

Universidad de las Ciencias Informáticas
Facultad 5



**Soft-Coordinator: Sistema para la
rehabilitación visual en menores con
Estrabismo y Ambliopía**

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



Autores: Rinués Núñez Ruiz

Lianet Guevara Alvarez

Tutor: MSc. Pedro Carlos Pérez Martinto

Co-tutor: Ing. Minardo G. González

Ciudad de La Habana
Julio, 2008

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 7 días del mes de julio del año 2008.

Lianet Guevara Alvarez

Firma del Autor

Rinués Núñez Ruiz

Firma del Autor

MSc. Pedro Carlos Pérez Martinto

Firma del Tutor

Ing. Minardo G. González

Firma del Cotutor

DATOS DE CONTACTO

MSc. Pedro Carlos Pérez Martinto

Profesor asistente e investigador agregado.

Correo electrónico: pcpmartinto@uci.cu

Graduado de Licenciatura en Educación, Especialidad Defectología. Máster en Educación Especial, Profesor Jefe de Metodología de la Investigación - UCI. Asesor de investigaciones Facultad 5. Especialista en tecnología Educativa - MINED.

Ing. Minardo G. González

Profesor de Programación I-UCI, Líder de proyecto de Juegos Consola.

Correo electrónico: mgonzalezl@uci.cu

Graduado en Ingeniería Informática en la UCI, 2007.

Facultad 5-UCI.

*Hombre es algo más que ser torrente vivo:
es entender una misión, ennoblecerla y cumplirla.*

José Martí

Agradecimientos

Agradecemos a nuestra Revolución y al Comandante Fidel Castro por permitirnos estudiar en esta Universidad de Excelencia.

A la Universidad de Ciencias Informáticas y a sus profesores por prepararnos política, ideológica y técnicamente como buenos Ingenieros.

A nuestro tutor Martinto, por su ayuda y apoyo siempre presente, por enseñarnos tanto de educación especial y por instarnos a crear tan maravilloso proyecto, a él muchas gracias.

A Osmany, por su ayuda incondicional.

A Minardo y Yirka, por estar siempre con nosotros.

A nuestros padres por su amor y apoyo en cada momento, por la confianza depositada en nosotros, a ellos les debemos cuanto somos, cada una de las letras de este trabajo es para ustedes.

A nuestros amigos y hermanos por su ayuda, su compañía, sus consejos y por confiar en nosotros.

A todos aquellos que de una forma u otra nos apoyaron, sin la ayuda de ustedes estas páginas estarían en blanco.

Dedicatoria

Dedicamos esta tesis a los niños cubanos que padecen de ambliopía y estrabismo y a los Centros Especiales de Rehabilitación Visual.

Dedicamos especialmente a nuestro tutor Martinto, por ser más que un guía, un amigo que estuvo siempre a nuestro lado, alguien que nos enseñó muchas cosas, confió siempre en nosotros y compartió algo de su sabiduría en instruir, más que a nuestras mentes a nuestras almas.

Dedico esta tesis a nuestra Universidad por estos cinco años inolvidables; a mis profes por haberme preparado como una ingeniera con grandes valores y conocimientos; a mi familia por su apoyo y comprensión; a mi papá por su confianza y cariño; a mi mamá por su bello amor; a Rinués por su comprensión, su confianza, su amistad; a Sonia, Yusla, Joel y Lisy por estar siempre conmigo.

A todos mis amigos, a los que están lejos.

A Wal, por ser, por estar, por existir.

A Yassel, dondequiera que estés.

Lianet

Dedico este trabajo:

A mi madre por ser la luz que me ha guiado hasta este momento de mi vida, este triunfo es suyo.

A mi padre por su constante preocupación por que este trabajo saliera y por mi bienestar.

A la memoria de aquellos que ya no están y que siempre me encaminaron en la vida, a mi abuelo Ibis, a mi abuela Mima, a mi tío Alberti, agradezco siempre el tiempo que compartí con ustedes.

A mis tres hermanas por sus muestras de cariño, por ser mis mejores amigas.

A Yury por estar a mi lado a lo largo de estos cinco años, por ser fuente inagotable de amor y comprensión.

A mi compañera de tesis, Lianet, gracias por tu apoyo y amistad.

A mi familia, por su preocupación y sus consejos, por apoyarme siempre.

A Edel y a Guzmán por ser los hermanos que no tuve, por su apoyo incondicional.

A todos los que de una forma u otra han hecho posible este sueño.

Rinués

Resumen

En nuestro país las Escuelas Especiales para la Rehabilitación Visual usan diferentes herramientas, instrumentos, técnicas, métodos clínico-pedagógicos y procedimientos para el tratamiento oftalmológico de los niños que padecen Ambliopía y Estrabismo. Muchos de estos equipos y herramientas como el Coordinator-Bangester han presentado problemas en su funcionamiento además de no tener los recursos psicológicos necesarios para una buena estimulación de los pacientes en el tratamiento. En los últimos 10 años se han realizado varias investigaciones científicas basadas en esta situación con el objetivo de buscar nuevas soluciones.

El uso de las TICs en las escuelas de Cuba ha creado condiciones favorables para el desarrollo del Proceso Docente Educativo. Estas tecnologías, además, pueden contribuir al desarrollo científico de nuestra sociedad y servir como una herramienta que colabore con los procesos rehabilitativos desarrollados en las Escuelas Especiales. Sin embargo no se había creado ningún sistema informático que contribuyera a ello.

Así surge Soft-Coordinator, un sistema de rehabilitación para menores que padecen de Estrabismo y Ambliopía y son tratados en estos centros especiales. El sistema, que se basa en los principios de funcionamiento de la herramienta Coordinator-Bangester, consistirá en un juego terapéutico en el que se controlaran los datos de los pacientes, así como los resultados de cada tratamiento que estos realicen.

En este documento se plasman los resultados del estudio realizado para la construcción del sistema, se explican los conceptos relacionados con el mismo, se hace un análisis de la propuesta del sistema, y se reflejan algunas recomendaciones para el mejoramiento futuro del mismo.

PALABRAS CLAVE

Ambliopía, Estrabismo, Coordinator-Bangester, Paciente, Administración, Juego, Tratamiento, Reporte.

ÍNDICE

Agradecimientos	I
Dedicatoria	II
Resumen	III
Introducción	1
Capítulo 1: Fundamentación Teórica	7
1.1 Introducción.....	7
1.2 Teorías física-ópticas y fisiológicas del proceso perceptivo	7
1.3 Coordinator-Bangester: una herramienta usada en Cuba	8
1.4 Investigaciones realizadas en Centros de Rehabilitación Visual	10
1.5 Aplicaciones para el tratamiento de niños estrábicos y ambliopes.....	11
1.6 Tecnología y Software	13
Capítulo 2: Características del Sistema	16
2.1 Introducción.....	16
2.2 Tendencias y Tecnologías actuales a considerar	16
2.2.1 Metodologías de desarrollo	16
2.2.2 Unified Modeling Language	19
2.2.3 Herramientas CASE	20
2.2.4 Entorno de Desarrollo Integrado	20
2.2.5 Entorno de diseño	22
2.3 Nuestra propuesta.....	22
2.4 Descripción del negocio	23
2.5 Reglas de negocio	24
2.6 Requerimientos del Sistema	25
2.6.1. Requerimientos Funcionales.....	25
2.6.2. Requerimientos No Funcionales	27
2.7 Actores y trabajadores del negocio.....	28
2.8 Patrones de Casos de Uso utilizados.	29
2.9 Descripción de los Casos de Uso del Modelo del Negocio.....	29
2.9.1 Diagrama de Casos de Uso del modelo del negocio	29
2.9.2. Especificación de los Casos de Uso y Diagramas de Actividades.	30
Capítulo 3: Análisis y Diseño del sistema	43
3.1 Introducción.....	43
3.2 Descripción del sistema propuesto	43

3.2.1 Módulo Manage	43
3.2.2 Módulo Game.....	44
3.3 Actor del sistema.....	45
3.4 Modelo de Casos de Uso del Sistema	45
3.4.1 Casos de Uso del Sistema	45
3.5 Casos de Uso del Sistema y Diagramas de Secuencia	48
3.6 Patrones usados en el diseño del Sistema	66
3.6.1 Patrones GRASP	66
3.6.2 Patrones GoF.....	66
3.7 Diagrama de clases	67
3.7.1 Paquete Manage.....	68
3.7.2 Paquete Game	71
Capítulo 4: Implementación	72
4.1 Introducción.....	72
4.2 Control de Datos	72
4.2.1 Diagrama de clases persistentes	72
4.2.2 Modelo de Datos	73
4.3 Principios de diseño	75
4.3.1 Interfaz del Módulo Manage.....	75
4.3.2 Interfaz del Módulo Game.....	76
4.3.3 Concepción general de la Ayuda	77
4.3.4 Tratamiento de excepciones y errores.....	78
4.4 Estándares de codificación	79
4.5 Arquitectura del Sistema	81
4.6 Diagrama de Componentes	81
4.6.1 Explicación de los componentes.....	83
Conclusiones	85
Recomendaciones	86
Bibliografía	87
Glosario de Términos	89
ANEXOS.....	91

INDICE DE FIGURAS

Figura 1 Correspondencia entre colores del espectro y longitud de onda.....	8
Figura 2 Equipo Coordinator-Bangester	9
Figura 3 Diagrama de Casos de Uso del Negocio.....	29
Figura 4 Diagrama de Clases del Modelo de Objetos del Negocio	42
Figura 5 Diagrama de Casos de Uso del Sistema	47
Figura 6 Diagrama de Secuencia CU “Gestionar Registro” Sección: Registrar Paciente.....	51
Figura 7 Diagrama de Secuencia CU “Gestionar Registro” Sección: Modificar Paciente.....	52
Figura 8 Diagrama de Secuencia CU “Gestionar Registro” Sección: Eliminar Paciente	53
Figura 9 Diagrama de secuencia CU “Gestionar Reporte” Sección: Crear Reporte.....	57
Figura 10 Diagrama de secuencia CU “Gestionar Reporte” Sección: Comparar Reportes...	58
Figura 11 Diagrama de secuencia CU “Gestionar Reporte” Sección: Eliminar Reporte.....	59
Figura 12 Diagrama de Secuencia CU “Gestionar Juego” Sección: Crear Juego	62
Figura 13 Diagrama de Secuencia CU “Gestionar Juego” Sección: Eliminar Juego	63
Figura 14 Diagrama de Secuencia CU “Mostrar mensajes”	64
Figura 15 Diagrama de Secuencia CU “Gestionar datos del juego”	65
Figura 16 Diagrama de Clases de Diseño	68
Figura 17 Subpaquete Report.....	69
Figura 18 Subpaquete Patient	70
Figura 19 Subpaquete Treatment	70
Figura 20 Paquete Game.....	71
Figura 21 Diagrama de Clases Persistentes	73
Figura 22 Diagrama del Modelo de Datos	74
Figura 23 Interfaz del Módulo Manage	75
Figura 24 Interfaz del Módulo Game	76
Figura 25 Paquete de Ayuda del Sistema	77
Figura 26 Tratamiento de errores y excepciones.....	78
Figura 27 Verificación de campos requeridos y en formato correcto.....	79
Figura 28 Ejemplo de codificación utilizada	80
Figura 29 Arquitectura del sistema	81
Figura 30 Diagrama de Componentes	82

INDICE DE TABLAS

Tabla 1 Requisitos Funcionales.....	27
Tabla 2 Actores del Negocio	28
Tabla 3 Trabajadores del Sistema	28
Tabla 4 Actor del Sistema.....	45
Tabla 5 Otros componentes.....	84

Introducción

Dentro del sistema de la Enseñanza Especial en Cuba, existen en la provincia de Ciudad de La Habana tres centros donde se brinda atención al **estrabismo** y la **ambliopía**. Dichos centros cuentan con equipos de terapia y ejercicios visuales para realizar el tratamiento, tanto en el consultorio oftalmológico de la escuela como durante las actividades docentes.

Algunos de estos equipos presentes en los consultorios oftalmológicos, no poseen los recursos psicológicos necesarios para el logro de una adecuada motivación y colaboración consciente por parte del niño en el tratamiento, además de que existen grandes dificultades para acceder a las nuevas tecnologías e insumos necesarios para la reparación de dichos equipos.

La aplicación de nuevas alternativas informáticas que correspondan y den cumplimiento a un grupo de teorías científicas y con funciones rehabilitativas de algún área en la que se presente discapacidades psicológicas o del aprendizaje así como de estructuras biológicas, permitirá contar una nueva vía para el tratamiento de éstas.

En discursos pronunciados por nuestro Comandante Fidel Castro Ruz, ha detallado las diferencias que hemos logrado con respecto a etapas anteriores y cuánto poseen estas escuelas. Expuso que *"...las escuelas para niños ambliopes --que no son los ciegos-- tienen que ser diseñadas especialmente (...) Las escuelas para niños ciegos y débiles visuales requieren igualmente instalaciones y equipos muy especiales. Es maravilloso ver después lo que aprenden esos niños y lo que hacen en esas escuelas tan humanas."*¹. Sin embargo producto del período especial muchos de esos equipamientos perdieron la posibilidad de que fueran reparados sistemáticamente, algunos caducaron y otros fueron dados de baja.

Desde hace un tiempo se vienen realizando estudios (Pérez Martinto, PC, 1999), con el fin de buscar alternativas más apropiadas y que trabajaran según los principios de funcionamiento de alguno de estos equipos unido a la confluencia de la utilización de requerimientos psicológicos necesarios en las etapas evolutivas de estos escolares.

¹ CASTRO RUZ, F. *Discurso pronunciado por el Presidente de la República de Cuba en la clausura del II Encuentro Mundial de Educación Especial celebrado en el Teatro "Karl Marx"*, Ciudad de La Habana, 1998.

Introducción

“Las nuevas y favorables condiciones desarrolladas en la actual Revolución Educacional, han permitido la introducción de las *Tecnologías de la Informática y las Comunicaciones* (TICs) con vistas de perfeccionar el *Proceso Docente Educativo* (PDE), que ha llegado a todo el sistema de enseñanza y educación”².

Las TICs pueden contribuir al desarrollo de la ciencia y ello supone una reconceptualización de la enseñanza en sentido amplio. Es aquí donde la pedagogía y el estudio de la tecnología educativa juegan un papel preponderante debido a la necesidad de tener perfectamente definido el tipo de educación a lograr y el proceso docente tiene sus propias leyes y con el uso del ordenador se deben esperar cambios en los métodos, objetivos y contenidos entre otras.

La reconceptualización de la educación con el uso de la computadora debe contribuir a un aprendizaje más rápido y eficiente en una atmósfera agradable, generadora de elevados gradientes motivacionales en el niño y el maestro, en el que se pueda particularizar diferencias individuales, y lograr los cambios esperados en aquellas áreas a corregir, estimular, desarrollar; acceder a nuevas informaciones, optimizar los procesos investigativos, perfeccionar la toma de decisiones, aspectos estos que toman su real significación en la atención de los niños con alguna discapacidad, todo esto resultado de las potencialidades de estos recursos tecnológicos en función de las necesidades del desarrollo de los escolares.

Por lo antes expuesto presentamos la siguiente **situación problémica**: en muchos de los centros educacionales donde se trata este tipo de discapacidad, no existen los medios necesarios para realizar adecuadamente un tratamiento integral a partir de que posean los recursos psicológicos necesarios que mantengan la atención de los escolares objeto de estudio. Además muchos de los equipos como el *Coordinator Bangester* poseen muchos años de explotación, lo cual ha llevado a su deterioramiento y poca recuperación. Es por esto que se hace necesario buscar nuevas soluciones con el uso de las TICs, como medios que extienden sus funciones hasta el ámbito clínico-pedagógico.

² SUÁREZ CASTRO, Y. “*Modelo de Software para la Estimulación Visual de los Escolares Ambliopes con Fijación Central*”. Tesis para optar por el título de Lic. En Educación. ISP EJV, Ciudad de La Habana, 2005.

Introducción

Teniendo en cuenta la importancia social que le concedemos al desarrollo de las funciones visuales en los niños ambliopes y estrábicos, y valorando los aportes teóricos que han sido planteados en investigaciones anteriores, consideramos ensayar el uso de un software que pueda ser utilizado con doble objetivo: rehabilitativo y académico, y lógicamente, lograr por parte del niño una mayor motivación y concentración en las tareas.

Por todo lo expuesto anteriormente se plantea como **problema científico** que *no existe un software elaborado sobre la base de principios de funcionamiento de equipos oftalmológicos que contribuya a la estimulación y rehabilitación visual de niños estrábicos o ambliopes.*

Surge entonces como **objetivo general** *implementar un software para la estimulación y rehabilitación visual de escolares estrábicos y ambliopes.*

El **objeto de estudio** en el que se enmarca este problema es el *proceso de rehabilitación y estimulación visual.* Siendo el **campo de acción** *la estimulación y rehabilitación visual de niños ambliopes mediante el desarrollo de habilidades óculo manuales desde la utilización de los ordenadores.*

Para cumplir los objetivos trazados, se desarrollaron las siguientes **tareas investigativas**:

- ❖ Realizar análisis de las concepciones científico teóricas que sustentan la rehabilitación visual mediante el uso de los software educativos.
- ❖ Estudiar el funcionamiento de la herramienta SceneToolKit 2.3 para su aplicación.
- ❖ Realizar un estudio del entorno de trabajo y de las herramientas más eficientes para la confección de los módulos.
- ❖ Definir metodología para realizar el desarrollo del software.
- ❖ Realizar el análisis y diseño del software a través de la metodología seleccionada.
- ❖ Desarrollar los módulos descritos en el diseño del software.

Con el desarrollo del Soft-Coordinator se espera obtener una aplicación informática que, basada en los principios de funcionamiento del equipo oftalmológico Coordinator-Bangester, pueda utilizarse como una herramienta con función rehabilitativa y educativa.

Métodos, técnicas y procedimientos utilizados

Para el desarrollo de este trabajo se tuvo en cuenta el empleo de diferentes métodos de investigación. “Podemos establecer dos grandes clases de métodos de investigación: los métodos lógicos y los empíricos. Los primeros son todos aquellos que se basan en la utilización del pensamiento en sus funciones de deducción, análisis y síntesis, mientras que los métodos empíricos, se aproximan al conocimiento del objeto mediante su conocimiento directo y el uso de la experiencia, entre ellos encontramos la observación y la experimentación”³.

Métodos teóricos

❖ Análisis y síntesis

Con este método no sólo se relacionan hechos aparentemente aislados y se formula una teoría que unifica los diversos elementos, sino también se distinguen los elementos de un fenómeno y se procede a revisarlos ordenadamente por separados. Consiste en la reunión racional de varios elementos dispersos en una nueva totalidad y en la extracción de las partes de un todo, con el objetivo de estudiarlas y examinarlas por separado. Se usó este método con el objetivo de conocer los fundamentos y teorías relacionadas con el objeto de estudio.

❖ Modelación

Es justamente el método mediante el cual se crean abstracciones con vistas a explicar la realidad. En el modelo se revela la unidad de lo objetivo y lo subjetivo. La modelación es el método que opera en forma práctica o teórica con un objeto, no en forma directa, sino utilizando cierto sistema intermedio, auxiliar, natural o artificial. Se

^{3,4} **OCHOA G., ANA B.** Métodos. *Monografías*. [En línea] [Citado el: 5 de Febrero de 2008.] <http://www.monografias.com/trabajos11/methods/methods.shtml>.

utilizó este método para estructurar teóricamente desde el análisis y el diseño los módulos, casos de usos, diagramas, etc.

❖ **Análisis histórico lógico**

Está vinculado al conocimiento de las distintas etapas de los objetos en su sucesión cronológica, para conocer la evolución y desarrollo del objeto o fenómeno de investigación se hace necesario revelar su historia, las etapas principales de su desenvolvimiento y las conexiones históricas fundamentales. “Mediante el método histórico se analiza la trayectoria concreta de la teoría, su condicionamiento a los diferentes períodos de la historia. Los métodos lógicos se basan en el estudio histórico poniendo de manifiesto la lógica interna de desarrollo y de su teoría; hallando el conocimiento más profundo de su esencia.”⁴ Se puso en práctica este método con el objetivo de conocer la evolución de las teorías relacionadas con la rehabilitación y la existencia de sistemas automatizados con ese fin.

Herramientas y Procedimientos

Para el desarrollo del sistema a implementar se usaron las siguientes herramientas:

- ❖ Visual Paradigm 5.3
- ❖ Visual C++ .NET 2003
- ❖ SceneToolkit 2.4
- ❖ 3D Studio Max
- ❖ Plugins 3DMax
- ❖ Adobe Photoshop CS

El contenido de este trabajo se estructura en cuatro capítulos:

En el **Capítulo I. *Fundamentación teórica***: se realiza un estudio del estado del arte, mostrándose el resultado de una investigación bibliográfica sobre el objeto de estudio. Se muestran ejemplos de aplicaciones existentes vinculadas al tema que se trata.

El **Capítulo II. Características del Sistema:** se describen las metodologías y tecnologías a utilizar en el desarrollo de la aplicación, analizando sus características, ventajas y desventajas, seleccionando las mejores propuestas para el trabajo. También se describe el negocio del objeto de estudio a través del Modelo del Negocio.

El **Capítulo III. Análisis y Diseño del Sistema:** se realiza un estudio y análisis del sistema a desarrollar, se definen y describen detalladamente los requisitos y casos de uso del sistema. Además del análisis y diseño del sistema se muestra el diagrama de clases del análisis. Se exponen los detalles relacionados con el diseño del sistema.

El **Capítulo IV. Implementación:** trata sobre la elaboración de la solución apoyándose en los diagramas de clases explicados en el capítulo anterior, se plantean los principios para la implementación de la arquitectura. Se detallan y explican las funcionalidades definidas anteriormente.

Capítulo 1: Fundamentación Teórica

1.1 Introducción

En el presente capítulo se brinda una visión general de los aspectos relacionados con el objeto de estudio. Se informa además sobre la descripción de los principales conceptos asociados al problema que son necesarios para un buen entendimiento del negocio. Se hace un estudio de las principales teorías, estudios e investigaciones acerca de la ambliopía y el estrabismo así como de los sistemas relacionados con el objeto de estudio que se han desarrollado.

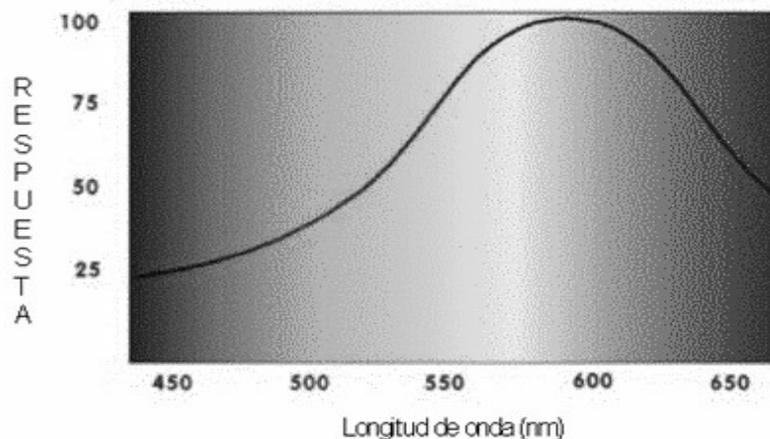
Se explica también de forma detallada las funciones que caracterizan las aplicaciones informáticas dentro del campo de acción, las tecnologías, procedimientos y herramientas que se proponen en el desarrollo del sistema así como una descripción de la propuesta de solución al problema planteando anteriormente en este documento.

