

Universidad de las Ciencias Informáticas

"Facultad de Entornos Virtuales"



Trabajo de Diploma para optar por el título
de Ingeniero en Ciencias Informáticas

Título: Módulo de tareas de juego para la intervención de la Discalculia.

Autores: Liliam Bárbara Morín Rentería
Luis Alberto Montoya Acosta

Tutor: Ing. Yulier Casas Estrada
Asesor: MSc. Pedro Carlos Pérez Martinto

Ciudad de la Habana, julio de 2008

Año 50 de la Revolución

DECLARACIÓN DE AUTORÍA

Declaramos que Liliam Bárbara Morín Renteria y Luis Alberto Montoya Acosta somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Liliam Bárbara Morín Renteria

Luis Alberto Montoya Acosta

Firma del Autor

Firma del Autor

Yulier Casas Estrada

Firma del Tutor

Autores: Liliam B. Morín Renteria y Luis A. Montoya Acosta

DATOS DE CONTACTO

Ing. Yulier Casas Estrada (ycasas@uci.cu)

Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI).

Es profesor adiestrado con dos años de graduado.

Imparte las asignaturas de programación.

MSc. Pedro Carlos Pérez Martinto pcpmartinto@uci.cu, adiaz991110@gmail.com

Graduado como Defectólogo (Educación Especial), 1990. Instituto Superior Pedagógico “Enrique José Varona”.

Profesor asistente e investigador agregado. Investigador del Centro de Estudios Sociopolítico y de Opinión, adjunto al Comité Central hasta 2005.

Profesor jefe de la asignatura Metodología de la Investigación Científica – UCI.

Asesor de investigaciones en la facultad 5.

AGRADECIMIENTOS

A la dirección de la Revolución y a la UCI, por permitirnos formar parte de este proyecto y contribuir a nuestra formación profesional.

A nuestro tutor Yulier por su preocupación y dedicación, por guiarnos en cada paso que dimos.

Al profesor Omar por su apoyo incondicional y paciencia con nosotros.

A todas las personas que de una forma u otra contribuyeron con el desarrollo de este trabajo.

En fin a todos los que día a día preguntaban cómo va la tesis.

Lily y Luiso

DEDICATORIA

A mi abuela, por su apoyo, comprensión y su infinito desvelo.

A mi mamá, por darme la vida y esperar lo mejor de mí.

A mi papá, por confiar en mí.

A Panchy, por ser un abuelo sin igual.

A mis tíos Leyda, Rodo, Abdy, la Pitufa, Carli y la Manca, en fin a toda mi familia, por soportarme y prepararme para la vida.

A Luis Alberto, por estar a mi lado en este largo camino.

A las DATILA, por su amistad, y porque simplemente sí.

A Tata y a Lucy, por estar ahí siempre.

A Rey, por tantas molestias a cualquier hora.

A todos mis amigos, que de una u otra forma me aportaron el aliento y el ánimo necesarios para seguir adelante.

A todos mis profesores por forjarme como profesional.

Lily

A mis padres que lo han dado todo por mí, por ser mi guía y ejemplo ayudándome a superar todos los obstáculos que encontré en el camino. A mí hermana por que vivo orgullosa de ella y es la mejor del mundo: lo que soy y seré se lo debo a ustedes.

A mis abuelos, mis tíos, mis primos, a toda mi familia que siempre me dieron su apoyo y ayudaron, por siempre estar y ser especiales.

A Lily y a su familia por brindarme su apoyo.

A los amigos de mi zona, a los socios del apartamento, en fin a mis amistades de la UCI que siempre me brindaron su ayuda, en especial a mi piquete.

A todas las personas que de una forma u otra hicieron posible este sueño.

Luiso

RESUMEN

En la actualidad, a nivel mundial se ha diseñado e iniciado la aplicación de estrategias que permiten convertir los conocimientos y las TIC (Tecnología de la Información y Comunicación) en instrumentos a disposición del avance de la educación. Cuba no se ha quedado atrás, en tal sentido se planifican metas determinadas por los principios de la pedagogía y la didáctica generales y especiales, para tratar la Discalculia, padecimiento que afecta el desarrollo del aprendizaje, desconocido en la actualidad por muchas personas y que afecta a uno de cada diez niños en el mundo. Atendiendo a los problemas que existen con la poca existencia y uso de medios para la intervención de esta enfermedad, el Centro de Neurociencias de Cuba propone una nueva solución que desarrolla de conjunto con la Universidad de las Ciencias Informáticas (UCI).

Como aporte innovador se propone desarrollar un juego para dar solución a la propuesta de los especialistas. Este trabajo se centra en realizar el análisis y diseño de los niveles del juego correspondientes a las tareas matemáticas: estimar cantidades y comparación. Además de la implementación de las funcionalidades principales del Kernel (núcleo del juego) vistas a través del primer nivel del mismo. La solución propuesta resulta interesante, entretenida, atractiva y manipulable gracias a las herramientas y técnicas que se utilizaron en su desarrollo además de que crea las bases para el desarrollo posterior del juego.

Este trabajo es de suma importancia porque es el primero de este tipo en el país y uno de los pocos existentes en el mundo, por lo que constituye un paso de avance en este campo.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	4
1.1 Importancia del tema.....	4
1.2 Definiciones de Discalculia.....	4
1.3 Estudio científico.....	5
1.4 Indicadores.....	6
1.5 Comportamiento en el mundo.....	7
1.5.1 Características demográficas.....	7
1.6 Las nuevas tecnologías aplicadas al desarrollo del proceso de enseñanza especial.....	8
1.6.1 Software educativo.....	8
1.6.2 Realidad Virtual.....	9
1.6.2.1 Aplicación a la enseñanza y sus manipulables.....	10
1.6.2.2 Tipos de Manipulables Virtuales.....	11
1.7 Iniciativas.....	12
1.7.1 The Number Race.....	16
1.8 Herramientas para la creación de aplicaciones didácticas.....	18
1.9 Engines o Motores Gráficos.....	19
1.9.1 G3D versión 7.....	19
1.9.2 Scene Toolkit versión 2.3.1.....	20
1.10 Características visuales de los juegos.....	21
1.10.1 Primera Persona.....	21
1.10.2 Tercera Persona.....	22
1.10.3 1ra vs 3ra.....	22
1.10.4 Generación de Entornos Dinámicos.....	23
1.11 Situación en Cuba.....	24
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	25
2.1 Introducción.....	25
2.2 Objetivo de Automatización.....	25
2.2.1 Intervención del Efecto de Magnitud Numérica y Distancia Numérica.....	25
2.2.2 Intervención de Subitización y Efecto de Magnitud Numérica.....	27
2.3 Información que se maneja.....	28
2.4 Descripción de la solución.....	28
2.5 Generalidades del Sistema.....	29
2.5.1 Uso de la herramienta Scene Toolkit.....	29
2.5.2 Características del lenguaje.....	30
2.5.3 Generación de Entornos Dinámicos.....	30
2.5.4 Juego en 1ra Persona.....	31
2.5.5 Comparación con otros sistemas.....	31
2.6 Reglas del Negocio.....	32
2.7 Modelo del Dominio.....	33

2.7.1 Glosario de Términos	33
2.7.2 Diagrama del Modelo de Dominio.....	35
2.8 Especificación de los requisitos del software.....	35
2.8.1 Requisitos Funcionales.....	36
2.8.2 Requisitos no Funcionales.....	37
2.9 Modelo de Casos de Usos del Sistema	38
2.9.1 Listado de Casos de Uso	39
2.9.2 Diagrama Casos de Uso	41
2.9.3 Expansión de los Casos de Uso	44
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....	54
3.1 Introducción.....	54
3.2 Diagrama de Paquetes de Clases de Diseño.....	54
3.2.1 Diagrama de clases del paquete Level	55
3.2.2 Diagrama de clases del paquete Kernel.....	55
3.2.3 Diagrama de clases del paquete STK.....	55
3.3 Diagrama de Paquetes	56
3.4 Descripción de las clases.....	56
3.5 Diagramas de secuencia	65
3.5.1 Diagrama Principal	65
3.5.2 Jugar Básico Nivel 1	66
3.5.3 Jugar Bajo Presión Nivel 1	66
3.5.4 Jugar Básico Nivel 2	67
3.5.5 Tomar Tarea.....	67
3.5.6 Manipular Animación	68
3.5.7 Otros casos de uso.....	68
3.5.7.1 Solicitar Ayuda Nivel 1	68
3.5.7.2 Solicitar Ayuda Nivel 2	69
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.....	70
4.1 Introducción.....	70
4.2 Implementación	70
4.2.1 Despliegue.	71
4.2.2 Diagrama de Componentes.....	71
4.3 Modelo de Prueba	73
4.3.1 Descripción de los CU de Prueba de Integración.	74
CONCLUSIONES.....	80
RECOMENDACIONES	81
BIBLIOGRAFÍA.....	82
ANEXOS.....	85
Anexo 1: Diagrama de Clases	85
Anexo 2: Diagrama de Secuencia.....	89
GLOSARIO DE TÉRMINOS GENERAL.....	94

INTRODUCCIÓN

En la actualidad se están produciendo profundas transformaciones sociales, económicas y políticas motivadas por el desarrollo de las TIC, se está afrontando un proceso de grandes cambios tecnológicos que permiten el desarrollo de la sociedad. Los procesos de enseñanza y aprendizaje actuales sufren cambios constantemente motivados por el auge de estas tecnologías que plantean nuevos paradigmas y revolucionan el mundo de la escuela, generando nuevos enfoques educativos centrados en el aprendizaje del alumno.

En el mundo existen diferentes formas de trastorno del aprendizaje de las matemáticas y las dificultades que estas presentan varían de acuerdo a la persona y afectan de modo diferente en cada momento su ciclo vital. Un ejemplo de esto se encuentra en la Discalculia que es un término que hace referencia a un amplio rango de problemas relacionados con el aprendizaje de las habilidades matemáticas. En el caso de los niños en edad escolar a medida que el aprendizaje de las matemáticas avanza, pueden tener dificultades en resolver problemas matemáticos básicos; lo preocupante no es el diagnóstico o la detección de la discapacidad sino la poca existencia de medios que permitan realizar su efectiva intervención.

Existen diferentes organizaciones que se han dado la tarea de combatir esta dificultad y gracias a esfuerzos de especialistas y científicos que han realizado varias investigaciones, se han presentado algunas herramientas y software que han podido un poco frenar esta discapacidad, aunque no ha sido suficiente, ya que en muchas partes del mundo todavía no existe un profundo conocimiento acerca de este tema.

Cuba no se ha quedado atrás, ha diseñado e iniciado la aplicación de estrategias que permiten convertir los conocimientos y las TIC en instrumentos a disposición del avance de la educación. En tal sentido se planifican metas ambiciosas determinadas por los principios de la pedagogía y la didáctica generales y especiales, que sean efectivos en la intervención de la Discalculia.

Por todo lo anterior el **problema científico** de este trabajo queda formulado:

¿Cómo propiciar el desarrollo de habilidades matemáticas de los escolares de sexto grado con dificultades en su aprendizaje?

El **objeto de estudio** es el proceso de desarrollo de juegos.

El **campo de acción** es el proceso de desarrollo de juegos en la intervención de la Discalculia en los niños de sexto grado.

Este trabajo se propone como **objetivos**:

- Diseñar los niveles correspondientes a las tareas matemáticas: estimación de cantidades y comparación, de un juego para la intervención de la Discalculia en escolares de 6to grado.
- Implementar los componentes principales del Kernel del juego para la intervención de la Discalculia.

Durante el desarrollo de este software se realizarán una serie de tareas para cumplir con el objetivo propuesto, como son:

- Estudio de la información sobre software o sistemas existentes para la intervención de la Discalculia.
- Análisis de la bibliografía relacionada con el tema.
- Análisis y Diseño del módulo de tareas del sistema.
- Determinación de las herramientas para el desarrollo del sistema.
- Implementación de un demo con las funcionalidades principales del Kernel vistas a través del primer nivel del juego.

Como resultado de este trabajo se pretende obtener una primera versión del Kernel de un juego en un escenario 3D, además del análisis y diseño de los niveles relacionados con dos de las

cuatro tareas que se llevan a cabo en la intervención de la Discalculia en niños de sexto grado, será flexible a la actualización futura y extensible a la aplicación para intervenir otros problemas como la dislexia.

El trabajo está dividido en 4 capítulos, en el primer capítulo se explica el trastorno y el estado en que se encuentra, se mencionan y explican cada uno de los sistemas utilizados actualmente, así como las técnicas que se han utilizado para la solución del problema. En el segundo capítulo se detalla la propuesta del sistema, que se hace para resolver el problema, así como también se enumeran los requisitos funcionales y no funcionales entre otras cosas importantes para la comprensión del mismo. En el tercer capítulo se muestra el análisis y diseño del sistema y se da una idea de cómo funciona el sistema visto desde dentro mediante los diagramas de clases y de secuencia que son necesarios para la comprensión del software. Por último se verá el capítulo cuatro que es el de implementación y prueba, donde se define la línea base ejecutable de la arquitectura de la aplicación, se realizan los diagramas de componentes y se definen los casos de prueba para poder realizar las pruebas de caja negra en el futuro.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Importancia del tema

Las Discalculias han recibido mucha menos atención que las alteraciones del código escrito (dislexias y las disgrafías). El grado de presión social no puede ser el factor fundamental, ya que sin un manejo adecuado de las matemáticas se ve seriamente afectada la existencia del ser humano (ello implica desde las compras diarias, el cobro de un salario, la estimación del valor de los objetos en el mercado, la disponibilidad del dinero y hasta la capacidad de ubicarse en una ocupación laboral por más sencilla que sea). Por otra parte, con frecuencia, los trastornos de las facultades aritméticas, han sido durante largo tiempo (y en gran parte todavía hoy), adscriptas a déficit primitivos del lenguaje, perdiendo entidad propia. (1)

1.2 Definiciones de Discalculia

“La Discalculia constituye un trastorno en la competencia numérica y las habilidades matemáticas las cuales se manifiestan en niños de inteligencia normal que no poseen lesiones cerebrales adquiridas.” (1)

C. Temple

“Es un trastorno estructural de las facultades matemáticas el cual se origina en alteraciones congénitas o genéticas de aquellas partes del cerebro que constituyen los substratos anatómicos y fisiológicos directos para la maduración de dichas facultades de manera acorde a la edad y sin que haya perturbaciones simultáneas en las funciones mentales generales.” (1)

L. Kosc

La discalculia es un trastorno específico del aprendizaje de los procesos cognitivos, razonamiento matemático y sistemas operacionales sin compromiso intelectual. No tiene nada que ver con la inteligencia, entendida globalmente, ni con los estudios que tenga el sujeto o los trastornos afectivos, aunque generalmente puede estar asociada a otras patologías pero sí está estrechamente relacionada con el

sistema de comunicación no verbal, es decir, que realiza percepciones globales y emplea procesamiento simultáneo para, posteriormente, transcodificar y analizar la información, lo cual origina dificultades en el razonamiento lógico-formal como, seriación, clasificación, descomposición, simbolización, correspondencia, organización, operación y comprensión aritmética. (2)

Puede presentarse bajo dos modalidades, de tipo primaria y secundaria. La primaria, o adquirida es aquella que afecta el inicio de las operaciones como la memorización, o el uso de las cantidades y los procedimientos, bien sea por consecuencia de compromisos afásicos (deterioro para la lectura y escritura de números), dificultades viso-constructivas y viso-espaciales, donde el niño puede entender los hechos matemáticos pero tener dificultades poniéndolos y organizándolos en el papel, entre otras. La discalculia secundaria se refiere a las que cursan en el desarrollo y pueden ser de tipo verbal, como por ejemplo, los niños con dificultad para designar y relacionar. Las de tipo operacional donde interviene el procesamiento matemático; léxicas y gráficas en las cuales se aprecia dificultad para leer y escribir números y cantidades. Entre otros tipos de discalculia está la protognósica, que requiere de la memoria de trabajo para manipular los datos y la ideognósica cuya relación es contar con la idea a fin de resolver cálculos mentales. (2)

1.3 Estudio científico

Fue la frenología de Gall y Surpzhim durante la segunda mitad del siglo XIX quienes aparte de introducir la noción básica de un cerebro diferenciado según funciones (germen de la noción de modularidad), atribuyeron una región a la que denominaron <cantidad> y otra <cálculo>. A fines del siglo XIX y en publicaciones del XX, es el oftalmólogo Hinshelwood quien hace mención por primera vez a un caso de Discalculia (caso originalmente descrito por Stephenson en 1905). El término “Acalculia” es acuñado por el neurólogo sueco Henschen, quien en dos artículos (1919-1920) hace referencia a la posibilidad de que la capacidad de calcular pueda afectarse de manera relativamente independiente por una lesión más o menos circunscrita del lóbulo parietal izquierdo (pacientes diestros). En 1926 Hans Berger, realiza una clasificación entre acalculias primarias y secundarias. (1)

Experimentos realizados en la década del 40 por el científico y humanista norteamericano L. Ronald Hubbard dieron como resultado que una persona además de su mente analítica tenía una segunda mente,

que Hubbard llamó **reactiva** porque en lugar de razonar, antes de actuar reaccionaba automáticamente ante una situación de peligro. Según Hubbard descubrió, y actualmente se dice que es un hecho científico comprobado, la mente reactiva y su contenido engrámico son el único origen de las enfermedades mentales y la delincuencia. En el caso de la Discalculia son órdenes comunes contenidas en el engrama, generalmente implantados por padres y maestros, tales como “**siempre confundes los números**”, “**calculas mal**”, “**siempre fallas en matemáticas**”, o cualquier otra sobre tener problemas en este tipo de operaciones. (3)

Científicos de la Universidad de Londres han demostrado la implicación del **sucu intraparietal** derecho del cerebro en la discalculia, en un trabajo realizado. Los autores del estudio fueron capaces de inducir discalculia, en sujetos normales, mientras trataban de realizar un ejercicio matemático consistente en comparar dos cifras de distinto tamaño y magnitud. Los sujetos normales, durante un corto espacio de tiempo tras la estimulación, justo en el momento de tener que establecer la comparación de ambas cifras, tuvieron un patrón de respuesta similar a la de los sujetos con discalculia que fueron usados como controles. La ubicación cerebral de donde tiene lugar el trastorno, de forma específica en la discalculia, podrá permitir en el futuro desarrollar técnicas de intervención específicas para tratar de superar esta dificultad. (4)

A manera de valoración la discalculia es un problema neurológico (cerebral) real. Es causado por una debilidad en la estructura de las neuronas y los conductos que subyacen en el concepto de los números que para un cerebro normal parece tan intuitivo. La clasificación de la discalculia como un fenómeno cerebral es lo que ofrece la posibilidad de un entrenamiento específico remedial basado en la comprensión de la neurociencia de las estructuras de las neuronas y los conductos que subyacen en el concepto de los números.

1.4 Indicadores

Los trastornos de Discalculia son frecuentes durante el desarrollo, sin embargo son los menos estudiados y muchas veces mal diagnosticados, la más estudiadas son las viso espaciales, o aquellas con indicadores claros de fallas de posicionamiento, dirección, inversiones, sustituciones, entre otras, a pesar de que en

este trastorno hay compromiso lingüístico, perceptual y de memoria, muchas veces como comorbilidad de otras patologías, como la dislexia, déficit de atención, problemas de memoria. Si de indicadores se trata, los más comunes son la dificultad en los conceptos, como confusión de símbolos abstractos, sentido pobre en el proceso de reversibilidad, dificultad lingüística, o poca comprensión de la explicación verbal, incapacidad para mantener dos variables simultáneas, visualización pobre y organización espacial, fallas de atención, o inconsistencia. (2)

1.5 Comportamiento en el mundo

1.5.1 Características demográficas

En el mundo, según las estadísticas, tan sólo un 1 % conoce la discalculia. Las asociaciones hacen campaña para que se extienda el conocimiento de esta dificultad, que afecta entre el 3-6% de la población mundial infantil, un porcentaje similar al de otros trastornos del desarrollo como la dislexia o el déficit de atención con hiperactividad.

- 6,4% en Eslovaquia
- 6,3% en los Estados Unidos de América
- 4,4% en Alemania
- 4,7% en Suiza
- 1,3% en el Reino Unido
- 6,5% en Israel (5)

Recientemente un estudio dirigido por Brian Butterworth del Instituto de Neurociencia Cognitiva del University College de Londres, y presentado en el Festival de Ciencia de Cheltenham, Reino Unido, se dieron a conocer unos datos sorprendentes: Una vez examinados a 1.500 niños/as se pudo detectar una incidencia de un 3% a un 6% de casos de discalculia, mientras el porcentaje de dislexia resulto ser mas bajo. Butterworth hizo un llamamiento a las autoridades británicas para detectar e intervenir de forma temprana los casos de discalculia en la población escolar. (6)

En Chile no existen cifras oficiales sobre niños con dificultades de aprendizaje, de las matemáticas (discalculia) o lecto-escritura (dislexia) pero los especialistas sostienen que el número de consultas con respecto a este tema ha ido en aumento en los últimos años. (7)

Como se ha observado no existe suficiente información acerca de este padecimiento en el mundo, el hecho de que solo el 1% conozca este trastorno parece que lo dice todo, aunque cabe destacar que existen organizaciones que se han dado la tarea de dar a conocer al mundo este trastorno.

1.6 Las nuevas tecnologías aplicadas al desarrollo del proceso de enseñanza especial

1.6.1 Software educativo

Hablar de software educativo en una época caracterizada por múltiples cambios en los distintos escenarios de la vida cotidiana implica reconocer cómo los procesos de modernidad por los que está atravesando la sociedad actual, han generado novedosas y muy diversas formas de interacción humana que, de una u otra forma, han afectado la práctica y el ejercicio del proceso de enseñanza aprendizaje. De ahí que el docente en activo tenga que enfrentarse a las TIC, a las que en numerosas ocasiones ve como la solución a sus problemas de enseñanza. (8)

Estudios realizados demuestran que los niños que han sido intervenidos con software educativo han mejorado en mayor grado, tanto cualitativa como cuantitativamente, en sus puntuaciones frente a los otros que no han sido intervenidos con estos programas informáticos. La intervención con el software educativo forma parte de una respuesta individual y contextualizada mucho más global. Sin embargo, se demuestra suficientemente la eficacia que representa el software educativo en la intervención de niños con trastornos generalizados del desarrollo. Estos programas le sirven para potenciar su autonomía personal y ayudan a su integración social. El niño con trastornos del desarrollo de aprendizaje puede motivarse a mostrar y compartir sus logros, a hablar, a interactuar con el ordenador o con alguien más generando verdaderos puentes cognitivos y desarrollando el interés por conocer. (9)

Con el software educativo los niños descubren su potencial, ya que ciertos programas les despiertan tanto interés, que se motivan a aprender. Muchos mejoran su autoestima, porque descubren que son capaces de entender cosas de las que no se creían capaces. Para algunos ha significado un verdadero descubrimiento de sus capacidades, y para otros la confirmación de que tienen una gran facilidad para aprender y entender los fenómenos que los rodean. Con estos programas se logra una alta concentración ya que los niños trabajan con material multimedia, y aprenden lecciones a través de imágenes, acciones de personas y diálogos. Uno de sus mayores aportes, en este sentido, es que los contenidos que se enseñan se transmiten a través de sentidos como la vista y la audición. Estas experiencias sensoriales permiten que el aprendizaje sea más efectivo y duradero.

