



*Universidad
de las
Ciencias Informáticas*

POO Interactivo Multimedia para el aprendizaje de la programación orientada a objetos.

Trabajo para optar por el título de Ingeniero Informático.

Autores: Roberlán Rodríguez Sánchez
Ernesto A. Quinta Castro

Tutor: Msc. Ernesto A. González Díaz

Agradecimientos

A Fabio y a Perdomo, por su preocupación y por apoyarnos.

A Jorge Luís Ortiz por su ayuda en diseño y su tiempo.

A Harold por sus cursos de programación.

A Mario González por sus ideas y su dedicación.

A Alexey y Yancy por sus aclaraciones.

A Osmany y a Noel por estar siempre ahí.

A Annilie por su paciencia.

A Marielis por su apoyo y paciencia.

A todos nuestros compañeros y compañeras de aula (y de año) por hacer mas fáciles todos estos años.

A nuestras familias por su apoyo y por confiar en nosotros.

A nuestra Revolución y nuestro Comandante por darnos todo para poder ser alguien en la vida.

A todos muchas gracias.

Índice

INTRODUCCIÓN.....	1
CAPÍTULO I.....	3
MULTIMEDIA, PRINCIPALES CONCEPTOS	3
1.1 Principales conceptos de multimedia.....	3
1.2 Tendencias de las aplicaciones multimedia.....	5
1.3 Tendencias pedagógicas en la aplicación de las multimedias.....	6
1.4 Ventajas y desventajas de las multimedias educativas.....	7
1.5 Requisitos para desarrollo de multimedia.....	10
1.6 Conclusiones.....	10
CAPÍTULO II.....	12
CARACTERÍSTICAS DEL SISTEMA.....	12
2.1 Objeto de automatización.....	12
2.2 Propuesta del sistema.....	12
2.3 Selección de la tecnología.....	14
2.3.1 Productos multimedia.....	14
2.3.2 ¿Qué es un programa o herramienta de autoría?.....	14
2.3.3 Tipo de aplicación a desarrollar y factores funcionales.....	15
2.3.4 Características principales de los programas de autoría.....	17
2.3.5 Tipos de programas de autoría.....	18
2.4 Programa seleccionado: Macromedia Flash.....	22
2.4.1 Información básica de Macromedia Flash 8.....	23
2.4.2 Sobre Flash.....	23
2.5 Metodologías de Desarrollo de Software.....	25
2.6 Conclusiones.....	30
CAPÍTULO III.....	32
DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	32

3.1	Introducción	32
3.2	Situación problémica.	32
3.3	Objeto de automatización.....	32
3.4	Modelo del sistema.....	32
3.4.1	Modelo de dominio.....	33
3.4.2	<i>Requisitos Funcionales</i>	33
3.4.3	Requisitos no Funcionales.....	34
3.4.4	Actores	36
3.4.5	Diagrama de Navegación	36
3.4.7	Diagrama de casos de uso del sistema.....	36
3.4.8	Descripción de los casos de uso.....	36
3.5	Diagrama de clases del modelo de objeto.....	41
3.6	Conclusiones.....	41
CAPITULO IV.....		42
CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.		42
4.1	Introducción.....	42
4.2	Patrones de diseño.....	42
4.1	Modelo análisis.....	42
4.1.1	Diagrama de presentación.....	42
4.4.2	Diagrama de Clase de Análisis.....	43
4.4.3	Diagrama de Clases de Diseño.....	43
4.4.4	Diagrama de Colaboración.....	43
4.4.5	Diagrama de Secuencia.....	44
4.5	Etapa de prueba.....	44
4.6	Modelo de implementación.....	44
4.6.1	Diagrama de componentes.....	44
4.7	Modelo de Prueba.....	44
4.7.1	Estrategia de pruebas.....	45
4.7.2	Pruebas de instalación.....	45
4.7.3	Pruebas de solidez.....	45
4.8	Conclusiones.....	45
CAPÍTULO V.....		46
ESTUDIO DE FACTIBILIDAD.....		46
5.1	Estimación de costo.....	46

5.2 Beneficios tangibles e intangibles.....	46
5.3 Conclusiones.....	47
RECOMENDACIONES.....	48
CONCLUSIONES.....	49
REFERENCIA BIBLIOGRÁFICA.....	50
ANEXO I.....	52
ANEXOS II.....	56
ANEXO III.....	81

Introducción.

Muy importante es el amplio uso de la tecnología de la informática y las comunicaciones en la docencia universitaria a partir de los cambios que se han venido produciendo en la enseñanza y aprendizaje, donde cada vez mas el estudio es protagonista de su propio aprendizaje.

En este entorno el amplio uso de las TIC en la docencia universitaria es esencial. Sin embargo en el desempeño de estos todavía existe múltiples problemas que van desde el inadecuado uso de la tecnología hasta la manera en que se ha interiorizado esto por los estudiantes.

Los resultados de esas insuficiencias de ven a continuación:

Cant. Estudiantes	Curso	Suspenso Mundial	A Mundial
2502	04-05	12	124
2386	03-04	48	193
2008	02-03	27	127

En el caso de la UCI los estudios y encuestas realizadas en esta dirección reportan lo siguiente:

- Problemas en la composición del claustro.
- Preparación y entrega de las clases con poco tiempo antes de impartirlas, lo cual atenta contra la calidad de la docencia.
- Insuficiente motivación y creatividad al impartir las clases por parte de profesores y alumnos ayudantes.
- Falta de comunicación con los estudiantes durante la clase y actividades vinculadas a la docencia por parte de algunos profesores.
- Insuficiente uso de los métodos, medios y técnicas de avanzada con que se cuenta.
- Insuficiente trabajo dirigido a la algoritmización en la asignatura.
- Alteraciones en los horarios, unido a la carga de los profesores.

- Falta de coordinación entre los diferentes factores para el desarrollo de las actividades. En ocasiones coinciden varias actividades.
- Mal funcionamiento de los sistemas en los Laboratorios en períodos durante el curso.
- No se efectúan todas las visitas a clases planificadas.

A partir de los problemas antes expresados la pregunta que se hace esta investigación es la siguiente:

¿Cómo producir un software multimedia dinámico, que motive el aprendizaje de P1 en los estudiantes de la UCI. ?

A partir de esta pregunta y de los problemas planteados, el objetivo de esta investigación consiste en elaborar un software para la enseñanza y aprendizaje de la asignatura P1 a través del empleo de la herramienta Macromedia Flash. Ello debe permitir dotar al estudiante y al profesor de habilidades interactivas para la enseñanza y el aprendizaje.

Aportes:

- Independiza al estudiante de los servidores de la universidad y contribuye a un empleo óptimo de la tecnología con que cuenta la UCI.
- Individualiza el aprendizaje.
- Permite un fácil mantenimiento de la aplicación así como de la información que se expone en ella.