1.2 Teorías física-ópticas y fisiológicas del proceso perceptivo

La vista es el sentido más importante para el hombre, puesto que la mayor parte de la información sobre el mundo circundante se obtiene a través de los receptores fotosensibles que se encuentran en el ojo. El desarrollo científico alrededor del año 1600 condujo a una serie de descubrimientos importantes. Se hablaba de las zonas o lugares donde se concebía la imagen visual y en 1604 el astrónomo alemán Juan Kepler, llegó a la conclusión de que ésta se formaba en la retina.

Más adelante el desarrollo de las ciencias físicas, anatómico – fisiológicas y neurológicas, permitieron demostrar que las imágenes se formaban en el cerebro y que la retina era la responsable de la transformación de la longitud de onda (impulso luminoso) en energía nerviosa. También que la lente o cristalino permitía aglutinar las ondas para dirigirlas hacia los puntos de la retina, produciéndose el fenómeno de la acomodación.

En el hombre existen limitaciones objetivas y fisiológicas que solamente posibilitan percibir visualmente una sección del espectro, desde los 400 hasta 750 nanómetros (nm), siendo la curva mayor de sensibilidad de 505 nm, correspondiente según el proceso de aprendizaje social al color verde amarillo. La correspondencia existente entre los colores del espectro y sus longitudes de onda, no así los matices de los colores por la dificultad subjetiva de su definición, aproximadamente es la siguiente:



Azul: 430nm

Verde - Amarillo: 505nm

Verde: 535nm

Rojo: 610nm

Figura 1 Correspondencia entre colores del espectro y longitud de onda

1.3 Coordinator-Bangester: una herramienta usada en Cuba

El proceso de rehabilitación visual del escolar con estrabismo y ambliopía se desarrolla sobre los procesos rehabilitativos de Pleóptica y Ortóptica, en los cuales uno de los principios de trabajo se basa en el desarrollo de actividades que permitan la convergencia visual, para ello se apoyan en la utilización del equipo oftalmológico Coordinator-Bangester que trabaja en esta dirección.

El Coordinator-Bangester es un equipo técnico que se caracteriza por:

- ❖ Ser un equipo con el cual debe interactuar el escolar mediante las orientaciones del especialista oftalmológico.
- ❖ Las tareas a las cuales se enfrenta el escolar corresponden a las etapas del desarrollo de su visión así como de su patología.
- ❖ Las tareas que tiene diseñada determinan actividades de localización de focos luminosos en un ambiente de dibujos animados no correspondiente a la realidad de los animados cubanos.

Fundamentación Teórica

- ❖ Unas de las tareas es la localización de fotos lumínicas (referido a lo anterior) de forma manual, estática que; aunque sin duda han fungido correctamente en el desarrollo de la fijación en los escolares estrábicos y ambliopes manifiesta dificultades técnicas.
- ❖ La otra tarea que incluye el localizador es el trazado de laberintos predeterminado y pintados en una placa metálica que forma parte de los test del equipo.
- ❖ Posee un lápiz eléctrico que permite al escolar trazar un camino sobre la placa y una vez que sobrepasa lo trazado con pintura y se desvía del recorrido, el contacto metálico del lápiz con la parte de la placa no pintada ejecuta un sonido eléctrico que avisa la realización incorrecta de la tarea.
- ❖ La estructura del equipo es sencilla, sin embargo sus componentes tienden a dejar de funcionar fácilmente lo que trae consigo que no se pueda utilizar en ninguno de los tratamientos oftalmológicos.
- ❖ Cada equipo poseía un costo inicial de 9,900 USD en el año 1990 cuando se iniciaron las investigaciones.

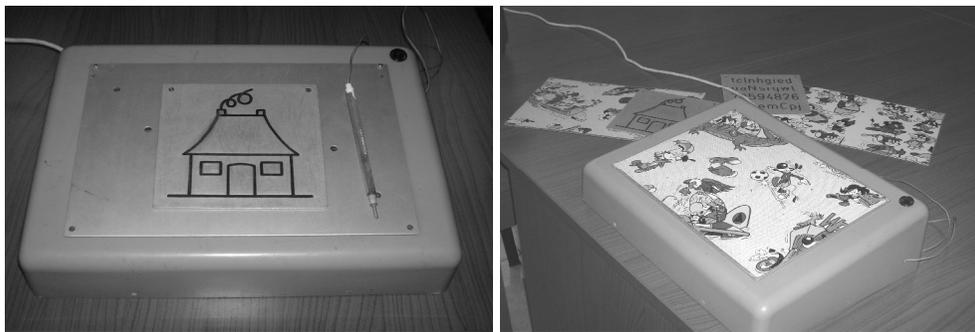


Figura 2 Equipo Coordinator-Bangester

Estos equipos además de no aportar desde el punto de vista psicológico a la motivación en el paciente presentan problemas con los insumos necesarios para su reparación. Es precisamente en esta dirección donde se buscan alternativas que generen gradientes motivacionales elevados que pongan al niño en condiciones óptimas para el proceso de desarrollo visual, y la informática se acerca a esta solución.

Desarrollar y ampliar la utilización de las computadoras en las escuelas, como herramienta tecnológica con una finalidad esencialmente pedagógica, orientadora del “saber saber” y del “saber hacer”, con el objeto de contribuir con el mejoramiento de la calidad de la educación; que permita a la persona mediante la comprensión de los códigos de las nuevas tecnologías, entender el mundo en que vive, adaptarse activamente a la sociedad y conscientes de que el conocimiento aquí y ahora, es dinamizador del crecimiento y herramienta fundamental para el cambio y la transformación social.

1.4 Investigaciones realizadas en Centros de Rehabilitación Visual

A partir del desarrollo de la teoría en las ciencias oftalmológicas y pedagógicas que confluyen en los tratamientos objeto de estudio; se han realizado diferentes estudios nacionales e internacionales con niños tanto de la edad preescolar como escolar.

Uno de los trabajos realizados fue “Resultados de la rehabilitación visual en ambliopes del Centro Oftalmológico Infantil”⁵ el cual es atendido directamente por el Hospital Oftalmológico Docente “Ramón Pando Ferrer”. Esta investigación se desarrolló con los objetivos de determinar las etiologías más frecuentes de las ambliopías, analizar la influencia de la fijación en la rehabilitación visual y evaluar la efectividad del tratamiento de rehabilitación. Fue de gran importancia tanto en el diagnóstico precoz como en los tratamientos en los períodos más sensibles del desarrollo visual.

En la búsqueda de referencia sobre otros trabajos realizados, más vinculados con el nuestro se encuentra “Método alternativo en el tratamiento de la ambliopía, trastornos de la visión binocular o ambos”⁶. En él se realizó un estudio longitudinal y descriptivo a un grupo de alumnos ambliopes, con trastornos de la visión binocular o ambos, pertenecientes a la Escuela de Rehabilitación Visual “Josué País” de Santiago de Cuba durante el curso escolar 1998-1999 a los cuales se les aplicó el sistema corrector pedagógico computarizado como alternativa y complemento de los métodos convencionales de tratamiento.

⁵ “Resultados de la Rehabilitación Visual en Ambliopes del Centro Oftalmológico Infantil”. 2002. Revista Cubana Oftalmológica, pág. 15.

⁶ FERNANDEZ GONZALEZ, M.E., y otros. 1999. “Método alternativo en el tratamiento de la ambliopía, trastornos de la visión binocular o ambos”. s.l. : Santiago de Cuba, 1999.

El empleo de la computación en estos niños mejoró considerablemente la agudeza visual promedio y la determinación de los potenciales evocados visuales constituyó un método eficaz para diagnosticar la ambliopía orgánica en aquellos cuya visión fue irrecuperable.

En el artículo “EVO: Sistema informático de entrenamiento visual para personas deficientes visuales”⁷ se analizan el desarrollo y evaluación del sistema informático EVO que, en el ámbito de la estimulación visual, permite la evaluación de capacidades visuales y facilita la aplicación de programas de entrenamiento visual, basados en juegos de ordenador adaptados a las características visuales del usuario.

En este artículo se afirma que la fotoestimulación es uno de los ámbitos en los que se ha hecho uso de los ordenadores en el contexto de la estimulación visual y plantean: “Los monitores de los ordenadores son una fuente de luz fácilmente adaptable a las condiciones ambientales y a diferentes tipos de usuarios. Además, el hecho de que las imágenes en pantalla se creen activando secuencialmente sus distintas partes, puede utilizarse como fuente de estimulación rítmica para los ojos. Por lo que los monitores se han convertido en la principal herramienta de algunos de los tratamientos más recientes en el ámbito de la ambliopía”.

La computación incluye juegos vinculados al estudio que motivan al niño y facilitan el restablecimiento de las funciones visuales para una mejor rehabilitación y aprendizaje, de ahí la importancia de formar en ellos el interés por el estudio; de modo que amplíen sus conocimientos y habilidades. Lo que nos permite corroborar como la influencia controlada de las radiaciones emitidas por los monitores computarizados, en el momento que el niño está realizando las actividades en este equipo puede favorecer el desarrollo de su visión.

1.5 Aplicaciones para el tratamiento de niños estrábicos y ambliopes

En el mundo existen iniciativas para tratar las afecciones visuales en niños estrábicos o ambliopes, las cuales hacen uso de la informática como herramienta para lograr la

⁷ RODRIGUEZ SOLER, J.J., y otros. 2001. “EVO: Sistema informático de entrenamiento visual para personas deficientes visuales”. 2001. págs. 5-15. Vol. 36.

rehabilitación de estos trastornos, que afectan a los infantes en los primeros estadios de vida.

Entre las compañías que se destacan en este campo está Visual Training, empresa de informática médica; dedicada específicamente al diseño y desarrollo de software para la rehabilitación visual. Una de sus herramientas más utilizadas es TOP VISION, el cual es un software para el entrenamiento de la visión, que usan numerosos oftalmólogos en sus clínicas y consultorios.

Otro programa especializado en problemas visuales es el INCISOFT Básico, el cual contiene programas con síntesis de voz (voz artificial que se asemeja a la voz humana) y magnificación que facilitan a personas ciegas y con baja visión el manejo en el computador de operaciones como procesamiento de textos, cálculos matemáticos, navegación en la Web, lectura de libros y consulta de palabras del idioma inglés en español.

Estas aplicaciones son comercializadas a altos precios, los cuales pueden llegar a más de 200 USD, lo cual no está al alcance de todos, pues los especialistas encargados de manejarlos también cobran altas tarifas por sus consultas.

Cuba posee un creciente desarrollo en el campo de las TICs, por la voluntad política de nuestro gobierno. Esto ha posibilitado un auge en la confección de software para diferentes sectores de la vida económica y social del país, entre los que se destaca el sistema educacional, muchos son los ejemplos que lo ilustran: la colección El Navegante, Multisaber, Feria de las Matemáticas, Problemas Matemáticos, etc. No obstante, en el campo de la Educación Especial esto no se ha comportado igual, pues a pesar que existen programas como el Jaws, el cual permite a los invidentes una total independencia en la computadora, no hay un software para la rehabilitación visual de niños ambliopes y estrábicos.

El desarrollo de una aplicación de este tipo se hace necesario, ya que el déficit y deterioro del equipamiento de las escuelas para estrábicos y ambliopes ha impedido una mayor celeridad en la compensación del defecto visual de sus alumnos, por lo que ha sido necesario utilizar otras variantes con la consecuente dilación en tiempo del tratamiento.

Con este software que nos proponemos desarrollar queremos darle solución a gran parte de todas estas problemáticas anteriormente planteadas pues esta herramienta además de colaborar en el proceso de rehabilitación visual servirá como un medio educativo en el proceso de enseñanza.

1.6 Tecnología y Software

La educación está en estos momentos sufriendo continuos cambios, descubriendo nuevos métodos para la enseñanza y el aprendizaje de sus educandos como son el uso de las nuevas TICs.

Los medios audiovisuales han pasado rápidamente de utopía a una palpable realidad. Repensar su uso óptimo y utilidad didáctica, perfeccionar las relaciones pedagógicas en el PDE resulta cada vez más necesario, debido a que las transformaciones que se han venido presentando en el ámbito de la educación han orientado el crecimiento y la diversidad de opciones educativas para la formación y actualización de los estudiantes y los profesores, la capacitación para el trabajo, la recreación y la utilización del tiempo libre. El uso de los medios audiovisuales rebasa el espacio del aula y trascienden el entorno de la escuela.

Existen variados criterios acerca de las funciones que se le atribuyen a los medios audiovisuales en el contexto del proceso enseñanza-aprendizaje, entre ellas están:

- ❖ **Función informativa:** Permite el estudio de la realidad a que se hace referencia y la describe lo más objetivamente posible.
- ❖ **Función motivadora:** Pretenden suscitar emociones y afectos, estimular el estudio y la búsqueda de nuevos conocimientos. Influye en la voluntad de los destinatarios, mediante la emotividad.
- ❖ **Función lúdica:** Pretende el goce del aprendizaje mediante el juego, fundamentalmente en los primeros grados.
- ❖ **Función investigativa:** Contribuye a incentivar la necesidad de encontrar nuevas alternativas para el conocimiento científico.
- ❖ **Función controladora:** Posibilita utilizarlos para el control del aprendizaje logrado por los estudiantes durante todo el proceso educativo.

- ❖ **Función estimulativa y rehabilitativa de funciones:** Pretende posibilitar que los escolares con determinadas disfunciones encuentren una alternativa más para su desarrollo o estimulación, a la vez que genera, a partir de su modelación, las vías idóneas para el desarrollo de su discapacidad.⁸

Las informaciones que llegan hoy a la escuela por medio de la TV, el video y la computación promueven el aprendizaje diferente en los niños y jóvenes. Es responsabilidad del maestro aprovechar de forma positiva y eficaz las potencialidades de estas tecnologías en la escuela ya que contribuyen nuevos estímulos para el desarrollo del proceso de enseñanza-aprendizaje. Un profesor creativo, práctico y proactivo considerará los medios audiovisuales e informáticos como sus aliados, útiles para el desarrollo del PDE y de herramienta de inestimable valor para hacer realidad la Revolución en la Educación que estamos comenzando.

La inserción del software educativo contribuye a que el estudiante interactúe con información proveniente de diferentes fuentes como: textos, gráficos, video, animaciones, fotografías, tablas, esquemas, mapas conceptuales y ejercicios. Todos ellos combinados hacen posible el desarrollo de habilidades intelectuales generales (observación, comparación, clasificación, valoración) que se manifiestan en el incremento de los procesos de análisis, síntesis, generalización y abstracción, como bases del pensamiento dirigido a penetrar en las esencias de las relaciones entre hechos y fenómenos.

Los software educativos pueden contribuir con el trabajo correctivo-compensatorio, es por ello que los beneficios de su utilización en los niños con necesidades educativas especiales se multiplican y constituyen un recurso de grandes posibilidades educativas, enriquecen su aprendizaje, acentúan sus fortalezas y eliminan el sentido del fracaso en los escolares⁹.

Una de las principales causas que se plantean en los "Estudios de Permanencia y Rehabilitación de escolares estrábicos y ambliopes" (1997-1998), que inciden en que

⁸ **PEREZ MARTINTO, P.C. 2005.** "La rehabilitación visual de escolares ambliopes con fijación central". *Documento de trabajo.UCI*. Ciudad de La Habana : s.n., 2005.

⁹ **PARLÁ MALEM, AMINE. 2004.** "Los Software Educativos en la Atención Psicoterapéutica a niños con Trastornos Emocionales y de la Conducta". *Trabajo de Diploma*. Ciudad de La Habana : s.n., Junio de 2004.

Fundamentación Teórica

los escolares permanecieran un promedio de 3 a 4 años en la escuela, es el estado actual de los equipos oftalmológicos no sustituidos, por esta razón no vemos esta tecnología solo como un medio para desarrollar los conocimientos sino también como un apoyo al proceso de rehabilitación que se realiza en el consultorio oftalmológico.

Para ello los software educativos deben cumplir con determinados aspectos: localización de objetos en el espacio, sus relaciones y formas de los objetos, uso del colorido, la multimedia, posibilidades para desarrollar la creatividad e independencia, posibilidades para vincular el contenido del software con la realidad y con el currículo escolar.

Es por esto que la computación constituye unos de los medios fundamentales de tratamiento, el diseño de software educativo ha tomado en consideración las necesidades educativas y los problemas visuales de estos alumnos, lo que ha contribuido a que muchos de estos niños se reincorporen, con el desarrollo visual apropiado, a su escuela primaria general.

De ahí, que en esta propuesta, se plantea una alternativa para que el maestro aproveche la posibilidad del desarrollo de la informática que tenemos hoy y la emplee como una vía de estimular el desarrollo de la agudeza visual de niños ambliopes con fijación central.

Lo importante es posibilitar que los niños ambliopes y estrábicos compensen sus necesidades y desarrollen sus potencialidades. Es por ello que, el software se puede incorporar al listado de alternativas destinadas a la rehabilitación visual de estos.

Capítulo 2: Características del Sistema

2.1 Introducción

En el presente capítulo, se hace un análisis de las tecnologías que pueden ser adecuadas para llevar a cabo el sistema que se pretende desarrollar. En algunos casos se hacen comparaciones que fundamentarán la propuesta final. Además se especificará la propuesta de solución al problema científico planteado anteriormente y los requisitos de esta propuesta.

También se enumeran los requisitos funcionales y no funcionales que debe tener el sistema que proponemos, lo que permite hacer una concepción general de este. Se describen los procesos del negocio que tienen que ver con el objeto de estudio.

2.2 Tendencias y Tecnologías actuales a considerar

La gran variedad tecnológica actual ya sea de hardware o de software brinda una solución efectiva tanto a grandes empresas que buscan una máxima seguridad, robustez y flexibilidad de programación, como en pequeños centros que necesitan sistemas económicos y de fácil uso. El reto, entonces, es encontrar una solución que garantice las necesidades de los centros de rehabilitación para estrámbicos y ambliopes y que requiera la menor cantidad de esfuerzo en su desarrollo, implantación y uso.

2.2.1 Metodologías de desarrollo

Para el desarrollo de un software debe tenerse en cuenta la necesidad de que la forma de trabajo sea organizada. Debe contarse con un proceso que ordene e integre las múltiples etapas del desarrollo, por lo que se necesita un método común. Para esto debe garantizar:

- ❖ Una guía para organizar las actividades del equipo de desarrollo.
- ❖ Una estrategia para ordenar las tareas de los desarrolladores por separado y del equipo como un todo.
- ❖ Una especificación de los **artefactos** que se deben desarrollar.
- ❖ Principios para establecer el control y la medición de los productos y actividades a desarrollar.

Características del Sistema

La ingeniería de software tiene varios modelos o paradigmas de desarrollo en los cuales se puede apoyar para la realización de software, de ellos podemos destacar los siguientes por ser los más utilizados y los más completos:

- ❖ Modelo en cascada o Clásico (modelo tradicional)
- ❖ Modelo en espiral (modelo evolutivo)
- ❖ Modelo de prototipos
- ❖ Desarrollo por etapas
- ❖ Desarrollo iterativo e incremental
- ❖ RAD (Rapid Application Development)

En el momento de seleccionar una metodología debemos preguntarnos si necesitamos una metodología ágil o una metodología guiada por un plan. Existen diferentes metodologías de desarrollo y las más usadas son:

- ❖ Rational Unified Process
- ❖ MSF for CMMI Process Improvement
- ❖ MSF Agile
- ❖ Scrum
- ❖ eXtreme Programming (XP)

El **Proceso Unificado de Rational** (RUP por sus siglas en inglés) es un proceso de desarrollo de software y junto con el **Lenguaje Unificado de Modelado** (UML por sus siglas en inglés), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. En realidad no es un proceso con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

Debido a esto RUP puede ser usado tanto como una metodología clásica o como un método ágil en dependencia de cómo lo adapte a su ambiente.

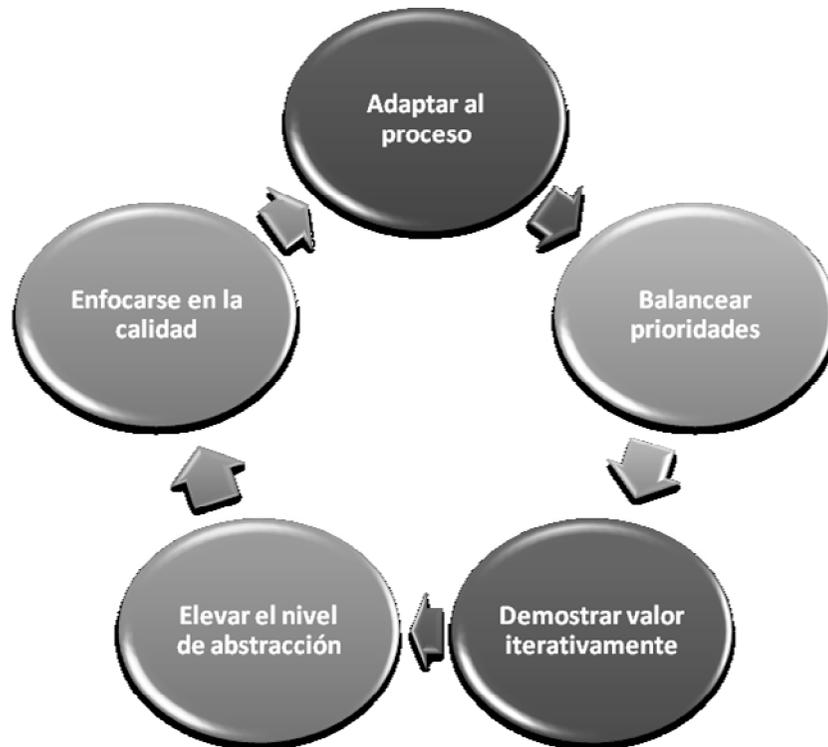


Figura 2. Principios del RUP

El RUP se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso incluyendo artefactos y roles. Sus principales características son:

- ❖ Forma disciplinada de asignar tareas y responsabilidades (Quién hace Qué, Cuándo y Cómo)
- ❖ Pretende implementar las mejores prácticas en Ingeniería de Software
- ❖ Desarrollo interactivo
- ❖ Administración de requisitos
- ❖ Uso de arquitectura basada en componentes
- ❖ Control de cambios
- ❖ Modelado visual del software
- ❖ Verificación de la calidad del software

Sin embargo el proceso descrito en el presente trabajo es un proyecto pequeño que debe ser entregado en un período corto de tiempo y que es desarrollado por pocas personas. Además es un proyecto donde la parte investigativa juega un papel importante por lo que no se tiene una visión clara de cómo han de implementarse las diferentes funcionalidades, y los requisitos pueden variar mucho en el tiempo de desarrollo. Por lo cual se optó por una versión ágil de RUP, dX.

El proceso dX es una versión totalmente dócil del RUP, está diseñado para aquellas personas deciden usar el RUP sin apartarse del **XP** (eXtreme Programming). Este proceso basado en la experiencia de otras metodologías ágiles, reduce los artefactos de RUP a los mínimos esenciales. Es considerada como un híbrido entre RUP y XP, ésta última considerada como una de las metodologías ágiles más populares, y de la que toma conceptos y prácticas como la programación en pares e incluir al cliente como parte fundamental del proyecto.

