



FACULTAD 8

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

*Análisis y diseño del juego interactivo
Energía para Aprender*



AUTORA: Aida María Saúco Peña
TUTORA: MSc. Delmis Caridad Báez Deniz

Ciudad de la Habana, julio del 2008
"Año 50 de la Revolución"

*“Se alcanza el éxito convirtiendo cada paso
en una meta y cada meta en un paso”*

C. Cortez

Declaración de Autoría

Declaro ser autor de la presente tesis y reconozco 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 ____ días del mes de julio del año 2008

Firma del Autor

Aida María Saúco Peña

Firma del Tutor

Delmis Caridad Báez Deniz

Agradecimientos

A mis padres por ser tan maravillosos conmigo, por su amor, educación y confianza en mí, por ser mi orgullo, por haber estado presentes siempre que los necesitaba, por su apoyo y guía indispensable a lo largo de mi vida como estudiante, por darme todooooo y tantooooo.

A mi hermana por darme la inspiración para seguir sus pasos, aguantar mis deseos de gobernar, ayudarme y estar siempre a mi lado.

A mi abuela por ayudarme y aconsejarme en todos los sentidos, por su cariño y dulzura.

A mi abuelo que tanto admiro por su carácter recto con un toque de jarana, por quererme y apoyarme.

A mis tías y tíos por su preocupación, ayuda y consejo en todo momento.

A mi prima Lili por estar ahí para mí, por siempre aconsejarme, apoyarme y estar a mi lado.

A mis amigas Deya, Mailyn por su apoyo, por haberme soportado y haber estado junto a mí estos 5 años en las buenas y malas.

A mi cuña (Yoandy), a Ale y a Yircy por su apoyo y preocupación.

A Adria y Frank por sus revisiones, por su apoyo y ayuda.

A mis amigos de Holguín por tantas cosas aprendidas y momentos juntos a las chicas y a los varones que tanta envidia les da que nos graduemos primero que ellos.

A Anie y Rene por su ayuda y por oírme y aguantarme cada vez que le llevaba una nueva historia de mi tesis.

A Henry Ernesto por su preocupación y ayuda.

A mi tutora Delmis por su ayuda y el tiempo dedicado.

A mis amigos del paquete de macromedia, a las chicas del apto y a todos los de mi grupo de los cuales he aprendido tantas cosas.

A Lara y Danyer en especial por su ayuda para la prueba de programación y poder llegar hasta aquí.

A la FEU por enseñarme entre otras cosas a no dormir, gracias a ella he conocido muchísima gente Ulises, Arcadio, Lilian, Humberto, Larín, Danelys, Aldo, Ramón gracias por los momentos, por haberlos conocido y aguantado y por acompañarme en este largo camino.

A la universidad y a la Revolución por la oportunidad de haber estudiado en este centro.

Dedicatoria

A mis padres por su amor y guía

A mi hermanita mayor

A mis abuelitos, mis tíos y primos

... A todos los que me ofrecieron su amor, amistad y me brindaron su apoyo para que viera realizados mis sueños...

Resumen

El presente trabajo está enmarcado en el objetivo de realizar el análisis y diseño del software educativo en forma de juego interactivo titulado: “Energía para Aprender”, relacionado con la Misión Revolución Energética en Venezuela, dirigido especialmente a niños de 9 a 12 años de edad, con el cual se pretende enseñar, de forma amena e interactiva, la importancia del aprovechamiento de la energía, la necesidad del cambio de bombillos, el uso de fuentes renovables y no renovables y otros temas de interés. Este documento recoge los resultados del trabajo realizado, donde se hace un estudio exhaustivo de los principales modelos, metodologías y estándares para el desarrollo de este tipo de software, así como de las tendencias y tecnologías actuales, permitiendo el análisis para seleccionar las más adecuadas que apoyen la solución del problema y las herramientas de desarrollo a emplear en la producción del mismo. Se emplea la metodología de desarrollo RUP y, como soporte de la misma, el lenguaje ApEM – L extensivo de UML, lográndose así identificar las funcionalidades y el modo en que estas deben ser implementadas para obtener un modelado que facilite, viabilice y apoye el proceso de producción. Como conclusiones generales, se determina que los objetivos propuestos han sido cumplidos de manera satisfactoria, aunque se propone una serie de recomendaciones para posteriores mejoras.

Índice

INTRODUCCIÓN.....	1
CAPÍTULO 1. FUNDAMENTACIÓN DEL TEMA	4
1.1 INTRODUCCIÓN DEL CAPÍTULO.....	4
1.2 SOFTWARE EDUCATIVO.....	4
1.2.1 Tecnología Multimedia: una nueva tecnología de comunicación e información	5
1.2.2 Tecnología Multimedia. Presentación de la información en forma de juegos	6
1.3 MODELOS, METODOLOGÍAS Y HERRAMIENTAS PARA LA CONSTRUCCIÓN DE SISTEMAS	6
1.3.1 Metodologías de desarrollo	7
1.3.1.1 Metodología a utilizar	13
1.3.2 Lenguajes de modelado	14
1.3.2.1 Lenguaje a utilizar	18
1.3.3 Herramientas de ingeniería asistida por computadoras (Computer Aided Software Engineering (CASE)) ...	19
1.4 HERRAMIENTAS PARA LA CREACIÓN DE SOFTWARE INTERACTIVO	20
1.4.1 Herramientas seleccionadas para el desarrollo del producto	22
1.5 CONCLUSIONES DEL CAPÍTULO	23
CAPÍTULO 2. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	24
2.1 INTRODUCCIÓN DEL CAPÍTULO	24
2.2 IDENTIFICACIÓN DE LA AUDIENCIA	24
2.3 ASPECTOS GENERALES.....	24
2.4 SOLUCIÓN PROPUESTA	25
2.5 ORIENTACIONES METODOLÓGICAS	26
2.6 MODELO DE DOMINIO	28
2.6.1 Identificación de los conceptos del dominio.....	28
2.6.2 Diagrama de clases del modelo del dominio	29
2.7 DESCRIPCIÓN DE LA FUNCIONALIDAD.....	30
2.7.1 Requerimientos funcionales	30
2.7.2 Requerimientos no funcionales	36

2.8 VISTA DE GESTIÓN DEL MODELO Y SUS SUBSISTEMAS.....	37
2.8.1 Vistas de presentación	39
2.8.1.1 Actores.....	40
2.8.1.2 Diagramas de estructura de presentación	40
2.8.1.3 Descripción textual de las vistas de presentación.....	46
2.8.1.4 Diagrama de estructura de navegación	67
2.9 CONCLUSIONES DEL CAPÍTULO	69
CAPÍTULO 3. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA	70
3.1 INTRODUCCIÓN DEL CAPÍTULO	70
3.2 MODELO DE DISEÑO.....	70
3.2.1 Diagrama de clases	70
3.2.2 Diagramas de interacción	76
3.2.3 Modelo de despliegue	81
3.4 CONCLUSIONES DEL CAPÍTULO	82
CAPÍTULO 4. ESTUDIO DE LA FACTIBILIDAD	83
4.1 INTRODUCCIÓN DEL CAPÍTULO	83
4.2 MÉTODO A UTILIZAR PARA LA ESTIMACIÓN DEL COSTO	83
4.2.1 Salidas externas	84
4.3 CÁLCULO DEL ESFUERZO, TIEMPO DE DESARROLLO, CANTIDAD DE HOMBRES Y COSTO.....	85
4.3.1 Beneficios tangibles.....	87
4.3.2 Beneficios intangibles.....	88
4.4 CONCLUSIONES DEL CAPÍTULO	88
CONCLUSIONES.....	89
RECOMENDACIONES	90
REFERENCIAS BIBLIOGRÁFICAS.....	91
BIBLIOGRAFÍA.....	93

Índice de figuras

FIGURA 1. DIAGRAMA DE CLASES DEL MODELO DE DOMINIO.....	30
FIGURA 2. REPRESENTACIÓN DE LA VISTA DE GESTIÓN DEL MODELO.....	39
FIGURA 3. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA PRESENTACIÓN.....	41
FIGURA 4. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA PRINCIPAL.....	41
FIGURA 5. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA SELECCJUGADOR.....	42
FIGURA 6. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA PRIMERNIVEL.....	42
FIGURA 7. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA SELECCJUEGOGUARDADO.....	43
FIGURA 8. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA CONFIGURACIÓN.....	43
FIGURA 9. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA PUNTAJE.....	44
FIGURA 10. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA SALIR.....	44
FIGURA 11. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA PREGUNTAS.....	44
FIGURA 12. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA HAPERDIDO.....	45
FIGURA 13. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA PREGUNTAEXTRA.....	45
FIGURA 14. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA MENÚ.....	45
FIGURA 15. DIAGRAMA DE ESTRUCTURA DE PRESENTACIÓN VISTA PAUSA.....	46
FIGURA 16. DIAGRAMA DE ESTRUCTURA DE NAVEGACIÓN GENERAL.....	68
FIGURA 17. DIAGRAMA DE ESTRUCTURA DE NAVEGACIÓN SUBSISTEMA NIVELES.....	69
FIGURA 18. DIAGRAMA DE CLASES VISTA PRESENTACIÓN.....	71
FIGURA 19. DIAGRAMA DE CLASES VISTA PRINCIPAL.....	72
FIGURA 20. DIAGRAMA DE CLASES VISTA SELECCJUGADOR.....	72
FIGURA 21. DIAGRAMA DE CLASES VISTA PRIMERNIVEL.....	73
FIGURA 22. DIAGRAMA DE CLASES VISTA HACIASEGNIVEL.....	74
FIGURA 23. DIAGRAMA DE CLASES VISTA IRA TERCER NIVEL.....	74
FIGURA 24. DIAGRAMA DE CLASES VISTA SELECCJUEGOGUARDADO.....	75
FIGURA 25. DIAGRAMA DE CLASES VISTA PUNTAJES.....	75
FIGURA 26. DIAGRAMA DE CLASES VISTA CONFIGURACIÓN.....	76
FIGURA 27. DIAGRAMA DE COLABORACIÓN VISTA PRESENTACIÓN.....	77
FIGURA 28. DIAGRAMA DE COLABORACIÓN VISTA PRINCIPAL.....	77
FIGURA 29. DIAGRAMA DE COLABORACIÓN VISTA SELECCJUGADOR.....	78
FIGURA 30. DIAGRAMA DE COLABORACIÓN VISTA PRIMERNIVEL.....	78
FIGURA 31. DIAGRAMA DE COLABORACIÓN VISTA HACIASEGNIVEL.....	79
FIGURA 32. DIAGRAMA DE COLABORACIÓN VISTA IRA TERCER NIVEL.....	79
FIGURA 33. DIAGRAMA DE COLABORACIÓN VISTA SELECCJUEGOGUARDADO.....	80
FIGURA 34. DIAGRAMA DE COLABORACIÓN VISTA CONFIGURACIÓN.....	80
FIGURA 35. DIAGRAMA DE COLABORACIÓN VISTA PUNTAJES.....	81
FIGURA 36. DIAGRAMA DE DESPLIEGUE.....	81

Índice de tablas

TABLA 1. DESCRIPCIÓN TEXTUAL DE LA VISTA PRESENTACIÓN	46
TABLA 2. DESCRIPCIÓN TEXTUAL DE LA VISTA PRINCIPAL.....	47
TABLA 3. DESCRIPCIÓN TEXTUAL DE LA VISTA SELECCJUGADOR	48
TABLA 4. DESCRIPCIÓN TEXTUAL DE LA VISTA PRESENTPRIMERNIVEL.....	50
TABLA 5. DESCRIPCIÓN TEXTUAL DE LA VISTA PRIMERNIVEL.....	52
TABLA 6. DESCRIPCIÓN TEXTUAL DE LA VISTA CIERREPRIMERNIVEL	58
TABLA 7. DESCRIPCIÓN TEXTUAL DE LA VISTA SELECCJUEGOGUARDADO	59
TABLA 8. DESCRIPCIÓN TEXTUAL DE LA VISTA CONFIGURACIÓN.....	60
TABLA 9. DESCRIPCIÓN TEXTUAL DE LA VISTA PUNTAJE.....	62
TABLA 10. DESCRIPCIÓN TEXTUAL DE LA VISTA PREGUNTA EXTRA	62
TABLA 11. DESCRIPCIÓN TEXTUAL DE LA VISTA PREGUNTAS	63
TABLA 12. DESCRIPCIÓN TEXTUAL DE LA VISTA HAPERDIDO	64
TABLA 13. DESCRIPCIÓN TEXTUAL DE LA VISTA PAUSA.....	65
TABLA 14. DESCRIPCIÓN TEXTUAL DE LA VISTA MENÚ.....	66
TABLA 15. DESCRIPCIÓN TEXTUAL DE LA VISTA SALIR	66
TABLA 16. MULTIPLICADORES DE ESFUERZO.....	85
TABLA 17. FACTORES DE ESCALA	86

Introducción

La electricidad es la fuente de energía que, como magia, permite la satisfacción de todas las necesidades de la humanidad. Una vez producida y a nuestra disposición se transforma en un mundo de diversas manifestaciones de la cual se obtiene: calor, luz, sonidos, imágenes, movimiento y todos los demás elementos esenciales para el desarrollo de la sociedad. Desde que despertamos hasta el momento en que nos acostamos, y aún durante el sueño, estamos acompañados en el hogar, en la comunidad, en la escuela, en la industria, en el transporte, en todas partes, por un constante flujo de energía, necesario para el desarrollo y crecimiento de un país.

La generación de electricidad se produce a partir de la utilización de combustibles fósiles como el petróleo y algunos derivados, procedentes de fuentes no renovables de energía, lo que equivale a decir que en un futuro se agotarán, de ahí que la disponibilidad de energía se haya convertido en un problema crucial.

El mundo está obligado a buscar una solución al respecto, teniendo en cuenta conceptos como el valor de la energía que se consume, su vital ahorro, eficiencia y uso racional e inteligente, beneficiándose con ello tanto la población como el estado. Se debe crear y mantener una cultura de apoyo hacia los recursos de los que disponemos, no como un lujo, ni como una consecuencia del desarrollo de la ciencia y la tecnología en la modernidad, sino por una gran necesidad, porque la vida de la humanidad en el futuro pelagra. Actualmente, varios países han encaminado sus esfuerzos por lograr esa conciencia energética en el pueblo, un ejemplo fehaciente de ello lo constituye Venezuela, quien está llevando a cabo un conjunto de acciones que realzan la importancia del aprovechamiento racional de la energía.

Las principales tareas que se están desarrollando en este sentido las impulsa el Ministerio del Poder Popular para la Energía y Petróleo de Venezuela (MENPET), quien cuenta con una Oficina de Comunicación y Relaciones Institucionales, que tiene entre sus misiones difundir a diferentes sectores información sobre la Revolución Energética, como vía alternativa para que se comiencen a romper paradigmas en cuanto al uso de la energía. Para lograr el objetivo anteriormente expuesto, se desea brindar materiales didácticos con informaciones que posibiliten el aprendizaje de conocimientos esenciales relacionados con la temática energética, puesto que actualmente no existen medios educativos que contribuyan a elevar el nivel cognitivo acerca de estos temas que afectan a todos.

Entre los proyectos propuestos por el ministerio, se presenta el de realizar un producto de software educativo en forma de juego interactivo, dirigido especialmente para los niños, que contenga estos contenidos de manera interesante, de forma tal que les sea atractivo, los motive a aprender y a comprender que pueden dar un gran aporte según las necesidades del país.

Animados por el deseo de fortalecer los tradicionales lazos de amistad, se han creado desde hace más de 5 años una serie de convenios de cooperación entre La República de Cuba y La República Bolivariana de Venezuela como parte de la Alternativa Bolivariana para las Américas (ALBA) y en este marco de convenios, partiendo de las ideas anteriormente señaladas, se le solicita la creación de este producto educativo que cumpla con las expectativas del MEMPET, a la Universidad de las Ciencias Informáticas (UCI), reconocido centro cubano de altos estudios con el que se han logrado cambios en el proceso educativo de la enseñanza superior y la industria informática, además ha acumulado una vasta experiencia en la creación de software, mejorando y perfeccionando con el paso del tiempo la calidad en el proceso de desarrollo. El equipo de programadores al que le fue asignado precisa de una visión lo más profunda posible del mismo, con las funciones requeridas, características, rendimiento, comportamiento bien definidos y enmarcados en un modelo pedagógico. Se debe intensificar y centrar en el producto el proceso de captura de requisitos, para luego analizarlos y traducirlos en una representación del software donde se pueda evaluar su calidad antes de que comience la codificación.

Si no se tienen en cuenta estos aspectos previos a la compleja construcción de una aplicación, los errores son muy probables, trayendo como resultado tiempo y dinero perdidos, frustración personal y clientes descontentos, de ahí que se le deba conceder gran importancia a este proceso. A esto hace alusión Brian Lawrence cuando expresó: “Gastamos tiempo - la mayor parte del tiempo del proyecto - no implementando o probando, sino intentando decidir qué construir.”

Teniendo en cuenta lo anteriormente expuesto se plantea como **problema científico** que: No existe la documentación requerida que permita guiar a los desarrolladores en el proceso de implementación de un software educativo en forma de juego interactivo sobre temas relacionados con la Misión Revolución Energética.

El problema científico se enmarca en el **objeto de estudio**: Proceso de desarrollo de Software educativo.

Para resolver el problema se plantea el siguiente **objetivo general**: Realizar el análisis y diseño de un software educativo en forma de juego sobre temas relacionados con la Misión Revolución Energética.

Situándose como **campo de acción**: El proceso de análisis y diseño de un software educativo.

La **idea a defender** radica en que, si se realiza el análisis y diseño del software educativo sobre temas relacionados con la Misión Revolución Energética, se podrá realizar la implementación del producto por parte del equipo desarrollador.

Para lograr el objetivo se plantean las siguientes **tareas de investigación**:

- Realizar un análisis y selección de las tecnologías y herramientas a emplear para el desarrollo del producto educativo.
- Realizar un estudio y selección de la metodología de análisis y diseño de sistemas informáticos, que facilite la creación y garantice la calidad del software.
- Realizar análisis y diseño de la aplicación.

El documento se encuentra estructurado en: la presente introducción, 4 capítulos, conclusiones, recomendaciones, referencias bibliográficas, bibliografía, glosario de términos y anexos.

En el **Capítulo 1: Fundamentación teórica** se presenta un análisis de las tendencias actuales del progreso del software con tecnología multimedia, así como de las metodologías de desarrollo y herramientas más comunes con el objetivo de seleccionar las más adecuadas o idóneas para ser empleadas en la solución del problema.

En el **Capítulo 2: Descripción de la Solución Propuesta** se realiza el modelado del entorno que se desea automatizar a través de un modelo de dominio. Se describen los requerimientos funcionales y no funcionales. Se representa, a través de la Vista de gestión del modelo, el sistema organizado en subsistemas y sus relaciones, así como las vistas de presentación que estos agrupan.

En el **Capítulo 3: Construcción de la Solución Propuesta** se realiza una construcción de la solución propuesta, exponiendo los artefactos esenciales del modelo de diseño que guiarán el trabajo de la implementación. Se presentan, fundamentando el proceso de desarrollo, los diagramas de clases y de interacción y queda plasmado el hardware necesario para el funcionamiento de la aplicación.

En el **Capítulo 4: Estudio de factibilidad** se lleva a cabo un estudio y análisis completo de la factibilidad, teniendo en cuenta aspectos que influyen en la ejecución de una aplicación como la planificación, el tiempo de desarrollo, los costos y los beneficios tangibles e intangibles que reportará el software.

Capítulo 1

Fundamentación del Tema

1.1 Introducción del capítulo

En este capítulo se realiza un estudio y valoración sobre el desarrollo actual del software y su evidente impacto hoy día, haciendo énfasis en el software educativo y su creación, usando tecnología multimedia, así como en los juegos interactivos como una manera de presentar la información al usuario. Se expone el resultado del análisis de algunas metodologías de desarrollo de software más comúnmente usadas, además de varias tecnologías y herramientas afines, teniendo en cuenta sus características, importancia, ventajas y desventajas, para seleccionar cuáles utilizar para el modelado y confección del producto.

1.2 Software Educativo

A lo largo de la última década, como consecuencia del avance tecnológico, los medios de enseñanza han evolucionado, facilitando la inclusión de los medios informáticos en el aula, que ayudan a favorecer el aprendizaje, pero: “No es la tecnología la que debe salir a la búsqueda de una aplicación educativa. Por el contrario, la idea es partir de la educación, la que ante una necesidad busca un soporte tecnológico para que sea socio en la construcción significativa del aprende.” [1]

La necesidad de introducir la tecnología en los procesos docentes se ha hecho evidente, dentro de los cuales juega un papel fundamental el software educativo.

“Software educativo, programas educativos y programas didácticos son programas para ordenador creados con una finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje.” [2]

Un software educativo no es más que una herramienta que facilitará impartir el conocimiento a una audiencia, además de adaptarlo al ritmo de trabajo y las necesidades de cada educando, para lograr diversas alternativas de aprendizaje en él y generar procesos cognitivos en su intelecto.

En la actualidad, las tendencias actuales de software educativo se inclinan hacia la preferencia de presentación de contenidos que utilizan las técnicas hipertexto, multimedia o hipermedia, esto ocurre

debido a la inclusión de las Tecnologías de la Informática y las Comunicaciones (TIC) que ha propiciado un marco ideal para que estas se desarrollen.

1.2.1 Tecnología Multimedia: una nueva tecnología de comunicación e información

La tecnología multimedia, con un gran potencial para transformar aspectos importantes, se ha convertido en el más reciente fenómeno tecnológico y cultural. Ha encontrado aplicaciones de una manera muy rápida en diferentes campos como la educación, la capacitación y la instrucción, así como en la publicidad y marketing, mostrando grandes beneficios aceptados por la sociedad.

“Multimedia es el uso de texto y gráficas, recursos tradicionales en una computadora, combinados con el video y sonido, nuevos elementos integrados bajo el control de un programa que permite crear aplicaciones enfocadas básicamente a la capacitación y el ofrecimiento de servicios... es una tendencia de mezclar diferentes tecnologías de difusión, de información, impactando varios sentidos a la vez, para lograr un efecto mayor en la comprensión del mensaje” [3]

Los sistemas con esta tecnología permiten un activo intercambio de información a través de sistemas informáticos y pueden resultar muy útiles en los contextos educativos, no sólo posibilitando ver y oír, sino también interactuar con el objeto en cuestión. Son materiales que integran diversos elementos textuales (secuenciales e hipertextuales) y audiovisuales (voz, sonido, música, video, animación, fotografía, imagen), además de la capacidad de ser un medio interactivo, dinámico y lleno de estímulos, donde el usuario experimenta con el sistema, sintiéndose participante y no un simple espectador.

“... la multimedia representa una ventaja como SOPORTE al proceso educativo, pues presenta y manipula la información en un lenguaje contemporáneo, que además permite a maestros y alumnos jugar con su estructura para lograr diferentes objetivos pedagógicos.” [4]

La tecnología multimedia permite navegar por el programa y buscar la información sin tener que recorrerlo todo, se puede aplicar como medio de aprendizaje e informativo, ofreciéndonos diferentes productos con actividades que facilitan la interacción tales como: preguntas, ejercicios, simulaciones, juegos, programas de aprendizaje y materiales de referencia, entre otros.

Es la base de los juegos de video, tiene aplicaciones en pasatiempos, cuentos y juegos infantiles interactivos, exploración de museos y ciudades a manera de visitas. [3]

1.2.2 Tecnología Multimedia. Presentación de la información en forma de juegos

“Al igual que en casi todas las áreas de nuestra vida, la única manera de conseguir que alguien haga algo es que desee hacerlo ¿por qué no hacer divertida la enseñanza?” [5]

La tecnología multimedia en forma de juego interactivo, hace más ameno el aprendizaje, demostrando que una clave para aprender fácilmente es asociarlo con entretenimiento. En la actualidad, esta manera de ser presentada al usuario, está probando de forma creciente su efectividad, no sólo como medio de ocio y entretenimiento, sino también educativo.

Hoy día muchas veces se usa el término multimedia para referirse al software interactivo con esta tecnología. ¿Y qué es interactividad en este contexto?

Se define la interactividad como la demanda de acción que efectúa el software al usuario. Esta acción-reacción puede tener distintos medios de expresión: tocar la pantalla, entrar texto en un determinado campo, realizar determinadas manipulaciones usando el mouse, ya sea clic, doble clic, arrastrar y tirar, etc. Desde el punto de vista del usuario, interactividad es la cantidad de control que este tiene sobre los contenidos. La verdadera interactividad, en cambio, implica una modificación, se pasa de una presentación lineal donde todo el control es del autor a la multilinealidad donde el control es del usuario. El software interactivo con tecnología multimedia implica un diálogo entre el usuario y los contenidos, ya que es este grado de actividad lo que va a hacerlo aprender y retener más rápido aquello que se propone. [6]

La finalidad de dichas aplicaciones es estimular el desarrollo en los niños y niñas de su propia capacidad en una amplia gama de situaciones y circunstancias, de forma tal que puedan aprender disfrutando y para lograr su diseño exitoso se deben tener en cuenta aspectos fundamentales como la apariencia, la forma estéticamente agradable y atractiva.

Luego del estudio del arte realizado se propone el desarrollo de un software educativo con tecnología multimedia presentado como un juego con entorno interactivo, que de una forma amena e instructiva convierta al usuario en el dueño de la secuencia y del tiempo dedicado al contenido.

1.3 Modelos, metodologías y herramientas para la construcción de sistemas

Debido al auge del software educativo durante los últimos años, gran parte de ellos han sido realizados en forma desorganizada y poco documentada. La diversidad y complejidad de las actividades, trae aparejado que la elaboración del producto sea difícil y riesgoso de controlar y si no somos capaces de llevar una guía de por medio, el resultado será la insatisfacción del cliente.

Para el desarrollo de un software con calidad se necesita de un plano en el que apoyarse, de un previo análisis de los aspectos generales. Habitualmente nos olvidamos de estas cuestiones de vital importancia y utilizamos herramientas a nivel de implementación, donde no se recoge información alguna, haciendo complicados la calidad y el mantenimiento.

“Para desarrollar un proyecto de software es necesario establecer un enfoque disciplinado y sistemático. Las metodologías de desarrollo influyen directamente en el proceso de construcción y se elaboran a partir del marco definido por uno o más ciclos de vida”. [7]

1.3.1 Metodologías de desarrollo

Una metodología de desarrollo establece la necesidad de considerar un diseño previo a la construcción del sistema y ofrece vías para recoger las especificaciones. Constituye un proceso para la producción organizada del software, empleando una serie de pasos, con técnicas y notaciones asociadas a cada uno.

...el software debe ser desarrollado con una metodología que contemple los aspectos pedagógicos y fundamentalmente hay que evaluarlo en un contexto similar al de uso... [8]

Las metodologías han probado ser realmente útiles y brindar un valor agregado al desarrollo del software; proporcionan información sobre las estrategias y técnicas didácticas, así como las herramientas que se utilizan. En los últimos años ha existido una tendencia al proceso de ingeniería, por lo que ya se han propuesto diferentes metodologías, cada una de las cuales cuenta con sus propias ventajas y desventajas, las que debemos tener en cuenta, analizarlas y determinar su alcance, para luego llegar a la conclusión de cuál usar que se adapte más a nuestro medio y a nuestra aplicación. A continuación se hace una valoración sobre algunas comúnmente usadas.

“...ninguna metodología hace el trabajo por sí sola, pero te podrá ayudar”. [9]

Programación Extrema

La metodología Extreme Programming (XP), como es llamada en inglés, ha recibido gran atención, convirtiéndose en una de las más exitosas en la actualidad. Se debe destacar su flexibilidad, ya que muestra efectividad cuando los requisitos son muy cambiantes. Todo proyecto que la siga se ha de dividir en iteraciones de aproximadamente 3 semanas de duración, tiene como principal objetivo producir el software a corto plazo, es una metodología liviana, ágil y está orientada más a las personas por sobre las herramientas, asumiendo las características de cambio acelerado y reemplazando la documentación detallada por la comunicación cara a cara. Se emplea para proyectos de un grupo

pequeño de personas. Hacen poco énfasis en la arquitectura de software, tiene pocos roles y artefactos, como particularidad debe tener el cliente como parte del equipo de desarrollo.

Consta de cuatro fases:

- *Planificación:* Definir las historias de usuario, crear un plan de publicaciones, iteraciones y reuniones bastante frecuentes, estimar la velocidad, XP aconseja la programación en parejas pues incrementa la productividad y la calidad del software desarrollado.
- *Diseño:* Se sugieren diseños simples y sencillos, elaborar glosarios de términos, identificar riesgos en pareja y reducirlo al máximo, revisar de nuevo estos códigos para procurar optimizar su funcionamiento.
- *Desarrollo o codificación:* El cliente es la parte más importante del equipo de desarrollo, su presencia es indispensable en las distintas fases de XP. La codificación debe hacerse teniendo en cuenta estándares de codificación ya creados. Se sugiere un modelo de trabajo colectivo, usando repositorios de código.
- *Pruebas:* Se realizan pruebas a los principales procesos, de tal manera que adelantándose en algo hacia el futuro, se puedan hacer pruebas de las fallas que pudieran ocurrir, o sea obtener los posibles errores.

XP plantea la planificación como un permanente diálogo entre los empresarios y desarrolladores, en la que los primeros decidirán: [10]

- *Alcance:* ¿Qué es lo que el software debe de resolver para que genere valor?
- *Prioridad:* ¿Qué debe ser hecho en primer lugar?
- *Composición de las versiones:* ¿Cuánto es necesario hacer para saber si el negocio va mejor con software que sin él?
- *Fechas de versiones:* ¿Cuáles son las fechas en la presencia del software?
- *Consecuencias:* Informar sobre las consecuencias de la toma de decisiones.
- *Procesos:* ¿Cómo se organizan el trabajo y el equipo?
- *Programación detallada:* Dentro de una versión ¿Qué problemas se resolverán primero?

Esta metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. Además

sostiene que el código es el único entregable que realmente importa, desplazando el rol del análisis y diseño durante la creación de software.

Metodología de Administración de Relaciones

Esta metodología, comúnmente llamada en inglés: Relationship Management Methodology (RMM), se define como un proceso de análisis y diseño, basado en conceptos del Modelo de diseño de Hipertexto (HDM) y desarrollo de aplicaciones de estructura estable. Se centra en el diseño, desarrollo y fases de construcción. Impone una disciplina de trabajo sobre el proceso de desarrollo de software, con el objetivo de asegurar la obtención de una aplicación que satisfaga los requerimientos del usuario y reúna estándares aceptables de calidad.

RMM constituye una metodología tentadora para el desarrollo del proceso por el desglose de las fases de la producción y la incorporación de diagramas para el diseño de la presentación, el comportamiento dinámico y la estructura de la navegación. [11]

Está basada en el tradicional Modelo Entidad - Interrelación (MER), donde se modela el dominio de información a considerar; y el diseño de M-Slices, estructuras conceptuales que permiten modelar las unidades de presentación de la aplicación.

Es una metodología que asume un ciclo de vida completo, compuesto por siete etapas, en las que se va modelando la estructura de la aplicación, antes de estas etapas se determinan objetivos, se analizan necesidades y tipos de navegación. [12]

- *Viabilidad*: se determinan los objetivos y el análisis de necesidades.
- *Análisis de los requerimientos de navegación*: Se determinan los tipos de navegación que se utilizarán en el proyecto.

En la etapa de diseño, la más importante, se recogen los tres primeros pasos:

- *Diseño del Diagrama Entidad Relación*: realizada una vez elaborado el diagrama de burbujas y normalización de las entidades.
- *Diseño del Diagrama de Slices*: se determina la forma en que la información en las entidades elegidas se presentará a los usuarios, y la forma en que pueden acceder a él.
- *Diseño del Diagrama RMDM o diseño de navegación*: se diseñan los caminos que permitan la navegación de hipertexto. Cada relación asociativa que aparecen en el diagrama ER se analiza y se indica el tipo de navegación entre las entidades.

- *Diseño del protocolo de conversión:* Cada elemento del diagrama RMDM se transforma en objetos.
- *Diseño de la interfaz de usuario:* Se desarrollan las pantallas para mostrar información al usuario.
- *Diseño del funcionamiento en runtime:* Se determinan qué botones de la pantalla serán visibles y cuáles invisibles.
- *Construcción:* Se desarrolla la aplicación sobre la base del diagrama RMDM en un lenguaje seleccionado.

Al final se realizan pruebas y evaluaciones experimentales para corregir errores encontrados.

“...En el lado opuesto del espectro, un trabajo artístico puede tener una estructura bastante difusa en la cual no se observen cambios frecuentes a través del tiempo, haciendo de RMM poco aplicable.” [13]

Marco de Solución de Microsoft

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. [14]

Esta metodología, más conocida por su nombre en inglés: Microsoft Solution Framework (MSF), se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas. Los modelos de los que se compone son los encargados de planificar las diferentes partes implicadas en el desarrollo del proyecto, estos son: Modelo de arquitectura, Modelo de equipo, Modelo de proceso, Modelo de gestión del riesgo, Modelo de diseño de proceso y Modelo de aplicación.

Consta de 4 fases: [14]

- *Estrategia y alcance:* Elaboración y aprobación del documento de alcance y estrategia definitivo, formación del equipo de trabajo y distribución de competencias y responsabilidades, elaboración del plan de trabajo, de la matriz de riesgos y del plan de contingencia.
- *Planificación y Prueba de concepto:* Documento de planificación y Diseño de arquitectura, Documento de plan de laboratorio, Prueba de concepto
- *Estabilización:* Selección del entorno de prueba piloto, Gestión de incidencias, Revisión de la documentación final de arquitectura, Elaboración de la documentación de formación y operaciones, Elaboración del plan de despliegue, Elaboración del plan de formación.

- *Despliegue:* Se llevarán a cabo los planes diseñados en la anterior, principalmente el de despliegue y el de formación: registro de mejoras y sugerencias, incluyendo mejoras aportadas por los fabricantes de software, revisión de las guías y manuales de usuario, rectificación de errores y obtención de los documentos de formación definitivos, entrega de los documentos definitivos, revisión, establecimiento de los estándares de calidad, entrega del proyecto y cierre del mismo, en base a la información y experiencia obtenidos.

MSF tiene las siguientes características:

- *Adaptable:* es parecido a un compás, usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
- *Escalable:* puede organizar equipos tan pequeños entre 3 ó 4 personas, así como también, proyectos que requieren 50 o más personas.
- *Flexible:* puede ser utilizada en el ambiente de desarrollo de cualquier cliente.
- *Tecnología Agnóstica:* puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

Proceso Unificado de Desarrollo de Software

Rational Unified Process (RUP), constituye la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos. Proporciona una aproximación disciplinada a la asignación de tareas y responsabilidades. RUP actúa como modelo y puede ser adaptado y extendido, requiere un equipo de trabajo capaz de administrar un proceso complejo que basa su desarrollo en ciclos que al concluir originan una versión del producto. [15]

RUP toma en cuenta las mejores prácticas en el modelo de desarrollo de software:

- Desarrollo de software en forma iterativa.
- Manejo de requerimientos.
- Utiliza arquitectura basada en componentes.
- Modela el software visualmente.
- Verifica la calidad del software.
- Controla los cambios.

Cada ciclo RUP consta de cuatro fases que llevan a cabo el cumplimiento de los objetivos propuestos y en su culminación el alcance de un hito siendo respectivamente: objetivos del ciclo de vida, arquitectura del ciclo de vida, funcionalidad operativa inicial y la versión del producto, estas son:

- Concepción o inicio: Comprender los requisitos y determinar visión y alcance del proyecto.
- Elaboración: Asignar recursos, especificar las características y definir la arquitectura.
- Construcción: Implementación, construir el producto operacional.
- Transición: Hacerlo operativo para los usuarios, nivel correcto de calidad para entregar.