Capítulo I

Multimedia, principales conceptos

Desde la década de los años 70 se comenzaron a dar los primeros pasos para la inserción de la computación en los procesos docentes. Con el de cursar de los años, se ha llegado a la conclusión de que el empleo de la computación en los procesos docentes de los distintos niveles de enseñanza, por si solo no resuelven todos los problemas.

Las tecnologías informáticas de última generación, pueden contribuir enormemente a solucionar un conjunto de problemas docentes que hoy en día existen, tales como, la escasez de bibliografía actualizada, la motivación de los estudiantes por una rama determinada del conocimiento, le ofrece al estudiante materiales didácticos de más calidad, así como la posibilidad de realizar cálculos con más rapidez y precisión, otro problema que nos pudiera ayudar a resolver la informática es la comunicación alumno profesor y entre los propios alumnos, puede contribuir a que tanto profesores y alumnos se mantengan actualizados, entre otros problemas. Ahora esto solo es posible si se utiliza adecuadamente, pues de no ser así puede crear el efecto inverso.

1.1 Principales conceptos de multimedia.

Si se habla de computación e informática no se puede dejar de hablar de software puesto que sin este la máquina no serviría de nada. Algunos desde el punto de vista de su función pueden tener la clasificación de software educativo.

Los software educativos los definimos como “Software diseñado y producido para la docencia o cualquier otro software que, aunque no haya sido diseñado con fines puramente educativos, se este empleando con el propósito de enseñar y aprender”.

El software educativo presenta varias clasificaciones pero en este capítulo solo abordaremos su clasificación según su función didáctica.

En este sentido abundan las clasificaciones, pero todas de una forma u otra consideran que existen:

- Tutoriales
- Entrenadores
- Simuladores
- Evaluadores

Los tutoriales consideramos que son los software educativos que cumplen con las funciones didácticas de motivar el estudiante, introducir y desarrollar la nueva materia.

En los tutoriales, la materia se le presenta a los estudiantes combinando imágenes, sonidos, videos y controlando el nivel de conocimientos alcanzado a través de preguntas, ejercicios. etc.

Algunos expertos plantean que tutorial es todo aquel software que suple al profesor pues tiene la función de presentar un conocimiento nuevo al estudiante. En esto no coincidimos plenamente, pues consideramos que por un lado es imposible sustituir al profesor, debido a que su función no es solamente presentar una información al estudiante para que este la convierta en conocimiento, sino que esta matizada de un componente afectivo que, “hasta ahora”, no hay máquina que capaz de suplirlo. Y por otro lado no solamente los software tienen la “capacidad” de presentar una información nueva al estudiante, los libros, las revistas, la televisión y otros medios, también pueden cumplir esta función.

Algunos autores plantean la existencia de tutoriales inteligentes, estos software realmente, parten de los fundamentos de la inteligencia artificial, que la definimos como la ciencia cibernética, que estudia los sistemas que simulan los procesos lógicos del pensamiento humano, esta ciencia tiene como bases la lógica matemática, la psicología cognitiva, la gramática, y las ciencias de la computación.

Con respecto a los entrenadores se puede plantear que su función didáctica es la ejercitar lo aprendido por el alumno, también conocido como repasador. Las simulaciones son programas que intentan dar una representación lo más realista posible de un dominio complejo. Este tipo de software educativo intenta explicar un fenómeno mediante una representación controlada del mismo en la computadora con apariencia lo más cercana posible al fenómeno real. Los programas de simulación, como su nombre lo indica, reproducen en forma simplificada y a través de un modelo una situación o fenómeno

(físico, químico, matemático, social), permitiéndole al estudiante explorar y actuar sobre dichos eventos. Los evaluadores tienen la función de, como bien su nombre lo indica evaluar al estudiante. Esto lo puede hacer de diferentes maneras, mediante distintos tipos de preguntas, de enlace, de verdadero o falso, de selección, de completamiento de frases o párrafos, etc. En el caso del software dirigido a los alumnos primarios esta evaluación se puede realizar mediante un juego.

La tendencia actual, en la producción de software educativos, en Cuba, no esta orientada ha hacer software puros tutoriales, simuladores o evaluadores, sino a producir software que lo integren todo, es decir que presente el nuevo contenido, que lo ejercite, lo evalúe y si tiene que simular o modelar, también lo haga.

Nuestra aplicación tampoco fue una excepción, esta desarrollada principalmente como tutorial, aunque también soporta simulaciones propias de la asignatura P1 para la cual fue diseñada.

1.2 Tendencias de las aplicaciones multimedia.

El término multimedia se refiere a una integración o agrupación de diferentes medios audiovisuales. Pero la acepción actual más habitual del mismo es la que hace referencia a aquellos programas que se desarrollan a través del ordenador, de tal modo que todo el sistema multimedia se apoya en un solo soporte. La multimedia se convierte así en un entorno de aprendizaje que combina las posibilidades educativas que ofrecen diferentes medios de comunicación interconectados y controlados a través de un ordenador. Se comienza a hablar entonces de la informática multimedia como un intento de combinar la capacidad autoexplicativa de los medios audiovisuales con el texto y fotografías, para crear un nuevo medio de comunicación único en la pantalla del ordenador.

Estamos pasando de una era industrial a una era donde la supremacía de individuos, organizaciones y naciones pasa por el manejo inteligente de la información. Por lo que se refiere a las tecnologías de los sistemas de información, entendiendo como tales las tecnologías de ordenadores, telecomunicaciones y automatización de oficinas, cabe decir que los sistemas multimedia han impulsado el desarrollo de las tecnologías de la información y de la comunicación. Gracias a las tecnologías de la información la multimedia ha hecho posible superar la idea de la información contenida en un texto para

introducimos en el campo de la comunicación audiovisual, de la transmisión de sensaciones y de innumerables novedades más. En cuanto a las tecnologías de la comunicación, se ha pasado de la tradicional división entre diferentes medios que compiten por un espacio educativo a un nudo de conexión único. Ahora el alumno no necesita desplazarse físicamente para consultar un texto, puesto que a través de su ordenador podrá acceder a cualquier clase de información, sin importar el lugar en que esté.

La característica principal de los sistemas multimedia es su gran flexibilidad así como la alta interactividad que poseen, pues permiten un aprendizaje autoguiado y autoiniciado, en el cual cada persona va construyendo su conocimiento, bien sea de manera individual o colectiva.

