSJ32
<b>Per-formativa</b>	Quey_If
<b>Agente Emisor</b>	Don Segundo
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Don Segundo
<b>Contenido: texto a enviar</b>	Oh.. nos descubrieron y ahora que haremos!?
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 80 – MSJ32



<b>MENSAJE</b>	
<b>ID Mensaje</b>	MSJ33
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Orah
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Orah
<b>Contenido: texto a enviar</b>	Simón, esos guardias me robaron los cristales del tiempo que me regalo mi padre! Si los desafías a una prueba quizás es la única manera en que las puedas recuperar y con la ayuda de ellas salvarnos del malvado brujo!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 81 – MSJ33

<b>MENSAJE</b>	
<b>ID Mensaje</b>	MSJ34
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Simón
<b>Agente Receptor</b>	Soldado
<b>Agente a quien se enviaría Respuesta</b>	Simón
<b>Contenido: texto a enviar</b>	No dejaré que le hagas más daño a mi gente guardia apestoso!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 82 – MSJ34



MENSAJE	
<b>ID Mensaje</b>	MSJ35
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Soldado
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Soldado
<b>Contenido: texto a enviar</b>	Ya es tarde enano!, he ordenado a los demás guardias rondar cerca de los debiluchos que quedan en este pueblucho! Es solo cuestión de minutos para que todos se empiecen a enfermar!!!! JUA, JUA, JUA
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 83 – MSJ35

MENSAJE	
<b>ID Mensaje</b>	MSJ36
<b>Per-formativa</b>	Call_for_Proposal
<b>Agente Emisor</b>	Simón
<b>Agente Receptor</b>	Soldado
<b>Agente a quien se enviaría Respuesta</b>	Don Segundo
<b>Contenido: texto a enviar</b>	Don Segundo hay que actuar!, necesito algo para evitar que esos guardias contagien a los ciudadanos que se encuentran en el pueblo... los tapabocas!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 84 – MSJ36



MENSAJE	
<b>ID Mensaje</b>	MSJ37
<b>Per-formativa</b>	Inform
<b>Agente Emisor</b>	Don Segundo
<b>Agente Receptor</b>	Simón
<b>Agente a quien se enviaría Respuesta</b>	Don Segundo
<b>Contenido: texto a enviar</b>	El tapabocas que uso cuando fumigo las plantas! Coincidentalmente lo tengo en el bolsillo, tómallo!, llévaselo a alguien y mientras vuelves buscaré más en mi casa... DATE PRISA!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 85 – MSJ37

MENSAJE	
<b>ID Mensaje</b>	MSJ38
<b>Per-formativa</b>	Call_for_Proposal
<b>Agente Emisor</b>	Orah
<b>Agente Receptor</b>	Simon
<b>Agente a quien se enviaría Respuesta</b>	Orah
<b>Contenido: texto a enviar</b>	Simón!, tienes que encontrar a los 5 ciudadanos fuera de Don Segundo que hay ahora en el pueblo antes de que sea tarde!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 86 – MSJ38



MENSAJE	
<b>ID Mensaje</b>	MSJ39
<b>Per-formativa</b>	Agree
<b>Agente Emisor</b>	Simón
<b>Agente Receptor</b>	Orah
<b>Agente a quien se enviaría Respuesta</b>	Simón
<b>Contenido: texto a enviar</b>	Excelente! Con la nueva información que tenemos podremos ayudar a los demás...!
<b>Lenguaje</b>	Español
<b>Codificación</b>	lenguaje natural
<b>Ontología</b>	No aplica. Porque se encuentra codificada en lenguaje natural.

Tabla 87 – MSJ39

CONVERSACION	
<b>ID Mensaje</b>	CNV01
<b>Descripción</b>	Esta conversación se lleva a cabo durante la primera escena, se realiza una breve introducción.
<b>Personaje/Agente</b>	<b>Personaje/Agente</b>
Orah	Simón
<b>Mensaje</b>	
Tabla 50 – MSJ01 Tabla 51 – MSJ02 Tabla 53 – MSJ04	Tabla 52 – MSJ03 Tabla 54 – MSJ05

Tabla 88 – Conversación Saludo

CONVERSACION	
<b>ID Mensaje</b>	CNV02
<b>Descripción</b>	En esta conversación Orah le comenta a Simón cual es su primer objetivo.
<b>Personaje/Agente</b>	<b>Personaje/Agente</b>
Orah	Simón
<b>Mensaje</b>	
Tabla 55 – MSJ06 Tabla 56 – MSJ08 Tabla 59 – MSJ10 Tabla 61 – MSJ12 Tabla 62 – MSJ13 Tabla 64 – MSJ15	Tabla 57 – MSJ07 Tabla 58 – MSJ09 Tabla 60 – MSJ11 Tabla 63 – MSJ14 Tabla 65 – MSJ16