En RUP se han agrupado las actividades en grupos lógicos definiéndose nueve flujos de trabajo principales. Los seis primeros son los flujos de ingeniería, estos son:

- Modelamiento de negocio: se entienden las necesidades del negocio.
- Requerimientos: se trasladan las necesidades del negocio a un sistema automatizado.
- Análisis y diseño: se trasladan los requerimientos dentro de la arquitectura de software.
- Implementación: se crea un software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- Prueba (Testeo): se debe asegurar que el comportamiento requerido es el correcto y que todo lo solicitado está presente.
- Instalación del software

Los tres últimos son de apoyo:

- Administración del proyecto: para el control de los horarios y recursos.
- Administración de configuración y cambios: permite guardar todas las versiones del proyecto.
- Ambiente: se realiza la administración del ambiente de desarrollo.

RUP es un proceso y en su modelación define como sus principales elementos a: los trabajadores, que serán las personas involucradas en cada proceso; los flujos de actividades, que definen cuando se realizará cada secuencia de actividades; las actividades que son los procesos que se llegan a determinar en cada iteración y los artefactos que se generan en cada flujo, pueden ser un documento, modelo o un elemento de modelo. (Ver Anexo 1)

El ciclo de vida de RUP se caracteriza por: [16]

- *Dirigido por casos de uso:* Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí los casos de uso guían el proceso de desarrollo y representarán su realización los diferentes modelos que se obtienen, como resultado de los diferentes flujos de trabajo.
- *Centrado en la arquitectura:* La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente.
- *Iterativo e Incremental:* RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Es práctico dividir el trabajo en miniproyectos, y cada uno de ellos es una iteración planificada que resulta en un incremento.

Es recomendable que a las iteraciones se les clasifique y ordene según su prioridad y luego cada una se convierta en un entregable al cliente, trayendo como beneficio la retroalimentación. Una particularidad de esta metodología es que se hace exigente el uso de artefactos, siendo por este motivo, una de las más importantes para alcanzar un grado de certificación en el desarrollo del software.

1.3.1.1 Metodología a utilizar

Luego del estudio realizado anteriormente, donde se pudieron analizar varias metodologías y comparar sus diferentes características y particularidades de empleo en varias situaciones, se determinó cuáles no se ajustan al proceso en cuestión, en el caso de XP a pesar de su utilidad, tiene como principal objetivo producir el software a corto plazo sin importar la documentación, está más orientada a las personas que a las herramientas y plantea que el usuario debe ser un integrante más del equipo de desarrollo, lo cual sería muy difícil en este caso. RMM por su parte se centra en el diseño, desarrollo y fase de construcción, impidiéndonos así su uso, pues para el cumplimiento del objetivo de esta propuesta se deben hacer estudios previos al diseño. Mientras MSF, además de proponer la realización de gran cantidad de documentación no aplicable para la solución de este problema, se centra en el modelo de equipo, dejando en un segundo plano las elecciones tecnológicas.

Se seleccionó la metodología RUP como la más adecuada, ya que además de definir ciclos de trabajo y permitir describir lo que se espera del software, permite un desarrollo centrado en una arquitectura sólida. Es adaptable para proyectos a largo plazo, proporciona un trabajo organizado por partes pequeñas que permiten una mejor calidad del producto final, a través del control de riesgos y la aplicación de pruebas.

1.3.2 Lenguajes de modelado

La metodología del Proceso Unificado (RUP) se aplica a una buena cantidad de productos y procesos de software en el mundo, pero para lograr su aplicación adecuada se hace necesario un soporte o base, con el que se alcance la compatibilidad en el análisis y diseño orientado a objetos. Es una metodología con un enfoque basado en modelos y para tal fin necesita un lenguaje bien definido para realizar el modelado del sistema a implementar; por lo que se decide realizar un análisis, basado en un estudio y evaluación comparativos, de variantes más difundidas en la actualidad, para de esta forma seleccionar el más idóneo.

A través de la historia se han desarrollado varios modelos de procesos de software, es por ello que existen diversas propuestas de lenguajes. En los últimos años, su construcción se ha convertido en una herramienta habitual en distintos ámbitos tales como el de la ingeniería de requisitos y el del modelado de procesos en organizaciones.

Un lenguaje de modelado es un conjunto de elementos o notaciones que se disponen de modo que ayuden a modelar parte de una aplicación. No todos cubren los mismos aspectos de un sistema, suelen estar ligados a metodologías de desarrollo y/o paradigmas para avanzar de una especificación inicial a un plan de implementación que debe conocer todo el equipo de desarrolladores. A continuación se muestra una breve descripción de importantes propuestas, dando así a conocer sus características principales.

Lenguaje Unificado de Modelado (UML)

Es el lenguaje de modelado de sistemas de software más conocido y utilizado por los modeladores en la actualidad. No tiene propietario y está basado en el común acuerdo de gran parte de la comunidad informática. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software, no pretende ser un método de desarrollo completo, pues no incluye un proceso de desarrollo paso a paso. UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas y cuenta con reglas para combinar tales elementos.

Este lenguaje propone un vocabulario que incluye tres categorías: elementos, relaciones y diagramas (Ver anexo 2). Los elementos son abstracciones de cosas reales o ficticias (objetos, acciones), las relaciones van a ser, como su nombre lo indica, las que enlacen los elementos entre sí y los diagramas son colecciones de elementos y sus relaciones.

UML consta de varias áreas conceptuales, estas son: estructura estática, comportamiento dinámico, construcciones de implementación, organización del modelo y mecanismos de extensión. A su vez, consta de ocho vistas: estática, de casos de uso, de implementación, de despliegue, de máquinas de estado, de actividad, de interacción y de gestión del modelo; para la modelación de los productos. [17]

El lenguaje unificado está diseñado a través de un lenguaje de diagramas y artefactos, distribuidos por cada una de las vistas, fácilmente ajustables para especificar aspectos distintivos de un sistema a modelar. Dichos diagramas se agrupan en cuatro categorías: de caso de uso, estructurales, de comportamiento y de implementación, siendo el segundo y el tercero los que interactúan directamente con las descripciones de los modelos estáticos estructurales y de comportamiento dinámicos.

Diagramas de estructura estática:

- *Diagrama de clases*: Clases, interfaces y colaboraciones; así como sus colaboraciones.
- *Diagrama de objetos*: Objetos y sus relaciones.
- *Diagrama de casos de uso*: Casos de uso, actores y sus relaciones.

Diagramas de comportamiento:

- *Diagramas de interacción (secuencia y colaboración)*: Objetos y sus relaciones, incluyendo los mensajes que pueden ser enviados entre ellos.
- *Diagrama de estados*: Muestra una máquina de estado que consta de estados, transiciones, eventos y actividades.
- *Diagrama de actividad*: Es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro de un sistema.

Diagramas de implementación:

- *Diagrama de componentes*: Organización y las dependencias entre un conjunto de componentes.
- *Diagrama de despliegue*: Configuración de nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos.

UML incluye todos los conceptos que se consideran necesarios para utilizar un proceso moderno iterativo, se basa en construir una sólida arquitectura para resolver requisitos dirigidos por casos de uso. Incluye aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. Se puede aplicar en gran variedad de formas para dar soporte a una metodología de desarrollo pero no especifica en sí mismo qué metodología o proceso usar. [18]

Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Además no es una guía para realizar el análisis y diseño, sino un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos.

UML no soporta todos los planteamientos de las aplicaciones con tecnología multimedia para modelar aspectos de la interfaz de usuario, sus características no se aplican en los entornos multimedia. Fue por esta y otras razones, gracias a la riqueza de UML y a sus facilidades de extensión, que se toma como base en el necesario desarrollo de un lenguaje para este tipo de productos: OMMMA-L, que se presenta como algo eficientemente realizable y facilita un gran rango de aspectos significativos.

Lenguaje orientado a objetos para la modelación de aplicaciones multimedia (OMMMA-L)

No podemos decir que OMMMA - L es un lenguaje nuevo, sino una extensión del UML que permite operar con productos con tecnología multimedia que se desarrollen en ambientes orientados a objetos y no es necesario aprenderlo, sino centrarse en la lógica de funcionamiento de este tipo de software e interpretar sus características extendidas.

OMMMA-L se encuentra sustentado en su modelación en cuatro vistas fundamentales, donde cada una se asocia a un tipo de diagrama en particular, modifica los diagramas originales de UML de: clases, secuencia y estado. Añade como parte de la vista de presentación espacial un nuevo diagrama: el diagrama de presentación, para la representación espacial de los elementos visuales del futuro software multimedia. [19]

- *Vista Lógica*: modelada a través del diagrama de clases, utilizando las mismas notaciones del de UML, pero incorporando las clases correspondientes a las medias. Divide en dos áreas dicho diagrama: una para la jerarquía de los tipos de media y otra para la modelación de la estructura lógica del dominio de la aplicación.
- *Vista de Presentación espacial*: modelada a través de los diagramas de presentación, los cuales tienen el propósito de declarar las interfaces de usuario con un conjunto de estructuras delimitadas en tamaño y área, dividiéndose en objetos de visualización e interacción.

- *Vista de Comportamiento temporal predefinido*: modelada por el diagrama de secuencia, el cual modela una secuencia de una presentación predefinida dentro de una escena, donde todos los objetos se relacionan al mismo eje del tiempo.
- *Vista de Control Interactivo*: modelado a través del diagrama de estado, sintácticamente igual al de UML, mas con la diferencia semántica de que el comportamiento especificado por el diagrama de secuencia se provoca automáticamente cuando se entra al estado correspondiente donde se hace referencia.

En fin con OMMMA - L se puede modelar la estructura a través de diagramas de objetos y clases, mientras que el comportamiento puede ser descrito en los diagramas de interacción, estado y actividad, además integra el comportamiento interactivo con el de procedimientos temporales para lograr la descripción de aplicaciones que reaccionan ante eventos externos y producen ejecuciones dinámicas predecibles en tiempo de ejecución.

En la última década se ha logrado en la modelación de entornos educativos, la incorporación de lenguajes de propósito general como UML, o sus extensiones como OMMMA - L. No obstante estos lenguajes no logran denotar todos los elementos estructurales, lógicos, pedagógicos y de patrones de ingeniería de las aplicaciones educativas cubanas en sus gráficos y semánticas utilizadas; lo que produce una deficiente documentación y modelación de este tipo de software. [20]

Lenguaje para la modelación de Aplicaciones Educativas Multimedia (ApEM - L)

Este lenguaje surge como una extensión de UML, tomando como bases teóricas principales a OMMMA- L, con sus elementos más significativos, y al estándar OCL-2.0, respetando lo que establece; dando así una nueva solución a la modelación de aplicaciones educativas multimedia en el entorno productivo de la Universidad de las Ciencias Informáticas (UCI).

ApEM - L no modifica la semántica de UML, sino que trabaja en estereotipos restrictivos, por lo que a su vez produce modificaciones descriptivas y decorativas en la representación de los componentes del lenguaje base. Utiliza el estándar internacional OCL (lenguaje de control de operaciones de un sistema informático), para la modelación de la programación orientada a objetos, además de que puede utilizar para su representación cualquiera de las herramientas de modelado que existen actualmente.

Áreas conceptuales de ApEM – L

Los conceptos y modelos de ApEM - L pueden agruparse en las siguientes áreas conceptuales:

- *Estructura lógica*: Compuesta por la vista estática y la vista de arquitectura.

- *Comportamiento dinámico*: Descrito por la vista de comportamiento y está compuesta por los diagramas: de actividad, de secuencia, de colaboración y de estados.
- *Gestión del modelo*: Compuesto por la vista de gestión del modelo y la vista de presentación.

Tanto ApEM-L como UML no establecen ninguna línea entre los diferentes conceptos y construcciones del lenguaje, pero por conveniencia del primero, este se ha dividido en varias vistas, modelando cada una de estas construcciones que representan un aspecto del sistema. La división ha sido sobre la base de las áreas conceptuales ya presentadas. [20]

Vistas y diagramas de ApEM – L

- Vista estática: compuesta por el diagrama de clases.
- Vista de arquitectura: compuesta por el diagrama de componentes y el diagrama de despliegue.
- Vista de comportamiento: compuesta por los diagramas de actividades, de estado, de secuencia y de colaboración.
- Vista de presentación: ha sido incorporada completamente al lenguaje base UML como aporte fundamental a la modelación de este tipo de aplicaciones, está compuesta por el diagrama de estructura de navegación y el diagrama de estructura de presentación.

1.3.2.1 Lenguaje a utilizar

Luego de realizar un análisis sobre algunos lenguajes de modelado de software se llega a la conclusión que UML, a pesar de ser el lenguaje más conocido y utilizado por los modeladores en la actualidad, no soporta, con sus vistas y diagramas, todos los aspectos de las aplicaciones multimedia de una forma adecuada, además de que no permite que se logre una representación eficiente de todos los elementos que desde el punto de vista pedagógico son abordados en el software educativo cubano; al igual que OMMMA-L, la cual, sin embargo es presentada como algo eficientemente realizable que facilita un gran rango de aspectos de aplicaciones multimedia interactivas. Se escoge para el modelado del software, el Lenguaje para la modelación de Aplicaciones Educativas Multimedia (ApEM – L), el cual surge con el propósito particular para la modelación de aplicaciones educativas, sin complicarse con elementos que lo convirtieran en un método de desarrollo de aplicaciones educativas, sino que abarcara sólo el área de la representación y la documentación de este tipo de productos.

1.3.3 Herramientas de ingeniería asistida por computadoras (Computer Aided Software Engineering (CASE))

Con el actual aumento de la complejidad en la producción de sistemas informáticos se hace necesario documentar toda la información relacionada con el proceso de desarrollo y, aparejado a esto, el conocimiento de herramientas de modelado visual que nos ayuden a comprender mejor la aplicación que estamos desarrollando; usarlas hace más portable la documentación a generar, proporciona la reutilización de componentes y permite generar automáticamente el código base. En este caso se utilizará Rational Rose debido a las necesidades y decisiones propias del proyecto, teniendo en cuenta las características de la metodología a aplicar.

Rational como soporte a la metodología

Rational es una herramienta de modelado visual de sistemas orientados a objetos, su vista de explorador está compuesta por cuatro de las vistas arquitectónicas: Vista de casos de uso, Vista lógica, Vista de componentes y Vista de despliegue, permitiendo así agrupar en paquetes los elementos de los modelos. Facilita la generación de código en lenguajes como C++, Visual Basic, Java y Ada. Soporta realizar ingeniería inversa por lo que se puede obtener un diseño a partir del código de un programa. Permite varias personas trabajando a la vez en un proceso iterativo controlado, para ello posibilita que cada desarrollador opere en un espacio de trabajo privado que contiene el modelo completo.

Rational Rose cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases y entregables. Propone la utilización de cuatro tipos de modelos para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico.” [16]

Esta herramienta enmarca los requerimientos y variables del diseño, permitiendo crear un ciclo de desarrollo más pequeño, robusto y adaptable. Evidenciando así las características:

- Reusabilidad: Mediante el modelado visual se pueden crear componentes (modelos), que al salvarlos pueden ser compartidos y reutilizados por varios proyectos, permitiendo que los cambios sean fácilmente incorporados a proyectos existentes.
- Desarrollo basado en componentes: El desarrollo basado en componentes se ha convertido en el proceso de diseño más efectivo, los usuarios de Rational pueden modelar sus componentes e interfaces sólo haciendo un “arrastrar y soltar” de los componentes del sistema hacia el diagrama de componentes.

En fin, esta herramienta permite hacer análisis y diseño para un producto de software ayudando a los equipos de desarrollo a construir, integrar, extender, modernizar e implantar.

1.4 Herramientas para la creación de software interactivo

Existen múltiples herramientas que manejan elementos de media asociados a la programación con el objetivo de lograr una funcionalidad didáctica, brindan el marco esencial para organizar y editar elementos de un producto interactivo. Estas son utilizadas para diseñar las interfaces del usuario, con el fin de presentar un proyecto en pantalla con un ambiente integrado del contenido y las funciones, según la decisión del cliente o el desarrollador en algunos casos.

Se pueden encontrar herramientas que son verdaderos editores de obras de este tipo, con un arsenal de recursos para la integración y el manejo de los diferentes tipos de información, incluyendo las animaciones. Algunas proporcionan funciones para establecer enlaces hipertextuales e hipermediales y dotan a la obra de una alta interactividad empleando múltiples y diversos modos de navegación. [21]

Algunas de las más potentes y utilizadas en el mundo para hacer software interactivos, que combinan las ventajas de la edición visual de la obra y la libertad y poder de la programación, son: Macromedia Flash, Director, Toolbook, entre otros, los cuales son utilizados según la decisión del cliente o el desarrollador en algunos casos.

Macromedia Director

Director es una poderosa herramienta, de fácil manejo y proporciona desarrollar aplicaciones sin apenas tener que programar. Permite la combinación de texto, gráfico, sonido, animación y video en un documento que se reproduce en el ordenador y que es presentado con múltiples detalles. Incorpora un rango de nuevas capacidades para satisfacer las necesidades evolutivas del desarrollador actual. Incluye nuevas y mejores eficiencias en el flujo de trabajo y la habilidad de crear contenido accesible para que sus presentaciones enriquecidas puedan ser disfrutadas por personas con discapacidades.

La filosofía seguida por este programa es la de una línea de tiempo durante la cual irán sucediendo diferentes acontecimientos según se vayan necesitando. Este proceso no tiene necesariamente por qué ser lineal ni continuo sino que permite detenerse en un punto del tiempo y saltar a otro en esa línea temporal. El lenguaje de programación orientado a objetos de Director (Lingo) agiliza los tiempos de desarrollo y ayuda a integrar a sus producciones una interactividad única y de alto nivel. [21]

Esta herramienta posee la capacidad para responder rápidamente a los cambios de los requisitos de trabajo para reducir los cambios complejos, ayudando a hacer cambios rápidos; son esenciales para

ayudar a las empresas a ser más ágiles y controlar los costes de desarrollo. Director a pesar de ser muy usada y permitir la combinación de medias requiere del uso de programas exteriores para crearlos, característica que en ocasiones impide su utilización, además de que no es multiplataforma.

Macromedia Flash

Desde hace ya algunos años la aparición de Flash, con sus diferentes versiones, ha traído hasta nuestras pantallas todo un mundo de interactividad, movimiento y color. Esta herramienta resulta fácil de aprender y sus archivos están creados para reproducirse rápidamente. Además posee grandes posibilidades para lograr presentaciones de animación y manejo del sonido, texto, video y gráficos.

Los archivos de Flash aparecen muy a menudo como animaciones en páginas Web, pues consigue dar dinamismo mediante animaciones y crear aplicaciones interactivas que permiten al usuario ver la web como algo atractivo, no estático. En la actualidad Macromedia ha ampliado Flash más allá de las animaciones simples, esto lo convierte en una herramienta de desarrollo completa, para crear principalmente elementos multimedia. Una de las versiones más utilizadas es Flash 8, que cuenta con varias prestaciones para incorporar un servicio de video mucho más integrado. [21]

Flash no es sólo un programa para crear gráficos sino que es un lenguaje de programación, pues mediante su lenguaje script orientado a objetos: ActionScript se pueden crear programas que, por ejemplo, busquen en una base de datos o interactúen con un programa en otro lenguaje. Además nos brinda la posibilidad de exportar la película para ser vista en diferentes sistemas operativos.

ToolBook

Toolbook ofrece un ambiente de programación orientado a objeto para construir proyectos, o libros, a fin de presentar gráficamente información, como dibujos, imágenes digitalizadas a color, textos, videos, sonidos, gráficos y animaciones, lo que hace de este programa una herramienta muy útil para la elaboración de software con tecnología multimedia. Su interfaz es muy fácil ya que cuenta con la apariencia de un programa de dibujo y utiliza como lenguaje de programación el Openscript, basado en el lenguaje script.

ToolBook tiene dos niveles de trabajo: el lector y el autor y se ejecutan los guiones a nivel de lector. A nivel autor se utilizan órdenes para crear nuevos libros, crear y modificar objetivo en las páginas y escribir guiones. ToolBook ofrece opciones de vinculación para botones y palabras claves, de forma que se puedan crear guiones de navegación, identificando la página a la que debe ir. [22]

Las aplicaciones creadas con Toolbook se asemejan a la estructura de un libro (book), donde cada una de sus pantallas se corresponde con las diferentes páginas del mismo. Además, Toolbook permite asignar un mismo fondo (background) en la aplicación (mediante una imagen, color o conjunto de objetos) para optimizar los recursos tanto de la aplicación como del propio ordenador. [23]

Scenetoolkit. Herramienta básica de visualización para el desarrollo de sistemas de realidad virtual o Motor de visualización 3D

Es una herramienta de gran utilidad para la construcción rápida de interfaces gráficas, permite no solamente la visualización de los entornos sintéticos sino además la aplicación de leyes físicas y matemáticas, animaciones, etc. Permite añadir nuevas librerías, por parte de los usuarios de la herramienta, en caso de que lo necesiten y se retroalimenta con las necesidades de los desarrolladores que las utilicen, así como de sus investigaciones. Esta fue creada con una arquitectura orientada a objetos, multiplataforma y configurable, empleando el Proceso Unificado de Software (RUP) e implementada en C++.

Es un importante apoyo para los programadores de juegos y simuladores, a través de la reutilización de código, pretendiendo brindarles funcionalidades de forma cómoda acorde con la tendencia mundial de desarrollo de los llamados “*engines*” o motores gráficos. Usa las librerías gráficas OpenGL y DirectX, sobre plataforma Windows y Linux. Su implementación se hizo usando como IDE el VisualStudio 2003 sobre Windows, y el Code::Blocks sobre Linux. [24]

Como integración a la herramienta básica de visualización SceneToolkit se crea el módulo de sonido Soundtoolkit, esta brinda funcionalidades para el manejo de sonido de forma cómoda, abstrayendo a los desarrolladores de los bajos niveles de programación, compatible con varios formatos y con posibilidades de adicionarle otros en futuras versiones.

Soundtoolkit fue desarrollada con OpenAL, libre y multiplataforma para sonido tridimensional. Es un sistema compatible con los sistemas de audio convencionales dado que el efecto de tridimensionalidad obtenido con este sistema alcanza un alto nivel de realismo y calidad acústica. El uso de esta lista de sonidos permite una cómoda manipulación y acceso, aprovechando también la gestión dinámica de la memoria que hace, y que optimiza el trabajo. [25]

1.4.1 Herramientas seleccionadas para el desarrollo del producto

Como resultado del estudio comparativo de algunas herramientas seleccionadas se puede resumir que Flash como herramienta para crear animaciones 2D funciona muy bien, sin embargo hay que evaluar

si realmente es necesario a la hora de usarlo para crear aplicaciones interactivas con aspectos de tridimensionalidad tales como juegos. Flash utiliza como apoyo a Director, la cual a pesar de ser muy usada y eficiente precisa de un cierto grado de programación para dar la interactividad requerida al juego que se desea desarrollar, además de que los editores que presenta son muy limitados y sólo permiten crear objetos muy sencillos, o para retocar los creados externamente. ToolBook por su parte requiere de muchas líneas de código para llevar a cabo el proceso de implementación y no cuenta prácticamente con editores de recursos, sólo incluye uno de mapas de bits que es muy básico.

Se escoge finalmente Scenetoolkit con su módulo integrado para el manejo de sonidos, pues cumple con todas las características antes expuestas. Esta herramienta, que a diferencia de las anteriores, tiene como fin la construcción específica de juegos, fue creada por algunos de los propios desarrolladores, por lo que es totalmente gratis y no necesita permiso de autor para su uso. Además de lo planteado, se selecciona pues los programadores no poseen conocimientos previos abarcadores de otras tecnologías, lo que sería imprescindible por la complejidad de las mismas, especialmente en el caso de Director y Toolbook, evitando así precisar de tiempo para su estudio, así como incumplir con la entrega en la fecha establecida del producto informático.

Para la programación se usará C++, el lenguaje de Scenetoolkit, que permitirá estructurar las medias y la información para mostrarla al usuario de manera adecuada.

1.5 Conclusiones del capítulo

En este capítulo se realizó un estudio, exponiendo elementos teóricos del software educativo e implicaciones de la informática en la educación actual, así como de la tecnología multimedia y su uso para la creación de juegos interactivos. Para el desarrollo de la propuesta se seleccionó RUP como la metodología a utilizar, entre otras analizadas a profundidad, fundamentando las particularidades necesarias para realizar el análisis y diseño del software que la distinguen como propuesta ideal en este proceso; conjuntamente con ApEM – L, pues como extensión de UML ofrece sus mismas comodidades, añadiéndole o modificándole diagramas que permiten un mejor modelado para este tipo de aplicaciones. Además, para el modelado fue escogida: Racional Rose, como soporte a la metodología y para la elaboración de la multimedia la herramienta para el manejo de sonidos, escenas y gráficos en Sistemas de Realidad Virtual: Scenetollkit.

Capítulo 2

Descripción de la solución propuesta

2.1 Introducción del capítulo

En este capítulo se comienza a realizar el análisis del producto, con el apoyo del Proceso Unificado de Desarrollo de Software. Se modela el entorno donde se usará el software y se identifican y describen sus principales conceptos asociados, reflejados mediante la realización de un modelo de dominio. Se definen los requerimientos funcionales y no funcionales por los que se regirá el sistema propuesto, que generan especificaciones correctas que describan con claridad, sin ambigüedades y en forma consistente las necesidades del cliente. Se incluye además una estructuración de las secciones y contenidos que deberá poseer la aplicación y se especifican los elementos de navegación del mismo.

2.2 Identificación de la audiencia

Cuando se decide realizar un producto software de alta calidad que cumpla con los objetivos para los que se concibió, una de las consideraciones más importantes es identificar correctamente y tener en cuenta el público al que va dirigido.

Es de suma importancia determinar las características del cliente a quien va dirigida la aplicación y encaminarse a satisfacer sus necesidades. Es solamente mediante una detallada comprensión de la audiencia que es posible concebir y construir mensajes y acciones que lleguen a esas personas y que generen en su pensamiento y en sus acciones la respuesta deseada. [26]

El juego interactivo “Energía para Aprender” está concebido para niños y niñas venezolanos, cuyas edades oscilan entre los 9 y 12 años de edad, que pertenecen a diferentes estados o zonas del país y pretende que ellos puedan contar con un producto, que de una forma amena e instructiva muestre las transformaciones que se están realizando en el país en el marco de la Misión Revolución Energética, a través de tareas o situaciones que requieran de astucia y rapidez para resolverse.

2.3 Aspectos generales

Antes de implementar este software educativo presentado en forma de juego interactivo es importante destacar por qué se decide hacer o que objetivos se persiguen con el mismo:

- Desarrollar conocimientos relacionados con la Misión Revolución Energética, el uso eficiente y seguro de la corriente y el mejoramiento de las condiciones medioambientales en Venezuela.
- Contribuir a la formación de habilidades para acometer acciones que propicien el uso eficiente de fuentes alternativas de energía y la toma de decisiones para la aplicación de medidas como el ahorro de energía eléctrica, y otros recursos, desde su radio de acción.
- Resolver situaciones de aprendizaje a partir de la propuesta de medidas y acciones de la Misión Revolución Energética que pueden aplicarse para evitar el uso no eficiente de la energía y la contaminación.
- Desarrollar habilidades relacionadas con la toma de decisiones en situaciones con diferentes alternativas de solución, retentiva y agilidad mental.

A sugerencia y petición del cliente se tratan los aspectos relacionados con la Misión en Venezuela de forma combinada. El juego se estructura según algunas de las direcciones de trabajo de la Misión Revolución Energética en Venezuela, las cuales son:

- Cambio de bombillos incandescentes por ahorradores.
- Utilización racional de la energía eléctrica y vías para la disminución de los niveles de contaminación.
- Utilización de la energía solar e hidráulica y el montaje de mini hidroeléctricas y de sistemas fotovoltaicos.
- Utilización de la energía eólica y el recorrido por el parque eólico.
- Uso eficiente y seguro de la corriente eléctrica.

2.4 Solución propuesta

Como solución al problema se realiza el modelado del software educativo con tecnología multimedia, presentado como un juego interactivo titulado: “Energía para Aprender”, creando un entorno educativo e interactivo, mediante la adecuación de los contenidos y el lenguaje utilizados en esta a cada entorno propuesto. Al acceder al juego el usuario hará un recorrido por diferentes zonas del país en las que se han desarrollado acciones en beneficio del pueblo en el marco de la Revolución Energética en Venezuela.

Durante la trayectoria han de salvarse algunas misiones cuyo cumplimiento contribuye al desarrollo de valores relacionados con la identidad y el amor a la patria a partir del conocimiento del modo de vida,

costumbres e historia de distintas regiones de la nación. Los contenidos incluidos en el juego, así como las actividades que debe resolver el jugador, estarán acorde con los objetivos, con el desarrollo psicológico del niño de esas edades y en correspondencia con los conocimientos que se imparten en la enseñanza primaria.

El juego estará estructurado en tres niveles, en los que intervendrán varios personajes, algunos identificativos del jugador, uno positivo, simbolizando el ahorro y otro negativo, representando la contaminación y el derroche energético. Dichos niveles familiarizarán al niño con las principales obras sociales llevadas a cabo en el marco de la Misión Revolución Energética en Venezuela y cada uno tratará contenidos específicos:

- Nivel 1: Abordará el tema de la importancia económica, social y ambiental del cambio de bombillos en el país. En él el personaje principal debe renovar la iluminación de la ciudad, cambiando los bombillos incandescentes por ahorradores.
- Nivel 2: Abordará el tema de la importancia del uso eficiente y seguro de la corriente eléctrica. En él el personaje debe identificar componentes industriales involucrados en la generación de energía, además de colocar filtros en una industria con el fin de disminuir los niveles de contaminación
- Nivel 3: Abordará el tema de la utilización combinada de las energías hidráulica y solar para como fuentes alternativas para la generación de energía eléctrica. En él el personaje debe instalar un sistema fotovoltaico y una mini hidroeléctrica, los cuales una vez montados abastecerá de corriente eléctrica a la comunidad.

Se tiene pensado que el juego cuente con dos niveles más, para completar así la relación de su estructura con las direcciones de trabajo de la Misión, esta idea será la base para enriquecer la aplicación, dichos niveles se pretende que aborden los siguientes contenidos:

- Nivel 4: Utilización de la energía eólica y el recorrido por el parque eólico.
- Nivel 5: Uso eficiente y seguro de la corriente eléctrica.

2.5 Orientaciones Metodológicas

Cuando se pretende lograr una formación integral en niños cuyas edades oscilan entre los 9 y 12 años, uno de los campos de acción y desempeño mencionados lo constituyen los juegos electrónicos, actividad a la que por demás dedican una parte considerable de su tiempo libre. “Energía para aprender” pretende que el niño pueda contar con un producto con el cual a la vez que juega aprende

sobre temas relacionados con la historia de su país y algunas de las transformaciones sociales que en él operan asociadas a la Misión Revolución Energética, contribuye además al desarrollo de valores asociados a la solidaridad, la responsabilidad, la laboriosidad y el patriotismo.

Desde cada nivel del juego se refuerzan elementos de la personalidad del niño que responden a los requerimientos básicos planteados en el perfil psicológico para estas edades por el Ministerio del Poder Popular para la Educación, los cuales se exponen desde la óptica y el contenido del juego y que contribuyen al desarrollo de:

- Su identidad venezolana sobre la base de una conciencia ambientalista y de respeto al acervo histórico cultural de la nación.
- Conocimientos sobre el quehacer científico y tecnológico del país en materia de energía y ambiente.
- Actitudes valorativas relacionadas la cultura del ahorro, el mantenimiento de la salud y en la detección de factores de riesgo y peligros de accidente relacionados con el consumo energético en las comunidades.
- La observación como método para establecer relaciones espaciales y temporales que le ayuden a ubicarse e identificarse dentro de un contexto y relacionarse con los distintos elementos que en él interactúan.
- El pensamiento crítico, la creatividad y la autonomía como vías para la solución de las tareas, la transformación de la información recibida y la elaboración de nuevos conocimientos.
- Habilidades de lectura, escritura, cálculo e interpretación en la resolución de problemas de la vida cotidiana relacionados con el consumo energético.
- Habilidades en el uso de las tecnologías de la información y la comunicación.

En correspondencia con lo anterior, el trabajo con el juego incluye el análisis de contenidos de enseñanza de las diferentes áreas y asignaturas de los grados de cuarto a sexto en la educación primaria, así como su contribución a la formación de los valores referidos, y que además potencien la cultura del ahorro de energético y la protección del medio ambiente.

Las tareas a desarrollar durante el juego contribuyen a la formación de la conciencia energética y ambientalista que se aspira, pues se podrá lograr la interiorización de valores tales como: la responsabilidad, la solidaridad, el patriotismo, la honestidad y la disciplina social, los que, entre otros

componentes, forman la educación político-ideológica y ciudadana que se debe lograr en las nuevas generaciones de venezolanos.

De este modo, el aprendizaje se hace más efectivo y duradero, la personalidad de los niños se va desarrollando como un todo íntegro, donde lo cognitivo y lo afectivo-volitivo, lo interno y lo externo, se insertan, formando una unidad dialéctica. Sobre esa base se orientan su conducta, sus actitudes, la expresión de sus criterios y opiniones, sus juicios y sus valoraciones, su toma de posiciones encaminada a contribuir a resolver los problemas que afronta el país relacionados con el uso eficiente de la energía y las transformaciones sociales que con este fin se llevan a cabo en Venezuela.

2.6 Modelo de Dominio

El análisis de sistemas implica determinar las necesidades del cliente para poder especificar los aspectos que sirvan como base para el desarrollo de un software siendo el “que” de un producto informático. Para capturar correctamente los requisitos se necesita tener un firme conocimiento del funcionamiento del objeto de estudio y lograr de esta forma construir una aplicación adecuada.

En este caso, existe poca estructuración de lo que se desea hacer, por lo que no se puede considerar que exista un negocio bien definido y consistente que permita determinar las responsabilidades, entidades y funcionamiento de la situación a modelar. Se decide representar los conceptos que se manejan en la situación existente a través de un Modelo de dominio, el cual es una de las alternativas que brinda RUP para la identificación de requisitos y la comprensión del contexto cuando existe poca estructuración en los procesos de negocio. Así los involucrados en el desarrollo tendrán a su alcance un vocabulario común que permita el entendimiento del contexto del sistema.

En este modelo se representan las clases conceptuales que pueden intervenir en el sistema y sus asociaciones, así como los objetos más importantes en el mismo. Estos objetos del dominio representan “cosas” que existen o los eventos que acontecen en el medio en el que se desenvuelve la aplicación. [27]

2.61 Identificación de los conceptos del dominio

- Entorno: Contenido general del juego interactivo relacionado con la Misión Revolución Energética de Venezuela.
- Usuarios: Niños de entre 9 y 12 años de edad que van a jugar y resolver los diferentes niveles para entretenerse y aprender.

- Niveles: Etapas o misiones en las que estará dividido el juego y que deben vencerse, su cumplimiento contribuirá al desarrollo de valores a partir del conocimiento del modo de vida, costumbres e historia de distintas regiones de la nación.
- El brillo de mi ciudad: Primer nivel del juego.
- Las redes de mi país: Segundo nivel del juego.
- Agua saludable: Tercer nivel del juego.
- Personajes: Actores que intervendrán en cada uno de los niveles que conforman el juego.
- Mascota: Personaje Luminito, servirá como guía en la realización de las tareas en cada uno de los niveles. Representa el ahorro y las transformaciones de la Misión Revolución Energética.
- Principal: Jugador que será protagonista en la ejecución de cada tarea y/o misión.
- Negativo: Personaje que participará como antagonista del jugador durante las tareas que se realizan. Es una identificación personalizada y representativa del derroche energético y la contaminación.
- Objetivos: Metas de cada nivel a alcanzar cuando se resuelvan las tareas y preguntas.
- Tareas: Metas a resolver en cada nivel según el propósito, a partir de la propuesta de medidas y acciones de la Misión, que pueden aplicarse para evitar el uso no eficiente de la energía y la contaminación.
- Puntaje: Puntos a acumular según se vaya resolviendo la tarea correspondiente a cada nivel.
- Preguntas: Cuestionarios de control relacionados con los temas que se tratan en cada nivel.

2.6.2 Diagrama de clases del modelo del dominio

En el modelo de dominio de la aplicación a crear, mostrado en la figura 2, se describe con un diagrama de clases conceptuales significativas, brindando una mayor comprensión de los eventos que suceden en el entorno que se desea modelar, en él se representan los conceptos descritos anteriormente.