Entre los múltiples sistemas multimedia, destaca el hipermedia, basado en hipertexto. Debido a que el hipertexto no es un sistema cerrado permite que el formador sea el que decida en cada momento a qué tipo de información desea acceder. El alumno podrá construir sus propios caminos de lectura saltando aquellos que no considere de su interés. El hipertexto rompe la idea tradicional de que el formador es el depositario de toda la información que ha de recibir el alumno, el cual tendría como misión tratar de aprender lo más fielmente posible todo lo que aquél le transmitiera.

1.3 Tendencias pedagógicas en la aplicación de las multimedias.

En el campo de la docencia, las transformaciones tecnológicas podrían llegar a imponer el reto, la necesidad y sobre todo la posibilidad de renovar las técnicas de enseñanza y el tipo de material docente que se pone a disposición de los estudiantes y maestros. Las condiciones actuales facilitan contar con herramientas de apoyo al proceso educativo de una manera mas dinámica, llena de estímulos paralelos, preparados para el cambio constante, intercomunicados e integrados. En este caso, la multimedia representa una ventaja como soporte al proceso educativo, pues presenta y manipula la información en un lenguaje contemporáneo, que además permite a maestros y alumnos jugar con su estructura para lograr diferentes objetivos pedagógicos.

Procesos educativos complementados por diferentes canales configuran una dinámica diferente en donde el conocimiento es apropiado de una manera mas dinámica por la

comunidad. El computador en el salón de clase se convierte en un recurso que pone la multimedia al alcance de la vida diaria como herramienta para producirla, compartirla y explorarla. A través de un mismo canal textos y material audiovisual se combinan generando búsqueda, manipulación, comparación y otra gran cantidad de posibilidades que proveen de nuevas posibilidades para aprender y explorar temas y contextos no solo a nivel individual sino con el apoyo y colaboración de compañeros y maestros.

1.4 Ventajas y desventajas de las multimedias educativas.

Sin duda el uso de estos atractivos e interactivos materiales multimedia (especialmente con una buena orientación y combinados con otros recursos: libros, periódicos...) puede favorecer los procesos de enseñanza y aprendizaje grupales e individuales. Algunas de sus principales aportaciones son las siguientes:

- **Proporcionar información.** En los CD-ROM o al acceder a bases de datos a través de Internet pueden proporcionar todo tipo de información multimedia e hipertextual

- **Avivar el interés.** Los alumnos suelen estar muy motivados al utilizar estos materiales, y la motivación (el querer) es uno de los motores del aprendizaje, ya que incita a la actividad y al pensamiento. Por otro lado, la motivación hace que los estudiantes dediquen más tiempo a trabajar y, por tanto, es probable que aprendan más.

- **Mantener una continua actividad intelectual.** Los estudiantes están permanentemente activos al interactuar con el ordenador y mantienen un alto grado de implicación e iniciativa en el trabajo. La versatilidad e interactividad del ordenador y la posibilidad de "dialogar" con él, les atrae y mantiene su atención.

- **Orientar aprendizajes** a través de entornos de aprendizaje, que pueden incluir buenos gráficos dinámicos, simulaciones, herramientas para el proceso de la información... que guíen a los estudiantes y favorezcan la comprensión.

- **Promover un aprendizaje a partir de los errores.** El "feed back" inmediato a las respuestas y a las acciones de los usuarios permite a los estudiantes conocer sus errores justo en el momento en que se producen y generalmente el programa les ofrece la oportunidad de ensayar nuevas respuestas o formas de actuar para superarlos.

- **Facilitar la evaluación y control.** Al facilitar la práctica sistemática de algunos temas mediante ejercicios de refuerzo sobre técnicas instrumentales, presentación de conocimientos generales, prácticas sistemáticas de ortografía..., liberan al profesor de trabajos repetitivos, monótonos y rutinarios, de manera que se puede dedicar más a estimular el desarrollo de las facultades cognitivas superiores de los alumnos.

- **Posibilitar un trabajo Individual y también en grupo,** ya que pueden adaptarse a sus conocimientos previos y a su ritmo de trabajo (por ello resultan muy útiles para realizar actividades complementarias y de recuperación en las que los estudiantes pueden autocontrolar su trabajo) y también facilitan el compartir información y la comunicación entre los miembros de un grupo.

Además de las ventajas que pueden proporcionar, también deben considerarse sus potenciales inconvenientes (superficialidad, estrategias de mínimo esfuerzo, distracciones...) y poner medios para soslayarlos.

A continuación se presenta un estudio más detallado de estas ventajas e inconvenientes potenciales de los materiales educativos multimedia:

VENTAJAS	INCONVENIENTES
<p>Interés. Motivación. Los alumnos están muy motivados y la motivación (el querer) es uno de los motores del aprendizaje, ya que incita a la actividad y al pensamiento. Por otro lado, la motivación hace que los estudiantes dediquen más tiempo a trabajar y, por tanto, es probable que aprendan más.</p>	<p>Adicción. El multimedia interactivo resulta motivador, pero un exceso de motivación puede provocar adicción. El profesorado deberá estar atento ante alumnos que muestren una adicción desmesurada.</p> <p>Distracción. Los alumnos a veces se dedican a jugar en vez de trabajar</p>
<p>Interacción. Continua actividad intelectual. Los estudiantes están permanentemente activos al interactuar con el ordenador y mantienen un alto grado de implicación en el trabajo. La versatilidad e interactividad del ordenador y la posibilidad de "dialogar" con él, les atrae y mantiene su atención.</p>	<p>Ansiedad. La continua interacción ante el ordenador puede provocar ansiedad en los estudiantes.</p>
<p>Desarrollo de la iniciativa. La constante participación por parte de los alumnos propicia el desarrollo de su iniciativa ya que se ven obligados a tomar continuamente nuevas decisiones ante</p>	<p>Aprendizajes incompletos y superficiales. La libre interacción de los alumnos con estos materiales (no siempre de calidad) a menudo proporciona aprendizajes incompletos</p>

