

Universidad de las Ciencias Informáticas

Facultad 8



Análisis y diseño de la multimedia informativa Barrio
Adentro

BARRIO *al alcance*
ADENTRO *de tus manos*

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN
CIENCIAS INFORMÁTICAS**

Autora: Laritza Paez Prieto

Tutor: Ing. Liván Peña Prieto

Ciudad de la Habana, Junio de 2007

Año 49 de la Revolució

PENSAMIENTO

“... lo que da al hombre el poder no es el mero conocimiento que viene del uso de los sentidos, sino, ese otro conocimiento más profundo que se llama Ciencia.”

José Martí

DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas (UCI) los derechos patrimoniales de la misma, con carácter exclusivo.

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

Laritz Paez Prieto

Liván Peña Prieto

Firma del Autor

Firma del Tutor

AGRADECIMIENTOS

Ante todo deseo expresar mi más sincero agradecimiento a la Revolución Cubana, que nos ha dado esta posibilidad.

A la Universidad de las Ciencias Informáticas, por haberme formado como profesional a la altura de nuestros tiempos.

A mis compañeros de cinco años de estudio y esfuerzo, por brindarme su amistad desinteresada, y compartir tantas cosas buenas y malas, que durarán en mi memoria para siempre.

A mis padres y familiares, por guiarme y apoyarme durante tantos años; y confiar en que podía lograrlo. Especialmente a mi mamá y a mi papá por ser los mejores padres del mundo, a mi hermano, a mis tías Taidy, Ismary y Magaly y a mi abuela Francisca por ser una madre para mí.

A todas las personas que de una forma u otra colaboraron con la realización de este trabajo. A todos, muchas gracias; sin ustedes no hubiese sido posible su terminación.

DEDICATORIA

Dedico este trabajo a mi mamá y a mi papá; a mi hermano menor Leodan; a mis abuelos Francisca y Nicolás; a mis tías queridas: Taidy, Ismary y Magaly; y a todos los que me han apoyado y han ayudado a la realización del mismo.

RESUMEN

En la actualidad las Nuevas Tecnologías Informáticas son muy usadas a nivel mundial para el desarrollo de software con el fin de resolver una problemática dada. Es por ello que el presente trabajo está enmarcado en la realización de un producto multimedia como medio informativo. La necesidad de un producto de este tipo surge debido a la carencia de información existente en el pueblo venezolano sobre unas de las misiones que se están llevando a cabo actualmente en el país, la misión Barrio Adentro, y a la carencia de un medio informático que reúna los aspectos más importantes de la misma. Con este software, se pretende hacer llegar todos, los aspectos más importantes de dicha misión de una manera amena e interesante, motivando así su consulta. Para ello se hace un análisis de algunas de las metodologías de desarrollo existente y lenguajes de modelado como son HDM, RMM, OOHDM, UML, OMMMA-L y RUP, seleccionando la más adecuada para el desarrollo de este software así como la herramienta más idónea para su elaboración. Utilizando OMMMA-L y RUP se desarrolla el levantamiento de los requerimientos funcionales y no funcionales del sistema y el análisis y diseño de la solución propuesta. Finalmente se realiza un estudio de factibilidad para determinar los beneficios tangibles e intangibles además de hacerse un análisis de los costos y beneficios que proporcionaría.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN DEL TEMA	5
1.1 Introducción	5
1.2 Estado del arte	5
1.2.1 Las Tecnologías de la Información y las Comunicación	5
1.2.1.1 ¿Qué son las TIC?	6
1.2.2 ¿Qué es Multimedia?	6
1.2.2.1 Multimedia Interactiva	7
1.2.2.2 ¿Dónde se aplica multimedia?	7
1.2.2.3 Principios base de la concepción y desarrollo de la multimedia	9
1.2.2.4 Principales ventajas de la tecnología multimedia	9
1.2.2.5 ¿Cuándo es necesario el uso de multimedia?	10
1.2.3 ¿Qué es Hipertexto?	10
1.2.3.1 Características básicas de un hipertexto	11
1.2.4 ¿Qué es Hipermedia?	12
1.3 Análisis de otras soluciones existentes	13
1.4 Descripción del objeto de estudio	13
1.4.1 Descripción general	13
1.4.2 Identificación de la audiencia	14
1.5 Conclusiones	15
CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES UTILIZADAS	16
2.1 Introducción	16
2.2 Tendencias y tecnologías actuales a considerar	16
2.2.1 Metodologías de desarrollo y Lenguajes de modelado de sistemas multimedia	16
2.2.1.1 HDM- A Model-Based Approach to Hypertext Application Design	17
2.2.1.2 RMM- Relationship Management Methodology	17
2.2.1.3 EORM- Enhanced Object Relationship Methodology	18
2.2.1.4 OOHDM- Object-Oriented Hypermedia Design Method	19
2.2.1.5 El Lenguaje de Modelado Unificado (UML)	19
2.2.1.6 El proceso Unificado de Modelado (RUP)	20
2.2.1.7 OMMMA-L	22
2.2.1.8 Metodología y Lenguaje propuesto	24
2.1.2 Herramientas de desarrollo de Sistemas multimedia	25
2.1.2.1 Macromedia Authorware	25
2.1.2.2 Toolbook	26

2.1.2.3 Flash MX 2004	26
2.1.2.3.1 ¿Qué gana el usuario con Flash?	27
2.1.2.3.2 Ventajas y desventajas de Flash MX	27
2.1.2.4 Director MX	28
2.1.2.5 Herramienta propuesta	28
2.2 Conclusiones	29
CAPÍTULO 3: DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA	30
3.1 Introducción	30
3.2 Descripción del Sistema propuesto	30
3.2.1 Descripción de la funcionalidad	31
3.2.1.1 Requerimientos Funcionales	31
3.2.1.2 Requerimientos no Funcionales	32
3.2.2 Modelo conceptual	34
3.2.2.1 Análisis de los conceptos del Dominio	34
3.2.2.2 Diagrama de clases del modelo del dominio	35
3.2.2.3 Diagrama de navegación	36
3.2.3 Modelo de Casos de uso del sistema	39
3.2.3.1 Determinación de los casos de uso	40
3.2.3.2 Diagrama de casos de uso	43
3.2.3.2.1 Descripción textual de los casos de uso Generales	45
3.2.3.2.2 Descripción textual de los casos de uso del Módulo 1	54
3.2.3.2.3 Descripción textual de los casos de uso del Módulo 2	56
3.3. Conclusiones	58
CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA	59
4.1 Introducción	59
4.2 Modelo de diseño	59
4.2.1 Diagramas de presentación	59
4.2.1.1 Diagramas de presentación del Módulo 1	60
4.2.1.1 Diagramas de presentación del Módulo 2	65
4.3 Modelo de Implementación	68
4.3.1 Modelo de componentes	68
4.4 Modelo de despliegue	75
4.5 Conclusiones	76
CAPÍTULO 5: ESTUDIO DE FACTIBILIDAD	79
5.1 Introducción	79
5.2 Planificación basada en Casos de Uso	80

5.2.1 Cálculo de puntos de casos de uso sin ajustar	80
5.2.1.1 Factor de Peso de Actores sin ajustar (UAW)	80
5.2.1.2 Factor de Peso de los Casos de Uso sin ajustar (UUCW)	82
5.2.2 Cálculo de los puntos de casos de uso ajustados	83
5.2.2.1 Factor de complejidad técnica (TCF)	83
5.2.2.2 Factor de ambiente (EF)	85
5.2.3 Estimación de esfuerzo a través de los puntos de casos de uso	87
5.3 Beneficios tangibles e intangibles	89
5.4 Análisis de costos y beneficios	89
5.6 Conclusiones	90
CONCLUSIONES	80
RECOMENDACIONES	81
REFERENCIAS BIBLIOGRÁFICAS	82
BIBLIOGRAFÍA	85
GLOSARIO DE TÉRMINOS	86

ÍNDICE DE TABLAS

Tabla 1. Formas de aplicación y usos alternativos de Multimedia (DÍAZ 1994)	_____	¡Error! Marcador no definido.
Tabla 2. Actores del sistema	_____	¡Error! Marcador no definido.
Tabla 3. Determinación del caso de uso Mostrar presentación	_____	¡Error! Marcador no definido.
Tabla 4. Determinación del caso de uso Controlar audio	_____	¡Error! Marcador no definido.
Tabla 5. Determinación del caso de uso Salir de la aplicación	_____	¡Error! Marcador no definido.
Tabla 6. Determinación del caso de uso Acceder a módulos	_____	¡Error! Marcador no definido.
Tabla 7. Determinación del caso de uso Mostrar contenido	_____	¡Error! Marcador no definido.
Tabla 8. Determinación del caso de uso Mostrar servicios	_____	¡Error! Marcador no definido.
Tabla 9. Determinación del caso de uso Jugar juego Sopa de Letras	_____	¡Error! Marcador no definido.
Tabla 10. Determinación del caso de uso Jugar Rompecabezas	_____	¡Error! Marcador no definido.
Tabla 11. Descripción del caso de uso Mostrar presentación	_____	¡Error! Marcador no definido.
Tabla 12. Descripción del caso de uso Controlar audio	_____	¡Error! Marcador no definido.
Tabla 13. Descripción del caso de uso Salir de la aplicación	_____	¡Error! Marcador no definido.
Tabla 14. Descripción del caso de uso Acceder a módulos	_____	¡Error! Marcador no definido.
Tabla 15. Descripción del caso de uso Mostrar contenido	_____	¡Error! Marcador no definido.
Tabla 16. Descripción del caso de uso Mostrar servicios	_____	¡Error! Marcador no definido.
Tabla 17. Descripción del caso de uso Jugar juego Sopa de Letras	_____	¡Error! Marcador no definido.
Tabla 18. Descripción del caso de uso Jugar Rompecabezas	_____	¡Error! Marcador no definido.
Tabla 20. Factores de Peso de los Casos de Uso sin ajustar	_____	¡Error! Marcador no definido.
Tabla 21. Factores de complejidad técnica	_____	¡Error! Marcador no definido.
Tabla 22. Factores de ambiente	_____	¡Error! Marcador no definido.
Tabla 23. Distribución del Esfuerzo por Flujo de Trabajo	_____	¡Error! Marcador no definido.

ÍNDICE DE FIGURAS

Figura 1. Multimedia _____	¡Error! Marcador no definido.
Figura 2. Esquema de Hipertexto _____	¡Error! Marcador no definido.
Figura 3. Fases e Iteraciones de la Metodología RUP _____	¡Error! Marcador no definido.
Figura 4. Modelo de Dominio _____	¡Error! Marcador no definido.
Figura 5. Diagrama de Navegación General _____	¡Error! Marcador no definido.
Figura 6. Diagrama de navegación del Módulo 1 _____	¡Error! Marcador no definido.
Figura 7. Diagrama de navegación del Módulo 2 _____	¡Error! Marcador no definido.
Figura 8. Diagrama de casos de uso Generales _____	¡Error! Marcador no definido.
Figura 9. Diagrama de casos de uso del Módulo 1 _____	¡Error! Marcador no definido.
Figura 10. Diagrama de casos de uso del Módulo 2 _____	¡Error! Marcador no definido.
Figura 11. Diagrama de presentación Entrada a Módulos _____	¡Error! Marcador no definido.
Figura 12. Diagrama de presentación Entrada a Módulo1 _____	¡Error! Marcador no definido.
Figura 13. Diagrama de presentación Antecedentes _____	¡Error! Marcador no definido.
Figura 14. Diagrama de presentación Servicios Nivel1, Nivel2 _____	¡Error! Marcador no definido.
Figura 15. Diagrama de presentación Servicios Nivel3 _____	¡Error! Marcador no definido.
Figura 16. Diagrama de presentación JuegosM1 _____	¡Error! Marcador no definido.
Figura 17. Diagrama de presentación Entrada M2 _____	¡Error! Marcador no definido.
Figura 18. Diagrama de presentación Requisitos _____	¡Error! Marcador no definido.
Figura 19. Diagrama de presentación JuegosM2 _____	¡Error! Marcador no definido.
Figura 20. Diagrama de componentes General _____	¡Error! Marcador no definido.
Figura 21. Diagrama de componentes Módulo 1 _____	¡Error! Marcador no definido.
Figura 22. Diagrama de componentes Juego M1 _____	¡Error! Marcador no definido.
Figura 23. Diagrama de componentes Módulo 2 _____	¡Error! Marcador no definido.
Figura 24. Diagrama de componentes Juego M2 _____	¡Error! Marcador no definido.
Figura 25. Diagrama de componentes Salida _____	¡Error! Marcador no definido.
Figura 26. Modelo de despliegue _____	¡Error! Marcador no definido.

INTRODUCCIÓN

En la actualidad el uso apropiado de las Tecnologías de la Informática y la Telecomunicaciones (TIC) ha sido un factor determinante en el fortalecimiento para enfrentar los problemas que surgen de la complejidad del mundo actual. Las TIC se han convertido en instrumentos transversales a la sociedad, penetran y se integran prácticamente en todas las actividades y sectores de la misma, y no es posible prescindir de ellas, puesto que pautan el tiempo, la manera de trabajar, aprender, comunicarse y en consecuencia la manera de gobernar.

Venezuela, no está exenta a este fenómeno de carácter universal y por ello en el año 2000 crea un Plan de Tecnologías de Información y Comunicación (Plan TIC), promovido por el Ministerio de Ciencia y Tecnología (MCT), que incorpora en sus Lineamientos, Políticas y Estrategias, el conjunto de elementos que facilitan el despliegue de una Plataforma Nacional de Tecnologías de Información (PNTI), democratizando el acceso a la información y el conocimiento, con el apoyo de las tecnologías de comunicación y las posibilidades que brinda la Internet.

Como parte de esta estrategia se establece un Convenio Integral entre Cuba y Venezuela, en el marco del cual, el Centro Nacional de Tecnologías de la Información (CNTI), propicia el avance de las TIC en los procesos educativos en Venezuela mediante el fomento, seguimiento y control de los proyectos educativos, dirigidos a la producción de contenidos digitales, metodologías, herramientas, capacitación, innovación e investigación en TIC, en los ámbitos de Educación, Salud y Ambiente.

En la Universidad de las Ciencias Informáticas (UCI), creada en Cuba en el año 2002 al calor de la Batalla de Ideas y como parte del soporte para la informatización del país, la producción de software y los servicios industriales, se está produciendo el producto Barrio Adentro, un software informativo que tiene como objetivo principal promover y divulgar de forma masiva y concreta información sobre una de las misiones que se están llevando a cabo actualmente en Venezuela como parte de un programa revolucionario, encaminado a saldar la deuda social contraída históricamente con la población excluida de este país, además de facilitar el acceso universal a la información y conocimientos en la sociedad venezolana.

La misión Barrio Adentro es una propuesta que rescata la filosofía y los objetivos de la Atención Primaria de Salud (APS) en Venezuela, creada como política de Estado, bajo la administración del presidente Hugo

Chávez con ayuda del gobierno de Cuba, para dar respuesta a la necesidad de acceso a los servicios de salud.

Esta Misión se divide en tres fases (ANÓNIMO 2005):

- Una **primera fase** de experimentación que consistió en medir el impacto y el apoyo de las comunidades, la constitución de los primeros Comités de Salud, la adaptación de los médicos y la evaluación de las enfermedades más frecuentes en cada localidad.
- Una **segunda fase** que consistió en la expansión del Plan Barrio Adentro hacia diversos estados del territorio nacional.
- Y una **tercera fase**, que consistió en la extensión masiva del Plan hasta alcanzar todos los estados y el distrito capital, llegando a un total de 10.179 médicos en todo el territorio nacional, cada uno de los cuales realiza 26 actividades médicas diarias que incluyen: consultas, educación para la salud, actividades de preparación de líderes comunitarios, promoción de salud, visitas directas a pacientes enfermos, y todo tipo de actividades relacionadas con la salud integral de las 250 familias que están a su cargo.

El Software informativo " *Barrio Adentro al alcance de tus manos* " surge como una necesidad por parte del gobierno venezolano debido a la poca información existente en el país sobre la misión con este nombre y el alcance de la misma. Con su creación se pretende hacer llegar a todos los aspectos más importantes de la misión, como son: Objetivos, Meta, Servicios que ofrece en cada uno de sus niveles, así como los Requisitos mínimos que se deben cumplir para la certificación de una edificación o terreno en un barrio, residencia o urbanización, entre otros aspectos.

Debido a todo lo antes planteado se evidencia la necesidad de materializar este software como medio de divulgación de información, lo cual se hará a través de una multimedia Informativa.

Este trabajo surge para dar solución a toda esta problemática, de aquí que nuestro **problema científico** consista en: ¿Cómo mitigar el desconocimiento del pueblo venezolano sobre la misión Barrio Adentro mediante el análisis y diseño de una multimedia informativa?

Los **aportes prácticos esperados** de nuestro trabajo son:

- La obtención de una correcta documentación del producto.

- La garantía de un mantenimiento futuro de nuestro sistema.
- El diseño de una instructiva aplicación Multimedia que reúna los aspectos más importantes de la misión Barrio Adentro.
- La socialización y expansión de la información acerca de la misión Barrio Adentro de una forma más dinámica y emotiva para mitigar así el desconocimiento de la población venezolana sobre la misma.

El **objeto de estudio** consiste en la aplicación de las Tecnologías de la Información y las Comunicaciones (TIC) al proceso de difusión de la información y de aquí se deriva que el **campo de acción** sea las aplicaciones multimedia para la divulgación de información.

Como **Hipótesis** se parte de la idea de que si se hace un buen análisis y diseño del software multimedia, será posible una correcta implementación del mismo, esperando poder mitigar la desinformación existente en el pueblo Venezolano sobre la misión Barrio Adentro.

Para dar solución a toda esta problemática planteada se ha definido como **objetivo de la investigación** el análisis y diseño de la multimedia informativa Barrio Adentro para un posterior desarrollo de la misma y para poder dar cumplimiento a esto y resolver la situación problemática planteada, se proponen las siguientes **tareas**:

- Hacer un estudio sobre el estado del arte del software multimedia a nivel mundial.
- Hacer un estudio de las metodologías y herramientas que se usan para el desarrollo de aplicaciones multimedia y seleccionar la más idónea.
- Analizar y definir las necesidades de funcionamiento de la futura aplicación teniendo en cuenta los requisitos del cliente.
- Diseñar el sistema a desarrollar.
- Hacer un estudio de factibilidad para determinar si el proyecto es factible o no, teniendo en cuenta los costos y beneficios del mismo.

Este trabajo estructura su contenido en 5 capítulos de los cuales se dará una breve explicación a continuación:

Capítulo 1: Se hace un breve estudio del estado del arte en el que se describen los conceptos de multimedia, hipermedia e hipertexto; se mencionan las aplicaciones multimedia y las ventajas que tienen su uso en los distintos sectores de la sociedad, además de hacerse una breve descripción del objeto de estudio teniendo en cuenta si es educativa, demostrativa o informativa y un análisis de otras posibles soluciones.

Capítulo 2: Trata sobre las tendencias y tecnologías actuales. Se analizan un grupo de metodologías de desarrollo y herramientas para de aquí proponer las que se emplearan en el desarrollo del presente trabajo.

Capítulo 3: Se hace una descripción del sistema propuesto según la metodología seleccionada, se describe su funcionalidad sacando requisitos funcionales y no funcionales y se realiza el modelo conceptual (modelos de dominio, conceptos, diagrama de navegación) y de casos de usos del sistema.

Capítulo 4: Se realiza, según la metodología utilizada, una descripción de los elementos del producto, especificando los pasos y/o estrategias para su construcción y prueba.

Capítulo 5: Se hace un estudio de factibilidad sobre el sistema, obteniendo los beneficios tangibles e intangibles y analizando los costos del desarrollo de esta propuesta.



CAPÍTULO FUNDAMENTACIÓN DEL TEMA

1.1 Introducción

En este capítulo se abordan los aspectos más generales relacionados con el tema de la multimedia, se describen conceptos tales como multimedia, hipertexto e hipermedia, necesarios para una mayor comprensión de este trabajo. Se hace un análisis sobre las aplicaciones multimedia y las ventajas que tienen su uso en los distintos sectores de la sociedad actual, además de una breve descripción del objeto de estudio teniendo en cuenta si es educativa, demostrativa o informativa y un análisis de otras posibles soluciones.

1.2 Estado del arte

1.2.1 Las Tecnologías de la Información y las Comunicación

El desarrollo de la tecnología de la información y de las comunicaciones ha sido responsable de una buena parte de los cambios sociales y productivos en el mundo en las últimas décadas. La información a la que accede el ser humano puede clasificarse de acuerdo con las fuentes de las que proviene y sus formas de almacenamiento y difusión en información recibida por vía genética, información procesada por vía cerebral, información cultural. Esta última involucra el conjunto de saberes que posee una sociedad en un determinado momento histórico y depende en gran parte de la organización y modos de producción, almacenamiento y difusión de la información a través de los sistemas educativos, de comunicaciones, informáticos y científico- tecnológicos de un país, para asegurar a sus habitantes el acceso al conocimiento (IZQUIERDO and RAMOS 2007).

1.2.1.1 ¿Qué son las TIC?

Se denominan Tecnologías de la Información y las Comunicación (**TIC**) al conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz, imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética. (ROSARIO 2005)

A continuación se mostrarán algunas de las características más importantes de las TIC.