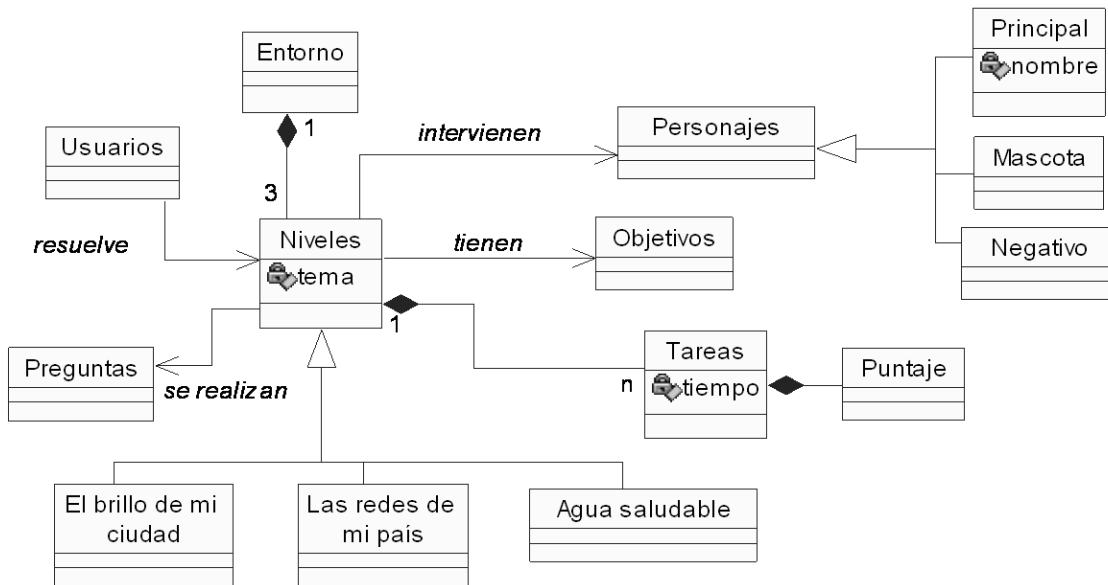


Figura 1. Diagrama de clases del Modelo de Dominio

2.7 Descripción de la funcionalidad

A través de los años se ha podido constatar que los requerimientos son la pieza fundamental en un proyecto de desarrollo de software, ya que marcan el punto de partida para actividades como la planeación, contribuyen a la identificación de las funcionalidades, el comportamiento de entrada y salida del sistema además generan especificaciones correctas que describen con claridad, sin ambigüedades y en forma consistente las características o cualidades que debe poseer la futura aplicación.

La especificación de requerimientos es la base que permite verificar si se alcanzaron o no los objetivos establecidos en el proyecto ya que estos son un reflejo detallado de las necesidades de los clientes o usuarios del sistema y es contra lo que se va a estar, verificando si se están cumpliendo las metas trazadas. [27]

Los requisitos de software pueden dividirse en dos categorías: como resultado de las peticiones del cliente se definen 13 funcionales y 7 no funcionales.

2.7.1 Requerimientos funcionales

A continuación se muestran los requerimientos funcionales del sistema, los cuales especifican acciones que el sistema debe ser capaz de realizar, sin considerar restricciones físicas.

Estos describen las transformaciones a realizar sobre las entradas para producir salidas. [24] Al tiempo que avanza el proyecto estos se convierten en los algoritmos, la lógica y gran parte del código del sistema.

1. Mostrar animación que da presentación al juego.
2. Mostrar la pantalla principal del juego.
3. Permitir acceder desde cualquier parte del juego a la pantalla principal.
4. Permitir seleccionar el jugador con el que el usuario desea jugar.
 - 4.1. Mostrar para la selección por el usuario ocho (8) prototipos de personajes, de ambos sexos, que respondan a las diferentes etnias del país.
 - 4.2. Permitir nombrar el personaje seleccionado por el usuario.
 - 4.3. Permitir guardar, la selección realizada por el usuario, del jugador con su nombre.
 - 4.4. Permitir comenzar el juego en el primer nivel después de realizada la selección del jugador.
5. Permitir configurar el juego.
 - 5.1. Permitir al usuario configurar el uso del teclado (usar las teclas direccionales o las teclas del sistema WASD, para realizar los movimientos requeridos en el juego) de acuerdo a su preferencia.
 - 5.2. Permitir al usuario aumentar o disminuir el volumen de acuerdo a su preferencia.
 - 5.3. Permitir al usuario configurar la dimensión de la pantalla (media pantalla o pantalla completa) de acuerdo a su preferencia.
 - 5.4. Permitir guardar los cambios de configuración realizados.
6. Comenzar el juego.
 - 6.1. Acceder al primer nivel del juego para comenzar a jugar.
 - 6.2. Permitir el acceso a cada uno de los niveles, en forma ascendente, en la medida que el usuario venza las tareas previstas a realizar en ellos.
 - 6.3. Mostrar oportunidades para vencer el juego.
 - 6.3.1. Visualizar en la pantalla el elemento de carácter informativo que indicará las cinco (3) posibilidades con las que cuenta el jugador para vencer el juego.
 - 6.3.2. Permitir eliminar una vida cada vez que el usuario pierda.
 - 6.3.3. Permitir terminar el juego cuando el usuario agote todas las vidas.
 - 6.3.4. Permitir aumentar una vida cuando el usuario alcance determinada puntuación.

Para cada uno de los niveles

- 6.4. Mostrar presentación de inicio del nivel.
- 6.5. Mostrar pantalla de inicio del nivel.
- 6.6. Permitir el acceso, en forma ascendente, a cada una de las pantallas que conformen el nivel en la medida que se venzan las tareas previstas en cada una de ellas.
- 6.7. Mostrar puntuación.
 - 6.7.1. Mostrar la puntuación obtenida en las pantallas pertenecientes a cada uno de los niveles.
 - 6.7.2. Actualizar la puntuación obtenida por el jugador cada vez que realice una acción que le otorgue puntos.
- 6.8. Permitir controlar diferentes parámetros ambientales relacionados con el consumo energético, de acuerdo al nivel en que se esté jugando.
 - 6.8.1. Visualizar haciendo uso de algún elemento medidor:
 - La temperatura planetaria, para el primer nivel.
 - El nivel de CO₂ atmosférico, para el segundo nivel.
 - 6.8.2. Permitir visualizar los cambios (aumento o disminución) del elemento medidor, según el nivel.
- 6.9. Permitir realizar las tareas de cada nivel contra reloj.
 - 6.9.1. Visualizar y controlar, haciendo uso de un cronómetro, el tiempo requerido para finalizar las acciones propuestas en cada nivel.
 - 6.9.2. Informar al usuario, a través de una locución, cuando le resta poco tiempo de juego y no ha culminado las tareas previstas en el nivel.
 - 6.9.3. Permitir finalizar el juego cuando el tiempo destinado para las tareas del nivel haya concluido y el usuario no haya podido terminar.
 - 6.9.3.1. Permitir que el usuario pueda reanudar el juego en el nivel que perdió.
 - 6.9.4. Permitir realizar determinada tarea, dentro del nivel, contra reloj.
 - 6.9.4.1. Visualizar y controlar, haciendo uso de un cronómetro (diferente al que registra la duración del nivel), el tiempo requerido para finalizar determinada tarea dentro del nivel.

6.9.4.2. Informar al usuario, a través de una locución, cuando le resta poco tiempo y no ha culminado la tarea prevista.

6.9.4.3. Permitir finalizar el juego cuando el tiempo destinado para la tarea haya concluido y el usuario no haya podido terminarla.

6.9.4.3.1. Permitir que el usuario pueda reanudar el juego en el nivel que perdió.

6.10. Permitir finalizar el juego cuando el usuario haya realizado erróneamente, por segunda vez, cualquier tarea prevista en el nivel.

6.10.1. Permitir que el usuario pueda reanudar el juego en el nivel que perdió.

6.11. Permitir realizar preguntas interactivas referentes al contenido del nivel que se está jugando.

6.11.1. Mostrar la pregunta interactiva.

6.11.1.1. Seleccionar de forma aleatoria una pregunta, de acuerdo al nivel en que se esté jugando.

6.11.1.2. Visualizar el enunciado de la pregunta y tres posibles respuestas.

6.11.2. Permitir que el usuario pueda responder la pregunta.

6.11.2.1. Permitir que las respuestas sean excluyentes, es decir, que se pueda seleccionar una sola respuesta.

6.11.2.2. Permitir continuar el juego si se responde correctamente la pregunta.

6.11.2.3. Mostrar otra pregunta si se responde de forma errónea.

6.11.2.4. Permitir finalizar el juego si se responde erróneamente la pregunta por segunda vez.

6.11.2.4.1. Permitir que el usuario pueda reanudar el juego en el nivel que perdió.

6.12. Permitir al final de cada nivel, el acceso a una pregunta opcional, según la decisión del usuario.

6.12.1. Mostrar la pregunta interactiva. (Tendrá el mismo comportamiento que las acciones descritas en los puntos 6.7.1.1 y 6.7.1.2)

6.12.2. Permitir que el usuario pueda responder la pregunta. (Tendrá el mismo comportamiento que la acción descrita en el punto 6.7.2.1)

6.12.2.1. Permitir aumentar el puntaje obtenido hasta el momento si se responde correctamente la pregunta.

6.13. Permitir poner el juego en pausa.

6.13.1. Permitir detener el juego justo en la acción que se esté realizando.

6.13.2. Permitir reanudar el juego, para quitar la pausa.

6.14. Permitir acceder a la pantalla principal del juego.

6.15. Permitir abandonar el juego.

7. Mostrar en diferentes momentos según corresponda los tres (3) personajes esenciales que intervienen en el juego.

Personaje principal (Jugador)

7.1. Mostrar el personaje principal durante todo el desarrollo de las acciones de cada uno de los niveles que conforman el juego.

7.2. Mostrar cada uno de los movimientos que el personaje principal realiza en cada nivel.

7.2.1. Permitir que el usuario pueda dirigir los movimientos y acciones que el personaje principal debe realizar.

7.2.2. Permitir los movimientos del personaje principal para realizar las tareas previstas en cada nivel, a partir del uso de las teclas (Sistema WASD y teclas direccionales) y el mouse según sea el caso.

Mascota del juego

7.3. Mostrar la mascota al inicio de cada nivel (después de la presentación del nivel) emitiendo una locución que exponga las características del nivel y explique las tareas que debe realizar en el mismo.

7.3.1. Permitir interrumpir la explicación de la mascota y que esta desaparezca de la pantalla al hacer clic sobre ella.

7.4. Mostrar la mascota emitiendo una locución en los momentos en que durante la ejecución de un nivel se requiera informar al jugador determinadas especificidades de la tarea que está desarrollando.

7.5. Permitir escuchar una locución emitida por la mascota al haberse consumido dos tercios del tiempo destinado para el nivel, anunciando que le resta poco tiempo al jugador para terminar las tareas previstas.

7.6. Permitir escuchar una locución emitida por la mascota, anunciando que el jugador ha perdido al haberse consumido el tiempo destinado para el nivel.

7.7. Mostrar la mascota al final de cada nivel, indicando al jugador que tiene la opción de responder, si lo desea, una pregunta opcional.

7.7.1. Permitir escuchar una locución emitida por la mascota anunciándole al jugador si respondió o no correctamente la pregunta.

Personaje negativo

7.8. Mostrar el personaje negativo, orientando la realización de una actividad de conocimiento (pregunta de control) en los momentos en que el jugador haya realizado erróneamente la tarea que debe desarrollar en el nivel.

7.8.1. Permitir escuchar una locución emitida por el personaje negativo anunciándole al jugador si respondió o no correctamente la pregunta de control.

7.9. Mostrar el personaje negativo interfiriendo en las tareas que debe realizar el jugador en el nivel, intentando en todo momento que este pierda el juego.

8. Emitir sonidos que simulen las acciones que se están visualizando en la pantalla, así como, para representar las tareas que el usuario esté realizando para vencer el nivel.

9. Permitir guardar en la computadora el juego.

9.1. Guardar automáticamente, sólo hasta el último nivel vencido del juego.

9.2. Guardar el juego con todas las particularidades de los niveles vencidos, el personaje seleccionado por el usuario y el puntaje obtenido hasta el momento.

10. Permitir abrir un juego guardado previamente en la computadora.

10.1. Abrir el juego al inicio del segundo, tercero, cuarto o quinto nivel en dependencia de cual haya sido el nivel en el cual se haya quedado el usuario cuando lo guardó.

10.2. Abrir el juego con todas las particularidades e información con las que el usuario lo había guardado.

11. Permitir guardar el nombre, el nivel y el puntaje del jugador al finalizar el juego (cuando haya terminado el último nivel), sólo si este está entre las diez (10) mejores puntuaciones guardadas hasta el momento.

12. Permitir mostrar la lista ordenada descendientemente de los usuarios con mejor puntaje obtenido.

12.1. Mostrar el nombre, el nivel y el puntaje de los diez (10) usuarios que hayan obtenido las mejores puntuaciones registradas en la computadora.

13. Permitir cerrar el juego desde cualquier pantalla en que se encuentre el usuario.

13.1. Mostrar una ventana de confirmación de salida.

13.2. Mostrar créditos una vez que el usuario confirme que desea salir.

2.7.2 Requerimientos no funcionales

A continuación se muestran los requerimientos no funcionales del sistema, estos expresan las características o propiedades que de una u otra forma puedan limitar el sistema, que hacen al producto atractivo, usable, rápido o confiable. En muchos casos estos requisitos son fundamentales en el éxito del producto.

Los Requisitos no funcionales no alteran la funcionalidad del producto, esto quiere decir que los funcionales se mantienen invariables sin importarle con que cualidades se relacionen. Sin embargo la razón fundamental de que esta funcionalidad sea parte del producto es brindarle a este las características deseadas. [19]

1. Requerimientos de apariencia o interfaz externa

- El producto tendrá una interfaz sencilla, intuitiva y amigable para sus usuarios.
- El producto deberá poseer en los entornos que expone, colores brillantes, ordenados en una gama coherente y adecuada para el rango de edad al que está destinado el juego.
- Los entornos del juego se diseñarán en 2D (dos dimensiones), utilizando en todo momento una armonía cromática adecuada a cada contenido.
- El recorrido por cada uno de los entornos que conforman el juego serán lineal en las cuatro direcciones.
- El producto deberá cumplir con las pautas de diseño definidas por parte del diseñador informacional.
- El producto proporcionará claridad y correcta estructura de las opciones que brinda, permitiendo la interpretación adecuada e inequívoca de estas, facilitando el uso de la aplicación.
- El sistema implementará la ejecución de acciones de una manera rápida, minimizando los pasos a ejecutar en cada proceso.

2. Usabilidad

- Los usuarios que utilizarán el sistema deberán tener conocimiento básico del manejo de la computadora, así como del trabajo con sistemas operativos visuales.

- La aplicación deberá poseer una interfaz asequible y funcional, tanto para usuarios expertos como para los que no tienen conocimientos profundos de informática.
3. Requisitos para la ayuda del sistema
 - El producto contará con una ayuda que contendrá las características de las misiones que deberá vencer en cada uno de los distintos niveles, las indicaciones de cómo jugar en los mismos, así como las opciones generales de las que podrá disponer en el juego.
 4. Requerimientos de Portabilidad
 - El producto podrá ser usado bajo el sistema operativo Windows.
 5. Requerimientos de Software
 - Computadora personal con sistema operativo Microsoft Windows 98 o superior.
 6. Requerimientos de Hardware
 - Procesador Pentium II o superior.
 - 64 MB de RAM (Mínimo).
 - Kit de Multimedia y Mouse.
 - Lector de CD-ROM.
 7. Restricciones en el diseño y la implementación
 - La herramienta de desarrollo de la aplicación será SceneToolKit (STK).

2.8 Vista de gestión del modelo y sus subsistemas

Cualquier sistema grande se debe dividir en unidades más pequeñas, de modo que las personas puedan trabajar con una cantidad de información limitada...La gestión del modelo consiste en paquetes y relaciones de dependencia entre ellos...La vista de gestión del modelo modela la organización del modelo en sí mismo. [17]

En el juego educativo Revolución Energética a construir se define la Vista de gestión del modelo, representada por 7 subsistemas y sus relaciones. (Ver figura 3)

Los subsistemas definidos agrupan a las vistas de presentación según las funcionalidades que realicen o los elementos en común que representen, mostrando a clientes, usuarios y desarrolladores los elementos en la interfaz de la aplicación.

- Subsistema General: Está compuesto por la vista de animación que da presentación al producto y la principal, a través de esta última se puede acceder a todas las vistas del subsistema funcional y a la vista de presentación del subsistema Niveles para comenzar a jugar)
 - ✓ Vista Presentación
 - ✓ Vista Principal
- Subsistema Funcional: Está compuesto por las vistas que permiten al usuario realizar diferentes funciones antes o durante el juego. Desde cualquiera de estas vistas se podrá acceder a la vista Principal, usando el subsistema Menú y a través de la vista SeleccJuegoGuardado se llega al subsistema Niveles.
 - ✓ Vista SeleccJuegoGuardado
 - ✓ Vista SeleccJugador
 - ✓ Vista Configuración
 - ✓ Vista Puntaje
- Subsistema Niveles: Está compuesto por las vistas correspondientes a los niveles a modelar, incluyendo las presentaciones. A través de cualquiera de sus vistas se puede llegar a la vista Principal, usando el subsistema Menú y durante el desarrollo de los niveles muestra las vistas del subsistema Opcional y usa el subsistema Pausa.
 - ✓ Vista PresentPrimerNivel
 - ✓ Vista PrimerNivel
 - ✓ Vista CierrePrimerNivel
 - ✓ Vista PresentSegNivel
 - ✓ Vista HaciaSegNivel
 - ✓ Vista DentroSegNivel
 - ✓ Vista FueraSegNivel
 - ✓ Vista CierreSegNivel
 - ✓ Vista PresentTercerNivel
 - ✓ Vista IraTercerNivel
 - ✓ Vista SaliendoTercerNivel
 - ✓ Vista CierreTercerNivel
- Subsistema Opcional: Está compuesto por las vistas opcionales de cada nivel.
 - ✓ Vista Preguntas
 - ✓ Vista HaPerdido
 - ✓ Vista PreguntaExtra

- Subsistema Salir: Esta compuesto por la vista que brinda la opción de salir.
 - ✓ Vista Salir
- Subsistema Menú: Esta compuesto por la vista para acceder a la vista Principal.
 - ✓ Vista Menú
- Subsistema Pausa: Esta compuesto por la vista que brinda la opción de poner el juego en pausa.
 - ✓ Vista Pausa

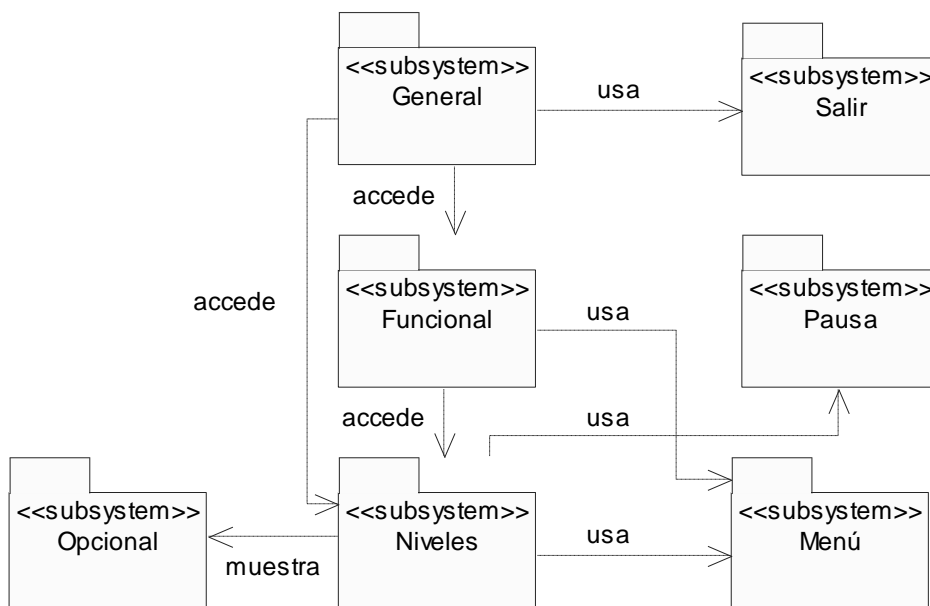


Figura 2. Representación de la Vista de Gestión del Modelo

2.8.1 Vistas de presentación

A menudo, un modelo es dividido en un número de vistas, cada una de las cuales describe un aspecto específico del producto o sistema en construcción. [20] ApEM – L tiene la vista de presentación como aporte fundamental a la modelación de aplicaciones de este tipo.

La vista de presentación es el artefacto que recoge la descripción de los elementos y su relación para el cumplimiento de una determinada función del sistema.

Es una descripción lógica de una parte funcional del sistema, expresando una secuencia de mensajes intercambiados entre dicha parte y los actores, y representa un área de la interfaz gráfica con

elementos gráficos contenidos que responden a dicha porción de funcionalidad, tales como: imágenes, elementos interactivos, animaciones y textos.

La vista que se describe en este epígrafe, que ha sido incorporada completamente al lenguaje base UML, permite utilizar su semántica original en la construcción de estructuras lógicas de presentación y navegación, incorporando un conjunto de estereotipos restrictivos y descriptivos para una mejor modelación, construyendo el diagrama de estructura de navegación y el de presentación. [20]

2.8.1.1 Actores

Los actores son el entorno del sistema, constituyen entidades que representan roles que son desempeñados por personas, otros sistemas o hardware externo que de alguna manera participan en la historia de un caso de uso o que interactúan con el sistema por decirlo de otra manera. En este caso el actor es un usuario que representa a un niño entre 9 y 12 años, que va a utilizar el sistema en busca de entretenimiento e información.

2.8.1.2 Diagramas de estructura de presentación

En este caso donde se modela una aplicación educativa, es de sumo interés para los desarrolladores todo lo relacionado con los elementos de presentación, por ello se modelan los diagramas de estructura de presentación, (ver figuras 3 - 15), con la guía del Lenguaje para la Modelación de Aplicaciones Educativas Multimedia (ApEM – L).

Para la mejor estructuración del modelo, se realiza una segunda clasificación sobre el concepto original de clase, definiendo dos nuevos tipos de estas: la *clase Estáticos* y la *clase Interacción*, las cuales seccionarán los elementos que cumplan con cada una de las características que denotan los propios nombres. La clase *estáticos* agrupará los componentes que solo tiene como función visualizar información, pero que no permiten interacción con el usuario. Lo contrario con los agrupados bajo la clase *interacción*, los cuales serán los elementos de la vista que permiten interacción del usuario con el sistema informático modelado. [20]