las respuestas del ordenador a sus acciones.	con visiones de la realidad simplista y poco profunda. La calidad de los aprendizajes generalmente no es mayor que utilizando otros medios.
Contacto con las nuevas tecnologías y el lenguaje audiovisual. Estos materiales proporcionan a los alumnos y a los profesores un contacto con las TIC, generador de experiencias y aprendizajes. Contribuyen a facilitar la necesaria alfabetización informática y audiovisual	Diálogos muy rígidos. Los materiales didácticos exigen la formalización previa de la materia que se pretende enseñar y que el autor haya previsto los caminos y diálogos que los alumnos seguirán en su proceso de descubrimiento de la materia. El diálogo profesor-alumno es más abierto y rico
Proporcionan información. En los CD-ROM o al acceder a bases de datos a través de Internet pueden proporcionar todo tipo de información multimedia e hipertextual.	Desorientación informativa. Muchos estudiantes se pierden en los hipertextos y la atomización de la información les dificulta obtener visiones globales. Los materiales hipertextuales muchas veces resultan difíciles de imprimir (están muy troceados)
Proporcionan entornos de aprendizaje e instrumentos para el proceso de la información, incluyendo buenos gráficos dinámicos, simulaciones, entornos heurísticos de aprendizaje.	Aislamiento. Los materiales didácticos multimedia permiten al alumno aprender solo, hasta le animan a hacerlo, pero este trabajo individual, en exceso, puede acarrear problemas de sociabilidad
En la Enseñanza a distancia la posibilidad de que los alumnos trabajen ante su ordenador con materiales interactivos de autoaprendizaje proporciona una gran flexibilidad en los horarios de estudio y una descentralización geográfica de la formación.	Dependencia de los demás. El trabajo en grupo también tiene sus inconvenientes. En general conviene hacer grupos estables (donde los alumnos ya se conozcan) pero flexibles (para ir variando) y no conviene que los grupos sean numerosos, ya que algunos estudiantes se podrían convertir en espectadores de los trabajos de los otros.
En Educación Especial es uno de los campos donde el uso del ordenador en general, proporciona mayores ventajas. Muchas formas de disminución física y psíquica limitan las posibilidades de comunicación y el acceso a la información; en muchos de estos casos el ordenador, con periféricos especiales, puede abrir caminos alternativos que resuelvan estas limitaciones.	Cansancio visual y otros problemas físicos. Un exceso de tiempo trabajando ante el ordenador o malas posturas pueden provocar diversas dolencias.

Estos materiales didácticos informáticos constituyen un recurso formativo complementario que debe utilizarse de la manera adecuada y en los momentos oportunos.

1.5 Requisitos para desarrollo de multimedia.

Todos los elementos usados en multimedia han existido antes. Las multimedia interactivas tejen cinco tipos de medios: texto, sonidos, gráficos, vídeo y animación. La producción final de las multimedia llega a ser interactiva solamente cuando estos pedazos distintos se ponen juntos en una manera organizada. Las multimedia interactivas no siguen un patrón linear. El lector no tiene que comenzar en la tapa y leer al fondo, pueden moverse alrededor dentro de la información que el autor ha presentado. El autor presenta ideas y proporciona los acoplamientos para que sus lectores naveguen.

El desarrollo de las multimedia se puede describir como el uso de muchos diversos medios de comunicación, tecnologías y plataformas de la entrega. Una producción de multimedia puede ser costosa. Su preparación requiere un equipo de personal especializado, cada uno con un sistema especial de habilidad. Este personal es el encargado de realizar un estudio detallado que permitan conocer cómo la gente aprende y cómo los materiales deben ser estructurados y coordinar el proceso del desarrollo. Que ayuden al usuario para el cual esta dirigido la multimedia a captar la mayor cantidad de información posible.

1.6 Conclusiones.

La Universidad de las Ciencias Informáticas surge con el objetivo de formar profesionales altamente calificados en la esfera de la informática y así fomentar el desarrollo de software en Cuba, con la máxima aspiración de convertir esta industria en la segunda entrada de capital extranjero al país.

Una de las mayores aplicaciones de la informática en el sector educacional es la producción de software educativo. Dado el elevado desarrollo y prestigio alcanzado en nuestro país por este sector en los años posteriores a la Revolución, se traza como una de las líneas estratégicas su producción y explotación tanto para el consumo nacional y la ayuda internacional, como para la exportación y obtención inmediata de beneficios económicos.

En nuestro caso nuestro trabajo trata de buscar soluciones y brindar nuevas vías para el mejoramiento de proceso docente educativo en nuestra universidad, incentivando a los estudiantes al estudio y preparación en la materia de programación orientada a objetos.

Capítulo II

Características del Sistema.

La idea de crear una herramienta para la asignatura P1 surge producto de una investigación que se realizó, donde los resultados arrojaron que desde los inicios de nuestro centro se han venido presentando deficiencias que se han visto reflejados en la cantidad de estudiantes que han tenido que ir a exámenes extraordinarios o mundiales y cuando arriban al segundo año los grandes problemas que presentan en la algoritmización, entre otras. Esto ha traído como consecuencia que el Departamento de Técnicas de Programación haya puesto en práctica la estrategia de repetir los exámenes parciales con el objetivo de aminorar la cantidad de estudiantes con resultados deficientes en la asignatura.

A raíz de esto surgió la idea de crear una multimedia que ayudara al proceso docente de esta asignatura con la capacidad de reunir métodos más interactivos y agradables para el estudiante y de esta forma motivar el estudio de P1.

2.1 Objeto de automatización.

La multimedia implementada pretende automatizar de manera parcialmente dinámica todos los métodos usados en la docencia en P1, acorde a los temas que se imparten, teleclases, conferencias, ejercicios y términos propios de la asignatura. Todo esto es englobado para así no depender de la red para que en caso de que esta falle no se vea afectado el proceso de aprendizaje de los estudiantes.

2.2 Propuesta del sistema.

El sistema se ideó para hacerlo de una forma sencilla y didáctica, donde el estudiante no se viera perdido al navegar dentro de la multimedia.

En el se deben exponer 4 menús que presentan los siguientes temas:

- Conferencias
- Ejercicios
- Glosario
- Biblioteca

En el módulo Conferencia se exponen todos los contenidos que se imparten durante el semestre. Esto se carga de forma dinámica desde ficheros texto que presentan en su interior un formato de HTML. Esta estructura dentro del fichero nos permite que las conferencias expuestas no sean tan estáticas. Dentro del HTML incluimos hipervínculos que se ven en el texto como palabras calientes con 2 tipos de funciones, una para cargar fotos que le brindará al sistema como dato el número de la foto al cargar y este lo hará de forma dinámica ya que la foto debe adecuar su nombre a determinada estructura (ej. imag 01.jpg) y localizarse en un directorio llamado imágenes. La otra función nos permite visualizar animaciones hechas en flash en un visor que nos permitirá correr o pausar nuestra animación según sea nuestro gusto. La función encargada de llamarla también nos brindara como dato el número de la animación que se debe visualizar, este debe tener también un determinado nombre acorde con el número que devuelve la función (ej. Animación 01.swf) y estar en una carpeta específica llamada animaciones.

El módulo Ejercicios nos expone ejercicios orientados en las clases prácticas. Aquí se utilizará el mismo formato de fichero pero no contendrá palabras calientes. Se expondrá en forma que cada botón hará referencia a una conferencia.

En el Glosario expondremos en un solo documento con igual formato todos los términos que el profesor crea pertinente que el estudiante debe conocer y dominar.