2.2.2 Unified Modeling Language

Unified Modeling Language (UML) es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema de software orientado a objetos.

UML ha puesto fin a las llamadas “guerras de métodos” que se han mantenido a lo largo de los 90, en las que los principales métodos sacaban nuevas versiones que incorporaban las técnicas de los demás. Con UML se fusiona la notación de estas técnicas para formar una herramienta compartida entre todos los ingenieros software que trabajan en el desarrollo orientado a objetos.

El Lenguaje Unificado de Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación.

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real.

2.2.3 Herramientas CASE

Las herramientas **CASE** son un conjunto de programas y ayudas que brindan ayuda y asistencia técnica a analistas, ingenieros de software y desarrolladores para el análisis de requisitos, modelado visual y documentación durante parte o todo el ciclo de vida de un proyecto de software.

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de modelado, demostraciones interactivas del lenguaje y proyectos UML.¹⁰

2.2.4 Entorno de Desarrollo Integrado

Microsoft Visual Studio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET; lo cual permite crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles.

2.2.4.1 Lenguaje de Programación

El C++ es un lenguaje de programación que abarca tres paradigmas de la programación: la programación estructurada, la programación genérica y la programación orientada a objetos. Las principales características del C++ son las

¹⁰ 2007. *Free Download Manager*. [En línea] 5 de Marzo de 2007. [Citado el: 20 de Diciembre de 2007.] [http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_\(M%C3%8D\)_14720_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p/).

Características del Sistema

facilidades que proporciona para la programación orientada a objetos y para el uso de plantillas o programación genérica (templates).

Además posee una serie de propiedades difíciles de encontrar en otros lenguajes de alto nivel:

- ❖ Posibilidad de redefinir los operadores (sobrecarga de operadores)
- ❖ Identificación de tipos en tiempo de ejecución (RTTI)
- ❖ C++ está considerado por muchos como el lenguaje más potente, debido a que permite trabajar tanto a alto como a bajo nivel.

2.2.4.2 SceneTollKit

SceneToolKit (STK) es la primera de las herramientas brindadas por el **Proyecto de Herramientas de Desarrollo para Sistemas de Realidad Virtual**, tiene 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. 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.¹¹

2.2.4.3 Plugins 3DMax

La Herramienta Básica para Sistemas de Realidad Virtual SceneToolKit permite, entre otras funciones, la carga y visualización de entornos creados en el 3DS Max, para lo cual han sido creados algunos **plugins** o macros para exportar ficheros legibles por la herramienta. Dicha exportación se hace desde el mismo 3DSMax.

Para el desarrollo de nuestro sistema usamos el plugin **px3DXExport.mcr** que no es más que un macro para la exportación de modelos en formato 3DX.

¹¹ 2007. SceneToolKit. Sistemas de Realidad Virtual. *Ayuda y Documentación*. UCI, Ciudad de La Habana : s.n., 2007.

2.2.5 Entorno de diseño

Para la realización del diseño del juego se utilizaron las siguientes herramientas:

2.2.5.1 Autodesk 3D Studio Max

Autodesk 3D Studio Max es un programa de creación de gráficos y animación 3D desarrollado por Autodesk Media & Entertainment. Fue desarrollado como sucesor para sistemas operativos Win32 del 3D Studio creado para DOS. Kinetix fue más tarde fusionada con la última adquisición de Autodesk, Discreet Logic.

3DS Max es uno de los programas de animación 3D más utilizados. Dispone de una sólida capacidad de edición, una omnipresente arquitectura de plugins y una larga tradición en plataformas Microsoft Windows. 3D Studio Max es utilizado en mayor medida por los desarrolladores de videojuegos, aunque también en el desarrollo de proyectos de animación como películas o anuncios de televisión, efectos especiales y en arquitectura.

2.2.5.2 Adobe Photoshop CS

Adobe Photoshop CS es una herramienta para la edición de imágenes, retoque fotográfico y creación de gráficos. Presenta un entorno completo para diseñadores y grafistas añadiendo un soporte para el trabajo con dibujos basados en gráficos vectoriales.

2.3 Nuestra propuesta

Teniendo en cuenta y tomando como base lo antes expuesto sobre los principales elementos, herramientas y procedimientos que se utilizarán y analizando las tecnologías disponibles se decidió utilizar el lenguaje C++ y Visual Studio para el desarrollo de una aplicación que permita aprovechar al máximo las ventajas de esta plataforma. Unido a esto se decidió implementar el desarrollo del sistema con la herramienta STK por las ventajas que nos brinda y el diseño se realizará con 3DSMax. Se decidieron estas herramientas por su fortaleza y por estar disponible en los servidores de la Universidad.

Es importante señalar que aunque el Visual Studio no es gratis, si lo son la Plataforma .NET y las versiones Express de Visual Studio, las cuales permiten el desarrollo de aplicaciones .robustas en .NET bajo ningún costo adicional.

Los planes de informatización de la sociedad junto a la migración a Software Libre así como la implantación en la UCI de tecnología de código abierto, eventualmente, obligará a las aplicaciones existentes que han sido desarrolladas y construidas en .NET a migrar hacia plataformas como Mono, la versión de código abierto de .NET. La solución aquí planteada puede ser migrada a esta plataforma de una manera segura, ya que aunque se han utilizado característica propias de la misma, mediante el uso de las librerías Boost de C++ se implementaría la migración del sistema ya que la comunicación entre los módulos, que es un aspecto fundamental para le funcionamiento del software, no utiliza elementos no estándar del lenguaje.

2.4 Descripción del negocio

Los centros de enseñanza especial de Ciudad Habana, a través del MSc. Pedro C. Pérez Martinto de la Universidad de las Ciencias Informáticas, propusieron la implementación de un software el cual fuera capaz de suplir las necesidades materiales de equipos de rehabilitación visual para escolares ambliopes y miopes que son atendidos en estos. Dicho producto debe basar su funcionamiento en el Coordinador Bangester, principal herramienta con que cuentan las escuelas especiales para el desarrollo de habilidades óculo manuales.

El Coordinador Bangester es un equipo rectangular, el cual tiene dos funciones fundamentales: fomentar la coordinación y entrenar la visión. Para realizar cada una de ellas posee una serie de implementos, muchos de los cuales no están integrados físicamente al equipo.

Para fomentar la coordinación, solamente utiliza una lámina de aluminio de forma rectangular, en la cual están pintados varios caminos, en los que el paciente (niño ambliope o estrábico) utilizando un puntero de aluminio debe tratar de realizarlos sin salirse de las líneas que lo describen.

El entrenamiento de la visión por otra parte, utiliza una serie de bombillos ubicados en el equipo y un grupo de láminas con dibujos que poseen orificios, por los cuales traspasa la luz de los bombillos. El tratamiento consiste en que el paciente debe

Características del Sistema

señalar, con el ojo bueno tapado, dónde parpadea uno de estos bombillos poniendo su dedo encima.

La propuesta de solución es el desarrollo de una sistema informático que, basado en el funcionamiento del Coordinator Bangester, será una combinación de las dos funciones del mismo, creando en un juego terapéutico el cual será un laberinto donde el paciente con la utilización del mouse o teclado recogerá una serie de objetos o figuras que aparecerán aleatoriamente en el contexto, lográndose la función coordinativa por la utilización del mouse o teclado para seguir un camino y así como la función rehabilitadora visual porque se le obliga al paciente realizar todo el proceso con el ojo bueno tapado.

En todo este proceso intervendrá el especialista o doctor encargado del tratamiento, el cual es el que comprobará si la terapia está dando resultados a través de una serie de reportes que debe ser capaz de dar el sistema.

2.5 Reglas de negocio

Las reglas del negocio son especificaciones que nos describen lo que se debe cumplir o aquellas condiciones que se deben satisfacer; describen la política del negocio. A continuación se explicarán los conceptos asociados al Modelo del Negocio.

- ❖ **Paciente:** Persona con padecimientos visuales (estrabismo o ambliopía) que se someterá al tratamiento.
- ❖ **Técnico:** Persona encargada del funcionamiento del sistema. Es quien registrará los pacientes en el sistema y configurará el(los) tratamiento(s) que estos recibirán.
- ❖ **Especialista:** Persona que se encarga de informarle al técnico el tipo de tratamiento que recibirá el paciente.
- ❖ **Solicitud de Tratamiento:** Petición por parte de un especialista al técnico para que se le aplique al paciente.
- ❖ **Tratamiento:** Proceso que se le aplica al paciente.
- ❖ **Juego:** Proceso que realizará el paciente en el tratamiento
- ❖ **Reporte:** Datos que se generan a partir de uno o varios tratamientos.

Características del Sistema

- ❖ **Listado de Registro de Pacientes:** Listado de pacientes que ha recibido al menos un tratamiento.
- ❖ **Listado de Registro de Tratamientos:** Listado de los tratamientos realizados a los pacientes registrados
- ❖ **Listado de Registro de Reportes:** Listado de los reportes realizados.

También se debe tener en cuenta los siguientes aspectos:

- ✓ Los pacientes no tiene permisos para acceder a la aplicación, sólo el técnico tendrá acceso a este.
- ✓ Los reportes que solicite el especialista los realizará el técnico.
- ✓ No se puede iniciar el juego del tratamiento si el paciente no está registrado.
- ✓ No deben existir reportes de tratamientos que no se han realizado.

2.6 Requerimientos del Sistema

Los requerimientos de un sistema definen las cualidades, funcionalidades y características que debe poseer un sistema de software. Estos se clasifican en dos tipos: *Requerimientos Funcionales* y *Requerimientos No Funcionales*.

2.6.1. Requerimientos Funcionales

De acuerdo con los objetivos planteados el sistema debe ser capaz de:

RF	Requisitos Funcionales	Requisitos Asociados
R.1	Gestionar el registro de paciente.	R.1.1-Crear un registro de paciente. R.1.2-Modificar un registro de paciente. R.1.3-Eliminar un registro de paciente.
R.2	Mostrar reportes de un paciente o de un grupo de pacientes determinados.	R.2.1-Mostrar los reportes en tablas o en gráficos. R.2.2-Crear un reporte. R.2.3-Comparar reportes. R.2.4-Eliminar reporte. R.2.5-Cargar datos del tratamiento.
R.3	Ejecutar el juego terapéutico.	R.3.1-Cargar datos del juego.

<p>R.4 Gestionar el juego terapéutico</p>	<p>R.4.1-Crear el juego terapéutico.</p> <p>R.4.1.1-Permitir escoger mapa de juego.</p> <p>R.4.1.2-Permitir escoger el paciente a tratar.</p> <p>R.4.1.3-Permitir escoger cantidad de minutos a jugar.</p> <p>R.4.1.4-Permitir escoger objetos a recoger.</p> <p>R.4.1.5- Permitir ejecutar el juego terapéutico.</p> <p>R.4.2-Los registros de juego de un paciente deben guardarse para utilizarse para realizar los reportes.</p> <p>R.4.2.1-Se guardaran los siguientes datos: puntuación, cantidad de objetos recogidos, fecha y tiempo de juego.</p> <p>R.4.3-Guardar datos del juego.</p> <p>R.4.4.-Guardar datos del tratamiento.</p>
<p>R.5 Gestionar el tratamiento del paciente.</p>	<p>R.5.1- Crear el mundo virtual.</p> <p>R.5.2- Cargar los modelos realizados en la herramienta 3D StudioMax.</p> <p>R.5.3- Mostrar mensajes al paciente.</p> <p>R.5.4- Mostrar los datos del paciente que está jugando.</p> <p>R.5.4.1-Mostrar la calificación que va obteniendo el paciente.</p> <p>R.5.4.2-Mostrar el tiempo de juego.</p> <p>R.5.5-Mostrar el tiempo que le queda por jugar al paciente.</p> <p>R.5.6-Permitir al paciente recoger los objetos en el mundo virtual.</p> <p>R.5.7- Colocar aleatoriamente los objetos en el mundo virtual.</p> <p>R.5.8-Permitir al paciente recorrer el</p>

mundo virtual.

Tabla 1 Requisitos Funcionales

2.6.2. Requerimientos No Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Se dividen en categorías o tipos los cuales se muestran a continuación:

Requisitos de apariencia o interfaz externa.

- ✓ El producto deberá ser fácil de usar y legible, de manera que el usuario pueda moverse dentro de la aplicación de forma intuitiva. Deberá usar términos y conceptos comunes entre el tipo de usuarios hacia donde está dirigida. Además tendrá un diseño sencillo, permitiendo que no sea necesario mucho entrenamiento para utilizar el sistema.

Requisitos de Software.

- ✓ El sistema deberá correr sobre Window98 o superior.

Requisitos de Hardware.

- ✓ Se necesita que la PC donde corra la aplicación posea una memoria RAM de 256 MB o superior y procesador de 1.9 GHz o superior para una mejor visualización de los entornos virtuales.

Restricciones en el diseño y la implementación.

- ✓ Las herramientas de desarrollo serán 3D Studio Max 8.0 y Visual Studio 2003.
- ✓ Lenguaje de programación a ser usado para la implementación será C++.
- ✓ Se debe utilizar las librerías del framework SceneToolkit versión 2.4, el cual es una herramienta creada por la Facultad 5 para el desarrollo de juegos.
- ✓ El sistema será desarrollado como una aplicación de escritorio.

- ✓ La herramienta CASE será Visual Paradigm 5.3.

Usabilidad.

- ✓ El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora.

Seguridad

- ✓ Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.
- ✓ Verificación sobre acciones irreversibles (eliminaciones).

2.7 Actores y trabajadores del negocio

ACTORES DEL NEGOCIO	JUSTIFICACIÓN
Especialista	Persona especializada en trastornos visuales y es la que dirige el proceso de rehabilitación.
Paciente	Persona con problemas visuales, específicamente que padece de estrabismo o ambliopía.

Tabla 2 Actores del Negocio

TRABAJADOR DEL NEGOCIO	JUSTIFICACIÓN
Técnico	Persona encargada de realizar las actividades del negocio.

Tabla 3 Trabajadores del Sistema

2.8 Patrones de Casos de Uso utilizados.

Un patrón es una solución aplicable a un problema que se presenta una u otra vez en el desarrollo de distintas aplicaciones en distintos contextos. En el desarrollo de nuestro sistema utilizamos el patrón CRUD (Creating, Reading, Updating and Deleting) este modela todas las operaciones que se pueden realizar sobre una parte de información, tal como crearla, visualizarla, buscarla, actualizarla y eliminarla.¹²

2.9 Descripción de los Casos de Uso del Modelo del Negocio

Los casos de uso le proporcionan a los analistas del sistema y clientes llegar a un acuerdo sobre los requerimientos del mismo.

2.9.1 Diagrama de Casos de Uso del modelo del negocio

Un diagrama de casos de uso del negocio representa gráficamente los procesos del negocio y su interacción con los actores del negocio.

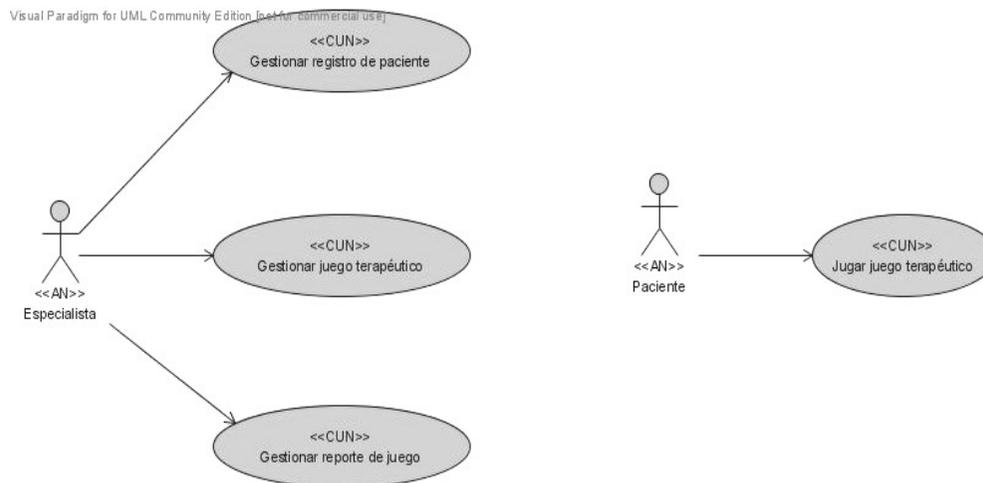


Figura 3 Diagrama de Casos de Uso del Negocio

¹² Entorno Virtual de Aprendizaje. *Conferencia 2 Arquitectura y Patrones de Diseño*. [En línea] [Citado el: 10 de Mayo de 2008.] <http://teleformacion.uci.cu/course/view.php?id=43>.

2.9.2. Especificación de los Casos de Uso y Diagramas de Actividades.

Los casos de uso del negocio, representan procesos que ocurren en el negocio. A continuación se explican los detalles de los casos de uso para su comprensión por parte de miembros del equipo de desarrollo y clientes. Se realizan además las especificaciones de los casos de uso del negocio y sus diagramas de actividades.

Un diagrama de Actividad demuestra la serie de actividades que deben ser realizadas en un caso de uso, así como las distintas rutas que pueden irse desencadenando en este. Un diagrama de actividad es utilizado en conjunción de un diagrama de caso de uso para entender cómo es utilizado el sistema y cómo reacciona en determinados eventos.

2.9.2.1 Caso de Uso “Gestionar registro de paciente”

Nombre:	Gestionar registro de paciente	
Actor del negocio:	Especialista(Inicia)	
Trabajador:	Técnico	
Precondiciones:	El especialista debe haber orientado un juego terapéutico como tratamiento a un paciente.	
Poscondiciones:	Se incluyó, modificó, vio o eliminó un Registro de Paciente.	
Resumen:	El caso de uso se inicia cuando el especialista se presenta a la consulta con la orientación de un juego terapéutico como tratamiento a un paciente determinado. El técnico gestiona el registro del paciente, o sea, introduce, modifica, ve o elimina los datos del paciente a realizar el juego. Seguidamente el técnico guarda los datos del registro en Listado de Registro de Pacientes, finalizando así el caso de uso.	
Flujo Básico		
Acción del actor	Respuesta del negocio	
1. El caso de uso se inicia cuando el especialista llega a la consulta y entrega los datos del paciente con		

tratamiento terapéutico al técnico.	
	2. El técnico recibe los datos del paciente.
	3. El técnico registra los datos del paciente en el Registro de Paciente.
	4. El técnico guarda el registro en el Listado de Registros de Pacientes.
	5. El técnico notifica al especialista el resultado de la operación.
6. El especialista verifica el Registro de Paciente.	
7. El caso de uso termina.	
Flujos Alternos	
6.a Los datos del paciente no se encuentran de manera correcta en el Registro de Paciente.	
Acción del actor	Respuesta del negocio
6.a.1 El especialista notifica al técnico que existen errores en los datos.	
	6.a.2 El técnico arregla los datos del paciente.
	6.a.3 Regresa al paso 4 del Flujo Básico.
Sección 1: Modificar Registro de Paciente	
Flujo Básico	
Acción del actor	Respuesta del negocio
1. El caso de uso se inicia cuando el especialista llega a la consulta y solicita rectificar los datos del paciente en el Registro de Paciente	
	2. El técnico recibe los nuevos datos del paciente.
	3. El técnico modifica los datos del paciente en el Registro de Paciente.
	4. El técnico guarda el registro en el Listado de Registros de Pacientes.

6.	El especialista verifica el Registro de Paciente.		
7.	El caso de uso termina.		
Sección 2: Eliminar Registro de Paciente			
Flujo Básico			
Acción del actor		Respuesta del negocio	
1.	El caso de uso se inicia cuando el especialista llega a la consulta y solicita eliminar el Registro de Paciente de un paciente determinado		
		2.	El técnico recibe los datos del paciente.
		3.	El técnico elimina el Registro de Paciente.
		4.	El técnico guarda el registro en el Listado de Registros de Pacientes.
5.	El especialista verifica el Listado de Registros de Pacientes.		
6.	El caso de uso termina.		
Flujo Alterno			
3.a Los datos del paciente no se encuentran en el Listado de Registro de Paciente.			
Acción del actor		Respuesta del negocio	
		3.a.1	El técnico notifica al especialista que no existe el Registro del Paciente.
3.a.2	Regresa al paso 5 del Flujo Básico.		
Sección 3: Ver Registro de Paciente			
Flujo Básico			
Acción del actor		Respuesta del negocio	
1.	El caso de uso se inicia cuando el especialista llega a la consulta y		

	solicita consultar el Registro de Paciente de un paciente determinado		
		2.	El técnico recibe los nuevos datos del paciente.
		3.	El técnico muestra el Registro de Paciente.
4.	El especialista verifica el Registro de Paciente.		
5.	El caso de uso termina.		
Flujo Alterno			
4.a Los datos del paciente no se encuentran en el Registro de Paciente.			
Acción del actor		Respuesta del negocio	
4.a.1	El especialista notifica al técnico que no existen los datos del paciente		
		4.a.2	El técnico registra los datos del paciente.
		4.a.3	Regresa al paso 3 del Flujo Básico.
4.b Los datos del paciente no se encuentran de manera correcta en el Registro de Paciente.			
4.b.1	El especialista notifica al técnico que existen errores en los datos.		
		4.b.2	El técnico arregla los datos del paciente.
		4.b.3	Regresa al paso 3 del Flujo Básico.
DIAGRAMAS DE ACTIVIDADES (Ver Anexo 1)			

2.9.2.2 Caso de Uso “Gestionar juego terapéutico”

Nombre:	Gestionar juego terapéutico
Actor del negocio:	Especialista(Inicia)
Trabajador:	Técnico
Precondiciones:	El paciente debe estar registrado en el Listado de Registros de Pacientes.
Poscondiciones:	Se incluyó, modificó, vio o eliminó un juego terapéutico.
Resumen: El caso de uso se inicia cuando el especialista se presenta a la consulta con la orientación de un juego terapéutico como tratamiento a un paciente determinado. El técnico gestiona el registro del juego, o sea, introduce, modifica, ve o elimina los datos del juego terapéutico a realizar por el paciente. Seguidamente el técnico guarda los datos del registro en Listado de Juegos, finalizando así el caso de uso.	
Flujo Básico	
Acción del actor	Respuesta del negocio
1. El caso de uso se inicia cuando el especialista llega a la consulta y entrega los datos del juego terapéutico a realizar por el paciente.	
	2. El técnico recibe los datos del juego y del paciente de lo realizará.
	3. El técnico verifica que exista el Registro del Paciente.
	4. El técnico registra los datos del juego en el Registro de Juego.
	5. El técnico guarda el registro en el Listado de Registros de Juegos.
	6. El técnico notifica al especialista el resultado de la operación.