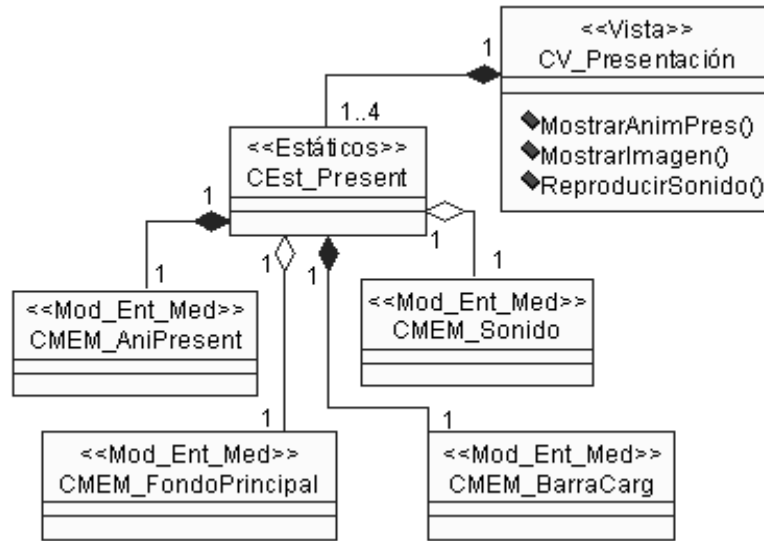


Figura 3. Diagrama de estructura de presentación Vista Presentación

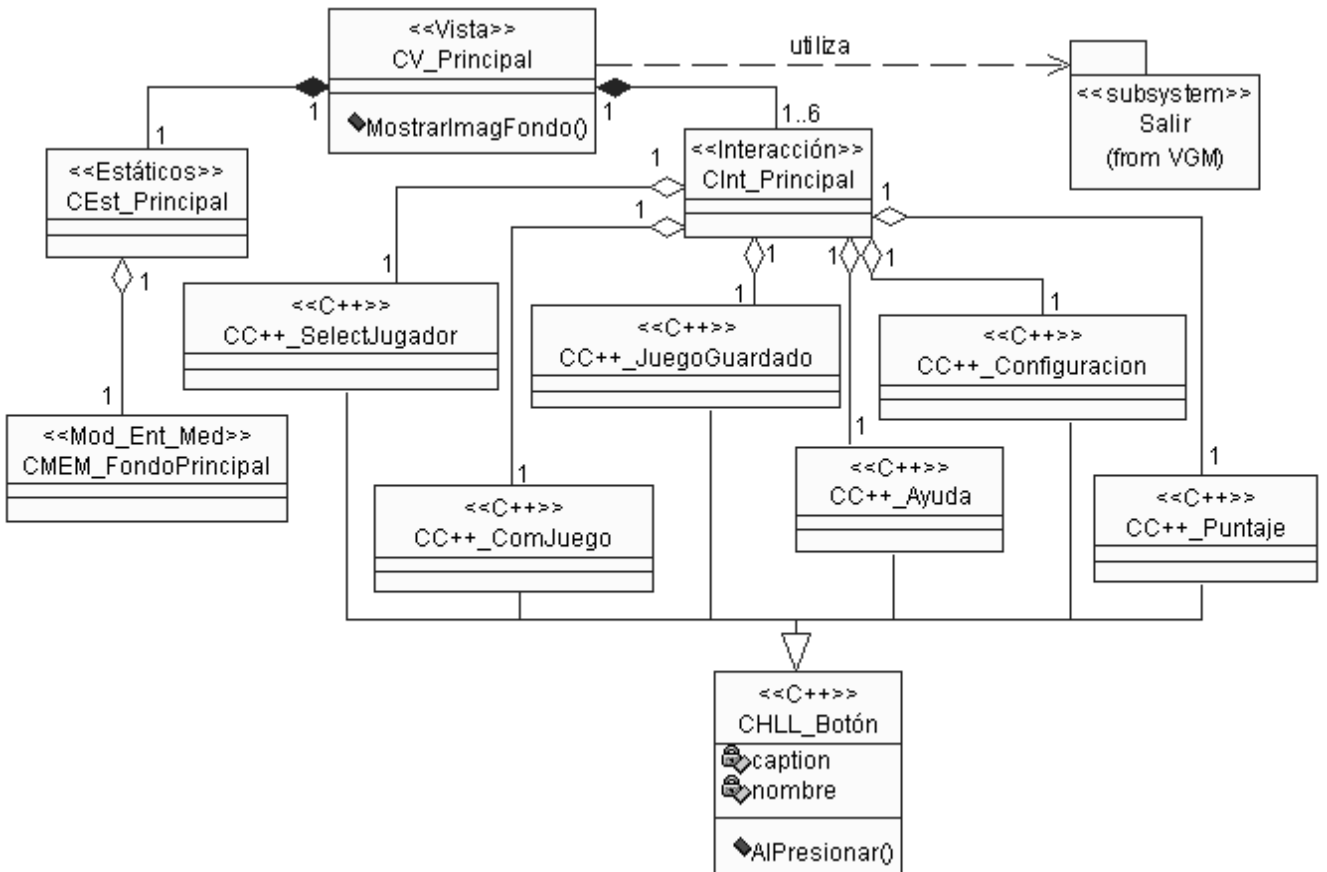


Figura 4. Diagrama de estructura de presentación Vista Principal

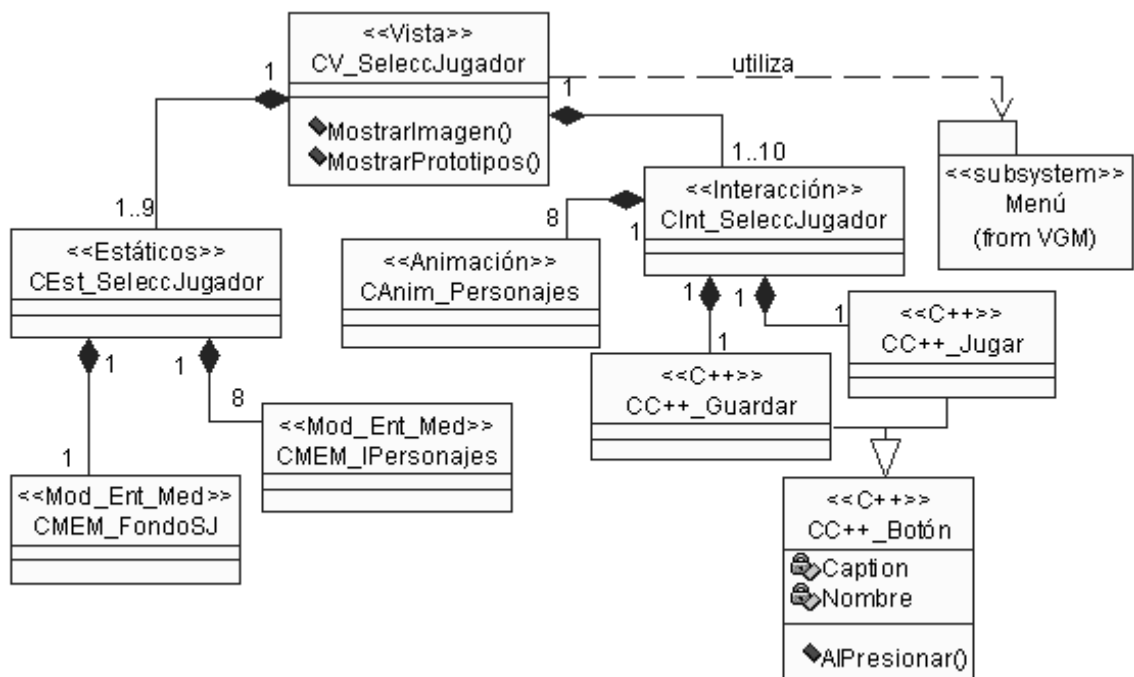


Figura 5. Diagrama de estructura de presentación Vista SeleccJugador

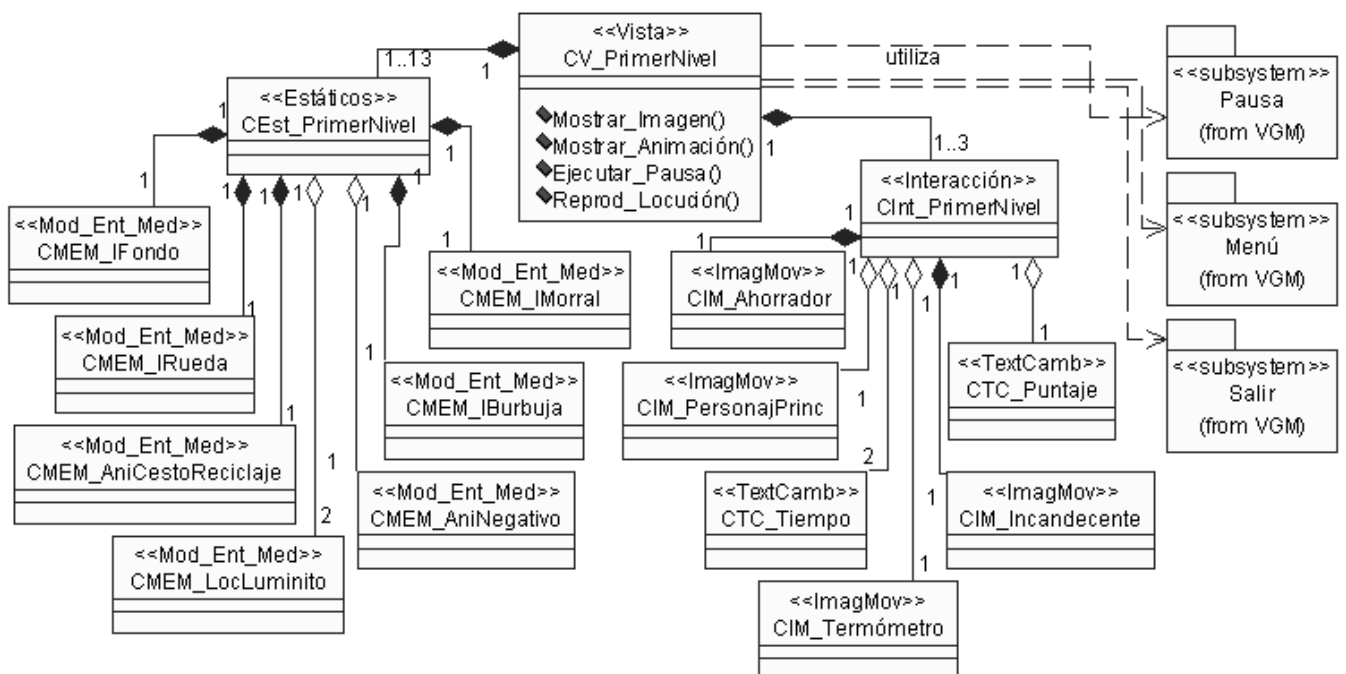


Figura 6. Diagrama de estructura de presentación Vista PrimerNivel

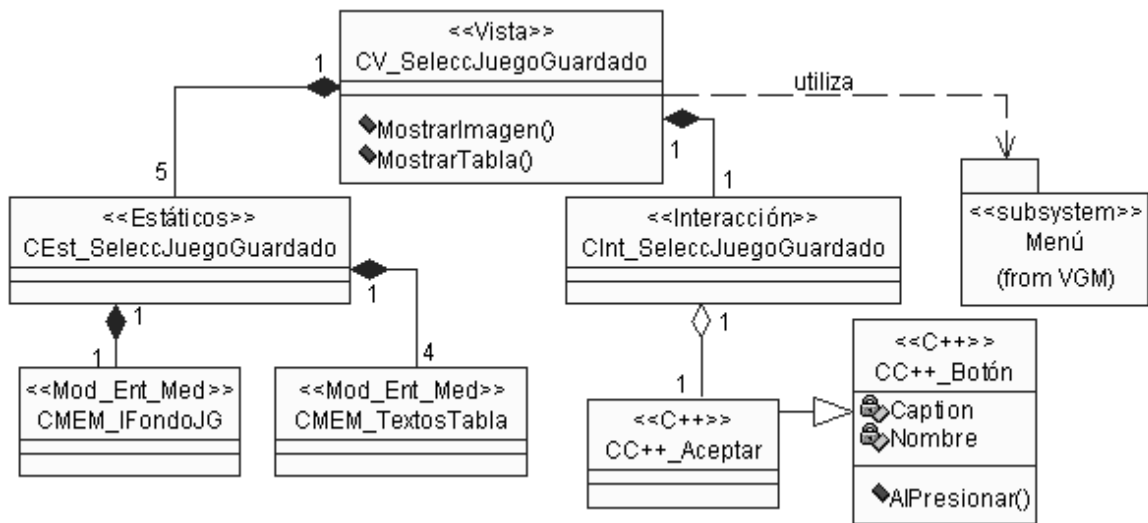


Figura 7. Diagrama de estructura de presentación Vista SeleccJuegoGuardado

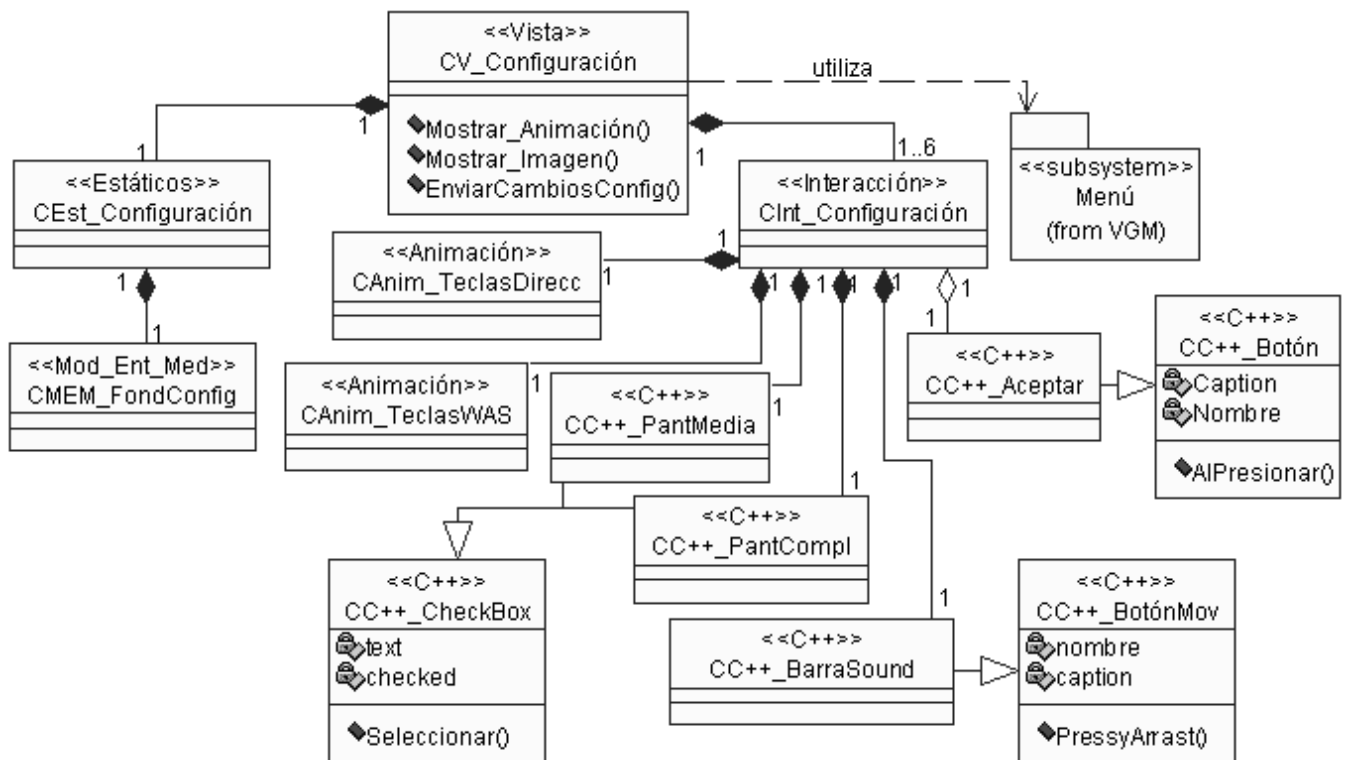


Figura 8. Diagrama de estructura de presentación Vista Configuración

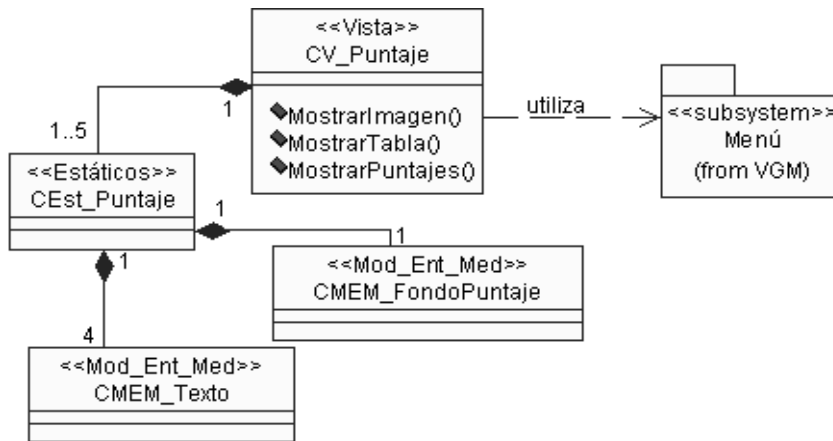


Figura 9. Diagrama de estructura de presentación Vista Puntaje

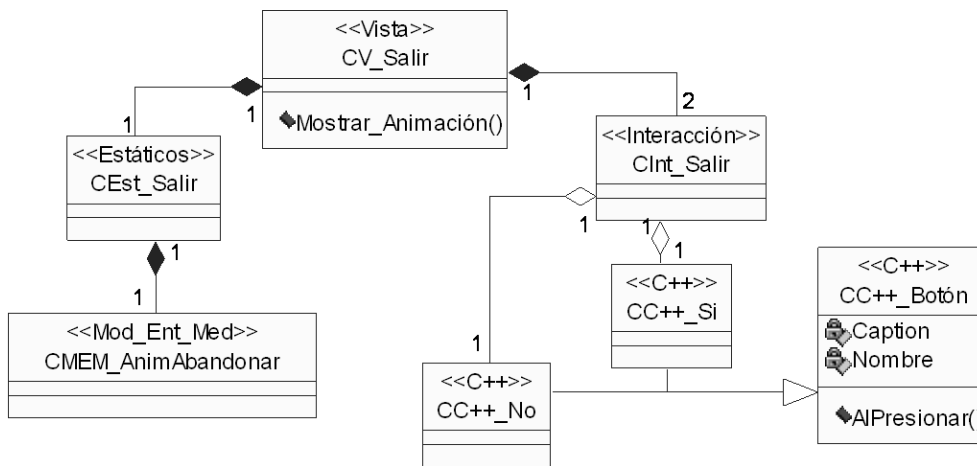


Figura 10. Diagrama de estructura de presentación Vista Salir

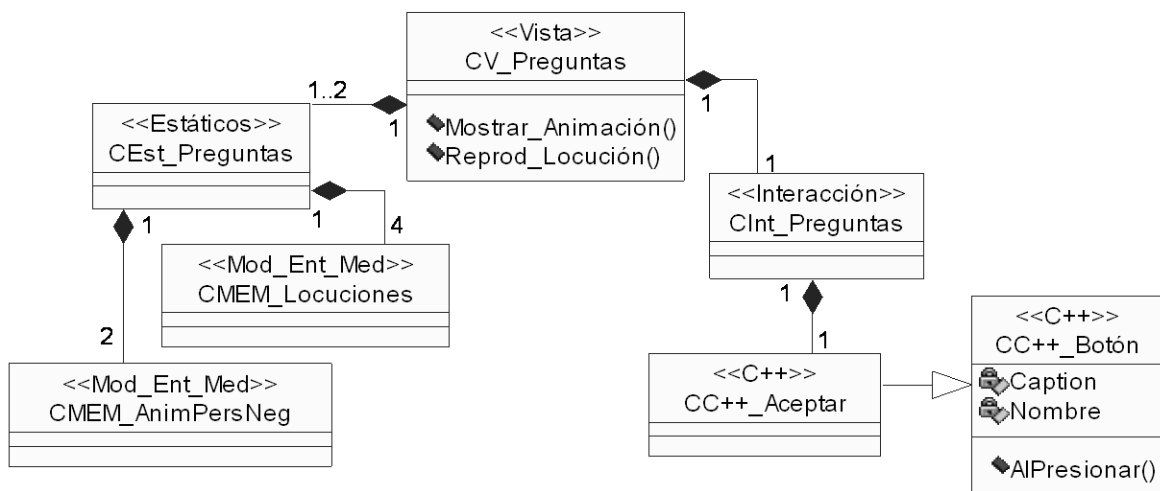


Figura 11. Diagrama de estructura de presentación Vista Preguntas

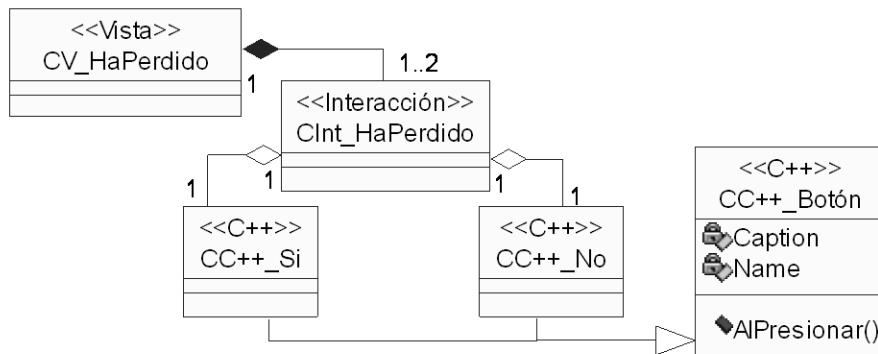


Figura 12. Diagrama de estructura de presentación Vista HaPerdido

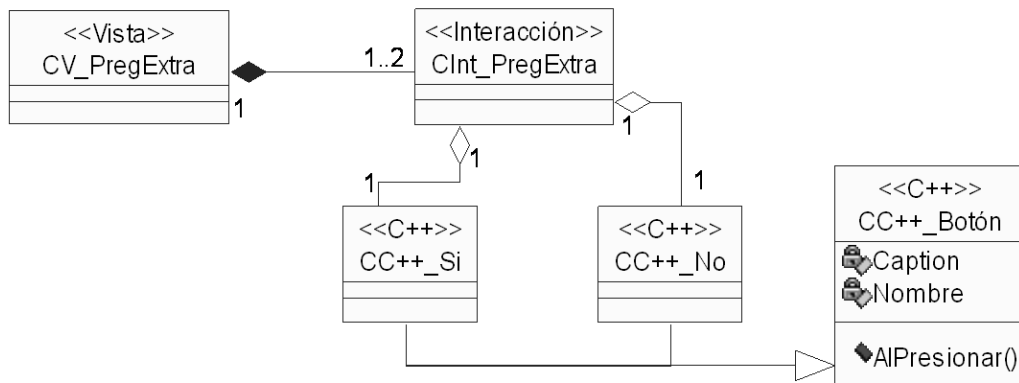


Figura 13. Diagrama de estructura de presentación Vista PreguntaExtra

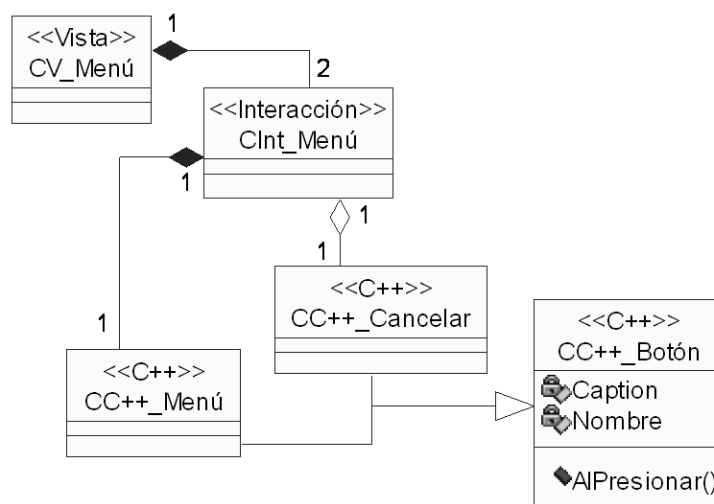


Figura 14. Diagrama de estructura de presentación Vista Menú

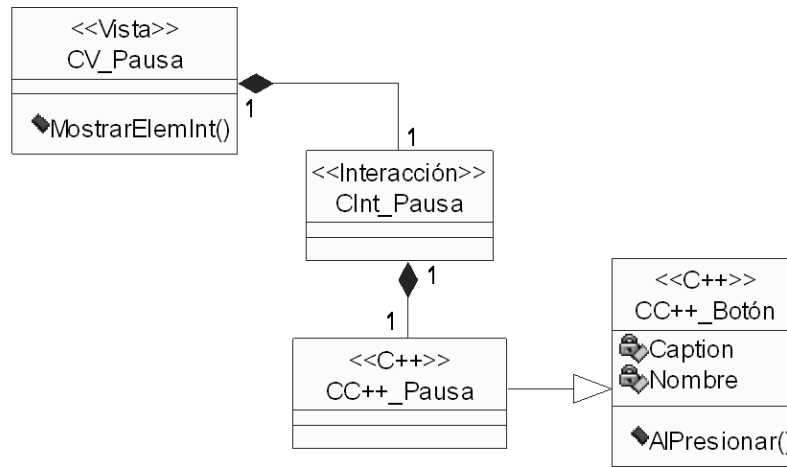


Figura 15. Diagrama de estructura de presentación Vista Pausa

Para ver los diagramas de estructura de presentación de los restantes niveles consultar anexos 4 - 10.

2.8.1.3 Descripción textual de las vistas de presentación

Para nuestro trabajo guiado por ApEM-L se describen detalladamente las vistas de presentación (ver las tablas 1-15), documentándolas para un mejor entendimiento por parte de clientes y desarrolladores, los que tendrán en sus manos un artefacto esencial y significativo para su trabajo que surge para una mejor descripción del contexto productivo del software educativo, bien explícito, incluyendo objetivos pedagógicos, elementos asociados a las vistas y medias a utilizar.

Tabla 1. Descripción textual de la vista Presentación

Descripción Textual de la Vista <i>Presentación</i>	
Actores de la Vista	Usuario
Propósito	Mostrar al usuario una presentación animada del producto sobre la Misión energética
Objetivos pedagógicos	-
Resumen	La vista se inicia cuando el usuario levanta la aplicación, se muestra la presentación de forma animada.
Vistas asociadas	Principal
Referencias	R1
Precondiciones	-
Poscondiciones	La Vista Principal se visualiza automáticamente cuando termina la Presentación.
Curso Normal de los Eventos	

Acciones del Actor	Respuesta del Sistema	Elementos de la Vista	
1. El usuario levanta la aplicación.	2. Se muestra la animación A01 que da presentación al producto y luego queda de fondo la imagen I04 . 3. Al finalizar la presentación, se muestra sobre la imagen una barra de estado “cargando” I03 hasta que el juego cargue completamente. 4. Se muestra la vista Principal .	2.1 La animación será en forma de presentación y su duración será de 10 segundos el patrón de colores a utilizar se generara en la gama de los cálidos y cieras.	
Prioridad		Crítica	
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I01	Mapa de contorno de Venezuela	En localización
	I02	Logo de la Revolución energética	Existente.
	I03	Barra de estado cargando	En construcción
	I04	Fondo relacionado con la Misión Revolución Energética en Venezuela	En construcción
	I60	Imagen de estrella de la bandera	Existente
Video o Animación	A01	Sobre el mapa de Venezuela (I01), se ubicarán en forma de arco las 8 estrellas de la bandera (I60) y finalmente se presenta la animación del logo de la misión energética (I02) y el nombre del juego (T05)	En construcción
Sonido	S01	Música tradicional venezolana	En localización
Texto	T05	Energía para aprender	Existente

Tabla 2. Descripción textual de la vista Principal

Descripción Textual de la Vista <i>Principal</i>	
Actores de la Vista	Usuario
Propósito	Permitir al usuario escoger entre las opciones de menú dentro de la aplicación y brindarle distintos niveles de ayuda con información
Objetivos pedagógicos	Permitir al usuario dominar las opciones del juego.
Resumen	La vista se inicia cuando el usuario levanta la aplicación, se muestran las opciones de trabajo y termina cuando el usuario selecciona alguna de estas
Vistas asociadas	Presentación, Principal, SeleccJugador, SeleccJuegoGuardado, Configuración, Puntaje, PrimerNivel, Salir.
Referencias	R2
Precondiciones	-

Poscondiciones		Las Vistas se visualizan de acuerdo a la elección del usuario.	
Curso Normal de los Eventos			
Acciones del Actor	Respuesta del Sistema		Elementos de la Vista
<p>3. El usuario selecciona una de las opciones de trabajo dentro de la aplicación accede a otra vista y termina la vista Principal.</p> <p>9. El usuario selecciona la opción "Ayuda"</p> <p>11. El usuario selecciona la opción "Salir"</p>	<p>1. Se muestra la imagen de fondo I05.</p> <p>2. Se muestra las opciones de trabajo dentro de la aplicación y las opciones de ayuda y salir.</p> <p>4. Si el usuario seleccionó la opción "seleccionar jugador" se muestra esta vista SeleccJugador.</p> <p>5. Si el usuario seleccionó la opción "comenzar juego" se ejecuta esta Vista PresentPrimerNivel y se termina la Vista Principal.</p> <p>6. Si el usuario seleccionó la opción "juego guardado" se ejecuta esta Vista JuegoGuardado.</p> <p>7. Si el usuario seleccionó la opción "Configuración" se ejecuta esta Vista Configuración.</p> <p>8. Si el usuario seleccionó la opción "Puntaje" se ejecuta esta Vista Puntaje</p> <p>10. Se accede a vista Presentación y se culmina la vista.</p> <p>12. Se accede a la vista Salir</p>		
Cursos Alternos de los Eventos			
Acción		Curso Alterno	
Prioridad		Crítica	
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I05	Fondo relacionado con la Misión Revolución Energética para la presentación de la vista principal.	

Tabla 3. Descripción textual de la vista SeleccJugador

Descripción Textual de la Vista SeleccJugador	
Actores de la Vista	Usuario
Propósito	Permitir al usuario seleccionar entre varias opciones de personajes el que desee para jugar.

Objetivos pedagógicos	Conocer la representación étnica característica del país, así como identificar los valores y hábitos culturales relacionados con cada una de ellas.		
Resumen	La vista se inicia cuando el usuario escoge la opción Seleccionar jugador, se le muestran 8 opciones posibles para escoger el personaje con el que se sienta más identificado y desea jugar y termina cuando el usuario selecciona uno.		
Vistas asociadas	Principal, PrimerNivel		
Referencias	R4		
Precondiciones	-		
Poscondiciones	Usuario listo para comenzar a jugar con un personaje específico seleccionado		
Curso Normal de los Eventos			
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista	
<p>2. Al pasar el Mouse sobre los personajes.</p> <p>5. Selecciona algún personaje</p> <p>7. Escribe texto en la región disponible para esto.</p> <p>9. Selecciona alguna de las opciones "Guardar", "Jugar"</p> <p>12. Si selecciona la opción "Menú"</p>	<p>1. Se muestra la imagen de fondo I06 y sobre esta las imágenes pertenecientes a los prototipos de personajes a seleccionar.</p> <p>3. Se sustituye la imagen del prototipo I07-I14 que tenga el Mouse arriba por otra.</p> <p>4. Se muestra debajo del personaje una región de "Introducción de Texto" para que el usuario escriba el nombre que le desea poner.</p> <p>6. Los elementos interactivos "Jugar" o "Guardar" se habilitarán.</p> <p>8. Si algún otro personaje tenía un nombre, este desaparecerá y solo se quedará activo el último que se escribió.</p> <p>10. Si ha sido seleccionada la opción "Guardar", el personaje al cual se le puso nombre se guardará para ser usado en el juego.</p> <p>11. Si ha sido seleccionada la opción "Jugar" se ejecuta la vista Primer Nivel y se termina esta vista.</p> <p>13. Accede a la Vista Menú.</p>	<p>1.1 Los prototipos de los personajes estarán alineados en dos hileras alternando y mezclando sexos y razas.</p> <p>1.2 Los botones "Guardar" y "Jugar" estarán inhabilitados.</p> <p>3.1 Las imágenes de sustitución serán semejantes a las anteriores pero con un halo luminoso alrededor. I15-I22</p> <p>4.1 Si antes el usuario había seleccionado otro personaje, la región de de texto del último desaparecerá dejando solo visible, el nombre, si lo escribió.</p> <p>6.1 Si el usuario deja la región de de texto en blanco, y ningún personaje tiene puesto nombre, los botones "Jugar" y "Guardar" permanecerán inhabilitados.</p> <p>10.1 El botón "Guardar" se inhabilita.</p>	
Prioridad	Crítica		
Medias a utilizar			
Media	Nombre	Descripción	Estado

Imagen	I06	Fondo de la vista Principal , muestra elementos relacionados con el juego en tonos de colores fríos y opacos.	
	I07	Prototipo de niño blanco	Existente
	I08	Prototipo de niña blanca	Existente
	I09	Prototipo de niño indio	Existente
	I10	Prototipo de niña india	Existente
	I11	Prototipo de niño mestizo	Existente
	I12	Prototipo de niña mestiza	Existente
	I13	Prototipo de niño negro	Existente
	I14	Prototipo de niña negra	Existente
	I15	Prototipo de niño blanco con halo luminoso que le rodea	Existente
	I16	Prototipo de niña blanca con halo luminoso que le rodea	Existente
	I17	Prototipo de niño indio con halo luminoso que le rodea	Existente
	I18	Prototipo de niña india con halo luminoso que le rodea	Existente
	I19	Prototipo de niño mestizo con halo luminoso que le rodea	Existente
I20	Prototipo de niña mestiza con halo luminoso que le rodea	Existente	
I21	Prototipo de niño negro con halo luminoso que le rodea	Existente	
I22	Prototipo de niña negra con halo luminoso que le rodea	Existente	

Tabla 4. Descripción textual de la Vista PresentPrimerNivel

Descripción Textual de la Vista PresentPrimerNivel	
Actores de la Vista	Usuario
Propósito	Mostrar al usuario una presentación donde se brindan orientaciones para el nivel.
Objetivos pedagógicos	Valorar la importancia del cambio de bombillos ahorradores y su impacto en el ambiente.
Resumen	La vista se inicia cuando el usuario selecciona la opción “Comenzar juego” y se muestra la Vista de presentación del primer nivel en la que “luminito” brinda las orientaciones al jugador para cumplir las tareas requeridas para el nivel.
Vistas asociadas	Principal, PrimerNivel, Pausa
Referencias	R6

Precondiciones		-	
Poscondiciones		-	
Curso Normal de los Eventos			
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista	
<p>3. El usuario hace clic sobre la animación A05</p> <p>5. El usuario selecciona alguna de las opciones “Esc” o “Pausa”.</p>	<p>1. Se muestra de fondo la interfaz del jugador I32, sobre ella se ejecuta la animación del elemento indicador del nivel A04 y al terminar esta se muestra la A05 de la mascota I61.</p> <p>2. Si se han seleccionado los personajes (8, 9, 10, 12) ofrece L01a y si el jugador se ha identificado con los personajes (7, 9, 11, 13), se escucha la locución L01b. Al finalizar la locución, la mascota se retira.</p> <p>4. La mascota desaparece, se ejecuta la animación del péndulo A06 y al finalizar se muestra la vista Primer Nivel.</p> <p>6. Si seleccionó “ESC” se muestra la vista Principal y termina esta.</p> <p>7. Si seleccionó “pausa” se muestra la Vista Pausa.</p>	<p>1.1 La imagen de fondo tendrá un color más opaco que la real del juego.</p> <p>1.2 El elemento de la animación A04 tendrá un movimiento pendular en arco que recorre la pantalla de un extremo al otro del área visual.</p> <p>1.3 La mascota de la animación A05 se mostrará por disolución en el medio de la pantalla.</p> <p>2.1 La mascota se retira de la misma forma disolviéndose en el centro de la imagen.</p> <p>4.1 El péndulo se detiene unos segundos en el centro y luego desaparece hacia la izquierda.</p>	
Prioridad		Crítica	
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I32	Representa la interfaz del jugador con el fondo correspondiente a la Vista PrimerNivel.	En construcción
	I34	Elemento que indica comenzar a jugar	En construcción
	I61	Mascota Luminito	En construcción
Animación	A04	Movimiento pendular del elemento indicador de inicio del nivel de derecha a izquierda, se detiene unos segundos en el centro y luego desaparece hacia la izquierda.	En construcción
	A05	Aparece la mascota por disolución en el medio de la pantalla, ofrece L01a o L01b y luego se retira disolviéndose en el centro de la imagen.	En construcción
Locución	L01a	¡Chica! El uso de bombillos no eficientes para la iluminación de la comunidad afecta la salud del	En construcción Luminito: La locución debe hacerse transmitiendo

		<p>planeta... ¿te fijaste que <u>alta</u> está su temperatura?... para ayudar a bajarla debes coleccionar los bombillos que caerán de la siguiente forma:</p> <p>Los <u>bombillos ahorradores</u> los colocarás en el <u>morral</u> ubicado en una esquina de tu pantalla (cambiar I56 por I46).</p> <p>Los <u>bombillos incandescentes</u> los <u>reciclaras</u> colocándolos en el <u>cesto</u> que se encuentra en extremo contrario (cambiar I54 por I55).</p> <p>Todas las tareas has de realizarlas antes de culminar el tiempo destinado para ...¡<u>Adelante!</u></p>	<p>entusiasmo al jugador y haciendo énfasis con la voz al pronunciar las palabras subrayadas y respetando las pausas marcadas con los puntos suspensivos y los saltos de párrafo. La voz a utilizar puede ser femenina o masculina pero siempre simulando la edad del jugador, niños entre 9 y 12 años.</p>
	L01b	<p>¡Chico! Las mismas indicaciones de la L01a para comenzar a jugar.</p>	<p>En construcción</p>

Tabla 5. Descripción textual de la vista PrimerNivel

Descripción Textual de la Vista <i>PrimerNivel</i>		
Actores de la Vista	Usuario	
Propósito	Instalar bombillos ahorradores en el sector residencial.	
Objetivos pedagógicos	Valorar la importancia del cambio de bombillos ahorradores y su impacto en el ambiente.	
Resumen	La vista se inicia cuando termina la presentación del nivel, se describe la tarea fundamental del nivel, que está relacionada con el cambio de bombillos incandescentes por ahorradores y su importancia de la misma, el entorno se divide en cuatro secciones en las que caen de forma aleatoria bombillos incandescentes y ahorradores unidos a burbujas de CO ₂ , los que el jugador ha de coleccionar, tarea por la que recibirá puntos al tiempo que disminuye la temperatura planetaria, estas acciones se repiten en cada una de las secciones del juego, aumentando en velocidad y frecuencia en la caída de los bombillos y el CO ₂ .	
Vistas asociadas	Principal, PresentPrimerNivel, CierrePrimerNivel, Menú, Salir, Pausa	
Referencias	R6, R7, R8, R9,R11	
Precondiciones	El usuario debe haber seleccionado un personaje para jugar en la Vista <i>SeleccJugador</i>	
Poscondiciones	-	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista
	<p>1. Se muestra un fondo I35 y sobre ella ubicados en diferentes partes de la pantalla información sobre:</p> <ul style="list-style-type: none"> • El puntaje T06 (se actualizará solo en 	<p>1.1 El fondo con una imagen del entorno del juego muestra una ciudad iluminada con tonos de color azul opaco y dividido en</p>

<p>7. Movimientos del niño hacia los lados y arriba trasladándose en el área para atrapar los bombillos.</p> <p>9. Al saltar y atrapar un bombillo ahorrador ó si el este toca el personaje aunque no haya saltado.</p> <p>11. Cada 2 bombillos ahorradores colectados</p> <p>13. Al saltar y atrapar un bombillo incandescente ó si el este toca el personaje aunque no haya saltado.</p> <p>15. Al saltar sobre una burbuja de CO₂ I45 o tocarla.</p> <p>17. Al no atrapar un bombillo (llegan a caer al suelo).</p>	<p>dependencia de las acciones)</p> <ul style="list-style-type: none"> • El cronómetro T07 (mostrará el tiempo mientras el juego va transcurriendo). • El termómetro con la temperatura planetaria I36 y sobre estará en lugar de la barra de mercurio la imagen I37. <p>2. Encima del fondo se colocan las I38, I39, I40 delimitando una sección de la otra. En la sección que está delante de la barra de seguridad I38, estará la I41.</p> <p>3. Se muestra el personaje de la imagen I42 (escogido al inicio del nivel usuario).</p> <p>4. Cesto de reciclado I54 (en el se vierten los Incandescentes colectados).</p> <p>5. Morral I56 (en él se vierten los bombillos ahorradores colectados).</p> <p>6. Comenzarán a caer ahorradores e incandescentes (I43, I44) y mientras entre ellos caerán también de forma aleatoria burbujas de CO₂ (I45).</p> <p>8. Utilizar el sistema WAS (W: saltar, A: izquierda D: derecha), ó las teclas direccionales (↑: saltar, ←: izquierda, →: derecha) según la configuración.</p> <p>10. Desaparece el bombillo y sobre el morral aparece y desaparece la I46.</p> <p>12. La barra de mercurio I37 disminuye su posición en dos unidades y se muestra la actualización del puntaje T06 el cual aumenta en 10 unidades.</p> <p>14. Desaparece el bombillo y sobre el cesto de reciclado aparece y desaparece la imagen I55 y se muestra el puntaje T06 que aumenta en 5 unidades.</p> <p>16. Se muestra la actualización del puntaje T06 que aumenta en 10 y la barra de mercurio I37 disminuye en dos.</p> <p>18. Desaparecen del área de juego, la barra de mercurio I37 aumenta su posición en dos. Si la I37 alcanzó su</p>	<p>cuatro secciones separadas unas de otras por una barra de seguridad</p> <p>1.2 Se observa el botón “menú”.</p> <p>3.1 El personaje estará en el centro del área de juego listo para realizar los movimientos.</p> <p>4.1 El cesto se ubica el en el extremo inferior derecho del área de juego.</p> <p>5.1 El morral se ubica en el extremo inferior izquierdo del área de juego.</p> <p>6.1 Los bombillos caerán lentamente y de forma aleatoria desde la parte superior de la imagen.</p> <p>10.1 La imagen I46 dará una sensación de destello para que el usuario se de cuenta que el bombillo que colectó está dentro del morral.</p> <p>14.1 La imagen I55 dará una sensación de destello para que el usuario se de cuenta que el bombillo que colectó ha sido reciclado</p>
--	--	--

<p>19. Al caerse el tercer bombillo ahorrador.</p> <p>22. Si consume dos tercios del tiempo total.</p> <p>24. Si consume el tiempo antes de terminar el nivel.</p> <p>26. El usuario colecta el séptimo bombillo ahorrador.</p> <p>29. El usuario realiza cualquiera de las acciones descritas en los números 7, 9, 11, 13, 15,17, 19, 22, 24, 26.</p> <p>31. Al coleccionar el séptimo bombillo ahorrador.</p>	<p>máxima posición no se moverá de lugar. Se muestra la el puntaje actualizado T06 el cual disminuye en 5. Si no hay puntos acumulados se mantiene en 0.</p> <p>20. Se detiene el juego (pausa) y con él los cronómetros, el puntaje; los bombillos, se activa la animación A12 y al mismo tiempo en el centro del área de juego se muestra la Vista Preguntas.</p> <p>21. Luego de la ventana emergente continúa el juego donde mismo y se reactivan los cronómetros y el puntaje.</p> <p>23. Se escuchará la L05.</p> <p>25. Se escucha la L06 y se muestra la vista Ha perdido.</p> <p>27. Sobre la sección que esta delante de la barra de seguridad se muestra I47, se activa la A15 al tiempo que la I38 se levanta mientras la I40 gira a favor de las manecillas del reloj y la I39 lo hace en sentido contrario. Se corre la I35 poco a poco hacia la izquierda hasta que se muestre la segunda sección de la imagen y un tercio de la tercera y vuelve a bajar la I38 que las delimita mientras la I40 gira a en contra de las manecillas del reloj y la I39 lo hace en sentido contrario. Al culminar A15 el personaje aparecerá en el centro de la nueva área de juego.</p> <p>28. Sobre la segunda sección se coloca la I48 y comienzan a caer lentamente (más rápido que en la sección anterior) y de forma aleatoria, ahorradores e incandescentes. Mientras entre ellos también burbujas de CO₂, de ellas un total 5 se repartirán para que caigan equitativamente.</p> <p>30. Se repiten las respuestas del sistema 8, 10, 12, 14, 18, 20, 21, 23, 25, 27 y 28.</p> <p>32. Sobre la segunda sección se muestra la imagen I49, se activa la A15 al tiempo que la I38 se levanta mientras la I40 gira a favor de las manecillas del</p>	<p>27.1 El morral (I56) y el cesto de reciclado (I54) estarán ubicados en la misma posición que en la sección anterior, el primero en el extremo inferior izquierdo del área de juego y el segundo en el inferior derecho.</p>
---	--	--

<p>34. Si el usuario realiza cualquiera de las acciones descritas en los números 7, 9, 11, 13, 15,17, 19, 22, 24, 26.</p> <p>36. Al coleccionar el séptimo bombillo ahorrador.</p> <p>39. Si el usuario realiza cualquiera de las acciones descritas en los números 7, 9, 11, 13, 15,17, 19, 22, 24, 26.</p>	<p>reloj y la I39 lo hace en sentido contrario. Se corre la I35 poco a poco hacia la izquierda hasta que se muestre la tercera sección de la imagen y un tercio de la cuarta y vuelve a bajar la I38 que las delimita mientras la I40 gira a en contra de las manecillas del reloj y la I39 lo hace en sentido contrario. Al culminar la A15 el personaje I42 aparecerá en el centro de la nueva área de juego.</p> <p>33. Sobre la tercera sección se coloca la I50 y comenzarán a caer lentamente (pero más rápida que en la sección anterior) y aleatoriamente, ahorradores e incandescentes. Mientras caerán también entre ellos burbujas de CO₂ representadas por la I45, de ellas un total de 5 se repartirán para que caigan equitativamente.</p> <p>35. Se repiten las respuestas del sistema 8, 10, 12, 14, 18, 20, 21, 23, 25, 27 y 28.</p> <p>37. Sobre la tercera sección, la que esta delante de la barra de seguridad se muestra la I51, se activa la A15 al tiempo que la I38 se levanta mientras la I40 gira a favor de las manecillas del reloj y la I39 lo hace en sentido contrario. Se corre la I35 poco a poco hacia la izquierda hasta que se muestre la cuarta sección de la imagen y vuelve a bajar la I38 que las delimita mientras la I40 gira a en contra de las manecillas del reloj y la I39 lo hace en sentido contrario. Al culminar la A15 el personaje (I40) aparecerá en el centro de la nueva área de juego.</p> <p>38. Sobre la cuarta sección de la I35, se coloca la I52 y comenzarán a caer lentamente (pero mucho más rápida que en la sección anterior) y de forma aleatoria, bombillos ahorradores e incandescentes. Mientras caerán también burbujas de CO₂, de ellas habrá en total 5 las que se repartirán para que caigan equitativamente.</p> <p>40. Se repiten las respuestas del sistema</p>	<p>37.1 El morral (I56) y el cesto de reciclado (I54) se observan ubicados en la misma posición que en la sección anterior, el primero en el extremo inferior izquierdo del área de juego y el segundo en el inferior derecho.</p>
--	---	--

<p>41. Al coleccionar el séptimo bombillo ahorrador.</p> <p>43. El usuario selecciona alguna de las opciones “pausa”, “esc” o “menú”</p>	<p>8, 10, 12, 14, 18, 20, 21, 23, 25, 27 y 28.</p> <p>42. Sobre la cuarta sección de la imagen I31 aparece la I53. Se detienen los indicadores de puntaje y cronómetro y se muestra la Vista Cierre Nivel.</p> <p>44. Si seleccionó “pausa” se muestra la Vista Pausa</p> <p>45. Si seleccionó “ESC” se detiene el juego y se muestra la Vista Salir</p> <p>46. Si seleccionó “menú” accede a la Vista Menú</p>		
Prioridad		Crítica	
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I35	Entorno del juego (ciudad iluminada) en 3 capas. a) Se representa la calle con edificios, conforma el primer plano del conjunto. b) Se representan las nubes, constituyen el segundo plano del conjunto. c) Se representan los cerros y el cielo, constituye el tercer plano del conjunto.	En construcción
	I36	Imagen del termómetro con la barra indicadora dividida en tres zonas de coloración diferente y en el extremo superior de esta la representación esquemática de calor.	En construcción
	I37	Barra de mercurio del termómetro.	En construcción
	I38	Barra de seguridad que separa las secciones de la ciudad, se coloca como una capa alfa sobre la imagen I35.	Existente
	I39	Rueda dentada ubicada en el extremo superior derecho de la barra de seguridad.	Existente
	I40	Rueda dentada ubicada en el extremo superior izquierdo de la barra de seguridad.	Existente
	I41	Capa alfa sobre la primera sección de la I31 que representa la iluminación de ventanas y calles de color amarillo.	Existente
	I42	Ubicar a los personajes de espalda y con el morral al hombro.	Existente
	I43	Bombillo ahorrador	Existente
	I44	Bombillo incandescente	Existente