- **Inmaterialidad** (Posibilidad de digitalización). Esta característica, ha venido a definir lo que se ha denominado como "realidad virtual", esto es, realidad no real. Mediante el uso de las TIC se están creando grupos de personas que interactúan según sus propios intereses, conformando comunidades o grupos virtuales.
- **Instantaneidad**. Podemos transmitir la información instantáneamente a lugares muy alejados físicamente, mediante las denominadas "autopistas de la información".
- **Aplicaciones Multimedia**. Las aplicaciones o programas multimedia han sido desarrollados como una interfaz amigable y sencilla de comunicación, para facilitar el acceso a las TIC de todos los usuarios. Una de las características más importantes de estos entornos es "*La interactividad*". Es posiblemente la característica más significativa. Otra de las características más relevantes de las aplicaciones multimedia, y que mayor incidencia tienen sobre el sistema educativo, es la posibilidad de transmitir información a partir de diferentes medios (texto, imagen, sonido, animaciones, etc.). Por primera vez, en un mismo documento se pueden transmitir informaciones multi-sensoriales, desde un modelo interactivo.

1.2.2 ¿Qué es Multimedia?

Multimedia es una tecnología digital que, a través de la computadora, integra diversos tipos de datos como: texto, gráficas, sonido, imagen fija, imagen en movimiento. Consiste en la integración de varias tecnologías de comunicación a través de la computación dando lugar a datos digitales, sonidos digitales, imagen digital, etc. (DÍAZ 1994)

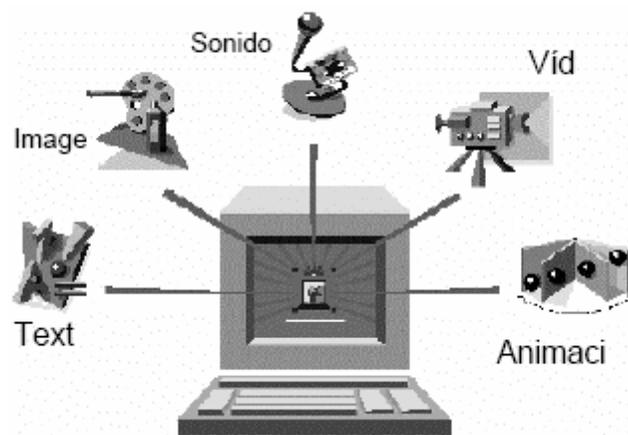


Figura 1. Multimedia (PÉREZ 2006)

Multimedia, podría ser denominada como una integración libre de tecnología que extiende y expande la forma en que interactuamos con una computadora, concepto que enriquece y amplía la interacción hombre-máquina, hoy en día lo vemos manifestado en diversas aplicaciones que incluyen enciclopedias históricas, aventuras científicas animadas y libros de cuentos y novelas interactivas (DÍAZ 1994).

1.2.2.1 Multimedia Interactiva

Es cuando se le permite al usuario final (el observador de un proyecto multimedia) controlar ciertos elementos de cuándo deben presentarse (RODRÍGUEZ).

1.2.2.2 ¿Dónde se aplica multimedia?

La multimedia es una tecnología que está encontrando aplicaciones, rápidamente, en diversos campos, por la utilidad social que se le encuentra. (DÍAZ 1994)

- **En la diversión y el entretenimiento.** Multimedia es la base de los juegos de video, pero también tiene aplicaciones en pasatiempos de tipo cultural como cuentos infantiles interactivos, exploración de museos y ciudades a manera de visitas digitales interactivas.
- **Multimedia en los negocios.** Las principales aplicaciones se dan en: la inducción, capacitación y adiestramiento de personal, la disposición rápida, accesible y procesamiento de altos volúmenes

de información, los kioscos de información, las presentaciones, intercambio y circulación de información. El trabajo en grupo o de equipo para elaborar proyectos.

- **En publicidad y marketing.** Las principales aplicaciones son: la presentación multimedia de negocios, de productos y servicios, la oferta y difusión de los productos y servicios a través de los kioscos de información.
- **En la difusión del saber y conocimiento.** La característica de la interactividad de multimedia, que permite navegar por el programa y buscar la información sin tener que recorrerlo todo, logra que la tecnología se aplique en los nuevos medios de dos modos diferentes y se use de tres formas alternativas (Véase la **Tabla 1**).
- **En la administración.** Multimedia permite tener a la vista los acostumbrados inventarios de productos, más que por columnas de números, por registros e inspecciones de cámaras de video de los estantes de almacén, realizados por el administrador de éste. Igualmente permite revisar y analizar reportes de clientes realizados por video, de manera más rápida y efectiva. La realización del trabajo en colaboración es, así mismo, posible, aún con personas que están en lugares distantes o diferentes.

Tabla 1. Formas de aplicación y usos alternativos de Multimedia (DÍAZ 1994)

MULTIMEDIA EN LOS NUEVOS MEDIOS	a) <i>Como medio de aprendizaje.</i>	<ul style="list-style-type: none"> • Por interacción, al ritmo personal, simulando situaciones reales. • Con juegos que agilizan habilidades.
	b) <i>Como medio informativo.</i>	<ul style="list-style-type: none"> • Conectado a bibliotecas electrónicas. • Accesando información, desde casa, por correo electrónico, etc.
LOS USOS DE LA	1. Medio de orientación. Presentaciones multimedia de índices de orientación en bancos y museos. Por módulos o kioscos de	

MULTIMEDIA	<p>información.</p> <p>2. Medio didáctico. Capacitación (interactividad y simulaciones). Dominio teórico previo a práctica. Posibilita conjugar actitudes y creatividad.</p> <p>3. Libro electrónico. Mediante el CD-ROM se puede tener acceso a libros y bibliotecas.</p>
-------------------	--

1.2.2.3 Principios base de la concepción y desarrollo de la multimedia

Algunos de los principios que son base de la concepción y desarrollo de la multimedia, son aspectos importantes de campos y disciplinas como: comunicación, psicología, educación, información (DÍAZ 1994).

De comunicación se emplean principios relativos al poder de atracción, significación y evocación, que tiene la imagen empleada como signo y símbolo. Se aprovechan principios relativos al poder de la imagen para impactar afectivamente. El principio de redundancia para aumentar el poder de recepción o percepción del mensaje, al emplear diversidad de canales y códigos que impactan ampliamente a los sentidos, como medios eferentes de la comunicación.

De psicología y educación se aplican los principios de la atención y el interés que tienen relación con los mecanismos del aprendizaje y comprensión.

Los desarrollos de la tecnología multimedia que tienden a una estandarización, a la compatibilidad de sistemas y al desarrollo de software potables, magnifican las ventajas de esta tecnología.

1.2.2.4 Principales ventajas de la tecnología multimedia

Las principales ventajas de la tecnología multimedia son (DÍAZ 1994):

- Posibilita la creatividad.
- Reduce el derroche de recursos técnicos, humanos y económicos (una PC con determinados programas, herramientas y periféricos equivale a pequeño estudio de producción).
- Concentra la atención, la mantiene por más tiempo y da lugar a un elevado poder de retención, potenciando la capacidad de aprendizaje.

- Es alternativa, con ventaja, a la función de los libros en el aprendizaje y la información y todo esto hace suponer que la multimedia incrementa el rendimiento del usuario final.

1.2.2.5 ¿Cuándo es necesario el uso de multimedia?

La multimedia, con sus características de interactividad, es necesaria en los casos en que se deba manejar información que esté bajo una de las siguientes tres situaciones (DÍAZ 1994) :

- Cuando hay necesidad de manejar grandes cantidades de información que se encuentra distribuida en numerosos fragmentos, registros o archivos.
- Cuando diversos fragmentos, registros o archivos pueden o deben relacionarse entre sí, de manera aleatoria y diferente para cada usuario.
- Cuando la mayoría de usuarios sólo necesitan parte de la información y tal parte puede ser diferente para cada uno de los usuarios en instantes diferentes.

1.2.3 ¿Qué es Hipertexto?

El hipertexto es una tecnología que organiza una base de información en bloques distintos de contenidos, conectados a través de una serie de enlaces cuya activación o selección provoca la recuperación de información (ANÓNIMO 2000). (Ver figura)

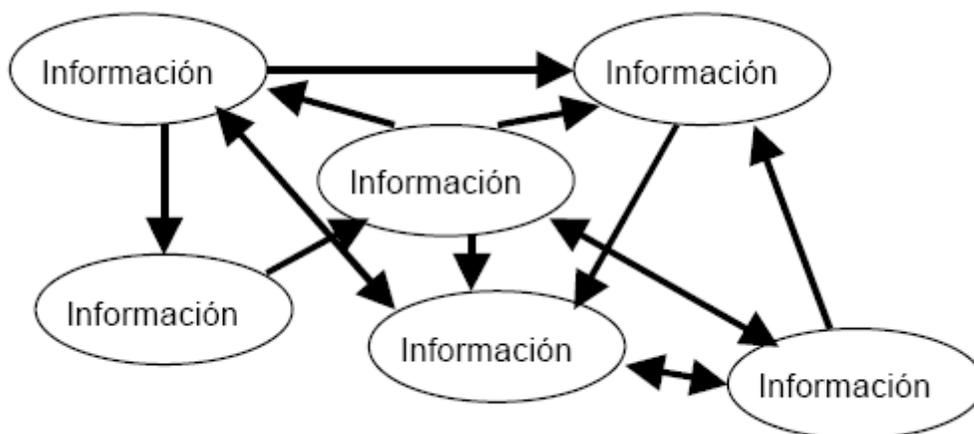


Figura 2. Esquema de Hipertexto (PÉREZ 2006)

Un hipertexto consta de los siguientes elementos: nodos o secciones, enlaces o hipervínculos y anclajes. Los nodos son las partes del hipertexto que contienen información accesible para el usuario. Los enlaces son las uniones o vínculos que se establecen entre nodos y facilitan la lectura secuencial o no secuencial por los nodos del documento. Los anclajes son los puntos de activación de los enlaces (ANÓNIMO 2007a).

Los hipertextos pueden contener otros elementos, pero los tres anteriores son los componentes mínimos. Otros elementos adicionales pueden ser los sumarios e índices. En este sentido, se habla, por ejemplo, de hipertextos de grado 1, 2, etc., según tengan la cantidad de elementos necesarios. Actualmente la mejor expresión de los hipertextos son las páginas Web navegables (ANÓNIMO 2007a).

1.2.3.1 Características básicas de un hipertexto

Un sistema hipertexto, en términos ideales, debe cumplir con las siguientes características (ANÓNIMO 2000) :

- Esta tecnología debe proveer un medio adecuado para organizar y presentar información poco o nada estructurada, no ajustada a esquemas tradicionales y rígidos como es el caso de las bases de datos. Pueden utilizarse esquemas jerárquicos para la utilización de sistemas de documentación de texto tradicionales, muy organizados o simplemente creando estructuras de redes con poco o ningún atributo de precedencia.
- Tener asociada una interfaz de usuario muy intuitiva, pues se pretende imitar el funcionamiento de la mente humana, haciendo uso de modelos cognitivos, por lo que el usuario no debería realizar grandes esfuerzos para obtener la información requerida.
- La información se encuentra distribuida y puede ser accesada en forma concurrente por varios usuarios, por lo tanto es un ambiente compartido.
- Es un ambiente colaborativo: un usuario puede crear nuevas referencias entre dos documentos cualesquiera en forma inmediata e independiente de los tipos de contenido, haciendo crecer su hiperdocumento, sin generar cambios en el hiperdocumento referenciado. Estas referencias pueden estar embebidas en el documento, de modo que aunque éste se cambiara de instalación, el enlace seguiría proporcionando acceso a la información referenciada.

- Tiene asociados varios mecanismos de recuperación y búsqueda de información a través de las navegaciones, ya sean dirigidas o no dirigidas.

Estas características hacen de este paradigma que sea utilizado en una variedad muy amplia de aplicaciones, en las cuales se tienen al menos gran cantidad de información organizada en distintos fragmentos y contextos, los cuales pueden estar relacionados entre sí, que el usuario necesita en forma discreta, y que pueda estar implantado en un ambiente electrónico o computacional (ANÓNIMO 2000).

De forma general, un sistema hipertexto gestiona fundamentalmente unidades de información enlazadas entre sí en forma de texto; aunque también puede presentar gráficos e imágenes. Sin embargo, no está preparado para presentar secuencias de medios audiovisuales que evolucionen en el tiempo, como las secuencias de imágenes o animaciones; ni los que requieren una gestión temporal para resolver problemas de sincronización y controlar eventos que genera su presentación, como puede ser el vídeo, que combina las imágenes con el sonido (PÉREZ 2006).

1.2.4 ¿Qué es Hipermedia?

La hipermedia, es la tecnología que nos permite estructurar la información de una manera no-secuencial, a través de nodos interconectados por enlaces. La información presentada en estos nodos podrá integrar diferentes medios. (texto, video, sonido, gráficos...)(MONTERO 2002). Es un sistema de bases de datos que provee al usuario una forma libre y única de acceder y explorar la información realizando saltos entre un documento y otro (ANÓNIMO 2000).

Una imagen con hipervínculos, y texto, es un ejemplo de hipermedia (ANÓNIMO 2006b).

Los sistemas hipermedia se basan en la suma de las potencialidades hipertextuales y multimediáticas aplicadas a un soporte abierto o *on line*, como es la red Internet. Actualmente, estas potencialidades convergen principalmente en la *World Wide Web*, la aplicación hipermediática de Internet. Estos sistemas permiten interconectar e integrar conjuntos de información prácticamente ilimitados, representados en múltiples materias expresivas que, además, pueden estar interconectadas. No obstante, algunos modelos de televisión por cable que combinan servicios interactivos y el acceso a Internet ya han comenzado a incorporarlo estos nuevos sistemas de interacción (PUIG).

Debido a todo esto podemos concluir que la diferencia entre un sistema hipertexto y un sistema hipermedia radica en el tipo de información contenida en sus nodos (PÉREZ 2006).

1.3 Análisis de otras soluciones existentes

En Venezuela actualmente no existe un producto en soporte multimedia que tenga centralizada toda la información referente a los aspectos mas importantes de la misión Barrio Adentro, como son sus inicios, objetivos, meta que persigue, población beneficiaria, servicios que se ofrecen a la población, así también como los requisitos mínimos que se deben cumplir para la certificación de una edificación o terreno en un barrio, residencia o urbanización. Con esta aplicación se pretende brindar al usuario la posibilidad de mantenerse informado sobre los servicios a los que tienen acceso como parte de la misión Barrio Adentro, además de contar con varios juegos que ayudan a una mayor retención de la información brindada y que la hacen mucho mas interesante y divertida.

Por otra parte en Venezuela se han desarrollado otras aplicaciones de este tipo, ya sean con fines informativos o educativos como son las multimedias de Organopónico, el Periódico Comunitario El mío, La ruta del chocolate, entre muchas otras, que se han creado con el fin de brindar información a un grupo determinado de personas sobre temas específicos de una manera mucho mas amena y concreta.

A pesar de todo esto para este tema en específico esta es la primera aplicación de su tipo, pues aunque en Internet aparece información referente a esta misión, no se había creado hasta el momento ningún medio informático en concreto que reuniera los aspectos más importantes de la misma.

1.4 Descripción del objeto de estudio

1.4.1 Descripción general

El desarrollo de la tecnología de la información y de las comunicaciones ha sido responsable de una buena parte de los cambios sociales y productivos en el mundo en las últimas décadas (ANÓNIMO 2006a).

Con el transcurrir del tiempo, el ser humano, a través de sus investigaciones y trabajo en base al enfrentamiento de sus necesidades, siendo una de ellas, la comunicación, ha generado muchos beneficios y comodidades para la sociedad, en relación al sistema de la comunicación, hoy en día hay nuevas tecnologías para una eficaz y rápida transmisión de información y comunicación (HUIDOBRO), siendo una de ellas la tecnología multimedia, tecnología esta muy utilizada actualmente a nivel mundial con diversos fines, un ejemplo de ello es el uso que se le da en el mundo empresarial para la publicidad y

el marketing, en la educación, en la diversión y el entretenimiento, en la difusión del saber y el conocimiento, entre otros posibles usos, como ya se ha mencionado anteriormente en este capítulo.

La misión Barrio Adentro es una de las misiones más importantes llevadas a cabo actualmente en Venezuela y de la cual el pueblo no tiene un total conocimiento, para socializar la información referente a ella y hacer posible que esta información llegue a todos por igual y de una forma amena e interesante, es que se determina la creación de un software multimedia, que en vista del objetivo que se quiere lograr con el, podemos decir que es totalmente informativo, pues este es precisamente su objetivo central, servir como medio de información a la población venezolana.

Para una mejor fundamentación y basándonos en la definición de Software educativo dada por el profesor Peré Marquez de la Universidad de Barcelona (GÓMEZ), podemos decir que esta multimedia no será educativa, ya que su creación no se hace con la finalidad de ser utilizada como medio didáctico, o sea, para facilitar los procesos de enseñanza - aprendizaje.

Un software educativo, ya sea una multimedia o no, es todo aquel programa, ayuda o medio que se pueda utilizar en un manejador de contenido, para facilitar el proceso de enseñanza – aprendizaje (GÓMEZ).

Teniendo en cuenta todo lo antes planteado se puede decir que el software a construir presenta carácter informativo, pues su propósito es informar al usuario sobre los aspectos generales más importantes de la misión y los servicios que se brindan en ella, a los cuales tiene derecho toda la población del país.

El software en cuestión se analizará según sus necesidades, estableciéndose así hacia quien va dirigido y el objetivo del mismo, podrá ser utilizado por cualquier persona que sepa manejar un computador y este interesado por el tema, este será su actor principal (HUNG).

1.4.2 Identificación de la audiencia

El software Barrio Adentro va dirigido a todas las personas niños, jóvenes y adultos de sexo femenino y masculino, que posean un nivel educativo básico con conocimientos previos en el manejo del computador, principalmente con habilidades en la utilización del ratón. La audiencia del software estará representada por todas aquellas personas interesadas en conocer sobre la misión Barrio Adentro y los servicios que se brindan en ella. Los usuarios de la aplicación no necesariamente tienen que tener conocimientos previos sobre el tema tratado y es precisamente con el objetivo de instruirse al respecto con el que van a acceder a la misma. La información disponible puede ser consultada con la frecuencia que el usuario estime

pertinente según su conveniencia y necesidad. Debido a lo sencilla de la aplicación no se hace necesario estructurarla por niveles en dependencia de la audiencia.

1.5 Conclusiones

En este capítulo se trataron los conceptos más importantes asociados al dominio del problema y al objeto de estudio en cuestión. Se definen los conceptos de multimedia, hipertexto e hipermedia y se plantean algunas de las aplicaciones y usos que tiene la tecnología multimedia actualmente a nivel mundial y en todas las esferas de la sociedad. Este capítulo sirve de base para el próximo capítulo en el cual se espera definir las metodologías y herramientas a usar en el desarrollo de este trabajo.

A modo de conclusión podemos decir que como características principales y distintivas de la multimedia se encuentran:

- La integración o mezcla de al menos tres de los diversos datos o información manejados por la computadora: texto, imagen, sonido, voz y video.
- La interactividad que propicia la relación del usuario con el programa y la interacción con la máquina, así como la posibilidad de colaboración o de trabajo en equipo.



CAPÍTULO TENDENCIAS Y TECNOLOGÍAS ACTUALES UTILIZADAS

2.1 Introducción

En este capítulo se hace un estudio de las tendencias y tecnologías actuales más usadas a nivel mundial. Se analizan un grupo de metodologías de desarrollo y herramientas para de aquí determinar cual es la más idónea para el desarrollo de la aplicación en cuestión.

2.2 Tendencias y tecnologías actuales a considerar

2.2.1 Metodologías de desarrollo y Lenguajes de modelado de sistemas multimedia

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software (MENÉNDEZ and ASENSIO 2005).

Una metodología puede seguir uno o varios modelos de ciclo de vida, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto pero no cómo hacerlo (ANÓNIMO). En las metodologías se van indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además qué personas deben participar en el desarrollo de las actividades y qué papel deben de tener. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla (MENÉNDEZ and ASENSIO 2005).

Las técnicas indican cómo debe ser realizada una actividad técnica determinada identificada en la metodología. Combina el empleo de unos modelos o representaciones gráficas junto con el empleo de unos procedimientos detallados. Se debe tener en consideración que una técnica determinada puede ser

utilizada en una o más actividades de la metodología de desarrollo de software. Además se debe tener mucho cuidado cuando se quiere cambiar una técnica por otra (MENÉNDEZ and ASENSIO 2005).

2.2.1.1 HDM- A Model-Based Approach to Hypertext Application Design

HDM (Hypermedia Design Model) [Garzotto 1993] es uno de los primeros métodos desarrollado para definir la estructura y la navegación propia de las aplicaciones multimedia. HDM se basa en el modelo Entidad-Relación, aunque amplía el concepto de entidad e introduce nuevos elementos, como las unidades o los enlaces (CUARESMA 2001).

En HDM se pretende especificar la aplicación mediante un modelo Entidad-Relación extendido que representa la estructura global de la aplicación sin entrar en detalles de desarrollo de los elementos unitarios (nodos de la aplicación). Sin embargo, HDM no supone una metodología para el desarrollo de aplicaciones multimedia, es simplemente una técnica de modelado, pues aunque los elementos definidos por HDM (entidades, perspectivas, enlaces, unidades, etc.) sirven para definir este tipo de aplicaciones, resultan insuficientes para guiar al diseñador en el proceso de desarrollo de las mismas (CUARESMA 2001).

Otro de los problemas que presenta HDM es que no trata los aspectos de interfaz y de múltiples medios de una manera concreta, asumiendo que se tratarán en un nivel más bajo de desarrollo (CUARESMA 2001).

En la actualidad la metodología HDM no se usa, pero ha servido como base a otras importantes metodologías como son RMM y OOHDM (CUARESMA 2001). Es además el primer intento de normalizar el desarrollo de aplicaciones multimedia aunque queda muy lejos de ser una propuesta metodológica para el desarrollo de sistemas de información global (CUARESMA 2001).