7.	El especialista verifica el Registro de Juego.	
8.	El caso de uso termina.	
Flujos Alternos		
3.a	Los datos del paciente no se encuentran en el Listado de Registros de Pacientes.	
	Acción del actor	Respuesta del negocio
		3.a.1 El técnico notifica al especialista que no está registrado el paciente a realizar el juego.
		3.a.2 El técnico registra los datos del paciente.
		3.a.3 Regresa al paso 4 del Flujo Básico.
7.a	Los datos del juego no se encuentran de manera correcta en el Registro de Juego.	
7.a.1	El especialista notifica al técnico que existen errores en los datos.	
		7.a.2 El técnico arregla los datos del paciente.
		7.a.3 Regresa al paso 4 del Flujo Básico.
Sección 1: Eliminar Registro de Juego		
Flujo Básico		
	Acción del actor	Respuesta del negocio
1.	El caso de uso se inicia cuando el especialista llega a la consulta y solicita eliminar el Registro de un juego determinado.	
		2. El técnico recibe los datos del juego.
		3. El técnico elimina el Registro de Juego.
		4. El técnico guarda el registro en el Listado de Registros de Juegos.
5.	El especialista verifica el Listado de Registros de Juegos.	
6.	El caso de uso termina.	

Flujo Alterno			
3.a Los datos del juego no se encuentran en el Listado de Registros de Juegos.			
Acción del actor		Respuesta del negocio	
		3.a.1	El técnico notifica al especialista que no existe el Registro del Juego.
3.a.2	Regresa al paso 5 del Flujo Básico.		
Sección 2: Ver Registro de Juego			
Flujo Básico			
Acción del actor		Respuesta del negocio	
1.	El caso de uso se inicia cuando el especialista llega a la consulta y solicita consultar el Registro de un juego determinado		
		2.	El técnico recibe los datos del juego.
		3.	El técnico muestra el Registro de Juego.
4.	El especialista verifica el Registro de Juego.		
5.	El caso de uso termina.		
Flujo Alterno			
4.a Los datos del juego no se encuentran en el Registros de Juegos.			
Acción del actor		Respuesta del negocio	
4.a.1	El especialista notifica al técnico que no existen los datos del juego		
		4.a.2	El técnico registra los datos del juego.
		4.a.3	Regresa al paso 3 del Flujo Básico.
4.b Los datos del juego no se encuentran de manera correcta en el Registro de Juego.			
4.b.1	El especialista notifica al técnico que existen errores en		

	los datos.		
		4.b.2	El técnico arregla los datos del juego.
		4.b.3	Regresa al paso 3 del Flujo Básico.
DIAGRAMAS DE ACTIVIDADES (Ver Anexo 2)			

2.9.2.3 Caso de Uso “Gestionar reporte de juego”

Nombre:	Gestionar reporte de juego.
Actor del negocio:	Especialista(Inicia)
Trabajador:	Técnico
Precondiciones:	El juego terapéutico orientado un paciente como tratamiento debe haberse realizado.
Poscondiciones:	Se creó,mostró, comparó, o eliminó un reporte del juego terapéutico.

Resumen: El caso de uso se inicia cuando el especialista se presenta a la consulta con la solicitud de reporte de un juego terapéutico realizado por un paciente determinado. El técnico gestiona el reporte del juego terapéutico, o sea, crea,muestra, compara o elimina los datos del reporte del juego terapéutico realizado,finalizando así el caso de uso.

Flujo Básico

Acción del actor	Respuesta del negocio
1. El caso de uso se inicia cuando el especialista llega a la consulta y solicita el reporte de un juego terapéutico realizado por un paciente determinado dado un parámetro.	
	2. El técnico recibe los datos del paciente y el parámetro a reportar.
	3. El técnico busca el paciente en el Listado de Registros de Pacientes.

	4.	El técnico busca los Registros de Juegos que corresponden al paciente en el Listado de Registros de Juegos.
	5.	El técnico muestra al especialista el resultado de la operación.
6.		El especialista verifica el reporte.
7.		El caso de uso termina.
Flujos Alternos		
3.a Los datos del paciente no se encuentran en el Listado de Registros de Pacientes.		
Acción del actor		Respuesta del negocio
	3.a.1	El técnico notifica al especialista que no existe el Registro del paciente.
3.a.2		El especialista verifica el Listado de Registros de Pacientes
3.a.3		El caso de uso termina.
4.a No existen Registros de Juegos correspondientes al paciente.		
	4.a.1	El técnico notifica al especialista que no existen Registros de Juegos correspondientes al paciente.
4.a.2		El especialista verifica el Listado de Registros de Juegos
3.a.3		El caso de uso termina.
Sección 1: Comparar Reportes de Juegos		
Flujo Básico		
Acción del actor		Respuesta del negocio
1.	El caso de uso se inicia cuando el especialista llega a la consulta y solicita comparar los reportes de varios juegos terapéuticos realizado(s) por un(os) paciente(s) determinado(s) según un parámetro	

	establecido.		
		2.	Recibe los datos de el(los) paciente(s) y el parámetro establecido para la comparación.
		3.	El técnico busca el(los) pacient(s) en el Listado de Registros de Pacientes.
		4.	El técnico busca los Registros de Juegos que corresponden a el(los) paciente(s) en el Listado de Registros de Juegos.
		5.	El técnico compara los reportes según el parámetro establecido.
		6.	El técnico muestra al especialista el resultado de la operación.
6.	El especialista verifica la comparación de los reportes		
7.	El caso de uso termina.		

Flujos Alternos

3.a Los datos de uno o varios pacientes no se encuentran en el Listado de Registros de Pacientes

		3.a.1	El técnico notifica al especialista que no existen los Registro de determinados pacientes.
3.a.2	El especialista verifica el Listado de Registros de Pacientes		
3.a.3	El caso de uso termina.		

4.a No existen Registros de Juegos correspondientes uno o varios pacientes.

		4.a.1	El técnico notifica al especialista que no existen Registros de Juegos correspondientes a los pacientes.
4.a.2	El especialista verifica el Listado de Registros de Juegos		
3.a.3	El caso de uso termina.		

Sección 2: Eliminar Reporte de Juego

Flujo Básico			
Acción del actor		Respuesta del negocio	
1.	El caso de uso se inicia cuando el especialista llega a la consulta y solicita eliminar el reporte de juego de un paciente determinado.		
		2.	El técnico recibe los datos del paciente.
		3.	El técnico verifica el Registro del paciente.
		4.	El técnico elimina el Reporte de juego correspondiente al paciente .
		5.	El técnico notifica al especialista que se realizó la operación
6.	El caso de uso termina.		
Flujo Alterno			
3.a Los datos del paciente no se encuentran en el Listado de Registro de Paciente.			
Acción del actor		Respuesta del negocio	
		3.a.1	El técnico notifica al especialista que no existe el Registro del Paciente.
3.a.2	El caso de uso termina.		
4.a No existen ningún reporte asociado al paciente.			
		3.a.1	El técnico notifica al especialista que no existe ningún reporte asociado al paciente.
3.a.2	El caso de uso termina.		
DIAGRAMAS DE ACTIVIDADES (Ver Anexo 3)			

2.9.2.4 Caso de Uso “Jugar juego terapéutico”

Nombre:	Jugar juego terapéutico
Actor del negocio:	Paciente(Inicia)
Trabajador:	Técnico
Precondiciones:	El juego terapéutico orientado un paciente como tratamiento debe haberse orientado por el especialista
Poscondiciones:	Se jugó el juego terapéutico.
Resumen: El caso de uso se inicia cuando el paciente se presenta a la consulta con la orientación del juego terapéutico. El técnico gestiona iniciará el juego terapéutico para que el paciente juegue,finalizando así el caso de uso.	
Flujo Básico	
Acción del actor	Respuesta del negocio
1. El caso de uso se inicia cuando el paciente llega a la consulta y presenta la orientación del juego terapéutico.	
	2. El técnico recibe la orientación del juego terapéutico y los datos del paciente.
	3. El técnico busca el paciente en el Listado de Registros de Pacientes.
	4. El técnico busca el Registro del juego en el Listado de Registros de Juegos.
	5. El técnico inicia el juego.
6. El paciente juega el juego terapéutico.	
7. El caso de uso termina.	
Flujos Alternos	
3.a Los datos del paciente no se encuentran en el Listado de Registros de Pacientes.	

Acción del actor	Respuesta del negocio
	3.a.1 El técnico notifica que no existe el Registro del paciente.
	3.a.2 El técnico registra los datos del paciente.
	3.a.3 Regresa al paso 4 del Flujo Básico.
4.a No existen Registros de Juegos correspondientes al paciente.	
	4.a.1 El técnico notifica que no existen Registros de Juegos correspondientes al paciente.
	4.a.2 El técnico crea el Registro del juego.
	4.a.3 Regresa al paso 5 del Flujo Básico.
DIAGRAMAS DE ACTIVIDADES (Ver Anexo 4)	

2.10 Diagrama de Clases del Modelo de Objetos

El diagrama de clases, como artefacto que se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos.

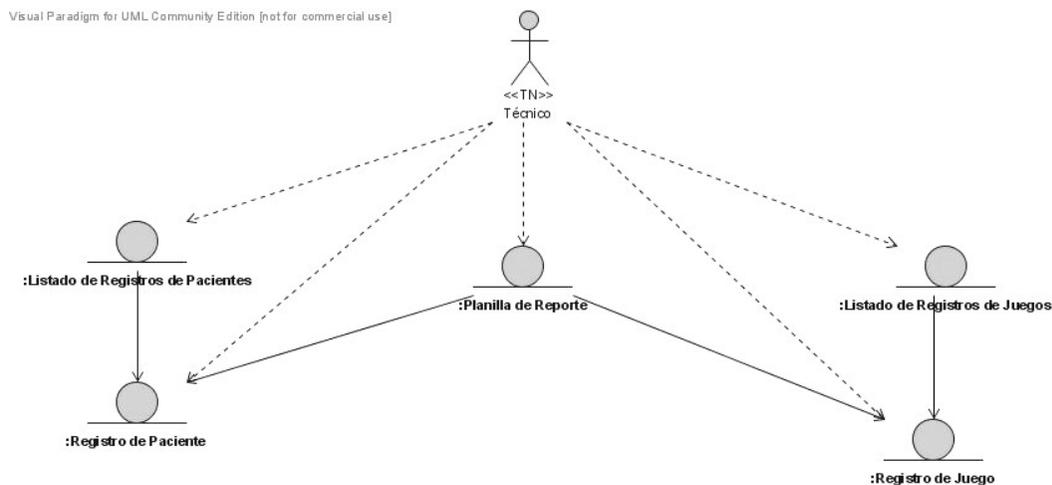


Figura 4 Diagrama de Clases del Modelo de Objetos del Negocio

Capítulo 3: Análisis y Diseño del sistema

3.1 Introducción

En el presente capítulo se hace la descripción de la propuesta que trae este trabajo, para ello se especificarán los módulos definidos en el diseño del sistema así como sus características esenciales, permitiendo además identificar mediante un Diagrama de Casos de Uso las relaciones de los actores que interactúan con el sistema, y las secuencias de acciones con las que interactúan. Se muestran además las principales funcionalidades y características del sistema definidas en los procesos de Análisis y Diseño.

3.2 Descripción del sistema propuesto

Para cumplir los objetivos propuestos anteriormente y teniendo en cuenta todos los requerimientos planteados, el sistema que se propone se ha dividido en dos módulos lógicos principales *Manage* y *Game*. Además el sistema registra los datos de cada juego realizado por los pacientes en los diferentes tratamientos que reciba para generar al final los reportes solicitados.

3.2.1 Módulo Manage.

Este módulo se encarga de la gestión de los registros que serán manejados por el software (Pacientes, Juegos y Reportes), brindando las funcionalidades siguientes:

- ❖ **Registrar Paciente:** permite el registro de los datos de cada paciente (Nombre y apellidos, CI, Edad, Padecimiento Visual, Tiempo de Tratamiento).
- ❖ **Modificar Paciente:** permite el cambio de datos de los pacientes.
- ❖ **Eliminar Paciente:** permite eliminar un paciente determinado del Registro de Pacientes. Esta funcionalidad eliminará los Juegos realizados por el paciente que se hayan registrado y los Reportes de estos.
- ❖ **Registrar Juego:** permite registrar un nuevo juego terapéutico registrando los datos de este (Paciente, Tiempo, Mapa).
- ❖ **Eliminar Juego:** permite eliminar un juego determinado.
- ❖ **Crear Reporte:** permite crear un reporte de un juego determinado.
- ❖ **Visualizar Reporte:** visualiza los reportes creados mediante tablas.

- ❖ **Comparar Reportes:** compara dos o más reportes seleccionados.
- ❖ **Eliminar Reporte:** permite eliminar un reporte determinado.

Es el encargado además de la configuración del software y mediante él se realiza el intercambio de información con el usuario. Permite al técnico todo el trabajo con la persistencia de los datos mediante las funcionalidades:

- ❖ **Salvar:** permite guardar los datos de los pacientes registrados así como de los juegos terapéuticos realizados por estos.
- ❖ **Cargar:** permite cargar los archivos con los datos de pacientes y juegos realizados anteriormente.

Este módulo también se encarga de ejecutar el juego terapéutico, a través de la funcionalidad Ejecutar, la cual muestra el listado de juegos creados y le da la opción de al técnico de ejecutar uno. Después de haberse ejecutado el juego seleccionado se salvarán los datos en un archivo binario con el siguiente formato:

- ❖ Datos del paciente.
- ❖ Mapa.
- ❖ Tiempo de juego.

3.2.1.1 Sistema de Reportes

Para crear el reporte se escogerá el paciente comprobando que haya recibido al menos un tratamiento. Se creará una lista con todos los tratamientos del paciente seleccionado y se generará un reporte.

3.2.2 Módulo Game

Se encarga de gestionar el tratamiento del paciente y de implementar todos los requisitos relacionados con el juego. Carga el archivo temporal binario que posee los datos del juego y que constituye el componente que garantiza la comunicación entre este módulo y el Manage, así como de salvar otro archivo binario con los datos del tratamiento efectuado al cerrarse la aplicación del juego.

Las principales funcionalidades que se implementan en este módulo son:

- ❖ **Crear Mundo**

- ❖ Recoger Objeto
- ❖ Recorrer Mundo
- ❖ Entrar y salir de la habitación
- ❖ Mostrar mensajes

3.3 Actor del sistema

ACTOR DEL SISTEMA	JUSTIFICACIÓN
Técnico	Persona encargada de interactuar con el sistema. Es quien tiene los permisos necesarios para el acceso a los datos gestionados por el software.

Tabla 4 Actor del Sistema

3.4 Modelo de Casos de Uso del Sistema

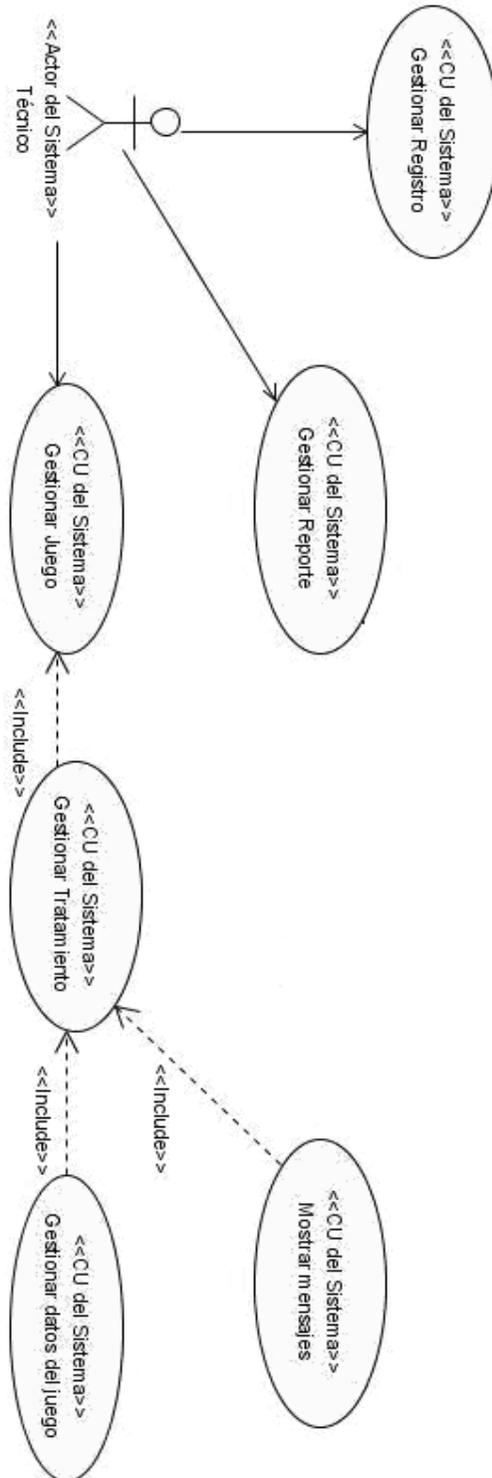
Para representar o modelar los requisitos de un sistema desde la perspectiva del usuario la forma más simple y efectiva es mediante el modelado de los CU. Los CU son utilizados para representar el funcionamiento de un sistema o negocio, o cómo desea el usuario que funcione el sistema a crear. No es una aproximación a la *Orientación a Objetos* (OO) sino una manera de modelar los procesos que intervienen de una forma u otra en el sistema. Siendo estos una manera muy buena de tratar el análisis de los sistemas orientados a objetos ya que son de forma general el punto de partida del análisis orientado a objetos con UML.

En esta sesión se presentan los casos de uso identificados para satisfacer los requerimientos funcionales del sistema:

3.4.1 Casos de Uso del Sistema

Un caso de uso representa una descripción de la secuencia de eventos o interacciones que se producen entre un actor y el sistema, cuando el actor usa el sistema para llevar a cabo una tarea específica. Los casos de uso del sistema son:

1. Gestionar Registro
2. Gestionar Reporte
3. Gestionar Juego
4. Gestionar Tratamiento
5. Gestionar Datos del Juego
6. Mostrar Mensajes



Visual Paradigm for UML Community Edition [not for commercial use]

Figura 5 Diagrama de Casos de Uso del Sistema

3.5 Casos de Uso del Sistema y Diagramas de Secuencia.

Mediante la *especificación de los casos de uso* se describe en detalle el funcionamiento de cada uno de ellos. A continuación se muestra la explicación de cada caso de uso, representando sus *diagramas de secuencia* correspondientes.

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada método de la clase. Contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y muestra la secuencia de mensajes intercambiados entre los objetos para llevar a cabo la funcionalidad descrita por el escenario. Los diagramas de secuencia se utilizan con frecuencia para validar los casos de uso. Documentan el diseño desde el punto de vista de los casos de uso.

Nombre del caso de uso	Gestionar Registro	
Actor	Técnico.	
Propósito	Gestionar los registros de los pacientes.	
Referencias	R1,R1.1,R1.2,R1.3	
Registrar Paciente		
Resumen: El Caso de Uso se inicia cuando el actor solicita registrar un paciente.		
Poscondición	Se registró el paciente y se mostró en el Registro de Pacientes.	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1- Se solicita registrar paciente donde se pasan los valores que serán registrados (Nombre y apellidos, CI, Edad,	1.1- Se guardan los datos del paciente (Nombre y apellidos, CI, Edad, Padecimiento, Tiempo de Tratamiento)	

Padecimiento, Tiempo de Tratamiento)	
	1.2- Se muestran los datos en el Registro de Pacientes (Nombre y apellidos, CI, Edad, Padecimiento, Tiempo de Tratamiento).
Curso alternativo	
	1.1- Si el paciente ya existe muestra mensaje de error "Error al registrar el paciente. Compruebe los datos".
Prioridad: Primario	
Modificar Paciente	
Resumen: El Caso de Uso se inicia cuando el actor solicita modificar los datos de un paciente.	
Precondición	Que el paciente esté registrado
Poscondición	Los datos del paciente se actualizaron
Curso normal de los eventos	
Acción del actor	Respuesta del sistema
1- Se solicita modificar los datos del paciente donde se pasan los nuevos valores que serán registrados.	1.1- Se actualizan y guardan los datos del paciente
	1.2- Se muestran los datos en el Registro de Pacientes.
Curso alternativo:	
	1.1- Si el paciente no existe muestra mensaje de error "Compruebe sus

		datos, el paciente no existe.”
Prioridad: Secundario		
Eliminar Paciente		
Resumen: El Caso de Uso se inicia cuando el actor solicita eliminar un paciente.		
Precondición	Que el paciente esté registrado	
Poscondición	Se eliminó el paciente	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1- Se solicita eliminar el paciente donde se identificará este.	1.1- Se busca el paciente y se elimina.	
	1.2- Se actualiza el Registro de Pacientes.	
Curso alterno		
	1.2- Si el paciente no existe muestra mensaje de error “Compruebe sus datos, el paciente no existe.”	
Prioridad: Secundario		

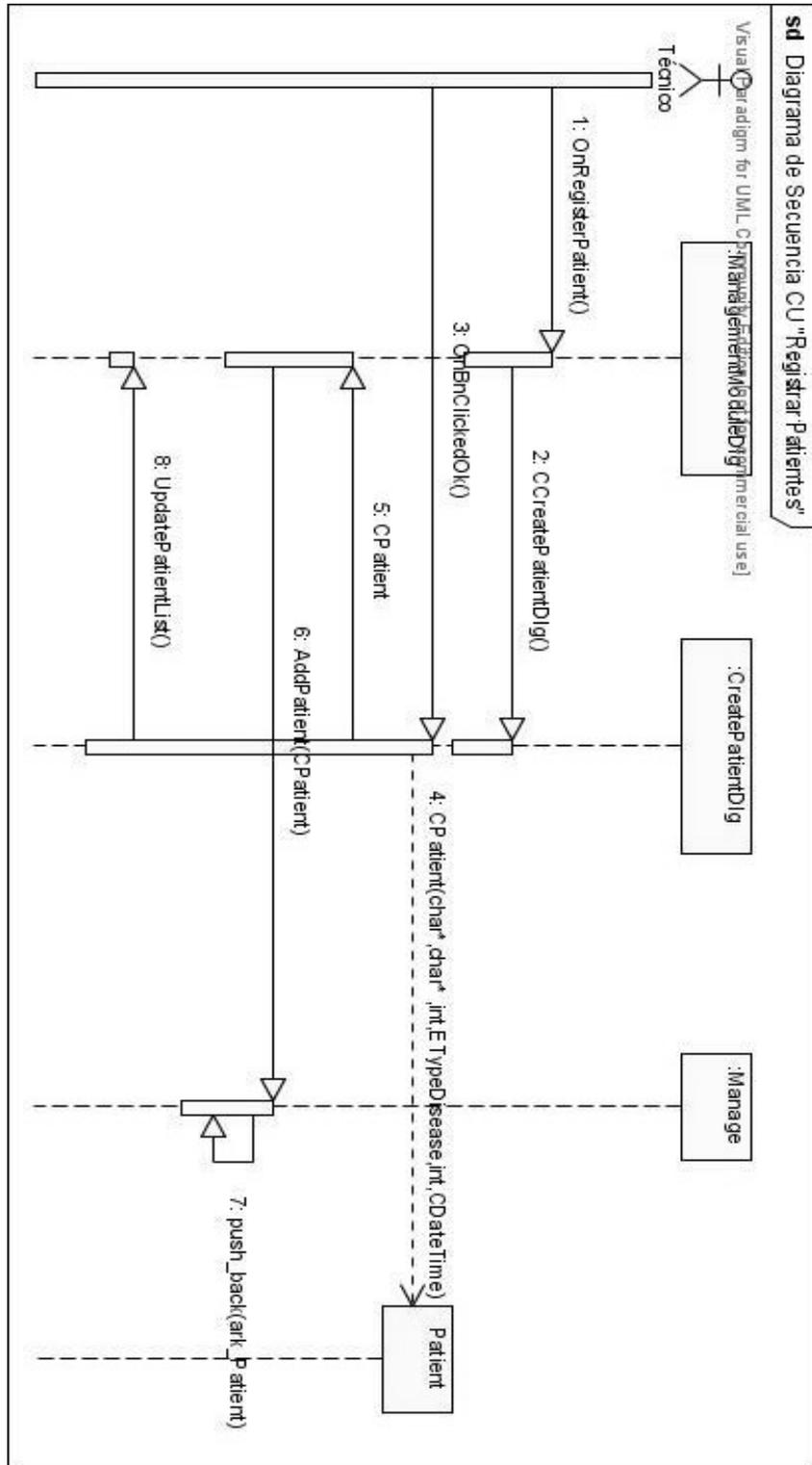


Figura 6 Diagrama de Secuencia CU "Gestionar Registro" Sección: Registrar Paciente