	I45	Burbuja de CO2	Existente
	I46	Imagen iluminada del morral	Existente
	I47	Imagen de iluminación de ventanas, color azul brillante (luz de neón), como una capa alfa sobre la primera sección de la ciudad sustituyendo la anterior.	Existente
	I48	Capa alfa sobre la segunda sección de la I35 que representa la iluminación de ventanas y calles de color amarillo.	En construcción
	I52	Capa alfa que se coloca sobre la tercera sección de la I35 que representa la iluminación de ventanas y calles de color amarillo.	En construcción
	I53	Imagen de iluminación de ventanas, color azul brillante (luz de neón), se coloca como una capa alfa sobre la cuarta sección de la ciudad.	En construcción
	I54	Cesto de reciclaje	Existente
	I55	Cesto de reciclaje iluminado	Existente
	I56	Morral, similar al que carga el personaje en su espalda.	Existente
	I62	Personaje negativo	Existente
Animación	A12	Personaje negativo aparece en la esquina inferior derecha de la pantalla y aumentando ligeramente su tamaño, mira hacia el personaje realiza gestos como haciéndole una pregunta. Animación con pocos pasos sólo para dar la sensación de movimiento.	En construcción
	A15	Traslado del niño I42 de una sección a la otra hasta. Esta animación deberá ser en seis pasos y entregadas como una secuencia de imágenes.	En construcción
Locución	L04	“has perdido”	El personaje negativo la hará transmitiendo inconformidad.
	L05	“de prisa, resta poco tiempo para culminar”	Luminito con voz de alerta
Texto	T06	Indicador de puntos que va obteniendo el jugador.	
	T07	00:00 (min. y seg. que se irán incrementando en dependencia del tiempo transcurrido en el juego)	

Tabla 6. Descripción textual de la Vista CierrePrimerNivel

Descripción Textual de la Vista CierrePrimerNivel			
Actores de la Vista	Usuario		
Propósito	Dar la posibilidad al usuario de responder una pregunta opcional para aumentar su puntuación.		
Objetivos pedagógicos	Propiciar la toma de decisiones ante una situación específica dado el contenido que se trabajó en el nivel.		
Resumen	La vista se inicia cuando el jugador termina el 1er nivel y se le da la posibilidad de responder una pregunta opcional. Se muestra la animación que premia al jugador y da paso al segundo nivel.		
Vistas asociadas	Principal, PrimerNivel, Preguntas, Menú, Pausa		
Referencias	R6, R7, R8		
Precondiciones	El usuario debe haber terminado la vista PrimerNivel		
Poscondiciones	-		
Curso Normal de los Eventos			
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista	
4. Selecciona la opción "Menú" o "Pausa"	1. Se muestra como fondo la I57 , luego se ejecuta la A08 y al mismo tiempo se muestra la Vista Pregunta Extra . 2. Se muestra la animación A09 y al terminar la Vista Presentación Segundo Nivel . 3. Se guardará automáticamente toda la información, incluyendo puntaje, nivel, personaje seleccionado, etc. 5. Accede a las Vistas Menú o Pausa .	1.1 La animación se ubica en la mitad izquierda del área de juego	
Prioridad	Crítica		
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I57	Muestra elementos relacionados con el nivel en tonos de colores fríos y opacos.	Existente
	I59	Bandera venezolana, en la zona de las estrellas está solo delimitada por una línea de puntos.	Existente
Animación	A08	La mascota (I58) muestra movimientos, aumenta ligeramente de tamaño señalando el cuadro de texto que va aparecer en la pantalla.	En construcción
	A09	Se agregan las dos primeras estrellas (I60) a la bandera del país (I59) mientras esta ondea en la mitad izquierda de la pantalla.	En construcción

Tabla 7. Descripción textual de la vista SeleccionadoGuardado

Descripción Textual de la Vista SeleccionadoGuardado			
Actores de la Vista		Usuario	
Propósito		Permitir al usuario seleccionar un juego que haya iniciado antes.	
Objetivos pedagógicos		-	
Resumen		La vista se inicia cuando el usuario escoge la opción Juego guardado, se le muestran los 10 últimos usuarios que hayan jugado para escoger y termina cuando selecciona o no uno.	
Vistas asociadas		Principal, Menú	
Referencias		R10	
Precondiciones		El usuario debe haber estado jugado antes.	
Poscondiciones		Vista del juego en el nivel donde estaba antes.	
Curso Normal de los Eventos			
Acciones del Actor	Respuesta del Sistema		Elementos de la Vista
2. El usuario pasa el cursor sobre cualquier fila de la tabla. 4. El usuario selecciona alguna fila. 6. El usuario selecciona alguna de las opciones "aceptar" o "cancelar".	1. Muestra la imagen de fondo I23 y sobre este una tabla con 3 columnas: No. Nombre y Nivel. Esta contendrá 10 filas (sin incluir la del título) en las cuales aparecerá el nombre y el nivel de los últimos 10 que hayan jugado. 3. La seleccionada cambia de color o resalta de forma diferente a las demás. 5. El botón "aceptar" se habilitará. 7. Si seleccionó "aceptar" se cargará el juego seleccionado y se mostrará la vista del nivel que corresponda. 8. Si seleccionó "cancelar" accede a la Vista Menú .		1.1 Los elementos del fondo estarán en tonos de colores fríos y opacos. 1.2 Los datos serán de los usuarios estarán ordenados por nivel. 1.3 El botón "aceptar" estará inhabilitado.
Prioridad	-		
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I23	Fondo de la vista Juego Guardado , muestra elementos relacionados con el juego.	Existente
Texto	T01	No.	
	T02	Nombre	
	T03	Nivel	

Tabla 8. Descripción textual de la vista Configuración

Descripción Textual de la Vista <i>Configuración</i>		
Actores de la Vista	Usuario	
Propósito	Permitir al usuario configurar el juego a su manera.	
Objetivos pedagógicos	-	
Resumen	La vista se inicia cuando el usuario selecciona la opción de configuración, y se le brindan opciones para configurar su propio juego antes de comenzar.	
Vistas asociadas	Principal, Nivel, Menú.	
Referencias	R5	
Precondiciones	-	
Poscondiciones	Juego a iniciar estará configurado según como lo hizo el usuario	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista
<p>4. Selecciona alguna de las dos animaciones</p> <p>7. Hace clic y mueve hacia la izquierda o hacia la derecha la I25 para el control del volumen.</p> <p>9. Selecciona la opción "Pantalla Completa" o "Pantalla Media".</p>	<p>1. Se muestra la imagen de fondo I24 y sobre este, dos animaciones A02 y A03: una representa el parpadeo de las teclas WAS sobre el teclado, la otra representa de las teclas direccionales.</p> <p>2. Se muestra un indicador que se desplaza por la barra de sonido para el control del volumen</p> <p>3. Se muestra un cuadro con un campo y checkbox encima sobre el que el jugador podrá seleccionar si desea jugar con la pantalla completa o media</p> <p>5. Si seleccionó la primera animación A02, esta se sustituye por la imagen I29, indicando que se escoge como la configuración a utilizar.</p> <p>6. Si seleccionó la segunda, esta se sustituye por la I30 indicando que se escoge como la configuración a utilizar.</p> <p>8. Según la dirección en la que se traslade aumentará o disminuirá el volumen.</p> <p>10. Según la selección desaparece la I28 y se ubica sobre el cuadro que corresponda a la opción que está activa.</p>	<p>1.1 La primera animación se realizará con colores azules.</p> <p>1.2 La segunda se realizará con una gama de colores verdes para diferenciarla de los colores del resto del juego de teclas.</p> <p>2.1 El botón "Guardar" se muestra inhabilitado.</p> <p>3.1 Antes de configurar el modo la pantalla estará predeterminada por el sistema como media.</p>

11. Selecciona alguna de las opciones “aceptar” o “cancelar”		12. Si seleccionó “Aceptar” se guardan los cambios realizados en la configuración para el juego, se muestra la vista Principal y termina la vista Configuración . 13. Si seleccionó “cancelar” accede a la Vista Menú y no guarda los cambios realizados.	
Prioridad		Crítica	
Mejoras		-	
Medias a utilizar			
Media	Nombre	Descripción	Estado
Animación	A02	Animación que representa el parpadeo de las teclas WAS sobre el teclado con colores azules.	En construcción
	A03	Animación que representa el parpadeo de las teclas direccionales izquierda, derecha y arriba sobre el teclado, con una gama de colores verdes para diferenciarla de los colores del resto del juego de teclas.	En construcción
Imagen	I24	Imagen de fondo de pantalla “Configurar”, muestra la imagen de un teclado, la barra de sonido con una pequeña bocina y los textos “teclado”, “volumen”, “media pantalla” y “pantalla completa”.	En construcción
	I25	Indicador que se desplaza por la barra de sonido para el control del volumen	En construcción
	I26	Cuadro que representa el campo sobre el cual el jugador podrá seleccionar el modo de juego “Pantalla completa”	En construcción
	I27	Cuadro que representa el campo sobre el cual el jugador podrá seleccionar el modo de juego “Media pantalla”	En construcción
	I28	CheckBox	En construcción
	I29	Sistema de teclas WAS con coloración azul brillante.	En construcción
	I30	Teclas direccionales con coloración verde brillante.	En construcción

Tabla 9. Descripción textual de la vista Puntaje

Descripción Textual de la Vista <i>Puntaje</i>			
Actores de la Vista		Usuario	
Propósito		Mostrar al usuario ver los mejores puntajes registrados en la PC.	
Objetivos pedagógicos		-	
Resumen		La vista se inicia cuando el usuario selecciona la opción Puntaje y se muestran los diez mejores puntajes registrados en la PC.	
Vistas asociadas		Principal, Menú	
Referencias		R12	
Precondiciones		-	
Poscondiciones		-	
Curso Normal de los Eventos			
Acciones del Actor		Respuesta del Sistema	Elementos de la Vista
2. Si selecciona la opción “menú” accede a la Vista Menú .		1. Se muestra un fondo I31 y sobre este una tabla con 4 columnas que tendrán como título: T01 , T02 , T03 y T04 y 10 filas (sin incluir la del título), esta tendrá el nombre, nivel y los mejores puntajes de los 10 usuarios registrados.	1.1 El fondo muestra elementos en tonos de colores fríos y opacos. 1.2 Estos datos aparecerán ordenados descendientemente por puntaje.
Prioridad			
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I31	Fondo de Puntaje , muestra elementos relacionados con el juego.	En construcción
Texto	T01	No.	
	T02	Nombre	
	T03	Nivel	
	T04	Puntaje	

Tabla 10. Descripción textual de la vista Pregunta Extra

Descripción Textual de la Vista <i>PreguntaExtra</i>	
Actores de la Vista	Usuario
Propósito	Incentivar al usuario a responder una pregunta extra
Objetivos pedagógicos	-
Resumen	Se muestra un mensaje preguntando si desea responder otra pregunta

Vistas asociadas	CierrePrimerNivel, Preguntas	
Referencias	R 6.12	
Precondiciones	El usuario debe haber terminado el Primer Nivel del juego	
Poscondiciones		
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista
2. Si selecciona "Si" 4. Si selecciona "No"	1. Se muestra un mensaje preguntando si desea responder otra pregunta y aumentar 100 puntos al puntaje: Si o No 3. Se mostrará la Vista Preguntas 5. Desaparece la Vista PreguntaExtra	

Tabla 11. Descripción textual de la vista Preguntas

Descripción Textual de la Vista Preguntas		
Actores de la Vista	Usuario	
Propósito	Realizar preguntas de conocimiento al usuario	
Objetivos pedagógicos	Brindar oportunidades al jugador para medir los conocimientos adquiridos en la realización de las tareas orientadas en cada nivel.	
Resumen	La vista se inicia desde cada nivel en diferentes circunstancias durante el juego o al elegir querer realizar una pregunta extra.	
Vistas asociadas	PrimerNivel, CierrePrimerNivel, HaPerdido	
Referencias	R6.11, R6.12	
Precondiciones	-	
Poscondiciones		
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista
2. Selecciona alguna de las respuestas y da clic en Aceptar 4. Si se accedió al usuario a través de la Vista Pregunta Extra 7. Si se accedió a la	1. Se muestra una pregunta con posibles asociadas la misma. 3. El elemento interactivo "Aceptar" estará inhabilitado. 5. Si la respuesta dada por el usuario es correcta. Se escucha la L07 y el puntaje obtenido por el jugador aumentará en 100 unidades. Desaparece la vista. 6. Si la respuesta dada por el usuario es incorrecta. Se escucha la locución L08 y desaparece la ventana.	

<p>vista a través de la Vista Primer Nivel luego de caerse el tercer bombillo.</p>	<p>8. Si la respuesta dada por el usuario es correcta se escucha la locución L02, desaparece la ventana emergente y se ejecuta la A13 sobre la pantalla del nivel.</p> <p>9. Si la respuesta dada por el usuario es incorrecta. Se escucha la locución L03 y se activa la animación A14.</p> <p>10. Se volverá a mostrar una vez más otra pregunta de forma aleatoria y se repetirán las mismas acciones descritas anteriormente. Nota: Solo se dará una oportunidad para errar, o sea la vista se mostrará dos veces nada más. Si la vista se muestra 2 veces y la respuesta es incorrecta, se escucha L04, desaparece y se muestra la vista HaPerdido.</p>		
<p>Prioridad</p>		<p>-</p>	
<p style="text-align: center;">Medias a utilizar</p>			
<p>Media</p>	<p>Nombre</p>	<p>Descripción</p>	<p>Estado</p>
<p>Locución</p>	<p>L02</p>	<p>“Respuesta correcta”</p>	<p>El personaje negativo hará la locución con voz triste.</p>
	<p>L03</p>	<p>“Respuesta incorrecta”</p>	<p>El personaje negativo hará la locución con voz alegre.</p>
	<p>L07</p>	<p>“Respuesta correcta, has ganado 100 puntos”</p>	<p>Luminito expresando alegría</p>
	<p>L08</p>	<p>“Respuesta incorrecta, tu puntaje se mantiene”</p>	<p>Luminito expresando tristeza</p>
<p>Animación</p>	<p>A13</p>	<p>Personaje negativo con estado de ánimo triste, disminuye de tamaño y desaparece de la vista.</p>	<p>En construcción</p>
	<p>A14</p>	<p>Personaje negativo (I62) con estado de ánimo alegre, crece un poco más que en la animación A13, mira hacia el personaje y realiza gestos como si estuviera haciéndole una pregunta.</p>	<p>En construcción</p>

Tabla 12. Descripción textual de la vista HaPerdido

<p style="text-align: center;">Descripción Textual de la Vista <i>HaPerdido</i></p>	
<p>Actores de la Vista</p>	<p>Usuario</p>
<p>Propósito</p>	<p>Decidir comenzar a jugar el nivel otra vez</p>
<p>Objetivos pedagógicos</p>	<p>Brindar la oportunidad de reintentar ganar el juego.</p>

Resumen	Cuando el usuario haya perdido un nivel por no responder correctamente las preguntas realizadas en el mismo o se le haya acabado el tiempo designado para realizar las tareas de algún nivel, se le preguntará si desea jugar de nuevo si acepta comienza el nivel sino va a la vista Principal .	
Vistas asociadas	PrimerNivel, Preguntas	
Referencias	R 6.10.1	
Precondiciones	-	
Poscondiciones		
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista
2. El usuario selecciona Si 4. El usuario selecciona No	1. Se muestra un mensaje comunicándole que ha perdido y se le pregunta si desea comenzar de nuevo. 3. Se mostrará la Vista de inicio del nivel (en dependencia de en cual haya perdido) para comenzar otra vez el mismo. 5. Se muestra la vista Principal	

Tabla 13. Descripción textual de la vista Pausa

Descripción Textual de la Vista Pausa		
Actores de la Vista	Usuario	
Propósito	Detener o activar el juego	
Objetivos pedagógicos		
Resumen	Desde alguna vista se detiene el juego o se activan las acciones.	
Vistas asociadas	Principal	
Referencias	R 6.13	
Precondiciones	-	
Poscondiciones	La vista vuelve a la normalidad o se detiene según la acción	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista
1. El usuario selecciona el elemento interactivo "pausa"	2. Se detienen las acciones de la Vista y si ya estaba en pausa entonces se volverán a activar las acciones de la misma.	

Tabla 14. Descripción textual de la vista Menú

Descripción Textual de la Vista <i>Menú</i>		
Actores de la Vista	Usuario	
Propósito	Tener un acceso directo desde algunas vistas hasta la Principal	
Objetivos pedagógicos	-	
Resumen	Desde alguna vista se da clic en un elemento interactivo que dará acceso a la Vista Principal	
Vistas asociadas	Principal	
Referencias	R3	
Precondiciones	-	
Poscondiciones	Vista Principal visualizada	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista
1. Se selecciona una de las opciones "menú", "esc" o "cancelar".	2. Se accede a la Vista Principal y se termina la vista Menú .	

Tabla 15. Descripción textual de la vista Salir

Descripción Textual de la Vista <i>Salir</i>		
Actores de la Vista	Usuario	
Propósito	Salir del juego	
Objetivos pedagógicos	-	
Resumen	Desde alguna vista se decide salir de la aplicación, se mostrará una petición de confirmación para decidir, luego termina la vista volviendo a iniciar la anterior o mostrando una animación de salida, según la opción.	
Vistas asociadas	Principal	
Referencias	R13	
Precondiciones	-	
Poscondiciones	La vista termina y cierra la aplicación o se mantiene la anterior.	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista

1. El usuario seleccionó algún elemento interactivo de salida.	2. Se muestra una ventana emergente con opciones de si esta seguro que desea salir.	
3. El usuario decide "Si"	4 Se muestra la animación A07 y al terminar esta se cerrará la aplicación.	
5. El usuario decide "No"	6. Termina la vista.	
Prioridad		-
Medias a utilizar		
Media	Nombre	Descripción
Animación	A07	Créditos
		Estado

Para ver las descripciones textuales del resto de los niveles consultar anexos 11 - 19

2.8.1.4 Diagrama de estructura de navegación

El sistema de navegación general de una multimedia es el que se utiliza a partir del Índice Principal o entre sus secciones principales. [21]

Debido a la ineficiencia del lenguaje base UML para modelar la navegación en el software educativo se ha definido el *diagrama de estructura de navegación*, el que permite representar cuáles serán los elementos de navegación del sistema y cómo se podrá acceder desde uno hacia el otro. El objetivo es proporcionarle una mayor comprensión y asimilación del producto al usuario, para el cual la facilidad de navegación es un objetivo esencial.

Para el desarrollo del diagrama de estructura de navegación se han definido, sobre el concepto de clase, una segunda clasificación de las mismas, por lo que se enriquece la semántica del concepto original de clases, quedando las siguientes: *clase menú*, *clase índice*, *clase consulta* y *clase botón*. [20] Tal como muestra en la figura 16 se realizó un diagrama de estructura de navegación general, en el que han sido representadas las diferentes vistas y las vías de acceso entre las mismas. Se agrupan, los niveles del juego modelados en el subsistema Niveles, para un mejor entendimiento y organización, cuya navegación se puede apreciar en la figura 17.

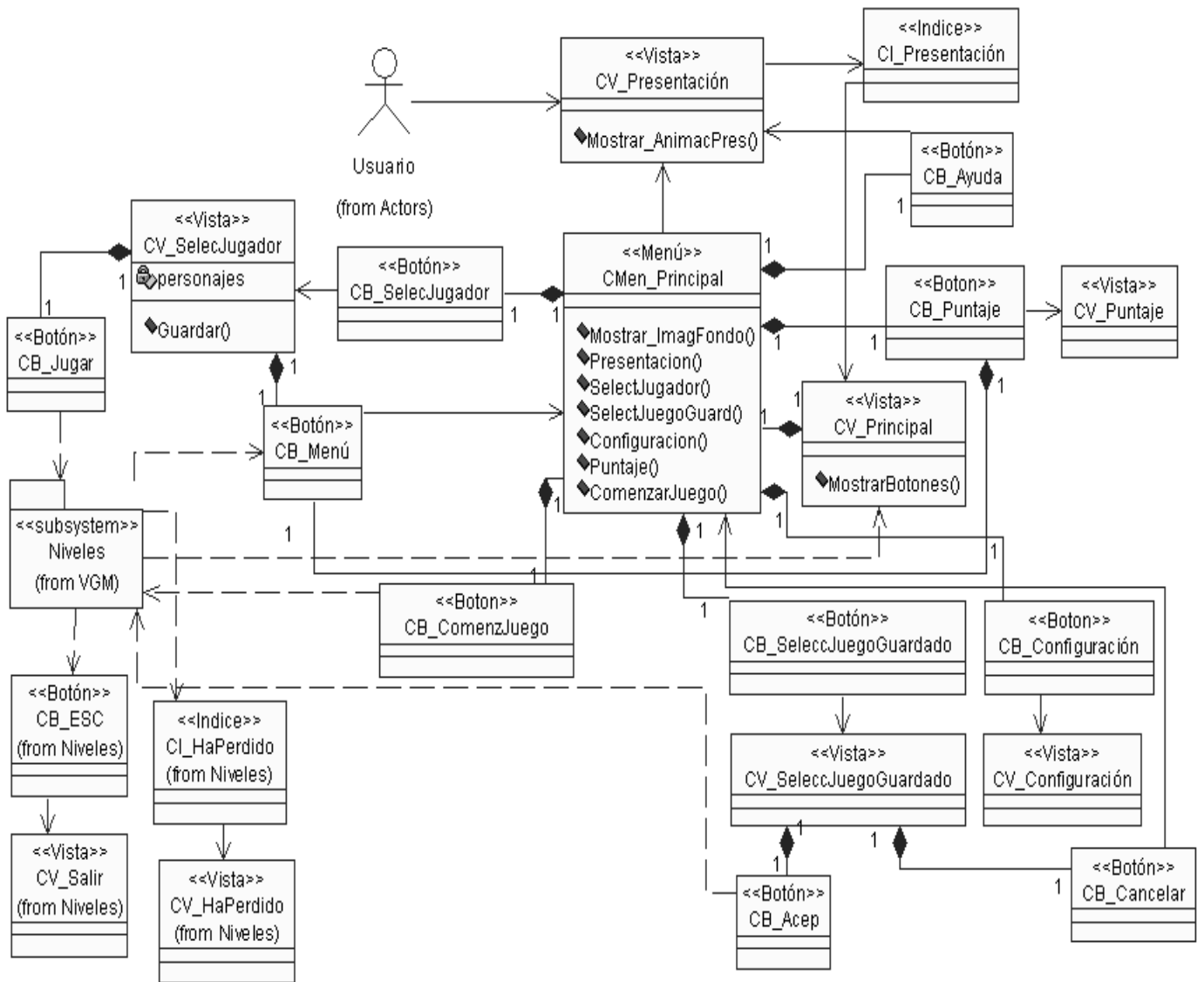


Figura 16. Diagrama de estructura de navegación general

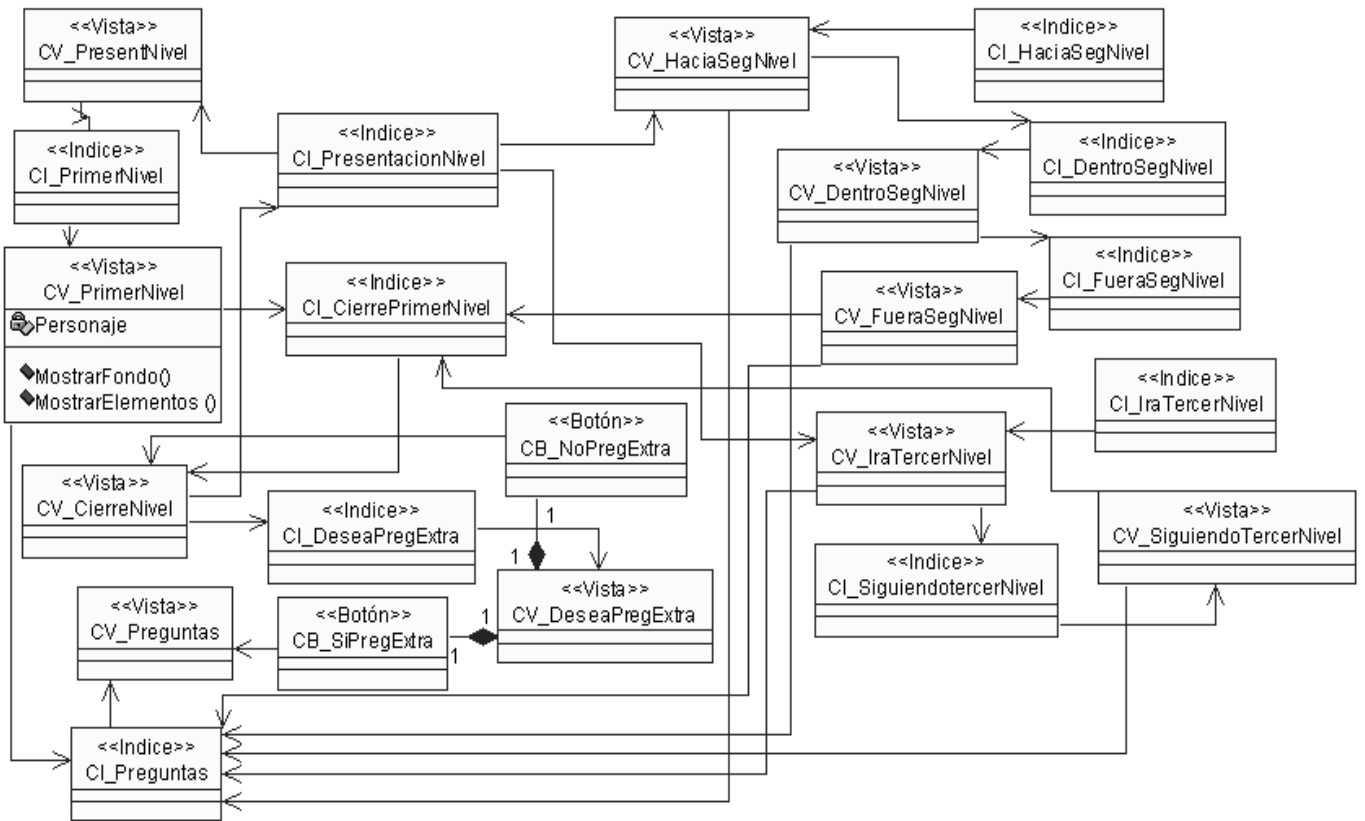


Figura 17. Diagrama de estructura de navegación Subsistema Niveles

2.9 Conclusiones del capítulo

Al finalizar este capítulo se obtuvo toda la información perteneciente al Modelo de dominio del entorno, mostrando sus conceptos fundamentales. Se definieron los requerimientos funcionales y no funcionales que se debe cumplir para obtener un producto más efectivo, además de la estructuración del sistema en 7 subsistemas y 24 vistas de presentación, a las cuales se le realizó sus correspondientes diagramas de estructura de presentación y descripciones textuales. También se incluyó el diagrama de estructura de navegación para permitir utilizar la semántica original de dicho lenguaje en la construcción de estructuras lógicas de navegación.

Capítulo 3

Construcción de la solución propuesta

3.1 Introducción del capítulo

En este capítulo se exponen los artefactos esenciales para el desarrollo que guiarán la implementación del software con tecnología multimedia, para su modelado, apoyado en el flujo de trabajo de diseño, se ha tomado como base el lenguaje de modelado ApEM-L, que ha sido lanzado como una propuesta extensiva de UML. Se construyen los diagramas de clases y de interacción, con pequeñas modificaciones o adiciones, propuestas para aplicaciones educativas, además se presenta el Modelo de despliegue, plasmando el hardware necesario para el funcionamiento de la aplicación.

3.2 Modelo de diseño

En el modelo de diseño es un modelo de objetos que se centra en cómo los requisitos funcionales y no funcionales, junto con otras restricciones del entorno de implementación, tienen impacto en el sistema que se desarrolla. [27]

Este modelo debe ser mantenido durante todo el ciclo de vida del software, sirve de abstracción de la implementación del sistema y es, de ese modo, utilizado como una entrada fundamental de las actividades de este flujo.

ApEM-L plantea tres áreas conceptuales en las que se agrupan los conceptos y modelos: Estructura lógica, Comportamiento dinámico y Gestión del Modelo. La primera esta compuesta en una de sus vistas por el diagrama de clases (son realizados en el modelo de diseño para un mejor entendimiento de cada una de las vistas de presentación) y la segunda se compone entre otros por los diagramas de interacción (representan la manera en que los objetos de la aplicación intercambian mensajes para darle cumplimiento a sus responsabilidades).

3.2.1 Diagrama de clases

Cualquiera de los modelos presentados por ApEM – L define los conceptos claves de la aplicación que modela, las propiedades internas de estos y sus relaciones. Estos conceptos son modelados como clases, describiendo cada una un conjunto de objetos que almacenan información y se comunican para

implementar su comportamiento. La información almacenada se representa como atributos de estas clases y las operaciones a través de los métodos de dichas clases.

Los diagrama de clases modelados, (ver figuras 18 - 26) pertenecientes al área de Estructura lógica y a la Vista estática de ApEM-L, sufre modificaciones de lo planteado semánticamente por UML. Está dividido en dos grandes zonas, la de la izquierda dedicada al árbol jerárquico de las *clases modelo entidad medias* que representan los recursos mediáticos de la aplicación y en la zona de la derecha del diagrama, las clases que controlan la *lógica del negocio* de la aplicación propiamente dicha.

Esta zona de la derecha vuelve a subdividirse en cuatro (4) zonas (Ver Anexo 3). La primera dedicada a las clases *vista*, la contigua a esta y en el extremo superior derecho dedicada a las clases *controladora*, inmediatamente debajo de esta sección, la destinada a las clases *modelo*, quedando una banda inferior derecha dedicada en su extremo derecho a las clases *modelo entidad persistentes* para el tratamiento de la información persistente de la aplicación [20]

Las clases CV de cada una de las vistas definidas heredan de una clase Vista que tiene como atributos: Id_Vista, Ancho, Alto y como operaciones o métodos: Cargar_Vista y Enviar_Evento.