CONVERSACION	
Tabla 66 – MSJ17 Tabla 67 – MSJ18	

Tabla 89 – Conversación guía Orah

CONVERSACION	
<b>ID Mensaje</b>	CNV03
<b>Descripción</b>	En esta conversación Orah brinda información a Simón para repartir los tapabocas.
<b>Personaje/Agente</b>	<b>Personaje/Agente</b>
Simon	Oritah
<b>Mensaje</b>	
Tabla 69 – MSJ20 Tabla 87 – MSJ39	Tabla 71 – MSJ22 Tabla 81 – MSJ33 Tabla 86 – MSJ38

Tabla 90 – Conversación Soldado

CONVERSACION	
<b>ID Mensaje</b>	CNV04
<b>Descripción</b>	En esta conversación soldado descubre la conspiración.
<b>Personaje/Agente</b>	<b>Personaje/Agente</b>
Soldado	Oritah
<b>Mensaje</b>	
Tabla 77 – MSJ29 Tabla 79 – MSJ31	Tabla 78 – MSJ30

Tabla 91 – Conversación Conspiración

CONVERSACION	
<b>ID Mensaje</b>	CNV05
<b>Descripción</b>	En esta conversación Don Segundo ayuda a Simón en su tarea de repartir tapabocas a los campesinos.
<b>Personaje/Agente</b>	<b>Personaje/Agente</b>
Simón	Don Segundo
<b>Mensaje</b>	



CONVERSACION	
Tabla 74 – MSJ26	Tabla 72 – MSJ24 Tabla 73 – MSJ25 Tabla 75 – MSJ27 Tabla 76 – MSJ28 Tabla 80 – MSJ32 Tabla 85 – MSJ37

Tabla 92 – Conversación repartir tapabocas

CONVERSACION	
<b>ID Mensaje</b>	CNV06
<b>Descripción</b>	En esta conversación el soldado irrumpe en el camino de Simón
<b>Personaje/Agente</b>	<b>Personaje/Agente</b>
Simón	Soldado
<b>Mensaje</b>	
Tabla 67 – MSJ18 Tabla 70 – MSJ21 Tabla 83 – MSJ35	Tabla 68 – MSJ19 Tabla 82 – MSJ34 Tabla 84 – MSJ36