2.2.1.2 RMM- Relationship Management Methodology

RMM es propuesta en 1995 por Tomas Izsakowitz, Arnold Kamis y Marios Kounfaris. Esta metodología propone un proceso basado en 7 fases o etapas en las que el diseñador va modelando la estructura de la aplicación y las posibilidades de navegación de la misma. La propuesta está basada en el modelo Entidad-Relación y en HDM (*Hypertext Design Model*). Partiendo de ellos define un nuevo modelo el RMDM (Relationship Management Data Model), que propone un lenguaje que permitirá describir los objetos del dominio, sus interrelaciones y los mecanismos de navegación hipertexto de la aplicación.

(CUARESMA 2001) RMM es apropiada para dominios con estructuras regulares (con clases de objetos bien definidas, y con claras relaciones entre esas clases). Por ejemplo, catálogos o *front-ends* de bases de datos tradicionales. Está orientada a problemas con datos dinámicos que cambian con mucha frecuencia, más que a entornos estáticos (HIPERTEXTO.INFO 2006).

Algo ventajoso y que hace interesante a RMM es que propone un proceso estructurado y definido a seguir para el desarrollo de estas aplicaciones. En este proceso, sin embargo, se echan en falta las primeras etapas a tener en cuenta en cualquier proceso de desarrollo software, como la captura de requisitos. Además, el proceso que ofrece es demasiado abierto en sus fases como para considerarse una herramienta de desarrollo adecuada, puesto que en la única fase en la que indica una técnica es en la que se hace uso del modelo RMDM (fase 2). Las otras fases quedan abiertas a la opción del diseñador (CUARESMA 2001). Otra de las deficiencias que presenta es que en el análisis, que se realiza por medio del diagrama entidad-relación, sólo se permiten relaciones con cardinalidad $1 \rightarrow 1$ o $1 \rightarrow N$ y no es posible establecer relaciones reflexivas.

La metodología RMM exige que se obtenga el esquema E/R que gobierna la lógica de la aplicación, pero muchos de los módulos pertenecientes a la aplicación no quedan documentados bajo este esquema. Esta metodología es de poca utilidad cuando la volatilidad de la información es baja o de características estables (LEIGHTON and LEIVA 2002) que es la situación de la información involucrada en este proyecto.

2.2.1.3 EORM- Enhanced Object Relationship Methodology

EORM nace a partir de RMM y HDM pero se orienta al paradigma de la orientación a objetos, siendo la primera propuesta de este tipo. Esta metodología propone un proceso iterativo que consiste en enriquecer un modelo de objetos para representar las relaciones existentes entre objetos (enlaces). Se estructura en tres fases: análisis, diseño y construcción (CUARESMA 2001).

EORM es una metodología sencilla que garantiza todas las ventajas que la orientación a objetos ofrece y aumenta las posibilidades de reutilización en las aplicaciones, gracias a la definición del repositorio o librerías de clases enlace (CUARESMA 2001).

Además separa la navegación de lo conceptual con lo cual garantiza la reutilización y un mantenimiento más fácil. Si hay un cambio en la navegación, lo conceptual no se modifica (CUARESMA 2001).

A pesar de todo esto esta metodología presenta varias insuficiencias pues el proceso metodológico que propone resulta insuficiente en muchos casos principalmente porque solo trata de manera específica los aspectos de almacenamiento y navegación, dejando a un lado temas como la funcionalidad del sistema o los aspectos de interfaz. Además, en ningún momento comenta las técnicas a seguir para obtener los modelos que propone o los productos que se deben generar en el desarrollo. EORM también deja a un lado un aspecto muy importante en la mayoría de las aplicaciones: la captura de requisitos. No sólo no ofrece ninguna propuesta sino que no indica ninguna que se pueda usar (CUARESMA 2001).

2.2.1.4 OOHDM- Object-Oriented Hypermedia Design Method

OOHDM es una metodología de desarrollo propuesta por Rossi y Schwabe para la elaboración de aplicaciones multimedia. OOHDM está basada en HDM, en el sentido de que toma muchas de sus definiciones, sobre todo en los aspectos de navegación. Sin embargo, OOHDM supera con creces a su antecesor, ya que no es simplemente un lenguaje de modelado, sino que define unas pautas de trabajo, centrado principalmente en el diseño, para desarrollar aplicaciones multimedia de forma metodológica, además de que propone un proceso predeterminado para el que indica las actividades a realizar y los productos que se deben obtener en cada fase del desarrollo (CUARESMA 2001).

A pesar de esto OOHDM no puede considerarse una metodología en el amplio sentido, ya que, aunque se detalla el proceso a seguir en lo que sería el diseño de la aplicación, no toma parte en otras fases como pueden ser la captura de requisitos o el análisis. Otra de las deficiencias que presenta es que no ofrece ningún mecanismo para trabajar con múltiples actores (CUARESMA 2001).

OOHDM nos ofrece una serie de ideas muy adecuadas a la hora de plantear una metodología de desarrollo que tenga en cuenta la navegación y la interfaz. Sin embargo, hay que limitar el uso de esta propuesta a aplicaciones multimedia o de la Web sencillas en las que la complejidad funcional sea mínima (CUARESMA 2001).

2.2.1.5 El Lenguaje de Modelado Unificado (UML)

El Lenguaje Unificado de Modelado (UML - *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un modelo del sistema, incluyendo 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 (ANÓNIMO 2007b).

El UML oficialmente se presentó cuando Rumbaugh, Booch y Jacobson unifican sus estudios con una semántica y notación, para lograr compatibilidad en el análisis y diseño orientado a objetos, permitiendo que los proyectos se asentaran en un lenguaje de modelado maduro y que los constructores de herramientas se enfocaran en producir características más útiles. (ENRÍQUEZ)

UML es un "lenguaje" para especificar y no un método o un proceso. Se usa para definir un sistema de software; para detallar los artefactos en el sistema; para documentar y construir, es el lenguaje en el que está descrito el modelo. UML se puede usar en una gran variedad de formas para soportar una metodología de desarrollo de software (tal como el Proceso Unificado de Rational), pero no especifica en sí mismo qué metodología o proceso usar. (ANÓNIMO 2007b)

2.2.1.6 El proceso Unificado de Modelado (RUP)

Rational Unified Process (RUP) es un proceso de Ingeniería de Software planteado por Kruchten (1996) cuyo objetivo es producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos. El RUP cubre el ciclo de vida de desarrollo de software y toma en cuenta las mejores prácticas en el modelo de desarrollo de software en particular las siguientes (DÍAZ-ANTÓN):

- Desarrollo de software en forma iterativa (repite una acción).
- Manejo de requerimientos.
- Utiliza arquitectura basada en componentes.
- Modela el software visualmente (Modela con Unified Modeling Language, UML)
- Verifica la calidad del software.
- Controla los cambios.

El RUP se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un

determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso). (ANÓNIMO 2006c)

El RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al final de cada ciclo, cada ciclo se divide en fases que finalizan con un hito donde se debe tomar una decisión importante:

- **inicio:** se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- **elaboración:** se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- **construcción:** se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.
- **transición:** se implementa el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los Objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes. Este ciclo de vida que se desarrolla por cada iteración en el RUP, es llevada bajo dos disciplinas (SANCHEZ 2004):

1. Disciplina de Desarrollo

- Ingeniería de Negocios: Entendiendo las necesidades del negocio.
- Requerimientos: Trasladando las necesidades del negocio a un sistema automatizado.
- Análisis y Diseño: Trasladando los requerimientos dentro de la arquitectura de software.
- Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado esta presente.

2. Disciplina de Soporte

- Configuración y administración del cambio: Guardando todas las versiones del proyecto.

- Administrando el proyecto: Administrando horarios y recursos.
- Ambiente: Administrando el ambiente de desarrollo.
- Distribución: Hacer todo lo necesario para la salida del proyecto (Ver figura)

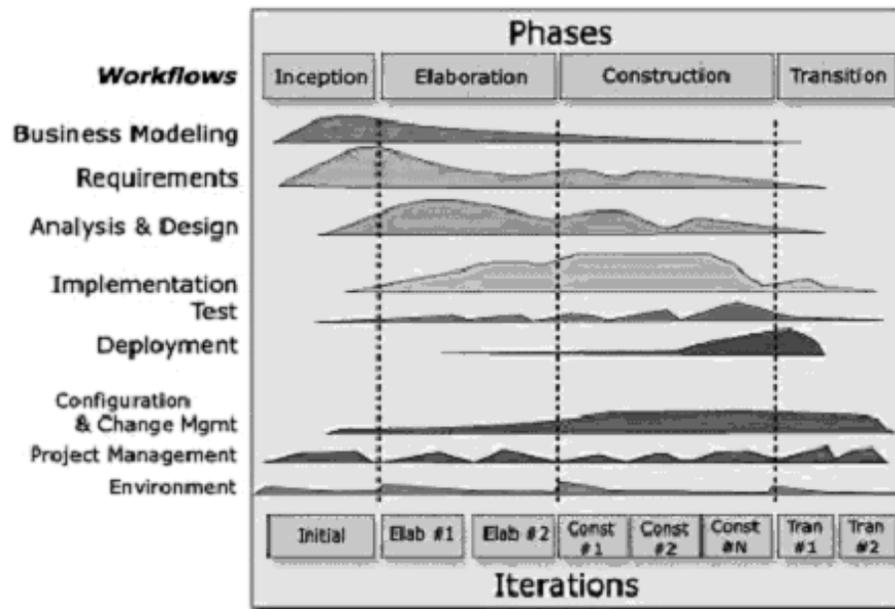


Figura 3. Fases e Iteraciones de la Metodología RUP (SANCHEZ 2004)

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.

2.2.1.7 OMMMA-L

OMMMA-L es un Lenguaje de Modelado Orientado a objetos para Aplicaciones Multimedia que se lanza como una propuesta de extensión de UML para la integración de especificaciones de sistemas multimedia basados en el paradigma orientado a objetos, y MVC (Modelo Vista Controlador) para la interfaz de usuario. MVC es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz

de usuario, y la lógica de control en tres componentes distintos, de forma que las modificaciones al componente de la vista pueden ser hechas con un mínimo impacto en el componente del modelo de datos. Esto es útil ya que los modelos típicamente tienen cierto grado de estabilidad (dependiendo de la estabilidad del dominio del problema que está siendo modelado), donde el código de la interfaz de usuario sea más robusto, debido a que el desarrollador está menos propenso a "romper" el modelo mientras trabaja de nuevo en la vista (ANÓNIMO 2007d).

OMMMA-L está sustentado en cuatro vistas fundamentales, donde cada una se asocia a un tipo de diagrama en particular. Estas vistas son (SAUER and ENGELS):

- Vista Lógica: modelada a través del Diagrama de Clases de OMMMA-L, extendido del Diagrama de Clases de UML, utilizando las mismas notaciones, pero incorporando las clases correspondientes a las medias: media continua y media discreta, generalizadas en una clase 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 de OMMMA-L, los cuales son de nueva aparición en la extensión de UML, dado que este último no contiene un diagrama apropiado para esta tarea. Estos diagramas tienen el propósito de declarar las interfaces de usuario con un conjunto de estructuras delimitadas en tamaño y área, dividiéndose en objetos de visualización (texto, gráfico, video, animación) e interacción (barras de menú, botones, campos de entrada y salida, scrolls, hipertextos con hipervínculos). Estos diagramas de presentación pueden ser divididos en capas virtuales de presentación donde en cada uno de ellas sólo se haga referencia a una clase específica de componentes (por ejemplo, una vista para los objetos de visualización y otra para los de interacción, u otro tipo de división para la representación de los intereses de los desarrolladores.
- Vista de Comportamiento temporal predefinido: modelada por el Diagrama de Secuencia de OMMMA-L, extendido a partir del diagrama de secuencia de UML. El Diagrama de secuencia modela una secuencia de una presentación predefinida dentro de una escena, donde todos los objetos dentro de un diagrama se relacionan al mismo eje del tiempo. En este diagrama se hace un refinamiento del eje del tiempo con la introducción de marcas de tiempo a través de diferentes tipos de intervalos; marcas de inicio y fin de ejecución que permite soportar su reusabilidad;

marcas de activación y desactivación de demoras en objetos de tipo media, posibilitando la modelación de las tolerancias de la variación de las restricciones de sincronización para los objetos media; activación compuesta de objetos media para la agrupación de objetos concurrentemente activos.

- Vista de Control Interactivo: modelado a través del Diagrama de Estado, extendido a partir del diagrama de estado de UML, sintácticamente igual a este último, mas con la diferencia semántica de que en el orden de unir los controles interactivos y predefinidos, no interrumpidos de los objetos, las acciones internas de estados simples tienen que llevar nombres de diagrama de secuencia en vez de diagramas de estado empotrados; queriendo esto decir que el comportamiento especificado por el diagrama de secuencia se provoca automáticamente cuando se entra al estado correspondiente donde se hace referencia.

Actualmente este lenguaje se evalúa en diferentes escenarios, como proyectos industriales para la especificación de servicios de información multimedia, y se investiga características adicionales de sincronía para su especificación en el lenguaje y la formalización de un modelo para la composición dentro y entre los diferentes diagramas de comportamiento.

2.2.1.8 Metodología y Lenguaje propuesto

Después de haber hecho un breve análisis de todas las metodologías tratadas anteriormente se decide basar la descripción de este trabajo en la metodología RUP y el lenguaje de modelado extendido OMMMA-L.

Se escoge RUP debido a que es una metodología abierta y adaptable al desarrollo de software multimedia, lo cual garantiza que se lleven a cabo sólo aquellas actividades y modelos que sean necesarios o útiles para el proyecto a desarrollar. El uso de este método de desarrollo permite elaborar un producto que garantiza la calidad del software, tanto en su proceso de desarrollo como en el producto final. Además dentro de esta metodología se incluyen cuestionarios que monitorean el proceso y el producto, así como también el manejo de los riesgos en el proceso de desarrollo de software. Esta última actividad implica el planteamiento de soluciones a los posibles problemas a través de estrategias de mitigación y los planes de contingencia. El no tomar en cuenta estos elementos tan importantes puede

llevar el desarrollo de un producto de software al fracaso o al no cumplimiento de las metas establecidas. También se toman en cuenta los requerimientos del usuario y del docente.

Para el modelado se selecciona OMMMA-L ya que a pesar de todas las facilidades y ventajas que ofrece UML y de ser uno de los lenguajes de modelados mas usados en la actualidad para el desarrollo de software no se adapta a las condiciones del actual trabajo, por lo que se opta por adoptar su versión extendida para software multimedia, OMMMA-L. Este nuevo lenguaje resulta muy fácil de aprender ya que como no es un lenguaje nuevo sino una extensión solo se necesita interpretar sus características extendidas, centrados a la lógica de funcionamiento de una multimedia, que es por lo general, sencilla. OMMMA-L soporta el modelado de los aspectos estructurales, funcionales y dinámicos de un sistema interactivo y su interfaz de usuario, se concentra en la funcionalidad desde la perspectiva del sistema de software y su sintaxis es definida explícitamente además de tener una semántica informal e intuitiva (ENGELS and NEU).

2.1.2 Herramientas de desarrollo de Sistemas multimedia

Las herramientas de desarrollo de multimedia brindan el marco esencial para organizar y editar los elementos de un proyecto multimedia, incluyendo gráficos, sonido, animaciones y secuencia de vídeo. Estas herramientas se utilizan para diseñar interactividad y las interfaces del usuario, a fin de presentar el proyecto en pantalla y combinar los diferentes elementos multimedia en un solo proyecto cohesionado (RODRÍGUEZ).

Los programas de desarrollo de multimedia brindan un ambiente integrado para unir el contenido y las funciones del proyecto. Incluyen en general las habilidades para crear, editar e importar tipos específicos de datos; incorporar datos de las secuencias de reproducción u hoja de señalizaciones, y proporcionar un método estructurado, o lenguaje, para responder a las acciones del usuario.

2.1.2.1 Macromedia Authorware

Macromedia Authorware es una herramienta única con la que puedes crear aplicaciones de carácter educativo (e-learning), totalmente dinámicas, interactivas y compatibles con los estándares. (SANTOS 2006)

El proceso de creación se lleva a cabo además en un entorno de trabajo totalmente visual, lo que hace que Macromedia Authorware sea muy fácil de usar incluso para usuarios de escasa experiencia.

Con esta herramienta se puede generar contenido de tipo educativo integrando gráficos, sonidos, animaciones textos y vídeos para crear con ellos toda una experiencia e-learning que además se puede publicar directamente en una página Web.

En la actualidad, Macromedia Authorware es una de las aplicaciones de autoría e-learning más utilizadas. Las aplicaciones de autoría como Authorware se usan principalmente para crear productos multimedia interactivos e instructivos, pero también se pueden utilizar para el desarrollo de prototipos de productos multimedia. (ANÓNIMO 2007c)

2.1.2.2 Toolbook

Toolbook es una herramienta creada para el desarrollo de aplicaciones multimedia en Windows basadas en un lenguaje orientado a objetos llamado OpenScript. 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 (page) 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. Toolbook ofrece dos niveles de trabajo diferente. El nivel autor (Author level) donde se diseña y modifica la aplicación multimedia y el nivel de lectura (Reader level) que permite visualizar el resultado de las modificaciones que se vayan realizando, además de donde se ejecuta la misma. (REGAÑA and BARZABAL 2004)

2.1.2.3 Flash MX 2004

Flash es el estándar para la creación de animaciones y gráficos vectoriales para uso en Internet. Los diseñadores de páginas Web usan Flash para crear una interfaz de navegación atractiva, compacta y con tamaño variable, también se pueden crear ilustraciones técnicas, animaciones de formato largo y cualquier otro sorprendente efecto gráfico para una página Web (SOFTONIC 2006).

Los gráficos y las animaciones en Flash MX se muestran de la manera más adecuada para la persona que los visualiza. En la animación para Web Flash MX ofrece sorprendentes efectos para disolver formas y crear transparencias. Las nuevas acciones de película permiten tener una increíble interactividad sin necesidad de usar ningún *script*.

Algunas de las características de Flash MX son: soporte de vídeo, interfaz con inspector de propiedades y panel de respuestas, carga dinámica de imágenes y sonidos, anclajes con nombre, nuevas herramientas

de diseño gráfico y mezclador de color, integración servidor-aplicación optimizada, desarrolla y reproduce contenido accesible, diseño para múltiples medios, componentes prefabricados de interfaz de usuario, etc.

2.1.2.3.1 ¿Qué gana el usuario con Flash?

Rapidez, riqueza gráfica e una interacción impensable e instantánea con cualquier elemento presente en el interfaz: formularios, gráficos, textos, imágenes, y la posibilidad de ver la totalidad del proceso en una sola pantalla, sin necesidad de avanzar y retroceder (VILLA 2003).

Con Flash, es posible una interacción cliente-servidor continua e invisible, reduciendo procesos complejos a una sola pantalla en la que errores del usuario pueden corregirse al instante sin consecuencias tan molestas como volver a la pantalla anterior o el reinicio de todo un proceso.

Técnicamente Flash soporta los estándares del mercado: integración con Microsoft .Net, J2EE - Websphere, Sun iPlanet y muchos lenguajes de script como ASP, PHP, JSP, así como WebServices, SOAP, y compatible con XML.

2.1.2.3.2 Ventajas y desventajas de Flash MX

A continuación se enunciarán algunas de las ventajas y desventajas de esta poderosa herramienta: (ALBALAT *et al.*)

Ventajas:

- Páginas Web: Consigue dar dinamismo mediante animaciones y crear aplicaciones interactivas que permiten al usuario ver la web como algo atractivo, no estático (en contraposición a la mayoría de las páginas, que están realizadas empleando el lenguaje HTML).
- Resulta fácil de aprender.
- Compatible con gran variedad de programas: permite la importación de archivos Freehand, Fireworks, Adobe Illustrator, archivos EPS y archivos PDF. Además, admite el formato DXF de AutoCAD.
- Los archivos Flash están creados para reproducirse rápidamente.
- Dispone de una poderosa capacidad de audio y video

Desventajas:

- Ocupa mas espacio un documento con una animación que simplemente con el contenido estático.
- Existen navegadores que no soportan Flash.
- Se pierde contenido al imprimir el documento.

2.1.2.4 Director MX

Macromedia Director es un programa que se usa para la creación de interactivos. Su funcionamiento e interfaz son similares a la de Flash, también tiene un lenguaje de script cuyo nombre es Lingo. Gracias a las posibilidades de este programa y este código de programación se pueden hacer interactivos bastante sofisticados, su uso es más extenso en el mundo de interactivos en CD que para Internet. La principal diferencia con Flash aparte de esa estética suavizada que a tantos nos gusta es que Director permite incorporar gráficos 3D como objetos independientes y permite programarlos para interactuar con ellos en tiempo real. (SAUTRAF 2004)

Entre las principales características de esta herramienta, se cuenta con una integración completa con la familia de productos Macromedia Flash MX, adopción de interfases eficientes para el usuario de Macromedia MX, soporte para Mac OS X, nuevas eficiencias en el flujo de trabajo y la habilidad para crear contenido accesible para que las presentaciones enriquecidas de Director puedan ser disfrutadas por personas con discapacidades. (ANÓNIMO 2002)

Además, Director MX soporta la mayoría de los formatos de video, audio, bitmap, 3D y vectores los que permiten reproducir archivos de video que duran horas o minutos de longitud y soportados con QuickTime, RealVideo y AVI. Los nuevos usuarios podrán crear contenidos a través de comportamientos del tipo "arrastrar y soltar", mientras que usuarios expertos pueden usar Lingo, el poderoso lenguaje scripting orientado a objetos de Director. (ANÓNIMO 2002)

2.1.2.5 Herramienta propuesta

Después de un breve análisis de todas las herramientas planteadas anteriormente y de algunas otras que no se mencionan aquí, se propone para la solución de este trabajo el uso de la herramienta Flash MX 2004 usando el lenguaje de programación Action Script 2.0.