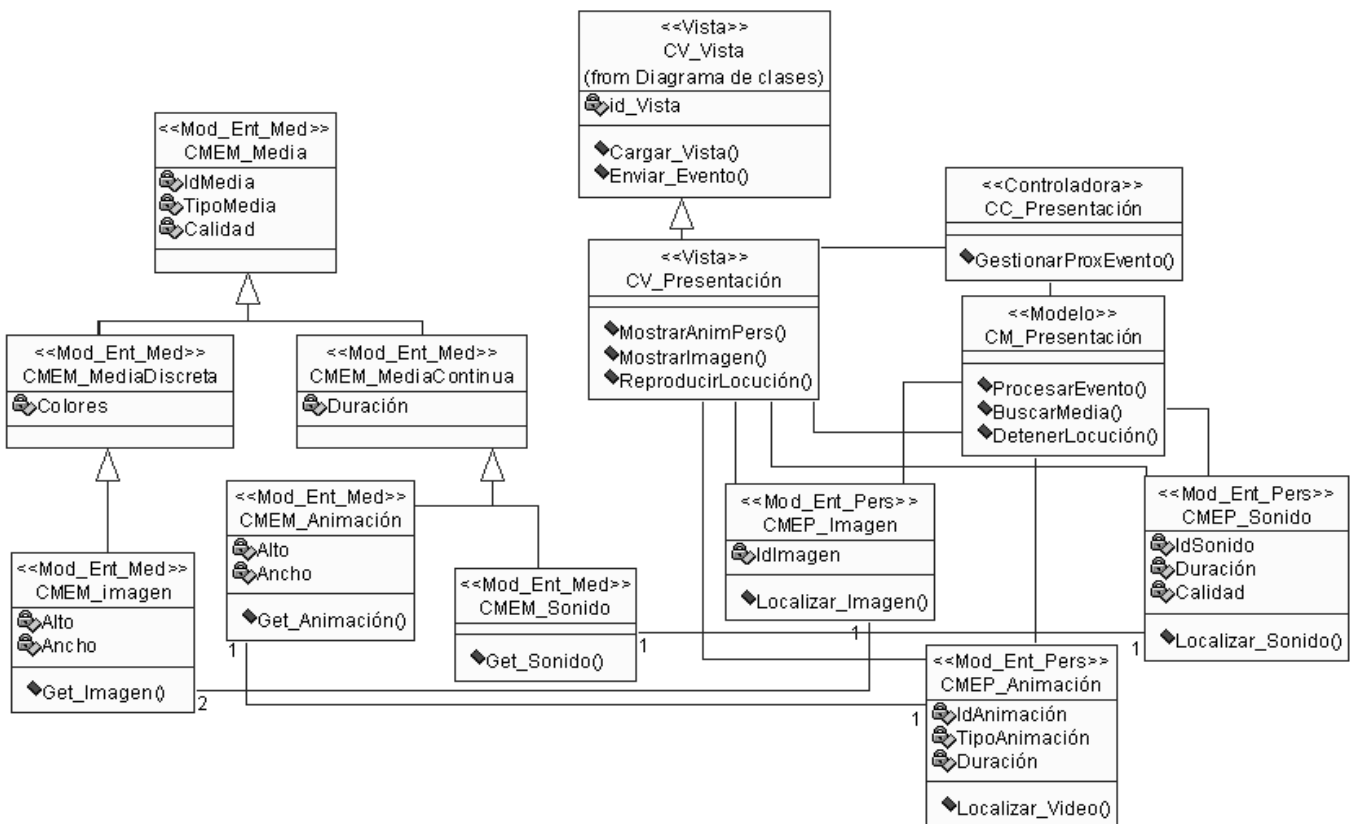


Figura 18. Diagrama de clases Vista Presentación

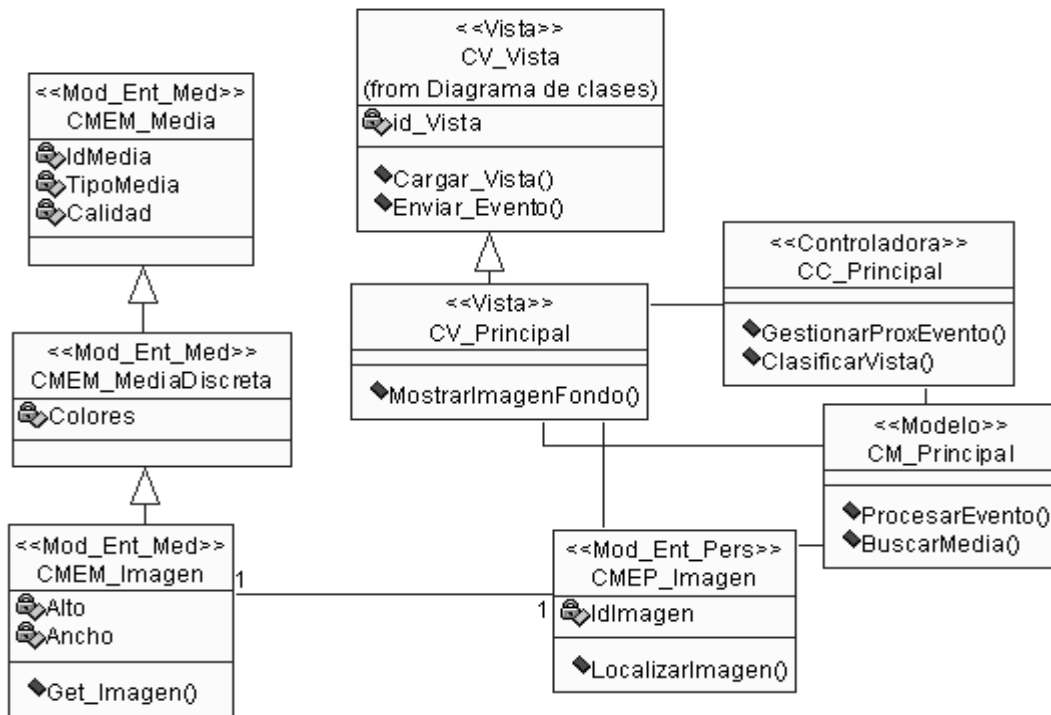


Figura 19. Diagrama de clases Vista Principal

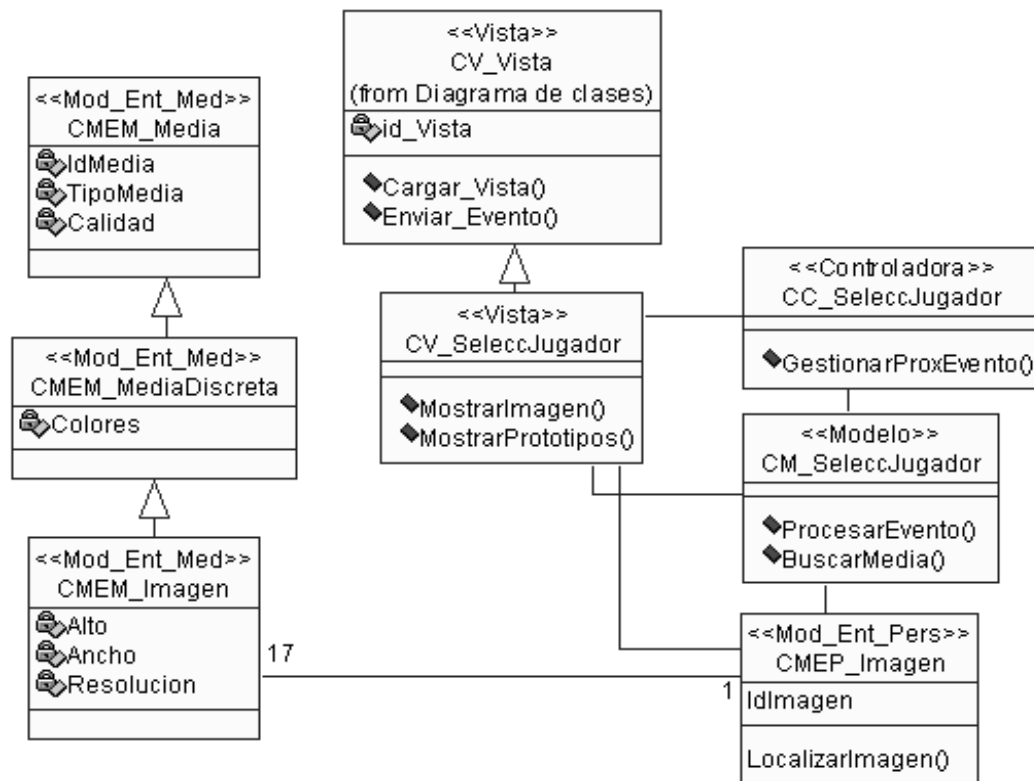


Figura 20. Diagrama de clases Vista SeleccJugador

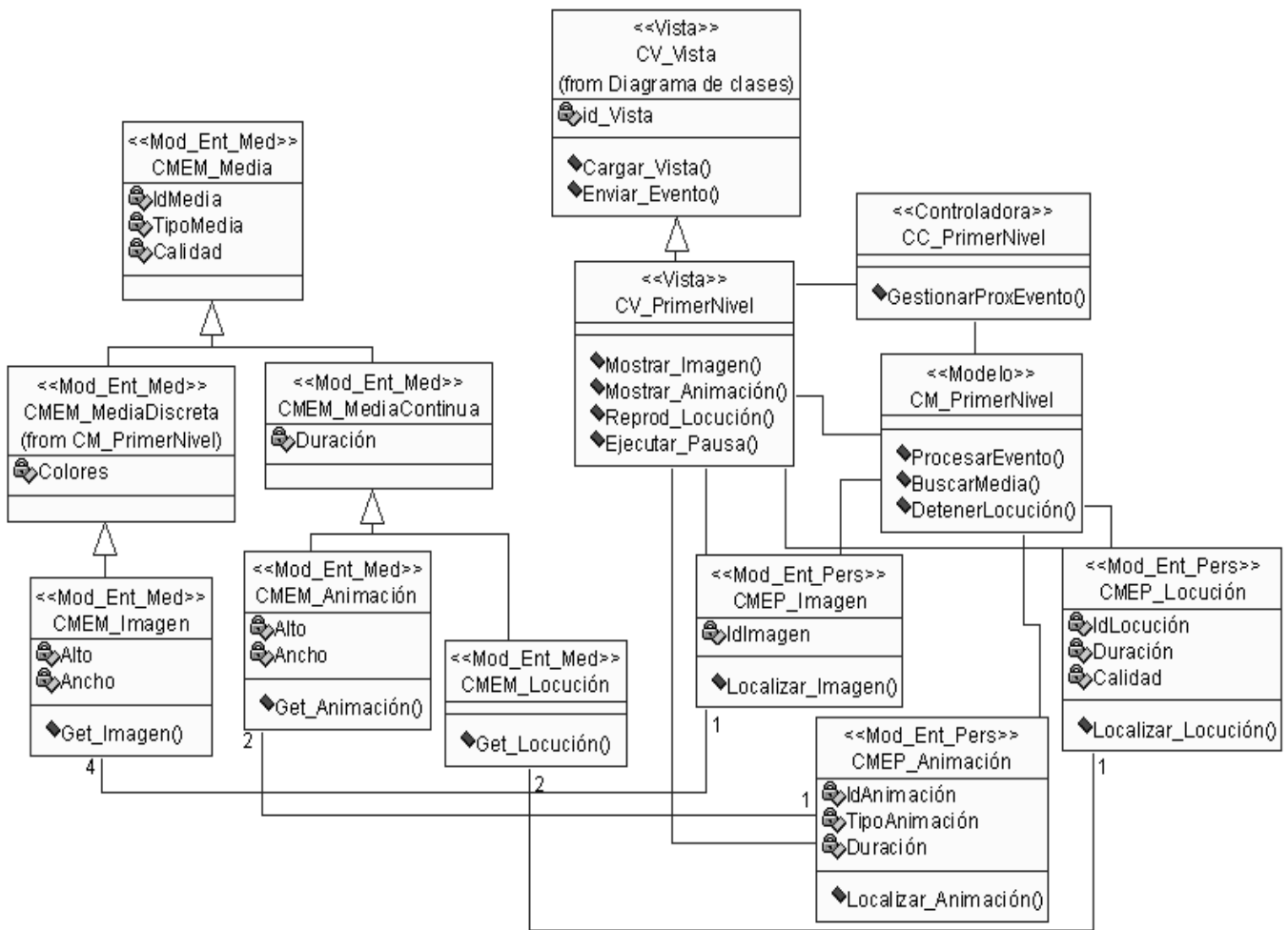


Figura 21. Diagrama de clases Vista Primer Nivel

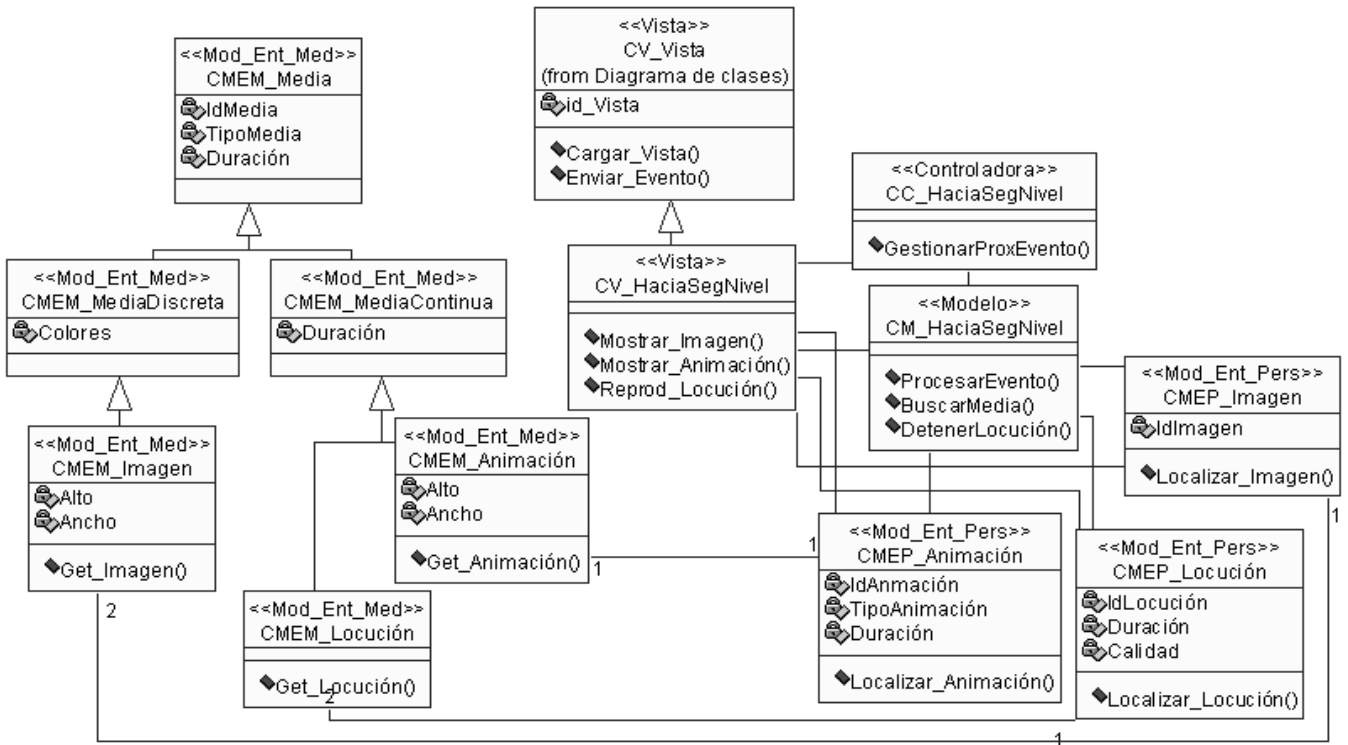


Figura 22. Diagrama de clases Vista HaciaSegNivel

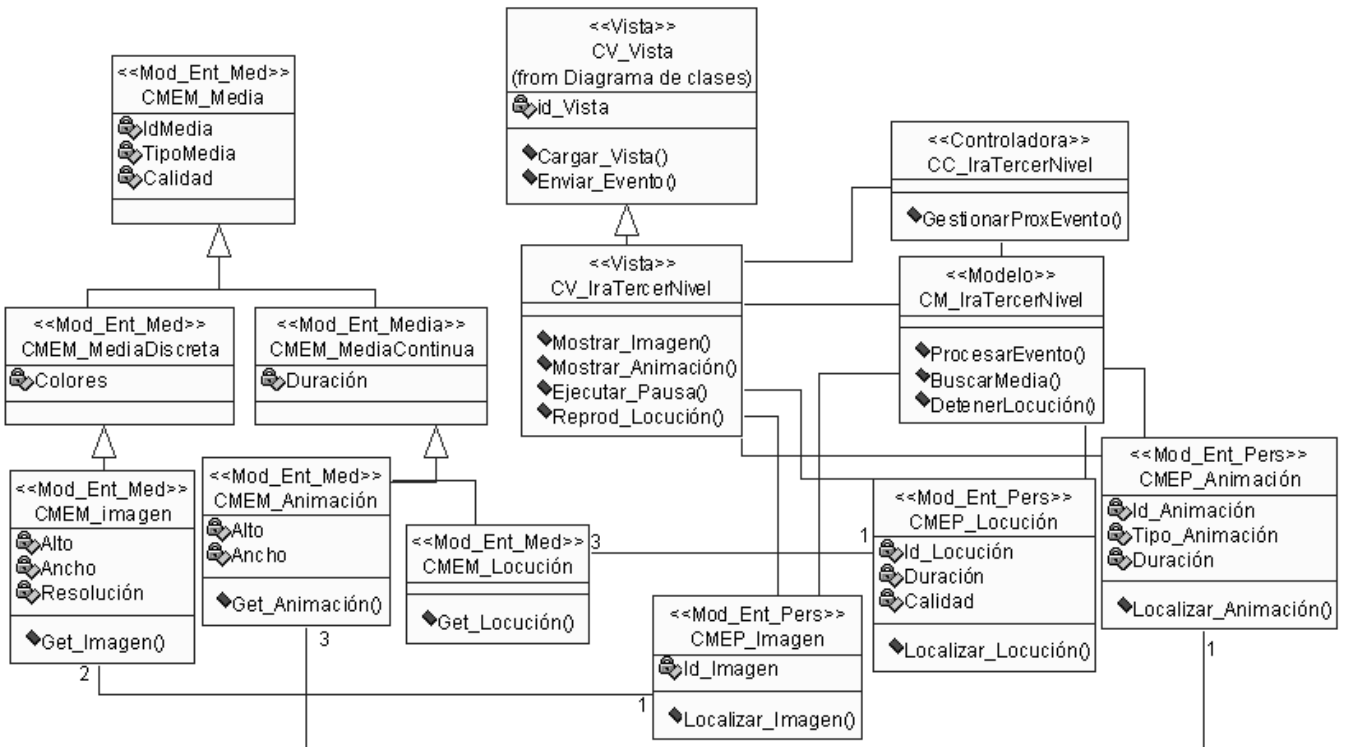


Figura 23. Diagrama de clases Vista IraTercerNivel

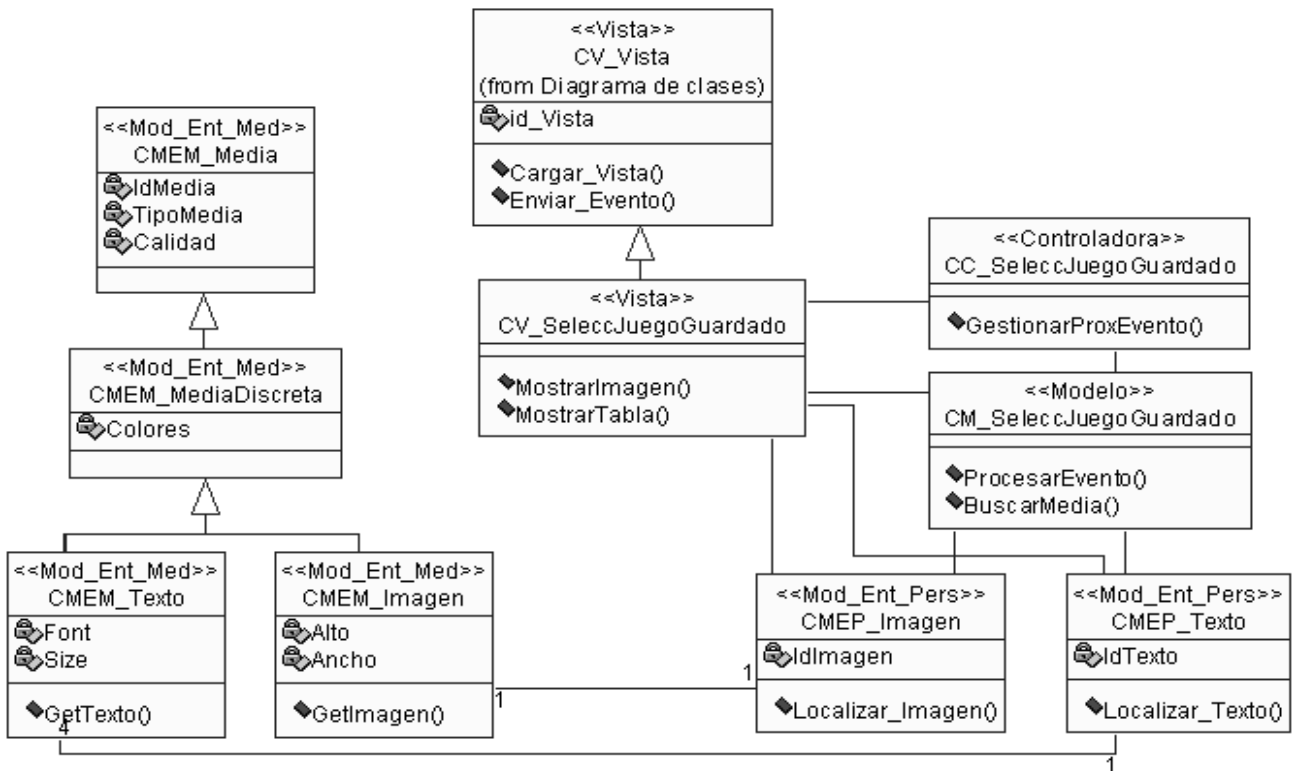


Figura 24. Diagrama de clases Vista SeleccionadoJuegoGuardado

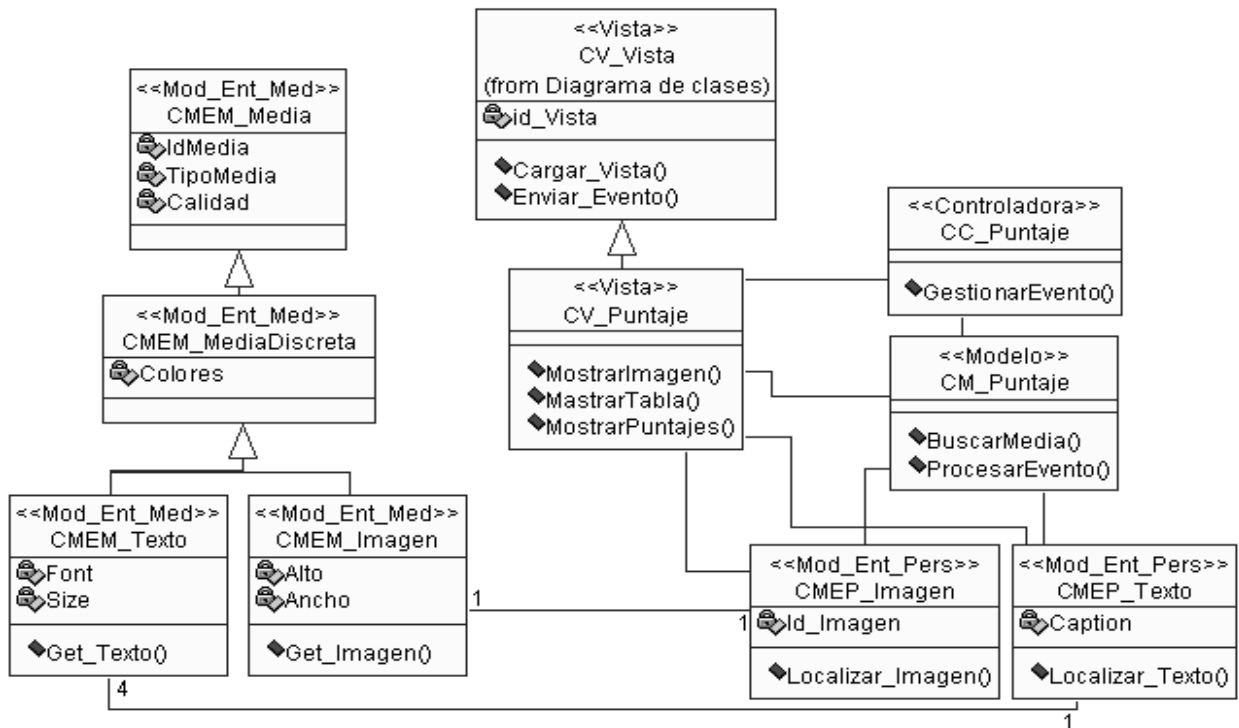


Figura 25. Diagrama de clases Vista Puntajes

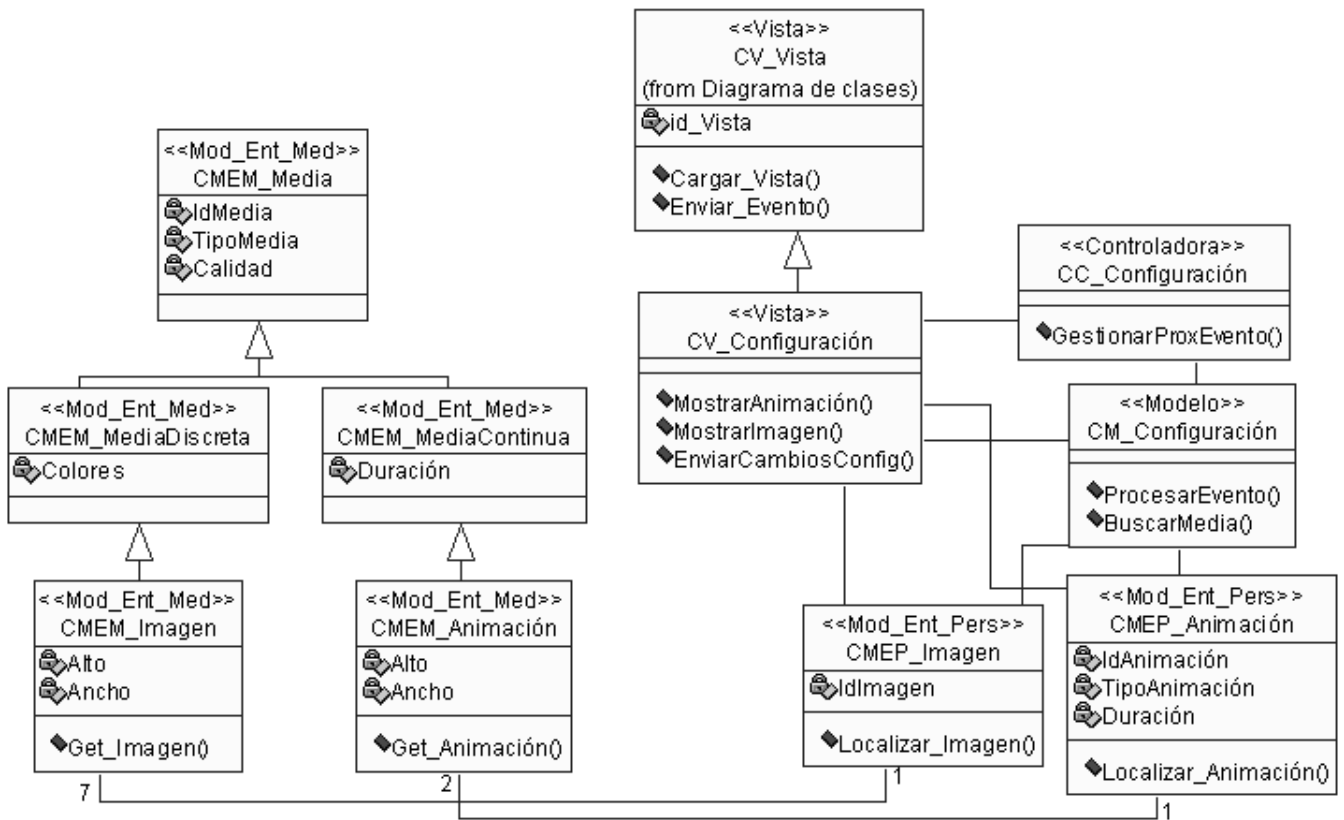


Figura 26. Diagrama de clases Vista Configuración

(Ver restantes Diagramas de clases en Anexos 20 - 22)

3.2.2 Diagramas de interacción

De acuerdo a la agrupación de los conceptos y modelos de ApEM – L, los diagramas de secuencia y colaboración pertenecen al área de Comportamiento dinámico y dentro de esta a la Vista de comportamiento. Con los diagramas de colaboración se logran identificar requisitos y responsabilidades sobre los objetos, creando enlaces entre ellos y añadiéndole mensajes, que deben denotar el propósito del objeto invocante en la interacción con el invocado. Se han representado los diagramas correspondientes a las vistas críticas que contienen medias para ser mostradas. (Ver Figuras 27 - 35 y restantes en Anexos 23 - 24)