Tabla 93 – Conversación Soldado

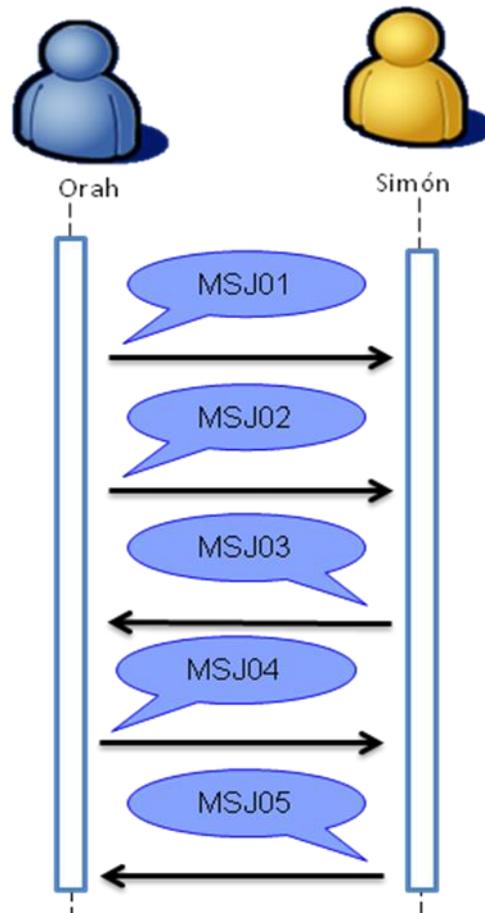


Figura 15 – Diagrama Conversación Saludo

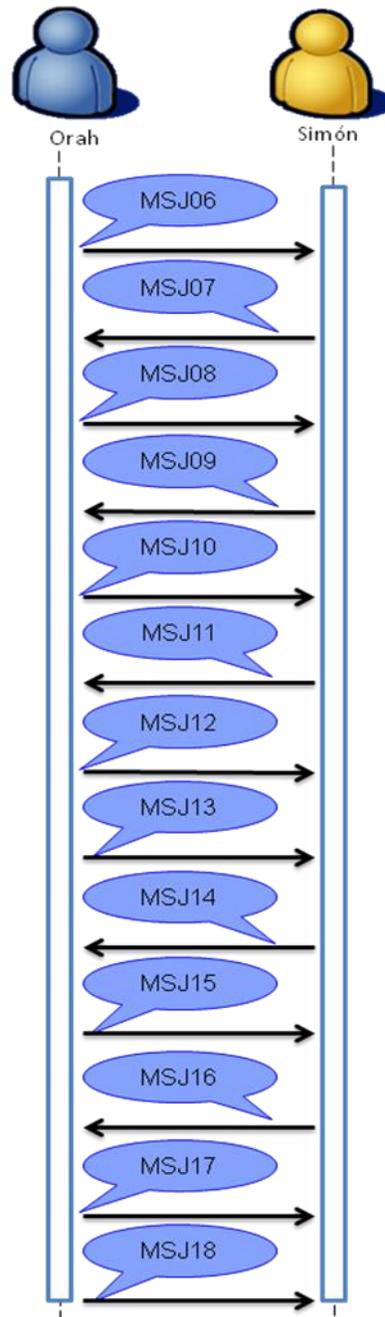


Figura 16 – Diagrama Conversación guía

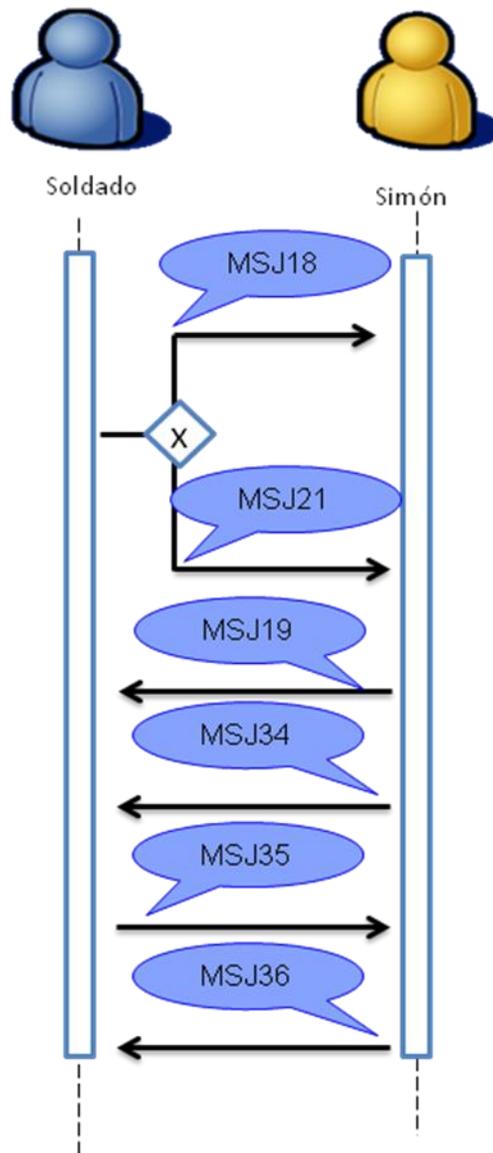


Figura 17 – Conversación Diagrama Conversación Soldado

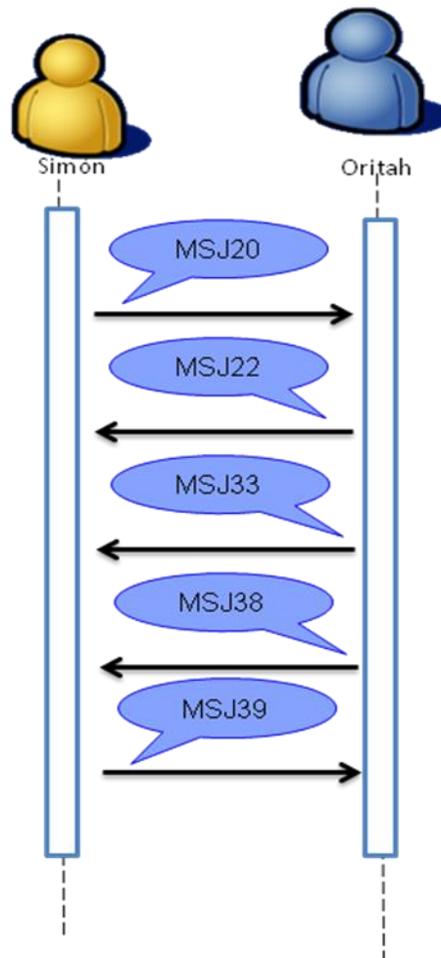


Figura 18 – Conversación Repartir tapabocas

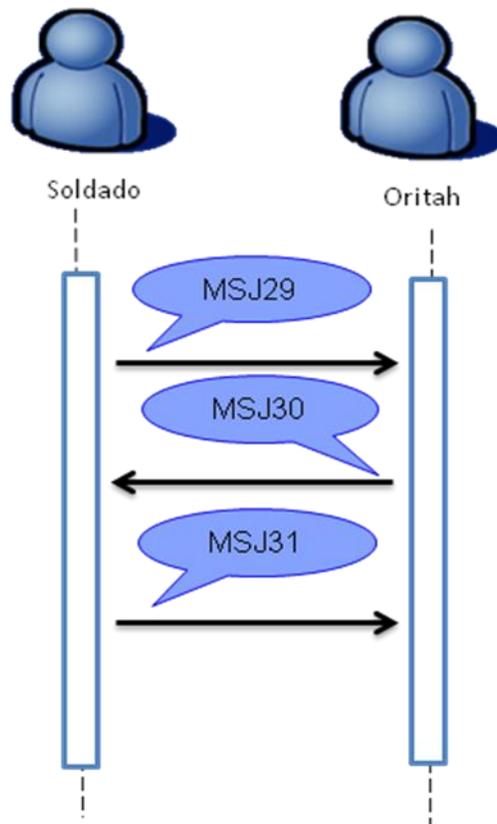


Figura 19 – Conversación Conspiración

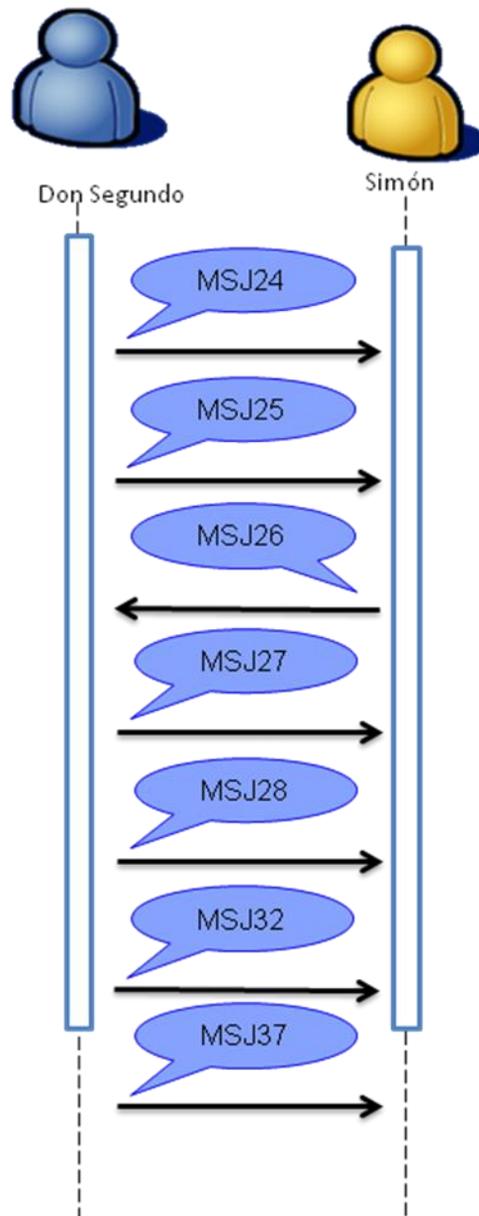


Figura 20 - Conversación Don Segundo

### C.4. Modelo de Organización

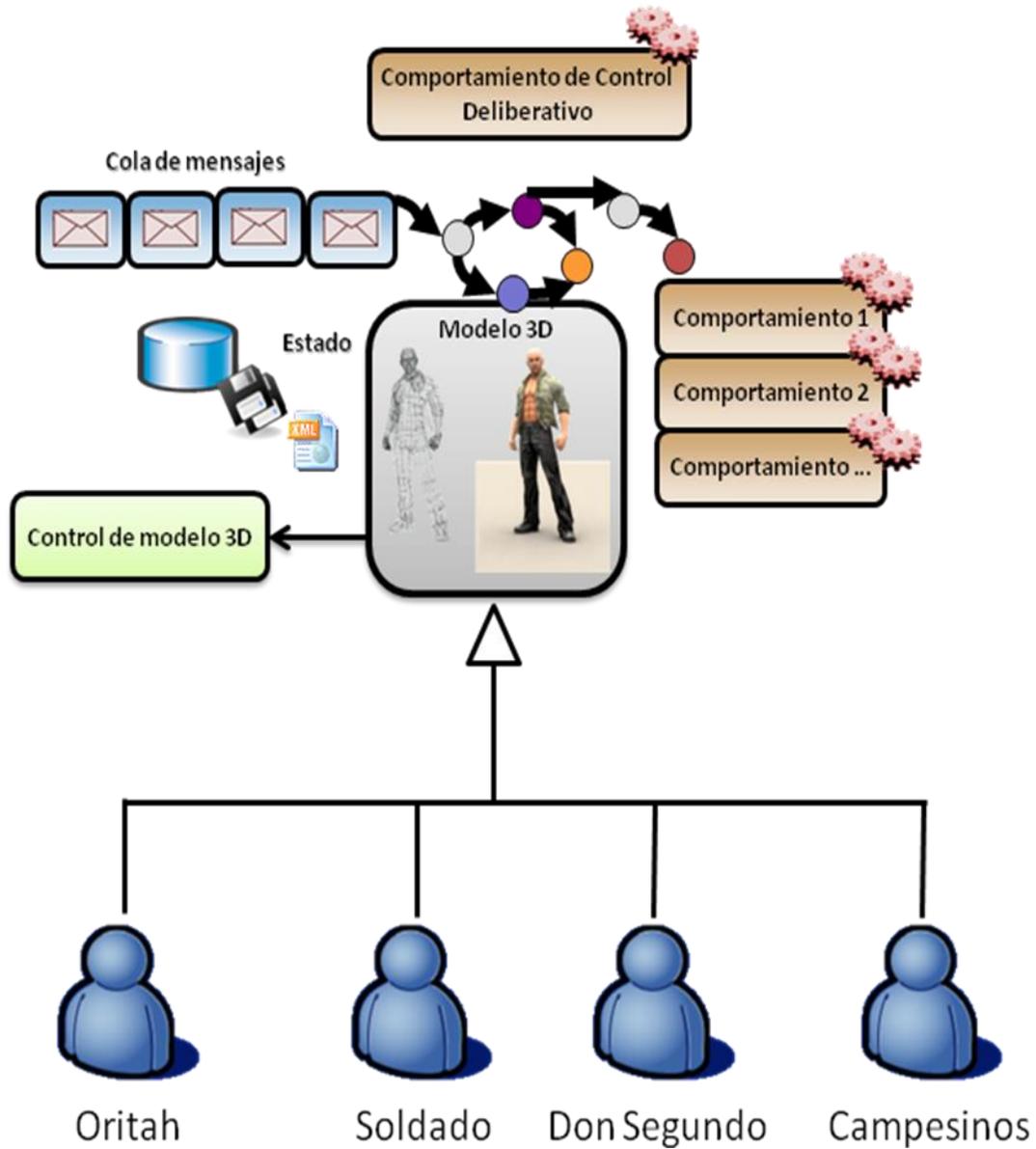


Figura 21 – Modelo de Organización “Simón y La Tuberculosis”



## C.5. Modelo de Diseño

En el presente ítem se presenta el modelo de diseño el cual tiene como objetivo principal la descripción de los componentes de los distintos modelos de la etapa de análisis de forma tal que se facilite su definición en un lenguaje de programación.