Macromedia Flash MX 2004 es una potente herramienta que consolida y mezcla en profundidad tanto el diseño como la programación. Flash MX 2004 trabaja con imágenes vectoriales muy utilizadas

mundialmente para la calidad y eficiencia de las imágenes y a la vez tiene dentro de su cerebro un lenguaje de Programación Orientado a Objetos (POO), en este caso ActionScript 2.0; casi tan potente como otros lenguaje de programación como Java, C++, Pascal, Visual Basic, JavaScrip o cualquier otro que se pudiese mencionar.

2.2 Conclusiones

En este capítulo se realizó un análisis completo de las tecnologías que serán utilizadas a lo largo del desarrollo de la aplicación propuesta, y se fundamentaron las elecciones del lenguaje, la metodología y la herramienta de desarrollo a utilizar. Una vez conocido todo esto, se puede empezar a desarrollar la propuesta de la aplicación.



CAPÍTULO DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

3.1 Introducción

En este capítulo se hace una descripción del sistema propuesto según la metodología seleccionada, determinando el modelo de dominio y los principales conceptos asociados. Se hace un análisis de los requisitos funcionales y no funcionales que debe cumplir la aplicación a elaborar y de aquí se determinan los casos de usos para la construcción del modelo de sistema, además de ofrecer una descripción del mismo. En el presente capítulo se realiza todo el diseño para tener una mejor comprensión de lo que se quiere obtener con el sistema.

3.2 Descripción del Sistema propuesto

Para dar cumplimiento a los objetivos propuestos al inicio de este trabajo, y teniendo en cuenta todos los requerimientos planteados, el sistema que se propone debe tener dos módulos principales:

- Módulo 1: ¿Qué es Barrio Adentro?
- Módulo 2: ¿Cómo construir un consultorio popular?

Donde ambos módulos serán completamente informativos.

Con esta aplicación se pretende transmitir la información más relevante sobre la misión Barrio Adentro y a ella podrán acceder todos los usuarios interesados en consultar sobre el tema.

3.2.1 Descripción de la funcionalidad

3.2.1.1 Requerimientos Funcionales

Una vez conocidos los conceptos que rodean al objeto de estudio, se debe analizar: ¿Qué debe hacer el sistema para que se cumplan los objetivos planteados al inicio de este trabajo?, para ello se enumeran, a través de requerimientos funcionales, las acciones que el sistema deberá ser capaz de realizar. Dentro de ellos se incluyen las acciones que podrán ser ejecutadas por el usuario, las acciones ocultas que debe realizar el sistema, y las condiciones extremas a determinar por el sistema. De acuerdo con los objetivos planteados el sistema debe ser capaz de:

Requisitos Generales

Referencia	Función
R1	Mostrar presentación general del producto.
R2	Controlar audio de fondo.
R3	Salir de la aplicación. <ul style="list-style-type: none"> • Mostrar créditos antes de cerrar la aplicación.
R4	Acceder a Módulos.
R5	Mostrar contenido. <ul style="list-style-type: none"> R5. 1. Mostrar ¿Qué es Barrio Adentro? R5. 2. Mostrar los Objetivos de Barrio Adentro. R5. 3. Mostrar los Antecedentes de Barrio Adentro. R5. 4. Mostrar la Población beneficiaria de Barrio Adentro. R5. 5. Mostrar la Meta de Barrio Adentro.

	<p>R5.6. Mostrar Servicios que ofrece Barrio Adentro.</p> <p>R5.6.1. Mostrar los Servicios que ofrece Barrio Adentro Nivel 1.</p> <p>R5.6.2. Mostrar los Servicios que ofrece Barrio Adentro Nivel 2.</p> <p>R5.6.3. Mostrar los Servicios que ofrece Barrio Adentro Nivel 3.</p> <p>R5.7. Mostrar los Requisitos mínimos para la certificación de una edificación.</p> <p>R5.8. Mostrar los Requisitos mínimos para la certificación de un terreno.</p>
--	--

Requisitos Módulo 1

Referencia	Función
R6	Jugar juego Sopa de Letras.

Requisitos Módulo 2

Referencia	Función
R7	Jugar juego Rompecabezas.

3.2.1.2 Requerimientos no Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

Apariencia o interfaz externa: La interfaz externa deberá ser lo más atractiva y clara posible para el usuario. Su funcionamiento debe ser fácil de comprender, permitiendo la utilización del sistema sin mucho entrenamiento.

Navegación: Desde cualquier pantalla se podrá acceder a cualquier módulo. Se podrá abandonar el programa desde cualquier pantalla, posterior a una confirmación.

Usabilidad: El sistema debe permitir el acceso a los usuarios y lograr una interfaz y navegación asequibles y funcionales tanto para usuarios expertos como para los que no tienen conocimientos profundos de informática, de forma que los usuarios que interactúen con la aplicación no presenten dificultades en la asimilación de la herramienta.

Rendimiento: En cuanto al tratamiento de la información el sistema debe garantizar la mayor eficiencia posible logrando que la velocidad de procesamiento sea tal alta como se pueda asegurando un tiempo óptimo de respuesta para evitar que la acción del usuario se torne agobiante y aburrida.

Soporte.

- PC con sistema operativo Windows 9x, Me, NT, XP, Mac OS 9 o superior, Linux.
- Lector de CD con tasa de transferencia de 24X (3.600KB/s) o superior.
- Tarjeta de Sonido de 16 bits.
- Bocinas o audífonos.

Portabilidad: El producto se almacenara en un CD.

Software.

- Rational Rose Enterprise Edition para la modelación del sistema.
- Para la construcción/modificación/visualización de imagines Microsoft Paint, Adobe Photoshop 8.0.
- Macromedia Flash MX 2004 para la creación de animaciones y videos.

Hardware: Para la utilización de la aplicación por los usuarios, estos requieren de una computadora con las siguientes características:

- Microprocesador: 486 a 66Mhz 42
- Memoria RAM: 32Mb
- Disco Duro: 2Gb
- Periféricos: Teclado, tarjeta de sonido, ratón, audífonos o bocinas.

3.2.2 Modelo conceptual

El modelo conceptual representa la estructura estática del sistema. Permite modelar y describir la información que maneja el sistema, además de su estructura. De manera clásica se ha denominado modelo estático y, aunque el concepto es el mismo, en los entornos de la ingeniería Web y los sistemas navegacionales se ha extendido más la terminología de modelo conceptual (CUARESMA 2004).

3.2.2.1 Análisis de los conceptos del Dominio

Teniendo en cuenta que en nuestro proyecto no existe un negocio definido a no ser el propósito de llevar a todos la información deseada sobre la misión Barrio Adentro con el objetivo de satisfacer la carencia de información existente y de hacerla asequible a todos, es que se determina realizar el modelo de dominio, ya que permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio de la aplicación en desarrollo. Esto ayuda a los usuarios, clientes y desarrolladores e interesados; a utilizar un vocabulario común para poder entender el contexto en que se emplaza la aplicación. Para capturar correctamente los requisitos y poder construir una aplicación adecuada se necesita tener un firme conocimiento del funcionamiento del objeto de estudio. Este modelo va a contribuir posteriormente a identificar algunas clases que se utilizarán en el sistema. Primeramente hay que identificar todos los conceptos que se utilizarán en el diagrama, mediante un glosario de términos sobre los nombres:

- Se considera como **Barrio Adentro** la misión que se está llevando a cabo actualmente en Venezuela como parte del programa social desarrollado por el presidente Hugo Chávez. Es la prestación de servicios con calidad, dentro y fuera del Consultorio Popular, para y con la comunidad, vinculándose con el resto de los programas sociales.
- Se considera como **Antecedentes** a todos los acontecimientos que dieron origen a la misión.

- Se considera como **Objetivos** a todo lo que se quiere lograr realizar con la realización de esta misión.
- Se considera como **Meta** al fin o propósito que se quiere alcanzar con esta misión.
- Se considera como **Población beneficiaria** a todas las personas que van a resultar beneficiadas con la puesta en práctica de la misión.
- Se considera como **Niveles** a las distintas etapas o fases con que cuenta la misión.
- Se considera como **Servicios** a los servicios de salud que se ofrecen como parte de la misión en dependencia del nivel en que se encuentre.
- Se considera como **Certificación** a un documento legal que acredita la propiedad de algo, ya sea del **terreno** o de un **consultorio**.
- Se considera como **Requisitos** a todas las exigencias que se deben cumplir para acreditar una certificación.
- Se considera como **Consultorio** a un lugar destinado a brindarle atención médica a la población y que cuente con todas las condiciones necesarias para esto.

3.2.2.2 Diagrama de clases del modelo del dominio

El modelo del dominio se describe mediante diagramas en OMMMA-L, específicamente con un diagrama de clases conceptuales significativas en el dominio del problema. A continuación se muestra el diagrama del modelo de dominio de la aplicación a crear.

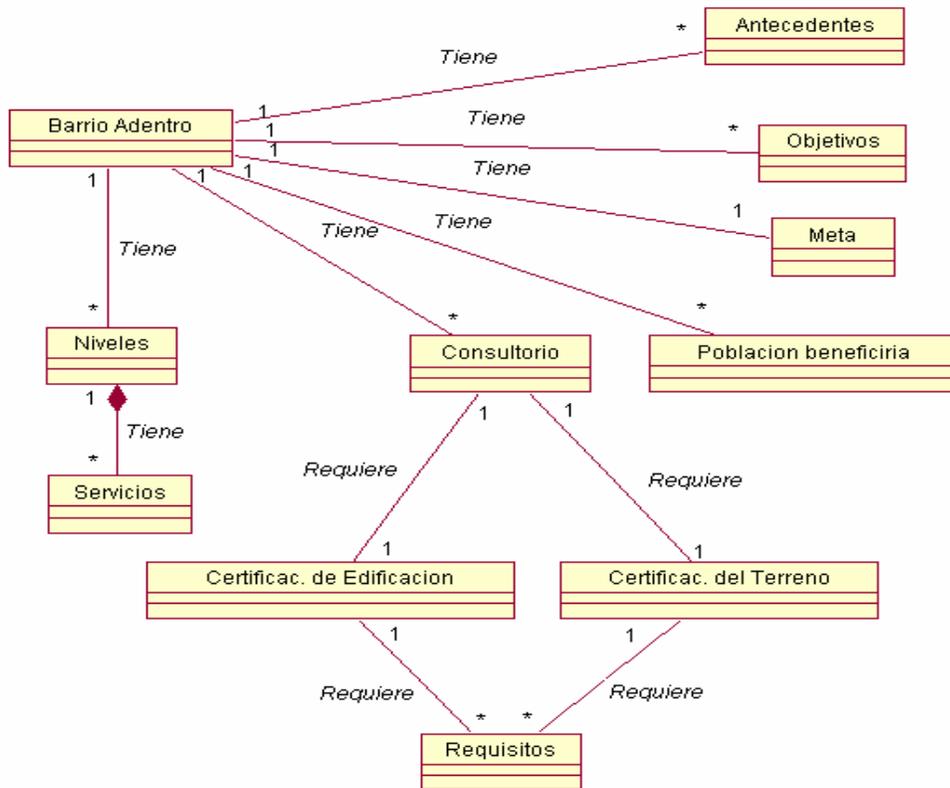


Figura 4. Modelo de Dominio

3.2.2.3 Diagrama de navegación

El diagrama de navegación brinda una visión de lo que se desea lograr, y nos ayuda a crear una idea del camino que se debe seguir.

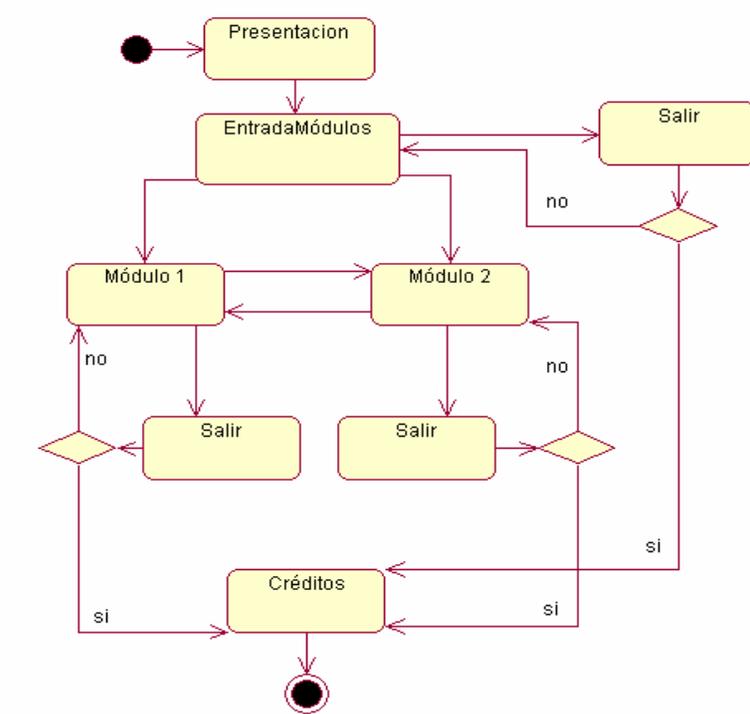


Figura 5. Diagrama de Navegación General

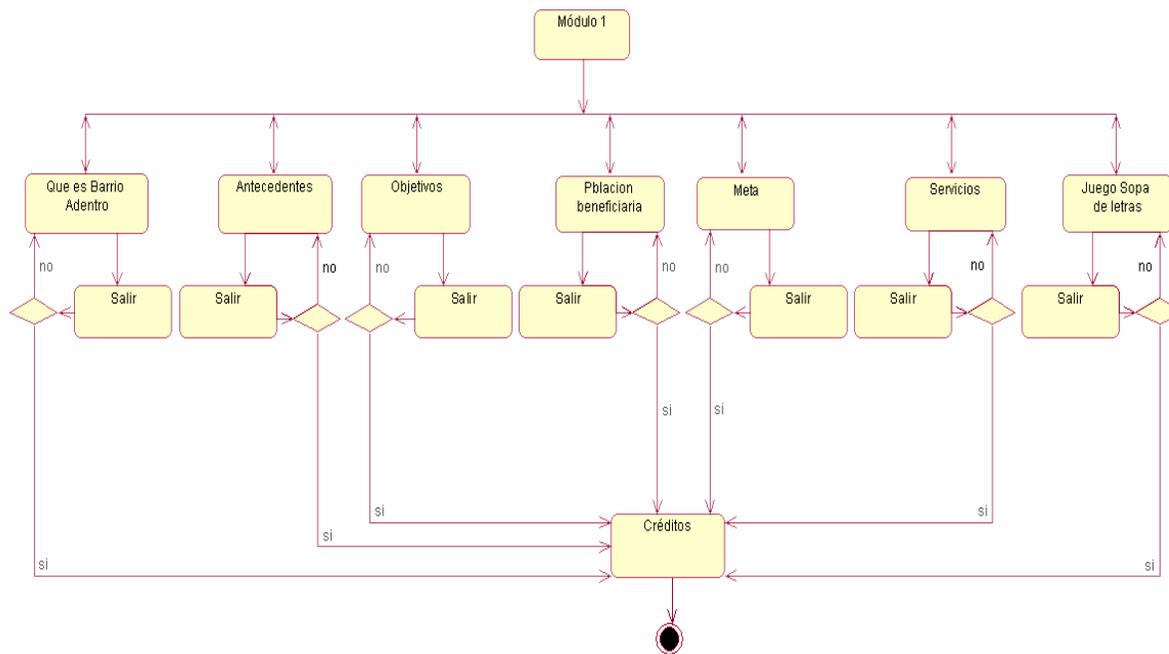


Figura 6. Diagrama de navegación del Módulo 1

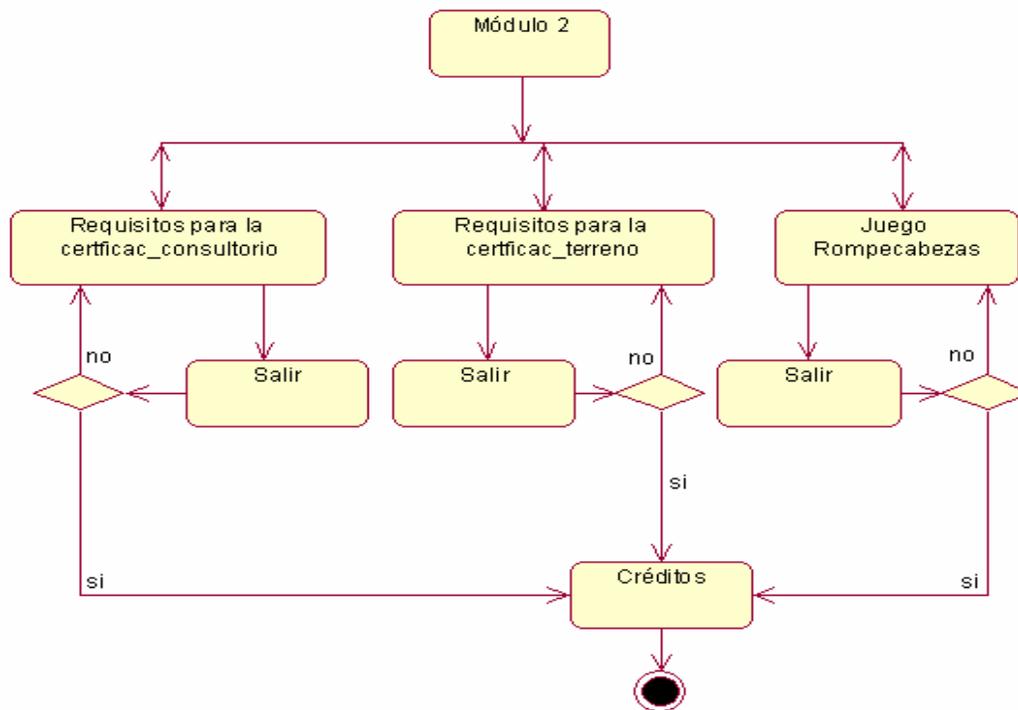


Figura 7. Diagrama de navegación del Módulo 2

3.2.3 Modelo de Casos de uso del sistema

Utilizando las facilidades que brinda OMMMA-L, se representarán los requisitos funcionales del sistema mediante un diagrama de casos de uso. Para ello hay que definir de acuerdo a lo planteado en los epígrafes anteriores, cuales son los actores que van a interactuar con la aplicación, y los casos de uso que van a representar las funcionalidades.

Un caso de uso es un documento narrativo que describe la secuencia de un actor (agente externo) que utiliza un sistema para completar un proceso. Un actor no es parte del sistema, sino un rol que se juega dentro del sistema, que puede intercambiar información o puede ser un recipiente pasivo de información y representa a un ser humano, a un software o a una máquina que interactúa con el sistema. En este caso interactúa un solo actor que se definirá a continuación:

Tabla 2. Actores del sistema

Actor	Justificación
Usuario	Representa a la persona que va a utilizar la aplicación para buscar información sobre la Misión Barrio Adentro.

3.2.3.1 Determinación de los casos de uso

A continuación se presentan los casos de uso determinados para satisfacer los requerimientos funcionales del sistema:

Tabla 3. Determinación del caso de uso Mostrar presentación

CU-1	Mostrar presentación.
Actor	Usuario
Descripción	El usuario ejecuta la aplicación dando lugar a la presentación general. Al concluir la presentación de la aplicación se dará paso automáticamente a la Pantalla de Inicio del producto.
Referencia	R1

Tabla 4. Determinación del caso de uso Controlar audio

CU-2	Controlar audio.
Actor	Usuario
Descripción	El usuario hace clic sobre la opción control de audio de la aplicación,

	pudiendo este activar o desactivar el audio de fondo según su deseo.
Referencia	R2

Tabla 5. Determinación del caso de uso Salir de la aplicación

CU-3	Salir de la aplicación.
Actor	Usuario
Descripción	El usuario hace clic sobre la opción salir de la aplicación. La aplicación muestra una pantalla para confirmar la decisión del usuario y en dependencia de su respuesta se cierra o no la aplicación.
Referencia	R3

Tabla 6. Determinación del caso de uso Acceder a módulos

CU-4	Acceder a módulos.
Actor	Usuario
Descripción	El usuario hace clic sobre uno de los dos módulos de la aplicación y esta muestra la pantalla principal del módulo seleccionado.
Referencia	R4

Tabla 7. Determinación del caso de uso Mostrar contenido

CU-5	Mostrar contenido.
-------------	---------------------------

Actor	Usuario
Descripción	El usuario hace clic sobre uno de los tópicos que ofrece la multimedia y la aplicación se encarga de mostrar la información contenida en el tópico seleccionado.
Referencia	R5.1, R5.2, R5.3, R5.4, R5.5, R5.7, R5.8

Tabla 8. Determinación del caso de uso Mostrar servicios

CU-6	Mostrar servicios.
Actor	Usuario
Descripción	El usuario hace clic sobre la opción “Servicios que ofrece” y se le muestra por defecto los servicios del Nivel 1, si el usuario desea consultar otro nivel que no sea este hace clic sobre el nivel a consultar y la aplicación le muestra la información correspondiente al nivel seleccionado, permitiendo consultar los distintos servicios que se ofrecen en dicho nivel.
Referencia	R5.6

Tabla 9. Determinación del caso de uso Jugar juego Sopa de Letras

CU-7	Jugar juego Sopa de Letras.
Actor	Usuario
Descripción	El usuario hace clic sobre la opción juegos del módulo 1. Después de encontrarse en la pantalla correspondiente puede seleccionar una de las 3 variantes del juego o ejecutar la ayuda.