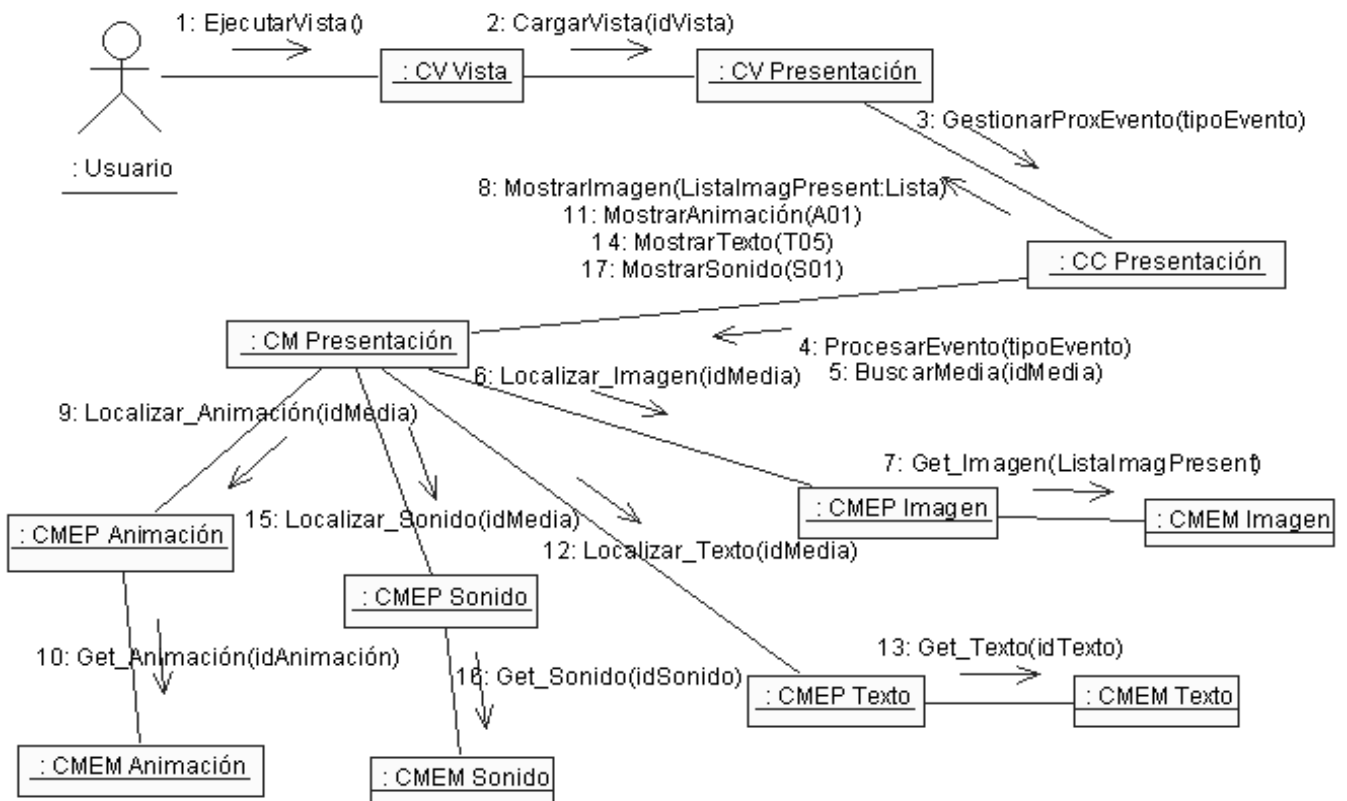


Figura 27. Diagrama de colaboración Vista Presentación

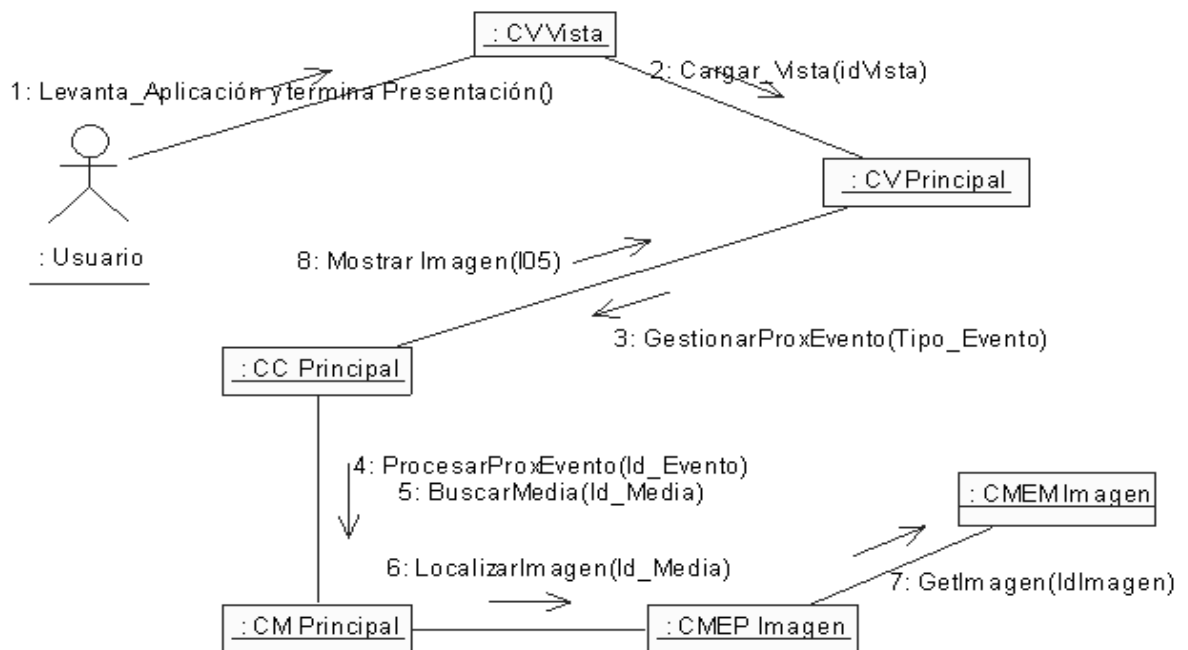


Figura 28. Diagrama de colaboración Vista Principal

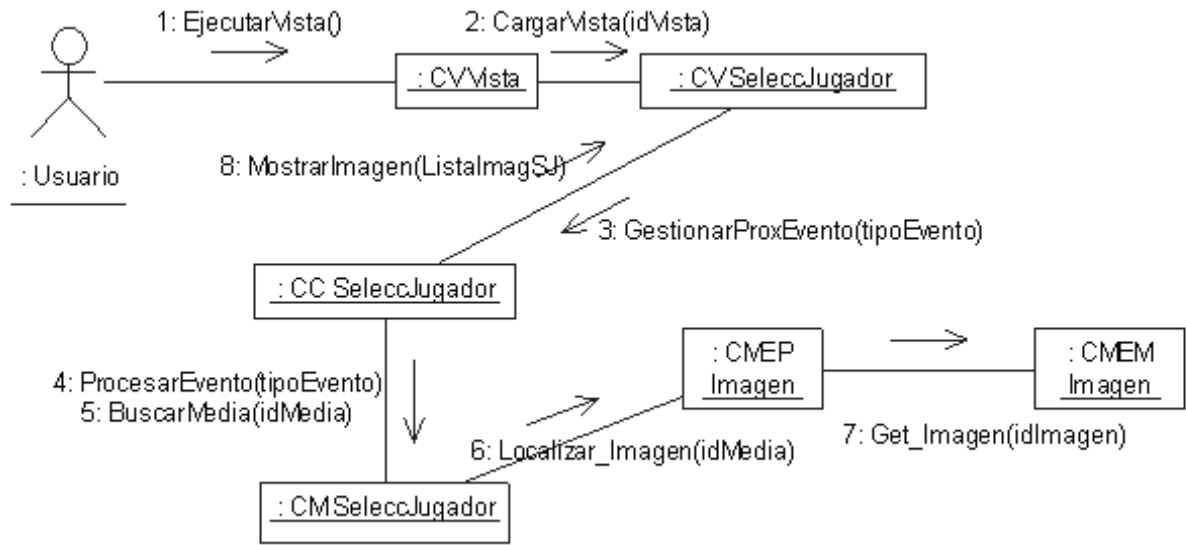


Figura 29. Diagrama de colaboración Vista SeleccJugador

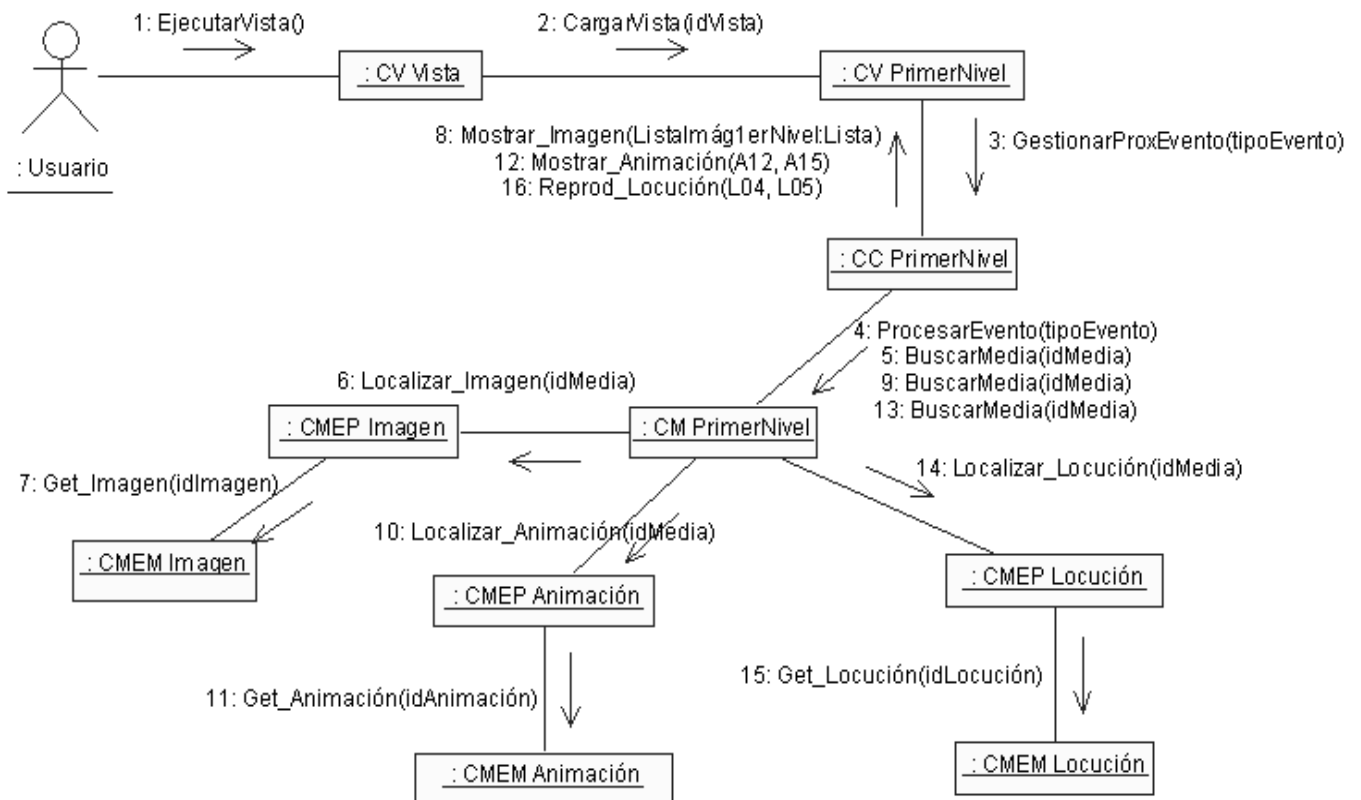


Figura 30. Diagrama de colaboración Vista PrimerNivel

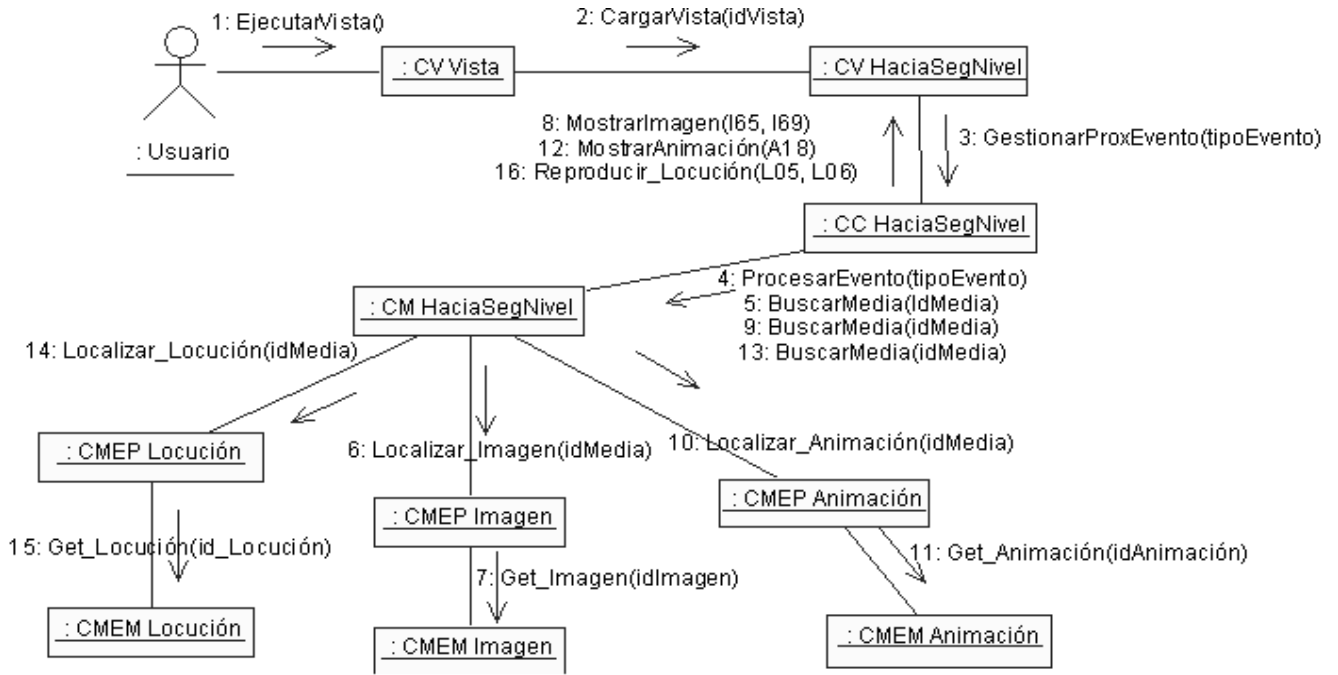


Figura 31. Diagrama de colaboración Vista HaciaSegNivel

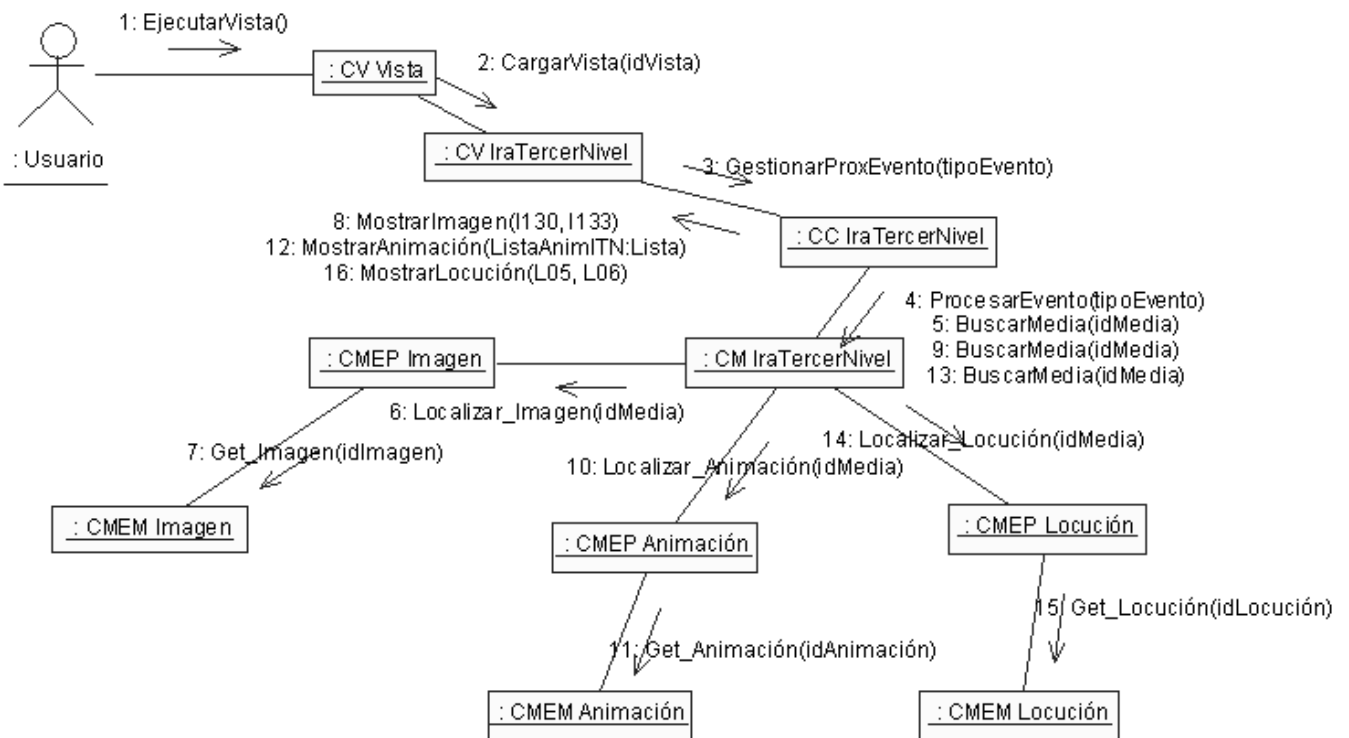


Figura 32. Diagrama de colaboración Vista IraTercerNivel

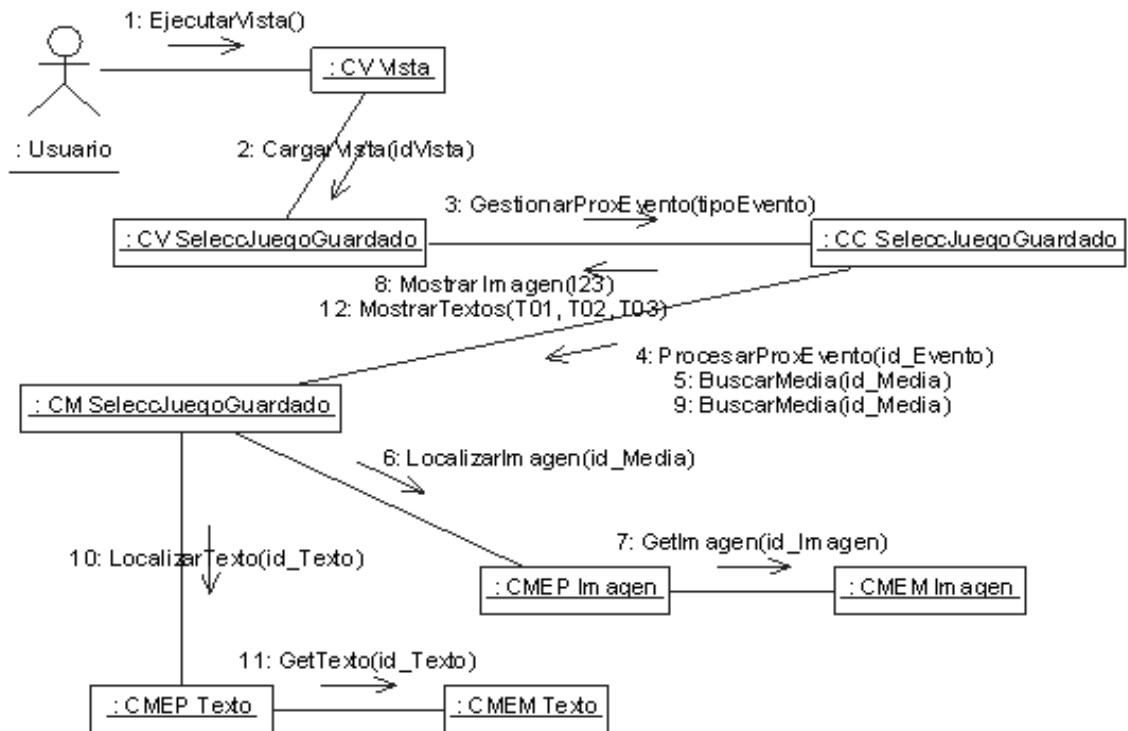


Figura 33. Diagrama de colaboración Vista SeleccionadoJuegoGuardado

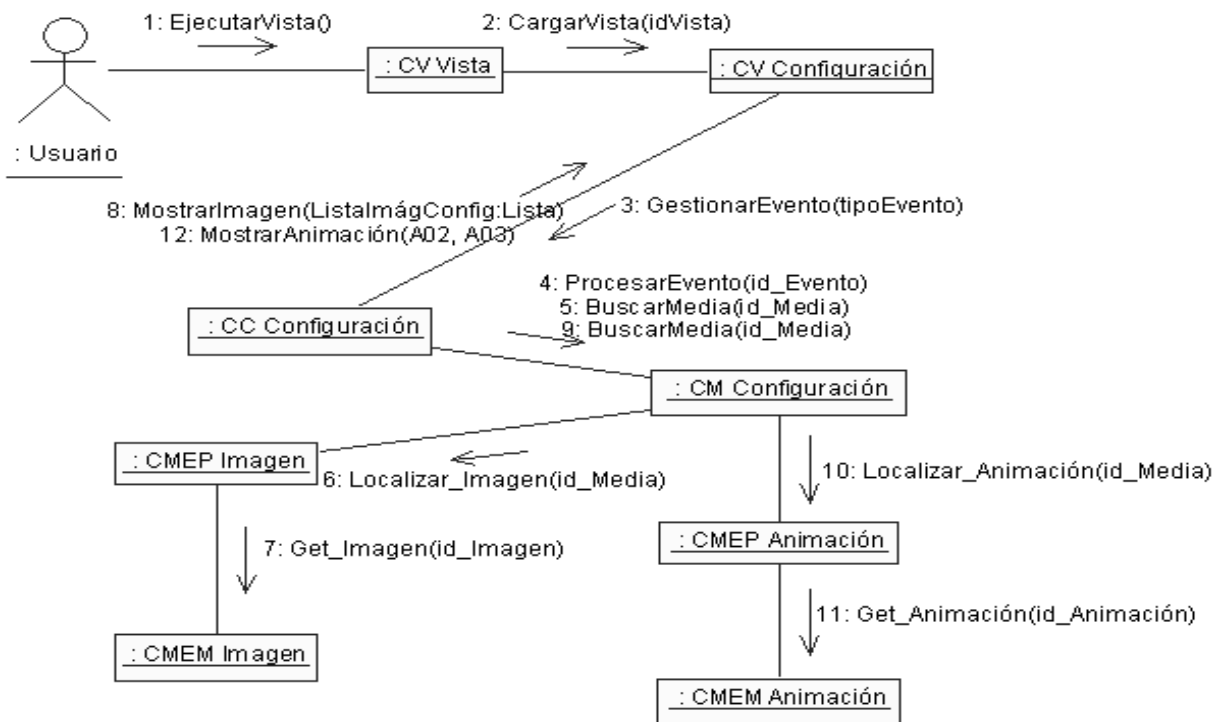


Figura 34. Diagrama de colaboración Vista Configuración

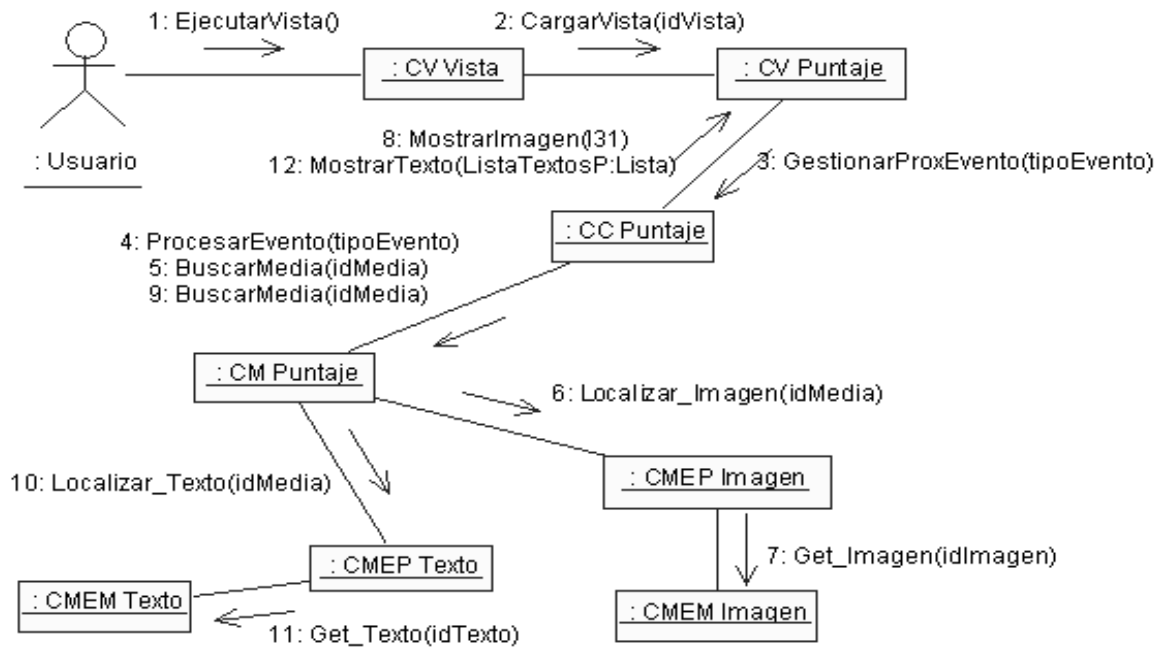


Figura 35. Diagrama de colaboración Vista Puntajes

3.2.3 Modelo de despliegue

El modelo de despliegue se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño. En este se realiza el diagrama de despliegue, perteneciente al área de Estructura lógica y a la Vista de arquitectura de ApEM-L. Este modela la arquitectura en tiempo de ejecución de un sistema, mostrando la configuración de los elementos de hardware (nodos) y cómo los elementos y artefactos del software se relacionan con ellos. El diagrama de despliegue para el software educativo “Energía para Aprender”, mostrado en la figura 16, corresponde con un procesador (ordenador o PC) que no necesita estar en red, solo una unidad lectura de CD y debe contar con los dispositivos señalados en los requerimientos no funcionales de hardware.

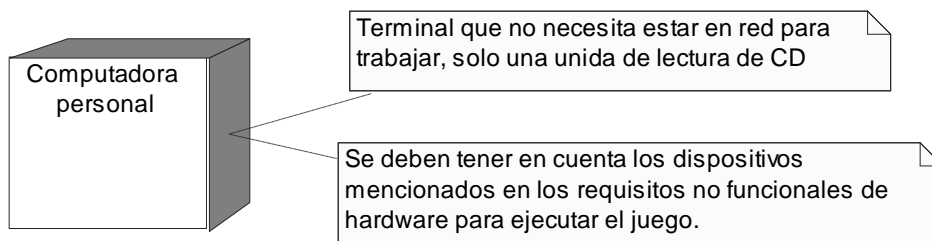


Figura 36. Diagrama de despliegue

3.4 Conclusiones del capítulo

En este capítulo se trataron de manera general los aspectos correspondientes a la construcción de la solución propuesta. Como resultado del trabajo, tomando como base el Lenguaje para la modelación de Aplicaciones Educativas Multimedia (ApEM - L), se realizaron los diagramas de clases del modelo de objetos en los cuales se presentan las medias relacionadas con cada uno de los escenarios y las clases correspondientes del lenguaje, así como las modelo, vista y controladoras, también se modelaron los diagramas de colaboración, representando los objetos y sus interacciones y quedó estructurada la arquitectura en tiempo de ejecución del sistema a través del diagrama de despliegue.

Capítulo 4

Estudio de la factibilidad

4.1 Introducción del capítulo

Para la realización de un proyecto es de suma importancia el análisis del costo y los beneficios que reportará, para obtener el tiempo de desarrollo en meses y la cantidad de personas que se necesitan para desarrollarlo, así como realizar una valoración acerca de si se puede llevar a cabo o no. En este capítulo se procede a realizar dicho análisis, con el objetivo de conocer además de las herramientas de desarrollo, la metodología utilizada, el problema que se desea resolver, las ventajas que proporciona el software, etc., otros recursos que se usan, como el tiempo de desarrollo, el recurso humano empleado y los gastos ocasionados. La intención de este capítulo es dejar demostrado la factibilidad del software de acuerdo al costo.

4.2 Método a utilizar para la estimación del costo

Debido al trabajo basado en ApEM – L usando vistas de presentación se hace necesaria la utilización de COCOMO II (Constructive Cost Model) para realizar el estudio de factibilidad, el cual se basa en el uso de ecuaciones matemáticas que permiten calcular el esfuerzo a partir de ciertas métricas de tamaño estimado, como el análisis de puntos de función y las líneas de código fuente (en inglés SLOC, Source Line Of Code). Es uno de los modelos más documentados en la actualidad y es correcto con referencia a los proyectos que lo han utilizado, aunque de ello no se debe desprender que deba ser válido siempre. Una preocupación es la adaptación de las ecuaciones exponenciales a organizaciones específicas, cosa que no parece inmediatamente fácil.

La planificación se realiza mediante puntos de función, los que permiten estimar el tamaño del software a partir de sus requerimientos, mientras que los casos de uso (los cuales no se usan en este modelado) permiten documentar los requerimientos del software. Ambos tratan de ser independientes de las tecnologías utilizadas para la implementación.

4.2.1 Salidas externas

Nombre de la salida externa	Cantidad de ficheros	Cantidad de elementos datos	Clasificación (Simple, Media y Compleja)
Mostrar presentación del producto	1	1	Simple
Mostrar la interfaz principal	1	1	Simple
Mostrar la interfaz para seleccionar un prototipo de jugador	1	1	Simple
Mostrar la interfaz para seleccionar un juego guardado	1	1	Simple
Mostrar la interfaz para la configuración	1	1	Simple
Mostrar la interfaz de los 10 mejores puntajes	1	1	Simple
Mostrar presentación del nivel	1	1	Simple
Mostrar la interfaz del primer nivel	1	1	Simple
Mostrar la interfaz que lleva al jugador hacia el segundo nivel	1	1	Simple
Mostrar la interfaz del segundo nivel	1	1	Simple
Mostrar la interfaz fuera del segundo nivel	1	1	Simple
Mostrar interfaz de cierre del nivel	1	1	Simple
Mostrar interfaz de preguntas	1	1	Simple
Mostrar interfaz comunicando que ha perdido	1	1	Simple
Mostrar interfaz de salida	1	1	Simple
Mostrar créditos	1	1	Simple

Elementos de información	Simples		Medios		Complejos		Subtotal de Puntos de función
	Cuenta	Peso	Cuenta	Peso	Cuenta	Peso	
Salidas externas	16	4	0	5	0	7	64
Subtotal (UFP)							64

Características del proyecto:

Puntos de función:

Características	Valor
Puntos de función desajustados	64
Lenguaje	C++
Instrucciones fuentes por puntos de función	64
Instrucciones fuentes	4096

Resultados:

Total de instrucciones fuentes: 4096

Miles de instrucciones fuentes (MF): 4.1 aproximadamente

4.3 Cálculo del esfuerzo, tiempo de desarrollo, cantidad de hombres y costo

Tabla 16. Multiplicadores de esfuerzo

Energía para Aprender Multiplicadores de esfuerzo	Valor	Justificación
RCPX (RELY, DATA, CLPX, DOCU)	1.18	La aplicación puede conducir fallas leves, fáciles de recuperar
RUSE	1.05	El código puede ser reutilizado para otros programas
PDIF (TIME, STOR, PVOL)	1.00	Plataforma estable que no se actualiza con frecuencia elevada
PERS (ACAP, PCAP, PCON)	1.00	El conocimiento de los programadores es normal o nominal
PREX (APEX, PLEX, LTEX)	1.22	La experiencia del personal es moderada
FCIL (TOOL, SITE)	1.00	Se usan herramientas nominales
SCED	1.00	Alta planificación

Tabla 17. Factores de escala

Energía para Aprender Factores de escala	Valor	Justificación
PREC	4.96	Procedencia: muy diferente a desarrollos previos
FLEX	4.05	Flexibilidad: ocasional. nominal
RESL	2.83	Riesgos: el plan identifica muchos riesgos críticos y establece hitos para resolverlos
TEAM	2.19	Cohesión de equipo: experiencia previa media operando como equipo
PMAT	4.68	Madurez de las capacidades: normal

Multiplicadores de esfuerzo

7

$$EM = \prod_{i=1}^7 E_{mi} = RCPX * RUSE * PDIF * PERS * PREX * FCIL * SCED = 1.5115$$

i = 1

Factores de escala

5

$$SF = \sum_{i=1}^5 SF_i = PREC + FLEX + RESL + TEAM + PMAT = 18.71$$

i = 1

Valores calibrados

$$A = 2.94; B = 0.91; C = 3.67; D = 0.24$$

$$E = B + 0.01 * SF$$

$$F = D + 0.2 * (E - B)$$

$$E = 0.91 + 0.01 * 18.71$$

$$F = 0.24 + 0.2 * (1.1 - 0.91)$$

$$E = 1.0971 \approx 1.1$$

$$F = 0.258 \approx 0.26$$

4.3.2 Beneficios intangibles

El análisis y diseño realizado y la documentación obtenida facilitan el trabajo, reduciendo el tiempo en el proceso de implementación y se logra una buena calidad del producto final.

4.4 Conclusiones del capítulo

En este capítulo se han listado los aspectos que de una u otra forma influyen en la ejecución de la aplicación con tecnología multimedia, la cual se desarrollará como parte de un proyecto de la universidad. Se muestran los costos a incurrir, el esfuerzo hombre - mes que requiere y el tiempo de desarrollo, destacando los beneficios que aporta la terminación del producto en cuestión.

Conclusiones

Como dijo una vez Leonardo Da Vinci: “Los que se enamoran de la práctica sin la teoría son como los pilotos sin timón ni brújula, que nunca podrán saber a dónde van”. Y este ha sido el propósito del presente trabajo, confeccionar un documento de apoyo a la comprensión de la situación problemática existente y su solución, a partir de la propuesta de diferentes artefactos que permitan la correcta implementación de un juego interactivo sobre temas relacionados con la Misión Revolución Energética en Venezuela.

- Se realizó un estudio sobre las tendencias, tecnologías y metodologías más usadas en la actualidad para la construcción de sistemas informáticos similares y se seleccionó como la metodología idónea para cumplimentar los procesos de análisis y diseño, al Proceso Unificado de Desarrollo (RUP), conjuntamente con ApEM – L, que surge como extensión de UML para la integración de especificaciones de sistemas con tecnología multimedia, tratados como software educativos.
- Se realizó la identificación de las funcionalidades que deberá poseer el producto, fomentando la definición de las mismas sobre una fuerte base pedagógica.
- Se construyó el modelo de dominio, lográndose una correcta comprensión del problema, así mismo para un trabajo más organizado y un mayor entendimiento de las características funcionales y visuales del software, se realizó la descripción de su contexto productivo a partir de la identificación de 7 subsistemas y 24 vistas de presentación, referenciadas en sus descripciones por los requerimientos funcionales definidos.
- Se modelaron los artefactos requeridos para guiar el proceso de implementación, exponiendo en cada caso el comportamiento y la interacción de los elementos que conforman cada entorno del juego interactivo.
- Se comprobó la factibilidad del software a partir de la realización del estudio de factibilidad.

De esta forma se puede afirmar que se alcanzó, satisfactoriamente el objetivo propuesto, creándose la modelación para la implementación de un entorno educativo e interactivo, mediante la adecuación de los contenidos y de las tareas o acciones a realizar en cada entorno.

Recomendaciones

Para finalizar este trabajo se sugieren algunas ideas:

- Aplicar el estudio investigativo realizado en la implementación del juego interactivo “Energía para Aprender”.
- Continuar con el estudio y profundización del modelado del software, así como en el Lenguaje para la Modelación de Aplicaciones Educativas (ApEM-L), con el fin de realizar la documentación y modelado de los nuevos niveles que completan el juego.
- Usar la documentación obtenida para ampliar las fronteras y realizar productos semejantes no sólo a los niños venezolanos, sino también a otros países en los que sea necesario crear y fortalecer una adecuada cultura energética.

Referencias bibliográficas

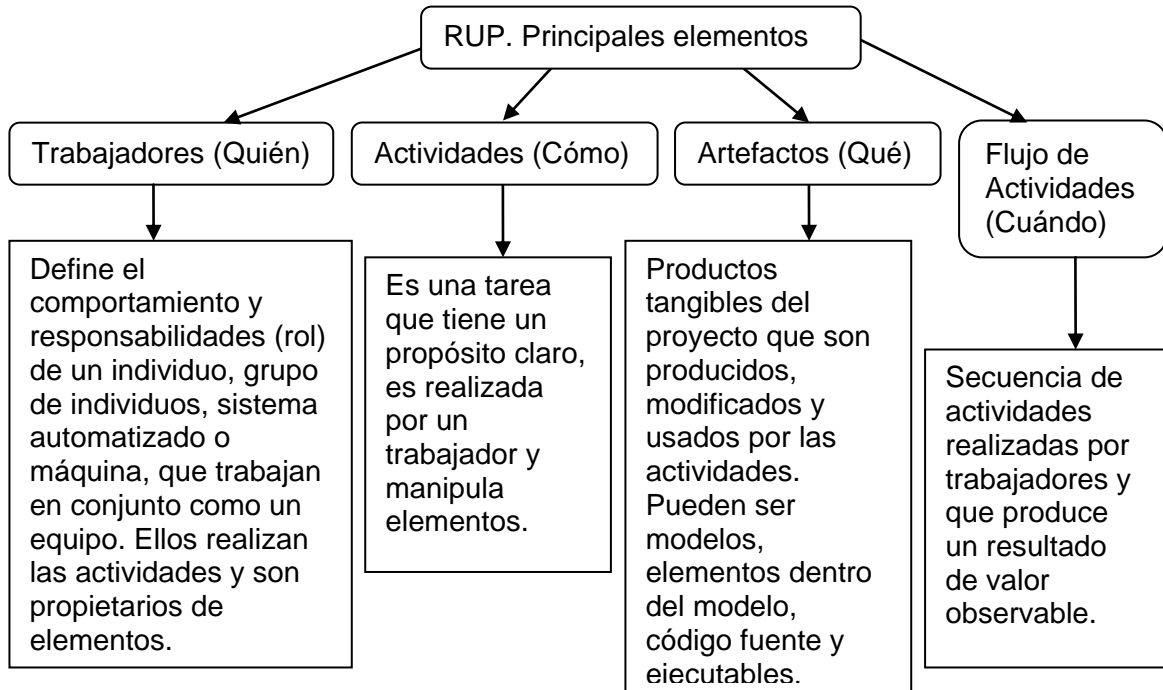
- [1] Condori, C. *Impacto de las nuevas tecnologías en educación*. 2004. [Disponible en:] http://www.ilhn.com/periodismo/archives/cat_trabajo_final.php.
- [2] Marqués, P. *El software Educativo*. Universidad Autónoma de Barcelona, 1995. [Disponible en:] http://www.karisma.org.co/documentos/softwareredp/clasif_software_educativo_de_pere.doc
- [3] Díaz, C. C. *La tecnología Multimedia: una nueva tecnología de comunicación e información. Características, concepciones y aplicaciones*, 1994. México. [Disponible en:] <http://iteso.mx/~carlosc/pagina/documentos/multidef.htm#guia>
- [4] Kimera. *La multimedia como facilitador en el proceso educativo*. 1999. [Disponible en:] <http://www.kimera.com/articulos/multimedia.html>
- [5] Dolores, M, M *La Tecnología Multimedia en el contexto del aprendizaje*. 2007. [Disponible en:] <http://www.enterate.unam.mx/Articulos/2007/mayo/art4.html>
- [6] COLABORATIVO, C. V. D. A. *¿Qué es Multimedia Interactiva?* 2006. [Disponible en:] <http://portal.educar.org/multimediam/blog/queesmultimedaiinteractiva#comment-262>
- [7] Piattini. *Análisis y Diseño Detallado de Aplicaciones Informáticas de Gestión*. 1996. Rama. Madrid.
- [8] Bork A. *El ordenador en la enseñanza. Análisis y perspectivas de futuro*. 1986.
- [9] Calero, S, M. *Una explicación de la programación extrema (XP)*. 2003. [Disponible en:] <http://www.apolosoftware.com/>
- [10] Pérez, Y, H. *Metodologías para el desarrollo de software educativo: Un estudio comparativo*. 2006.
- [11] Jiménez, S. V. *Propuesta de una estructura organizacional del proceso de producción de un software multimedia a través de RUP y una extensión de UML para hipermedia*. 2006. Universidad de las Ciencias Informáticas, Ciudad Habana.
- [12] Isakowitz, T, Stohr E, A. *RMM: a methodology of structured hypermedia design*.1995. New York. University. [Disponible en:] <http://jmis.bentley.edu/rmm/papers/rmd.pdf>
- [13] Isakowitz, T, S, Edgard, A, Balasubramanian, P. *RMM: Metodología para el Diseño Estructurado de Hipermedios*.

- [14] Mendoza, M, S. *Metodologías de desarrollo de software*. 2004. Perú. [Disponible en:] <http://www.informatizate.net-MetodologíasDeDesarrolloDeSoftware.htm>
- [15] Jacobson, I. Booch, G y Rumaugh, J. *El proceso unificado de desarrollo de software vol. 1*. Cap. 9. Editorial Félix Varela. 2004. Ciudad de la Habana.
- [16] Colectivo de autores., *Introducción a la Ingeniería de Software. El Proceso Unificado de Desarrollo (RUP)*, Dpto de Ingeniería y Gestión de Software 2007 UCI, Ciudad Habana.
- [17] Booch, G; Jacobson, I; y Rumbaugh, J. *El Lenguaje Unificado de Modelado. Manual de Referencia*. 1ra edición. Madrid: Addison-Wesley. 2000
- [18] Lorente A, E; Martínez, Y y Díaz, A. *Plantilla para el montaje dinámico de los productos de la colección multisaber*. 2006. Ciudad Habana.
- [19] Souchay, D, F. *Multimedia aprendarte, centralización de información*. 2007. [Disponible en:] <http://www.ilustrados.com/publicaciones/EEAkulVyEVyKTwmnEi.php>
- [20] Ciudad, F, R. *ApEM - L como una nueva solución a la modelación de aplicaciones educativas multimedia en la UCI*. 2007. Universidad de las Ciencias Informáticas, Ciudad Habana.
- [21] Bermúdez, H, E. *Multimedia Historia Universal Volumen I*. 2007. Universidad de las Ciencias Informáticas, Ciudad Habana.
- [22] Avalos, F, P. *Multimedia Educativa para los niños de la Enseñanza Primaria con disgrafía escolar*. 2007. Universidad de las Ciencias Informáticas, Ciudad Habana.
- [23] REGAÑA, C. B. and L. M. T. BARZABAL. *Atención a la diversidad y multimedia: el diseño de materiales curriculares un reto al alcance de todo*. 2004. [Disponible en:] <http://www.monografias.com/trabajos903/atencion-diversidad-multimedia/atencion-diversidad-multimedia2.shtml>
- [24] Polo Productivo Realidad Virtual Facultad 5 *SceneToolKit Herramienta Básica para Desarrollo de Sistemas de Realidad Virtual*. 2008. Universidad de las Ciencias Informáticas. Ciudad Habana.
- [25] Abreu, L, R, Marrero, D, L. *SoundToolKit Módulo Para el Manejo de Sonidos en Sistemas de Realidad Virtual*. 2007. Universidad de las Ciencias Informáticas, Ciudad Habana.
- [26] Frascara, J. *Definiendo audiencia*. [Disponible en:] http://www.wolkoweb.com.ar/apuntes/textos/definiendo_audiencia.rtf.
- [27] Arias, M. *La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software*. 2005. Universidad de las Ciencias Informáticas, Ciudad Habana.

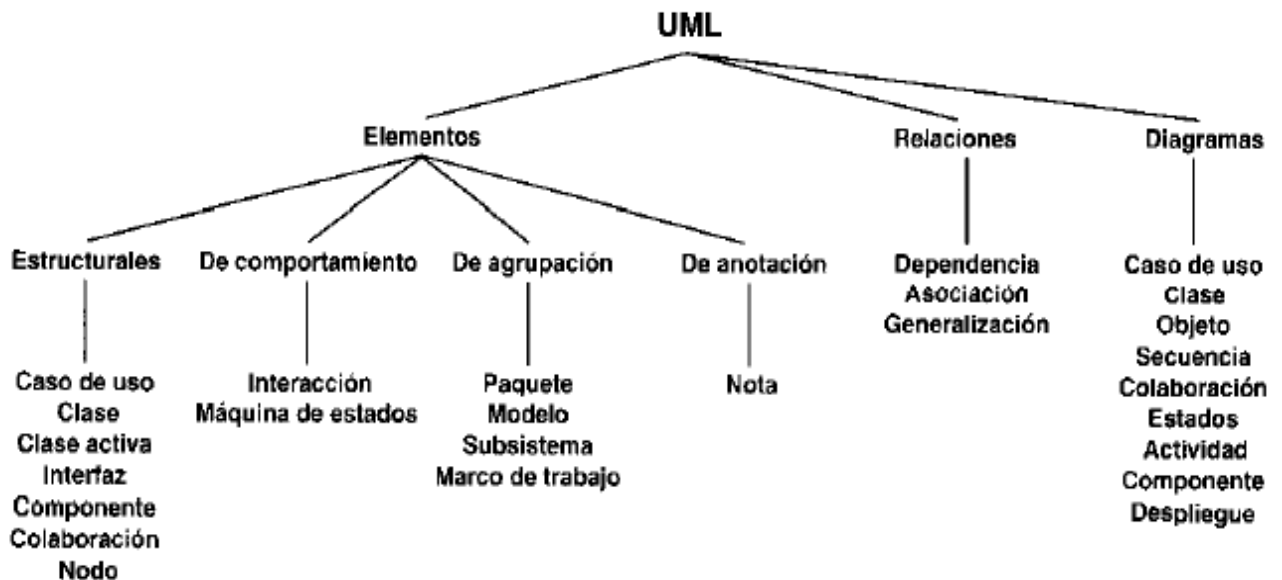
Bibliografía

1. Albalat, V. H; Peris, A; Esparza, A. *Macromedia Flash*. 1993. [Disponible en:] <http://www.lsi.uji.es/asignatura/obtener.php?letra=5&codigo=44&fichero=1115202606544>
2. ANÓNIMO. *Contenidos Educativos en Tecnología de Información y Comunicación - CETIC*, 2005. [Disponible en: <http://www.cetic.cnti.gob.ve/software2005.htm>]
3. Ciudad, F, A. *Lenguaje para la Modelación de Aplicaciones Educativas. Versión 1.0*. Presentación al Polo de Software Educativo. Abril 2008. Ciudad Habana.
4. "COCOMO II: *Una Familia de Modelos de Estimación*, [Disponible en:] <http://www ldc.usb.ve/~teruel/ci4713/clases2001/cocomo2.html>
5. Díaz Catalá, S, Solenzal Fernández, G. *Multimedia Auto-Aprende*. 2006. Universidad de las Ciencias Informáticas, Ciudad Habana.
6. Enríquez, B. A. M. *El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML*, 2005. [Disponible en:] <http://www.monografias.com/trabajos16/lenguaje-modelado-unificado/lenguaje-modelado-unificado.shtml>
7. GRAELLS, D. P. M. *Multimedia educativo: clasificación, funciones, ventajas, e inconvenientes*. 2004. [Disponible en:] <http://dewey.uab.es/pmarques/funcion.htm>
8. Hernández, R. L. y Coello, S, G. *El paradigma cuantitativo de la investigación científica*. 2002. Universidad de las Ciencias Informáticas, Ciudad Habana.
9. LARMAN, C. *UML y Patrones*.1999.
10. *Métodos de estimación. COCOMO. Fórmulas y tablas*. Universidad Carlos III de Madrid.
11. Pascual, J. *Herramientas profesionales para la creación de aplicaciones multimedia*, 1998. Barcelona, [Disponible en:] http://209.85.215.104/search?q=cache:AEm7WXJXfs4J:roble.pntic.mec.es/~sblanco1/pagina_n.htm+desventajas+de+ToolBook&hl=es&ct=clnk&cd=13&gl=cu
12. Pressman, R. S. *Ingeniería del Software: Un enfoque práctico*. 5ta ed. 2005. La Habana: Félix Varela, vol. I.
13. Schmuller, J. *Aprendiendo UML en 24 horas*. 1999.