### C.5.1. Diseño De Agentes

SISTEMA PERSONAJE/AGENTE	
<b>Id:</b>	AG001
<b>Nombre del Personaje/Agente:</b>	Orah
<b>Arquitectura</b>	Deliberativo
<b>Lenguaje</b>	C# - XNA
<b>Sub-Sistema</b>	No tiene asociado un Sub-Sistema
<b>Referencias Cruzadas</b>	

Tabla 94 – Arquitectura Orah

SISTEMA PERSONAJE/AGENTE	
<b>Id:</b>	AG002
<b>Nombre del Personaje/Agente:</b>	Don Segundo
<b>Arquitectura</b>	Deliberativo
<b>Lenguaje</b>	C# - XNA
<b>Sub-Sistema</b>	No tiene asociado un Sub-Sistema
<b>Referencias Cruzadas</b>	

Tabla 95 – Arquitectura Don Segundo

SISTEMA PERSONAJE/AGENTE	
<b>Id:</b>	AG003
<b>Nombre del Personaje/Agente:</b>	Soldado
<b>Arquitectura</b>	Deliberativo
<b>Lenguaje</b>	C# - XNA
<b>Sub-Sistema</b>	No tiene asociado un Sub-Sistema
<b>Referencias Cruzadas</b>	

Tabla 96 – Arquitectura Soldado



SISTEMA PERSONAJE/AGENTE	
<b>Id:</b>	AG004
<b>Nombre del Personaje/Agente:</b>	Campesino