La biblioteca contendrá una galería de imágenes con las imágenes ya expuestas u otras más si se desea. También aparecerán aquí las teleclases. En un principio se pensó poner estas teleclases en formato swf para reducir el tamaño de estas, pero surgió el problema de que deben ser cargadas en memoria y debido a su tamaño en ocasiones se volvía bastante pesada su reproducción. Entonces se optó por usar el formato flv de Flash que es un poco mas grande que este pero es muy rápido para cargar su contenido y la pérdida de calidad durante la reproducción es mínima.

También se le agregaron unos botones de servicio para imprimir y otro para oír una melodía de fondo. Con el botón de imprimir se imprime el contenido de una conferencia, ejercicio o término del glosario que se esté exponiendo en ese momento. El de la música activará una melodía de fondo solo si desea porque es psicopedagógicamente incorrecto forzar al estudiante a oír una melodía cuando estudia si esta no es de su gusto.

2.3 Selección de la tecnología

2.3.1 Productos multimedia

Los productos multimedia son materiales que pueden incluir niveles de interacción sumamente complejos; la integración de audio, video, simuladores y elementos gráficos con alto grado de resolución, permiten la creación de materiales ricos y atractivos.

Cuando se opta por este medio electrónico, uno sabe de antemano que se está dirigiendo a un grupo de personas que cuenta con ciertos recursos particulares, Inclusive, es importante informar al usuario sobre los requisitos de equipo necesarios para la visualización del producto.

La visualización de un producto multimedia no requiere de la conexión a ningún servidor ni la instalación de programas adicionales para el funcionamiento correcto del software, todas las herramientas se encuentran integradas en el mismo disco compacto, el cual, en algunos casos, tiene un programa ejecutable que deberá de instalarse en la computadora para después poder correr el programa. Otros productos multimedia ni siquiera requieren de este proceso de instalación y ejecutan directamente el programa interactivo.

Sin embargo, la actualización de la información no se puede dar de manera inmediata, requiere de un mayor trabajo en equipo, el cual concluye con la quema de un disco.

Por lo tanto, el adoptar una herramienta de autoría adecuada dependerá mucho de la habilidad en el manejo y administración de los recursos y medios audiovisuales, y del gusto personal por el esquema de trabajo presentado por cada uno de estos programas.

2.3.2 ¿Qué es un programa o herramienta de autoría?

Los programas de autoría brindan un marco de trabajo para organizar e integrar los elementos visuales y sonoros de un proyecto multimedia; incorporan en una estructura funcional la interfaz del usuario y lo que es más importante, el diseño de las funciones y herramientas que estarán disponibles, lo cual medirá, en cierta forma, el grado de interacción de la multimedia.

Estas características son el resultado del diseño del programa del cual también se derivan las ligas de navegación y las ligas hipertextuales que darán acceso a la información en diferentes niveles de búsqueda e intención.

2.3.3 Tipo de aplicación a desarrollar y factores funcionales.

Muchos desarrolladores de sistemas multimedia sugieren una regla general para la selección del software a utilizar, ésta se encuentra representada en la tabla 2.

Tabla.2

Paquete	Aplicación
Flash, mTropolis	Presentaciones lineales con una gran cantidad de imágenes o sincronización de audio y video.
Toolbook	Entrenamiento interactivo por computadora, programas de evaluación y diagnóstico.
Delphi	Kioskos o demos
Visual Basic	Plataformas de pruebas de utilidad.
C++	juegos

No existen reglas predeterminadas para decidir el paquete idóneo, no obstante se pueden tomar en cuenta algunos factores funcionales que ayudarán a seleccionar el software de autoría; estos se listan en la tabla siguiente:

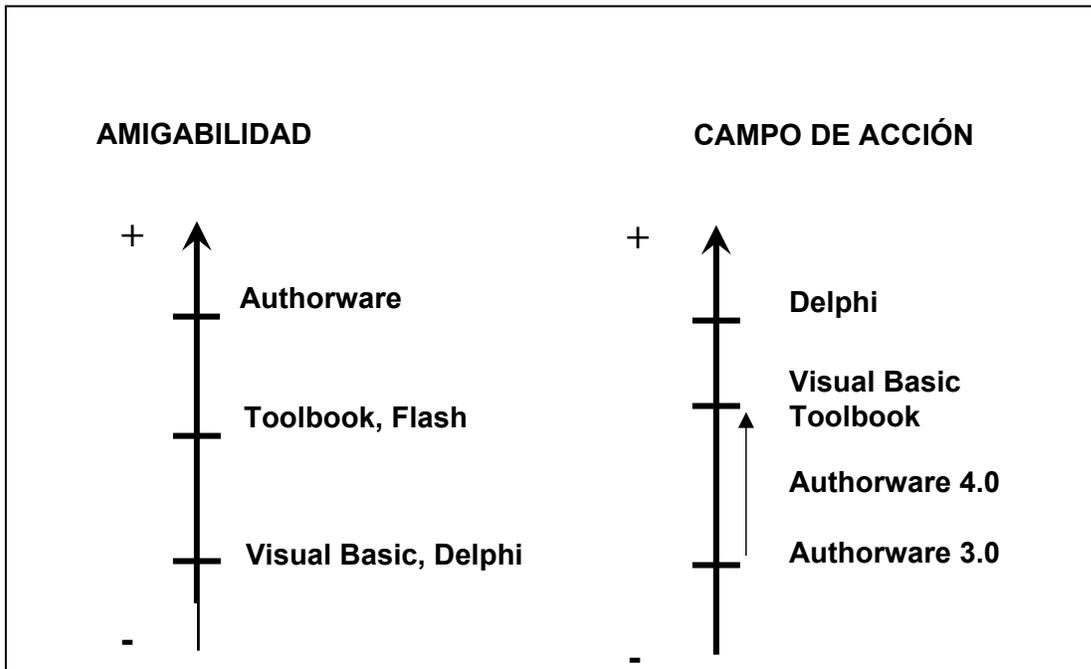
Tabla.3

Factores funcionales
Manejo de hipertextos.
Consulta a bases de datos.
Empleo de medios visuales y sonoros (imágenes, audio, animaciones, video).
Creación y control de ventanas "viewers".
Grado de interacción usuario-sistema deseado.
Capacidad de implementar rutinas específicas (soporte de algún lenguaje de programación: Lingo, OpenScript, entre otros).
Características del equipo de reproducción y sistema operativo a usar (especificar la plataforma de reproducción: PC, IBM, Macintosh).
Tiempote desarrollo (facilidad de uso).

Así mismo, hay que tomar en cuenta los requerimientos de cómputo mínimos que exigen los paquetes de autoría para trabajar y los formatos gráficos que reconocen, ya que en algunas ocasiones, resulta de gran ayuda conocer estos datos al momento de elegir el programa.

En la figura 1 se muestra la comparación entre diferentes herramientas de autoría con base en la facilidad y rapidez para desarrollar las aplicaciones y en relación con su campo de acción.