Destacar que a pesar de lo positivo e impactante que es el software educativo, se puede decir que este constituye un recurso didáctico y no un fin en el aprendizaje. Por lo que debe ser considerado y utilizado como una herramienta más para mejorar el aprendizaje de los niños y no desmarcarse de la insustituible labor psicopedagógica que realizan los profesionales que interactúan directamente con los niños con necesidades educativas especiales. (10)

1.6.2 Realidad Virtual

La realidad virtual es un fenómeno que se percibe a través de los sentidos que se desarrolla en un espacio ficticio, en un paréntesis de la realidad. La realidad virtual es un sistema interactivo que permite sintetizar un mundo tridimensional, creándose una ilusión de realidad. Es una técnica fotográfica de 360 grados, que permite moverse hacia arriba o hacia abajo, realizar acercamientos o alejamientos; se tiene el control absoluto de los movimientos; y ofrece un realismo de estar en el sitio.

Con la realidad virtual se simulan todas las posibles percepciones de una persona, como los gráficos para la vista, sonido, tacto e incluso sensaciones de aceleración o movimiento. Todas estas sensaciones diferentes deben ser presentadas al usuario de forma que se sienta inmerso en el universo generado por el ordenador, hasta el punto de dejar de percibir la realidad y ser engañado, sentirse transportado (al otro lado de la pantalla) como si de un universo nuevo se tratase. (11)

Este mundo sintético puede mostrar la realidad (como en un simulador de vuelo) o ser totalmente artificial (como en un videojuego). Básicamente, la realidad virtual puede ser inmersiva y no inmersiva.

En la **realidad virtual inmersiva**, el usuario se encuentra en un entorno tridimensional (3D), con el que interactúa mediante guantes, gafas o un casco especial. En la realidad **virtual no inmersiva**, el usuario interactúa con un entorno representado en su monitor mediante el teclado, el ratón u otros dispositivos. Es una experiencia multisensorial y bidimensional; su coste es bajo y es aceptada con facilidad y rapidez por los usuarios. (12)

Las aplicaciones de realidad virtual requieren hardware y software adecuados. Tienen por objetivo que el usuario se sumerja en un entorno sintético de forma interactiva, autónoma y en tiempo real.

1.6.2.1 Aplicación a la enseñanza y sus manipulables

La realidad virtual es una tecnología especialmente adecuada para la enseñanza, debido a su facilidad para captar la atención de los estudiantes mediante su inmersión en mundos virtuales relacionados con las diferentes ramas del saber, lo cual puede ayudar en el aprendizaje de los contenidos de cualquier materia.

Los estudiantes no sólo pueden leer textos y ver imágenes dentro de un casco de realidad virtual, sino que además pueden escuchar narraciones, efectos de sonido y música relacionados con el tema que están aprendiendo. Por medio del uso de los guantes de datos, los estudiantes pueden sentir la textura, dimensiones e inclusive la temperatura de objetos virtuales que existen dentro del mundo virtual.

Este es un recurso didáctico del que los profesores se pueden servir para motivar y atraer la atención de los estudiantes a través de los gráficos tridimensionales de alta calidad y de alto grado de interactividad ofrecida por los sistemas virtuales. Cada vez es mayor el número de centros de enseñanza en los que se utilizan aplicaciones de este tipo. (13)

Resultados encontrados por investigaciones sobre cómo la tecnología puede mejorar el aprendizaje, se enfocan en una serie de ayudas virtuales que facilitan el aprendizaje llamadas: Manipulables Virtuales, y ayudan a los estudiantes a entender conceptos esenciales en áreas como matemáticas mediante la representación de temas, en forma más sencilla. Las investigaciones han demostrado que la tecnología

puede impulsar profundos cambios en lo que aprenden. Utilizando la capacidad del computador para posibilitar simulaciones, enlaces dinámicos e interactividad, el estudiante regular puede alcanzar un dominio extraordinario de conceptos sofisticados. Algunos de estos manipulables (Visualizaciones, Modelos y Simulaciones) han probado ser herramientas poderosas para enseñar conceptos matemáticos y científicos. (14)

1.6.2.2 Tipos de Manipulables Virtuales

Simulaciones: Una de las formas más efectivas y fáciles de integrar las TIC en las materias del currículo es mediante el uso de simulaciones. Muchas de estas se encuentran disponibles en Internet para propósitos educativos, en la mayoría de los casos sin costo. Algunas son interactivas, es decir, que permiten al estudiante modificar algún parámetro y observar en la pantalla el efecto producido por dicho cambio. Otras posibilitan además configurar el entorno, esto es, que los educadores pueden programarlas para que aparezcan distintos elementos y diferentes tipos de interacciones. Una de las cualidades que poseen las simulaciones es el alto grado de motivación que despiertan en los estudiantes y poder llegar a resultados a través de un proceso de ensayo y error (orientado por el profesor). Este proceso les permite descubrir conceptos matemáticos e ir construyendo un puente entre las ideas intuitivas y los conceptos formales.

Visualización: La visualización juega un papel muy importante en la enseñanza de las matemáticas y su mayor impacto se puede ver cuando los estudiantes logran visualizar un concepto o problema. “Visualizar un problema significa entenderlo en términos de un diagrama o de una imagen visual. La visualización en matemáticas es un proceso en el que se forman imágenes mentales con lápiz y papel, o con la ayuda de tecnología, y se utiliza con efectividad para el descubrimiento y comprensión de nociones matemáticas”.

Representaciones Tridimensionales: Los computadores hacen posible la creación de imágenes interactivas de objetos tridimensionales que se pueden rotar y cambiar de posición para facilitar su estudio.

Juegos: Los juegos enseñan a los escolares a dar los primeros pasos en el desarrollo de técnicas intelectuales, potencian el pensamiento lógico, desarrollan hábitos de razonamiento y enseñan a pensar

con espíritu crítico. Estos por la actividad mental que generan, son un buen punto de partida para la enseñanza de la matemática, y crean la base para una posterior formalización del pensamiento matemático. El juego, debido a su carácter motivador, es uno de los recursos didácticos más interesantes que puede romper la aversión que los alumnos tienen hacia la matemática. (14)

1.7 Iniciativas

En el mundo existen diferentes organizaciones encargadas de sensibilizar a la sociedad acerca de los diferentes problemas de aprendizaje y dar respuesta a la necesidad tan inminente de atender a esta parte de la población. Un ejemplo de esto es el **DISLECAN** cuyo objetivo es el de guiar a toda persona interesada en realizar un diagnóstico sobre su posible dificultad de aprendizaje, para que pueda contactar con los especialistas adecuados y más familiarizados con los últimos resultados que se hayan podido obtener en investigaciones, tanto a nivel nacional como internacional.

En septiembre del 2003 se celebró un mini simposio sobre diseño de software para la rehabilitación de la discalculia en Francia. Uno de los propósitos de este proyecto fue el de unir científicos especializados en el tema de la discalculia para intercambiar experiencias. Se presentaron diferentes herramientas y software para combatir la discalculia, entre los que se destacaron:

Brian Butterwords mencionó un estudio en Rochester en adultos y en niños con juegos computarizados donde los resultados muestran que las habilidades de la atención mejoraron considerablemente.

Sharon Griffin, Clark University. USA coautor del programa Number World (Mundo de Números) explicó que este es un programa matemático basado en la investigación para niños que abarca 4 niveles y que se basa en la enseñanza del conocimiento y comprensión de los números.

El desarrollo del programa fue motivado y guiado por tres grandes ideas de la investigación cognitiva desarrolladora.

- La primera fue un modelo de desarrollo del conocimiento de los números.

- La segunda fue la realización de la estructura conceptual línea mental numérica que la mayoría de los niños construyeron, que vincula la comprensión de los números con su comprensión de la cantidad, proporciona las bases de la comprensión numérica y para el aprendizaje exitoso de la aritmética en la escuela.
- La tercera es el descubrimiento de que hay un sustancial número de niños que viven en comunidades de bajo ingreso y que entran a la escuela a la edad de 6 años sin esta estructura conceptual.

Griffin plantea que la mayoría de las actividades de este programa están en formato de juegos y que están agrupadas en 5 mundos separados, cada uno con un lenguaje visual diferente para representar números. Para ilustrar el programa Griffin seleccionó una secuencia de actividades en el mundo de las líneas porque la representación lineal proporciona las analogías espaciales más cercanas para la estructura conceptual de la línea numérica mental que el programa trata de enseñar.

Anna Wilson de Nueva Zelanda presentó un software para la rehabilitación del conocimiento numérico central. Este software remedial para la discalculia se desarrolla por INSERN en colaboración con OECD para la investigación neurológica (cerebral) y el proyecto de sitio Web para las ciencias del aprendizaje. Sus objetivos generales son diseñar un software de juegos adoptivos para la comprensión numérica de los niños. Para este fin ellos adaptarán tests para la discalculia y sus tipos. Ellos llevarán a cabo un estudio de pilotaje para evaluar la efectividad del software de juegos para la solución de discalculia que ocurre conjuntamente con la dislexia o con la discalculia pura.

Los 4 principios instructivos principales de este software son:

- Proporcionar una extensiva práctica de comparación numérica para incrementar la precisión y la confianza en la representación cuantitativa
- Fortalecer los vínculos entre la magnitud y los códigos verbales numéricos.
- Enfatizar en el vínculo entre los números y el espacio
- Enfatizar en el conteo por la calibración de la representación de la cantidad.

Wilson después demostró una propuesta de software de interface para esta intervención con una representación de una línea de números en la pantalla. La línea de números se mueve a lo largo con los

carros, puntos y pequeñas animaciones. Se adicionan sonidos como aplausos y otros. En la medida en que el programa se ejecute, eventualmente el andamiaje verbal se elimina y solo queda la representación de dígitos arábigos con magnitudes numéricas incrementadas y distancia numérica disminuida entre los números. Así como la inclusión de operaciones numéricas de adición y substracción. El juego adoptivo tendrá por consiguiente distintas dimensiones difíciles de distancia, magnitud, notación, operación y se incluyen en el tablero obstáculos para agregar un nivel de complejidad.

Bruce MacCandlis mencionó un programa desarrollado por Dan Schuartz donde el usuario realmente “enseña al agente” lo que podría ayudar a hacer el programa más interactivo.

Diana Laurillard sugiere una red de juegos donde los jugadores compiten y pueden verbalizar y hablar después de las matemáticas. Como alternativa el grupo sugiere que el programa puede incorporar y realizar predicciones en la pantalla para inducir la interacción. (15)

En Italia se ha trabajado también con este problema y se han hecho algunos trabajos para la discalculia. Existe un software didáctico realizado por el Profesor Svano Pulga en colaboración con el Dr. Enrico Savelli donde se trabaja en:

- El conteo de los elementos numéricos
- Ordenamiento numérico
- Signos de comparación
- Valor posicional de la cifra
- Lectura del número a la letra y viceversa

Estos ejercicios se concibieron para la superación de las dificultades y la reducción de la discapacidad en el área de las operaciones lógico-matemáticas.

En Colombia se realizó un software interactivo de educación infantil orientado a la Web para el aprendizaje de las operaciones básicas de las matemáticas por varios autores y la Fundación del Instituto Tecnológico en Cartagena.

Es un proyecto matemático que consiste en favorecer el aprendizaje y desarrollo intelectual de los niños, por medio de actividades de identificación de conceptos matemáticos, que contribuyan con el proceso de

maduración mental, para que el niño construya un pensamiento lógico y crítico a través de un mundo de juegos didácticos. Los niños recibieron la orientación para realizar las operaciones a través de un software interactivo o multimedia que permita la interacción del niño con el sistema. (16)

En Chile muchas empresas tecnológicas han decidido tenderles una mano a los niños con este padecimiento mediante el diseño de programas que los ayudan a integrarse mejor, tanto en el mundo educativo como en el social. (7)

Existen varios sitios de Internet que ofrecen software y juegos orientados al aprendizaje y desarrollo de las habilidades matemáticas. Entre estos se encuentra el sitio español **Software Educativo**, contando con programas especialmente ideados para adquirir y desarrollar las principales capacidades cognitivas (memoria, atención, razonamiento, coordinación visomotora).

Entre los programas cuenta con el de Reducción de Hermes que es un sistema interactivo especialmente diseñado para apoyar el tratamiento y corrección de déficit de aprendizaje de niños y para ayudar a prevenir y evitar de forma temprana posibles casos de fracaso escolar. Por sus características, también constituye una herramienta eficaz para la integración de la diversidad en las aulas. El programa ofrece 60 actividades interactivas de reeducación, todas ellas con 10 niveles de dificultad, destinadas al refuerzo y recuperación de las materias de matemáticas y lenguaje, así como de potenciación y desarrollo de las habilidades de memoria, atención, concentración, percepción, orientación y cálculo. Hermes pone a disposición de los centros escolares, los logopedas, los psicólogos, los padres, etc. un eficaz sistema de reeducación que permite apoyar tanto la corrección de problemas leves de aprendizajes como casos más graves: discalculia, dislexia, disgrafía. (17)

Otro sitio donde se habla de este tema es el **Tutor (Profesor)**, aquí se encuentra una serie de productos especializados en el diagnóstico y tratamientos para los niños con trastornos "**dys**" (discalculia, dislexia, dispraxia, déficit de atención, etc).

El sitio **Iprase Trentino** ofrece una colección de software de varias materias libremente para las escuelas elementales y medias. Entra las materias se encuentran las matemáticas contando con 57 juegos matemáticos presentando simulaciones, multimedia, hipermedia y tutoriales.

También en el sitio de **Sectormática** se han encontrado varios software diseñados para la educación especial en el aprendizaje de las matemáticas desarrollando diferentes habilidades para el trastorno de la discalculia y acalculia, elaborado por profesionales de distintos países de Latinoamérica en un Proyecto del Consejo Interamericano para el Desarrollo Integral de la Organización de Estados Americanos. Se encuentra además un software elaborado en flash para potenciar a niños con trastornos específicos del cálculo, en este software el niño debe utilizar procesos mentales como; atención y concentración para poder memorizar números.

La Federación austriaca EÖDL (Erster Österreichischer Dachverband Legasthenie) participa de forma continua en proyectos de investigación sobre las dificultades específicas de aprendizaje en Florida, EEUU, y ha adaptado una metodología de diagnóstico y de intervención a nivel pedagógico con todas las personas con este padecimiento.

Sus líneas de investigación sostienen que cada una de las personas con dislexia y/o discalculia percibe la realidad de forma distinta debido a sus diferentes percepciones sensoriales a nivel visual, auditivo y de orientación espacial. (18)

Cabe destacar que de las investigaciones que se han hecho hasta ahora, la mayoría se basan en el aspecto pedagógico-psicológico del aprendizaje de los niños. En cuanto al desarrollo de software o aplicaciones no son todavía lo suficiente como para dar respuesta en todas las partes del mundo.

1.7.1 The Number Race

"The Number Race" o Carrera de Números, fue confeccionado en el 2004 por Anna Wilson & Stanislas Dehaene, es un juego adaptable diseñado para el reabogamiento de discalculia. El software fue diseñado por las posibles causas de un déficit de fondo en la mente. Como meta trazada se encontraba el realce del sentido de número para la cual provee un entrenamiento intensivo en la confrontación numérica y hace énfasis sobre los enlaces entre números y espacio. Fijar con cemento los enlaces en medio de representaciones simbólicas y poco simbólicas de número, para esto se usaron repetidas técnicas

societarias por medio de los códigos árabes, verbales y de cantidad. El papel de información simbólica como una base logra hacer decisiones aumentando progresivamente. Además, los niveles más altos del software son diseñados para proveer el entrenamiento en los hechos de adición pequeña y de sustracción, aunque este entrenamiento está restringido para un rango pequeño de números y provee conceptualmente representaciones orientadas, concretas de operaciones.

Para esto se realizó un estudio donde se escogió una cierta cantidad de niños franceses que completaron un entrenamiento adaptable en la confrontación numérica utilizando media hora cada día, durante cuatro días a la semana sobre un período de cinco semanas. Fueron probados antes y después de la intervención en su actuación en tareas numéricas de fondo: conteo, comprensión transcodificación base-10, enumeración, adición, sustracción y confrontación simbólica, poco simbólica y numérica. Estas tareas fueron diseñadas para medir componentes básicos de cognición numérica, y se basaron en trabajo en pacientes adultos del acalculia así como también en el trabajo reciente en discalculia para el desarrollo. (19)

El objetivo del juego es muy simple. En la pantalla se presentan dos números, y el estudiante debe elegir el mayor de ambos, de tal forma que esa cifra le indicará la cantidad de casillas a avanzar. La presentación de los números se muestra en formatos cada vez más simbólicos, y ocasionalmente, también aparecen representados a través de sencillas sumas y restas. A medida que el juego adquiere más dificultad, el tiempo de respuesta se va acortando. Si no se emite una elección a tiempo, el enemigo se hará con el número mayor.

Sobre el tablero, el alumno compite contra un enemigo, a la vez que debe evitar diversas trampas que lo hacen retroceder. Si su personaje llega a la meta antes que el competidor, es premiado con una recompensa. Una vez obtenidas siete recompensas, se puede liberar un nuevo personaje.

Este programa básico de entrenamiento ha sido diseñado para refinar y automatizar el llamado “sentido numérico”, es decir, la habilidad para entender de forma rápida y automática el significado de los números (por ejemplo, reconocer que el número 7 es mayor que el 3, y que entre ambos existen 4 unidades de diferencia). Dicha habilidad supone la base para la comprensión de las matemáticas y posibilita la capacidad de manipular cantidades, interviniendo en los procesos de comprensión y cálculo aritmético.

Es importante destacar que el diseño del juego, al estar adaptado a las habilidades de cada estudiante y al proporcionar constantemente recompensas al niño sea cual sea su progresión, hace que el estudiante se muestre motivado a continuar con el entrenamiento de su “sentido numérico” y a la vez ayude a desarrollar una visión positiva hacia las tareas matemáticas. (20)

1.8 Herramientas para la creación de aplicaciones didácticas

En este epígrafe, se habla de las principales herramientas de autor (sistemas y lenguajes de autor) que están disponibles hoy día para microordenadores con interfaz gráfica, señalando las ventajas que aportan para producir software educativo.

JClic: La actualización del famoso clic, que tanto se ha usado y se usa sobre todo en educación primaria. Con Java instalado se puede acceder en línea a gran cantidad de aplicaciones. JClic está formado por un conjunto de aplicaciones informáticas que sirven para realizar diversos tipos de actividades educativas: rompecabezas, asociaciones, ejercicios de texto, palabras cruzadas. Las actividades no se acostumbra a presentar solas, sino empaquetadas en proyectos.

WIMS: WWW Interactive Multipurpose Server, (WWW Servidor Interactivo de Matemáticas) es una aplicación que permite crear y hospedar ejercicios interactivos, orientada fundamentalmente a la enseñanza de las matemáticas, la física y la química pero que incluye herramientas de construcción de ejercicios utilizables en cualquier disciplina. Hay servidores WIMS con un número creciente de actividades.

JavaHotPotatoes: Es un conjunto de seis herramientas de autor que le permiten elaborar ejercicios interactivos basados en páginas Web de seis tipos básicos. La interactividad de los ejercicios se consigue mediante JavaScript, posteriormente se podrán publicar dichas páginas en un servidor Web.

Se compone de los siguientes elementos:

JQuiz (ejercicios basados en preguntas)

JCloze (ejercicios de rellenar huecos)

JMatch (ejercicios de emparejamientos)

JMix (ejercicios para desordenar palabras o frases)

The Masher (para crear unidades enlazadas a partir de diferentes ejercicios)

JCross (crucigramas).

Squeak: Una herramienta multimedia que permite desarrollar contenidos. Puesta en práctica completamente equipada de la fuente moderna, abierta con gran alcance en lenguaje de programación y ambiente. Es alto-portable, incluso su máquina virtual se escribe enteramente en el lenguaje que lo hace fácil eliminar errores, analizar, y cambiar. Es el vehículo para una amplia gama de proyectos de los usos de las multimedias, plataformas educativas al desarrollo comercial del uso de la red.

Descartes: Como herramienta didáctica tiene como principal finalidad promover nuevas formas de enseñanza y aprendizaje de las Matemáticas.

Ofrece materiales didácticos que:

- Son controlables por el profesor en un tiempo razonable.
- Son fáciles de usar por los alumnos, no tienen que emplear tiempo en su aprendizaje.
- Cubre los contenidos del currículum correspondiente al curso donde se vaya a usar.
- Son adaptables por cada profesor a la didáctica. (8)

1.9 Engines o Motores Gráficos

Un motor de juego es el componente del software de fondo de un juego u otra aplicación interactiva con gráficos de tiempo real. Provee las tecnologías subyacentes, simplifica auge, y a menudo posibilita el juego para funcionar con plataformas múltiples. Hace referencia a una serie de rutinas de programación que permiten el diseño, la creación y la representación del juego. (21)

1.9.1 G3D versión 7

G3D es un motor de 3D de grado comercial disponible como una fuente abierta (licencia BSD). Es usada en juegos, técnicas demostrativas, trabajos de investigación, simulaciones militares y cursos universitarios.

Proporciona un conjunto de rutinas y estructuras tan comunes que son necesitadas en casi todos los programas de gráficos. Hace que las bibliotecas de bajo nivel como OpenGL y los sockets sean fáciles de usar sin límites en su funcionabilidad y ejecución. G3D es una base altamente optimizada y sólida como una roca sobre la cual construir una aplicación.

G3D no contiene gráfico de escena o rutina GUI (interfaz gráfica de usuario). Por eso, ofrece mucha más flexibilidad a la hora de estructurar los programas. G3D esta dirigido para usuarios que están familiarizados con C++, Direct X y OpenGL.

G3D es construida en OpenGL porque es una plataforma independiente y permite el acceso a más nuevas características que el Direct X. Muchos juegos de primera clase (incluyendo el Doom 3) están escritos con OpenGL. Se puede además implementar cualquier efecto de un juego en G3D. A diferencia de motores de alto nivel G3D no limita el acceso a rasgos de bajo nivel.