<b>Arquitectura</b>	Deliberativo
<b>Lenguaje</b>	C# - XNA
<b>Sub-Sistema</b>	No tiene asociado un Sub-Sistema
<b>Referencias Cruzadas</b>	

Tabla 97 – Arquitectura Campesino

### C.5.2. Diseño De Plataforma

PLATAFORMA	
<b>Id:</b>	PL001
<b>Nombre de la Plataforma:</b>	Jatt.Agentes
<b>Descripción:</b>	Para la implementación de los agentes se desarrollo una API , la cual se explica en el <b>¡Error! No se encuentra el origen de la referencia.</b>
<b>Lenguaje:</b>	C#
<b>Hardware:</b>	<ul style="list-style-type: none"><li>• Procesador Pentium a 600MHz, recomendado 1GHz (un gigahertz)</li><li>• Sistema Operativo Windows 2000 o superior.</li><li>• 192Mb en Ram, recomendado 256Mb</li><li>• 1.3Gb de disco duro disponible</li><li>• Monitor de 800x600 a 256 colores, recomendado 1024x768 a 16 bits.</li></ul>
<b>Software:</b>	Visual Studio C# 2005 Express Editon
<b>Usuario:</b>	Orah – Don Segundo – Soldado - Campesino
<b>Referencias Cruzadas:</b>	