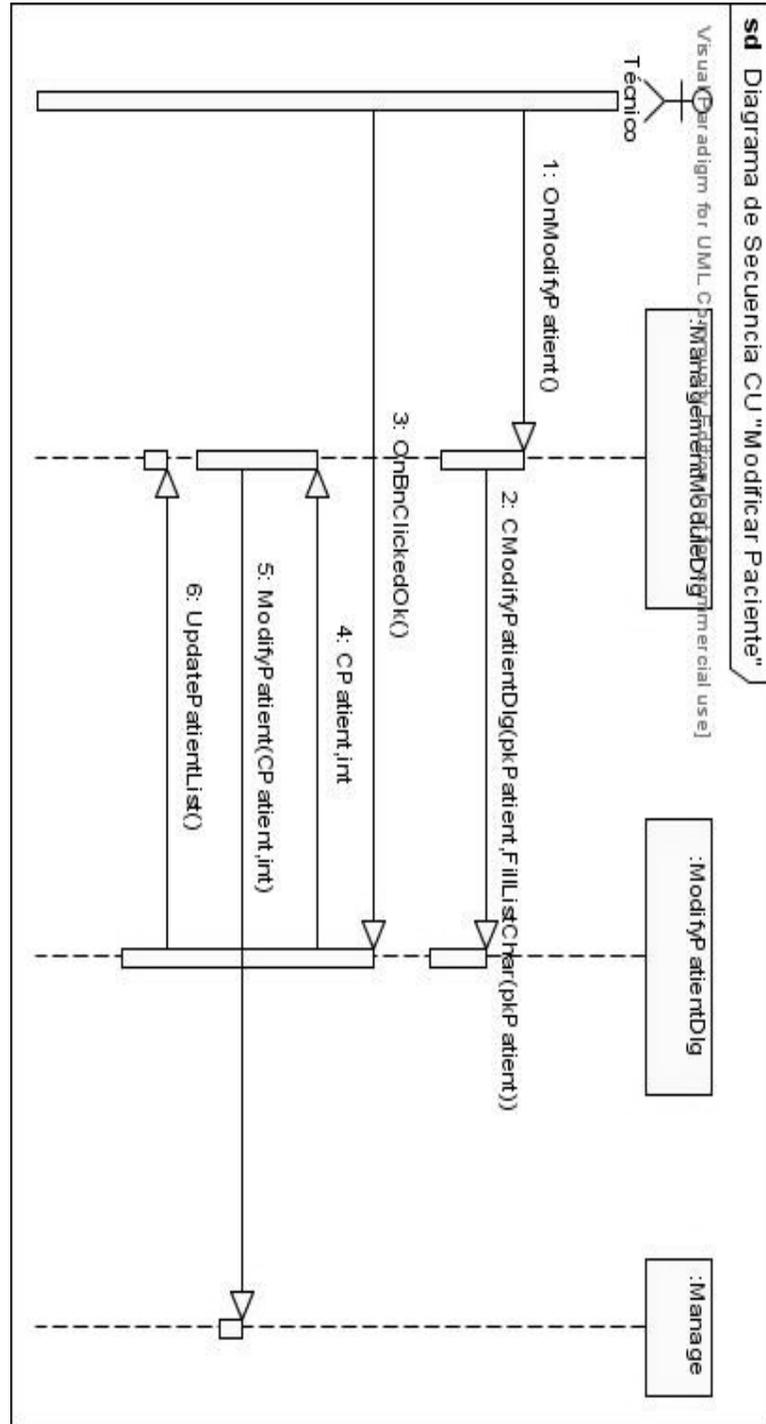


Figura 7 Diagrama de Secuencia CU "Gestionar Registro" Sección: Modificar Paciente

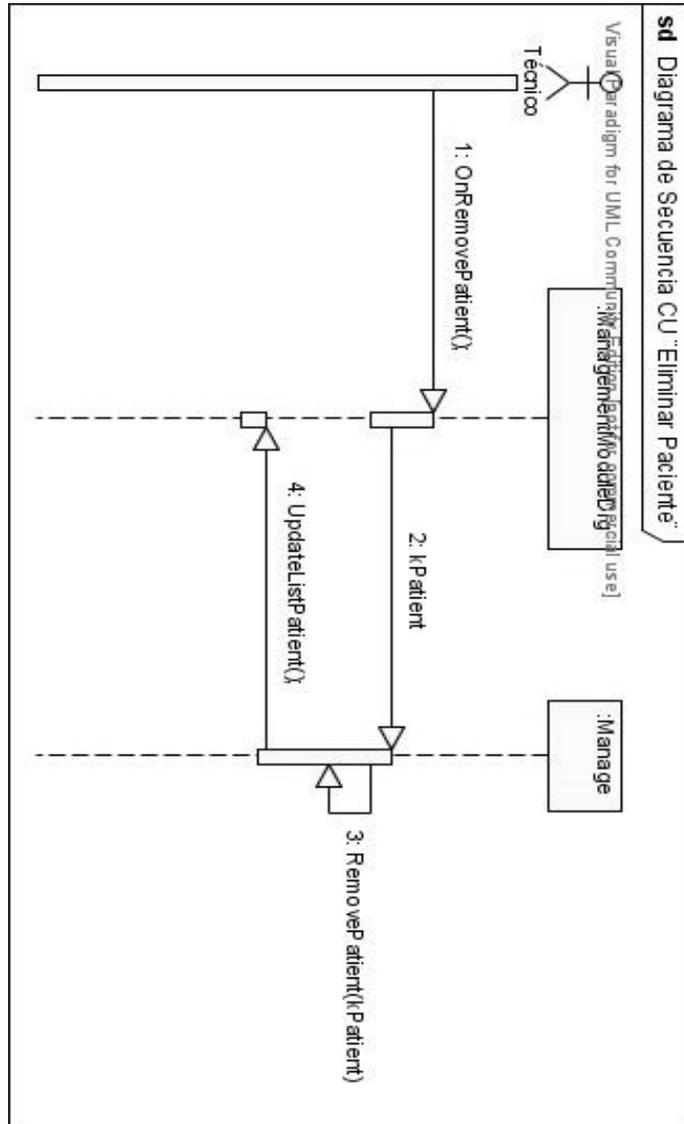


Figura 8 Diagrama de Secuencia CU "Gestionar Registro" Sección: Eliminar Paciente

Nombre del caso de uso	Gestionar Reporte	
Actor	Técnico.	
Propósito	Gestionar los reportes de los tratamientos realizados por los pacientes.	
Referencias	R2,R2.1,R2.2,R2.3,R2.4	
Crear Reporte		
Resumen: El Caso de Uso se inicia cuando el actor solicita registrar un reporte.		
Precondición	El paciente al que se le realizará el reporte debe estar registrado y debe haber realizado al menos un tratamiento.	
Poscondición	El reporte se registra y debe mostrarse en el Registro de Reportes.	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1- Se solicita crear un reporte especificando el paciente y el parámetro a reportar.		
	1.1- Se crea el reporte de los tratamientos realizados por el paciente	
Curso alterno		
	1.1- Si el paciente no hay realizado ningún tratamiento muestra mensaje de error "El paciente no ha realizado ningún tratamiento".	
Prioridad: Secundario		

Comparar Reportes	
Resumen: El Caso de Uso se inicia cuando el actor solicita la comparación entre varios reportes	
Precondición	Que los reportes estén registrados
Poscondición	Se mostrará la comparación entre los reportes
Curso normal de los eventos	
Acción del actor	Respuesta del sistema
1- Se solicita la comparación entre reportes definiendo cuáles se compararán y con respecto a qué aspecto(Tiempo jugado, Puntuación, Objetos Capturados)	
	1.1- Se realiza la comparación entre los reportes
	1.2- Se muestran el resultado de la comparación.
Curso alterno	
	1.1- Si no se especifica el parámetro a comparar muestra mensaje de error."Debe especificar el parámetro de comparación"
	1.2- Si no existen los reportes muestra mensaje de error."No existen los reportes"
Prioridad: Secundario	
Eliminar Reporte	

Resumen: El Caso de Uso se inicia cuando el actor solicita eliminar un reporte.	
Precondición	Que el reporte esté registrado
Poscondición	Se eliminó el reporte
Curso normal de los eventos	
Acción del actor	Respuesta del sistema
1- Se solicita eliminar un reporte donde se identificará este.	
	1.1- Se busca el reporte y se elimina.
	1.2- Se actualiza el Registro de Reportes.
Curso alterno	
	1.2- Si el reporte no existe muestra mensaje de error. "No existen el reporte"
Prioridad: Secundario	

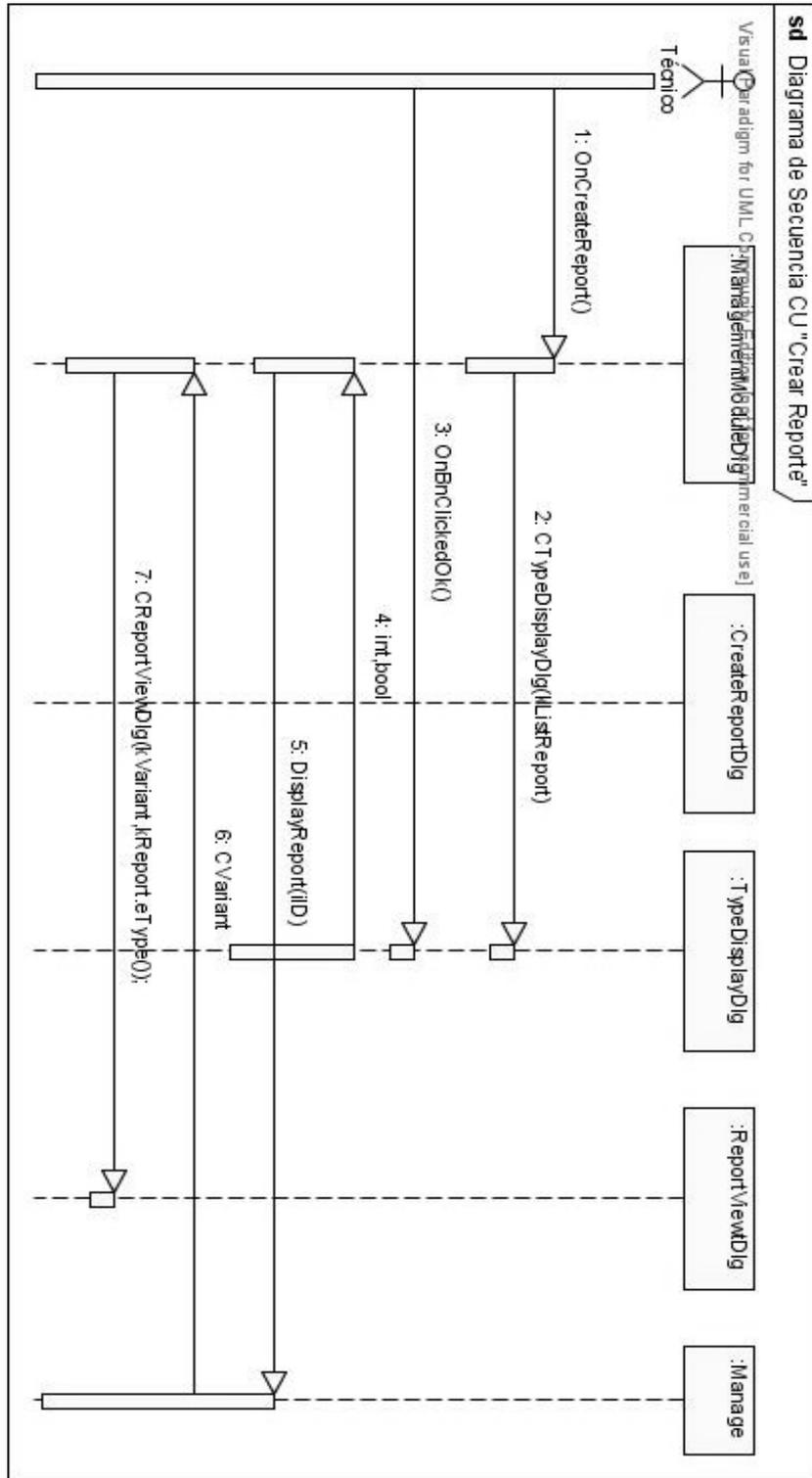


Figura 9 Diagrama de secuencia CU “Gestionar Reporte” Sección: Crear Reporte

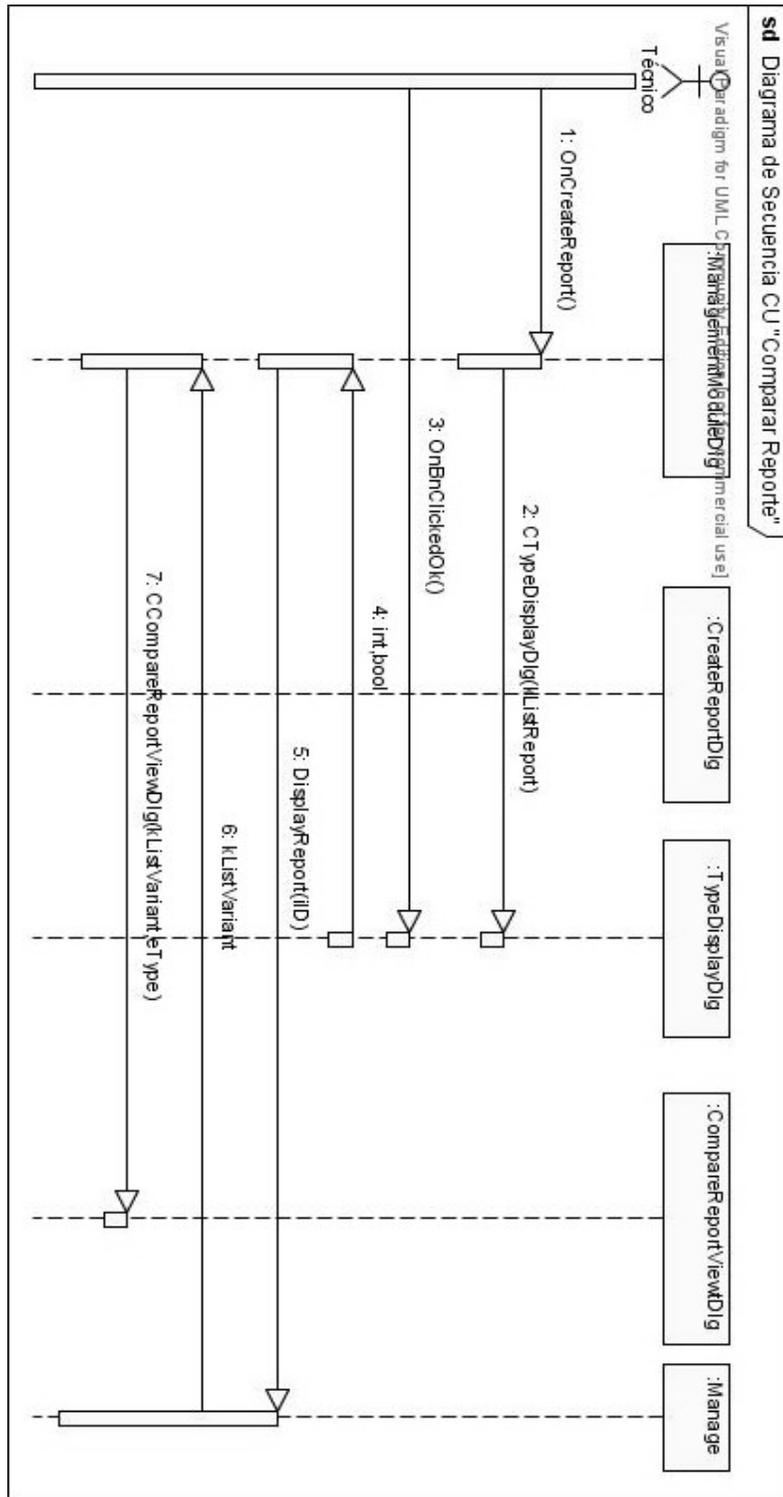


Figura 10 Diagrama de secuencia CU "Gestionar Reporte" Sección: Comparar Reportes

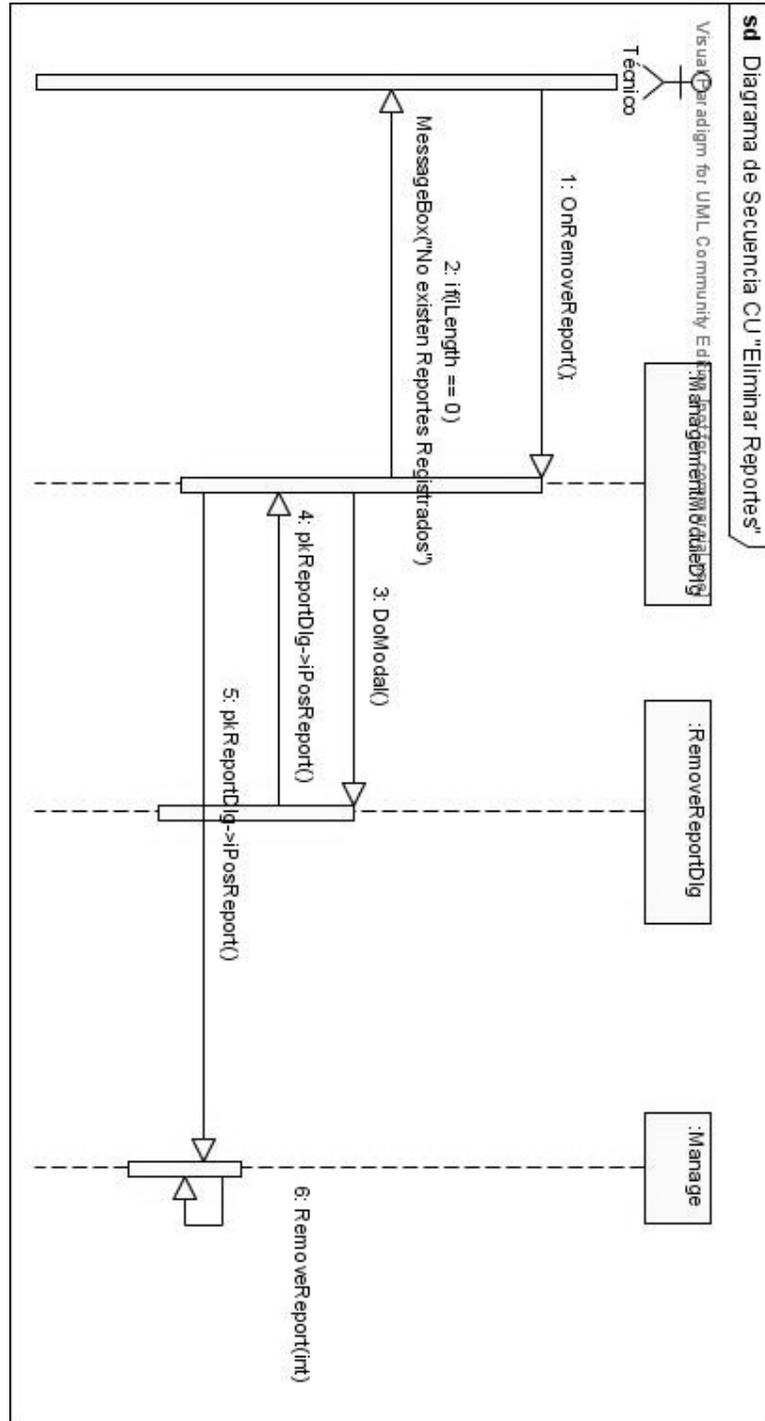


Figura 11 Diagrama de secuencia CU "Gestionar Reporte" Sección: Eliminar Reporte

Nombre del caso de uso	Gestionar Juego	
Actores	Técnico.	
Propósito	Gestionar los reportes de los tratamientos realizados por los pacientes.	
Referencias	R3,R4,R4.1,R4.1.1,R4.1.2,R4.1.3,R4.1.5,R4.2	
Crear Juego		
Resumen: El Caso de Uso se inicia cuando el actor solicita registrar un reporte.		
Precondición	El paciente que realizará el juego debe estar registrado.	
Poscondición	Se creará el juego.	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1- Se solicita crear un juego especificando el paciente y los datos del juego(Mapa y Tiempo de juego)		
	1.1- Se crea el juego con los parámetros especificados.	
	1.2- Se inicia el juego.	
Curso alterno		
	1.1- Si el paciente no está registrado muestra mensaje de error "El paciente no está registrado"	
	1.2- Si se especifican incorrectamente los datos del juego muestra mensaje de error. "Debe escoger un mapa" "El tiempo de	

		tratamiento es de 15 a 30 minutos”.
Prioridad: Secundario		
Eliminar Juego		
Resumen: El Caso de Uso se inicia cuando el actor solicita eliminar un juego.		
Precondición	Que el juego esté registrado	
Poscondición	Se eliminó el juego	
Curso normal de los eventos		
Acción del actor	Respuesta del sistema	
1- Se solicita eliminar un juego donde se identificará este.		
	1.1- Se busca el juego y se elimina.	
	1.2- Se actualiza el Registro de juegos.	
Curso alterno		
	1.2- Si el juego no existe muestra mensaje de error.”No existe el juego”	
Prioridad: Secundario		