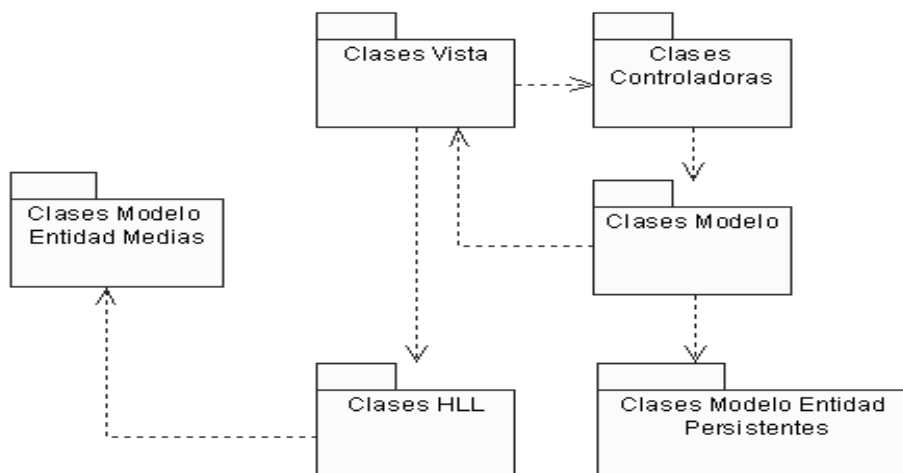
Anexos



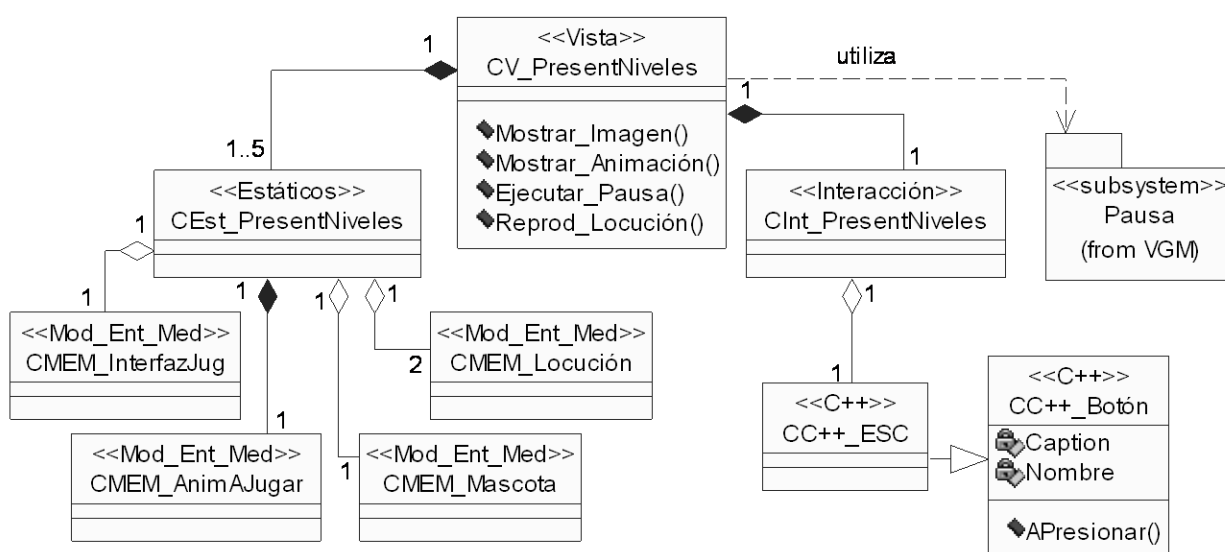
Anexo 1. Principales elementos del RUP



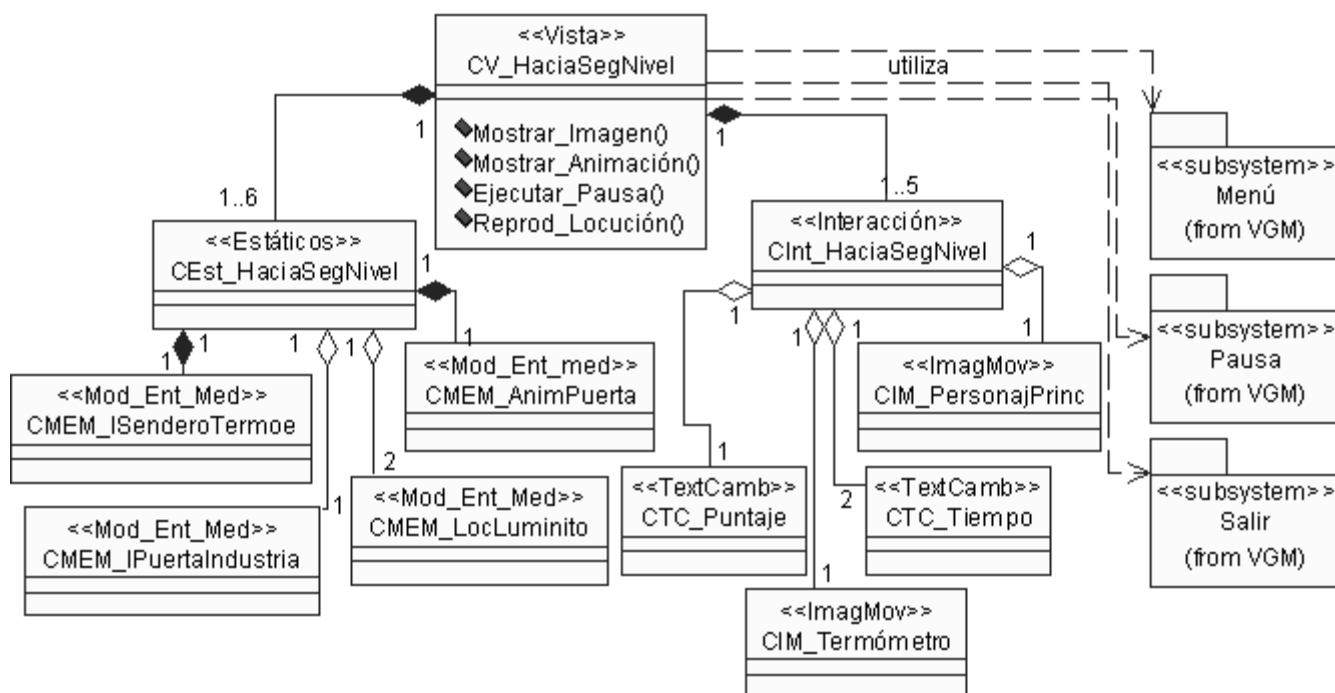
Anexo 2. Vocabulario de UML



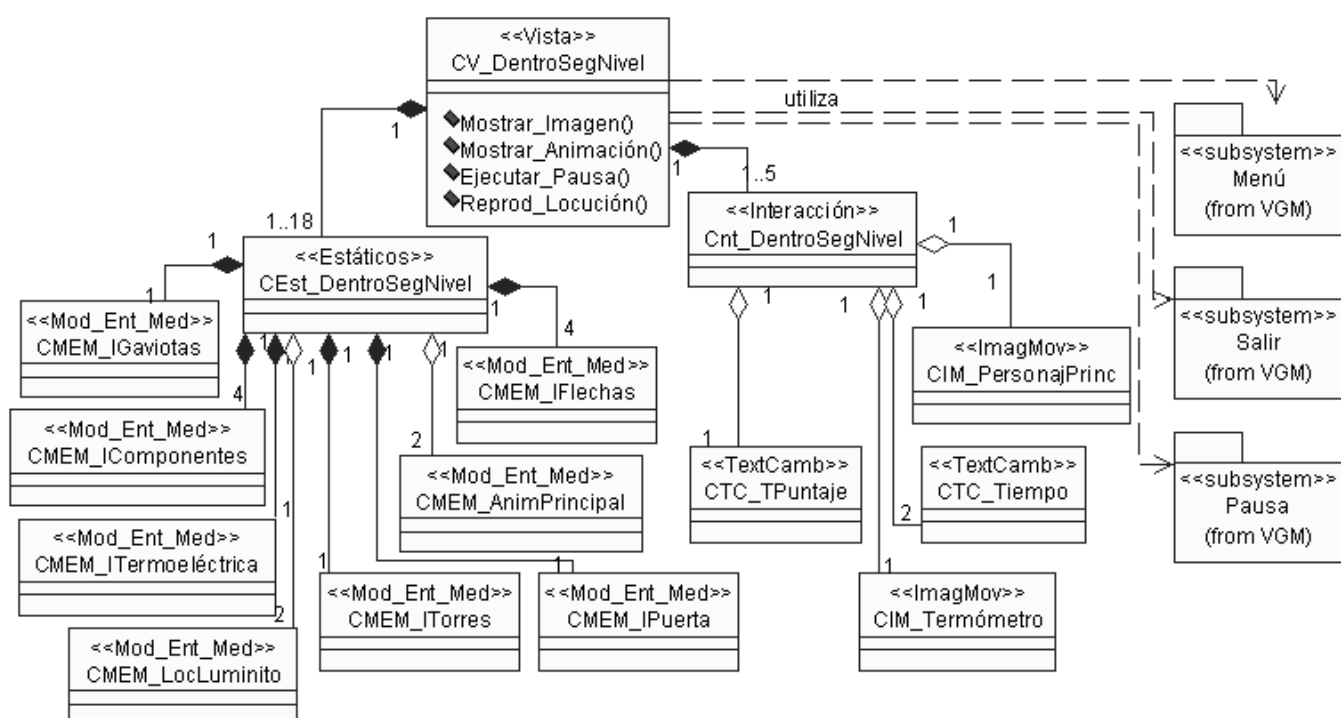
Anexo 3. Vista de Gestión de un diagrama de clases de ApEM – L distribuido por sesiones



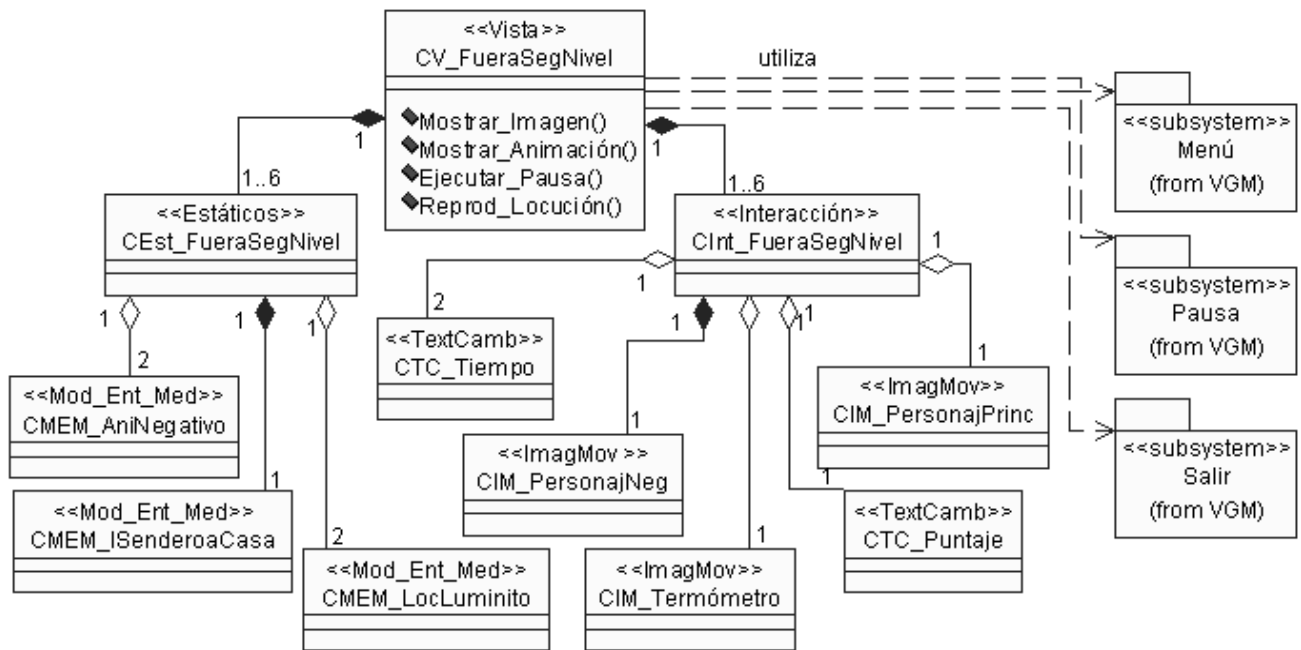
Anexo 4. Diagrama de estructura de presentación Vista PresentPrimerNivel, Vista PresentSegundoNivel y Vista Present TercerNivel (Las vistas de presentación poseen elementos estáticos e interactivos con iguales nombres y características)



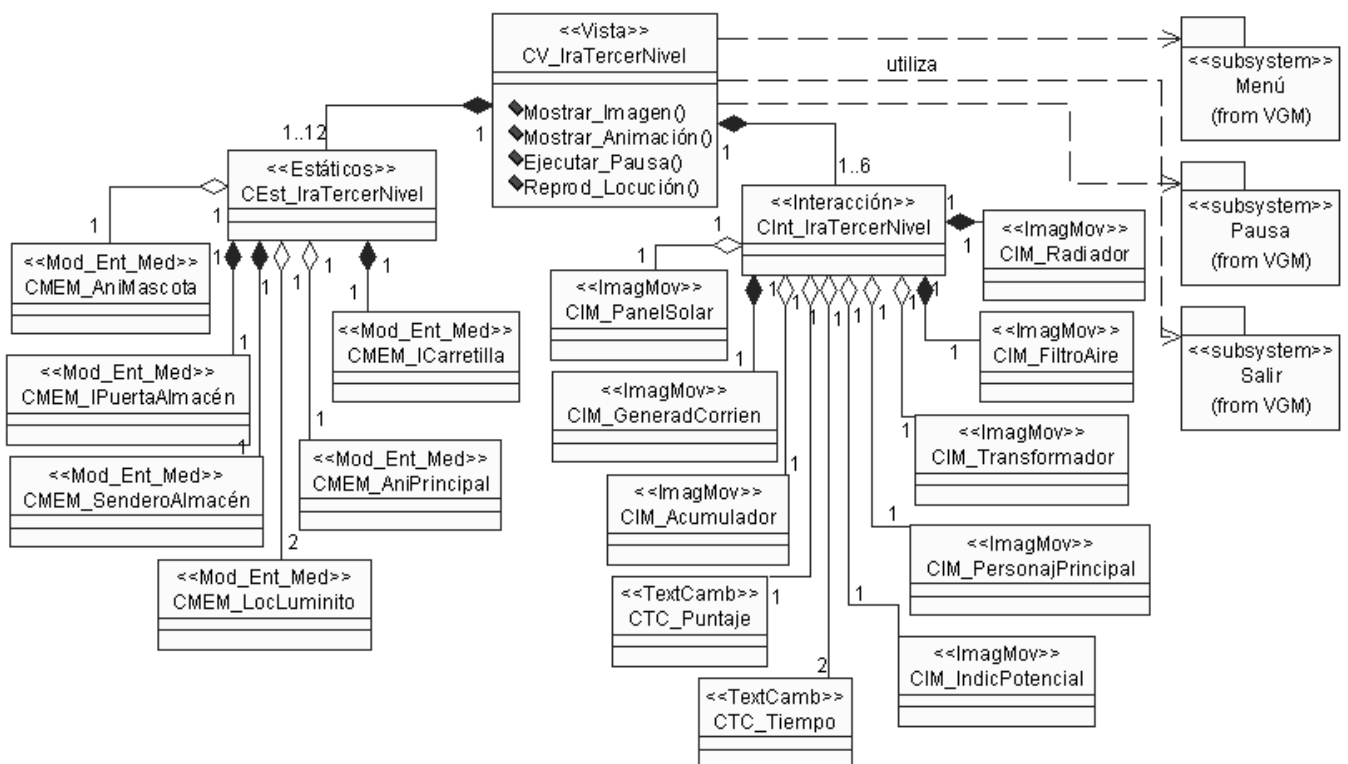
Anexo 5. Diagrama de estructura de presentación Vista HaciaSegNivel



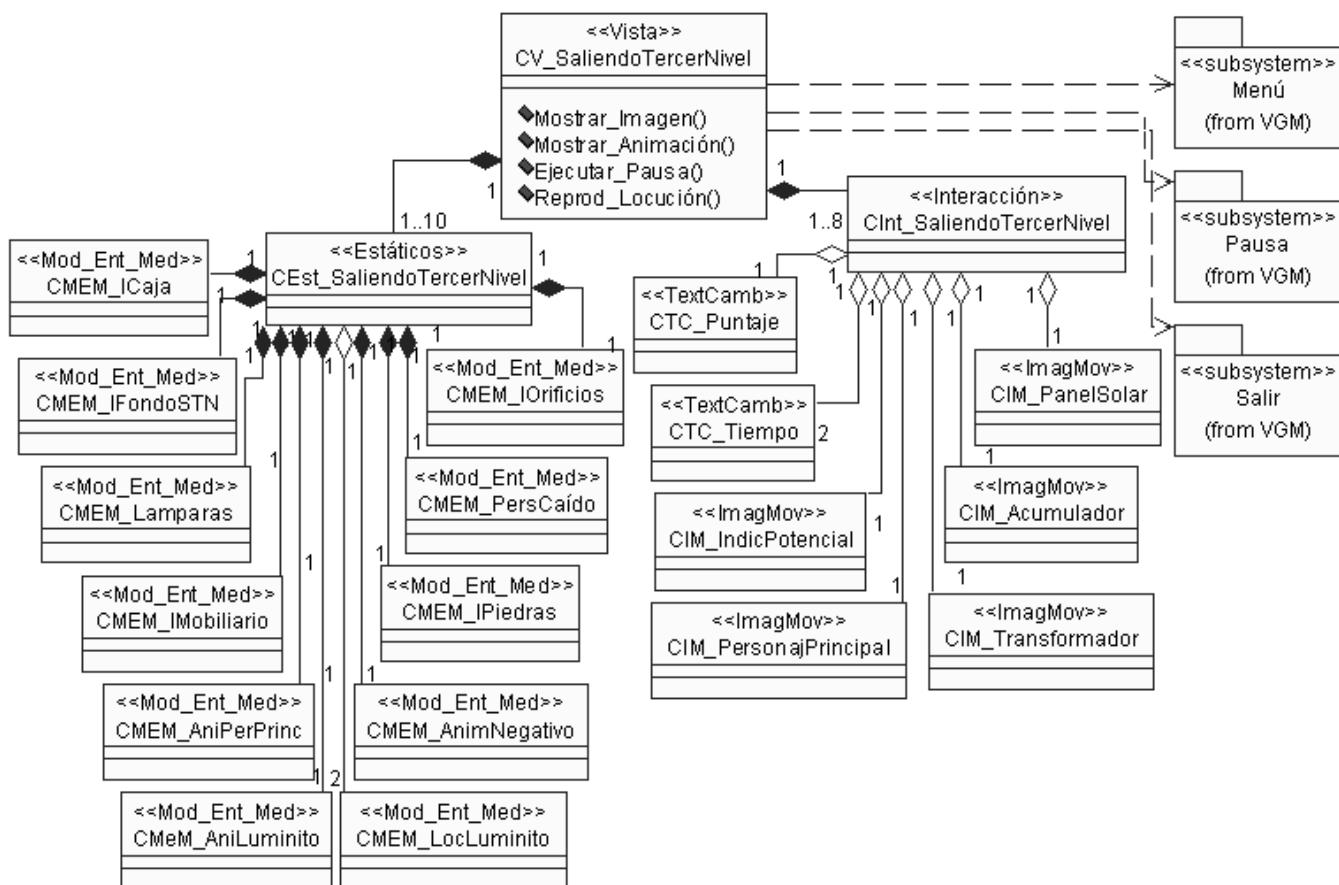
Anexo 6. Diagrama de estructura de presentación Vista DentroSegNivel



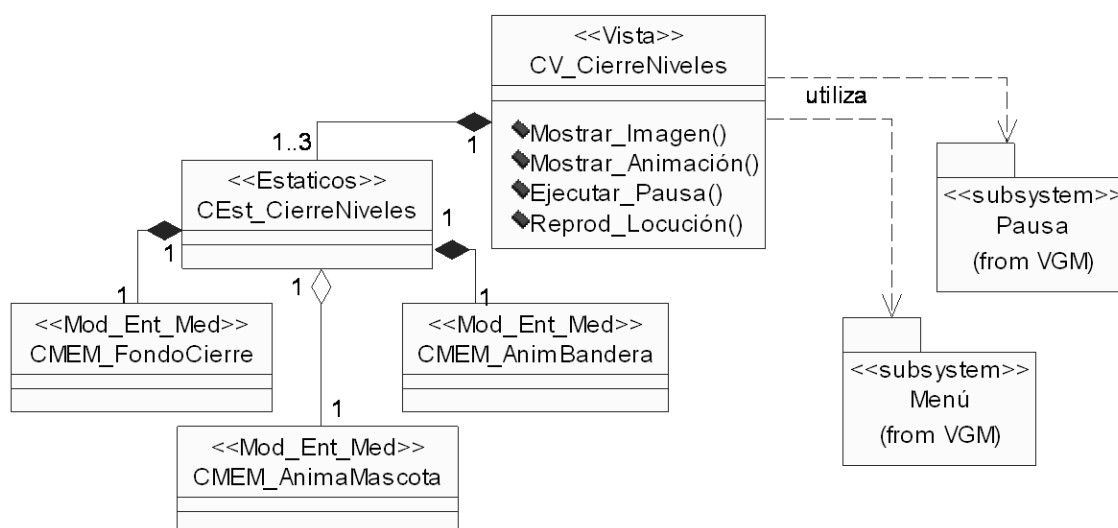
Anexo 7. Diagrama de estructura de presentación Vista FueraSegNivel



Anexo 8. Diagrama de estructura de presentación Vista IraTercerNivel



Anexo 9. Diagrama de estructura de presentación Vista SaliendoTercerNivel



Anexo 10. Diagrama de estructura de presentación Vista CierrePrimerNivel, Vista CierreSegNivel y Vista CierreTercerNivel (Poseen elementos estáticos e interactivos con iguales nombres y características)

Descripción Textual de la Vista <i>PresentSegNivel</i>			
Actores de la Vista		Usuario	
Propósito		Mostrar al usuario una presentación donde se brindan orientaciones para el nivel.	
Objetivos pedagógicos		Valorar la importancia del uso eficiente y seguro de la corriente eléctrica.	
Resumen		La vista se inicia luego del cierre del Primer Nivel y en ella "Luminito" brinda las orientaciones al jugador para cumplir las tareas requeridas para el nivel.	
Vistas asociadas		CierrePrimerNivel, HaciaSegNivel, SeleccionadoJuegoGuardado Principal, Pausa	
Referencias		R6	
Precondiciones		El usuario debe haber seleccionado un personaje para jugar en la Vista <i>SeleccionadoJugador</i>	
Poscondiciones		-	
Curso Normal de los Eventos			
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista	
<p>3. El usuario selecciona la animación A20</p> <p>5. El usuario selecciona alguna de las opciones "Esc" o "Pausa".</p>	<p>1. Se muestra de fondo la interfaz I63, sobre ella se ejecuta la animación A16 y al terminar esta se muestra la A20.</p> <p>2. Luego se ejecuta la animación A17 y al finalizar se muestra la Vista HaciaSegNivel</p> <p>4. Se interrumpe la ejecución de la animación A20, y se ejecuta la respuesta del sistema numero 2.</p> <p>6. Si seleccionó "ESC" muestra la vista Principal y da fin a la de presentación.</p> <p>7. Si seleccionó "pausa" se muestra la Vista Pausa.</p>	<p>1.1 La imagen de fondo tendrá un color más opaco que la real del juego.</p> <p>1.2 El elemento de la animación A16 tendrá un movimiento pendular en arco que recorre la pantalla de un extremo al otro del área visual.</p>	
Prioridad		Crítica	
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I63	Imagen que representa la interfaz con el fondo correspondiente al segundo nivel.	En construcción
	I64	Elemento indicador de inicio del nivel que contienen el texto: Nivel 2 "Las redes de mi país"	En construcción
	I68	Elemento que indica comenzar a jugar para el nivel 2	En construcción
Animación	A16	Movimiento pendular del elemento indicador de	En construcción

		inicio del nivel (I64) de derecha a izquierda, se detiene unos segundos en el centro y luego desaparece hacia la izquierda.	
	A17	Movimiento pendular del elemento indicador de inicio del juego (I68) de derecha a izquierda, se detiene unos segundos en el centro y luego desaparece hacia la izquierda.	En construcción
	A20	Aparece la mascota (I61) se mostrará por disolvencia en el medio de la pantalla, ofrece la L09 y al finalizar se retira de la misma forma.	En construcción

Anexo 11. Descripción textual de la Vista PresentSegNivel

Descripción Textual de la Vista <i>HaciaSegNivel</i>		
Actores de la Vista	Usuario	
Propósito	Describir las acciones a realizar por el jugador al iniciar el entorno del segundo nivel.	
Objetivos pedagógicos	-	
Resumen	Vista donde se muestran las acciones a realizar por el jugador en el primer entorno del nivel, recorre rápidamente y sin salirse del recorrido el sendero que lo lleva hasta la termoeléctrica.	
Vistas asociadas	Principal, PresentSegNivel, DentroSegNivel, Menú, Salir, Pausa, HaPerdido	
Referencias	R6, R7, R8, R9,R11	
Precondiciones	-	
Poscondiciones	-	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del Sistema	Elementos de la Vista
	1. Se muestra un fondo I65 y sobre ella ubicados en diferentes partes de la pantalla información sobre: <ul style="list-style-type: none"> • El puntaje T06 (se actualizará solo en dependencia de las acciones) • El cronómetro T07 (mostrará el tiempo mientras el juego va transcurriendo). • El cronómetro 2 T08 (mostrará el tiempo automáticamente que va transcurriendo al desarrollarse determinadas tareas para las que se destina un tiempo extra al jugador.). • El termómetro I66 para medir el CO₂ atmosférico y sobre él en lugar de una barra de mercurio estará la imagen I37. 	1.1 Se observa el botón “menú”. 1.2 El cronómetro 2 inicialmente aparecerán sin ningún cambio en su numeración hasta que se indique lo contrario.

<p>3. Movimientos del niño hacia los lados y hacia arriba para trasladarse dentro del área de juego.</p> <p>5. Se traslada desde el extremo inferior izquierdo del fondo por todo el sendero I65 sin salirse de este y luego de forma rápida va hasta la puerta de la termoeléctrica.</p> <p>7. Llega a la puerta de la industria I69 y se detiene frente a ella.</p> <p>9. Si consume dos tercios del tiempo total.</p> <p>11. Si consume el tiempo antes de terminar las acciones del nivel.</p> <p>13. El usuario selecciona alguna de las opciones “pausa”, “esc” o “menú”</p>	<p>2. En el extremo inferior izquierdo del área de juego se localiza el personaje principal I67 listo para comenzar (el escogido al inicio del juego).</p> <p>4. Utilizar el sistema WASD (W: saltar obstáculos o subir escaleras, A: izquierda D: derecha), ó las teclas direccionales (↑: saltar, ←: izquierda, →: derecha) según la configuración.</p> <p>6. Se ejecutan los movimientos simulados según las teclas utilizadas.</p> <p>8. Se ejecuta la A18 y al terminar esta se muestra la vista DentroSegNivel.</p> <p>10. Se escuchará la L05.</p> <p>12. Se escucha la L06 y se muestra la vista Ha perdido.</p> <p>14. Si seleccionó “pausa” se muestra la Vista Pausa</p> <p>15. Si seleccionó “esc” se detiene el juego y se muestra la Vista Salir</p> <p>16. Si seleccionó “menú” accede a la Vista Menú</p>		
Prioridad		Crítica	
Medias a utilizar			
Media	Nombre	Descripción	Estado
Imagen	I37	Barra de mercurio del termómetro.	En construcción
	I65	<p>Sendero que recorre el jugador hasta la entrada de la termoeléctrica, en el último plano ha de observarse la industria con dos torres, la 1 más ancha que la 2, ambas emanan humo negro. La imagen está compuesta por tres capas que se superponen una sobre otra y cada una se mueven de forma independiente.</p> <p>a) Capa1: El poblado y la industria.</p>	En construcción

		b) Capa 2: Las nubes oscuras, extremadamente contaminadas por la industria, esta se sustituirá por otra donde las nubes estarán limpias c) Capa 3: El cielo que estará azul oscuro, como contaminado, esta será sustituida por otra donde estará limpio.	
	I66	Indicador de niveles de CO ₂ con la barra dividida en tres zonas de coloración diferente.	En construcción
	I67	Los personajes de las imágenes I07- I14 estarán de costado y con morral al hombro.	Existente
	I69	Puerta de entrada a la termoeléctrica	En construcción
Animación	A18	Apertura de la puerta de entrada de la industria I69 , al mismo tiempo que el niño I67 se traslada 3 cm y queda ubicado dentro del juego luego de atravesar la puerta.	En construcción
Locución	L05	“de prisa, resta poco tiempo para culminar”	Luminito con voz de alerta
	L06	“lo sentimos, el tiempo concluyó, has perdido”	Luminito con frustración.
Texto	T06	Indicador de puntos que va obteniendo el jugador.	
	T07	00:00 (min. y seg. que se irán incrementando según el tiempo transcurrido en el juego)	
	T08	00:00 (min. y seg., se irán incrementando según el tiempo transcurrido durante la tarea para la cual se activará el cronómetro 2)	

Anexo 12. Descripción textual de la vista HaciaSegNivel

Glosario de términos

Animación: Representación sucesiva de una secuencia de imágenes que produce la sensación de estar viendo imágenes en movimiento. Para ello, a cada imagen de una animación se le modifica un pequeño detalle para mantener el movimiento tan fluido como sea posible.

Artefacto: Es una pieza tangible que crean, modifican y usan las personas encargadas de realizar determinadas actividades durante el proceso de desarrollo del software.

Capítulo: Se le llama capítulo a la primera división que tiene el contenido. El contenido está agrupado en capítulos y estos a su vez en epígrafes.

CD-ROM: Es un disco compacto pregrabado, del que se puede leer información pero no borrarla ni modificarla.

Code::Blocks: Es una herramienta de IDE usada para el desarrollo de programas en lenguaje C++. Construido para satisfacer las más exigentes necesidades de sus usuarios, entre ella correr en diversos sistemas operativos.

Diagrama: Representación gráfica, mediante la utilización de signos convencionales, del proceso que sigue la información en un programa determinado.

Dominio: Área de conocimiento o actividad caracterizada por un conjunto de conceptos y terminologías comprendidos por los practicantes de ese dominio.

Entidades: Objetos concretos o abstractos que presentan interés para el sistema.

Estereotipos: Los estereotipos son rasgos o características muy potentes. Permiten un rango de modificaciones desde ligeros cambios notacionales hasta redefiniciones del lenguaje base.

Fases: Período de tiempo entre dos hitos principales de un proceso de desarrollo

Flujo de trabajo: Realización de un caso de negocio o parte de él. Puede describirse en términos de diagramas de actividad, que incluyen a los trabajadores participantes, las actividades que realizan y los artefactos que producen.

Hardware: Componente físico tecnológico que trabaja o interactúa de algún modo con la computadora.

Hipertexto: Un hipertexto es un documento digital o no, que se puede leer de manera no secuencial. Un hipertexto tiene los siguientes elementos: secciones, enlaces o hipervínculos y anclajes.

Hipermedia: En contextos específicos, se identifica hipermedia como extensión del término Hipertexto, en el cual audio, video, texto e hipervínculos generalmente no secuenciales, se entrelazan para formar un continuo de información, que puede considerarse como virtualmente infinito desde la perspectiva de Internet.

IDE: Entorno de desarrollo integrado, es un programa compuesto por un conjunto de herramientas para un programador.

Internet: Sistema de redes de computación ligadas entre sí, con alcance mundial, que facilita servicios de comunicación de datos como registro remoto, transferencia de archivos, correo electrónico y grupos de noticias. Internet es una forma de conectar las redes de computación existentes que amplía en gran medida el alcance de cada sistema participante.

Interactivo: Un sistema es interactivo cuando permite un diálogo continuo entre el usuario y la aplicación, respondiendo esta a las órdenes de aquel.

Interfaz: Es el medio a través del cual un usuario interactúa con un sistema.

Iteraciones: Conjunto de actividades llevadas a cabo de acuerdo a un plan y a criterios de evaluación que lleva a producir una versión, ya sea interna o externa.

Lenguaje de programación: Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente.

Lingo: Es el lenguaje de programación que lleva incorporado Macromedia Director, un programa de autoría. Permite integrar con relativa facilidad texto, imágenes, sonidos y video digital, siendo una alternativa a lenguajes más tradicionales, como el C/C++, porque el desarrollo de la aplicación es mucho más rápido y flexible.

Mascota del juego: Personaje que sirve como guía en la realización de las tareas en cada uno de los niveles. Identificación personalizada que representa el ahorro y las transformaciones de la Misión. Revolución Energética.

Medias: Es el término que representa información multimedia como texto, imagen, sonido, animación, video.

Modelo: Descripción de las características estáticas, dinámicas o ambas de un tema, presentada en varias vistas.

Metodología: Se refiere a los métodos de investigación en una ciencia.

Nivel: Sección del juego en la que se realizan tareas específicas asociadas a una determinada misión (cambio de iluminación, producción de corriente eléctrica, utilización de fuentes renovables de energía, etc).

Ordenador: Es una máquina o un sistema de tratamiento de la información que realiza operaciones automáticas para las cuales ha sido previamente programada (Ejemplo: una computadora personal).

Pantalla: Es la agrupación visual de elementos de medias contenidas en una vista determinada.

Personaje negativo: Personaje que participa como antagonista del jugador durante las tareas que se realizan en cada uno de los niveles. Identificación personalizada y representativa del derroche energético y la contaminación.

Personaje principal: Es el protagonista en la ejecución de cada tarea y/o misiones que se desarrollan en el juego.

Plataforma: Es precisamente el principio, ya sea de hardware o software, sobre el cual un programa puede ejecutarse. Ejemplos típicos incluyen: arquitectura de hardware, sistema operativo, lenguajes de programación y sus librerías de tiempo de ejecución.

Proceso de Desarrollo del software: Conjunto de actividades necesarias para transformar los requisitos en un conjunto consistente de artefactos que representan un producto software y para transformar cambios en dichos requisitos en nuevas versiones del producto software.

Proyecto: Esfuerzo de desarrollo para llevar un sistema a lo largo de un ciclo de vida.

Puntaje: Puntos acumulados por el personaje principal durante el cumplimiento de cada tarea

Requisitos: Condición o capacidad que debe cumplir un sistema.

Sistema: Es un conjunto de elementos relacionados y autocontenido.

Sistema WASD: Referencia a las teclas W, A, S, D del teclado.

Software: Es un término genérico que designa al conjunto de programas de distinto tipo (sistema operativo y aplicaciones diversas) que hacen posible operar con el ordenador.

Software educativo: Programa con fines educativos.

Subsistema: Es un agrupamiento semántico útil de clases u otros subsistemas. Posee un conjunto de interfaces que ofrece a sus usuarios.

Tarea: Actividades específicas a desarrollar por el personaje principal relacionadas con cada una de las misiones y que marcan pautas para el otorgamiento de puntos al jugador.

Tecnologías de la Información y las Comunicaciones (TIC): Conjunto de avances tecnológicos que proporciona la informática, las telecomunicaciones y las tecnologías audiovisuales, que comprenden los desarrollos relacionados con los ordenadores, Internet, la telefonía, los "mas media", las aplicaciones multimedia y la realidad virtual. Estas tecnologías básicamente proporcionan información, herramientas para su proceso y canales de comunicación.

Tecnología: Conjunto de teorías y de técnicas que permiten el aprovechamiento práctico del conocimiento científico.

UML: Lenguaje estándar para el modelado de software - lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema con gran cantidad de software. Permite a los desarrolladores visualizar el producto de su trabajo en esquemas o diagramas estandarizados.