G3D acepta imágenes en formato PNG, TGA, BMP, JPG, PCX, PPM, PGM, PBM, ICO y DDS.

Dr. Morgan McGuire ha sido el director del proyecto G3D y guía del equipo de Windows desde el 2000. Es profesor asistente de ciencias de la computación en la Universidad de Williams y consultante de Software 3D. (22)

1.9.2 Scene Toolkit versión 2.3.1

En la Universidad de Ciencias Informáticas de Cuba, la Facultad 5 se planteó la necesidad de crear un grupo de herramientas de apoyo a los programadores de aplicaciones finales, acorde con la tendencia mundial de desarrollo de los llamados “*engines*”.

Scene Toolkit es la primera y una de las herramientas brindadas por el Proyecto de Herramientas de Desarrollo para Sistemas de Realidad Virtual, y surge como una respuesta a esta necesidad, teniendo como objetivo básico agrupar las funcionalidades comunes a cualquier sistema de realidad virtual, de manera que se les facilite el trabajo a los programadores de juegos y simuladores a través de la reutilización de código.

Esta herramienta, en desarrollo actualmente, permite no solamente la visualización de los entornos sintéticos sino además la aplicación de leyes físicas y matemáticas, animaciones, etc., usando las librerías gráficas OpenGL y DirectX, sobre plataforma Windows y Linux. A la vez, se retroalimenta con las necesidades de los programadores que las utilicen, así como de sus investigaciones.

La implementación de la herramienta se hizo en C++ utilizando como IDE el VisualStudio 2003 sobre Windows, y el Code: Blocks (versión svn 3586) sobre Linux. Incluye las estructuras de la *std* para el trabajo con estructuras como listas y otras.

Está dividida en 3 capas fundamentales: una capa *Engine* donde se encuentra todo el procesamiento importante de carga y manejo de objetos; una capa *Renderer* donde se define con que librería gráfica dibujar los entornos (OpenGL y DirectX por el momento), y que permite añadir nuevas librerías por parte de los usuarios de la herramienta en caso de que lo necesiten; y una capa *Application* donde se hace el manejo de eventos del Sistema Operativo específico (Windows y Linux para esta versión), donde el usuario de la herramienta puede desarrollar su propia interfaz para cualquier SO. (23)

1.10 Características visuales de los juegos

1.10.1 Primera Persona

Los **juegos de acción en primera persona**, también llamados **First Person Shooters (FPS)** en inglés, son un género de videojuegos popularizado masivamente en años recientes. En dichos videojuegos, el jugador tiene la impresión de estar dentro del escenario virtual (que generalmente se proyecta en tres dimensiones) y tiene el mismo punto de vista que tendría de estar presente en ese escenario. Además tiene la posibilidad de utilizar armas y vehículos que pueda encontrar en el escenario. El término FPS también es utilizado por equipos de respuesta táctica como la persona que dispara primero. (24)

1.10.2 Tercera Persona

La **tercera persona** es una de las vistas más frecuentes de los videojuegos de estilo aventura gráfica, videojuegos de rol, etc. Esta vista tiene como característica que el personaje que se maneja se ve de cuerpo entero y generalmente de espaldas. En juegos como Silent Hill, Resident Evil o casi cualquier Survival Horror se utiliza con frecuencia esta vista pues da más sensación de inmersión en el juego. Es ideal para juegos en los que la acción se basa más en la búsqueda de objetos o indicios para descubrir la trama del juego. (25)

1.10.3 1ra vs 3ra

Jugadores y expertos en el tema han reflejado sus puntos de vista en cuanto a las ventajas y desventajas que presentan cada una de estas formas de juego.

Cuando se refieren a tercera persona, se puede tomar una perspectiva mas clara de la situación de tu personaje con respecto a los rivales. Desde un escondite se puede apreciar si te ven o no te ven, por lo que se puede controlar más el espacio. Existe un mayor plano donde se ubica tu personaje, enemigos, visión del entorno que te rodea donde se pueden plasmar mejor las estrategias y movimientos. Se desarrolla una mejor jugabilidad como la estética general del juego, viendo mejor al personaje y la situación.

Por otra parte en cuanto a primera persona aporta mayor realismo ya que te llegas a meter por completo en la piel del personaje, la acción está mucho más presente en estos juegos, además de la realidad que aporta jugarlos. Sus gráficos son extraordinarios y la dinámica del juego hace que sea muy entretenido. Por otra parte, presenta menor libertad en los escenarios debido a la poca vista que tienen las cámaras y movimientos más lentos.

Como algo positivo destacar que estudios realizados en la Universidad de Rochester, en Estados Unidos, demuestra que quienes utilizan esta forma de jugar enseñan al cerebro a procesar mejor la información visual.

Muchas opiniones existen en torno a este dilema si un juego en tercera persona es mejor que uno en primera persona o viceversa. Un juego tanto en primera persona o en tercera puede llegar a tener una buena interfaz gráfica y un ambiente bastante real, la utilización o no de una de las dos variantes está en dependencia de lo que se quiera lograr con el juego, aunque puedan influir otros factores.

1.10.4 Generación de Entornos Dinámicos

En el cambiante mundo de los juegos por computadoras gracias al desarrollo alcanzado en el hardware y software (aún insuficientes) y a su demanda en crecimiento (% del mercado de las TIC que ocupa) ha experimentado la necesidad de creación de nuevas tecnologías de hardware y software para su desarrollo. Debido a la gran complejidad que presentan, el incremento del tiempo y esfuerzo está dado por niveles, los cuales son unidades que conforman un juego. Se estima que el tiempo que le toma a un equipo de desarrollo para hacer un juego con 14 horas de contenidos relevantes es de un año.

La manera anterior de generación es completamente manual, otra opción que mejora el tiempo de desarrollo y el costo es la generación automática y proporciona un relevante apoyo al diseño de entorno y a la disminución de su costo. Está comprobado que el tiempo en la generación de un nivel de un juego de interiores de cualquier tamaño puede tomar de 2 segundos a 6 minutos.

Otra manera de generar puede ser dinámicamente, o sea en tiempo de ejecución que aporta grandes ahorros además que motiva al jugador a que juegue nuevamente por su variable contenido; SimCity es un ejemplo de ello.

Por lo general los entornos construidos dinámicamente contienen un mecanismo que regula su construcción en tiempo de ejecución y que proporciona una alta posibilidad de reutilización de los diferentes componentes que conforman el entorno. Entre la generación dinámica de entornos también se encuentra la de Entornos Dirigidos que se centra en direccionar al jugador por un único y posible camino siempre diferente tanto en forma como en contenido y donde un regreso hacia atrás puede o no ocurrir al igual que su fin, siempre dependiendo de los objetivos del mismo. (26)

1.11 Situación en Cuba

En Cuba los pedagogos, maestros y especialistas del aprendizaje han realizado y realizan grandes esfuerzos en la búsqueda de soluciones o software informáticos determinados por los principios de la pedagogía y la didáctica generales y especiales, que sean efectivos en la intervención de la discalculia.

Según datos de la Organización Mundial de la Salud (OMS) señalan que una de cada cuatro personas, en algún momento de su vida, padece de un trastorno mental causado por disfunción cerebral. Algunos niños, con inteligencia normal, no aprenden una asignatura como es el caso de discalculia referida a las dificultades en las matemáticas.

Con un enfoque multidisciplinario Cuba trabaja en la detección e intervención temprana de discapacidades, en particular en los primeros tres años de vida, con el fin de garantizar el cuidado de la salud poblacional. (27)

Además cuenta con el Centro Nacional de Investigaciones Científicas y en esta línea trabaja su Centro de Neurociencias, cuyo espectro de tareas abarca un complejo estudio de la conectividad del cerebro para el pesquisaje de problemas cognitivos en niños y encauzar sus soluciones. Con ese enfoque se enfrentaron los primeros estudios de discalculia en el 2004 en Ciudad de la Habana.

En estos momentos se han tomado medidas concretas por parte de las autoridades, en concreto de los Ministerios de Educación y Sanidad para realizar un estudio a nivel nacional y medir la envergadura de la incidencia de esta dificultad específica de aprendizaje.

La UCI conjuntamente con el Centro de Neurociencias de Cuba se encuentran estudiando la posibilidad de crear un juego para intervenir la discalculia en los niños de sexto grado.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

En este capítulo se detalla la solución propuesta al problema presente a partir del análisis de las características del sistema a realizar, se definen los procesos involucrados realizando un Modelo del Dominio que agrupa todos los conceptos necesarios para lograr la comprensión del módulo a realizar, se enumeran y detallan los requisitos funcionales y no funcionales, se exponen mediante el Diagrama de Casos de Uso del Sistema los actores y los procesos involucrados en el sistema, definiendo las relaciones e interacción que existe entre ellos.

Con la propuesta que se expone se pretende brindar una solución que mejore en gran medida los problemas existentes actualmente en esta materia en el país.

2.2 Objetivo de Automatización

Como se menciona, el objetivo es automatizar el proceso de aprendizaje fundamentado en las tareas de comparación y estimar cantidades de un juego para la intervención de la Discalculia.

En estas tareas se presentarán los estímulos primeramente con tiempo ilimitado. A partir de la primera sesión del juego se calculará el TRP (Tiempo de Reacción Promedio) que demora el niño en dar la respuesta en función del número de objetos, esto lo hace el Algoritmo Adaptativo (AA) así como la determinación de las tareas y estímulos a mostrar.

2.2.1 Intervención del Efecto de Magnitud Numérica y Distancia Numérica

La primera vez que el niño juegue se realizará antes de la intervención de cada tarea un entrenamiento de N repeticiones (N=número de distancia numéricas que usan en los estímulos).

La dificultad de esta tarea depende de la distancia a que se encuentren las numerosidades a comparar (distancias largas vs. cortas) y del formato de presentación de los estímulos.

Tarea:

Comparación de Números

Descripción:

Aparecen rocas con números que cuando caen hay que dar clic sobre ellos para ordenarlos, pero en este nivel existe un margen de error que al alcanzarse provoca la aparición del adversario, que en este caso es el guardián, listo para comenzar a competir.

Procesos:

- Efecto de Distancia Numérica: las numerosidades que se encuentran más distantes (1-9) son comparadas más rápidamente que las numerosidades que se encuentran cercanas (1-2).
- Efecto de Magnitud Numérica: si la distancia numérica es constante, es más rápido el procesamiento de las numerosidades más pequeñas.

Se construirán los estímulos combinando el formato de presentación (no simbólico, arábigo y verbal) y las distancias numéricas (largas: 6, 7, 8 y cortas: 1,2).

Se presentarán primero conjuntos de objetos (* vs. *****) para ser comparados, posteriormente números arábigos (1 vs. 6) y finalmente numerales (uno vs. seis, presentados verbalmente).

Se presentarán primero distancias largas y luego distancias cortas.

Se presentarán sesiones con las mismas distancias para entrenar el Efecto de Magnitud.

Ejemplo:

La complejidad de la tarea aumenta al transitar de distancias numéricas grandes (son más fáciles) a pequeñas (3 y 4 están a una distancia numérica de 1 que es pequeña mientras que 3 y 9 están a una distancia de 6 que es grande)

Ej: ítems sencillos: 1 vs. 9 (distancia 8)

Ej ítems con dificultad moderada: 2 vs. 6 (distancia 4)

Ítems difíciles: 1 vs. 2 (distancia 1)

2.2.2 Intervención de Subitización y Efecto de Magnitud Numérica

La primera vez que el niño juegue se realizará antes de la intervención de cada tarea un entrenamiento de N repeticiones (N=número de numerosidades que usan en los estímulos).

Tarea:

Estimación de Cantidades

Descripción:

Aparecen cuadraditos con números que cuando caen hay que dar clic sobre ellos la cantidad de veces que indica su magnitud, además desde el comienzo estará el adversario, que en este caso es un pulpo, listo para competir en cuanto se produzca un error.

La dificultad de esta tarea depende del tiempo de presentación de los estímulos y de la cantidad de objetos a estimar.

Procesos:

- Subitización: presentación de cantidades de objetos comprendidas entre 1 y 4.
- Conteo: presentación de cantidades de objetos comprendidas entre 5 y 9.

Ejemplo:

La complejidad de la tarea aumenta al transitar de magnitudes pequeñas (son más fáciles) a grandes y de código no simbólico a números.

2.3 Información que se maneja

1. Registro de usuario

Se recogen los datos característicos de cada usuario (niño).

2. Registro de Estímulos

Número de estímulos que se le pone al niño.

3. Registro de Numerosidad

Cantidad de objetos que contiene cada estímulo.

4. Registro de Respuesta

Recoge la respuesta que da al responder cada estímulo.

5. Fallos y Aciertos

Se controla la cantidad de fallos así como las respuestas correctas de cada estímulo.

6. Registro de Tiempo Límite

Recoge si respondió correctamente en el tiempo establecido.

7. Registro de Tiempo de Acción

Si no respondió correctamente recoge el tiempo que demoró en responder.

2.4 Descripción de la solución

Atendiendo a los problemas que existen con el poco uso de medios para el trato de esta enfermedad, el Centro de Neurociencias de Cuba propone una nueva solución que ha sido pensada y desarrollada para generar nuevos ambientes de aprendizaje que incorporen significativamente las tecnologías de información y comunicación en los procesos de enseñanza - aprendizaje que forman parte del proceso de formación escolar.

Este trabajo se centra en desarrollar los ambientes gráficos para la solución propuesta por los especialistas. Como aporte innovador se propone incorporar las tecnologías informáticas y de educación a los contextos del aula, con un alto sentido pedagógico e investigativo. Su acción directa se centra en los escenarios de enseñanza- aprendizaje, constituido en esencia por el profesor y sus alumnos, lo que a su vez involucra de manera responsable a los directivos, gestores del proceso educativo.

La razón por la que se escogió la solución computarizada es que estas son motivadoras y entretenidas para un entrenamiento intenso en una tarea positiva. Es fácil mantener la atención en una computadora y ellas pueden proporcionar una retroalimentación positiva. Estas pueden presentar una interfaz atractiva e interesante a la vista de los niños y pueden ser diseñadas para su uso con una mínima supervisión de los adultos. Además, los juegos adoptivos para el entrenamiento de la distinción fonética demostraron ser altamente exitosos para el tratamiento de la dislexia.

2.5 Generalidades del Sistema

2.5.1 Uso de la herramienta Scene Toolkit

En la aplicación que se desarrollará se pretende usar la herramienta Scene Toolkit desarrollada por el Proyecto de Herramientas de Desarrollo para Sistemas de Realidad Virtual de la Facultad 5 de la Universidad de Ciencias Informáticas de Cuba. Haciendo uso de esta herramienta estándar se garantiza la política y estrategia trazada por la facultad aprovechando todas las ventajas y facilidades de desarrollo que presenta, y contando con el personal capacitado para el trabajo. Esta herramienta

ofrece soberanía tecnológica ya que al ser creada por los programadores de la facultad se dispone de ella en cualquier momento, evitando toda la política de globalización que existe también en cuanto a las licencias de los programas y derechos de autor.

2.5.2 Características del lenguaje

Para este software o juego se utilizará como IDE (Integrated Development Environment) la herramienta Visual Studio 2003 para Windows ya que este es el sistema operativo que tendrán las máquinas en donde se instalará el juego. Destacar que para este software no se utiliza el framework que trae el Visual Studio ya que este será nada más utilizado como compilador. Por lo que el lenguaje que se usará para su realización es el C++ que es el que mejor se domina. La razón por la que se tomó la decisión de emplear el C++ fue porque como es conocido, este lenguaje, a pesar de su complejidad, es muy potente y veloz, ya que presenta funciones a bajo nivel, permitiendo que se acelere el procesamiento. Además, tiene muy bien desarrollado el paradigma de la programación orientada a objetos (POO). Se trabajará con funciones del C++ estándar, lo que permitirá migrar a otros compiladores para su posterior comercialización, logrando compatibilidad con el software libre, en caso de que se quiera cambiar de Sistema Operativo planificando una futura migración a Linux, por lo que el juego no tendrá ninguna modificación.

2.5.3 Generación de Entornos Dinámicos

Las razones por la que se escogió trabajar con la generación de entornos dinámicos son tres fundamentales:

El entorno no tiene un final bien definido ya que el algoritmo adaptativo es el que determina cuándo debe terminar el entorno, o sea termina cuando el niño cumple con los objetivos del nivel.

La generación permite hacer uso de la reutilización, algo sumamente importante que permite el ahorro de recursos y una variedad bastante grande de entornos donde se puede desarrollar los niveles. (Requisitos no funcionales)

Permite crear una gama de entornos con un mínimo de entornos diseñados (ahorra el tiempo de desarrollo dado que el diseño es bastante poco).

2.5.4 Juego en 1ra Persona

Se propone que sea un juego en primera persona. Uno de los objetivos de este software es que los niños una vez que estén interactuando con dicho juego se centren en las tareas y el contenido de este. Desde el punto de vista de recursos esta opción contribuye a que se consuma la menor cantidad de recursos del Sistema Operativo donde se instalará el juego, haciendo que sea más óptimo ya que en primera persona al desarrollar la parte gráfica se utiliza menor cantidad de polígonos, texturas, se eliminan las animaciones del personaje lo que hace que los entornos estén menos cargados y de esta forma se ahorran recursos de visualización. Estos aspectos hacen que el juego sea más manipulable y permiten que se ajuste a los requerimientos mínimos exigidos para este.

2.5.5 Comparación con otros sistemas

A la hora de realizar la comparación con otros sistemas o software que se han hecho para este padecimiento, se decide tomar como punto de referencia al juego The Number Race debido a que es uno de los pocos que se ha realizado en el mundo con el fin de dar tratamiento o intervención a la discalculia, además de que es conocido por el equipo de trabajo.

Number Race presenta las siguientes características:

- Tiene 1 tarea.
- Consta de 14 niveles.

- Está hecho en 2D (dos dimensiones).
- Posee audio: música e instrucciones a lo largo del juego.
- Está destinado a los niños a partir de la edad de 3 años.

El caso de la solución que se propone se pretende alcanzar los siguientes aspectos:

- Tiene 4 tareas.
- Consta de 6 niveles.
- Está hecho con la combinación de técnicas 2D con 3D (tres dimensiones).
- Usa la generación de entornos dinámicos.
- Posee animaciones.
- Este juego está destinado a los niños de 6to grado.

Como se ha podido observar cada juego tiene sus particularidades, aunque ambos presentan los mismos objetivos a pesar de la diferencia del rango de edades que trabajan. Pero desde el punto de vista de la interfaz y de la parte gráfica, el juego que está en desarrollo tendrá escenarios más llamativos en 3D y combinaciones con 2D que posibilitan mejores gráficos lo que puede resultarle más atractivo a los niños unido al uso de la generación de entornos dinámicos brindando una mayor dinámica al juego influyendo en la motivación del niño, incluye 3 tareas más y un mayor número de ejercicios por lo que se espera un resultado positivo a la hora de su aplicación.

2.6 Reglas del Negocio

Para el nivel 1

1. Para la comunicación con los entes aliados y también para resolver algunas tareas se necesita energía.
2. El héroe deberá vagar por el anillo en una pequeña nave recogiendo cargas energéticas.
3. Las cargas más viejas tienen más puntos luminosos que las más jóvenes.
4. Hay un límite de tiempo para la navegación en el anillo.

5. Si se recogen las cargas incorrectamente esto se cuenta como desacierto y al llegar al 15 % entrará en el anillo el guardián y comenzará a robar de una en una las cargas.

Para el nivel 2

6. Aparecerán cajas con números y deben hacer clic sobre ellas las veces que indica el número que tiene cada una de ellas.

7. Algunas cajas no tendrán ninguna señal, pero es posible saber a través de telepatía el número de estrellas en su interior al acumular la energía suficiente liberando a las otras estrellas.

8. Si no se realiza la tarea de forma correcta esto se contará como desacierto y el pulpo que es el oponente de este nivel resuelve la tarea.

2.7 Modelo del Dominio

Teniendo en cuenta que en el módulo de las tareas a desarrollar no se logra determinar el proceso del negocio con fronteras bien establecidas, entonces se realizará el modelo del dominio identificando los eventos que suceden en el entorno en el que trabaja el sistema que permitirán mostrar al usuario los principales conceptos que se manejan en el sistema en desarrollo.

La realización de este modelo del dominio ayuda a todas las personas involucradas en el proyecto a entender vocablos y términos, a capturar de forma correcta los requisitos, y a agrupar todos los conceptos necesarios que permiten identificar algunas clases que formarán parte del diseño final del módulo.

2.7.1 Glosario de Términos

El primer paso es identificar todos los conceptos que se manejan en el modelo del dominio facilitando la familiarización con los términos utilizados.

Algoritmo Adaptativo: Conjunto de operaciones y cálculos que siguiendo una metodología establecen cuándo ha llegado el final del juego y la complejidad de las tareas a realizar.

Camino: Ruta que sigue el jugador.

Colisión entre Objetos: Choque o encuentro de dos o más entes.

Dinámica de Movimiento: Conjunto de movimientos que siguen determinadas reglas.

Entorno Dinámico: Ambiente o entorno del juego que se construye en tiempo real.

Jugador: Ente o personaje que representa al niño.

Tareas: Operaciones que deben ser realizadas por el jugador y que son determinadas por el algoritmo adaptativo.

Oponente: Ente o personaje que interactúa con el jugador en forma de reto.

Aliado: Ente o personaje que interactúa con el jugador ofreciéndole ayuda.

Puntuación: Acumulación de puntos que se adquieren al realizar satisfactoriamente las tareas.