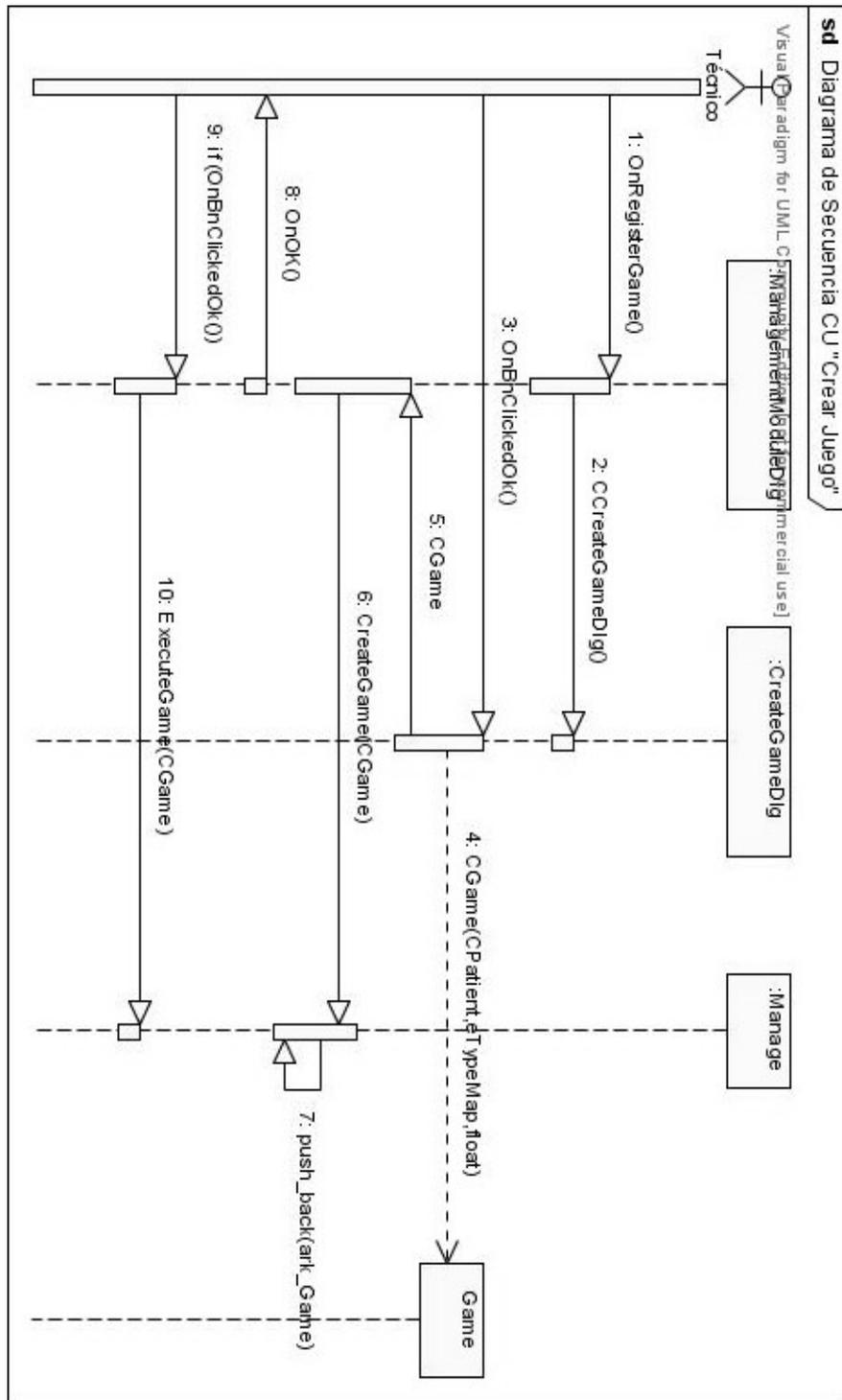


Figura 12 Diagrama de Secuencia CU "Gestionar Juego" Sección: Crear Juego

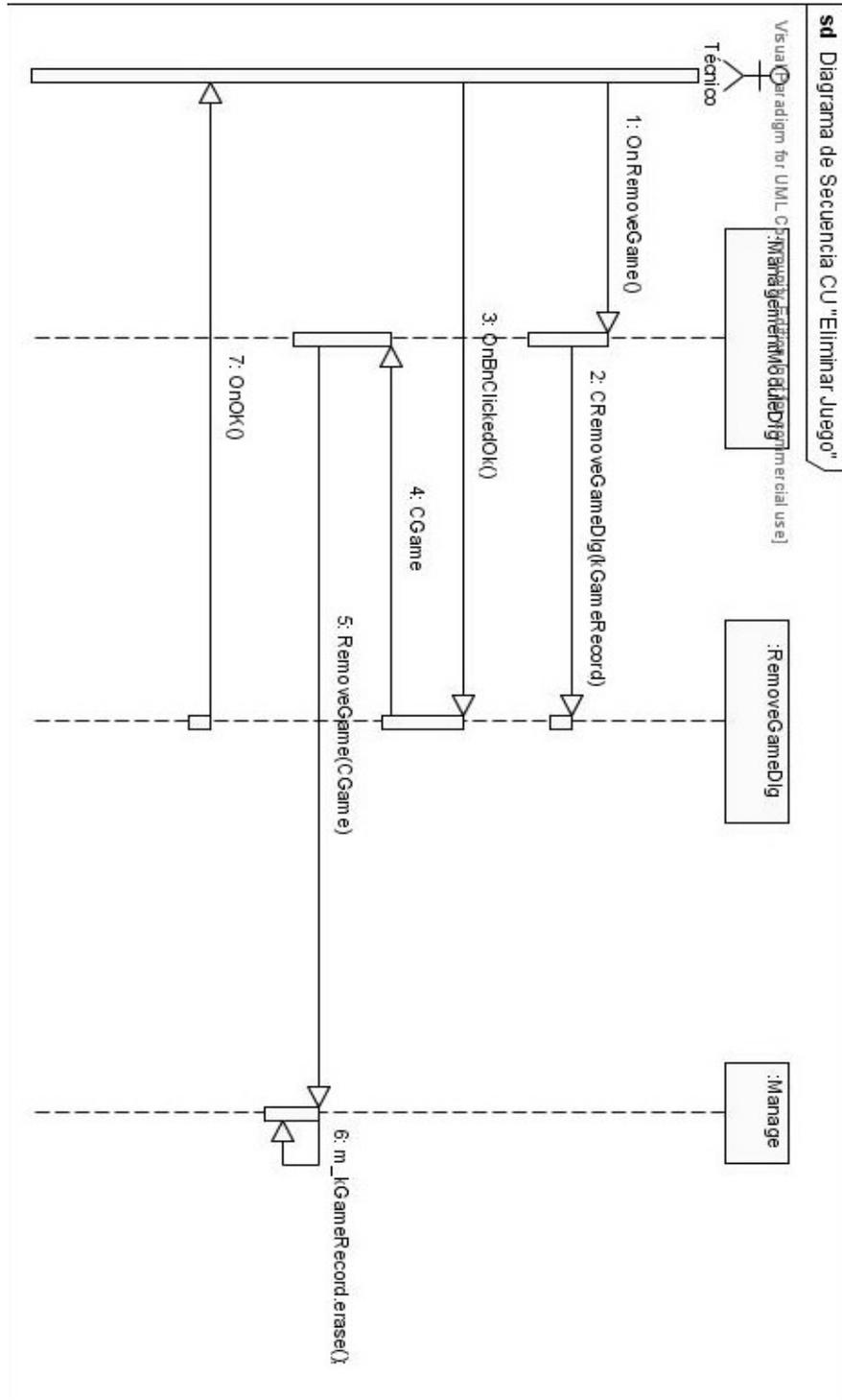


Figura 13 Diagrama de Secuencia CU "Gestionar Juego" Sección: Eliminar Juego

Nombre del caso de uso	Mostrar mensajes.
Actores	
Propósito	Mostrar instrucciones de juegos y los datos del tratamiento.
Resumen:	El Caso de Uso se inicia cuando el CU “Gestionar Tratamiento” inicia el juego terapéutico
Referencias	R5.3,R5.4,R5.4.1,R5.4.2,R5.5,
Poscondición	Mensaje mostrado y datos visualizados.
Curso normal de los eventos	
Acción del actor	Respuesta del sistema
1- Se solicita mostrar mensaje.	1.1- Muestra mensaje.
Prioridad: Secundario.	

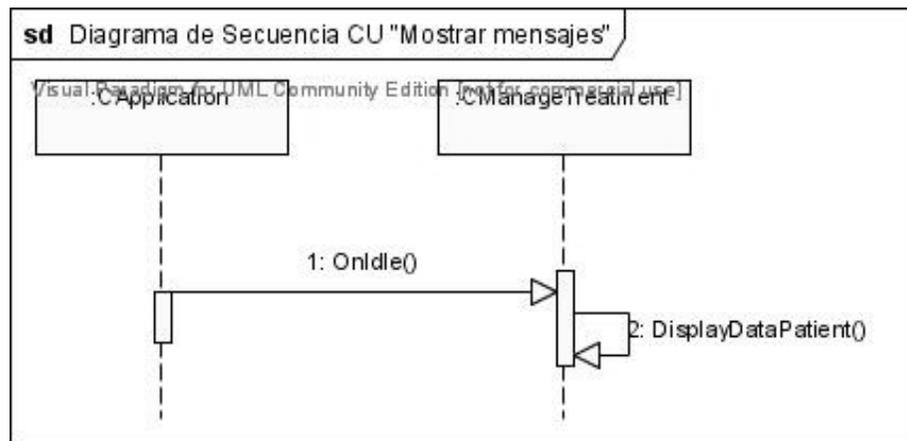


Figura 14 Diagrama de Secuencia CU “Mostrar mensajes”

Nombre del caso de uso	Gestionar datos del juego.
Actores	
Propósito	Guarda los datos del tratamiento realizado
Resumen:	El Caso de Uso se inicia cuando el CU “Gestionar Tratamiento” inicia el juego terapéutico
Referencias	R4.1.4,R4.2,R4.2.1,R5,R5.1,R5.2
Poscondición	Se realizó el juego y se guardaron los datos de este.
Curso normal de los eventos	
Acción del actor	Respuesta del sistema
1- Se solicita iniciar el juego.	1.1- Se guardarán todos los datos del juego(Tiempo jugado, objetos capturados, puntuación obtenida, tiempo medio de recogida de objetos.)
Prioridad: Secundario.	

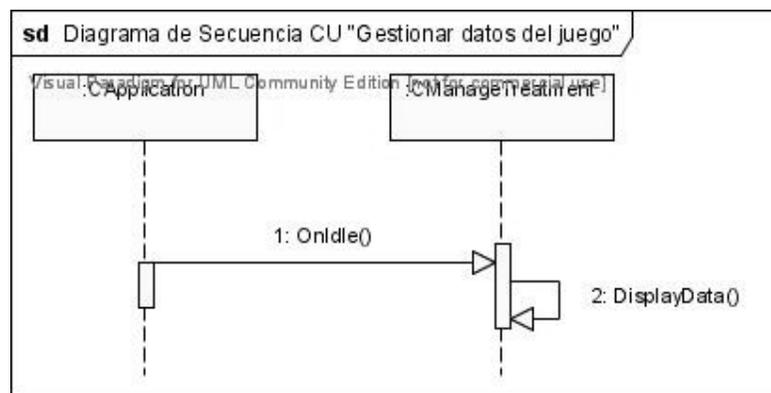


Figura 15 Diagrama de Secuencia CU “Gestionar datos del juego”

3.6 Patrones usados en el diseño del Sistema

Un patrón de diseño es una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reusable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias.

3.6.1 Patrones GRASP

Para la asignación de responsabilidades se decidió trabajar con GRASP (General Responsibility Assignment Software Patterns). Aunque se considera que más que patrones propiamente dichos, son una serie de “buenas prácticas” de aplicación recomendables en el diseño de software.

Entre los patrones de este tipo se encuentra el Controlador, el cual sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado. Sirve para asignar una responsabilidad de recibir o manejar un mensaje de evento del sistema a una clase que representa el sistema global, dispositivo o subsistema; o representa un caso de uso en el que tiene lugar el evento del sistema a menudo.

Podemos ver su utilización en el diseño de la interfaz CManagementModuleDlg, de la capa View, la cual maneja todos los eventos del sistema con los que interactúa el usuario y los implementa a través de la clase CManage.

Otro patrón usado es el Experto, el cual es un principio básico de asignación de responsabilidades, indicando que la responsabilidad de la creación de un objeto debe recaer sobre la clase que conoce toda la información necesaria para crearlo. Esto lo podemos observar en la clase CManage la cual lleva el registro de todos los objetos de las clases entidades con los que trabaja el sistema, responsabilizándose con su creación, modificación y destrucción.

3.6.2 Patrones GoF

Para el diseño se utilizaron los patrones GoF los cuales son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. En este sistema se utilizó el patrón

Facade o fachada el cual provee de una interfaz unificada sencilla que haga de intermediaria entre un cliente y una interfaz o grupo de interfaces más complejas.

El uso de este patrón se representa en la clase interfaz CManagementModuleDlg que gestiona la interacción entre el objeto de la clase CManage y las demás interfaces del sistema, abstrayéndolo de la implementación de las mismas.

3.7 Diagrama de clases

Un *diagrama de clases* es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre ellos.¹³

Para comprender mejor el diagrama de clases del diseño del presente trabajo, se han separado las mismas en dos paquetes atendiendo a su funcionalidad, dichos paquetes se han dividido a su vez en otros subpaquetes para una mayor organización y legibilidad.

¹³ **Wikimedia Foundation, Inc.** Diagrama de clases. *Wikipedia*. [En línea] [Citado el: 6 de Mayo de 2008.] http://es.wikipedia.org/wiki/Diagrama_de_clases.

reportes generados a partir de las clases mencionada anteriormente. El Paquete *Manage* en general permite a la aplicación el control de los datos persistentes.

El paquete *Game* contiene las clases que controlarán del desarrollo del juego terapéutico. Este paquete se relaciona con *Manage* ya que su funcionamiento dependerá de los datos gestionados por las clases contenidas en este paquete.

A continuación modelaremos y explicaremos los subpaquetes del paquete *Manage*.

3.7.1.1 Subpaquete Report

Este subpaquete contiene a la clase *CReport* la que se encargará de crear los reportes según los parámetros establecidos y mostrarlos. Para esto se vinculará con las clases *CPatient* y *CVariant*.

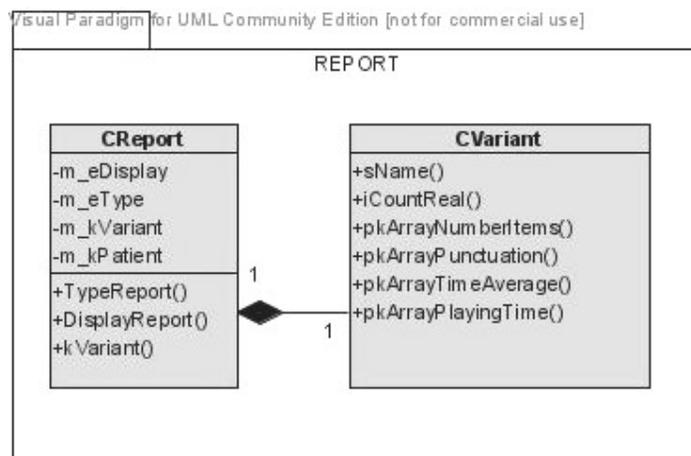


Figura 17 Subpaquete Report

3.7.1.2 Subpaquete Patient

En este subpaquete se controlarán los pacientes registrados mediante la clase *CPatient*. Esta clase se relaciona con *CDateTime* para realizar un control del tiempo de registro de cada paciente, datos que serán utilizados en los tratamientos y reportes.

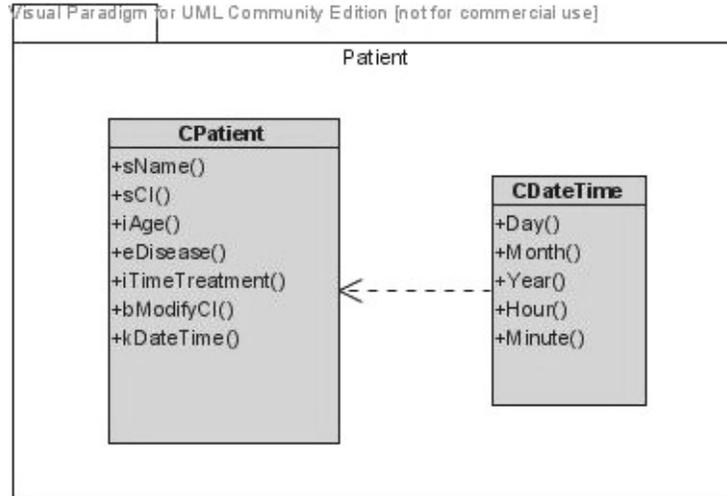


Figura 18 Subpaquete Patient

3.7.1.3 Subpaquete Treatment

La clase *CTreatment*, contenida en este subpaquete es la encargada del control de los datos de los tratamientos realizados por cada paciente. Gestiona las variables del juego terapéutico (puntuación, número de objetos, tiempo de juego) y los datos del paciente que lo realiza (paciente, fecha) para posteriormente realizar los reportes.

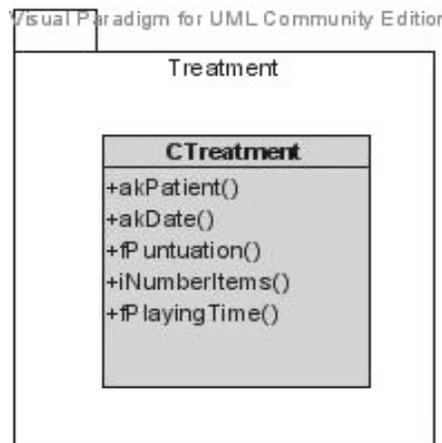


Figura 19 Subpaquete Treatment

3.7.2 Paquete Game

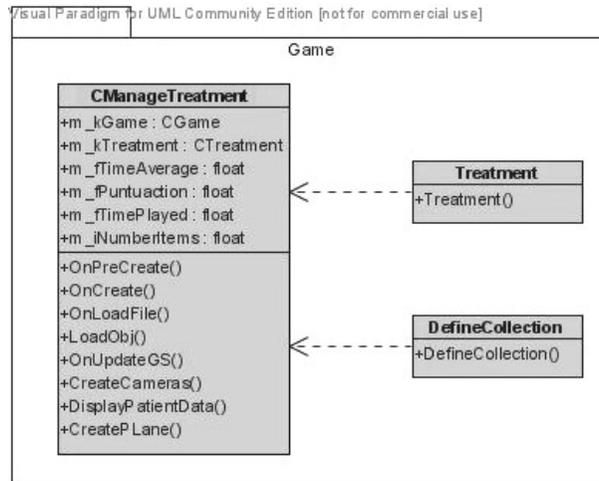


Figura 20 Paquete Game

Este paquete contiene las clases que hacen posible la visualización del entorno del juego terapéutico así como la interacción con las librerías de STK.

Capítulo 4: Implementación

4.1 Introducción

En este capítulo se representarán y detallarán los artefactos que constituyen las clases de cada módulo de la aplicación, así como la relación existente entre ellos. Se presentarán también el modelo de datos y diagramas de las clases persistentes para lograr una mejor comprensión de la información que será gestionada y almacenada.

Se expondrán también la metodología seguida para la arquitectura del sistema, la elaboración de las interfaces de los módulos de la aplicación, la estructura de la ayuda y los reportes realizados.

Se realizarán una especificación de los estándares seguidos en la codificación con el objetivo de lograr una mejor organización y un estilo propio del software. Se presentará además el modelo de implementación definiendo los componentes que forman la estructura del sistema.

4.2 Control de Datos

Para un buen entendimiento de cómo se realiza el control de datos en el sistema se modelará el diagrama de las clases persistentes que lo componen y el modelo de datos, este último se corresponde con la representación física de la misma.

4.2.1 Diagrama de clases persistentes

Las *clases persistentes* son aquellas que en una aplicación implementan las entidades del problema de negocio. No todas las instancias de una clase persistente se considera que estén en el estado persistente, una instancia puede en cambio ser transitoria o estar separada.

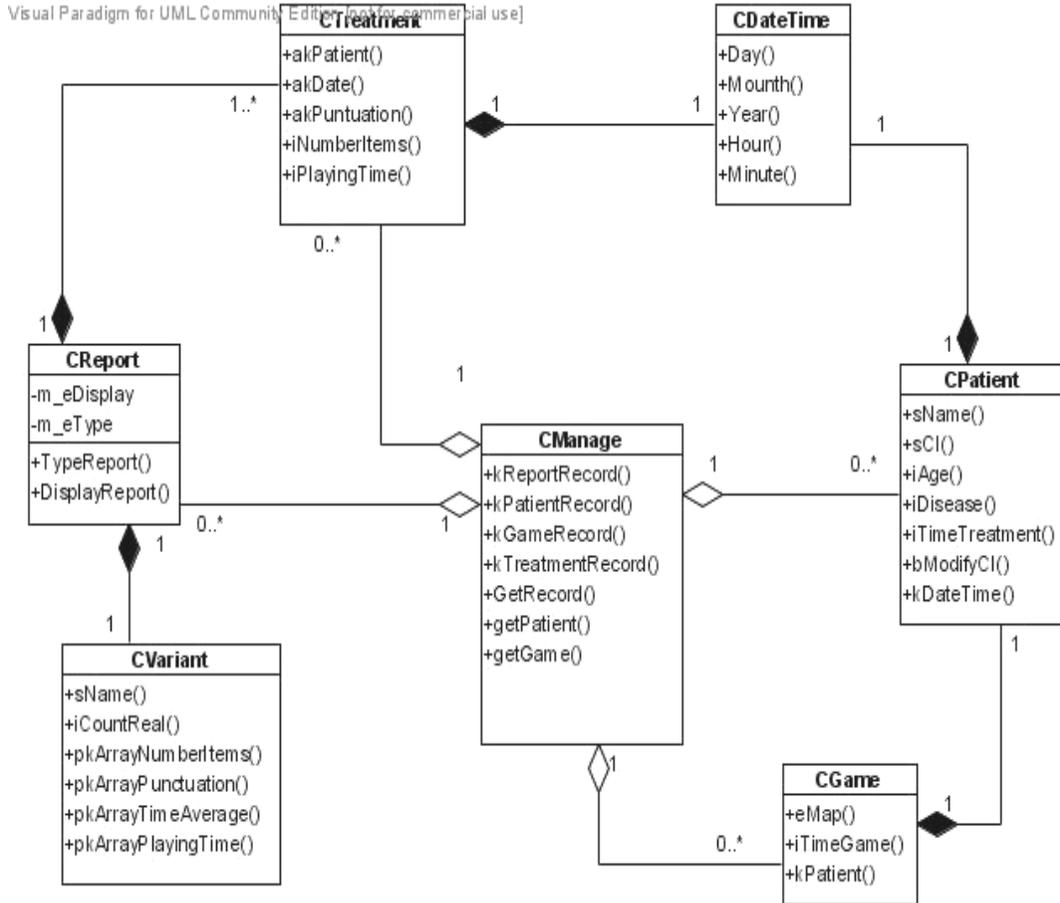


Figura 21 Diagrama de Clases Persistentes

4.2.2 Modelo de Datos

Un *modelo de datos* es aquel que describe de una forma abstracta cómo se representan los datos en un sistema de información o en un sistema de gestión de base de datos. Básicamente consiste en una descripción de algo conocido como *contenedor de datos* (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores.