Referencia	R6
-------------------	----

Tabla 10. Determinación del caso de uso Jugar Rompecabezas

CU-8	Jugar Rompecabezas.
Actor	Usuario
Descripción	El usuario hace clic sobre la opción Juegos del módulo 2, la aplicación muestra la pantalla correspondiente y el usuario comienza a interactuar con el juego.
Referencia	R7

3.2.3.2 Diagrama de casos de uso

Los diagramas donde se representan las relaciones existentes entre los actores y los casos de uso se muestran a continuación.

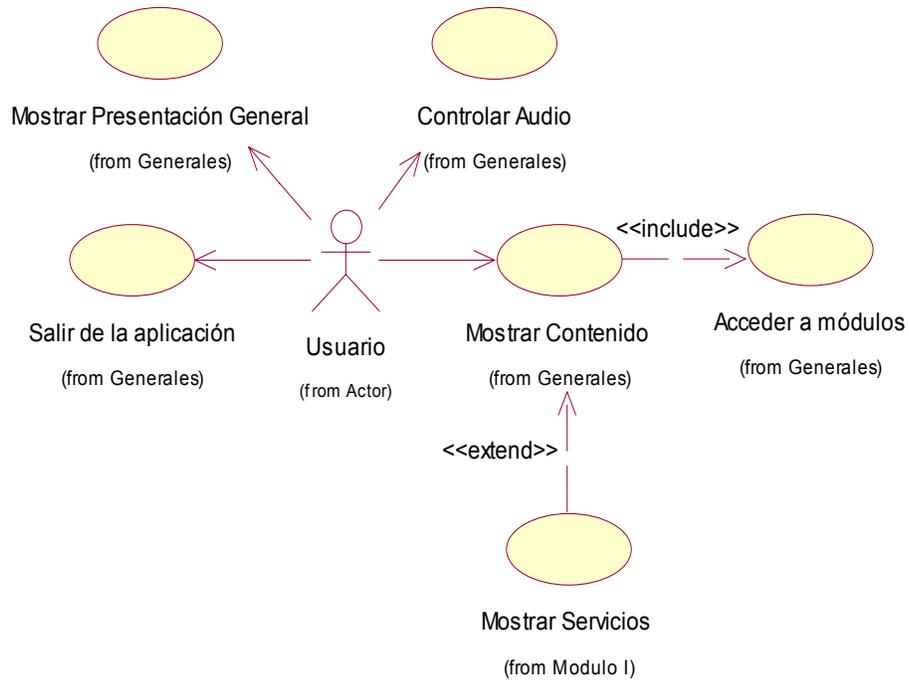


Figura 8. Diagrama de casos de uso Generales

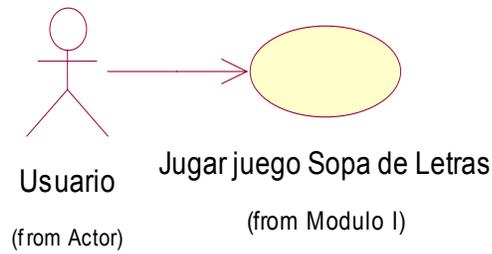


Figura 9. Diagrama de casos de uso del Módulo 1

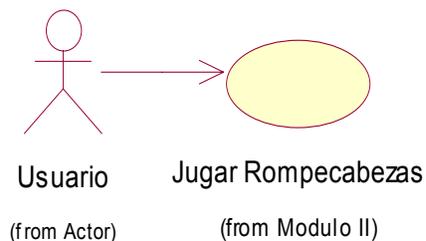


Figura 10. Diagrama de casos de uso del Módulo 2

3.2.3.2.1 Descripción textual de los casos de uso Generales

Tabla 11. Descripción del caso de uso Mostrar presentación

Nombre del CUS	Mostrar presentación
Actores	Usuario (inicia).
Propósito	Mostrar la presentación de la aplicación.
Resumen	El caso de uso inicia con la ejecución de la aplicación por parte del usuario dando lugar a la presentación general en la cual el cursor del ratón no estará visible. La presentación puede ser anulada por parte del usuario al pulsar la tecla “Esc” del teclado. Al concluir la presentación de la aplicación se dará paso automáticamente a la Pantalla de Inicio del producto.
Referencias	R1
Precondiciones	
Flujo normal de los eventos	

Acción del actor		Respuesta del sistema
1. El usuario ejecuta la multimedia por primera vez.		1.1. Carga la presentación de la multimedia Barrio Adentro. 1.2. Ejecuta la presentación y al concluir va a la Pantalla de Inicio terminando el CU.
Flujo alternativo de los eventos		
2. El usuario presiona la tecla Esc del teclado para abortar la presentación.		2.1. Aborta la presentación y pasa a la Pantalla de Inicio finalizando el CU.
Requerimientos funcionales	no	
Poscondiciones		Mostrar la pantalla de inicio.
Prioridad		Secundaria
Mejoras		

Tabla 12. Descripción del caso de uso Controlar audio

Nombre del CUS	Controlar audio
Actores	Usuario (inicia).
Propósito	Permitir el control del audio de la aplicación.
Resumen	El caso de uso inicia cuando el usuario hace clic sobre la opción de control de audio de la aplicación, pudiendo este

	activarla o desactivarla a su antojo.	
Referencias	R2	
Precondiciones		
Flujo normal de los eventos		
Acción del actor	Respuesta del sistema	
1. El usuario hace clic sobre el icono del volumen.	1.1. Cambia el estado del audio : a) Si el audio esta en on, lo pone en off. b) Si el audio esta en off lo pone en on. Finaliza el CU.	
Flujo alternativo de los eventos		
Requerimientos no funcionales		
Poscondiciones	Cambio del estado del audio.	
Prioridad	Secundario	
Mejoras		

Tabla 13. Descripción del caso de uso Salir de la aplicación

Nombre del CUS	Salir de la aplicación
-----------------------	-------------------------------

Actores	Usuario (inicia).	
Propósito	Permitir la salida del usuario de la aplicación.	
Resumen	El caso de uso inicia cuando el usuario hace clic sobre la opción salir, de la aplicación. La aplicación muestra una pantalla para confirmar la decisión del usuario y en dependencia de la respuesta del usuario, se cierra o no la aplicación.	
Referencias	R3	
Precondiciones		
Flujo normal de los eventos		
Acción del actor	Respuesta del sistema	
1. El usuario hace clic sobre la opción salir, de la aplicación.	1.1. Verifica que el usuario desea finalizar la salida, mostrando un cartel de confirmación sobre la pantalla en que se esta trabajando.	
2. El usuario confirma que desea salir de la aplicación.	2.1. Carga la pantalla final, muestra los créditos y finaliza la ejecución de la aplicación, terminando el CU.	
Flujo alternativo de los eventos		
2. a-) El usuario desea continuar ejecutando la aplicación, hace clic en la	2.1. a-) Cierra el cartel de confirmación y continua mostrando la pantalla en la que	

opción no.	se estaba trabajando, finalizando el CU.
Requerimientos no funcionales	
Poscondiciones	
Prioridad	Secundario
Mejoras	

Tabla 14. Descripción del caso de uso Acceder a módulos

Nombre del CUS	Acceder a módulos	
Actores	Usuario (inicia).	
Propósito	Permitir el acceso a los dos módulos de la aplicación.	
Resumen	El caso de uso inicia cuando el usuario solicita consultar uno de los dos módulos de la aplicación y esta muestra la pantalla principal del módulo seleccionado.	
Referencias	R4	
Precondiciones	Que haya culminado el caso de uso Mostrar presentación general.	
Flujo normal de los eventos		
Acción del actor	Respuesta del sistema	
1. El usuario solicita el acceso a unos de	1.1. Va a la pantalla correspondiente del	

<p>los módulos de la aplicación.</p>	<p>módulo seleccionado y la muestra.</p> <p>a) Si es Módulo 1, va a la pantalla principal del módulo, iniciando por defecto en el tópico ¿Qué es Barrio Adentro? Muestra la información y locución correspondiente.</p> <p>b) Si es Módulo 2, va a la pantalla principal del módulo 2. Se escucha una locución dando la bienvenida al módulo y explicando en que consiste el mismo. Da la posibilidad de seleccionar a cual de los dos tópicos de contenidos de este módulo se quiere acceder primero, “Requisitos mínimos para la construcción de una certificación” o “Requisitos mínimos para la certificación de un terreno”.</p> <p>Finaliza el CU.</p>
<p>Flujo alternativo de los eventos</p>	
<p>Requerimientos no funcionales</p>	
<p>Poscondiciones</p>	
<p>Prioridad</p>	<p>Critico</p>
<p>Mejoras</p>	

Tabla 15. Descripción del caso de uso Mostrar contenido

Nombre del CUS	Mostrar contenido	
Actores	Usuario (inicia).	
Propósito	Mostrar la información referida al tópico seleccionado.	
Resumen	El caso de uso inicia cuando el usuario solicita información acerca de los tópicos que ofrece la multimedia, luego la aplicación se encarga de mostrar la información solicitada.	
Referencias	R5.1, R5.2, R5.3, R5.4, R5.5, R5.7, R5.8	
Precondiciones	Que haya culminado el caso de uso Mostrar presentación general.	
Flujo normal de los eventos		
Acción del actor	Respuesta del sistema	
1. El usuario hace clic sobre el tópico deseado.	1.1. Muestra la pantalla con la información y locución correspondiente, finalizando el caso de uso.	
Flujo alternativo de los eventos		
Requerimientos no funcionales		
Poscondiciones	El contenido de los tópicos seleccionados se ha mostrado al	

	usuario.
Prioridad	Critico
Mejoras	

Tabla 16. Descripción del caso de uso Mostrar servicios

Nombre del CUS	Mostrar servicios	
Actores	Usuario (inicia).	
Propósito	Dar la posibilidad de seleccionar el Nivel que se quiere consultar.	
Resumen	El usuario inicia el caso de uso cuando hace clic sobre la opción “Servicios que ofrece” y se le muestra por defecto los servicios del Nivel 1, si el usuario desea consultar otro nivel que no sea este hace clic sobre el nivel a consultar y la aplicación le muestra la información correspondiente al nivel seleccionado.	
Referencias	R5.6	
Precondiciones		
Flujo normal de los eventos		
Acción del actor	Respuesta del sistema	
1. El usuario hace clic sobre la opción “Servicios que ofrece”.	1.1. Muestra por defecto la información y locución referente al nivel 1 y los distintos servicios que se brindan en él.	

2. El usuario desea consultar otro nivel.	2.1. Si elige el nivel 2, ver Sección Nivel 2. 2.2. Si elige el nivel 3, ver Sección Nivel 3.
Sección Nivel 2	
1. Hace clic sobre el nivel 2.	1.1. Muestra la información y locución referente a este nivel y los distintos servicios que se brindan en él.
2. Selecciona un servicio.	2.1. Muestra la información y locución referente al servicio seleccionado.
Sección Nivel 3	
1. Hace clic sobre el nivel 3.	1.1. Muestra la información y locución referente a este nivel.
Flujo alternativo de los eventos	
Requerimientos no funcionales	
Poscondiciones	

Prioridad	Critico
Mejoras	

3.2.3.2.2 Descripción textual de los casos de uso del Módulo 1

Tabla 17. Descripción del caso de uso Jugar juego Sopa de Letras

Nombre del CUS	Jugar juego Sopa de Letras	
Actores	Usuario (inicia).	
Propósito	Posibilitar que el usuario interactúe con el juego Sopa de Letras.	
Resumen	El usuario inicia el caso de uso cuando hace clic sobre la opción juegos. Después de encontrarse en la pantalla correspondiente puede seleccionar una de las 3 variantes del juego o ejecutar la ayuda. Si hace una selección correcta de una palabra la aplicación emite un sonido de aplausos, si la selección es incorrecta se emite un sonido desaprobatorio.	
Referencias	R6	
Precondiciones		
Flujo normal de los eventos		
Acción del actor	Respuesta del sistema	
1. El usuario desde cualquier pantalla del Modulo1 hace clic sobre la opción	1.1. Muestra los juegos y la ayuda correspondiente, mientras se escucha una	

Juegos.	locución.
2. El usuario hace clic sobre una de las opciones dentro de la Pantalla Juegos.	2.1. Si el usuario selecciona alguno de los juegos propuestos, ver Sección Escoger Juego. 2.2. Si el usuario selecciona la ayuda, ver Sección Mostrar Ayuda.
Sección Escoger Juego.	
1. El usuario hace clic sobre uno de los 3 posibles juegos en pantalla.	1.1. Pone en rojo el juego seleccionado y lo muestra.
2. El usuario hace una selección correcta de una palabra en el juego.	2.1. Marca la palabra en azul, la quita de la lista y emite un sonido de aplausos para indicarle al usuario aprobación.
3. Completa todas las palabras del juego.	3.1. Muestra un cartel felicitando al usuario por su finalización.
4. Cierra el cartel de felicitación y hace clic en otro tópico.	4.1. Va a la pantalla que le indique el usuario, finalizando el CU.
Flujo alternativo de los eventos	
2.b) El usuario realiza una selección incorrecta de una palabra dentro del juego.	2.1.b) Emite un sonido desaprobatorio para indicarle al usuario su error.

Flujo alternativo de los eventos	
3.b) El usuario aborta el juego (abandona el juego sin haber concluido).	3.1.b) Va a la pantalla que le indique el usuario, finalizando el CU.
Sección Mostrar Ayuda.	
1. Hace clic sobre la opción Ayuda.	1.1. Muestra en un cartel la ayuda de los juegos.
2. Consulta la ayuda, al terminar la cierra.	2.1. Cierra el cartel de la ayuda. Vuelve a Sección Escoger juego.
Requerimientos funcionales	no
Poscondiciones	
Prioridad	Secundario
Mejoras	

3.2.3.2.3 Descripción textual de los casos de uso del Módulo 2

Tabla 188. Descripción del caso de uso Jugar Rompecabezas

Nombre del CUS	Jugar Rompecabezas
Actores	Usuario (inicia).
Propósito	Posibilitar que el usuario interactúe con el juego.
Resumen	El caso de uso inicia cuando el usuario selecciona la opción Juegos del Modulo 2 y comienza a interactuar

	con el.
Referencias	R7
Precondiciones	
Flujo normal de los eventos	
Acción del actor	Respuesta del sistema
1. El usuario hace clic sobre la opción Juegos del módulo 2.	1.1. Muestra el juego correspondiente mientras se escucha una locución con las orientaciones para comenzar a jugar.
2. El usuario coloca una imagen del rompecabezas.	2.1. Si esta mal colocada el sistema regresa la imagen a su estado inicial. Al cabo de tres intentos fallidos, el sistema lleva la imagen al lugar correcto.
3. El usuario concluye el juego.	2.2. Muestra un cartel de felicitación.
3. El usuario cierra el cartel y hace clic sobre otro de los tópicos de la aplicación.	3.1. Va a la pantalla que le indique el usuario, finalizando el CU.
Flujo alternativo de los eventos	
3.b). El usuario aborta el juego, aun sin finalizar hace clic sobre otro de los tópicos de la aplicación.	3.1.b) Va a la pantalla que le indique el usuario, finalizando el CU.

Requerimientos funcionales	no	
Poscondiciones		
Prioridad		Secundaria
Mejoras		

3.3. Conclusiones

En este capítulo se comenzó a desarrollar la propuesta de solución, obteniéndose a partir del análisis de los conceptos que intervienen en el dominio del problema, un listado con las funciones que debe tener la aplicación, representadas mediante un Diagrama de Casos de Uso del sistema de los cuales se hizo una descripción textual describiendo las acciones de los actores de la aplicación y las respuestas que se debería dar. Después de haber concluido todo esto se puede comenzar a construir la aplicación, tratando de que se cumplan todos los requerimientos y las funciones que han sido consideradas necesarias en este capítulo.



CAPÍTULO CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

4.1 Introducción

En este capítulo se modelan los artefactos que ayudan a manejar las complicaciones que implican la construcción de la aplicación propuesta. Para ello los componentes de la aplicación se tratan como clases, utilizando para ello el lenguaje extendido OMMMA-L. Además, se realiza el modelo de implementación en el cual se describe cómo se implementan los elementos del modelo de diseño en términos de componentes y se realiza el modelo de despliegue para una mejor comprensión de la correspondencia entre la arquitectura software y la arquitectura hardware.

4.2 Modelo de diseño

4.2.1 Diagramas de presentación

Los Diagramas de Presentación son un artefacto nuevo dentro del lenguaje de modelado OMMMA-L, sirven para describir la parte estática del modelo a través de una descripción intuitiva de la distribución espacial de objetos visuales de la interfaz de usuario. Estos diagramas tienen el propósito de declarar las interfaces de usuario con un conjunto de estructuras delimitadas en tamaño y área, dividiéndose en objetos de visualización (texto, gráfico, video, animación) e interacción (scrolls, barras de menú, botones, campos de entrada y salida, hipertextos con hipervínculos).

Los Diagramas de Presentación utilizan las mismas notaciones que el Diagrama de Clases de UML, pero incorporando las clases correspondientes a las medias.

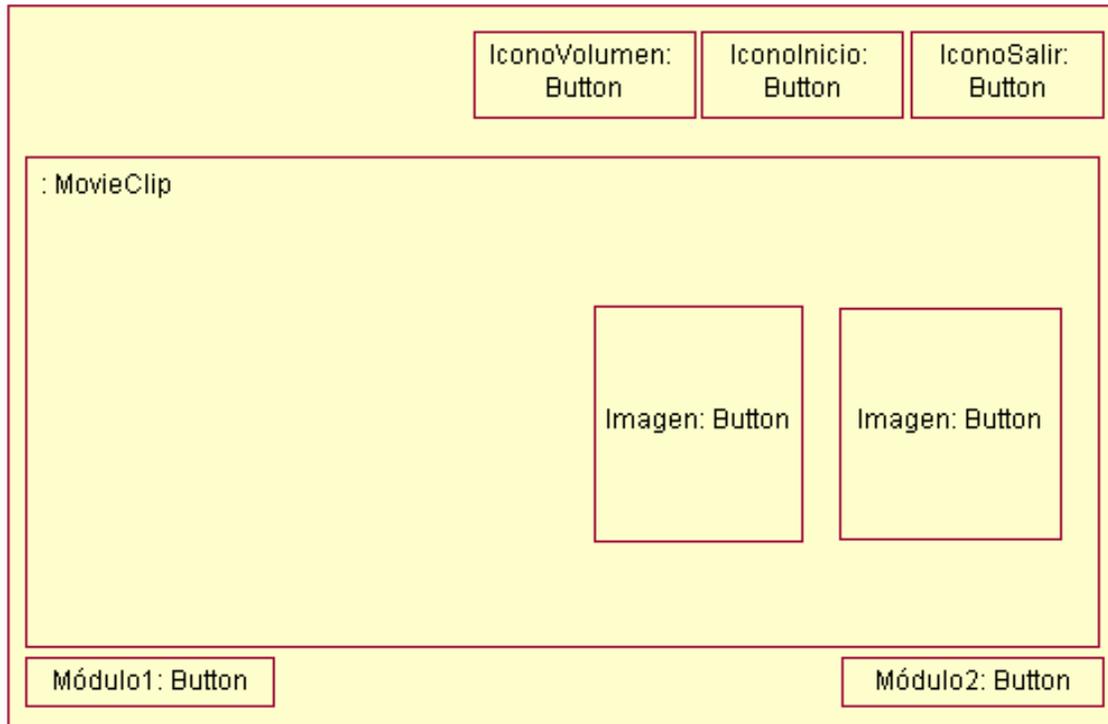


Figura 11. Diagrama de presentación Entrada a Módulos

4.2.1.1 Diagramas de presentación del Módulo 1

La pantalla de presentación Entrada a Módulo1 coincide con las pantallas “Que es Barrio Adentro”, “Meta”, “Objetivos” y “Población beneficiaria”.