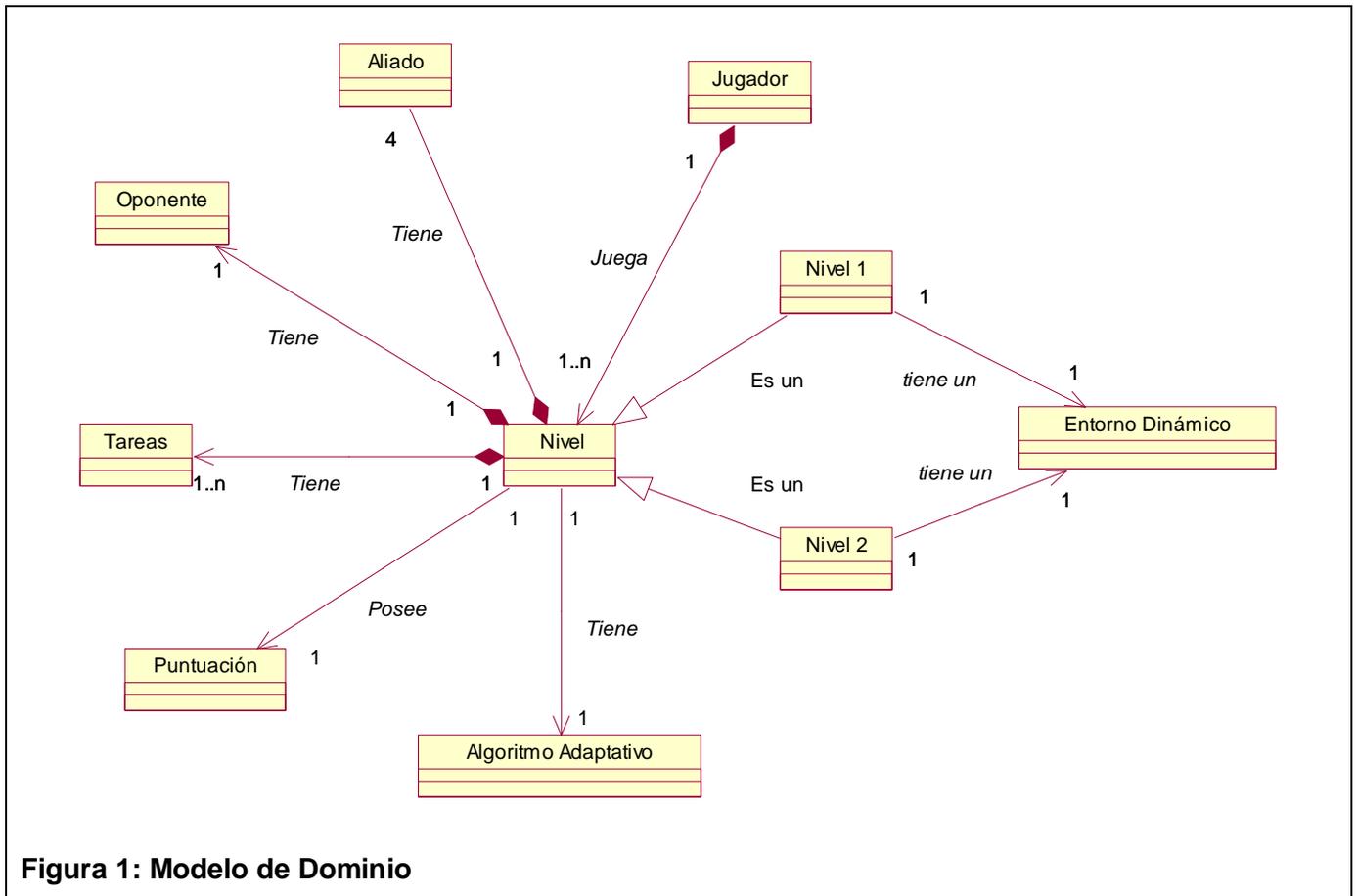
Planeta: Entorno con características particulares donde existen oponentes y aliados y además deben realizarse tareas específicas.

Nivel: Planeta donde hay que alcanzar una puntuación realizando las tareas.

Nivel 1: Planeta Corredoris.

Nivel 2: Planeta Oceanía.

2.7.2 Diagrama del Modelo de Dominio



2.8 Especificación de los requisitos del software

Conociendo los conceptos ya planteados, se puede pasar a lo que el sistema debe hacer para dar cumplimiento a los objetivos planteados. Y es en este flujo de trabajo donde se enumerarán a través de requerimientos funcionales las funciones que el sistema debe ser capaz de realizar. Incluyendo las acciones que el usuario debe ejecutar como las que debe realizar el sistema internamente. En este

epígrafe solo se mencionan los requerimientos que se corresponden con el trabajo. El sistema debe ser capaz de:

2.8.1 Requisitos Funcionales

RF 1- Manipular animación.

- 1.1 – Cargar Tirilla de animación
- 1.2 – Definir Frame Inicial de la animación
- 1.3 – Definir Frame Final de la animación
- 1.4 – Ejecutar Animación.

RF 2 – Mover Jugador

- 2.1 – Mover hacia Delante
- 2.2 – Mover a la Izquierda.
- 2.3 – Mover a la Derecha.
- 2.4 – Mover hacia Atrás.

RF 3 – Generar Entorno

- 3.1 – Crear grafo del entorno.
- 3.2 – Reconfigurar el contenido de diseño.
- 3.3 – Obtener las tareas de Algoritmo Adaptativo

RF 4 – Jugar Básico Nivel 1.

- 4.1 – Jugar Bajo Presión Nivel 1.

RF 5 – Jugar Básico Nivel 2.

RF 6 – Solicitar Ayuda Nivel 1.

RF 7 – Solicitar Ayuda Nivel 2.

RF8 – Mostrar el estímulo.

RF 9 – Verificar vencimiento del tiempo para el estímulo.

9.1 – Se va verificando constantemente si vence el tiempo en que va a estar mostrándose el estímulo.

9.2- Si se vence se retira el mismo.

RF 10 – Retirar el estímulo.

RF 11 – Registrar el resultado. (Acierto o No)

11.1 – El niño da respuesta a un estímulo.

11.2 – Se guarda si la respuesta fue correcta o no.

RF 12 – Tomar Tarea

12.1 – Tomar tarea (el jugador).

12.2 – Tomar tarea (oponente).

RF 13 – Mover oponente en cualquier dirección.

RF 14 – Cargar nivel.

2.8.2 Requisitos no Funcionales

Usabilidad: Los futuros usuarios del sistema serán niños con problemas neurológicos. El producto debe ser atractivo para ellos ya que deben utilizarlo como hobby sin el conocimiento de que están siendo tratados a través del juego.

Soporte: En una versión inicial deberá ser compatible con la plataforma Windows, pero debe estar preparado para que con rápidas modificaciones pueda migrar hacia Linux.

Software: Sistema Operativo Windows.

Hardware: Microprocesador PENTIUM 4. Más de 2 GHz de microprocesador, memoria RAM de 224 Mb, resolución de 800x600.

Diseño e implementación: Se programará en C++ y se regirá por la filosofía de la Programación Orientada a Objetos.

2.9 Modelo de Casos de Usos del Sistema

En este epígrafe se brindará una explicación precisa sobre los principales casos de uso del sistema. Con las facilidades que brinda el UML se capturan los requisitos funcionales del sistema para representarlos mediante un diagrama de casos de uso. Para ello se tendría que definir cuáles serían los actores o actor que van a interactuar con el sistema y los casos de uso que van a representar las funcionalidades.

Actor del sistema

Un actor no es parte del sistema, es un rol de un usuario, que puede intercambiar información o ser un recipiente pasivo de información y representa a un ser humano, a un software o a una máquina que interactúa con el sistema. En este caso interactúa un sólo actor que se define a continuación:

Actor	Descripción
Usuario	Interactúa con el sistema, maneja y dirige al personaje principal.

2.9.1 Listado de Casos de Uso

Los casos de uso representan fragmentos de funcionalidad del sistema, en ellos se describe la secuencia determinada de eventos que realiza un actor al interactuar con la aplicación. En este epígrafe se expondrán los casos de uso de los niveles 1 y 2, y los que recogen las funcionalidades principales del Kernel, ya que esos son los que corresponden al trabajo.

Casos de Uso del Kernel

Tabla 1 Descripción CU Manipular Animación

CU-1	Manipular Animación
Actor	Usuario (niño)
Descripción	Se encarga de ejecutar las animaciones de cada objeto en función del estado en que se encuentre.
Referencia	RF 13, RF 14

Tabla 2 Descripción CU Mover Jugador

CU-2	Mover Jugador
Actor	Usuario (niño)
Descripción	El sistema mueve el personaje hacia la dirección indicada por el niño.
Referencia	RF 13, RF 14

Casos de Uso de los Niveles 1 y 2

Tabla 3 Descripción CU Generar Entorno

CU-3	Generar Entorno
Actor	Usuario (niño)
Descripción	Mediante este caso de uso se generan los caminos del juego, se ubican las tareas correspondientes y se configura el contenido de diseño.
Referencia	RF 9, RF 13, RF 14

Tabla 4 Descripción CU Jugar Básico Nivel 1

CU- 4	Jugar Básico Nivel 1
Actor	Usuario (niño)
Descripción	El niño deberá recolectar cargas de modo ordenado para acumular la energía necesaria para la realización de las tareas.
Referencia	RF 7, RF 9, RF 10, RF 11, RF 12,RF 15

Tabla 5 Descripción CU Jugar Bajo Presión Nivel 1

CU-4.1	Jugar Bajo Presión Nivel 1
Actor	Usuario (niño)
Descripción	El niño llega al 15% de desaciertos y tendrá que disputarse las cargas con el guardián que introduce el sistema.
Referencia	RF 7, RF 9, RF 10, RF 11, RF 12

Tabla 6 Descripción CU Jugar Básico Nivel 2

CU-5	Jugar Básico Nivel 2
Actor	Usuario (niño)
Descripción	El niño deberá cumplir determinadas tareas y tendrá al pulpo como oponente haciéndole resistencia.
Referencia	RF 7, RF 9, RF 10, RF 11, RF 12,RF 16

Tabla 7 Descripción CU Solicitar Ayuda Nivel 1

CU-6	Solicitar Ayuda del Nivel 1
Actor	Usuario (niño)
Descripción	El niño solicita ayuda del nivel 1 en una o mas ocasiones.
Referencia	RF 13

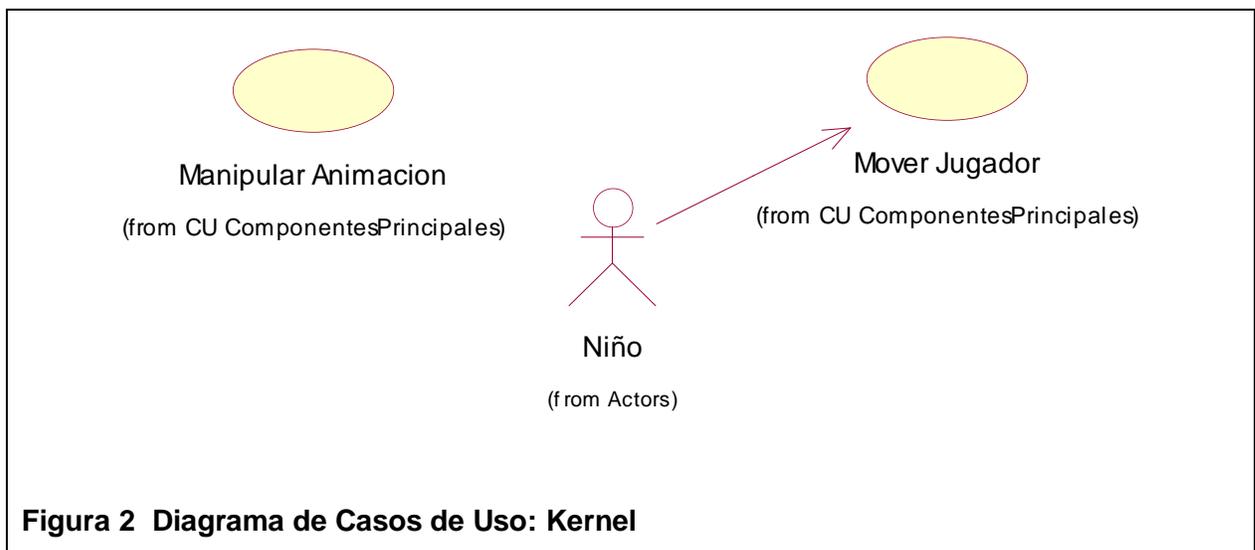
Tabla 8 Descripción CU Solicitar Ayuda Nivel 2

CU-7	Solicitar Ayuda del Nivel 2
Actor	Usuario (niño)
Descripción	El niño solicita ayuda del nivel 2 en una o mas ocasiones.
Referencia	RF 14

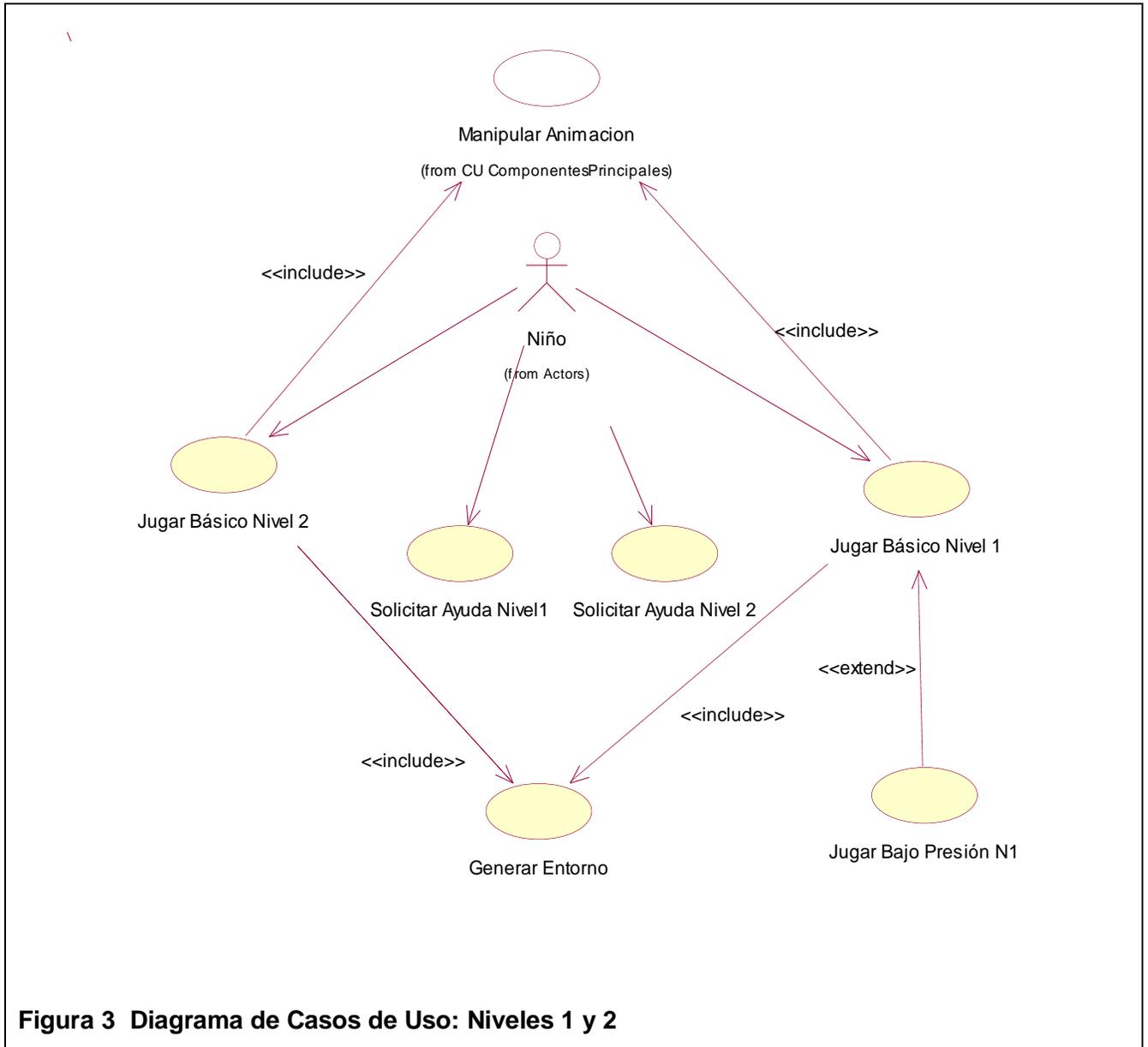
2.9.2 Diagrama Casos de Uso

Teniendo en cuenta los principales procesos que se deben manejar y las dependencias que pueden existir entre ellos, el diagrama de casos de uso del sistema queda dividido en dos diagramas debido a que se tratan módulos que son independientes y de esta forma se hace posible una mayor

comprensión del trabajo. El diagrama de casos de uso del sistema queda confeccionado de la siguiente forma: diagrama de casos de uso del Kernel, donde se agrupan los principales procesos que son comunes para todos los niveles y diagrama de casos de uso de los Niveles 1 y 2, en el que se manejan todos los procesos involucrados en estos dos niveles.



El CU - Manipular Animación es iniciado por los CU- Jugar Básico Nivel 1 y CU- Jugar Básico Nivel 2.



El modelo de casos de uso ayuda a los usuarios y a los desarrolladores a llegar a un acuerdo sobre cómo utilizar el sistema.

El trabajo se va a desarrollar en un único ciclo de desarrollo donde se logrará la realización en los diferentes flujos de trabajo de todos los casos de uso propuestos. Se hará uso de un solo paquete que agrupará toda la información.

2.9.3 Expansión de los Casos de Uso

Mediante los casos de uso expandidos se describe paso a paso la secuencia de eventos que los actores utilizan para completar un proceso usando el sistema.

Casos de Uso del Kernel

Tabla 9 Caso de Uso expandido: Mover Jugador

Caso de Uso:	Mover Jugador	
Actor:	Usuario(Niño)	
Resumen:	El sistema mueve al jugador hacia la dirección indicada por el mismo.	
Precondiciones:	Debe estar cargado uno de los niveles	
Referencias	RF 13, RF 14	
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección “Mover hacia adelante”		
Acción del Actor	Respuesta del Sistema	
1. El niño presiona la tecla de dirección hacia adelante.	1.1 El sistema mueve al jugador hacia adelante aumentando la velocidad de esta mientras el niño mantenga presionada la tecla.	
Sección “Mover hacia atrás”		
Acción del Actor	Respuesta del Sistema	

2. El niño presiona la tecla de dirección hacia atrás.	2.1 El sistema mueve al jugador hacia atrás mientras el niño mantenga presionada la tecla.
Sección “Mover a la izquierda”	
Acción del Actor	Respuesta del Sistema
3. El niño presiona la tecla de dirección hacia la izquierda.	3.1 El sistema mueve al jugador hacia la izquierda mientras el niño mantenga presionada la tecla.
Sección “Mover a la derecha”	
Acción del Actor	Respuesta del Sistema
4. El niño presiona la tecla de dirección hacia la derecha.	4.1 El sistema mueve al jugador hacia la derecha mientras el niño mantenga presionada la tecla.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz	
Poscondiciones	Se mueve el jugador según lo indique el niño.

Tabla 10 Caso de Uso expandido: Manipular Animación

Caso de Uso:	Manipular Animación.
Actores:	Usuario(Niño)
Resumen:	Se encarga de ejecutar las animaciones de cada objeto en función del estado en que se encuentre.
Precondiciones:	Debe estar cargado uno de los niveles
Referencias	RF 13, RF 14
Prioridad	Media
Flujo Normal de Eventos	
Sección “Ejecutar animación”	
Acción del Actor	Respuesta del Sistema

	<ol style="list-style-type: none"> 1. El sistema ubica el frame inicial de la animación. 2. El sistema ubica el frame final de la animación. 3. El sistema ejecuta la animación.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz	
Poscondiciones	Se ejecuta la animación.

Casos de Uso de los Niveles 1 y 2

Tabla 11 Caso de Uso expandido: Jugar Básico Nivel 1

Caso de Uso:	Jugar Básico Nivel 1
Actores:	Usuario(Niño)
Resumen:	El niño podrá acumular la energía necesaria para lo cual debe recolectar cargas de modo ordenado.
Precondiciones:	Que las reservas estén por debajo del 20%
Referencias	RF 7, RF 9, RF 10, RF 11, RF 12,RF 15
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- Solicita Moveirse por el entorno	<ol style="list-style-type: none"> 1.1- El sistema genera la posición del entorno necesaria de acuerdo a la posición del usuario (Se llama al CU Generar Entornos) 1.2 - El sistema permite su movimiento (CU Mover Jugador). 1.3 - El sistema le muestra el conjunto de cargas de acuerdo a la distancia requerida.
2- Recoge las cargas energéticas	2.1 - El sistema registra los aciertos llamando al CU

ordenadas haciendo clic sobre estas en el tiempo designado.	Gestionar AA. 2.2 - Ejecuta sonido "Alegría 2.3 - El Sistema señala las cargas según se valla seleccionando correctamente.
4- El usuario recoge todas las cargas energéticas satisfactoriamente.	5 – El AA determina que finalizó entonces pasa al siguiente nivel y así termina el CU.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
2- No selecciona las cargas correctamente o en el tiempo asignado por el sistema.	2.1- El sistema registra desacierto llamando al CU Gestionar AA. 2.2- Ejecuta sonido "Tristeza 2.3- Si se llegó al límite de fallos el sistema llama al (CU Extendido Jugar Bajo Presión Nivel 1).
Prototipo de Interfaz	
Poscondiciones	El usuario (niño) puede o no haber recolectado la energía necesaria.

Tabla 12 Caso de Uso expandido: Jugar Bajo Presión Nivel 1

Caso de Uso:	Jugar Bajo Presión Nivel 1
Actores:	Usuario(Niño)
Resumen:	El niño llega al 15% de desaciertos y tendrá que disputarse las cargas con el guardián que introduce el sistema.
Precondiciones:	15% de desaciertos.
Referencias	RF 7, RF 9, RF 10, RF 11, RF 12
Prioridad	Opcional
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1-Recoge las cargas haciendo clic en las mismas de manera ordenada pero llega al 15% de desaciertos.	1.1- Introduce un guardián el cual evita que cumpla su objetivo satisfactoriamente. Aparece cada cierto tiempo y se roba una carga del conjunto.

	1.2-Ejecuta sonido “Tristeza”
2-Termina el ordenamiento.	2.1- El sistema contabiliza las cargas acumuladas en la competencia. 2.2 Registra los Resultados en el AA mediante el CU Gestionar AA.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz	
Poscondiciones	El usuario (niño) puede o no haber recolectado la energía necesaria.

Tabla 13 Caso de Uso expandido: Jugar Básico Nivel 2

Caso de Uso:	Jugar Básico Nivel 2
Actores:	Usuario (niño)
Resumen:	El pulpo se mantiene junto al personaje principal durante todo el nivel, haciéndole resistencia para cumplir con las tareas, y en caso de que se termine el tiempo para resolver la tarea o esta no sea resuelta correctamente el pulpo resolverá la tarea y el niño pasará a resolver la tarea siguiente.
Precondiciones:	El usuario se encuentra jugando en el nivel.
Referencias	RF 7, RF 9, RF 10, RF 11, RF 12,RF 16
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1 El usuario solicita moverse en dirección de una determinada tarea.	1.1 El sistema genera el Entorno con sus contenidos. (Se llama al CU Generar Entorno) 1.2 Ejecuta el Caso de Uso Mover Jugador 1.3 El pulpo (sistema) se mueve junto al muchacho en la misma dirección. (Se llama al CU Manipular

	Animación)
2 El usuario entra en el radio de acción de una determinada tarea.	2.1 El sistema muestra la tarea.
3 El usuario selecciona la caja con la cantidad de estrellas a tomar.	3.1 El sistema contabiliza el acierto, llamando al CU Gestionar AA. 3.2 El sistema ejecuta sonido “Alegría”.
	4 Cuando el usuario cumple con todas las tareas seleccionadas por el algoritmo adaptativo, este decide terminar el nivel y finaliza así el caso de uso.
Flujos alternos	
Acción del Actor	Respuesta del Sistema
3 El usuario no logra tomar la tarea en el tiempo calculado por el sistema o la realiza de forma incorrecta.	3.1 El sistema contabiliza desacierto (Llama al CU Gestionar AA). 3.2 El sistema ejecuta sonido de “Tristeza”. 3.3 El pulpo (sistema) soluciona la tarea y le da la oportunidad al usuario de resolver la siguiente tarea a modo de estímulo para el niño.
Poscondiciones	Queda actualizado el resultado del cumplimiento de las tareas.