Tabla 98 – Arquitectura de la plataforma



## BIBLIOGRAFIA

- 
- [1] **Ausubel, D. P. y Sullivan E. V.** *El desarrollo infantil. 3. Aspectos lingüísticos, cognitivos y físicos.* Barcelona. Paidós.1983.
  - [2] **Bijou, Sydney W.** (1982): *Psicología del desarrollo infantil.* México, D. F. Trillas.
  - [3] **Calvo, A.** *Ocio en los noventa: los videojuegos.* Tesis Doctoral Palma de mallorca: Universitat de les Illes Balears. 1997.
  - [4] **Licona, Ana Liliam.** *Los videojuegos como actividad lúdica del siglo XXI: el caso de Latinoamérica.*
  - [5] **Meix, joan Isern.** *La mala influencia de los juegos violentos.* [http://www.meristation.com/v3/des\\_noticia.php?id=8531&pic](http://www.meristation.com/v3/des_noticia.php?id=8531&pic) EEUU. 2002. visitada el 14 de marzo de 2007.
  - [6] *Universidad europea de Madrid. Observatorio del videojuego* <http://www.uem.es/web/cin/cin2/observatorio/> visitada el 14 de marzo de 2007.
  - [7] **Mancini, Pablo.** *Videojuegos, sociabilidad y aprendizaje.* [http://videojuegos.educ.ar/vi/usuarios-pedagogicos/videojuegos\\_sociabilidad\\_y\\_apr.php](http://videojuegos.educ.ar/vi/usuarios-pedagogicos/videojuegos_sociabilidad_y_apr.php) visitada el 14 de marzo de 2007.
  - [8] **Canovas, Guillermo.** *Estudio realizado por protégeles. Videojuegos, menores y responsabilidad de los padres.* Diciembre del 2005.
  - [9] **Rodríguez, Elena.** *Jóvenes y Videojuegos: Espacio, significación y conflictos (FAD), Los tipos agrupados de videojuegos.* España. 2002. <http://www.injuve.mtas.es/> .
  - [10] *Intentando crear videojuegos. Metodología SCUMM2 (personajes).* España 2005. <http://admin.blog.mundo-omepet.com/index.php?op=Default&postCategoryId=2&blogId=1> visitada el 16 de marzo de 2007.
  - [11] **González González Alfredo; Baqueiro Espinosa Omar, Meza Cota Emmanuel** *Metodologías de la I.A. (Agentes autónomos y Redes neuronales Supervisadas) aplicadas a NPCs (Non player characters),* Universidad Autónoma de Baja California Sur. Departamento de Sistemas Computacionales. 2004. <http://www.redcientifica.com/doc/doc200401210112.html> visitada el 5 de marzo del 2007.
  - [12] **Chandler, Rafael.** *It Builds Character: Character Development Techniques in Games.* [http://www.gamasutra.com/features/20050810/chandler\\_01.shtml](http://www.gamasutra.com/features/20050810/chandler_01.shtml) 2005 visitada el 14 de marzo de 2007.
  - [13] **Ferber, J. y Gutknecht, O.** *A Meta-Model for the Analysis and Design of Organizations in Multi-Agent Systems.* Actas de conferencia. Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS98), IEEE CS Press. 1998.
  - [14] **G. Martin, Ignacio.** *La Industria del Videojuego: Un Negocio en Alza.* Madrid: CEPREDE, 2006, Vol. XI.
  - [15] **Crawford, Chris.** *The Art of Computer Games.* Washington: Washington State University Vancouver, 2000.