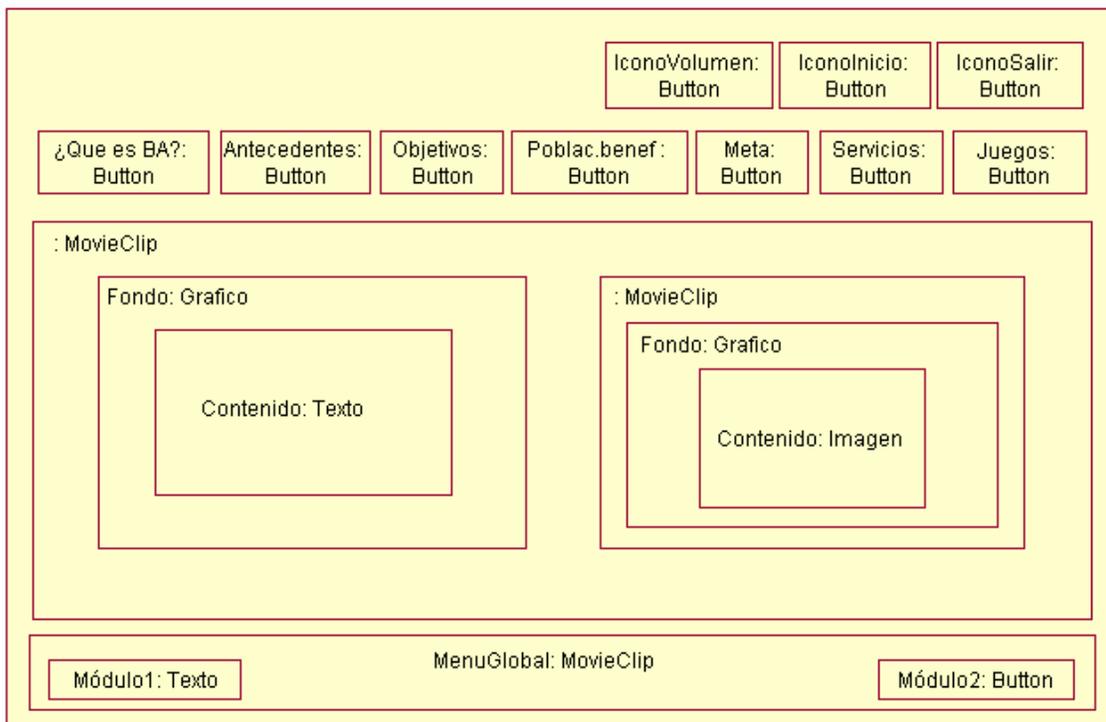


Figura 12. Diagrama de presentación Entrada a Módulo1

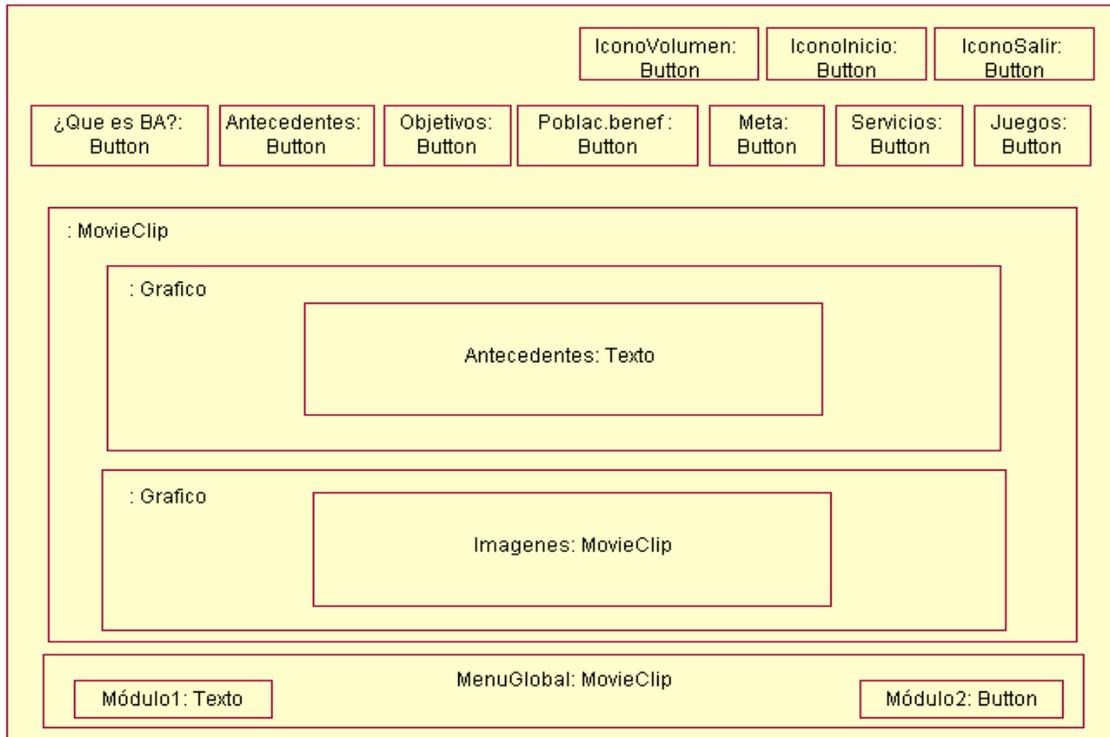


Figura 13. Diagrama de presentación Antecedentes

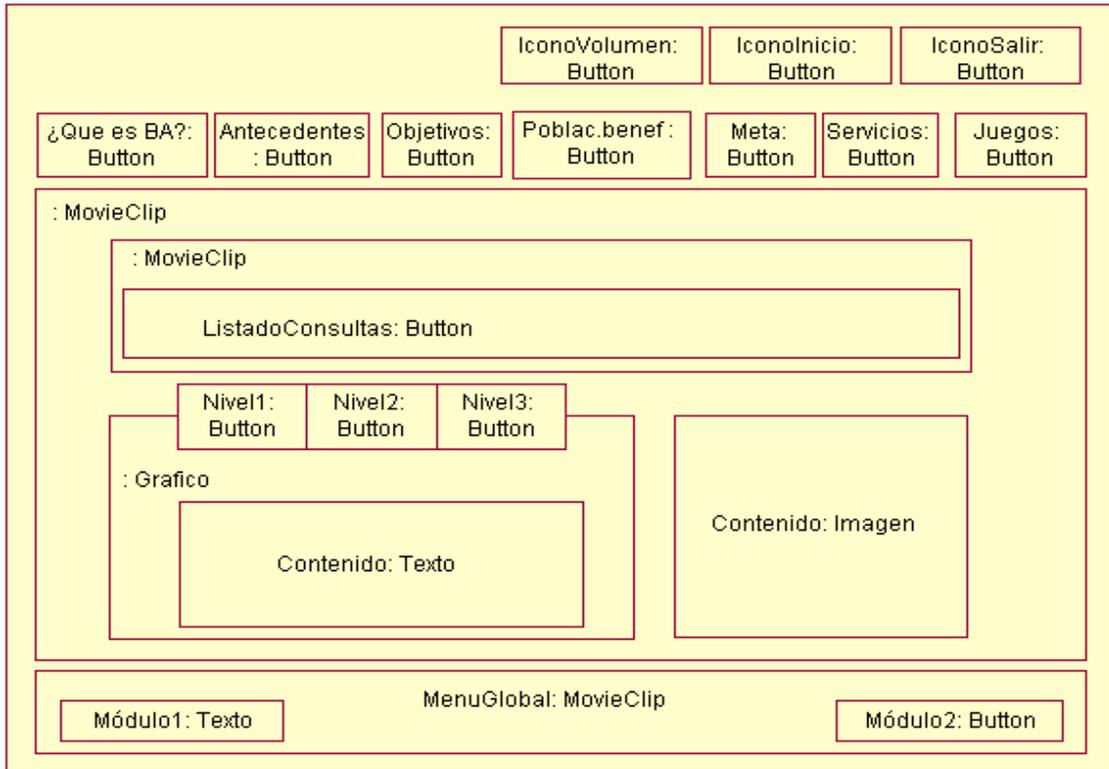


Figura 14. Diagrama de presentación Servicios Nivel1, Nivel2

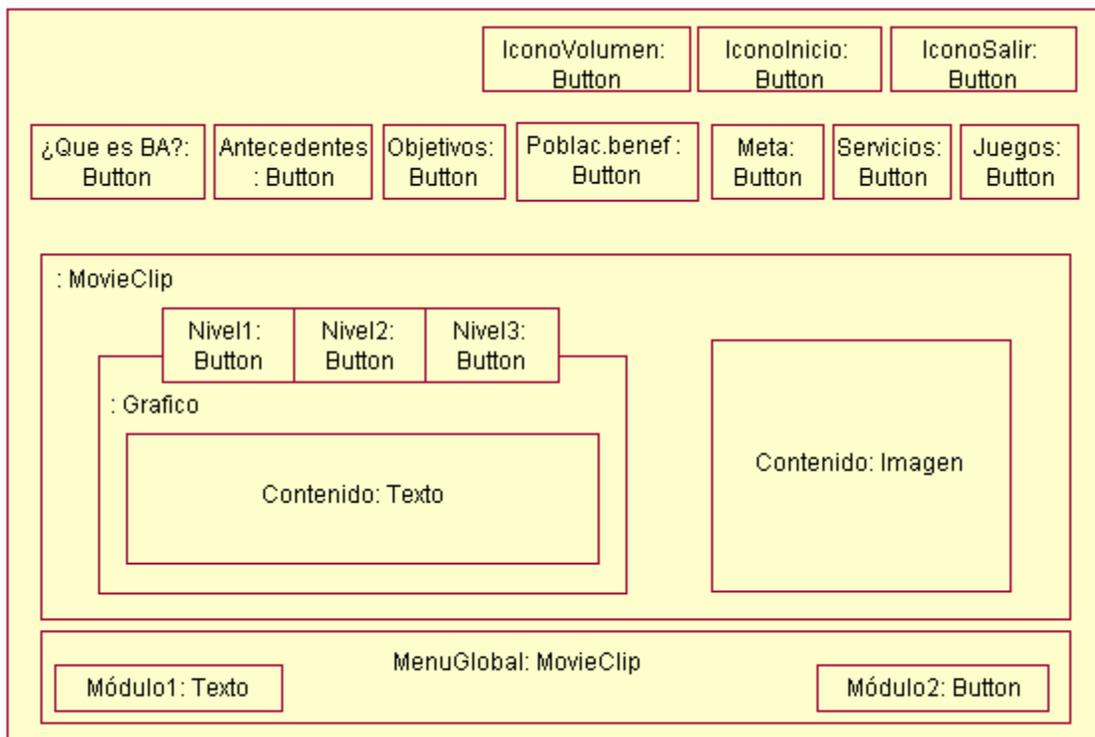


Figura 15. Diagrama de presentación Servicios Nivel3

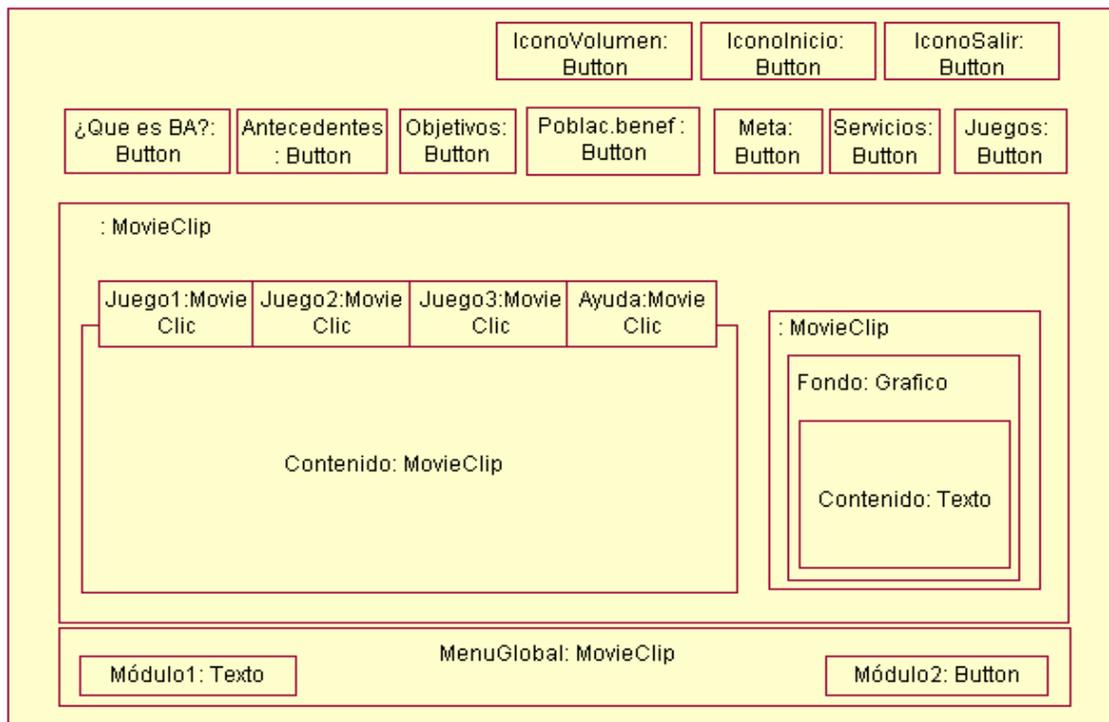


Figura 16. Diagrama de presentación JuegosM1

4.2.1.1 Diagramas de presentación del Módulo 2

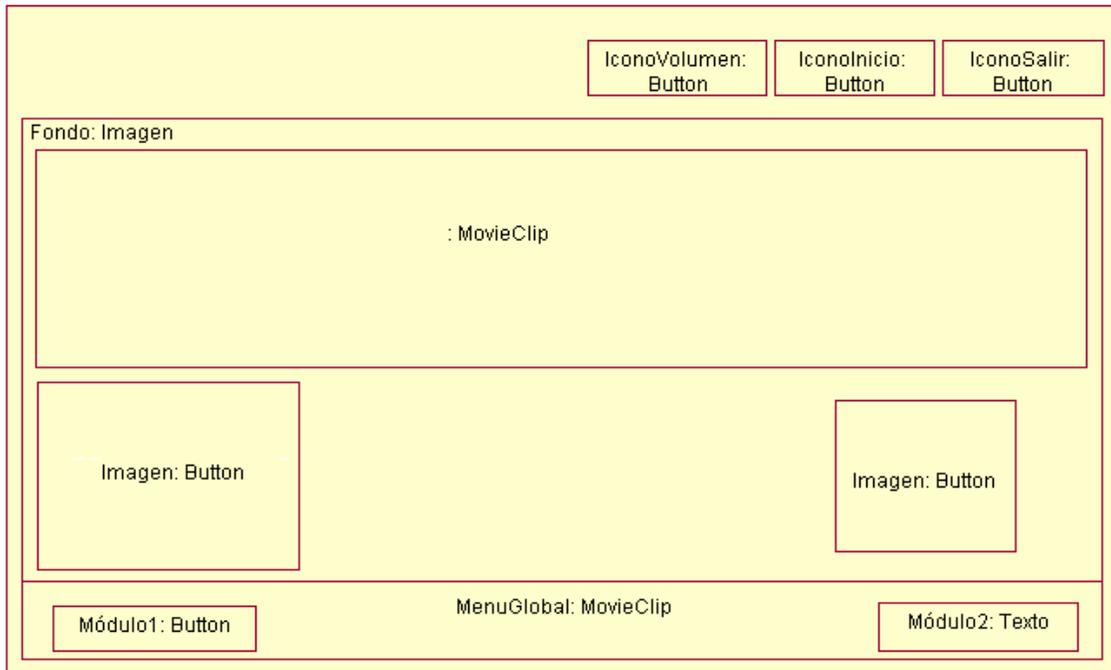


Figura 17. Diagrama de presentación Entrada M2

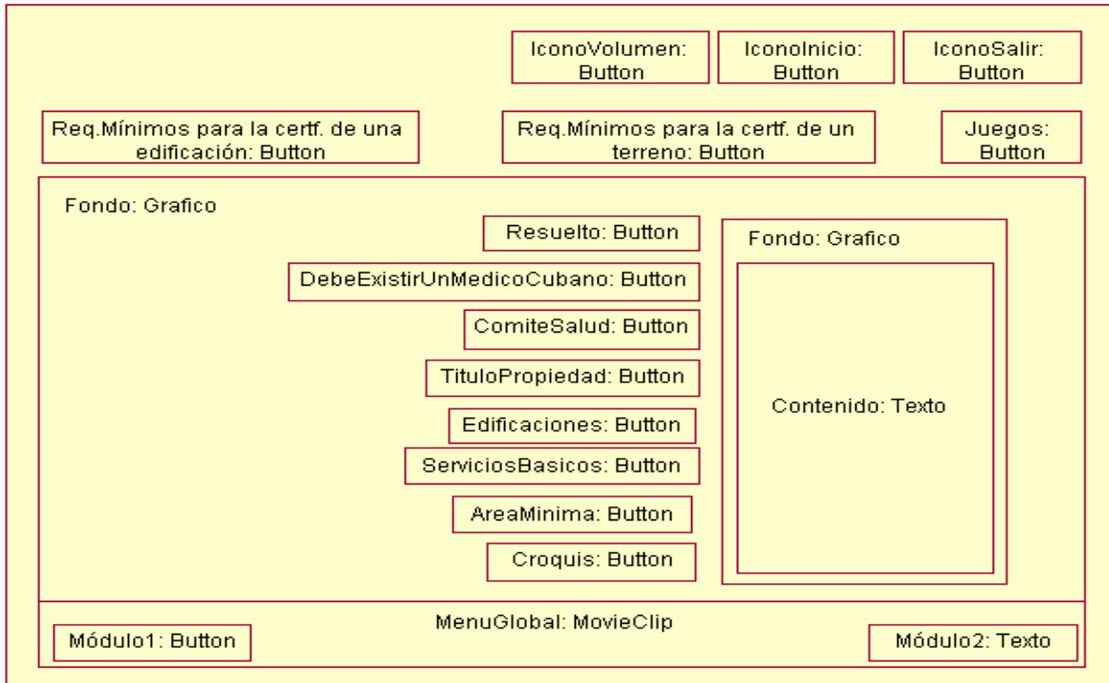


Figura 18. Diagrama de presentación Requisitos

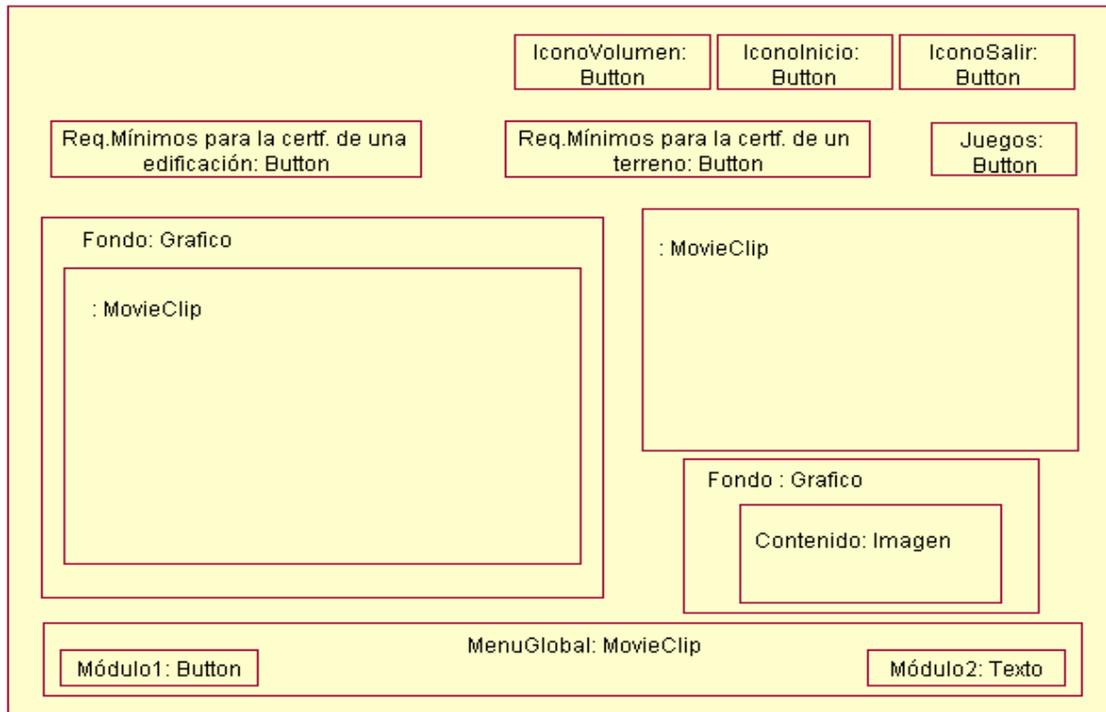


Figura 19. Diagrama de presentación JuegosM2

4.3 Modelo de Implementación

El modelo de implementación describe cómo se implementan los elementos del modelo de diseño en términos de componentes, y a su vez como se organizan y se relacionan unos con otros de dichos componentes; definiendo un componente como el empaquetamiento físico de los elementos de un modelo, como es el caso de las clases del modelo de diseño.

4.3.1 Modelo de componentes

El modelo de componentes describe los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Las relaciones de dependencia se utilizan en el modelo de componentes para indicar que un componente utiliza los servicios ofrecidos por otro componente.

En este modelo se representan las dependencias entre componentes software, incluyendo componentes de código fuente, componentes del código binario, y componentes ejecutables. Un módulo de software se puede representar como componente. También se pueden mostrar interfaces y las dependencias de llamada entre componentes, usando flechas con líneas discontinuas desde los componentes, a las interfaces de otros componentes (ANÓNIMO).

A continuación se muestran los diagramas de componentes de la aplicación a construir.