Tabla 14 Caso de Uso expandido: Generar Entorno

Caso de Uso:	Generar Entorno
Actores:	Usuario (niño)
Resumen:	Este caso de uso es el encargado de generar los caminos del juego, ubicar las tareas correspondientes y de configurar el contenido de diseño.
Precondiciones:	El usuario se encuentra jugando.
Referencias	RF 9, RF 13, RF 14
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

<p>1 El usuario se mueve en una dirección determinada.</p>	<p>1.1 Crea el grafo de entorno, o sea un conjunto de nodos que conforman la estructura del camino durante el transcurso del juego</p> <p>1.2 Reconfigura el contenido de diseño haciendo las rotaciones pertinentes a los objetos de la escena a ambos lados o en uno solo de manera aleatoria.</p> <p>1.3 Solicita y obtiene del AA la secuencia de tareas necesarias para recrear el contenido del juego</p> <p>1.4 Ubica las tareas obtenidas en diferentes posiciones a través del camino, teniendo en cuenta que la distancia entre ellas está dada por la formula:</p> $dtn = dtn - 1 + k$ <p>Donde n representa la tarea y k constante de descanso entre tareas.</p>
<p>Flujos alternos</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p></p>	<p></p>
<p>Poscondiciones</p>	<p>Queda generado el entorno con todo el contenido.</p>

Tabla 15 Caso de Uso expandido: Solicitar Ayuda Nivel 1

<p>Caso de Uso:</p>	<p>Solicitar Ayuda del Nivel 1</p>
<p>Actores:</p>	<p>Usuario(Niño)</p>
<p>Resumen:</p>	<p>El niño esta jugando el nivel 1 y solicita ayuda en una o más ocasiones.</p>
<p>Precondiciones:</p>	<p>El usuario cuenta con la suficiente energía para solicitar la ayuda.</p>
<p>Referencias</p>	<p>RF 13</p>
<p>Prioridad</p>	<p>Media</p>
<p>Flujo Normal de Eventos</p>	
<p>Sección “Solicitar Nivel de Ayuda del Nivel 1”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>

<p>2. El niño solicita el nivel de ayuda del nivel 1.</p>	<p>1. El sistema verifica si el nivel de energía está acorde para la solicitud de ayuda. 2.1 El sistema analiza el valor de n y este se incrementa en 1. 2.2 Se ejecuta el nivel de ayuda según el valor de n.</p>
<p>Sección “Repetir Nivel de Ayuda”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>1. El niño solicita la repetición de un nivel de ayuda del nivel 1.</p>	<p>2. El sistema repite la ayuda según el valor de n. 3. El sistema ejecuta el nivel de ayuda del nivel 1 según el valor de n. 3.1 Si n = 1 se ejecuta el nivel 1 de la ayuda escuchándose:” Recuerda que debes recoger las cargas en un orden específico. En este caso(decir el orden solicitado en la tarea)” 3.2 Si n = 2 se ejecuta el nivel 2 de la ayuda escuchándose:”Puedes tratar de contar en voz baja o mentalmente para saber qué número sigue o antecede a otro. ¡Prueba! ” 3.3 Si n = 3 se ejecuta el nivel 3 de la ayuda escuchándose:”Vamos a hacerlo juntos: yo te muestro los números y tú terminas de ordenarlos” 3.4 Si n = 4 se ejecuta el nivel 4 de la ayuda escuchándose:”Te voy a mostrar cómo se ordenan estas cargas (decir el orden solicitado en la tarea) Fíjate bien”</p>
<p><i>Prototipo de Interfaz</i></p>	
<p>Flujos Alternos</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p> </p>	<p> </p>

Prototipo de Interfaz	
Poscondiciones	El valor de n para el nivel de ayuda está en cero.

Tabla 16 Caso de Uso expandido: Solicitar Ayuda Nivel 2

Caso de Uso:	Solicitar Ayuda del Nivel 2
Actores:	Usuario(Niño)
Resumen:	El niño esta jugando el nivel 2 y solicita ayuda en una o más ocasiones.
Precondiciones:	El usuario cuenta con la suficiente energía para solicitar la ayuda.
Referencias	RF 14
Prioridad	Media
Flujo Normal de Eventos	
Sección “Solicitar Nivel de Ayuda”	
Acción del Actor	Respuesta del Sistema
2. El niño solicita el nivel de ayuda del nivel 2.	<p>1. El sistema verifica si el nivel de energía está acorde para la solicitud de ayuda.</p> <p>2.1 El sistema analiza el valor de n y este se incrementa en 1.</p> <p>2.2 Se ejecuta el nivel de ayuda según el valor de n.</p>
Sección “Repetir Nivel de Ayuda”	
Acción del Actor	Respuesta del Sistema
1. El niño solicita la repetición de un nivel de ayuda del nivel 2.	<p>2. El sistema repite la ayuda según el valor de n.</p> <p>3. El sistema ejecuta el nivel de ayuda del nivel 2 según el valor de n.</p> <p>3.1 Si $n = 1$ se ejecuta el nivel 1 de la ayuda escuchándose:” Recuerda que debes hacer clic sobre la caja tantas veces como estrellas hay que liberar”</p> <p>3.2 Si $n = 2$ se ejecuta el nivel 2 de la ayuda</p>

<p>escuchándose:” Puedes tratar de contar en voz baja cada vez que hagas clic para que no te equivoques ¡Prueba! ”</p> <p>3.3 Si $n = 3$ se ejecuta el nivel 3 de la ayuda escuchándose:” "Vamos a hacerlo juntos: el número de estrellas que hay es el número que va detrás de " (decir el número arábigo sucesor de la respuesta correcta) " y antes de " (decir el número arábigo al antecesor de la respuesta correcta)."</p> <p>3.4 Si $n = 4$ se ejecuta el nivel 4 de la ayuda escuchándose:” Te voy a mostrar cómo se liberan las estrellas."</p>	
Prototipo de Interfaz	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz	
Poscondiciones	El valor de n para el nivel de ayuda está en cero.

En este capítulo se hizo un esbozo detallado de la propuesta de solución que brinda este trabajo para cumplir con los objetivos planteados. Se ha analizado cada una de las partes que tiene el juego. Se enumeraron los requisitos funcionales y los no funcionales, además de que se explicó de manera muy detallada cada uno de los casos de uso. También se mostró el diagrama de casos de uso que da una visión más clara de cómo funcionarán los mismos. Una parte clave dentro del capítulo es la comparación del sistema propuesto con uno de los existentes en el mundo en la actualidad, que ayuda a tener una idea más clara de qué es lo que se está proponiendo y de cómo será respecto a lo que ya existe.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción

En la elaboración de este capítulo se define la estructura del sistema en estudio, siempre basándose en los requisitos funcionales y no funcionales que fueron seleccionados en las etapas anteriores. Se presentan los artefactos involucrados en el diseño, que serían Diagramas de clases y Diagramas de secuencia de la realización de los casos de uso. Se trató de realizar un diseño entendible y que le brinde al usuario una interfaz amigable. Además, se tuvo muy en cuenta que en este trabajo solo se expondrán dos de los niveles del juego, incluyendo las funcionalidades principales del Kernel, que es un módulo a parte. Finalmente se tratan los principios del diseño de la aplicación.

3.2 Diagrama de Paquetes de Clases de Diseño

En este epígrafe se hace referencia a los diagramas de clases que componen el sistema. Lo que constituye uno de los elementos más importantes dentro de un proyecto de software. Mostrando una visión completa del sistema así como las clases con sus métodos, atributos y las relaciones entre estas. Un software se puede dividir por paquetes, ese es el caso del sistema que se propone en este trabajo.

Las clases utilizadas en el módulo están organizadas por paquetes distinguiendo las responsabilidades de cada una de ellas. El sistema propuesto consta de tres paquetes, el Level, el Kernel y el STK (Scene Toolkit). El paquete Level es el que contiene todas las clases específicas de cada nivel. El paquete del Kernel se refiere a todo lo que es común para todos los niveles que tiene y tendrá el juego, es el núcleo y en él se encuentran las clases que son utilizadas de forma general en todos los niveles. El paquete de STK se refiere a todo lo relacionado con la herramienta con la que se trabaja que es la utilizada en el desarrollo de juegos y simuladores, es el motor gráfico desarrollado en la facultad 5 de la Universidad de Ciencias Informática.

3.2.1 Diagrama de clases del paquete Level

El paquete Level es donde se encuentran las clases que tienen que ver específicamente con los 7 niveles del juego, pero este trabajo solo abarca los niveles 1 y 2 en los que se realizan todas las operaciones que hacen que el sistema cumpla con los objetivos planteados.

En la Figura 1 del Anexo 1 se puede ver el Diagrama de clases del paquete Level.

3.2.2 Diagrama de clases del paquete Kernel

Este paquete agrupa las funcionalidades del juego y contiene los elementos principales del sistema, se puede decir que es el paquete fundamental dentro del mismo, aunque no tendría razón de ser sin los demás paquetes.

Está constituido por varias clases, agrupadas en tres paquetes que son: Algoritmo Adaptativo, que agrupa las clases que determinan el trabajo con las tareas, los estímulos y las que registran la información; Presentación, en el que se encuentran las clases relacionadas con la interfaz y Main Component, que contiene las clases principales para el manejo de los niveles.

Este paquete es prácticamente el motor que ejecuta la mayoría de las funcionalidades del sistema y destacar que guarda relación con el paquete STK. En la Figura 2 del Anexo 1 se puede ver el Diagrama de clases del paquete Kernel.

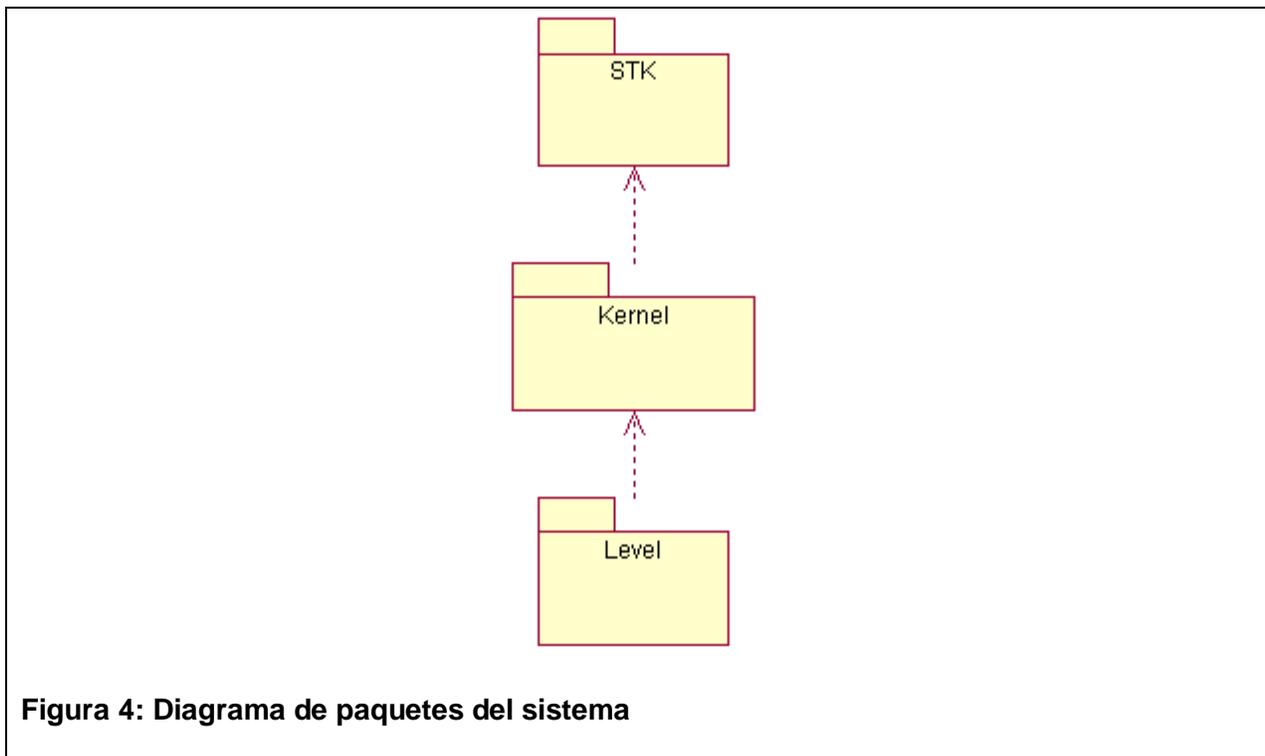
Se decide realizar un diagrama adicional a partir de la clase COponente para un mayor entendimiento de todo lo relacionado con la misma, ya que es un componente fundamental del sistema. En la figura 3 del Anexo 1 se puede observar dicho diagrama.

3.2.3 Diagrama de clases del paquete STK

Este paquete es el que corresponde al engine que se utiliza y que facilita el trabajo con los gráficos. Está constituido por numerosas clases pero solo se utilizan las que se relacionan con el Kernel. En la Figura 4 del Anexo 1 se puede ver el Diagrama de clases del paquete STK.

3.3 Diagrama de Paquetes

Aquí se muestra la relación entre todos los paquetes antes expuestos. Comprender esta relación es importante ya que se puede ver de forma general cómo funciona todo el sistema. El paquete principal es el Kernel que es el que se relaciona con el resto de los paquetes, y desde él se manejan todos los procesos del mismo, aunque el paquete Level es donde se llevan a cabo las funcionalidades del trabajo.



3.4 Descripción de las clases

A continuación se dará una descripción detallada de cada clase del sistema, incluyendo sus atributos y métodos, de forma que se logre una mayor comprensión del mismo.

Paquete Level

Nombre: CLevel1	
Atributo:	Tipo:
1. m_fLimitTime	float
Para cada operación:	
Nombre:	Descripción:
PlayLevel	Inicia el juego.
GetSelectedTask	Devuelve información de la tarea seleccionada.
TakeTask	Toma la tarea.
LoadLevel	Carga el nivel.
UpdateTimer	Actualiza el tiempo que demora la realización de la tarea.
GetTask	Proporciona la tarea.
bLimitWrongTasks	Devuelve true cuando el player llega al 15% de desacierto.
CheckStayTask	Verifica el orden de los elementos en la Línea Numérica.
pkCreateTask	Crea una tarea.
ActivateTask	Se encarga de Analizar cuándo mostrar el estímulo esta es redefinida en cada nivel ya que la manera y el momento de mostrar la tarea es diferente.
Animate TakeTask	Proporciona el efecto de animación al tomar la tarea.
GetEnergy	Guarda la energía acumulada por el personaje principal.

Nombre: CProtectman

Atributo:	Tipo:
1. m_aiTasktaken	int*
Para cada operación:	
Nombre:	Descripción:
Compete	Da inicio a la intervención del guardián en el juego.
GeneralMove	Proporciona el movimiento general del guardián.
Animate	Proporciona el efecto de animación de cada acción del guardián.

Nombre: CLevel2	
Atributo:	Tipo:
1. m_fLimitTime	float
Para cada operación:	
Nombre:	Descripción:
PlayLevel	Inicia el juego.
GetSelectedTask	Devuelve información de la tarea seleccionada.
TakeTask	Toma la tarea.
LoadLevel	Carga el nivel.
UpdateTimer	Actualiza el tiempo que demora la realización de la tarea.
GetTask	Proporciona la tarea.
CheckStayTask	Verifica el número de clic así como el tiempo entre los clics.
pkCreateTask	Crea una tarea.
ActivateTask	Se encarga de Analizar cuándo mostrar el estímulo, esta es redefinida en cada nivel ya que la manera y el momento de mostrar la

	tarea es diferente.
Animate TakeTask	Proporciona el efecto de animación al tomar la tarea.
GetCantStars	Devuelve la cantidad de estrellas.

Nombre: Adversario(pulpo)	
Atributo:	Tipo:
1. m_bRandom	bool
Para cada operación:	
Nombre:	Descripción:
Compete	Da inicio a la intervención del pulpo en el juego.
GeneralMove	Proporciona el movimiento general del pulpo.
Animate	Proporciona el efecto de animación de cada acción del pulpo.

Nombre: CEnviromentCreator	
Atributo:	Tipo:
Para cada operación:	
Nombre:	Descripción:
GenerateEnviroment	Genera dinámicamente el entorno.

Nombre: CControlAnimation	
Atributo:	Tipo:
Para cada operación:	
Nombre:	Descripción:
PlayAnimation	Ejecuta la animación correspondiente a la

	acción invocada.
--	------------------

Paquete Kernel

Nombre: CAlly	
Atributo:	Tipo:
1. m_asMessageForLevelHelps	char**
2. m_apkSounds	CSound*
3. m_aokSoundsElement	CSound*
4. m_iCurrLevel	int
Para cada operación:	
Nombre:	Descripción:
HelpLevelInformation	Define y ordena la ejecución de cada uno de los Niveles de Ayuda. En su interior están las Funciones: HelpLevel1 ()... hasta HelpLevel4 (). Todas son redefinidas en las clases Hijas.
HelpLevel1	Consta de Sonidos, "Textos Asociados", etc... que son redefinidos para cada nivel.
HelpLevel2	Consta de Sonidos, "Textos Asociados", etc... que son redefinidos para cada nivel.
HelpLevel3	Consta de Sonidos, "Textos Asociados", etc... que son redefinidos para cada nivel.
HelpLevel4	Consta de Sonidos, "Textos Asociados", etc... que son redefinidos para cada nivel.

Nombre: CComplexTask	
Atributo:	Tipo:
1. m_iMaxLineNumber	int

2. m_iMinLineNumber	int
3. m_iCount	int*
Para cada operación:	
Nombre:	Descripción:
bSelectTask	Selecciona una tarea.
Animation	Proporciona el efecto de animación de cada acción correspondiente a una tarea.

Nombre: CLevel	
Atributo:	Tipo:
1. m_eStayCurrTask	EStayCurrTask
2. m_fTimeCurrTask	float
3. m_ildCurrentTask	int
Para cada operación:	
Nombre:	Descripción:
PlayLevel	Inicia el juego.
GetSelectedTask	Devuelve información de la tarea seleccionada.
TakeTask	Toma la tarea.
LoadLevel	Carga el nivel.
UpdateTimer	Actualiza el tiempo que demora la realización de la tarea.
GetTask	Proporciona la tarea.
bLimitWrongTasks	Devuelve true cuando el player llega al 15% de desacierto.
CheckStayTask	Actualiza el estado de selección de la tarea que está realizando ahora. NO_DID si la tarea o aun no ha sido realizada CORRECT si la completó correctamente. INCORRECT si se equivocó.

pkCreateTask	Crea una tarea.
ActivateTask	Se encarga de Analizar cuándo mostrar el estímulo, esta es redefinida en cada nivel ya que la manera y el momento de mostrar la tarea es diferente.
Animate TakeTask	Proporciona el efecto de animación al tomar la tarea.

Nombre: COponent	
Atributo:	Tipo:
1. int m_iStay	int
Para cada operación:	
Nombre:	Descripción:
Compete	Da inicio a la intervención del oponente en el juego.
GeneralMove	Proporciona el movimiento general del oponente.
Animate	Proporciona el efecto de animación de cada acción del oponente.

Nombre: CPlayer	
Atributo:	Tipo:
1. m_fScore	int
2. m_aiEnergy	int *
3. m_bForward	bool
4. m_bLookRight	bool
5. m_bLookLeft	bool
6. m_bLBackward	bool
7. m_eStayAnimate	ETypeStayAnimate

Para cada operación:	
Nombre:	Descripción:
PlayLevel	Inicia el juego.
AddPoints	Adiciona los puntos acumulados por el personaje principal.
Animate	Proporciona el efecto de animación de cada acción del personaje principal.
MovePlayer	Proporciona movimiento al personaje principal.
bBackward	Proporciona el movimiento del personaje principal hacia atrás.
bLookLeft	Proporciona el movimiento del personaje principal hacia la izquierda.
bLookRight	Proporciona el movimiento del personaje principal hacia la derecha.
bForward	Proporciona el movimiento del personaje principal hacia adelante.
SelectElement	Esta función llama a bCheckLevelTask para verificar y actualizar correctamente la recogida de las tareas

Nombre: CSingleTask	
Atributo:	Tipo:
1. m_iCount	int
Para cada operación:	
Nombre:	Descripción:
bSelectTask	Selecciona una tarea.
Animation	Proporciona el efecto de animación de cada acción correspondiente a una tarea.

Nombre: CTask	
Atributo:	Tipo:
1. m_bSelectTask	bool
2. m_fTime	float
3. m_eEstimateTask	EEstimateType
4. m_ild	int
5. m_eStayTask	EAnimStayTask
Para cada operación:	
Nombre:	Descripción:
bSelectTask	Selecciona una tarea.
Animation	Proporciona el efecto de animación de cada acción correspondiente a una tarea.

Nombre: STaskDate	
Atributo:	Tipo:
1. m_bRight	bool
2. m_fTime	float
3. m_iCount	int
Para cada operación:	
Nombre:	Descripción:

En el caso de las demás clases no es necesario realizar una tabla como las anteriores para su explicación, ya que estas son componentes utilizados por el sistema.

Paquete Level

CEnvironmentCreator: Esta clase se reutiliza, ya que se tomó hecha.

CSubEnvironment: Se utiliza para la generación de entornos.

Paquete STK

CApplication: Es la clase interfaz de la herramienta STK, de la que se heredó para hacer el juego.

CNeuroApplication: Es la clase específica del juego.

CCameraNode: Es la cámara del juego, la visión del software.

CNode: Es la entidad de la estructura que contiene la información geométrica de cualquier componente del juego y todas las transformaciones gráficas.

CRender: Es la clase encargada de dibujar. Permite dar funcionalidad al objeto que se selecciona a través de un clic en la pantalla.