Figura1. Comparación de software de autoría.



El término amigabilidad se refiere a la facilidad que brinda el paquete para aprender sus funciones y utilizar las herramientas que posee para estar en condiciones de producir un título multimedia. Como puede observarse, en la figura anterior (figura 1), Authorware tiene la interfaz programa-desarrollador más sencilla de aprender, sin embargo hay quienes diferirán en este punto.

También se puede observar en la figura 1, que el campo de acción aumenta conforme la amigabilidad disminuye. El campo de acción se refiere a la capacidad que tiene el paquete

para desarrollar rutinas o acciones complejas no contempladas de origen (manejo de memoria, control de dispositivos externos y optimización de los recursos de la computadora, por mencionar algunos).

2.3.4 Características principales de los programas de autoría.

- ***Edición***

La mayoría de los programas de autoría incluyen dentro de su utilería, herramientas para la edición de imágenes fijas y de texto, no obstante están orientados a la integración de medios, por ello, sus funciones están limitadas a la simple manipulación de la imagen, o a introducir texto sencillo con ciertas características y restringido al conjunto de fuentes utilizadas por Windows. Si se desea diseñar una imagen con buena calidad se debe utilizar software dedicado e instalar fuentes adicionales que permitan ampliar el conjunto de tipografías disponibles.

- ***Programación***

Esta característica difiere de un programa de autoría a otro, pues esta basada en el tipo de herramienta usada; por el momento se mencionara que puede ser a través de las siguientes alternativas:

a) *Programación visual con íconos.* Es quizás la manera más sencilla para el proceso de autoría y la programación de eventos, ejemplos de estos programas son Action, Authorware e IconAuthor. Authorware cuenta además con un conjunto de funciones que le da mayor flexibilidad en la programación, aunque no a un nivel como el que presenta el grupo siguiente.

b) *Programación con un lenguaje de scripts.* Este nivel de programación es más poderoso que el anterior, ya que se amplía el campo de acción del paquete. Hypercard, SuperCard, Flash y Toolbook poseen esta característica.

Es recomendable que el paquete proporcione herramientas tales como un buen *debugger* (detector de errores lógicos y de sintaxis en los programas), un editor de texto robusto para la programación de los scripts y una referencia sintáctica en línea.

c) *Programación con lenguajes tradicionales, como C o Pascal.* En proyectos complejos frecuentemente se necesita programar extensiones al lenguaje de scripts para acceder directamente al sistema operativo de la computadora; estas extensiones pueden ser comandos o funciones externas al Macintosh (XCMDs o XFCNs), y llamadas a DLLs (Dynamic Link Libraries) o Windows MCIs (Windows Media Control Interface) en una PC.

- ***Pruebas de autor***

Cuando se desarrolla una aplicación multimedia es necesario verificar cómo se ve y si funciona la aplicación en la “vista” del usuario final. Generalmente se tienen disponibles dos “vistas”, la del autor y la del usuario final. Lo cual es de gran ayuda, pues se pueden corregir a tiempo eventos y errores no deseados.

- ***Distribución***

Para la distribución del sistema multimedia es necesario realizar una versión ejecutable del programa de autor, llamada *run-time*. La versión de run-time posibilita que el usuario final ejecute la aplicación sin tener el paquete de autoría completo instalado en su computadora, lo cual le ahorra espacio en su disco duro. Esta versión del programa no le permite al usuario realizar cambios al contenido, estructura o programación del proyecto, por lo que se garantiza la integridad del sistema y los derechos del autor.

2.3.5 Tipos de programas de autoría.

- ***Basados en paginas o tarjetas***

Los sistemas de autoría basados en páginas proporcionan una manera fácil de organizar los elementos de una multimedia, mediante una metáfora realizada con las páginas de un libro.

Toolbook, HyperCard, Plus y Visual Basic son ejemplos de este tipo de ambiente de trabajo.

Este tipo de paquetes están basados en objetos; donde los botones, los campos de texto, las imágenes, el background, las páginas o tarjetas, inclusive el proyecto/libro son objetos. Cada objeto tiene propiedades, algunas de las cuales pueden cambiar programando ciertas rutinas; además cada objeto puede tener asociado su propio script, ejecutado cuando este se afecta a través de un evento, como puede ser un clic de ratón.

Cuando sucede un evento, el programa envía un mensaje interno que sirve para programar cierta respuesta, el cual pasa a través de la jerarquía de objetos presentes en el proyecto/libro. La prioridad que tengan objetos dentro del proyecto/libro dependerá de cada paquete.

A la par que el mensaje viaja a lo largo de estos niveles jerárquicos, busca manejadores (handlers) en el script de cada objeto; si encuentra alguno que tenga el nombre especificado dentro del mensaje, entonces el sistema de autoría ejecuta la tarea descrita por el *handler*.

Las figuras que aparecen a continuación muestran los ambientes de trabajo de algunos de los paquetes existentes dentro de esta categoría (ver figura 1).

En forma general y de manera personal, cabe mencionar que Toolbook es el punto medio de los tres tipos de programas de autoría integrados en este apartado, pues posee una interfaz programa-desarrollador más amigable que la que presenta Visual Basic, permite un mayor control de los elementos en relación con Authorware. La única desventaja sería que no permite tener una vista global de la estructura del programa, por este motivo se deberá ser muy cuidadoso y organizado en este sentido. Para equilibrar esta desventaja, a partir de la versión 4.0, Toolbook cuenta con un visualizador de objetos, el Object Browser, el cual permite observar, mediante una ventana deslizable, las páginas, los objetos y sus propiedades, y la relación que guardan entre sí.

En el caso de Visual Basic (VB), la información se organiza en “formas”, y es en ellas donde se colocan los objetos visuales y sonoros. Cada objeto posee propiedades que pueden ser actualizadas mediante programación. Algunas rutinas o acciones que tienen como origen algunos paquetes de autoría deben ser implementadas en VB, esta situación

aunque es una desventaja, en algunas ocasiones permite tener un mayor control de los objetos y los eventos asociados a ellos (ver figura 2).

- ***Basado en íconos***

Las herramientas basadas en íconos proporcionan una metáfora de trabajo parecida al concepto de Diagrama de Flujo en computación, donde, a través de una programación visual de eventos se organizan e integran los elementos de la multimedia (ver figura 3).

El primer paso consiste en construir la estructura o diagrama de eventos, arrastrando a la línea de flujo el ícono correspondiente (ver ejemplos de íconos en la figura 4).

Cada ícono posee propiedades y puede incluir un menú de opciones, imágenes y sonidos. Como se observa en la figura 7, el diagrama de flujo describe gráficamente la estructura lógica del proyecto; ejemplos de este tipo de paquetes son Authorware, IconAuthor y HSC Interactive.