Un modelo de datos consiste en:

- ❖ Objetos (entidades que existen y que se manipulan)
- ❖ Atributos (características básicas de estos objetos)

❖ Relaciones (forma en que enlazan los distintos objetos entre si)¹⁴

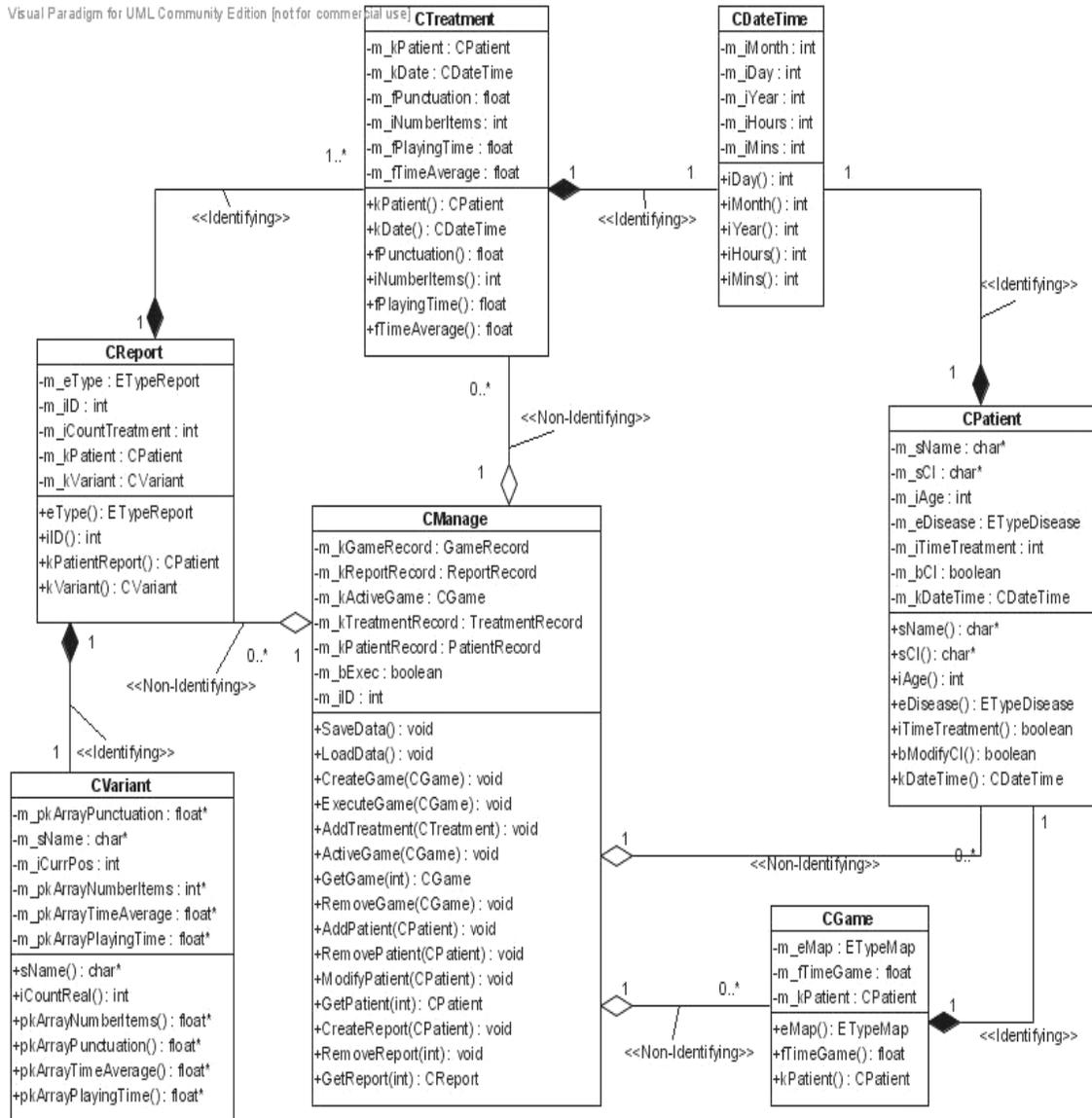


Figura 22 Diagrama del Modelo de Datos

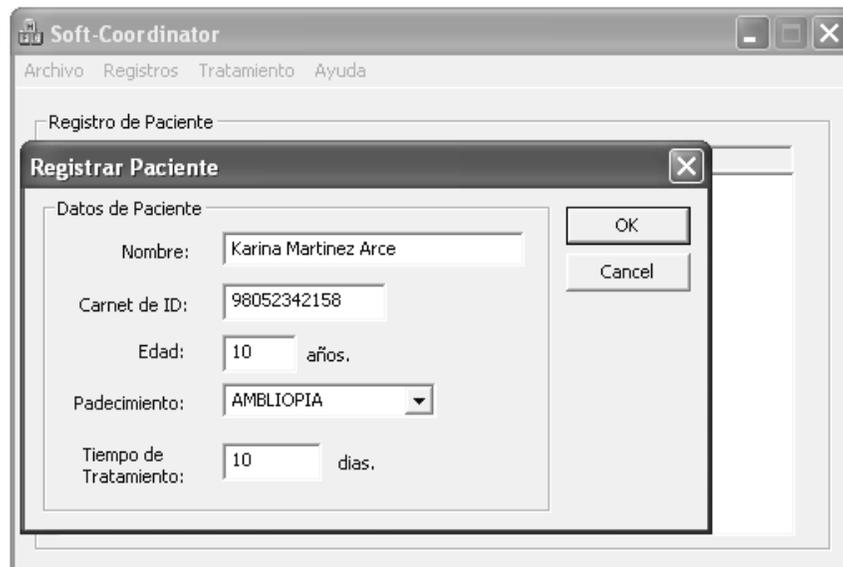
¹⁴ Wikipedia. [En línea] Wikimedia Foundation, Inc. [Citado el: 10 de Junio de 2008.] http://es.wikipedia.org/wiki/Modelo_de_datos.

4.3 Principios de diseño

Para la elaboración del diseño se pensó principalmente en los usuarios finales que, en este caso, serán los técnicos. Estos no en todos los casos poseen los conocimientos óptimos de computación, por lo que se eligió el diseño de una interfaz amigable, comunicativa e intuitiva. Con este diseño creado el sistema cumple con los requisitos no funcionales y con los principios de funcionamiento de la herramienta Coordinator Bangester.

4.3.1 Interfaz del Módulo Manage

En el diseño del Módulo Manage se seleccionaron los colores gris, azul y negro. Se eligió el uso del azul por ser un color relajante; el gris por ser neutral, provocar la sensación de estabilidad y orden, razón por la que es utilizado en la mayoría de las aplicaciones para diferentes Sistemas Operativos. El color rojo se utilizó para resaltar los mensajes de errores cometidos al registrar algún dato requerido o por operaciones no válidas.



Soft-Coordinator

Archivo Registros Tratamiento Ayuda

Registro de Paciente

Registrar Paciente

Datos de Paciente

Nombre: Karina Martinez Arce

Carnet de ID: 98052342158

Edad: 10 años.

Padecimiento: AMBLIOPIA

Tiempo de Tratamiento: 10 días.

OK

Cancel

Figura 23 Interfaz del Módulo Manage

4.3.2 Interfaz del Módulo Game

Para el diseño del Módulo Game se seleccionaron diferentes colores y formas. Se eligió un entorno de un laberinto para crear habilidades de orientación en el espacio. Los colores que más predominan son el verde, amarillo y verde-amarillo por las propiedades explicadas en capítulos anteriores. El color rojo se utilizó para resaltar los mensajes. Para las figuras interactivas de los diálogos se escogieron personajes conocidos por los niños para crear más familiaridad con el sistema.



Figura 24 Interfaz del Módulo Game

4.3.3 Concepción general de la Ayuda

El sistema cuenta con un manual de ayuda que esta disponible desde la misma aplicación, en dicho manual, se explica cómo realizar el trabajo con el sistema y las principales funcionalidades de este.



Figura 25 Paquete de Ayuda del Sistema

4.3.3.1 Herramientas de desarrollo de la Ayuda

Para el desarrollo de la Ayuda se utilizó la herramienta RoboHelp X5. Macromedia® RoboHelp® X5 es una herramienta de autoría de ayuda estándar y es utilizada por los desarrolladores y escritores técnicos para crear sistemas y documentación de ayuda profesionales para el escritorio y aplicaciones basadas en web. Con RoboHelp X5, se pueden aprovechar su soporte para XML, capacidad de importar y exportar PDF,

administración de contenidos, fuerzas de trabajo distribuidas y funcionalidades de autoría.

4.3.4 Tratamiento de excepciones y errores.

Para prevenir errores por parte del usuario en la manipulación de datos en el sistema, sólo se le brindan las opciones mínimas necesarias, a la hora de efectuar cualquier operación, por ejemplo, se deshabilitan ciertos botones si el usuario no tiene que utilizarlos en ese momento o se muestran mensajes que comunicarán la usuario el tipo de error cometido al registrar los datos.

Mediante una combinación de validación se garantiza que los datos que proporcione el usuario al sistema mantengan su integridad y consistencia al almacenarse. Se verifican además los campos obligatorios, y se revisa el tipo de datos.

Se manipulan las excepciones que podrían ocurrir en tiempo de corrida de la aplicación, mostrando mensajes al usuario haciendo un reporte del tipo de excepción ocurrida y la forma correcta de manejar los datos para eliminarla.

```
try
{
    . . .
}
catch (CException* err)
{
    err->ReportError();
}
catch (CError e)
{
    MessageBox(e.getInfo(), "Mensaje de Error");
}
```

Figura 26 Tratamiento de errores y excepciones

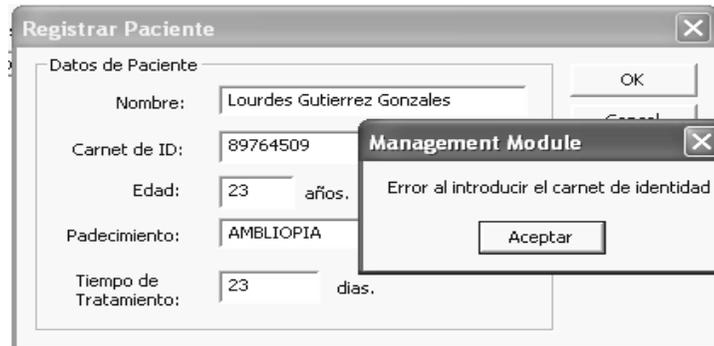


Figura 27 Verificación de campos requeridos y en formato correcto

4.4 Estándares de codificación

La utilización de un estándar para la codificación resulta muy ventajosa. Entre sus ventajas se encuentran:

- ❖ Reducción de errores.
- ❖ Se obtiene un código comprensible y fácil de leer.
- ❖ Se garantiza una buena comunicación entre los programadores del equipo
- ❖ Facilitar el mantenimiento del software

El código sigue algunos estándares propuestos en el uso de la herramienta STK, teniendo en cuenta y respetando los estándares de codificación para C++. Para la programación se usó el idioma inglés, debido a la simplicidad de las palabras, que no se acentúan y al nivel de difusión que ha alcanzado este idioma en la informática.

A continuación se muestran ejemplos de estándares usados:

Tipo	Descripción	Ejemplo
Clases	Se utiliza el indicador “C” para indicar que es una clase.	<ul style="list-style-type: none"> class CPatient
Declaración de variables	Los nombres de las variables comienzan con un identificador del tipo de dato al que correspondan.	<ul style="list-style-type: none"> bool m_bCI; int m_iAge; float m_fTimeGame;

Tabla 13. Estándares de codificación

```

...

void CReport::CreateVariant(TreatmentRecord arg_kTreatment)
{
    for(TreatmentRecordIterator iter = arg_kTreatment.begin();
        iter != arg_kTreatment.end(); iter++)
    {
        m_kVariant.AddValues((*iter).iNumberItems(),
            (*iter).fPunctuation(), (*iter).fTimeAverage(), (*iter).fPlayingTime());
    }
    m_kVariant.Name(m_kPatient.sName());
    m_iCountTreatment = arg_kTreatment.size();
}

...

```

Figura 28 Ejemplo de codificación utilizada

4.5 Arquitectura del Sistema

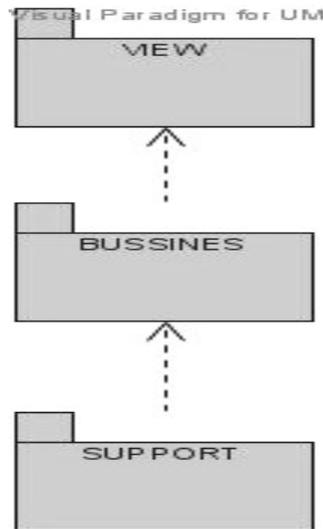


Figura 29 Arquitectura del sistema

El sistema se desarrolló siguiendo la estructura representada en la Figura 29. Este se dividió en tres niveles o capas (View, Bussines y Support), las que estarán directamente vinculadas en el intercambio de datos. En el nivel *View* se encontrarán todos los formularios del Módulo Manage donde el usuario intercambiará su información con el sistema. El próximo nivel es *Bussines* en el que se controlará todo el proceso del negocio descrito anteriormente. Y el último nivel es *Support* el cual contendrá los archivos binarios que se encargarán de la persistencia de los datos del sistema.

4.6 Diagrama de Componentes

El *diagrama de componentes* muestra la relación entre componentes físicos de software (archivos, cabeceras, módulos, paquetes, etc.), sus dependencias, su comunicación su ubicación y otras condiciones. Son utilizados para modelar la vista estática de un sistema. Muestra la organización y las dependencias entre un conjunto de componentes. No es necesario que un diagrama incluya todos los componentes del

4.6.1 Explicación de los componentes

Para una mejor comprensión de cada componente se explicarán los mismos secuencialmente.

❖ **Paquete Manage**

En este paquete se encuentran los componentes que contienen las implementaciones de las clases que se ocupan de la persistencia de los datos de ciertas entidades del sistema. Estos datos son los utilizados para realizar los tratamientos y reportes de los pacientes.

❖ **Paquete Game**

En este paquete se encuentran los componentes que contienen las implementaciones de clases que se ocupan del desarrollo del juego terapéutico.

❖ **Paquete View**

En este paquete se encuentran los componentes que contienen las implementaciones de las clases que se encargan del intercambio de información con el usuario.

❖ **Paquete STK**

En este paquete se encuentran los componentes que contienen las implementaciones de clases del STK empleadas por el paquete Game para su desarrollo.

❖ **Paquete Visual Studio Library**

En este paquete se encuentran las clases que implementa el Visual Studio que serán usadas para facilitar el trabajo con el Framework que este implementa.

❖ **Paquete Estándar Library C++**

Este paquete contiene las librerías estándares de C++ que se utilizan en el desarrollo de la aplicación.

Existen otros componentes que nos fueron especificados anteriormente y estos son:

Componente	Propósito
GameTemp.dat	Este archivo se encargará de guardar los datos del juego que se cargarán para iniciar el tratamiento.
TreatmentTemp.dat	Este archivo guardará los datos del juego que utilizará el sistema para realizar los reportes

Tabla 5 Otros componentes

Conclusiones

Con este trabajo se propone una alternativa de solución a las dificultades que presenta el proceso de rehabilitación visual en las Escuelas Especiales para niños estrábicos y ambliopes.

Esta primera versión de Soft-Coordinator posibilitará, con el cumplimiento de los principios de funcionamiento de algunos de los equipos, la validación de teorías científicas elaboradas con el fin del proceso de rehabilitación en estos escolares, relacionadas con el desarrollo de habilidades visuales.

Es un sistema capaz de gestionar los datos de los pacientes y brindar reportes del tratamiento que estos reciban mediante el juego terapéutico; marcando el comienzo del uso de las TICs en procesos rehabilitativos visuales y brindando un banco de datos que puede ser reutilizado en otras aplicaciones futuras por los especialistas dedicados al estudio de estos procesos.

El sistema se desarrolló siguiendo la metodología RUP, y se utilizaron representaciones UML para la modelación de todas las fases del proyecto. También está provisto de un ambiente cómodo, fácil de entender, que cumple los estándares de diseño y utiliza técnicas modernas de programación orientada a objetos.

Recomendaciones

Los objetivos generales de este trabajo han sido logrados y se ha creado la primera versión de Soft-Coordinator. Pero a lo largo de su desarrollo, han ido surgiendo ideas que podrían implementarse en un futuro, de forma que se logre una la aplicación más útil y efectiva, para lo cual se recomienda:

1. Diseño y creación de otros mapas a utilizar en el juego terapéutico por parte de diseñadores especialistas.
2. Extender el sistema de manera que pueda ser utilizado no sólo por el técnico, sino también a otro personal que requiera de un control de datos y reportes de los pacientes.
3. Ampliar los módulos del sistema y brindar más funcionalidades adecuando los tratamientos a otros tipos de padecimientos visuales.
4. Implementarle a los módulos un sistema de seguridad en sus próximas versiones.
5. Continuar el desarrollo de este sistema, adicionándole nuevas funcionalidades, adecuándolo más a las especificaciones de cada proceso de tratamiento realizados en las Escuelas Especiales, haciéndolo más útil y provechoso.

Bibliografía

❖ FUENTES BIBLIOGRÁFICAS

Araujo, J., y otros. 2005. *"Integration and Transformation of UML Models"*. s.l. : ECOOP Workshops, 2005.

Arlow, J. y Neustadt, I. 2002. *"UML and the Unified Process: Practical Object-Oriented Analysis and Design"*. ISBN 0201770601. s.l. : Addison-Wesley, 2002.

Ávila Saint, N. 1998. *"Un reto a la minusvalía desde la escuela especial para deficientes visuales en Santiago de Cuba"*. La Habana : s.n., 1998. En libro de Resúmenes del II Encuentro Mundial de la Educación Especial.

Barraga, N. 1997. *"Textos Reunidos"*. Madrid : ONCE, 1997. Vol. Cap.1."Libro sobre Baja Visión".

Bell Rodríguez, R. 1997. *"Educación Especial. Razones, visión actual y desafíos"*. Ciudad de La Habana : Pueblo y Educación, 1997.

Ciancia, A.O. 1966. *"Ortóptica y Pleóptica. Los tratamientos re-educativos del estrabismo"*. Buenos Aires : Mcchi, 1966. pág. 255.

Crespo, S. 2000. "Rehabilitación visual. Un proceso". *Jornada Internacional sobre los aspectos socio-educativos del deficiente visual*. Madrid : s.n., 2000.

Martín Ford, Dayelín. 2008. Tesis para optar por el título de Master. *Bange-Soft: Alternativa para el tratamiento visual de escolares con NEE*. Ciudad de La Habana : ISP " E. J. Varona", 2008.

Ferreiro Gravie, R. 1986. *"Anatomía y Fisiología del Desarrollo e Higiene Escolar"*. Ciudad de La Habana : Pueblo y Educación, 1986.

La magnificación de pantallas como ayuda a los deficientes visuales: El sistema MEGA. **Alonso, F. 1997.** 23, Madrid : ONCE, 1997, Revista sobre ceguera y deficiencia visual.

Litvak, A.G. 1981. *"Tiflopsicología"*. Moscú : Litvak-Vipo Vneshtorgizdat, 1981.

Martin Ford, D. 2003. "Los Software Educativos en el proceso de corrección y/o compensación del estrabismo y la ambliopía". *Trabajo de Diploma*. Ciudad de La Habana, Cuba : s.n., 2003.

Martín Ford, D. 2008. "Bange-Soft: Alternativa para el tratamiento visual de escolares con NEE.". *Tesis de Maestría*. Ciudad de La Habana : ISP " E. J. Varona", 2008.

Morris, E. y Coautores. 1999. *"Baja Visión. Programa para desarrollar eficiencia en el funcionamiento visual"*. Córdoba : s.n., 1999.

Pérez Martinto, P.C. 1999. "Propuesta de Software Educativo para la Estimulación Visual de los escolares Ambliopes con Fijación Central". *Tesis de Maestría*. Ciudad de La Habana : Centro de Referencia Latinoamericana para la Educación Especial - CELAEE - IPLAC, 1999.

Pérez, G. y Coautores. 1998. "*Metodología de la Investigación Educativa*". Ciudad de La Habana : Pueblo y Educación, 1998.

Sánchez, D. 1998. "Alternativas para el desarrollo visual del niño en estrabismo y ambliopía". *Tesis de Maestría*. Ciudad de La Habana : s.n., 1998.

Velásquez, B. y Coautor. 2003. "*Análisis de la agudeza visual con diferentes técnicas en pacientes ambliopes*". México : México, 2003.

Vigostki, L.S. 1998. "*Obras Escogidas*". Ciudad de La Habana : Pueblo y Educación, 1998. Vols. V- LS.

❖ FUENTES PUBLICÍSTICAS

"*Micro mundos lúdicos interactivos*". **Galvis Panguero, A. 1998.** 12, México : s.n., 1998, Revista tecnología y comunicación educativas.

❖ FUENTES DIGITALES

Agile Models Distilled: Potential Artifacts for Agile Modeling. [En línea] [Citado el: 30 de Noviembre de 2007.] <http://www.agilemodeling.com/artifacts/> .

Méndez Sánchez, Teresita de J., y otros. Resultados de la rehabilitación visual en ambliopes del Centro Oftalmológico Infantil. [En línea] [Citado el: 10 de Febrero de 2008.] http://bvs.sld.cu/revistas/oft/vol15_2_02/oft09202.htm.

Monografías. *¿Cómo investigar en educación?* [En línea] [Citado el: 10 de Diciembre de 2007.] <http://www.monografias.com/trabajos13/artinves/artinves.shtml>.

Monografías. *Métodos*. [En línea] [Citado el: 5 de Febrero de 2008.] <http://www.monografias.com/trabajos11/metodos/metodos.shtml>.

Moráquez Iglesias, Arabel. El método Delphi. [En línea] [Citado el: 10 de Marzo de 2008.] <http://www.gestiopolis.com/canales6/eco/metodo-delphi-estadistica-de-investigacion-cientifica.htm>.

UML basics: An introduction to the Unified Modeling Language. [En línea] [Citado el: 3 de Enero de 2008.] <http://www-128.ibm.com/developerworks/rational/library/769.html#footnotes>.

Glosario de Términos

Actores: Agentes externos al sistema que de alguna manera participa en la historia del caso de uso.

Ambliopía: Conocido también como *ojo vago* o *perezoso* consiste en la pérdida parcial de la visión. Normalmente afecta a un ojo, pero a veces es bilateral por existir defectos importantes de refracción en ambos ojos, especialmente astigmatismos graves. Déficit de agudeza visual aún con la mejor compensación posible.

Caso de Uso: Especifica la secuencia de eventos y acciones de un agente externo (actores) que utiliza el sistema para completar un proceso, incluyendo las alternativas dentro de la secuencia.

Clase: Definiciones de las propiedades y comportamiento de un tipo de objeto concreto. La instanciación es la lectura de estas definiciones y la creación de un objeto a partir de ellas.

Entidad: Corresponde al objeto que se quiere representar. Son elementos reales o abstractos, acerca de las cuales se requiere registrar información.

Estrabismo: Es la desviación del alineamiento de un ojo en relación al otro. Implica la falta de coordinación entre los músculos oculares. Esto impide fijar la mirada de ambos ojos al mismo punto en el espacio, lo que ocasiona una visión binocular incorrecta que puede afectar adversamente en la percepción de la profundidad.

Modelo: En términos generales, es una representación del sistema que se analiza. En Informática, es una representación de conceptos del dominio del problema.

Multi-plataforma: Término utilizado frecuentemente en informática para indicar la capacidad o características de poder funcionar o mantener una interoperabilidad de forma similar en diferentes sistemas operativos o plataformas. Por ejemplo la posibilidad de utilizar un programa o software determinado en sistemas Windows y Linux.

Objeto: Entidad provista de un conjunto de propiedades o atributos (datos) y de comportamiento o funcionalidad («métodos»). Corresponden a los objetos reales del mundo que nos rodea, o a objetos internos del sistema.