3.5 Diagramas de secuencia

Los diagramas de secuencia muestran un modelo de cómo interactúa el usuario con el sistema y de cómo interactúan los objetos del sistema entre sí. A continuación se explicará el flujo de sucesos de los diagramas de secuencia de los casos de uso.

3.5.1 Diagrama Principal

Flujo de sucesos

Este diagrama de secuencia se puede ver en la Figura 1 del anexo 2.

Se decide hacer este diagrama de secuencia debido a que es común para varios casos de uso, es donde inician los diagramas de secuencia de los casos de uso: Mover Jugador, Jugar Básico Nivel 1 y Jugar Básico Nivel 2.

Primeramente se actualizan los estados de movimiento a partir de las teclas de dirección, para esto el usuario tuvo que apretar la tecla correspondiente al sentido en que quería mover al personaje, hasta aquí es el alcance del caso de uso Mover Jugador.

Seguidamente se actualizan los estados de selección de tareas de la función Pick () perteneciente al Render que permite seleccionar un elemento 3D mediante un clic, informando al sistema la geometría del entorno que fue seleccionada. El sistema realiza el ciclo principal encargado de llamar a la función MovePlayer () a través de la que se accede a la función PlayLevel () que realiza toda la lógica interna específica del nivel.

3.5.2 Jugar Básico Nivel 1

Flujo de sucesos

Este diagrama de secuencia se puede ver en la Figura 2 del anexo 2, en él se observa cómo se lleva a cabo el caso de uso Jugar Básico Nivel 1.

Primeramente se actualizan los estados de movimiento y de selección de tareas, proceso que se puede ver en el diagrama de secuencia Diagrama Principal, se crea el entorno y se toma la tarea, proceso referenciado en el diagrama de secuencia Tomar Tarea, donde se gestiona todo lo que tiene que ver con el Algoritmo Adaptativo, se registran los datos de tiempo y puntuación así como cualquier información de la tarea en cuestión. Si la tarea es realizada de forma incorrecta se marca como desacierto y en caso contrario se marca como acierto y se analiza el por ciento de desaciertos para ejecutar o no el caso de uso Jugar Bajo Presión Nivel 1.

3.5.3 Jugar Bajo Presión Nivel 1

Flujo de sucesos

Este caso de uso se puede ver en el diagrama de secuencia del caso de uso Jugar Básico Nivel 1 que se observa en la Figura 2 del anexo 2.

Este caso de uso tiene lugar dentro del caso de uso Jugar Básico Nivel 1 a partir de que el jugador acumula un 15 por ciento de desaciertos mientras realiza una tarea, en ese momento se llama a la función Compete () y el guardián comienza a competir y se ejecuta el efecto de animación correspondiente a su estado.

3.5.4 Jugar Básico Nivel 2

Flujo de sucesos

Este diagrama de secuencia se puede ver en la Figura 3 del anexo 2, en él se observa cómo se lleva a cabo el caso de uso Jugar Básico Nivel 2.

Primeramente se actualizan los estados de movimiento y de selección de tareas, proceso que se puede ver en el diagrama de secuencia Diagrama Principal, se crea el entorno, se le da la orden al pulpo de comenzar a moverse ejecutándose el efecto de animación correspondiente y se toma la tarea, proceso referenciado en el diagrama de secuencia Tomar Tarea donde se gestiona todo lo que tiene que ver con el Algoritmo Adaptativo, se registran los datos de tiempo y puntuación así como cualquier información de la tarea en cuestión. Si la tarea es realizada de forma correcta se marca como acierto y en caso contrario se marca como desacierto, entonces se llama a la función Compete () y el pulpo comienza a competir y se ejecuta el efecto de animación correspondiente.

3.5.5 Tomar Tarea

Flujo de sucesos

Este diagrama de secuencia se puede ver en la Figura 4 del anexo 2.

Primeramente se activa la tarea y si está disponible se actualiza el tiempo. A continuación se comprueba si el estado de la tarea es correcto, en ese caso se registra como acierto, se guarda la puntuación y tiempo de la tarea en una estructura y además se le adiciona la puntuación al jugador. En

caso de no ser correcta la tarea se registra como desacierto, la puntuación se pone en cero, el tiempo se toma como el máximo y estos datos se guardan en una estructura. Después se registra la información en el AA y se crea la próxima tarea que es definida por el AA.

3.5.6 Manipular Animación

Flujo de sucesos

Este diagrama de secuencia se puede ver en la Figura 5 del anexo 2, en él se observa cómo se lleva a cabo el caso de uso Manipular Animación.

Según sea el estado en que se encuentre cualquier elemento del entorno ya sea personaje, tarea, oponente u otro, el sistema carga la tirilla de animación correspondiente, define frame inicial y final, chequea si la animación es cíclica o no y la ejecuta. En caso de ser cíclica se ejecuta del frame inicial al final y después se vuelve a mover al inicial, en caso contrario se ejecuta del frame inicial al final.

3.5.7 Otros casos de uso

A continuación se describirán los flujos de sucesos de los casos de uso: Solicitar Ayuda Nivel 1 y Solicitar Ayuda Nivel 2. A estos casos de uso no se les realizó el diagrama de secuencia debido a la gran sencillez de los mismos, aunque por sencillos no son menos importantes, ya que ambos ejecutan acciones fundamentales que ayudan en gran medida a que el sistema cumpla con los objetivos para los que fue diseñado.

3.5.7.1 Solicitar Ayuda Nivel 1

Flujo de sucesos

Primeramente el sistema verifica si el nivel de energía del personaje es suficiente para la solicitud de ayuda. A continuación de que el niño solicita el nivel de ayuda del nivel 1 el sistema analiza el valor de n (número del nivel de ayuda), lo incrementa en 1 y ejecuta el nivel de ayuda según su valor.

En caso de que se solicite la repetición de un nivel de ayuda del nivel 1, el sistema lo ejecuta según el valor de n , si $n = 1$ se ejecuta el nivel 1 de la ayuda del nivel 1 incrementándose hasta llegar a 4 que son los niveles de ayuda reglamentados.

3.5.7.2 Solicitar Ayuda Nivel 2

Flujo de sucesos

Primeramente el sistema verifica si el nivel de energía del personaje es suficiente para la solicitud de ayuda. A continuación de que el niño solicita el nivel de ayuda del nivel 2 el sistema analiza el valor de n (número del nivel de ayuda), lo incrementa en 1 y ejecuta el nivel de ayuda según su valor.

En caso de que se solicite la repetición de un nivel de ayuda del nivel 2 el sistema lo ejecuta según el valor de n , si $n = 1$ se ejecuta el nivel 1 de la ayuda del nivel 2 incrementándose hasta llegar a 4 que son los niveles de ayuda reglamentados.

Al concluir este capítulo se tiene el diseño del sistema a desarrollar, ya que se explica de forma detallada, con la ayuda de los diagramas de secuencia, cómo será la interacción entre las clases para el desarrollo de cada uno de los casos de uso. Además, se hizo el diagrama de clases para cada uno de los paquetes del sistema que se corresponden a los objetivos del trabajo, con una descripción de cada una de estas, especificando cuál es la función de cada uno de sus atributos y métodos. Después de concluido el diseño, se puede pasar a la fase de implementación.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA

4.1 Introducción

En este capítulo se define la línea base ejecutable de la arquitectura de la aplicación. En él se encuentran los diagramas de componentes y los casos de prueba. El diagrama de componentes que se presenta, describe los componentes a construir, su organización y dependencia entre los nodos físicos. La etapa de prueba que es uno de los pasos más importantes que se realiza en todo software, y los casos de prueba diseñados en este capítulo permiten verificar el correcto funcionamiento de las principales funciones de la aplicación, de manera que se pueda comprobar si fueron cumplidos los requerimientos del sistema, demostrar que el software funciona y que además se cumplieron los objetivos planteados.

4.2 Implementación

En la implementación se comienza con el resultado del diseño y se implementa el sistema en términos de componentes. La mayor parte de la arquitectura del sistema es capturada durante el diseño, siendo el propósito principal de la implementación el desarrollar la arquitectura y el sistema como un todo. La implementación tiene como propósitos específicos: planificar las integraciones de sistema necesarias en cada iteración, distribuir el sistema asignando componentes ejecutables a nodos en el diagrama de despliegue, implementar las clases y subsistemas encontrados durante el diseño y probar los componentes individualmente, e integrarlos compilándolos. La implementación es el centro durante las iteraciones de construcción, aunque también se lleva a cabo trabajo de implementación durante la fase de elaboración, para crear la línea base ejecutable de la arquitectura, y durante la fase de transición, para tratar defectos tardíos. Los diagramas de despliegue y componentes, que son artefactos generados en este flujo de trabajo conforman lo que se conoce como un modelo de implementación al describir los componentes a construir y su organización y dependencia entre nodos físicos en los que funcionará la aplicación.

4.2.1 Despliegue.

En el caso del software desarrollado no es necesario realizar el diagrama de despliegue ya que solamente se necesita una computadora para llevar a cabo el trabajo.

4.2.2 Diagrama de Componentes.

Los diagramas de componentes describen los elementos del sistema y sus relaciones, y muestran las opciones de realización, incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de las aplicaciones informáticas. Pueden ser archivos cuando representan un documento que contiene código fuente o datos, un documento, bibliotecas de objetos, los ejecutables o tablas de una base de datos.

El sistema está dividido en tres paquetes pero solo se desarrollaron diagramas de componentes de dos de ellos, dentro del paquete Level un diagrama de los dos primeros niveles y dentro del paquete Kernel un diagrama de Main Component, debido a que son los paquetes en que se centra el trabajo. A continuación se muestran los diagramas.

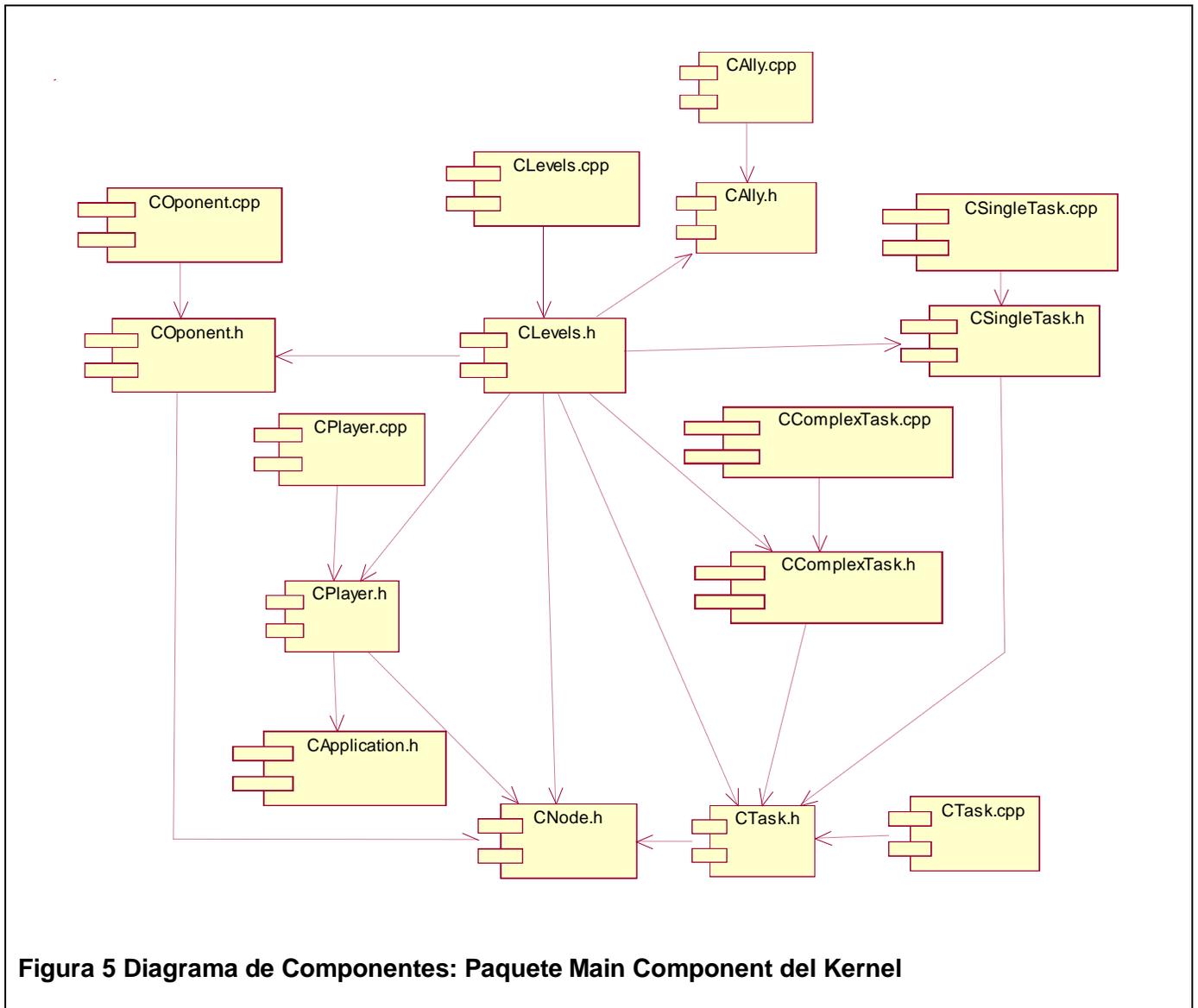
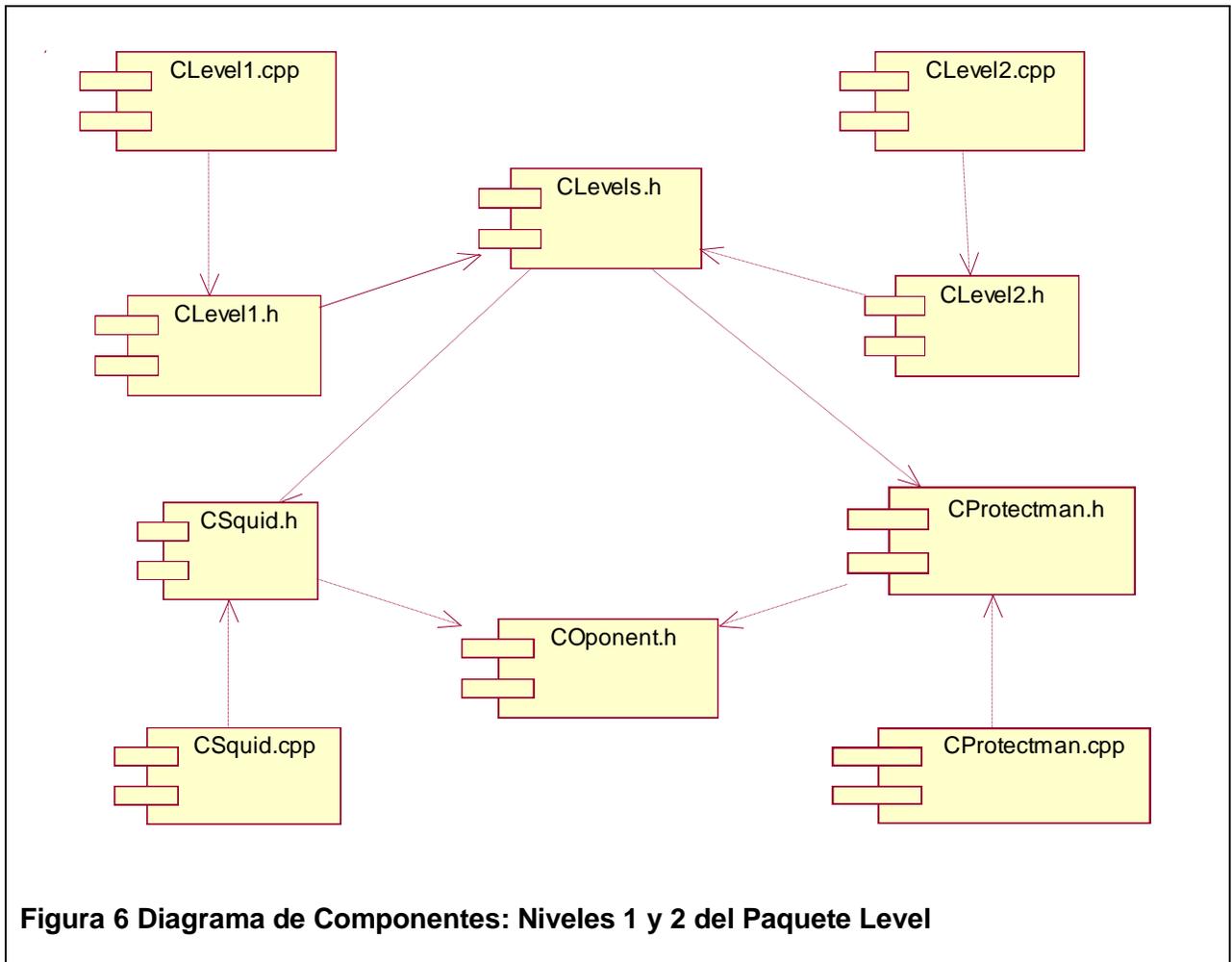


Figura 5 Diagrama de Componentes: Paquete Main Component del Kernel



4.3 Modelo de Prueba

La prueba de software es un elemento crítico para la garantía de la calidad del mismo y representa una revisión final de las especificaciones del diseño y de la codificación. El objetivo de la prueba de software es descubrir errores en la aplicación. Para conseguir este objetivo se planifica y se ejecuta una serie de pasos que van revisando los elementos del software más significativos. En todas las fases del desarrollo del proyecto hay que probar el software que se va construyendo, aunque en la fase de construcción se centran los mayores esfuerzos de este flujo. El modelo de pruebas es una colección de

casos de prueba, procedimientos de prueba y componentes de prueba, que permiten probar los componentes ejecutables en el modelo de implementación.

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados, denominado caso de prueba. Cada caso de prueba especifica una forma de probar el sistema, y son aplicados como pruebas de regresión en cada iteración.

Se pueden realizar muchos casos de prueba para determinar que un caso de uso es completamente satisfactorio. Existen las llamadas pruebas de “caja negra” y las de “caja blanca”. Una prueba de “caja negra” es una prueba del comportamiento observable externamente del sistema, mientras que una prueba de “caja blanca” prueba la interacción interna entre los componentes del sistema.

En este trabajo se describirán solamente las pruebas de “caja negra” para los casos de uso, para probar la interacción entre el usuario y el sistema, que se satisfagan las precondiciones y poscondiciones, y que se siga la secuencia de acciones intermedias especificadas por el caso de uso. Con el propósito de comprobar que todos los casos de uso de la aplicación sean revisados, se diseñaron los casos de prueba más abarcadores para cada caso de uso, que permiten comprobar las funciones básicas del sistema.

4.3.1 Descripción de los CU de Prueba de Integración.

Caso de uso	Mover Jugador
Caso de prueba	Mover jugador hacia adelante.
Entrada	Se presiona la tecla hacia adelante del cursor.
Resultado	El sistema mueve el personaje en función del estado definido.
Condiciones	El juego debe estar iniciado en un nivel.

Caso de uso	Mover Jugador
Caso de prueba	Mover jugador hacia atrás.
Entrada	Se presiona la tecla hacia atrás del cursor.
Resultado	El sistema mueve el personaje en función del estado definido.
Condiciones	El juego debe estar iniciado en un nivel.

Caso de uso	Mover Jugador
Caso de prueba	Mover jugador hacia la izquierda.
Entrada	Se presiona la tecla hacia la izquierda del cursor.
Resultado	El sistema mueve el personaje en función del estado definido.
Condiciones	El juego debe estar iniciado en un nivel.

Caso de uso	Mover Jugador
Caso de prueba	Mover jugador hacia la derecha.
Entrada	Se presiona la tecla hacia la derecha del cursor.
Resultado	El sistema mueve el personaje en función del estado definido.
Condiciones	El juego debe estar iniciado en un nivel.

Caso de uso	Manipular Animación
Caso de prueba	Ejecutar una animación lineal.
Entrada	Establece frame inicial y final animación.
Resultado	Se ejecuta la animación.
Condiciones	El juego debe estar iniciado en un nivel.

Caso de uso	Manipular Animación
Caso de prueba	Ejecutar una animación cíclica.
Entrada	Establece frame inicial y final animación.
Resultado	Se ejecuta la animación y se mueve hacia el frame inicial.
Condiciones	El juego debe estar iniciado en un nivel.

Caso de uso	Jugar Básico Nivel 1
Caso de prueba	Recolección de tareas.
Entrada	Se da clic en el orden de selección.
Resultado	El estado de la tarea es correcto.
Condiciones	Tarea activa.

Caso de uso	Jugar Básico Nivel 1
Caso de prueba	Recolección de tareas.
Entrada	Se da clic de forma desordenada.
Resultado	El estado de la tarea es incorrecto.
Condiciones	Tarea activa.

Caso de uso	Jugar Básico Nivel 1
Caso de prueba	Chequear tiempo de tarea.
Entrada	El niño intenta completar la tarea y se le acaba el tiempo.
Resultado	Se toma como un fallo o desacierto.
Condiciones	Tarea activa.

Caso de uso	Jugar Bajo Presión Nivel 1
Caso de prueba	Recolección de tareas.
Entrada	El niño intenta recoger las cargas.
Resultado	Aparece el guardián y se roba las cargas de una en una.
Condiciones	15% de desaciertos.

Caso de uso	Jugar Básico Nivel 2
Caso de prueba	Entrar al perímetro de una tarea.
Entrada	Distancia entre el jugador y la tarea.
Resultado	En función de la distancia la tarea se hace más nítida.
Condiciones	El usuario ha iniciado el nivel 2.

Caso de uso	Jugar Básico Nivel 2
Caso de prueba	Resolver tarea.
Entrada	Cantidad de clic según el número de la tarea.
Resultado	Si son iguales el estado de la tarea es correcto.
Condiciones	El usuario ha iniciado el nivel 2.

Caso de uso	Jugar Básico Nivel 2
Caso de prueba	Oponente entra en acción.
Entrada	Se activa el estado del oponente.
Resultado	Oponente intenta resolver tarea.
Condiciones	Tarea incorrecta.

Caso de uso	Solicitar Ayuda Nivel 1
Caso de prueba	Solicitar nivel de ayuda.
Entrada	El niño solicita la ayuda.
Resultado	Según sea la secuencia de energía determinada ejecuta el nivel de la ayuda.
Condiciones	El usuario ha iniciado el nivel 1.

Caso de uso	Solicitar Ayuda Nivel 1
Caso de prueba	Repetir nivel de ayuda.
Entrada	El niño solicita la ayuda de nuevo.
Resultado	Ejecuta el nivel de ayuda correspondiente.
Condiciones	Tener suficiente energía.