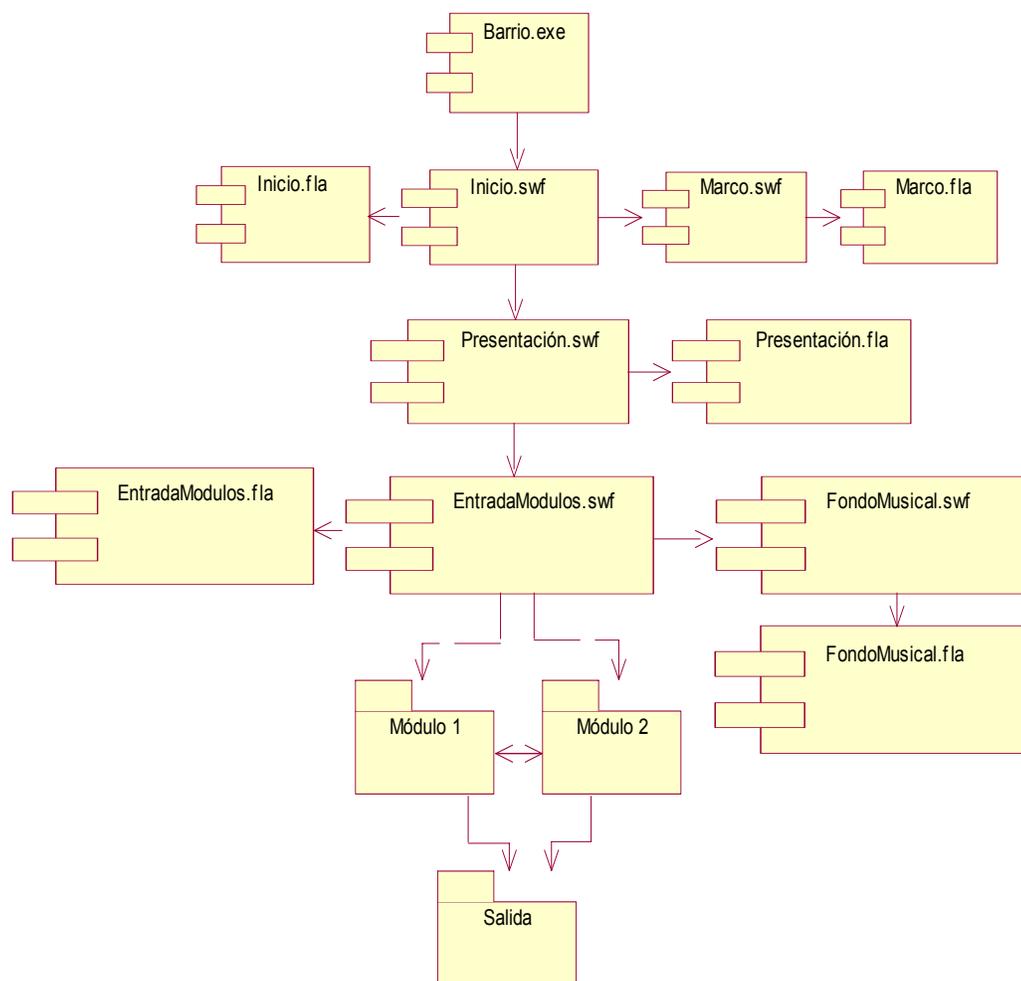


Figura 20. Diagrama de componentes General

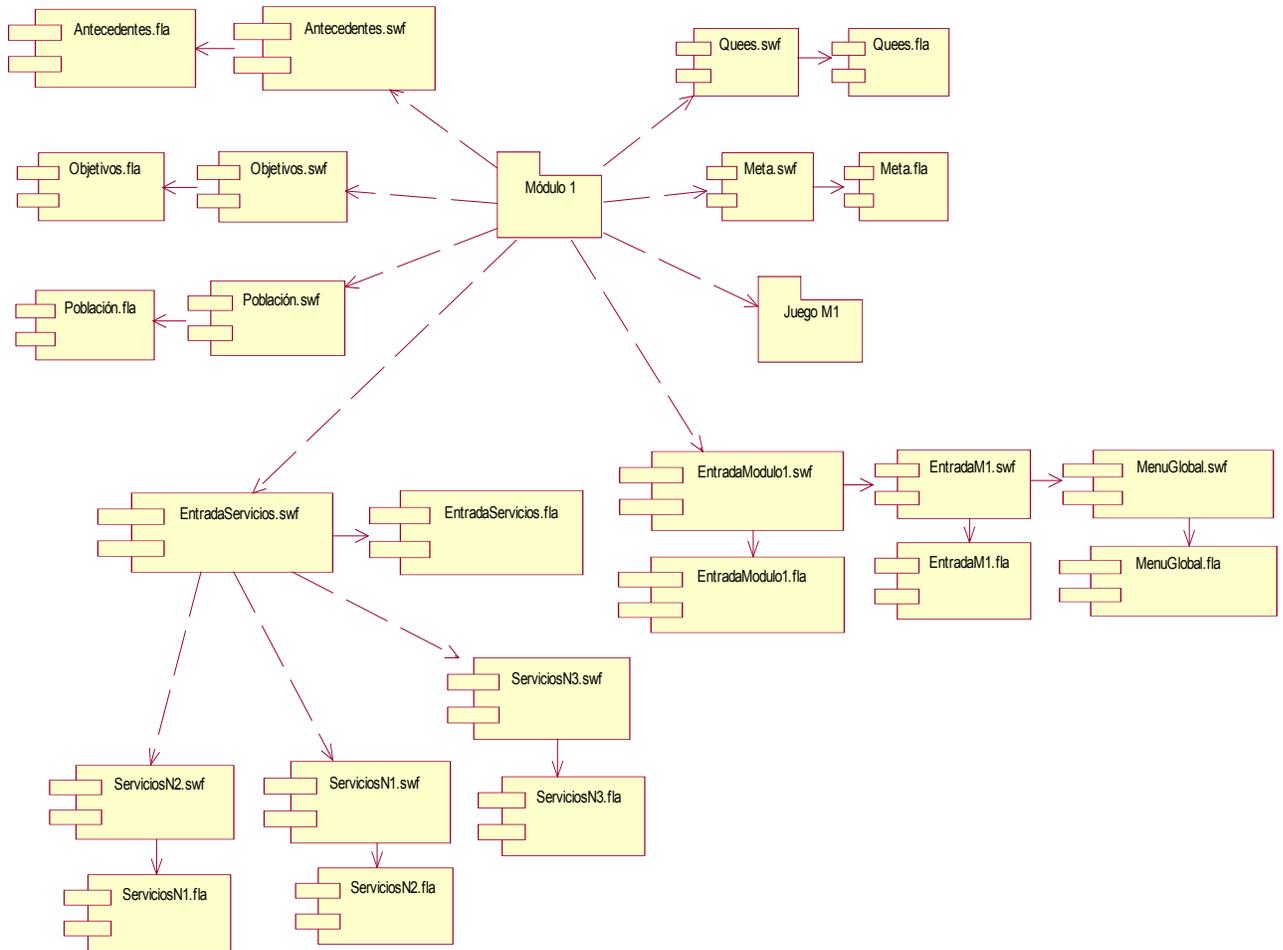


Figura 21. Diagrama de componentes Módulo 1

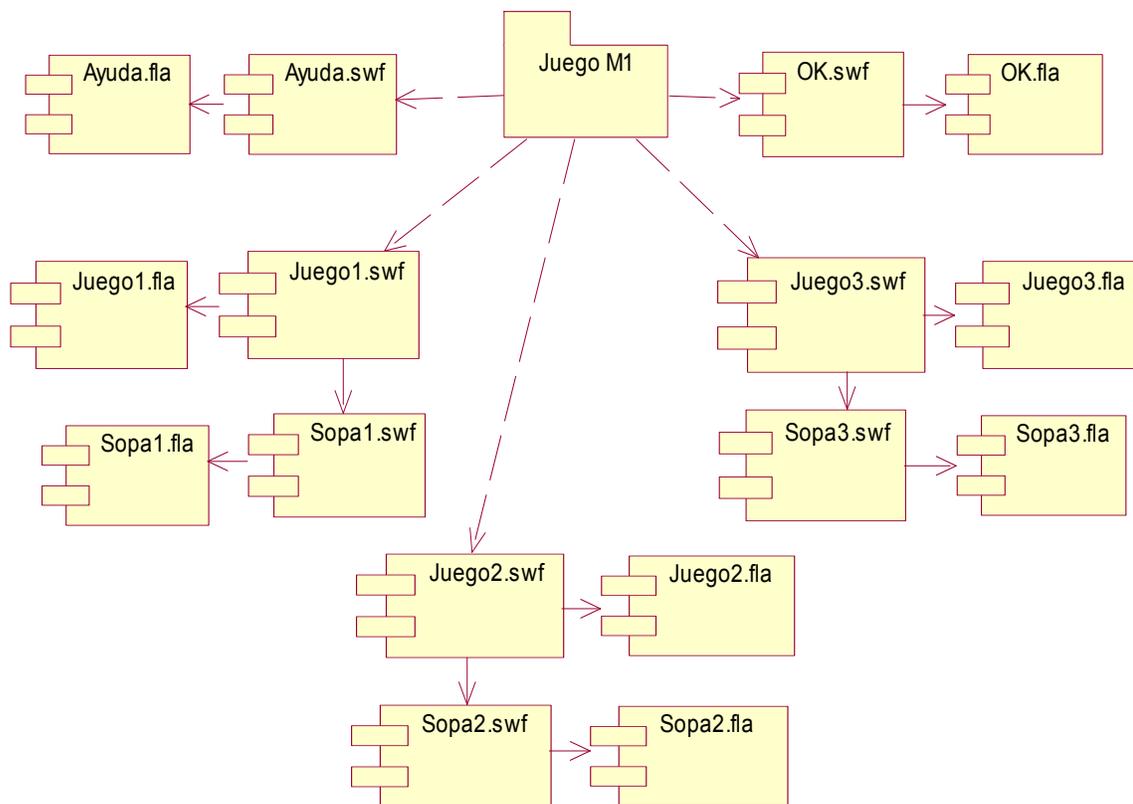


Figura 22. Diagrama de componentes Juego M1

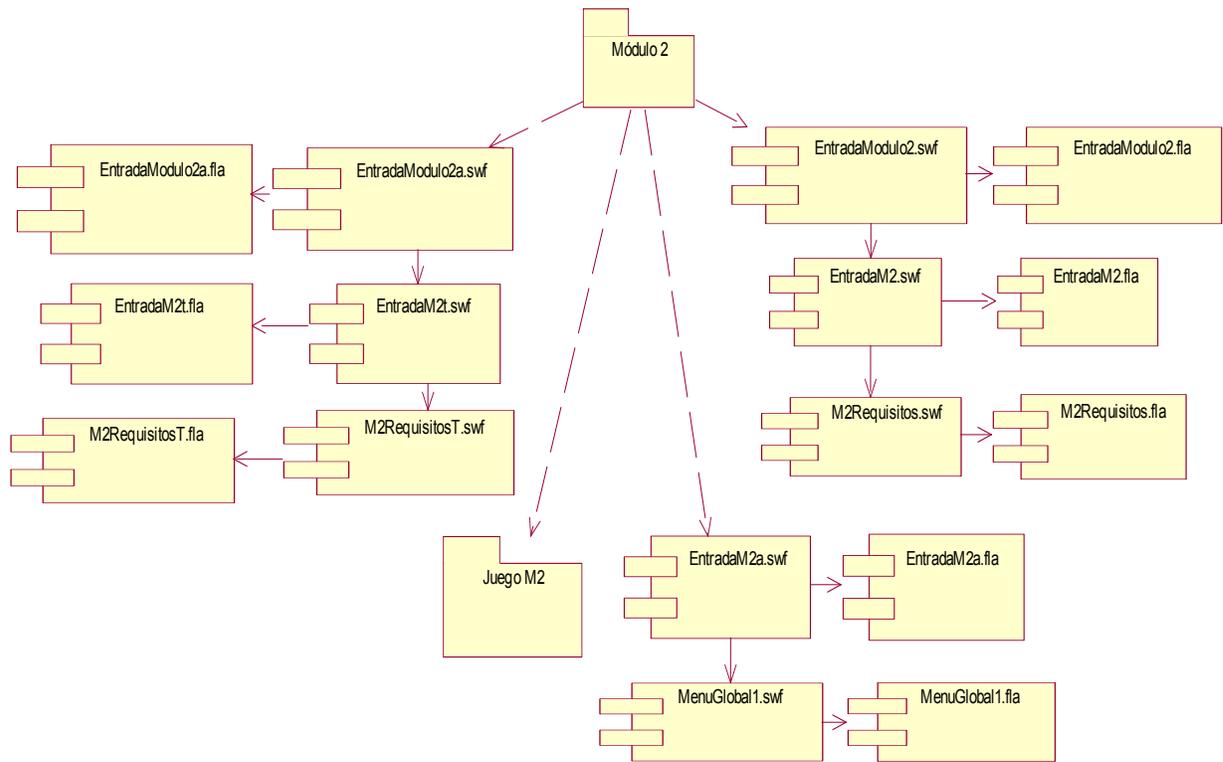


Figura 23. Diagrama de componentes Módulo 2

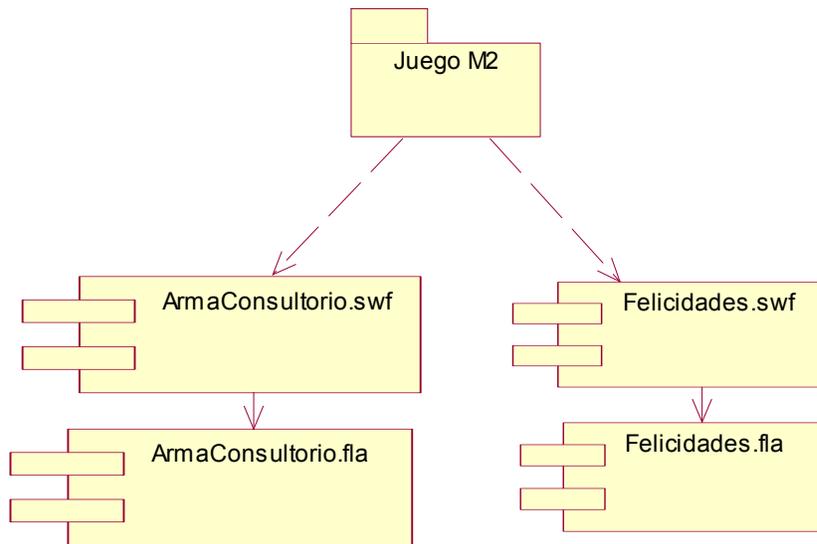


Figura 24. Diagrama de componentes Juego M2

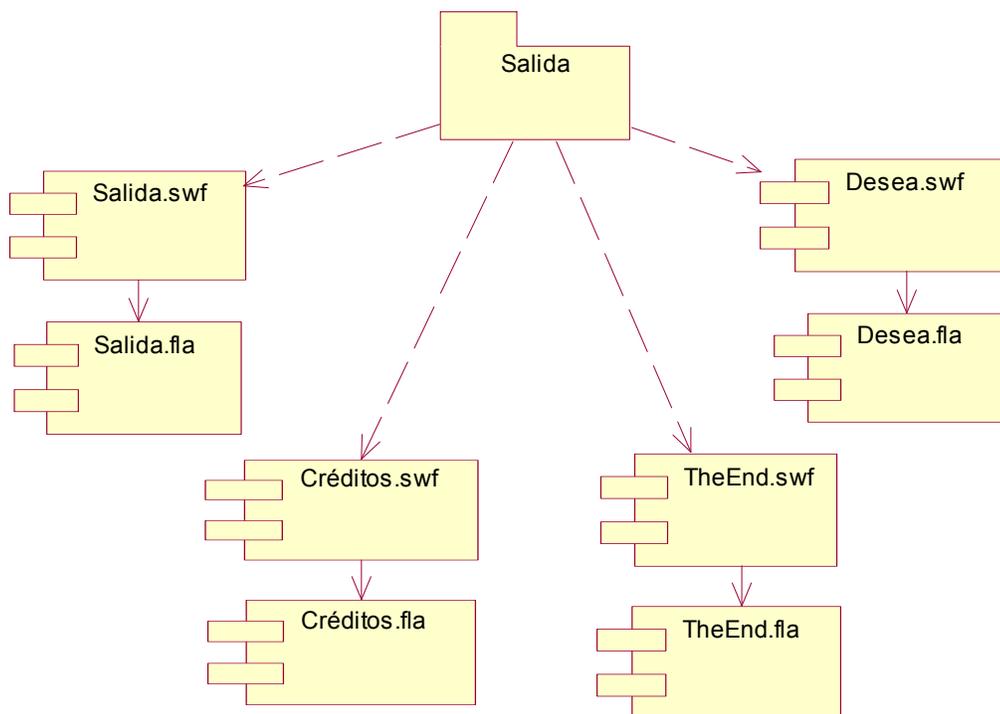


Figura 25. Diagrama de componentes Salida

4.4 Modelo de despliegue

El modelo de despliegue describe la distribución física del sistema en nodos de información; así como muestra como están distribuidos los componentes de software entre los distintos nodos. Incorpora los elementos establecidos en la arquitectura para completar la descripción física de la aplicación; permite comprender la correspondencia entre la arquitectura software y la arquitectura hardware.

En el caso de este proyecto el modelo de despliegue consiste en un solo ordenador ya que aquí será donde radique la aplicación, y desde donde el usuario va a poder acceder a ella y consultarla. Esta PC u ordenador personal necesita tener los kit de multimedia y una unidad de lectura de CD, no necesariamente necesita estar conectada a la red.

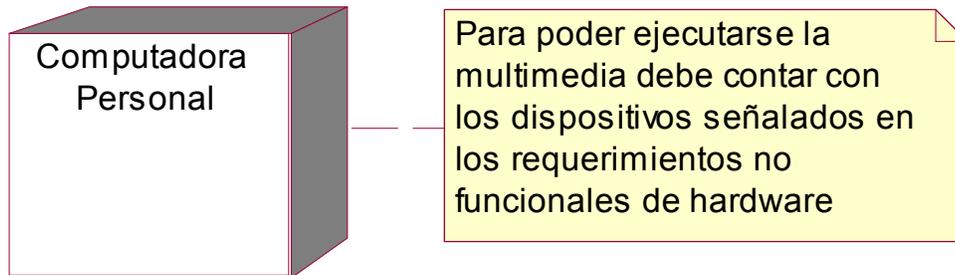


Figura 26. Modelo de despliegue

4.5 Conclusiones

En este capítulo se mostraron varias vistas para llevar a cabo el proceso de implementación del sistema. Se utilizaron diagramas de presentación para explicar la lógica de la aplicación, y se realizó el modelo de componentes y el modelo de despliegue para una mejor comprensión del funcionamiento de la misma. En este momento, ya se tiene confeccionada completamente la propuesta que trae este trabajo.



CAPÍTULO ESTUDIO DE FACTIBILIDAD

5.1 Introducción

En la actualidad existen varios métodos para demostrar la factibilidad de un producto, algunos de estos son el método de Casos de Usos, el de Análisis de Puntos de Función y el COCOMO II.

El método de Casos de Usos es uno de los métodos más efectivos para capturar la funcionalidad de un sistema, este método permite documentar los requerimientos de un sistema en términos de Actores y Casos de Uso (PERALTA).

Para la estimación del tamaño de un sistema a partir de sus requerimientos, es el de Análisis de Puntos de Función. Ésta técnica permite cuantificar el tamaño de un sistema en unidades independientes del lenguaje de programación, las metodologías, plataformas y/o tecnologías utilizadas, denominadas Puntos de Función (PERALTA).

Por otro lado, el SEI (del inglés, Software Engineering Institute) propone desde hace algunos años un método para la estimación del esfuerzo llamado COCOMO II. Éste método está basado en 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) (PERALTA).

Para la realización de este trabajo y su análisis de factibilidad el método que se ha propuesto y el que se va a usar es una variante más reciente denominada Análisis de Puntos de Casos de Uso, la cual es en cierta medida similar al Análisis de Puntos de Función.

En este capítulo se hace un estudio de factibilidad sobre el sistema utilizando el método de Puntos de Casos de Usos, obteniendo los beneficios tangibles e intangibles y analizando los costos del desarrollo de esta propuesta.

5.2 Planificación basada en Casos de Uso

La estimación mediante el análisis de Puntos de Casos de Uso es un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

5.2.1 Calculo de puntos de casos de uso sin ajustar

El cálculo de Puntos de Casos de usos sin ajustar se calcula a partir de la siguiente ecuación:

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

Donde,

UUCP: Puntos de Casos de Uso sin ajustar

UAW: Factor de Peso de los Actores sin ajustar

UUCW: Factor de Peso de los Casos de Uso sin ajustar

5.2.1.1 Factor de Peso de Actores sin ajustar (UAW)

El Factor de Peso de Actores sin ajustar se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Actores se establece teniendo en cuenta en primer lugar si se trata de una persona o de otro sistema, y en segundo lugar, la forma en la que el actor interactúa con el sistema. Los criterios se muestran en la siguiente tabla:

Tabla 19. Factor de Peso de los Actores sin ajustar

Tipo de Actor	Descripción	Factor de Peso	Cantidad de Actores
Simple	Otro sistema que	1	0

	interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)		
Medio	Otro sistema que interactúa con el sistema a desarrollar Mediante un protocolo o una interfaz basada en texto.	2	0
Complejo	Una persona que interactúa con el sistema mediante una interfaz grafica.	3	1

Como el Usuario constituye un actor de tipo complejo, ya que se trata de una persona utilizando el sistema mediante una interfaz gráfica, al cual se le asigna un peso de 3, el factor de peso de los actores sin ajustar resulta:

$$UAW = \sum \text{cant actores} * \text{Peso}$$

$$UAW = 1 * 3 = 3$$

$$UAW = 3$$

5.2.1.2 Factor de Peso de los Casos de Uso sin ajustar (UUCW)

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia. Los criterios se muestran en la siguiente tabla:

Tabla 190. Factores de Peso de los Casos de Uso sin ajustar

Tipos de Casos de Uso	Descripción	Factor de Peso	Casos de Uso
Simple	El caso de Uso contiene de 1 a 3 transacciones.	5	8
Medio	El caso de Uso contiene de 4 a 7 transacciones.	10	0
Complejo	El caso de Uso contiene más de 8 transacciones.	15	0

Cada uno de los casos de uso para la realización de la aplicación en cuestión contiene de 1 a 3 transacciones, por lo que se tienen entonces 8 casos de uso de tipo simple (peso 5), con lo cual el factor de peso de los casos de uso sin ajustar resulta:

$$\text{UUCW} = \Sigma \text{cant CU} * \text{Peso}$$

$$\text{UUCW} = 8 \times 5 = 40$$

$$\text{UUCW} = 40$$

Finalmente, los Puntos de Casos de Uso sin ajustar resultan

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

$$\text{UUCP} = 3 + 40 = 43$$

5.2.2 Cálculo de los puntos de casos de uso ajustados

$$\text{UCP} = \text{UUCP} * \text{TCF} * \text{EF}$$

Donde:

UCP: Puntos de Casos de Usos Ajustados

UUCP: Puntos de Casos de Usos Sin Ajustar

TCF: Factor de Complejidad Técnica

EF: Factor de Ambiente

5.2.2.1 Factor de complejidad técnica (TCF)

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante. En la siguiente tabla se muestra el significado y el peso de cada uno de éstos factores:

Tabla 201. Factores de complejidad técnica

Factor	Descripción	Peso	Valor asignado
T1	Sistema distribuido	2	0

T2	Objetivos de performance o tiempo de respuesta	1	5
T3	Eficiencia del usuario final	1	2
T4	Procesamiento interno complejo	1	0
T5	El código debe ser reutilizable	1	0
T6	Facilidad de instalación	0.5	5
T7	Facilidad de uso	0.5	4
T8	Portabilidad	2	5
T9	Facilidad de cambio	1	4
T10	Concurrencia	1	0
T11	Incluye objetivos especiales de seguridad	1	1
T12	Provee acceso directo a terceras partes	1	5

T13	Se requieren facilidades especiales de entrenamiento a usuarios	1	1
-----	---	---	---

El Factor de complejidad técnica se calcula mediante la siguiente ecuación:

$$\text{TCF} = 0.6 + 0.01 * \Sigma (\text{Peso}i * \text{Valor asignado}i)$$

Según los valores anteriores el factor de complejidad técnica sería:

$$\text{TFC} = 0.6 + 0.01 * (0+5+2+0+0+0.25+0.20+10+4+0+1+5+1)$$

$$\text{TFC} = 0.6 + 0.01 * 31.45$$

$$\text{TFC} = 0.9145 = 0.91$$

5.2.2.2 Factor de ambiente (EF)

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente. El cálculo del mismo es similar al cálculo del Factor de complejidad técnica, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5.