Requerimientos: Descripción de las necesidades o deseos de un producto. En la fase de análisis y diseño de los sistemas resulta imprescindible identificar y documentar cada requerimiento.

Sistema: Conjunto de elementos interrelacionados entre sí que tienen un propósito determinado. En términos de gestión, cuenta con varios procesos básicos: las entradas, el procesamiento, el almacenamiento y las salidas.

Software: Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.

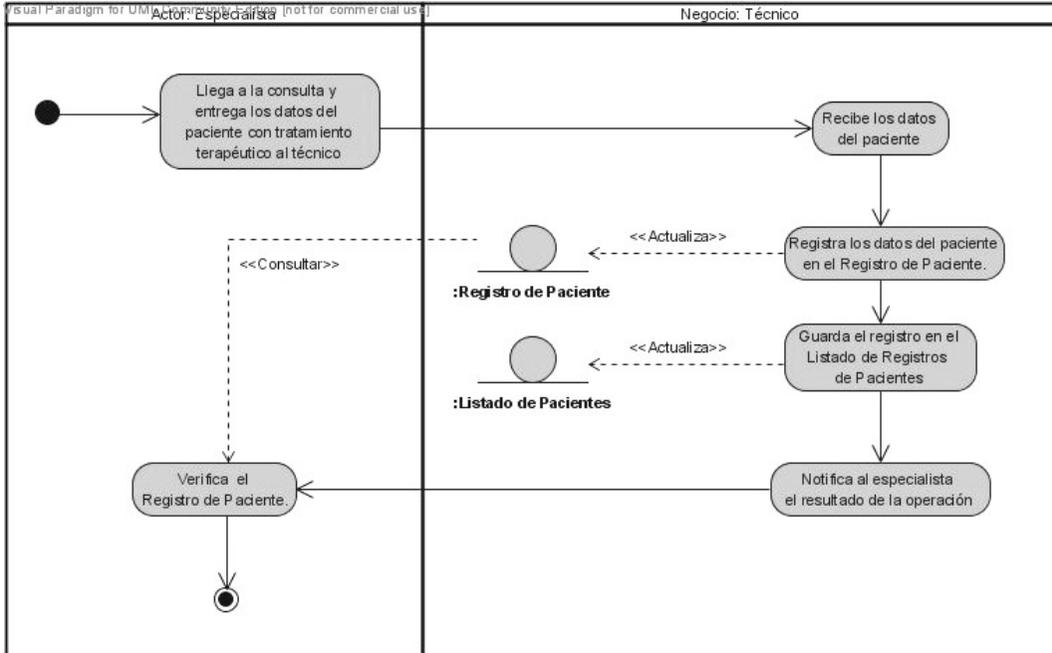
UML: (Unified Modeling Language / Lenguaje Unificado para la Construcción de Modelos) Se define como un "lenguaje que permite especificar, visualizar y construir

Glosario de Términos

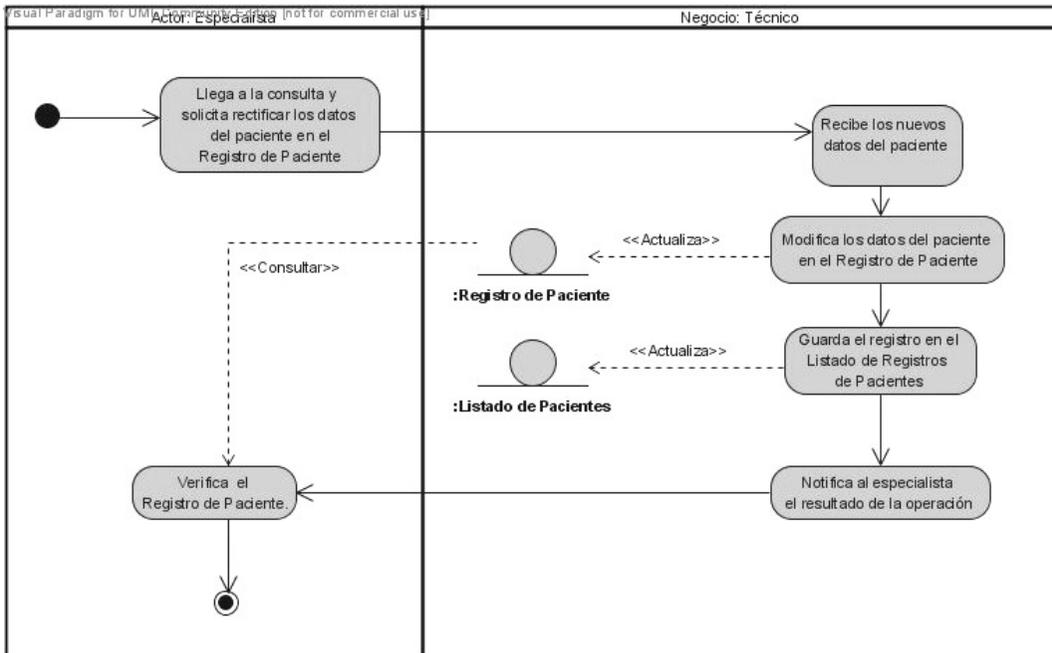
los artefactos de los sistemas de software”. Es un sistema rotacional (que, entre otras cosas, incluye el significado de sus notaciones) destinado a los sistemas de modelado que utilizan conceptos orientados a objetos. Pero este no guía al desarrollador en la forma de realizar el análisis y diseño orientados a objetos ni le indica cuál proceso de desarrollo adoptar.

ANEXOS

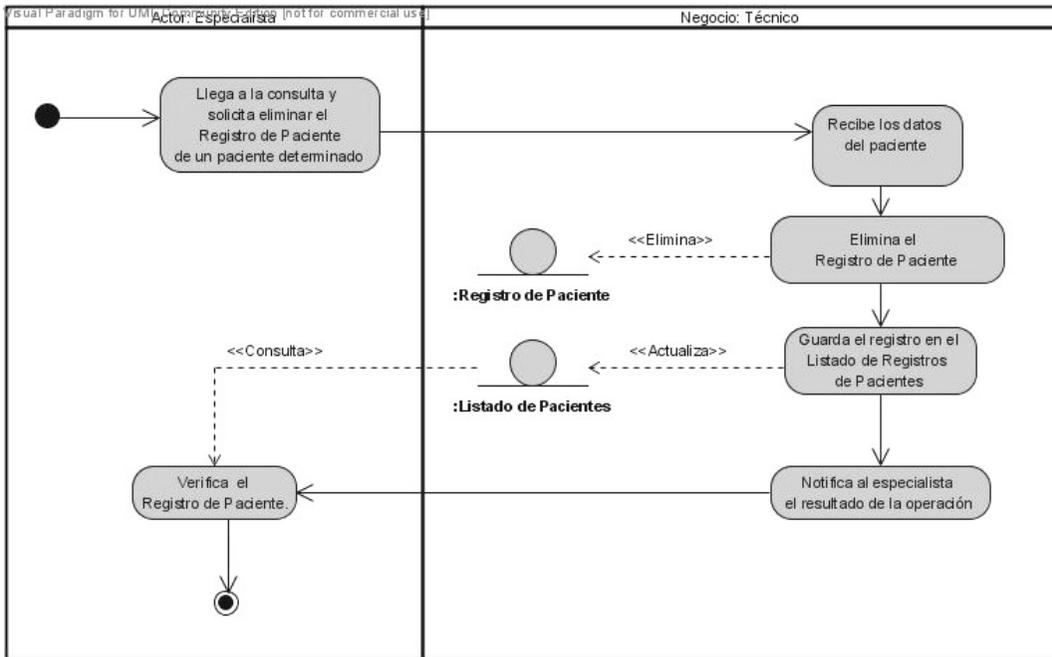
1. Diagrama de Actividades CUN Gestionar Registro de Paciente



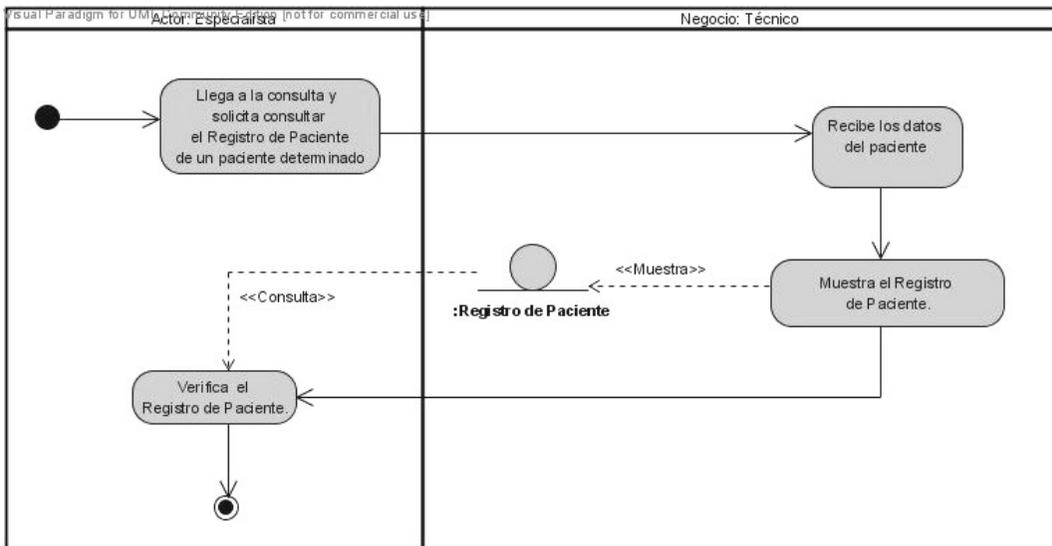
❖ Sección 1: Modificar Registro de Paciente



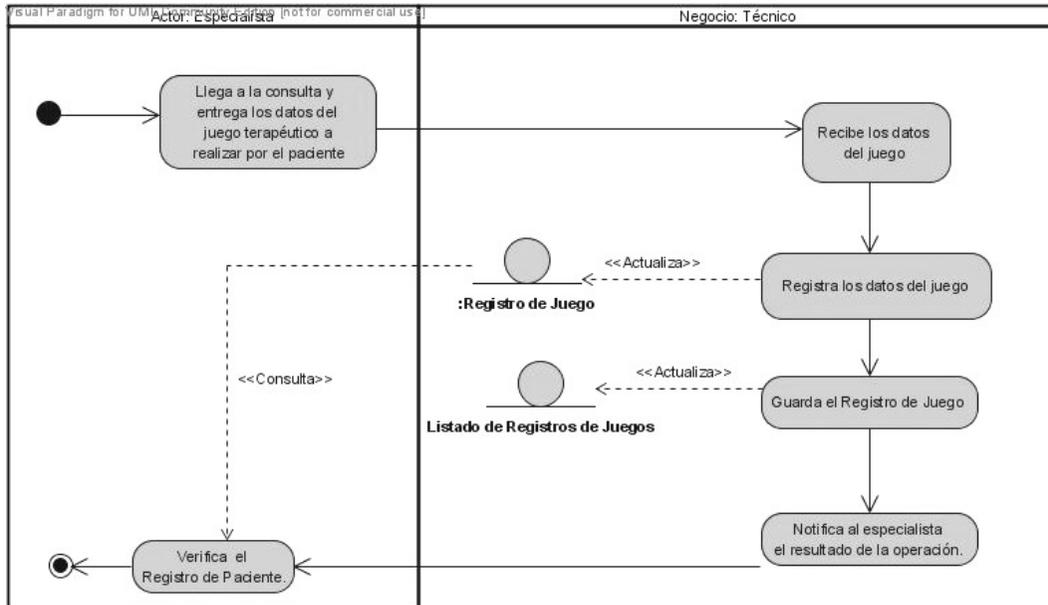
❖ Sección 2: Eliminar Registro de Paciente



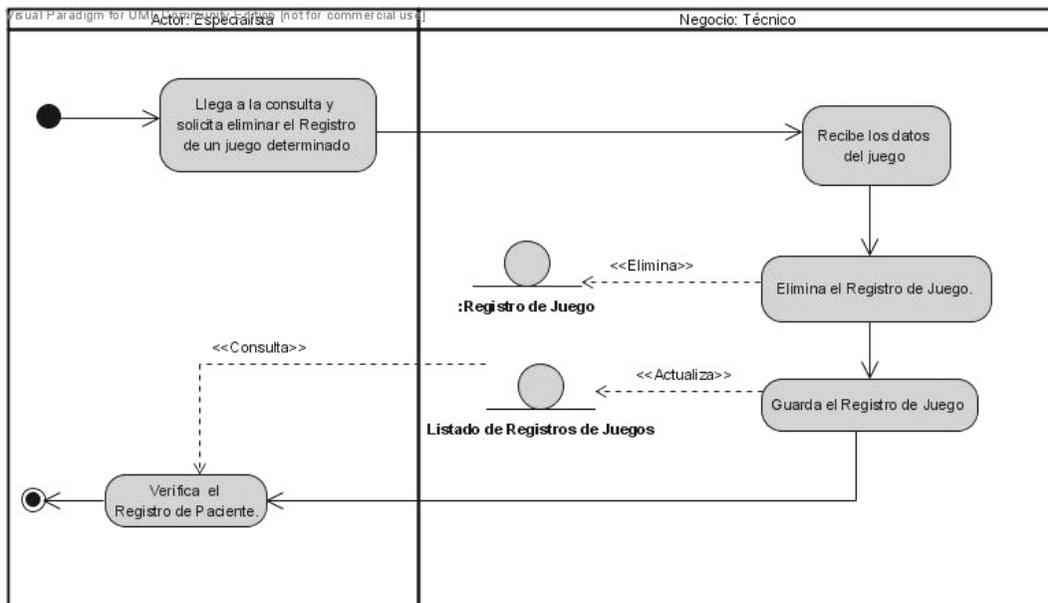
❖ Sección 3: Ver Registro de Paciente



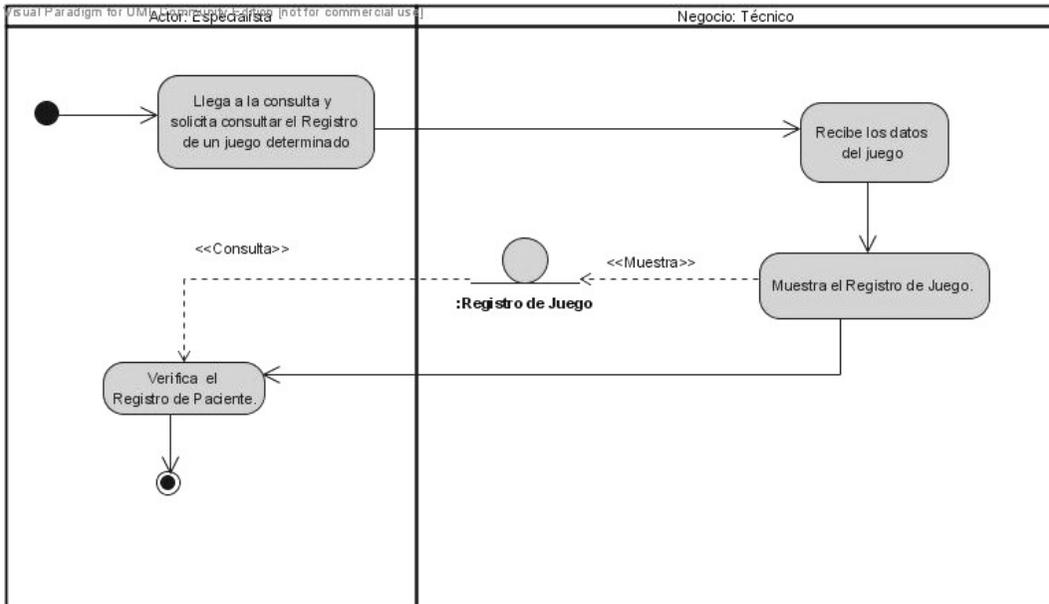
2. Diagrama de Actividades CUN Gestionar Juego Terapéutico



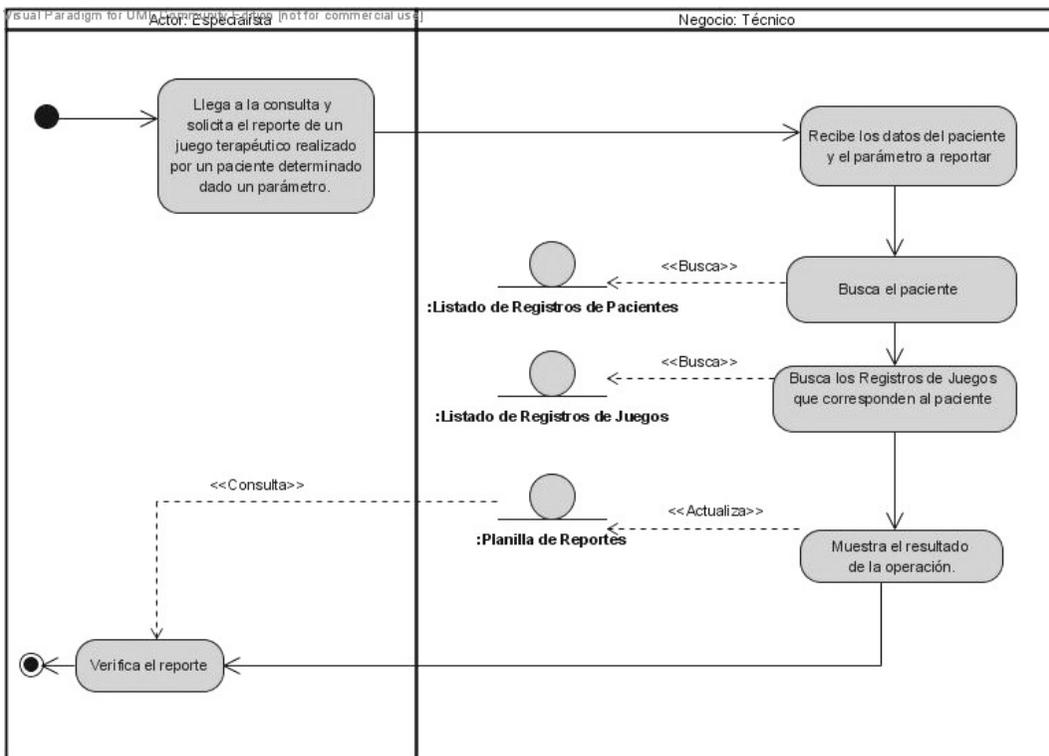
❖ Sección 1: Eliminar Juego Terapéutico



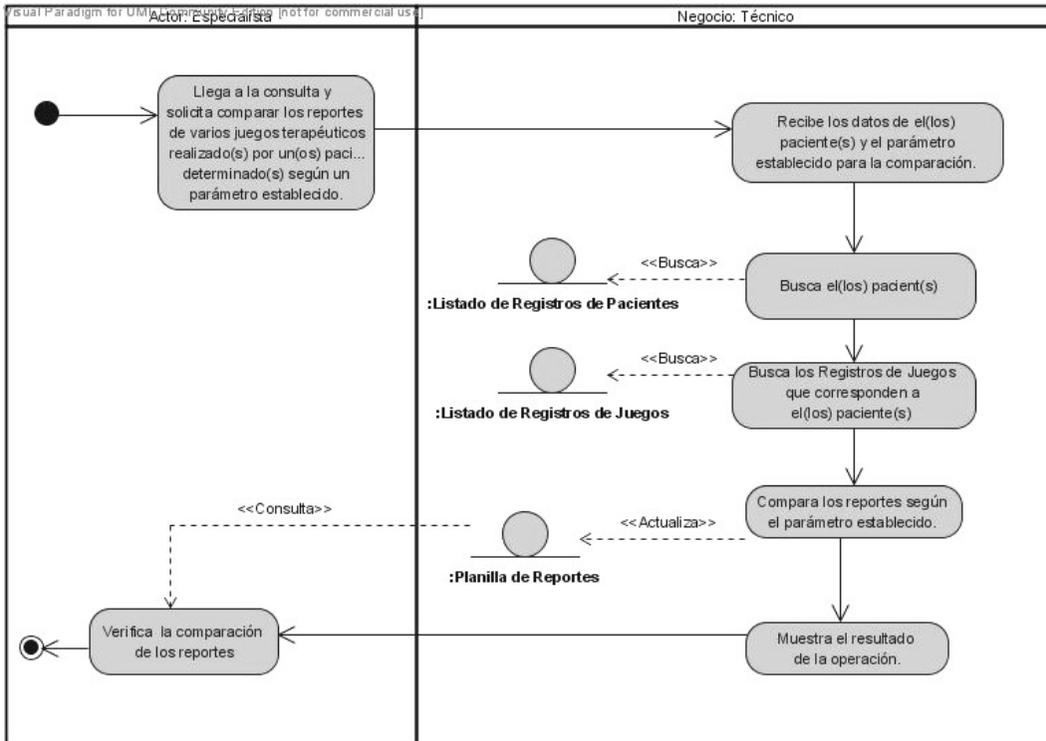
❖ Sección 2: Ver Registro de Juego Terapéutico



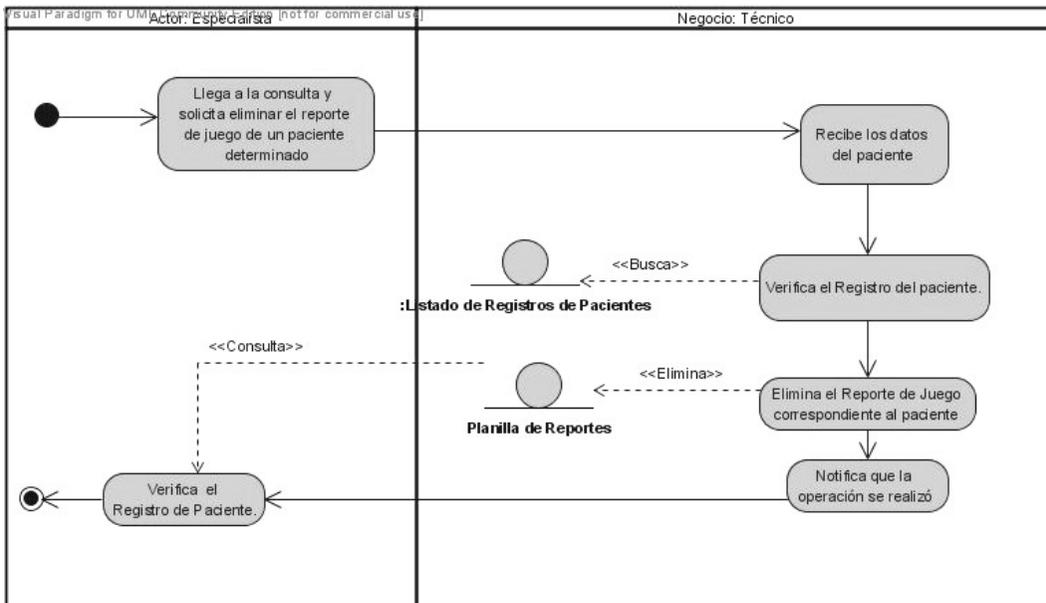
3. Diagrama de Actividades CUN Gestionar Reporte de Juego



❖ Sección1: Comparar Reportes de Juegos



❖ Sección 2: Eliminar Reporte de Juego Terapéutico



4. Diagrama de Actividades CUN Jugar

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.