Caso de uso	Solicitar Ayuda Nivel 2
Caso de prueba	Solicitar nivel de ayuda.
Entrada	El niño solicita la ayuda.
Resultado	Según sea la secuencia de energía determinada ejecuta el nivel de la ayuda.
Condiciones	El usuario ha iniciado el nivel 2.

Caso de uso	Solicitar Ayuda Nivel 2
Caso de prueba	Repetir nivel de ayuda.
Entrada	El niño solicita la ayuda de nuevo.
Resultado	Ejecuta el nivel de ayuda correspondiente.
Condiciones	Tener suficiente energía.

Al terminar este capítulo se puede decir que se logró el cumplimiento de los objetivos trazados. Además se realizaron los casos de prueba que permitirán en el futuro la realización de las pruebas de caja negra.

CONCLUSIONES

Los estudios referentes a las teorías de la Discalculia permitieron la orientación hacia el cumplimiento de los objetivos generales de los módulos a desarrollar.

El análisis y diseño de los niveles del juego correspondientes a las tareas estimar cantidades y comparación así como de los componentes principales del Kernel, posibilitará la implementación de los niveles correspondientes a todas las tareas matemáticas propuestas para el juego.

Se demostró la funcionalidad del sistema propuesto, llegando a resultados muy positivos con la implementación de las funcionalidades principales del Kernel vistas a través del primer nivel del juego.

RECOMENDACIONES

En el desarrollo de soluciones informáticas o software para la intervención de la Discalculia aún queda mucho camino por andar. En el país se han dado ya los primeros pasos y entre estos se encuentra el presente trabajo que ha demostrado su utilidad con el demo propuesto. Por esta razón se recomienda:

- Dar continuación a este trabajo, implementando los niveles correspondientes a todas las tareas matemáticas propuestas para el juego.
- Desarrollar futuras versiones del juego compatibles con otros sistemas operativos, en específico Linux.
- Utilizar los componentes principales del Kernel para ampliar esta propuesta hacia la intervención de trastornos como la dislexia, entre otros.

BIBLIOGRAFÍA

Referencias Bibliográficas

1. **Dansilio, Sergio.** Discalculias: Perspectivas y aspectos neuropsicológicos. [Online] 2007. <http://www.fnc.org.ar/pdfs/dansilio.pdf>.
2. **Bocaranda, Blanca García.** Cuando 2 y 2 son cinco. [Online] 2004. <http://www.gentiuno.com/articulo.asp?articulo=1285>.
3. **Velmont, Horacio.** El Asombroso descubrimiento de la mente reactiva automática. [Online] 2006. <http://www.grupoelron.org/autoconocimientoysalud/discalculia.htm>.
4. **Martínez, F.** Encuentran la zona del cerebro implicada en la discalculia. [Online] 2007. <http://dislexianews.blogspot.com/2007/03/encuentran-la-zona-del-cerebro.html>
5. **Universidad de Barcelona.** Bases neuroanatómicas y neurofuncionales de la discalculia del desarrollo. [Online] 2007. http://www.ub.es/neuroscience/templates/Prevalencia_genero.htm.
6. **Dislecan.** La discalculia - El patito feo de la familia de las DEAS. [Online] 2008. http://dislecan.blogspot.com/2008_06_10_archive.html.
7. **Varas, Sofía.** Software para aprender a aprender. [Online] 2004. <http://mouse.latercera.cl/2004/productos/01/23/index.asp>.
8. **González Avila, María Gloria.** Herramientas de Autoría de Software Educativo. [Online] 2007. <http://mgonzalez6244.blogspot.com/2007/05/herramientas-de-autora-de-software.html>.
9. **Hurtado, Torres; Bautista, Juan.** Las nuevas tecnologías aplicadas a las necesidades educativas especiales. Análisis de software educativo para atender a alumnos y alumnas con autismo y/o trastornos generalizados del desarrollo (T.G.D.S). Estudios de caso. [Online] 2002. http://www.cibernetia.com/tesis_es/MATEMATICAS/CIENCIA_DE_LOS_ORDENADORES/ENSE%20DIANZA_CON_AYUDA_DE_ORDENADOR/2.
10. **Q, Paula.** Niños en acción (Material anexo) El computador en la educación. [Online] 2007. <http://revolucioninfantil.blogia.com/2007/061901--material-anexo-el-computador-en-la-educacion-php>.
11. **Magdalena.** Concepto de Realidad Virtual. [Online] 2007. <http://www.monografias.com/trabajos11/realitua/realitua.shtml>.
12. **Cugota, Lluís; Salas, Carles.** La realidad virtual (RV). [Online] 2007. <http://www.infonomia.com/if/articulo.php?id=29>.
13. **Hilera, José R; Otón, Salvador; Martínez, Javier.** Aplicación de la Realidad Virtual y VRML en la enseñanza. [Online] 2005. <http://www.ucm.es/info/multidoc/multidoc/revista/num8/hilera-oton.html#aplicaciones>.
14. **Eduteka.** Los Manipulables en la enseñanza de las matemáticas. [Online] 2003. <http://www.eduteka.org/Manipulables.php>.
15. **CERI (Centre for Educational Research and Innovation).** Mini-symposium on the design of rehabilitation software for dyscalculia. [Online] 2003. <http://www.oecd.org/dataoecd/50/39/18268884.pdf>.

16. **Reyes Barón, Liliana Margarita; León Zambrano, Leslie Sofía; Mosquera Ortega, Yennifer.** Software interactivo de educación infantil orientado a la Web para el aprendizaje de las operaciones básicas de las matemáticas: “Un método significativo para el Saber”. [Online] 2004. <http://www.tecnologicocomfenalco.edu.co/iacademica/sistemas/Agora/articulos/educativo.doc>.
17. **Tea Ediciones S.A.** Software Educativo. http://www.teaediciones.com/teaasp/software_educativo.asp . [Online] 2005.
18. **Dislecan.** Dislecan en Austria. [Online] 2007. http://dislecan.blogspot.com/2007_03_24_archive.html.
19. **Wilson, Anna; Revkin, Susannah; Cohen David; Cohen, Laurent; Dehaene, Stanislas.** An open trial assessment of "The Number Race", an adaptive computer game for remediation of dyscalculia. [Online] 2006. <http://www.behavioralandbrainfunctions.com/content/2/1/20>.
20. **Unite de Neuroimagerie Cognitive.** Spanish v2.3, La carrera de los números. [Online] 2006. <http://www.unicog.org/main/pages.php?page=Number%20Race%20Download..>
21. **Wikipedia.** Game Engine. [Online] 2008. http://en.wikipedia.org/wiki/Game_engine.
22. **McGuire, Morgan.** G3D Documentation version 7.00. [Online] 2007. <http://g3d-cpp.sourceforge.net/manual/index.html>.
23. **Polo Productivo Realidad Virtual Facultad 5 (UCI).** *Herramienta Básica para Desarrollo de Sistemas de Realidad Virtual. Scene Toolkit version 2.3.1.* 2008.
24. **Wikipedia.** Acción en primera persona. [Online] 2008. http://es.wikipedia.org/wiki/Acci%C3%B3n_en_primera_persona.
25. **Wikipedia.** Tercera Persona. [Online] 2008. http://es.wikipedia.org/wiki/Tercera_persona.
26. **Adams, David.** Automatic Generation of Dungeons for Computer Games. [Online] 2002. <http://www.dcs.shef.ac.uk/intranet/teaching/projects/archive/ug2002/pdf/u9da.pdf>.
27. **Zito, Míriam.** Detección temprana de discapacidades, una realidad en Cuba. [Online] 2006. <http://www.radiorebelde.com.cu/noticia/salud/salud1-161106.htm>.

Bibliografía Consultada

1. **Giordano, Luis; y otros.** *Discalculia Escolar, Dificultades en el aprendizaje de las Matemáticas.* Buenos Aires : Ateneo, 1978 .
2. **María, José Martínez.** *Problemas escolares, Dislexia, Disgrafía y Discalculia.* Barcelona : Cimel Kapeluz, 1982.
3. **Jacobson, Ivar and otros, y.** *El Proceso Unificado de Desarrollo de Software.* 2000.
4. **Pressman.** *Ingeniería del Software un enfoque práctico.* 2001.
5. **El Foro de Mundogamers.** ¿Qué prefieres juegos en tercera o en primera persona? [Online] 2008. <http://www.mundogamers.com/foros/thread27476.html>.
6. **Tucci, Giancarlo.** TUTOR - software per apprendere - dislessia, discalculia, disattenzione. [Online] 2004. <http://www.leggiscrivi.it>.
7. **Iprase Trentino.** [Online] 2006. http://www.iprase.tn.it/prodotti/software_didattico/giochi/matematica/index_en.asp.
8. **Psicopedagogía.** Software Educativos Específicos. [Online] 2007. http://aprendoluegoexisto.blogspot.com/2007_04_01_archive.html.
9. **ELPAIS.** Los juegos de acción en primera persona mejoran la vista. [Online] 2007. http://www.elpais.com/articulo/internet/juegos/accion/primera/persona/mejoran/vista/elpeputec/20070207elpepunet_3/Tes.
10. **Pullga, Svano and Savelli, Enrico.** Laboratorio 6 Software per la discalculia. [Online] 2006. <http://www.nataliavisalli.it/livellob/modulo7/laboratori/laboratorio6.pdf>.
11. **Cerda, Carol.** Dejando Huellas en la puerta del cielo. [Online] 2007. <http://n0anyel.blogspot.com/2007/10/las-matematicas-un-aprendizaje-tan.html> .
12. **Uriarte, Rosina.** Estimulación Temprana y Desarrollo Infantil. [Online] 2007. <http://estimulacionydesarrollo.blogspot.com/2007/05/discalculia.html> .

ANEXOS

Anexo 1: Diagrama de Clases

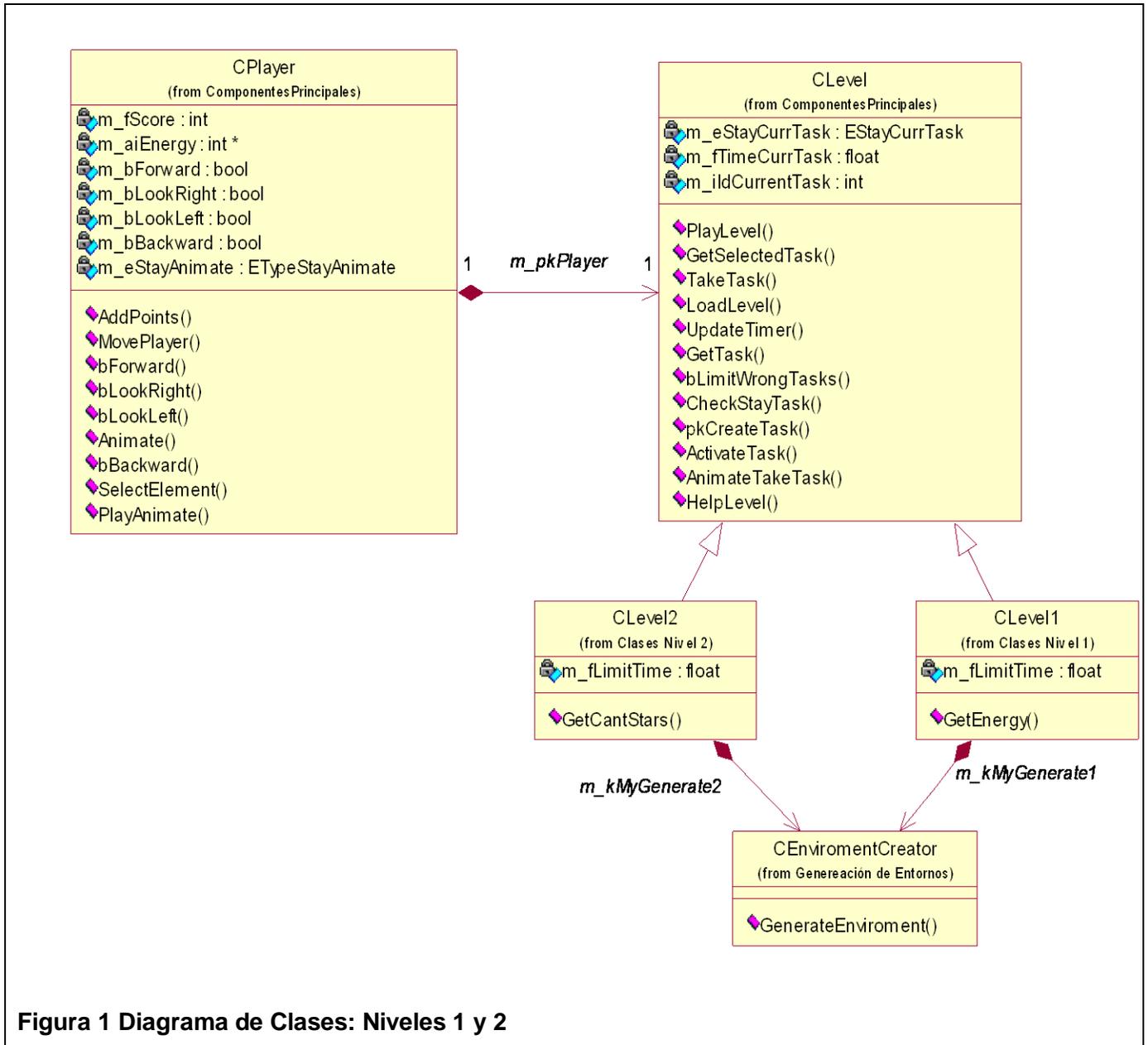


Figura 1 Diagrama de Clases: Niveles 1 y 2

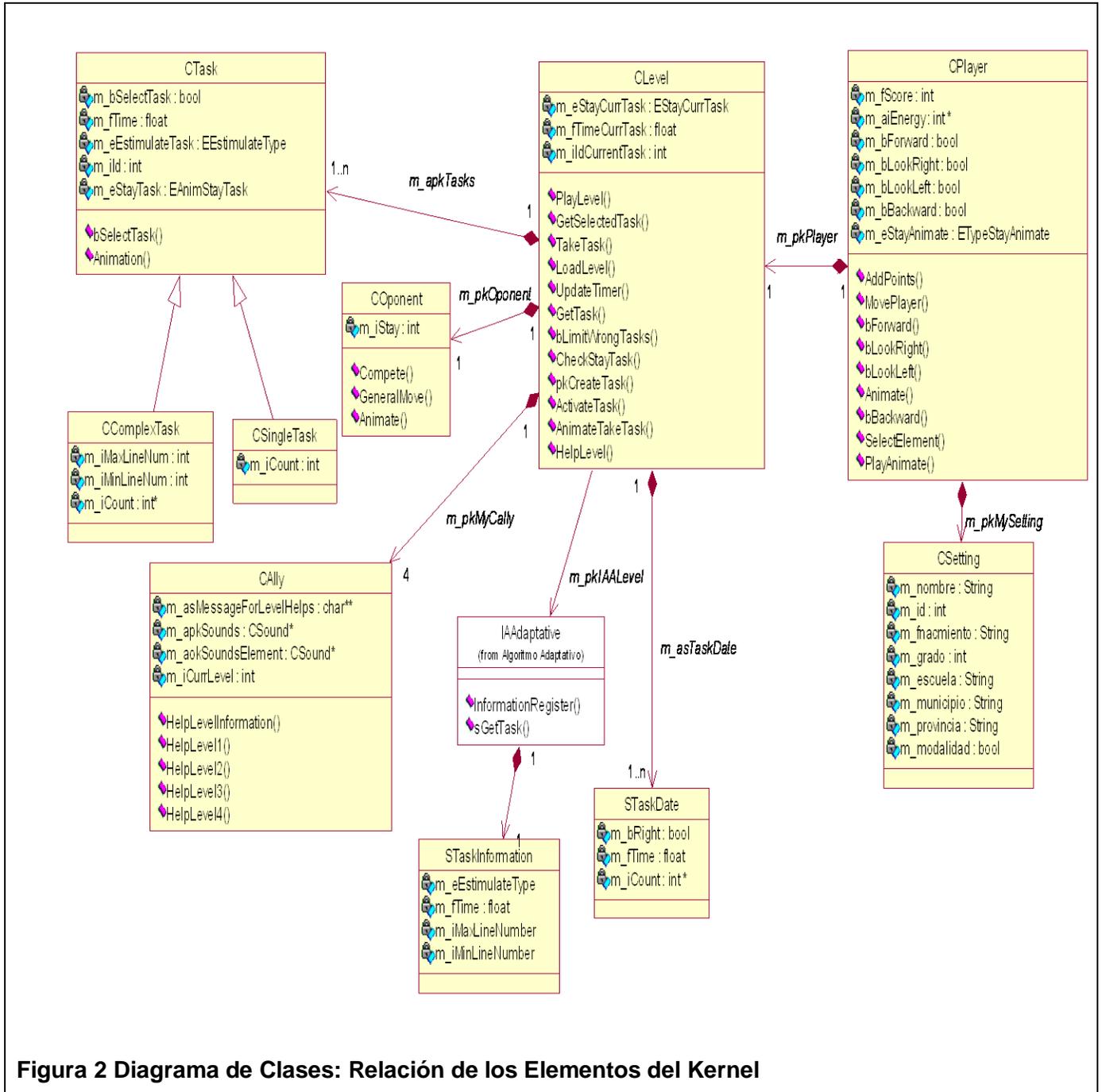
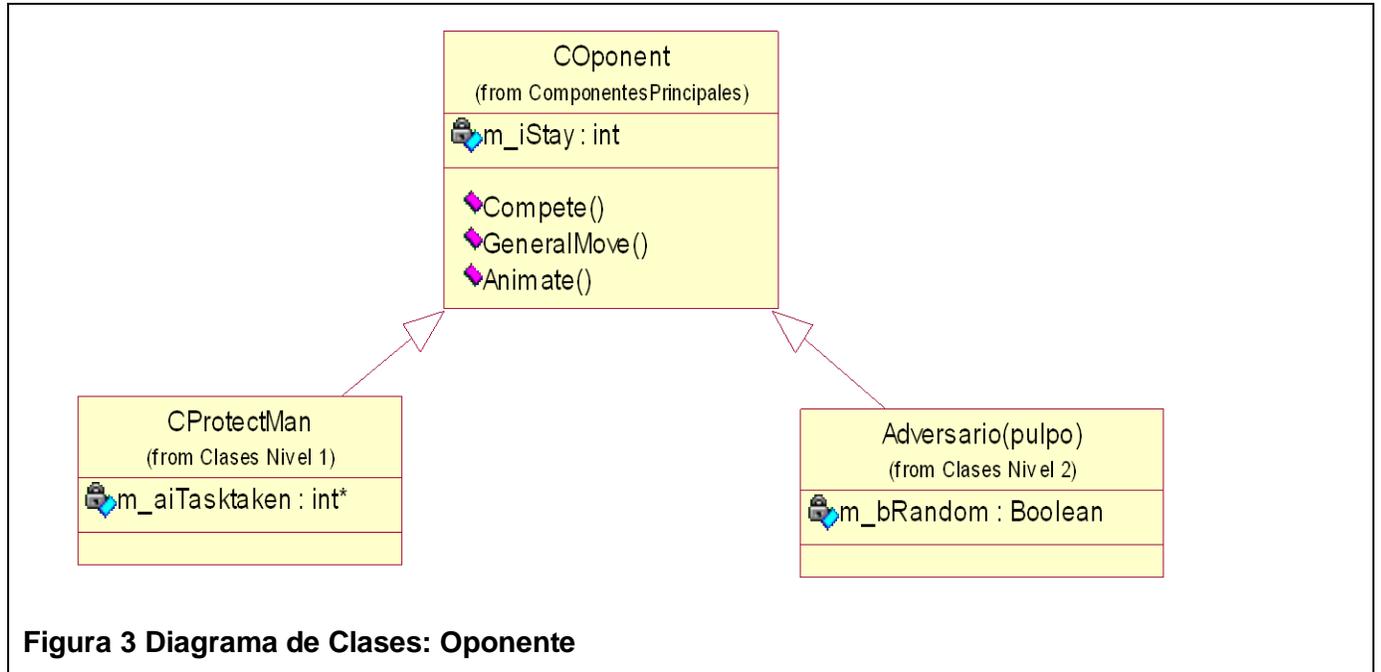