Con Authorware se pueden realizar aplicaciones sofisticadas sin tener que programar, solamente colocando los íconos en la línea de flujo se puede efectuar una secuencia de eventos y actividades, donde se incluyan interacciones con el usuario y toma de decisiones.

Además, Authorware es una herramienta práctica para hacer “storyboards” porque permite cambiar fácilmente una secuencia, aumentar opciones o reestructurar interacciones arrastrando o borrando íconos; también posee más de 200 variables de sistema y funciones para capturar, manipular y desplegar datos. Si se desea, se pueden programar pequeñas rutinas abriendo una ventana de cálculo en el lugar donde se deba ejecutar la rutina e introducir las funciones y variables que realizarán las tareas especificadas por el programador.

- ***Basada en el tiempo***

Las herramientas basadas en el tiempo se vuelven cada vez más comunes dentro de los programas de autoría. Cada una de ellas usa su propia interfaz de usuario para manejar los eventos en el tiempo; muchas utilizan una línea de tiempo para darle secuencia a los

eventos que suceden durante una presentación multimedia y con frecuencia despliegan niveles de elementos de esta, ó eventos a lo largo de una escala con incrementos altamente precisos; otros organizan largas secuencias de cuadros gráficos y adicionan el factor de tiempo ajustando cuadro a cuadro la ejecución de la aplicación. Action, Animation Works Interactive y Flash son ejemplos representativos de este rubro.

Al igual que Authorware, Flash es un programa desarrollado por Macromedia, cuya orientación principal es la producción de animaciones/videos y la sincronización de audio con animación; sin embargo, posee un lenguaje de programación llamado Action Script, que extiende su campo de acción y lo vuelve un programa poderoso para la producción de aplicaciones interactivas multimedia, y aunque su curva de aprendizaje es significativamente alta, una vez dominado el paquete, se coloca dentro de las herramientas mas potentes para el desarrollo de multimedias.

Flash integra los elementos de la multimedia mediante dos ventanas de trabajo, la Biblioteca y el Escenario. La Biblioteca es una base de datos visual que contiene las imágenes, el audio, el video y demás medios que se desean incluir en la aplicación (ventana derecha de la figura 9). El Escenario permite unir todos los elementos de la Biblioteca en una línea de tiempo. Esta ventana se encuentra localizada en la parte centro inferior de la figura 9. Además, Flash tiene la capacidad de exportar la aplicación final a un archivo de video .MOV, si es lo que se desea, sino puede crear un programa .EXE para que el usuario pueda consultarlo desde su computadora (ver figura 5).

A modo de conclusión, como se mencionó anteriormente, es necesario que independientemente del software de autoría elegido, se tenga la opción de crear una versión ejecutable de la aplicación final. Esta tarea será fácil o un poco más elaborada dependiendo de la herramienta que se utilice.

En el caso de Flash, la aplicación final es un programa ejecutable (.EXE) que solo requiere, si es el caso, que los archivos de video con extensión .MOV solo requiere los drivers propios de QuickTime. De esta manera es difícil obtener el programa fuente a partir del programa ejecutable, lo que permite proteger la autoría de la persona o institución que lo desarrolló, lo cual es una gran ventaja.

Toolbook es un caso diferente, debido a que el programa final aunque tiene extensión .EXE en realidad es el programa fuente con un pequeño encabezado; esta característica permite ejecutar la aplicación haciendo uso del run-time de Toolbook, lo que posibilita al usuario consultar el sistema multimedia sin necesidad de que tenga el programa de autoría completo; alternativamente permite colocar un passWord para evitar que el programa sea alterado por personas ajenas a los autores.

2.4 Programa seleccionado: Macromedia Flash

En nuestro caso, hemos seleccionado el Flash ya que es en la actualidad la herramienta más extendida para la producción de sistemas multimedia en el mundo profesional. Sus diferentes versiones, la facilidad de manejo y la inclusión de gran número de herramientas de producción son las bases fundamentales con las que cuenta este programa. Sus principales ventajas sobre los demás programas de autoría son:

1. La programación con un lenguaje de script, la cual presenta un nivel de programación poderoso, incluso, mas poderoso que el de programación visual por iconos, ya que amplía el campo de acción del paquete.
2. Macromedia Flash 8 presenta niveles de integración nunca antes vistos con la familia de aplicaciones creadas por Macromedia. Ahora, Director y la amplia gama de herramientas de desarrollo del programa Macromedia Director MX 2004 pueden combinar las fuerzas de este poderoso producto, para así construir presentaciones interactivas altamente efectivas, aplicaciones, juegos y medios de instrucción. También Flash 8 nos permite añadir hipervínculos a plataformas Web, contenido para aplicaciones tridimensionales y aplicaciones simples, tanto en Windows como en Macintosh.
3. Sus herramientas están basadas en el tiempo. Cada una de ellas usa su propia interfaz de usuario para manejar los eventos en el tiempo; muchas utilizan una línea de tiempo para darle secuencia a los eventos que suceden durante una presentación multimedia y con frecuencia despliegan niveles de elementos de multimedia o eventos a lo largo de una escala con incrementos altamente precisos.

2.4.1 Información básica de Macromedia Flash 8.

Action Script es el lenguaje propio de Flash, herramienta específica diseñada para la creación de contenidos multimedia. Como lenguaje, Action Script ofrece diferentes características de alto nivel:

- * Gramática: comandos en idioma inglés, muy parecidos a los lenguajes BASIC.
- * Sintaxis: construcción sencilla de comandos, se escriben casi como el inglés.
- * Funciones: conjunto de comandos que realizan tareas complejas.
- * Programación estructurada.
- * Orientación a objetos.
- * Flash ofrece un entorno de programación “amigable”, permitiendo organizar y depurar el código fácilmente.

2.4.2 Sobre Flash...

Flash es una herramienta para la creación de contenidos multimedia.

Flash está planteado entorno a la metáfora del teatro (o cine).

En una obra de teatro se encuentra:

- * La obra
- * El reparto
- * El guión
- * El escenario

En Flash hay:

- * Película (Película)
- * Biblioteca (Reparto)
- * Línea de tiempo (Guión)
- * Escenario (Escenario)

Película

Toda “obra” creada en Flash se llama Película

Una película puede tener diferentes formas de distribución:

*"Work in progress": cuando aún se está modificando la película. Se pueden modificar todos sus elementos. Se almacena como un fichero x fla, y solamente puede abrirse con Flash.

* Distribución off-line: Flash originalmente fue diseñado para la creación de CD-ROMs. Para distribuir una película terminada en CD-ROM, se "cierra" convirtiéndola en un fichero x.exe.