- [16] **Tanguay, David.** *A guide to create the ideal adventure game.* California: Adventure Classic Gaming, 2006.
- [17] **Egri, Lajos.** *Art of Creative Writing.* Ohio: UArt, 1998.
- [18] **Zerbst, Stefan y Düvel, Oliver.** *3D Game Engine Programming.* Boston : Series Editor, 2004.
- [19] **Feil, John y Scattergood, Marc.** *Beginning Game Level Desing.* Boston : Thomson Course Technology PTR,, 2005.
- [20] . **Carrasco Polaino, Rafael.** *Propuesta De Tipología Básica De Los Videojuegos De PC Y Consola 7,* Villanueva : ICONO 14, 2006, Vol. I.
- [21] **Sheldon, Lee.** *Character Development And Storytelling For Games.* Bostom : Thomson Course Technology PTR, 2005.
- [22] **Handler Miller, Carolyn.** *Digital Storytelling.* Oxford : ELSEVIER, 2004.
- [23] **Ferber, J. Gutknecht, O.** *A Meta-Model for the Analysis and Design of Organizations in Multi-Agent Systems. Actas de conferencia. Proceedings of the Third International Conference on Multi-Agent Systems.* Boston : IEEE CS Press, 1998.
- [24] **Russell, Stuart.** *Inteligencia Artificial: un enfoque moderno.* México : Prentice - Hall, 1996.
- [25] **Franklin, Stan y Graesser, Art.** *Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages.* United States : Springer-Verlag, 1996.
- [26] . **Jennings, Nicholas. Wooldridge, Michael.** *Intelligent Agents: Theory and Practice* Manchester : Knowledge Engineering Review, 1994.
- [27] **DeLoach, Scott.** *Analysis and Design using MaSE and AgentTool.* Ohio : MAICS, 2001. Vol. XII.
- [28] **Samara Ruiz, Maria Eguimendia.** *Metodología para sistemas multiagente, MaSE.* Vol. 1.
- [29] . **Wooldridge, Michael. Jennings, Nicholas y Kinny, David.** *The Gaia Methodology for Agent-Oriented Analysis and Design. 27,* Boston: Kluwer Academic Publishers, 2000, Vol. I.
- [30] **Gómez-Sanz, J.** *Modelado de Sistemas Multi-Agente.* PhD thesis, Departamento de Sistemas Informáticos y Programación, Universidad Complutense Madrid, 2002.
- [31] **GRASIA.** Grupo de Agentes de Software: Ingeniería y aplicaciones. *Pagina Principal INGENIAS.* [En línea] [Citado el: 14 de Junio de 2007.] [http://grasia.fdi.ucm.es/ingenias/Spain/ejemplos/ejemplo\\_actividades/analisis-inicio.php](http://grasia.fdi.ucm.es/ingenias/Spain/ejemplos/ejemplo_actividades/analisis-inicio.php).
- [32] **Caire, G., Leal, F., Chainho, P., Evans, R., Garijo, F., Gomez-Sanz, J. J., Pavon, J., Kerney, P., Stark, J.,and Massonet, P.** *Eurescom P907: MESSAGE - Methodology for Engineering Systems of Software Agents.* [En línea] [Citado el: 14 de Junio de 2007.] <http://www.eurescom.de/public/projects/P900-series/p907/default.asp> . 2002.
- [33] **Iglesias Fernández, Carlos Ángel.** *Definicion de una Metodologia Para el Desarrollo de Sistemas Multiagente.* Madrid-España : Universidad Politécnica de Madrid, 1998.
- [34] **Collis, Jaron y Divine, Ndunu.** *ZEUS Technical Manual.* Estados Unidos: British Telecommunication plc., 1999.
- [35] **Nwana, Hyacinth, Ndumu, Divine y Lee, Lyndon.** *ZEUS: An Advanced Tool-Kit for Engineering Distributed Multi-Agent Systems. CiteSeer.IST.* [En línea] [Citado el: 22 de Septiembre de 2007.] <http://citeseer.ist.psu.edu/nwana98zeus.html>.
- [36] *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial.* No.18 (2003), pp. 51-63. ISSN: 1137-3601. © AEPIA.
- [37] **Despain, Wendy.** *Professional Techniques for Video Game A K Peters Ltd.; 1 edition 2008.*
- [38] **Aguilar, J. Hidrobo, F. y Cerrada M.** *A Methodology to Specify Multiagent Systems. Lecture Notes in Artificial Intelligence, 4496:92–101, 2007.*