Figura 2 Diagrama de Clases: Relación de los Elementos del Kernel

**Figura 3 Diagrama de Clases: Oponente**

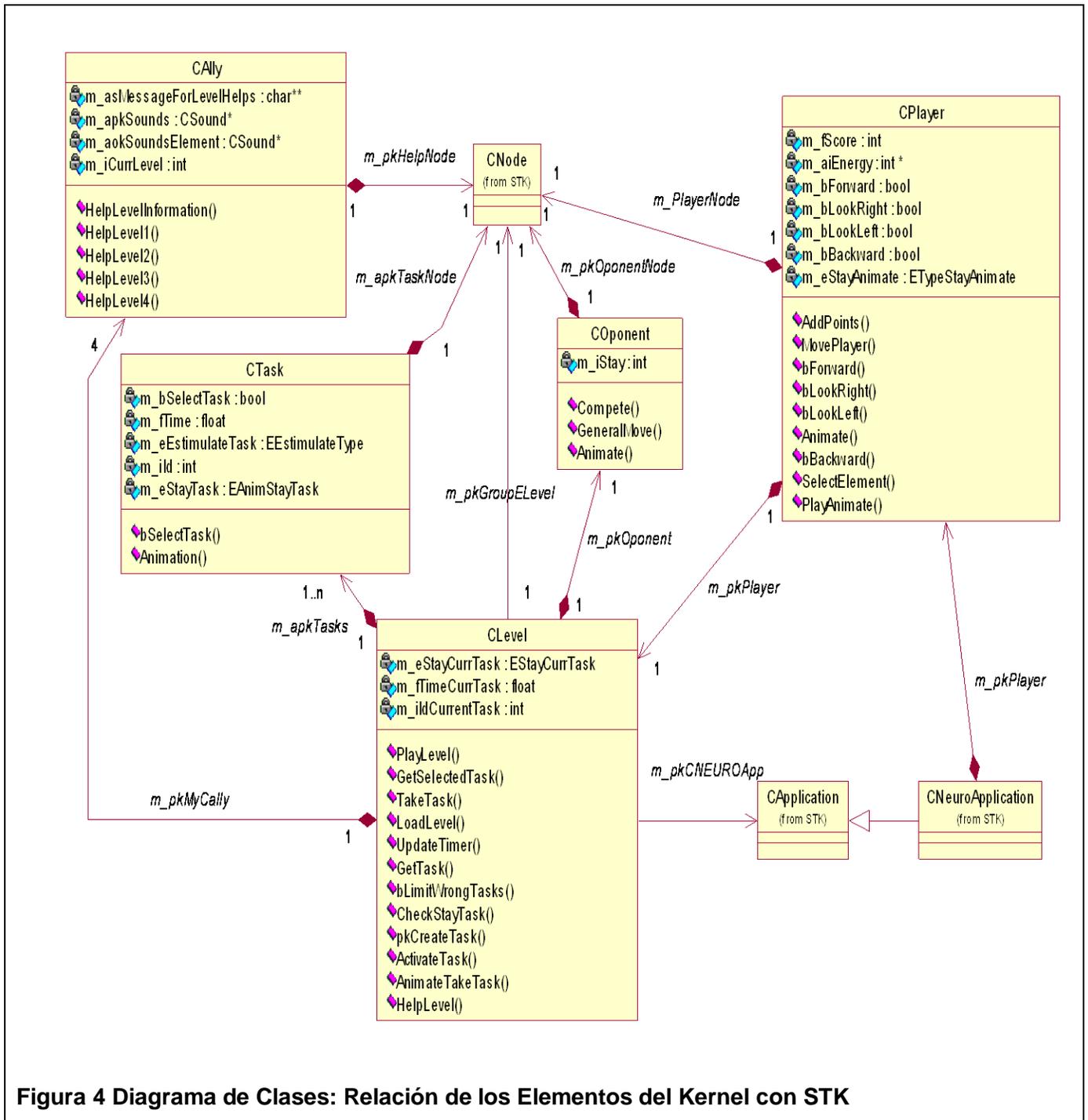


Figura 4 Diagrama de Clases: Relación de los Elementos del Kernel con STK

Anexo 2: Diagrama de Secuencia

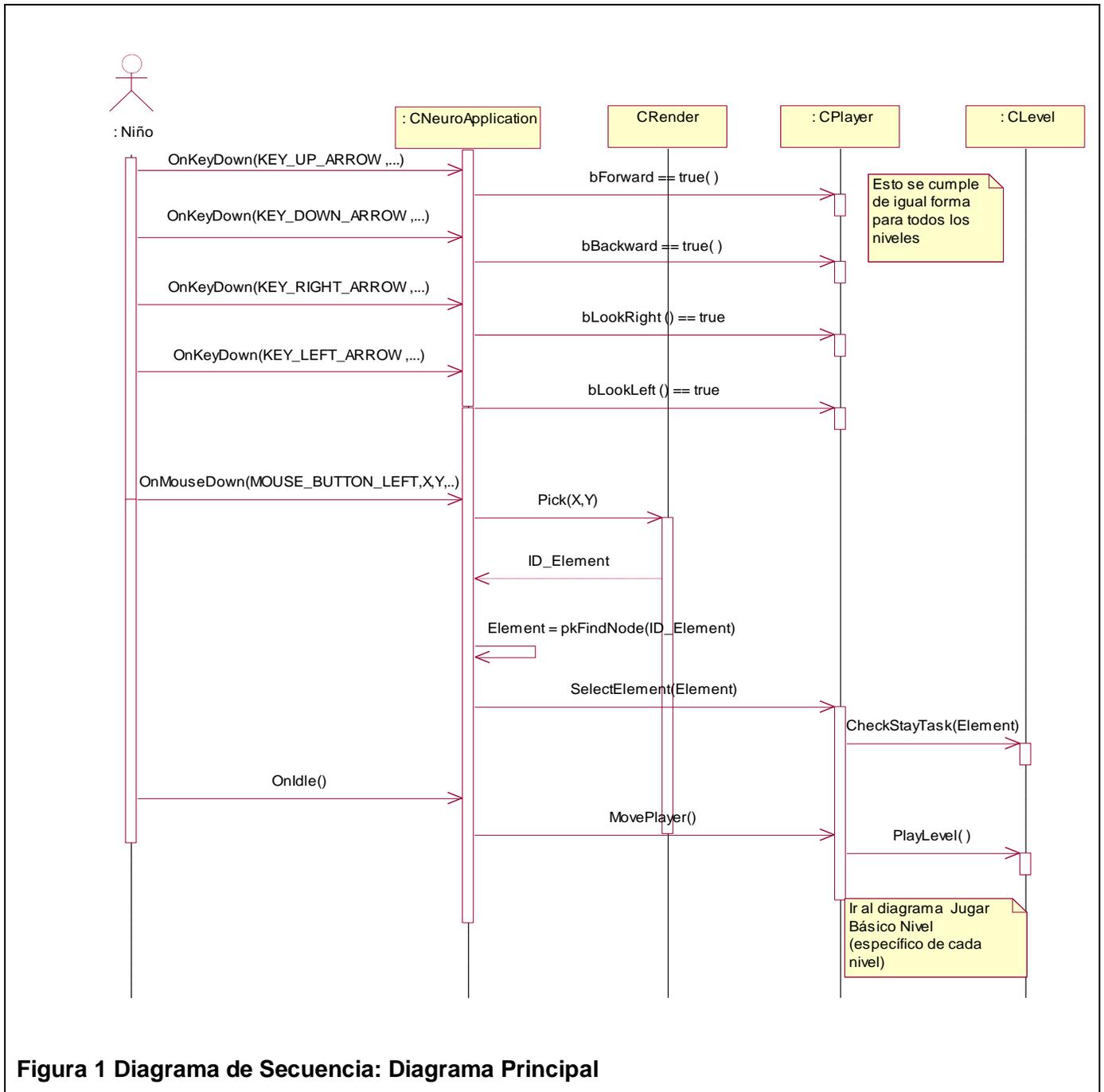
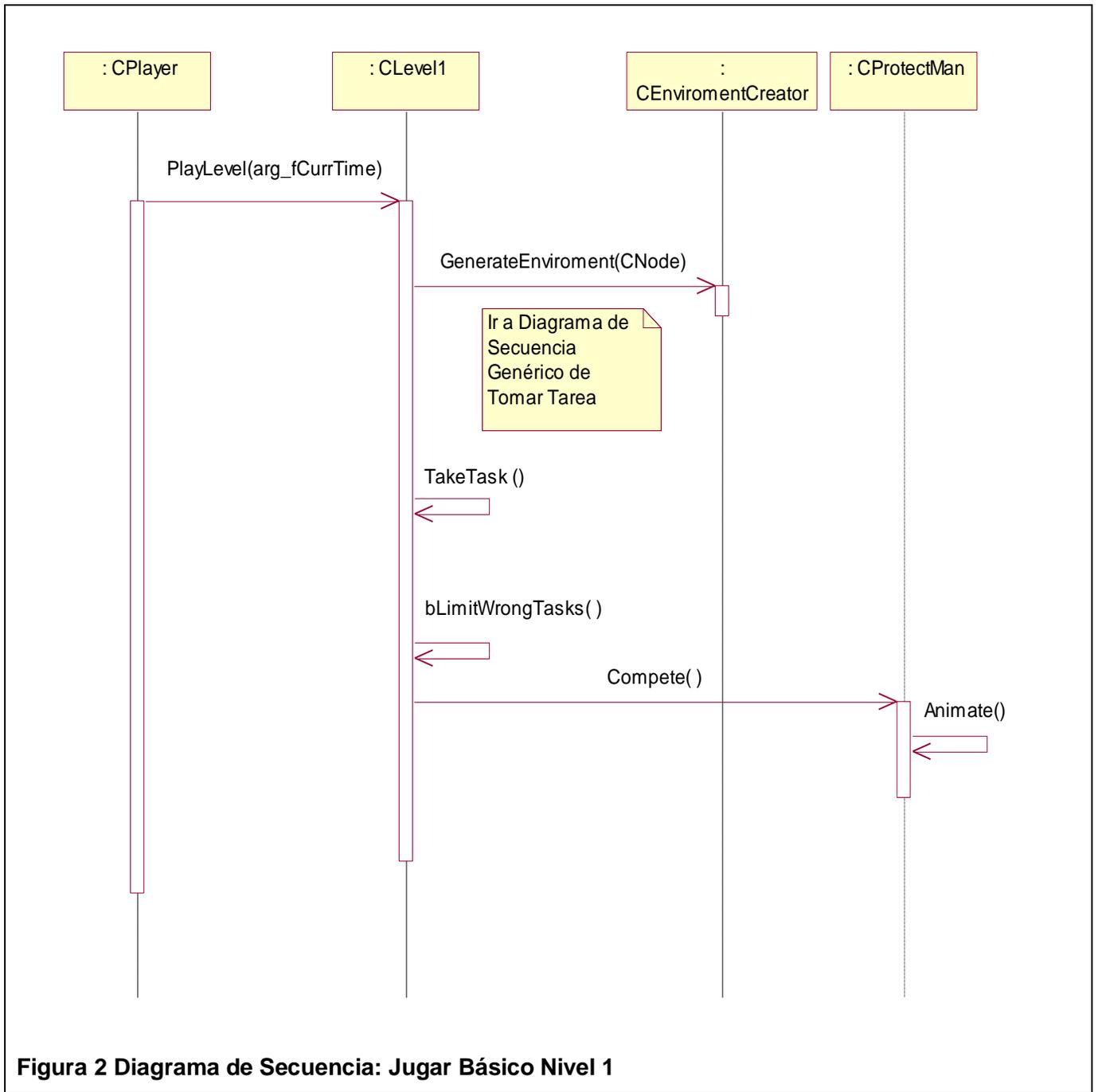


Figura 1 Diagrama de Secuencia: Diagrama Principal



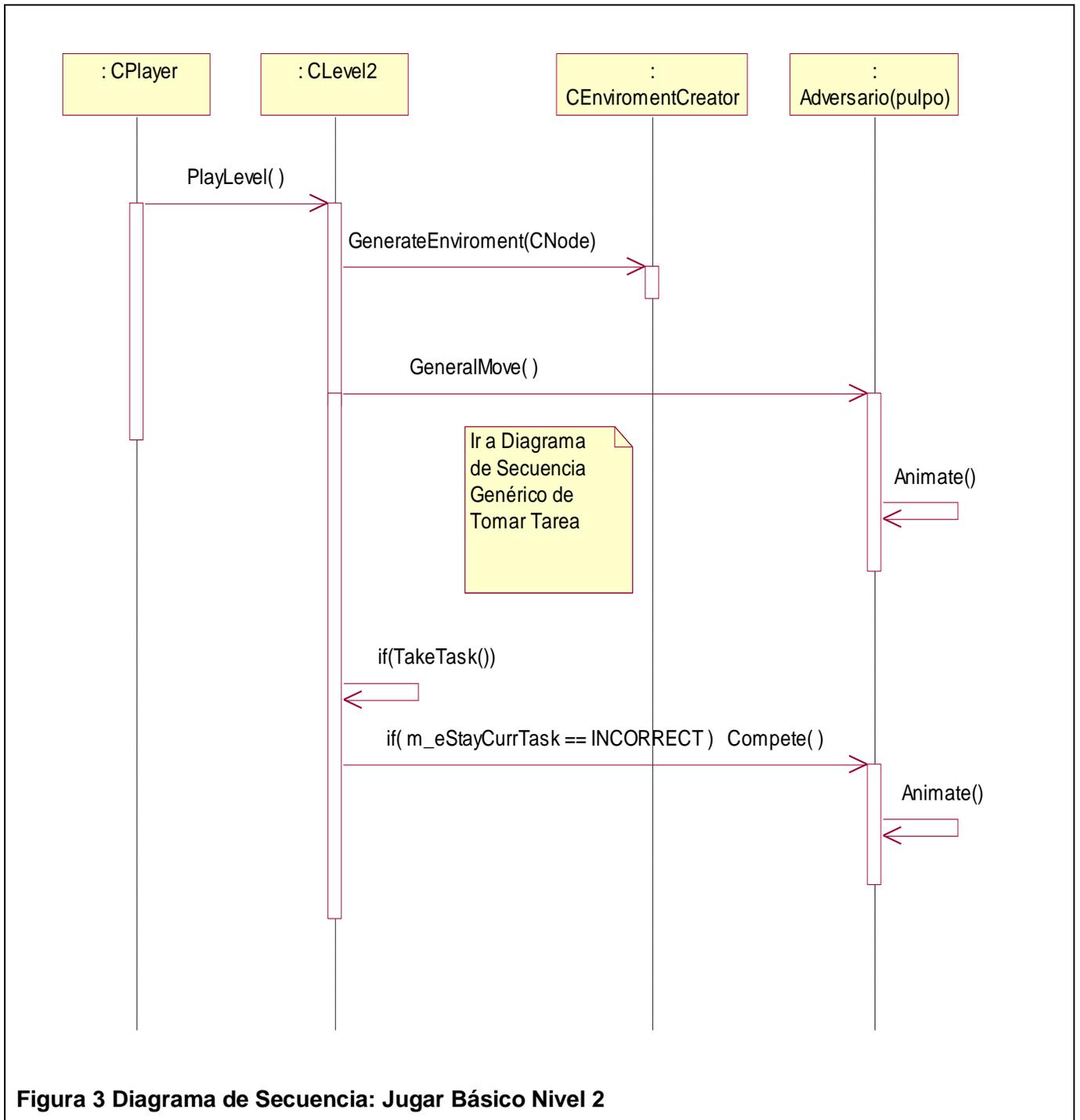


Figura 3 Diagrama de Secuencia: Jugar Básico Nivel 2

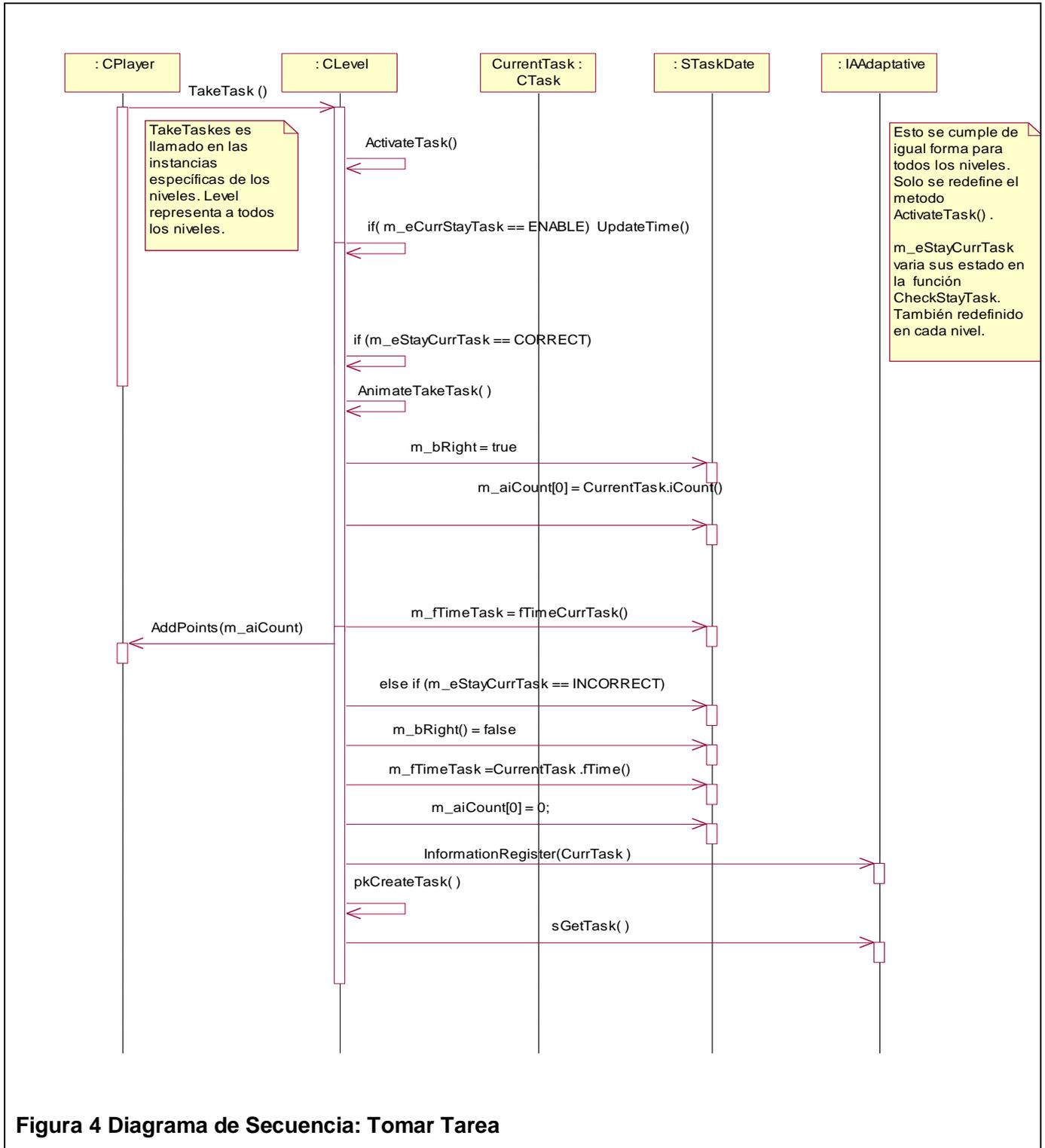
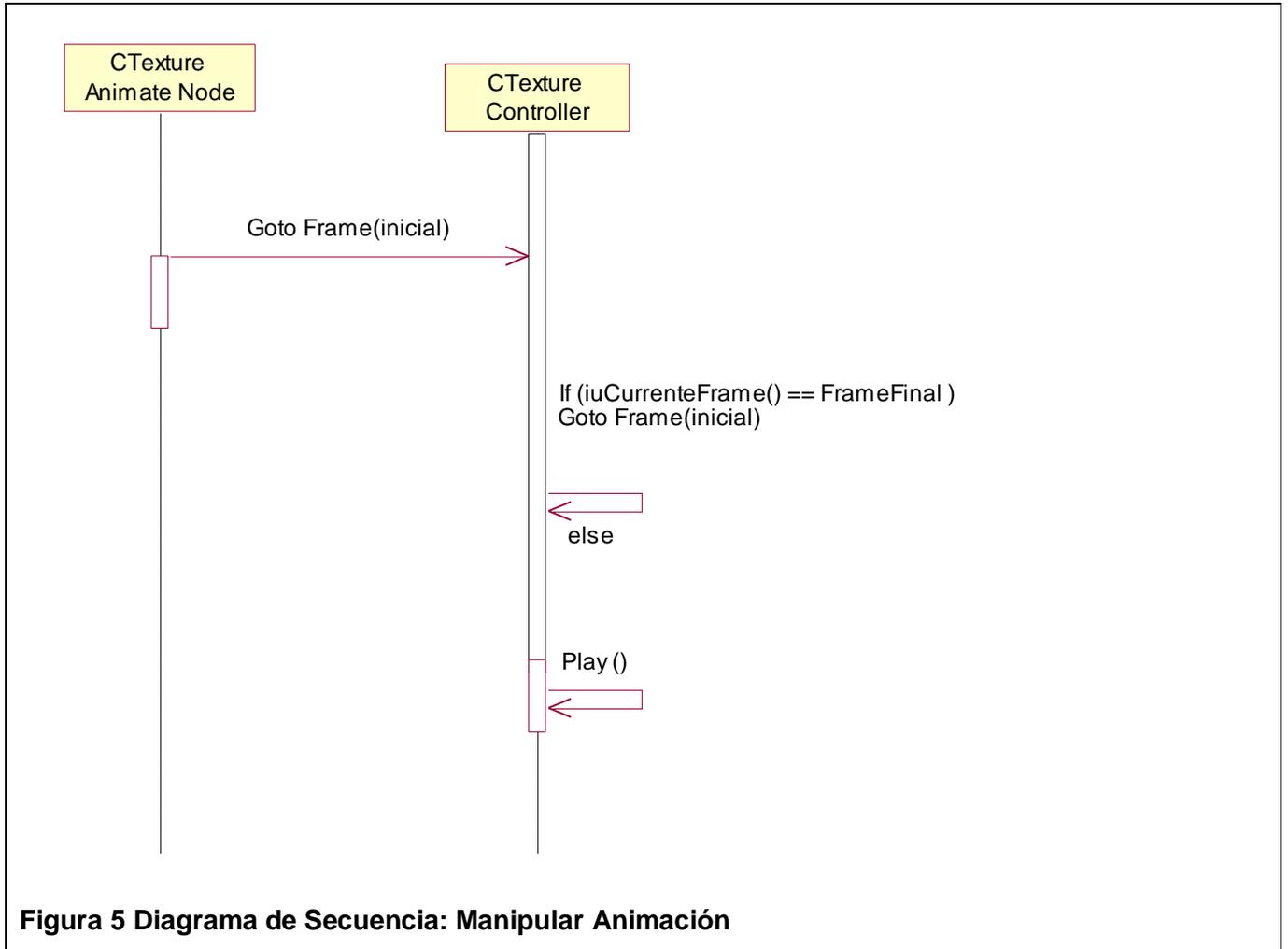


Figura 4 Diagrama de Secuencia: Tomar Tarea



GLOSARIO DE TÉRMINOS GENERAL

A:

Acalculia: Incapacidad de realizar operaciones aritméticas, asociada con lesiones en el cerebro, no se trata de una dificultad de aprendizaje sino de un trastorno específico del cálculo. Corresponde a un síntoma de la Demencia Cortical. Del latino (a = negativo + cálculo = calcular)

Aliado: Ente o personaje que interactúa con el jugador ofreciéndole ayuda.

Animación: Simulación de un movimiento creada por la muestra de una serie de imágenes o fotogramas.

Algoritmo Adaptativo: Conjunto de operaciones y cálculos que siguiendo una metodología establecen cuándo ha llegado el final del juego y la complejidad de las tareas a realizar.

C:

Camino: Ruta que sigue el jugador.

Colisión entre Objetos: Choque o encuentro de dos o más entes.

D:

Dinámica de Movimiento: Conjunto de movimientos que siguen determinadas reglas.

E:

Entorno Dinámico: Ambiente o entorno del juego que se construye en tiempo real.

I:

IDE: Integrated Development Environment, en español Entorno de Desarrollo Integrado

J:

Jugador: Ente o personaje que representa al niño.

N:

Nivel: Planeta donde hay que alcanzar una puntuación realizando las tareas.

Nivel 1: Planeta Corredoris.

Nivel 2: Planeta Oceanía.

O:

Oponente: Ente o personaje que interactúa con el jugador en forma de reto.

P:

Personaje: Actor de la escena de un mundo de realidad virtual, que soporta acciones (como un tipo de comportamiento), y que tienen entre sus atributos, cualidades físicas y emocionales que serán usadas a la hora de ejecutar las acciones, así como determinados roles (ver rol).

Planeta: Entorno con características particulares donde existen oponentes y aliados y además deben realizarse tareas específicas.

Puntuación: Acumulación de puntos que se adquieren al realizar satisfactoriamente las tareas.

T:

Tareas: Operaciones que deben ser realizadas por el jugador y que son determinadas por el algoritmo adaptativo.