En la siguiente tabla se muestra el significado y el peso de cada uno de éstos factores.

Tabla 212. Factores de ambiente

Factor	Descripción	Peso	Valor asignado
E1	Familiaridad con el modelo de proyecto utilizado	1.5	3

E2	Experiencia en la aplicación	0.5	3
E3	Experiencia en orientación a objetos	1	4
E4	Capacidad del analista líder	0.5	4
E5	Motivación	1	5
E6	Estabilidad de los requerimientos	2	3
E7	Personal part-time	-1	0
E8	Dificultad del lenguaje de programación	-1	2

El Factor de ambiente se calcula mediante la siguiente ecuación:

$$EF = 1.4 - 0.03 * \sum (\text{Peso}_i * \text{Valor asignado}_i)$$

$$EF = 1.4 - 0.03 * (4.5 + 1.5 + 4 + 0.20 + 5 + 6 + 0 - 2)$$

$$EF = 1.4 - 0.03 * 19.20 = 1.4 - 0.576$$

$$EF = 0.82$$

Finalmente, los **Puntos de Casos de Uso ajustados (UCP)** resultan:

$$UCP = UUCP * TCF * EF$$

$$UCP = 43 * 0.91 * 0.82$$

UCP =32.09

5.2.3 Estimación de esfuerzo a través de los puntos de casos de uso

Para el cálculo de la estimación del esfuerzo a través de los Puntos de Casos de Uso existen varios criterios, en este trabajo el cálculo de estos valores se hará siguiendo el siguiente criterio:

1. Se contabilizan cuántos factores de los que afectan al Factor de Ambiente están por debajo del valor medio (3), para los factores E1 a E6.
2. Se contabilizan cuántos factores de los que afectan al Factor de Ambiente están por encima del valor medio (3), para los factores E7 y E8.
3. Si el total es 2 o menos, se utiliza el factor de conversión 20 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 20 horas-hombre.
4. Si el total es 3 o 4, se utiliza el factor de conversión 28 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 28 horas-hombre.
5. Si el total es mayor o igual que 5, se recomienda efectuar cambios en el proyecto, ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

El esfuerzo en horas-hombre viene dado por:

$$E = UCP * CF$$

Donde,

E: Esfuerzo estimado en horas-hombre

UCP: Puntos de Casos de Uso Ajustados

CF: Factor de Conversión

Como en este caso el factor de conversión es 20 debido a que la suma de los factores es 2, la ecuación resultante quedaría:

$$E = 32.09 * 20$$

$$E = 641.8 \text{ Horas-Hombre}$$

Se debe tener en cuenta que éste método proporciona una estimación del esfuerzo en horas-hombre contemplando sólo el desarrollo de la funcionalidad especificada en los casos de uso.

Finalmente, para una estimación más completa de la duración total del proyecto, hay que agregar a la estimación del esfuerzo obtenida por los Puntos de Casos de Uso, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software.

Para ello se puede tener en cuenta el siguiente criterio, que estadísticamente se considera aceptable. El criterio plantea la distribución del esfuerzo entre las diferentes actividades de un proyecto, según la aproximación que se muestra en la siguiente tabla:

Tabla 223. Distribución del Esfuerzo por Flujo de Trabajo

Actividad	Porcentaje	Horas-Hombre
Análisis	10.00%	160.4
diseño	20.00%	320.9
Programación	40.00%	641.8
Pruebas	15.00%	240.7
Sobrecarga (otras actividades)	15.00%	240.7
Total	100.00%	1604.5

Con éste criterio, y tomando como entrada la estimación de tiempo calculada a partir de los Puntos de Casos de Uso, se pueden calcular las demás estimaciones para obtener la duración total del proyecto.

Si el esfuerzo total resultante en la tabla anterior es de 1604.5 horas-hombre y por cada mes solo voy trabajar 176 horas, eso daría un **ET** = 9.116 mes-hombre. Si en el proyecto trabajan 6 hombres entonces el tiempo de desarrollo sería de 1.52 mes.

Después de haber obtenido todos estos valores podemos calcular el **Costo Total (CT)** del proyecto, el cual sería:

CT= Salario Mensual * Cantidad de Hombres* Tiempo de Desarrollo

Salario Mensual= \$225

Cantidad de hombres=6

Tiempo de desarrollo= 1.52 meses

CT=225 * 6 * 1.52 = \$2052

5.3 Beneficios tangibles e intangibles

La aplicación multimedia Barrio Adentro es un producto comercial netamente informativo siendo su principal objetivo precisamente el de informatizar a la población venezolana sobre la misión con este nombre, es por ello que como beneficios intangibles asociados al desarrollo de esta multimedia se señalan los siguientes:

- Centralización de la información disponible sobre la misión Barrio Adentro y los servicios que se brindan como parte de ella.
- Ahorro de tiempo en la búsqueda de información referente a la misión Barrio Adentro.
- Disponibilidad de información sobre la misión Barrio Adentro en todas las comunidades de Venezuela de una forma accesible para todos.
- Aumento de la velocidad de propagación de información.
- Aumento de la cultura sobre la misión Barrio Adentro.

5.4 Análisis de costos y beneficios

La aplicación propuesta tiene un costo real, según los cálculos realizados en este capítulo, aproximado a los \$2052 pesos por concepto de salario contando con 6 hombres para su desarrollo en un tiempo alrededor de 1.52 mes.

Los desarrolladores con que se cuenta para la realización de esta aplicación son estudiantes y no reciben salario por lo que analizando los beneficios tangibles e intangibles, en especial el ahorro en tiempo, la mayor productividad de aplicaciones multimedia y la disminución del esfuerzo resulta evidente que es factible el desarrollo de dicha herramienta.

El desarrollo de esta aplicación no supone grandes gastos de recursos, ni tampoco de tiempo, su fácil utilización y navegabilidad no genera daño físico alguno a los manipuladores de este, ni a los dispositivos utilizados para su uso. La interfaz está diseñada cuidadosamente y resulta agradable al entorno del usuario.

5.6 Conclusiones

Después de haber concluido este capítulo con el estudio de factibilidad realizado, queda demostrado, teniendo en cuenta el tiempo de desarrollo y el esfuerzo para desarrollar la aplicación propuesta conjuntamente con los beneficios intangibles organizacionales que están por encima de los costos, la viabilidad de la realización del proyecto en cuestión.

CONCLUSIONES

Con la elaboración de este trabajo se demostró la importancia y facilidades que brinda el sistema a desarrollar como medio informativo sobre la misión Barrio Adentro, haciendo uso de las tecnologías que se han provisto para ello. En este trabajo se hizo un análisis del desarrollo de aplicaciones multimedia en el mundo y de las metodologías y herramientas más usadas, determinando las más idóneas a emplear.

Después de un análisis de los objetivos y tareas planteado al inicio de este trabajo se llegó a las siguientes conclusiones:

- Se obtuvo una documentación resultante para ser consultada por los programadores a la hora de realizar la implementación del producto y en caso de que se le quieran agregar nuevos módulos o funcionalidades a la aplicación.
- El software *“Barrio Adentro al alcance de tus manos”* resume la información más relevante sobre la Misión Barrio Adentro y los servicios que se brindan como parte de la misma, sirviendo como fuente de información a toda la población venezolana.
- Se provee de un sistema informático dinámico e interactivo motivando la consulta de sus usuarios, además de contar con una interfaz amigable y sencilla logrando una adecuada orientación de su consultor.

RECOMENDACIONES

Después de haber concluido este trabajo y haber logrado dar cumplimiento al objetivo propuesto al inicio del mismo añadimos a continuación algunas sugerencias o recomendaciones que deben tenerse en cuenta para trabajos futuros:

- Transportar *“Barrio Adentro al alcance de tus manos”* a un funcionamiento en red, vinculándolo a otros sitios informativos.
- Usar *“Barrio Adentro al alcance de tus manos”* como ejemplo para la creación de otras aplicaciones informativas sobre disímiles temas.

REFERENCIAS BIBLIOGRÁFICAS

- ALBALAT, A.; V. HERRERO, *et al.* *Macromedia Flash MX*. Disponible en: <http://www.lsi.uji.es/asignatura/obtener.php?letra=5&codigo=44&fichero=1115202606544>
- ANÓNIMO. *Barrio Adentro*, 2005. [Disponible en: <http://www.cadela.gov.ve/descargas/MisionBarrioAdentro.pdf>]
- . *Bloque 3: Tecnologías de la información y de las comunicaciones*, 2006a. [Disponible en: http://www.educ.ar/educar/docentes/cbc/final.jsp?url=CBC_EGB117314%2F503.HTML&nivel=1&area=-1&tipo=&id=120662]
- . *Conceptos y definiciones de hipertexto*, 2000. [Disponible en: <http://www.ldc.usb.ve/~abianc/hipertexto.html>]
- . *Diagramas de Componentes*. Disponible en: <http://www.creangel.com/uml/componente.php>
- . *Herramienta multimedia contempla nuevas capacidades: MACROMEDIA LANZA DIRECTOR MX*, 2002. [Disponible en: <http://www.diariopyme.cl/newtenberg/1267/article-27352.html>]
- . *Hipermedia*, 2006b. [Disponible en: <http://es.wikipedia.org/wiki/Hipermedia>]
- . *Hipertexto*, 2007a. [Disponible en: <http://es.wikipedia.org/wiki/Hipertexto>]
- . *Lenguaje Unificado de Modelado* 2007b. [Disponible en: <http://es.wikipedia.org/wiki/UML>]
- . *Macromedia Authorware*, 2007c. [Disponible en: http://es.wikipedia.org/wiki/Macromedia_Authorware]
- . *Metodologías de Desarrollo de Software*. Disponible en: <http://alarcos.inf-cr.uclm.es/doc/ISOFTWAREI/Tema04.pdf>
- . *Modelo Vista Controlador*, 2007d. [Disponible en: http://es.wikipedia.org/wiki/Modelo_Vista_Controlador]
- . *Rational Unified Process*, 2006c. [Disponible en: http://es.wikipedia.org/wiki/Rational_Unified_Process]
- CUARESMA, M. J. E. *Metodologías para el desarrollo de sistemas de información global: análisis comparativo y propuesta*, 2001. [Disponible en: <http://www.lsi.us.es/docs/informes/EstadoActual.pdf>]
- . *Modelos y técnicas para la especificación y el análisis de la navegación en sistemas software. Tesis Doctoral*, 2004. [Disponible en: <http://www.lsi.us.es/docs/doctorado/tesis/tesis.pdf>]
- DÍAZ-ANTÓN, M. G. *PROPUESTA DE UNA METODOLOGÍA DE DESARROLLO DE SOFTWARE EDUCATIVO BAJO UN ENFOQUE DE CALIDAD SISTÉMICA*. Disponible en: <http://www.academia-interactiva.com/ise.pdf>

- DÍAZ, C. C. *LA TECNOLOGIA MULTIMEDIA: Una Nueva Tecnología de Comunicación e Información. Características, concepciones y aplicaciones.*, 1994. [Disponible en: <http://iteso.mx/~carlosc/pagina/documentos/multidef.htm>]
- ENGELS, G. and S. S. B. NEU. *Integrating Software Engineering and User-centred Design for Multimedia Software Developments.* Disponible en: <http://wwwcs.upb.de/cs/agengels/Papers/2003/EngelsSauerNeu-HCC03.pdf>
- ENRÍQUEZ, A. M. B. *El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML.* Disponible en: www.monografias.com/trabajos16/lenguaje-modelado-unificado/lenguaje-modelado-unificado.shtm
- GÓMEZ, H. D. B. *LA EVOLUCIÓN DEL SOFTWARE EDUCATIVO.* Disponible en: http://www.karisma.org.co/documentos/software/redp/LA_EVOLUCION_DEL_SOFTWARE_EDUCATIVO.doc
- HIPERTEXTO.INFO. *Modelo RMM,* 2006. [Disponible en: http://www.wikilearning.com/rmm_metodologia_de_administracion_de_relaciones_rmdm_modelo_de_datos_de_administracion_de_relaciones-wkccp-14327-1.htm]
- HUIDOBRO, J. M. *Tecnologías de información y comunicación* Disponible en: <http://www.monografias.com/trabajos37/tecnologias-comunicacion/tecnologias-comunicacion2.shtml>
- HUNG, R. A. *Soporte Hipermedia para el Mantenimiento, Reparación y Ensamblaje de Computadoras Personales a Nivel Básico.* Disponible en: <http://www.monografias.com/trabajos11/antep/antep.shtml>
- IZQUIERDO, L. V. and J. G. RAMOS. *GESTA virtual. Educación a Distancia en sistemas educativos,* 2007. [Disponible en: <http://www.abed.org.br/publicue/cgi/cgilua.exe/sys/start.htm?UserActiveTemplate=4abed&inford=154&sid=108>]
- LEIGHTON, B. S. and Y. F. LEIVA. *Una propuesta Metodológica para el Diseño de Interfaces y Mapas de Navegación en Aplicaciones Hipermediales,* 2002. [Disponible en: http://www.inf.udec.cl/revista/ediciones/edicion4/paper_metodologia.PDF]
- MENÉNDEZ, R. and B. ASENSIO. *Metodologías de desarrollo de software,* 2005. [Disponible en: http://www.wikilearning.com/metodologia_de_desarrollo_de_software-wkccp-3617-1.htm]
- MONTERO, Y. H. *Diseño Hipermedia centrado en el usuario,* 2002. [Disponible en: <http://www.nosolousabilidad.com/articulos/hipermedia.htm>]
- PERALTA, M. *ESTIMACIÓN DEL ESFUERZO BASADA EN CASOS DE USO.* Disponible en: <http://www.itba.edu.ar/capis/rtis/rtis-6-1/estimaci%F3n-del-esfuerzo-basada-en-casos-de-usos.pdf>
- PÉREZ, Y. M. *Conferencia 1: Introducción a la multimedia,* 2006. p.

- PUIG, C. T. I. *Del hipertexto al hipermedia. Una aproximación al desarrollo de las obras abiertas*. Disponible en: http://www.iaa.upf.es/formats/formats2/tom_e.htm
- REGAÑA, C. B. and L. M. T. BARZABAL. *Atención a la diversidad y multimedia: el diseño de materiales curriculares un reto al alcance de todos* 2004. [Disponible en: <http://www.monografias.com/trabajos903/atencion-diversidad-multimedia/atencion-diversidad-multimedia2.shtml>]
- RODRÍGUEZ, D. F. Z. *MULTIMEDIA*. Disponible en: <http://www.monografias.com/trabajos10/mmedia/mmedia.shtml>
- ROSARIO, J. *La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación*, 2005. [Disponible en: <http://www.cibersociedad.net/archivo/articulo.php?art=218>]
- SANCHEZ, M. A. M. *Metodologías De Desarrollo De Software*, 2004. [Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html]
- SANTOS, E. *Crea aplicaciones educativas en un entorno totalmente visual*, 2006. [Disponible en: <http://macromedia-authorware.softonic.com/ie/55052>]
- SAUER, S. and G. ENGELS. *Extending UML for Modeling of Multimedia Applications*. Disponible en: <http://www.itec.uni-klu.ac.at/~harald/proseminar02/sauer1.pdf>
- SAUTRAF. *Director MX y Lingo (Gross, Phil) > Opiniones > Es lo peor...* 2004. [Disponible en: http://www.ciao.es/Director_MX_y_Lingo_Gross_Phil_Opinion_799249]
- SOFTONIC, E. D. *Crea animaciones vectoriales para páginas Web*, 2006. [Disponible en: <http://macromedia-flash.softonic.com/ie/20878>]
- VILLA, L. *Flash ¿Cuál es el problema?*, 2003. [Disponible en: <http://www.w3c.org/TR/1999/REC-html401-19991224/loose.dtd>]

BIBLIOGRAFÍA

- ANÓNIMO. *Contenidos Educativos en Tecnología de Información y Comunicación - CETIC*, 2005. [Disponible en: <http://www.cetic.cnti.gob.ve/software2005.htm>]
- . *Ministerio de Energía y Petróleo – Misiones*, 2007. [Disponible en: <http://www.mem.gob.ve/misiones/index.php>]
- . *Misión Barrio Adentro*, 2006. [Disponible en: http://es.wikipedia.org/wiki/Misi%C3%B3n_Barrio_Adentro]
- CONTRERAS, D. J. *Barrio Adentro*, 2004. [Disponible en: http://www.gerenciasocial.org.ve/bsocial/bs_03/bs_03_pdf_point/jueves/barrio_adentro.pdf]
- JACOBSON, I.; G. BOOCH, *et al. El proceso unificado de desarrollo de software.*, Editorial Félix Varela, 2004. p.

GLOSARIO DE TÉRMINOS

ActionScript: es un lenguaje de programación orientado a objetos (OOP), utilizado en especial en aplicaciones Web animadas realizadas en el entorno Macromedia Flash.

Análisis (flujo de trabajo): flujo de trabajo fundamental cuyo propósito es analizar los requisitos descritos en la captura de requisitos, mediante su refinamiento y estructuración. Su objetivo es lograr una comprensión mas precisa de los requisitos y obtener una descripción de estos que sea fácil de mantener y que nos ayude a dar estructura al sistema en su conjunto, incluyendo la arquitectura.

Aplicación (sistema): sistema que ofrece a un usuario final un conjunto coherente de casos de usos.

Artefacto: pieza de información tangible que es creada, modificada y usada por los trabajadores al realizar actividades; representa un área de responsabilidad, y es candidata a ser tenida en cuenta para el control de la configuración. Un artefacto puede ser un modelo, un elemento de un modelo, o un documento.

Centrado en la arquitectura: en el contexto del ciclo de vida del software, significa que la arquitectura de un sistema se usa como un artefacto primordial para la conceptualización, construcción, gestión y evolución del sistema en desarrollo.

Componente: Es una parte física reemplazable de un sistema que empaqueta su implementación y es conforme a un conjunto de interfaces a las que proporciona su realización. Los componentes existentes en el dominio de la implementación son unidades físicas en los computadores que se pueden conectar con otros componentes, sustituir, trasladar, archivar, etc.

Cliente: persona, organización grupo de personas que encarga la construcción de un sistema, ya sea empezando desde cero, o mediante el refinamiento de versiones sucesivas.

Despliegue: ocurre cuando varios trabajos mas o menos independientes (flujos de control, procesos) se distribuyen entre diferentes dispositivos hardware.

Diagrama: representación gráfica de una colección de elementos de modelado. Un diagrama muestra representaciones de elementos semánticos del modelo, pero su significado no se ve afectado por la forma en que son representados.

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

Dirigido por casos de uso: en el contexto del ciclo de vida del software, indica que los casos de uso se utilizan como artefacto principal para definir el comportamiento deseado para el sistema, y para comunicar este compartimiento entre las personas involucradas en el sistema. También indica que los casos de uso son la entrada principal para el análisis, diseño, implementación y pruebas del sistema, incluyendo la creación, verificación y validación de la arquitectura del sistema.

Microsoft: Compañía de software más grande del mundo. Fue fundada en 1975 por Paul Allen y Bill Gates. Aunque también se conoce por sus lenguajes de programación y aplicaciones para computadores personales, el éxito sobresaliente de Microsoft se debe a sus sistemas operativos DOS y Windows.

Modelo: captura una vista de un sistema del mundo real. Es una abstracción de dicho sistema, considerando un cierto propósito. El modelo describe completamente aquellos aspectos del sistema que son relevantes a su propósito, y a un apropiado nivel de detalle.

Modelado visual: visualización de productos (artefactos) en esquemas o diagramas estandarizados.

Portabilidad: grado en el que un sistema, en funcionamiento en un determinado entorno de ejecución, puede ser convertido fácilmente en un sistema funcionando en otro entorno de ejecución.

Proceso de desarrollo de software: conjunto total de actividades necesarias para transformar los requisitos de un cliente en un conjunto consistente de artefactos que representan un producto software y, en un punto posterior en el tiempo, para transformar cambios en dichos requisitos en nuevas versiones del producto software.

Requisito: condición o capacidad que debe cumplir un sistema.

Requisito funcional: requisito que especifica una acción que debe ser capaz de realizar el sistema, sin considerar restricciones físicas; requisito que especifica comportamiento de entrada/salida de un sistema.

Requisito no funcional: requisito que especifica propiedades del sistema, como restricciones del entorno o de implementación, rendimiento, dependencias de la plataforma, mantenibilidad, extensibilidad o fiabilidad. Requisito que especifica restricciones físicas sobre un requisito funcional.