- [39] **Ferreiro Barreiro, Divina.** *Estandar FIPA "Foundation For Intelligent Physical Agents"*. Vigo : s.n., 2002.
- [40] **Vieira, Sebastian.** *FIPA - ACL*. Boston: s.n., 2000.
- [41] **Botía, Juan A.** *Construcción de Agentes Software con JAVA+JADE* Madrid : s.n., 2005.
- [42] **Catá, Jordi y Méndez, Alex.** *Comparativa de Motores Gráficos para Videojuegos*. Madrid : s.n., 2002.
- [43] **Nischke, Benjamin.** *Professional XNA Game Programming: For Xbox 360 and Windows*. Hannover : Wrox Press, 2007.
- [44] **Salen, Katie. Zimmerman, Eric.** *Rules of Play: Game Design Fundamentals*. The MIT Press, 2004
- [45] **Aguilar, J., Rivas F., Hidrobo F., Cerrada M.** *Análisis y Diseño de Sistemas Multiagente Usando la Metodología MASINA*. Reporte Técnico – Universidad de los Andes (CEMISID), January 2004.
- [46] **OGRE.** *Ogre 3D. Open source Graphics Engine*. [En línea] 03 de Marzo de 2007. [Citado el: 25 de Octubre de 2007.] <http://www.ogre3d.org>.
- [47] **Rumbaugh, J, y otros.** *Object-Oriented Modeling and Design*. s.l. Prentice-Hall, 1991.
- [48] **Lopez, Jaime, Yuly Botina.** *Agentes Inteligentes para adaptar la presentación de contenidos para cursos de educación en línea*. Tesis de Pregrado Universidad del Cauca. 2008.
- [49] **Pressman, Roger.** *Ingeniería del Software*. Mexico : Mac Graw Hill, 2005.
- [50] **Ouarda Bourass, Miguel Angel Almasa.** *Metodologia GAIA*. 0506, s.l. : Revista TAIA, Vol. 1.
- [51] **Patricia Charlton, Yan Chen, Fredrik Espinoza, Abe Mamdani, Olle Olsson, Jeremy Pitt, Fergal Somers, and Annika Waern.** *An open agent architecture supporting multimediaservices on public information kiosks*. In Proceedings of PAAM'97, London, U.K 1997).
- [52] **AMASE:** *Agent-based Mobile Access to Information Services* [En línea] 18 de Diciembre de 2001. [Citado el: 09 de Diciembre de 2008.] <http://cordis.europa.eu/infowin/acts/analysys/products/thematic/agents/ch3/amase.htm>
- [53] **ABROSE:** *A Co-operative Multi-Agent Based Framework for Electronic Marketplace* [En línea] 18 de Diciembre de 2001. [Citado el: 09 de Diciembre de 2008.] <http://cordis.europa.eu/infowin/acts/analysys/products/thematic/agents/ch3/abrose.htm>
- [54] **Chess D., Harrison C., Kershenbaum A,** "Mobile Agents: Are They A Good Idea?". IBM Research Division. T. J. Watson Research Center. 1995. [Citado el: 09 de Diciembre de 2008] URL - <http://citeseer.ist.psu.edu/28994.html>.
- [55] **Mejía Salaza Maria Helena,** *Aplicación de la Metodología GAIA, ANFORA* ISSN 01216538 Volumen 19 Serie 19 Págs. 119 - 138 , Univ Autónoma de Manizales, 2004
- [56] **Ierache, Jorge Salvador.** *Elaboracion de una aproximacion metodologica para el desarrollo de software orientado a sistemas multiagente*. Madrid-España : s.n., 2003.
- [57] **Gómez Sanz, Jorge J.** *Metodologías para el desarrollo de sistemas multi-agente*, Facultad de Informática, Universidad Complutense, Avda. Complutense Madrid, ISSN: 1137-3601, 2003
- [58] **Morales Pedro Cuesta, Gómez Rodríguez Alma María, Rodríguez Martínez Francisco Javier,** *Desarrollo de un Sistema Multi-agente con MaSE y JADE*, Revista de la Asociación de Técnicos de Informática, ISSN 0211-2124, Nº. 170, 2004
- [59] **Henao Cálad Mónica,** *CommonKADS-RT: Una Metodología para el Desarrollo de Sistemas Basados en el Conocimiento de Tiempo Real*, Tesis doctoral, Valencia, España. Abril de 2.001
- [60] **Larman, Craig.** *UML y Patrones*. 2003.
- [61] **Baiz Quevedo, Frank.** *La construcción del personaje*. [En línea] [Citado el: 9 de septiembre de 2007.]



- [62] **Bacca, Alvaro.** “Simon Y La Tuberculosis: Experiencia Aplicativa Del Videojuego Como Herramienta Emergente Para Educar”. Tesis de Pregrado – Diseño Grafico. Universidad del Cauca. 2008.
- [63] **Kirmse, Andrew.** *Game Programming Gems 4.* Massachusetts : CHARLES RIVER MEDIA.
- [64] **Sánchez Crespo, Daniel Dalmau.** *Core Techniques and Algorithms in Game Programming.* Barcelona: New Riders Publishing, 2003.
- [65] **Bourg, David y Seeman, Glenn.** *AI for Game Developers .* United States : O'Reilly Media, 2004.
- [66] **J. Rumbaugh, M. Blaha, W. Premerlani and V.F Eddy.** *Object-Oriented Modeling and Desing.* s.l. : Prentice Hall, 1991.
- [67] **Grasia, grupo de agentes software: Ingenieria y Aplicaciones.** Grupo de agentes software: Ingenieria y Aplicaciones. [En línea] [Citado el: 13 de agosto de 2007.] <http://grasia.fdi.ucm.es/Spain/index.php>.
- [68] **Francisco Jose Gallego Duran, Faraon Llorens Largo, Ramon Rizo Aldeguer.** *Breve Analisis de algunas metodologias de diseño de SMA.* San Vicente del Raspeig : s.n., 2004.
- [69] **Dautenhahn, K.; Bond, A.H.; Canamero, L.; Edmonds, B.** *Socially Intelligent Agents Creating Relationships with Computers and Robots Series: , Vol. 3* ISBN: 978-1-4020-7057-0