* Para los lenguajes compilados: el fichero que contiene el código es diferente al fichero ejecutable. Compilar implica convertir el código (texto) en algo ejecutable.

* Distribución on-line: Actualmente, Flash puede generar contenido para Internet. Cualquier película de Flash puede transformarse a formato Shockwave.

Cuando se está creando una película, se puede manipular, modificar, añadir o eliminar elementos.

Las propiedades de una película pueden modificarse utilizando la herramienta "Propiedades" (ver figura 6).

Principales tipos de miembros del reparto:

* **Bitmaps:** Gráficos o imágenes. Se pueden importar los formatos BMP, GIF, JPEG, LRG, PhotoShop, MacPaint, PNG, PICT, Targa, TIFF.

* **Texto:** En Flash existen campos que contienen caracteres con formato. Se pueden crear en Flash mediante la ventana de herramientas o importar ficheros creados con procesadores de texto.

* **Optimización de fuentes:** Incorpora también opciones de legibilidad para fuentes pequeñas, haciendo la lectura de nuestros textos más agradables y de alta legibilidad. Además de poder modificar la optimización, Flash permite también la selección de configuraciones preestablecidas para textos dinámicos y estáticos.

* **Sonidos:** Se pueden importar diferentes formatos de ficheros de sonido: MP3, AIFF, WAV y Shockwave.

* **Animación:** Flash 8 permite un mayor control de las interpolaciones habilitando un modo de edición desde el que se podrá modificar independientemente la velocidad en la que se apliquen los diferentes cambios de rotación, forma, color, movimiento, etc, de nuestras interpolaciones.

- * **Vectores:** figuras complejas formadas por líneas capaces de doblarse o curvarse. Se crean en Flash utilizando en la ventana Herramienta la Herramienta de Subselección.
- * **Director:** Director permite incluir películas de Flash dentro del reparto y ser modificadas con otras herramientas que este posee.
- * **Vídeo:** Para facilitar el resultado con formatos de vídeo, Flash 8 incluye un códec independiente de calidad superior capaz de competir con los mejores códecs de vídeo actuales con un tamaño de archivo mucho más pequeño. Además de una gran posibilidad de revestimientos para los controles de éste en nuestra película.
- * **3D:** Modelos tridimensionales creados con algún programa 3D.
- * **Scripts:** En estos miembros se almacena el código escrito en Action Scripts de la película.
- * **Java Script:** Debemos destacar además, que Flash puede trabajar conjuntamente con Java Script y ejecutar códigos y funciones sin problemas.
- * **Diseño:** Permite el uso de efectos visuales que nos facilitarán la creación de animaciones, presentaciones y formularios más atractivos y profesionales, así mismo, pone a nuestra disposición mecanismos para hacer este trabajo más cómodo y rápido, tales como la existencia de filtros y modos de mezcla.

2.5 Metodologías de Desarrollo de Software.

Uno de los temas más comunes en el mundo de la informática hoy en día es el de las metodologías de desarrollo de software: cómo trabajar eficientemente evitando las catástrofes que conllevan al fracaso de un gran porcentaje de proyectos. Una metodología tiene como objetivo aumentar la calidad del software que se produce en todas y cada una de sus fases de desarrollo, por medio de una mayor transparencia y control sobre el proceso; “producir lo esperado en el tiempo esperado y con el coste esperado” .

Durante los últimos años se han desarrollado dos corrientes en lo referente a las metodologías de desarrollo de software, las llamadas “pesadas” y las llamadas “ligeras o ágiles”. Las primeras se basan en la idea de conseguir el objetivo común por medio de orden y documentación, mientras que las segundas tratan de lograrlo por medio de la comunicación directa e inmediata entre aquello que intervienen en el proceso. Analizaremos a continuación tres de las más conocidas, sus características, ventajas y desventajas.

- **MultiMet como proceso de producción.**

Como propuesta para sentar los pilares del proceso, se estudió Multimet, una metodología de diseño nacional que describe etapas generales de la organización de un proyecto informático de multimedia. Su objetivo es que cada especialista componente del equipo de desarrollo conozca la aplicación de forma integral y pueda dirigir su trabajo hacia un fin común.

Se inicia con un estudio preliminar donde deben quedar definidos algunos elementos básicos relacionados con las necesidades de los usuarios, como las necesidades y los objetivos, la tecnología necesaria, el personal de desarrollo, un estudio del mercado potencial y la estrategia de comercialización. En adición se confecciona un plan que incluye todas las etapas del desarrollo con fecha de inicio, de terminación y responsables. Se hace un estudio de factibilidad económica y técnica centrado en la relación costos - beneficios, el impacto del producto final, costo de los elementos que hacen falta para el desarrollo, crecimiento potencial en el mercado y recursos disponibles. Luego de este estudio se determina si es factible o no desarrollar el producto y continuar con el resto de las etapas.

La siguiente es la etapa de definición de contenidos, donde se definen los objetivos desde el punto de vista de la aplicación propiamente dicho, teniendo en cuenta si es educativa, demostrativa o informativa, con la identificación del usuario final del sistema, basado en que los criterios de diseño están en función de su satisfacción. Se especifican los temas que serán tratados, su orden de aparición y teniendo en cuenta el nivel de detalle individual, la forma en que será estructurado pautando cada elemento de media a utilizarse y las restricciones de diseño.

La etapa de especificación de contenidos recopila toda la información referente a los objetos media a utilizar, mostrando un diagrama de flujo que tipifica la composición y navegación a través de módulos de pantallas; elabora el guión de contenidos donde describe cada media incorporada en una pantalla y la descripción de los eventos de interacción del usuario o propios del sistema a un nivel muy general. No modela la arquitectura del producto, sino la idea de su funcionamiento.

En el paso de implementación, considera preparada toda la información a incluir y el funcionamiento integral del sistema desde el punto de vista de las acciones del usuario, selecciona entonces la herramienta de autor a utilizar y comienza el montaje del software.

Por último en la fase de prueba garantiza la revisión por dos puntos de vista: solidez de la información y el funcionamiento adecuado. Elabora un plan de pruebas propia, espera la revisión del usuario y se centra en los aspectos de distribución del producto.

Analizando algunos aspectos de esta metodología señalamos la descripción lineal del proceso y cómo la selección de la herramienta a desarrollar se realiza en una fase cercana a la implementación y después de la elaboración. Es en la etapa de inicio donde se debe decidir con qué herramienta trabajar, para poder orientar la estructura y descripción del contenido hacia las potencialidades de la misma, y no esperar a la etapa de producción para indagar cuál se ajusta mejor al contenido. Cabe reflexionar qué sucedería si luego de varios meses de trabajo no aparece una herramienta que se ajuste al diseño propuesto, o resulta más factible hacerlo por una que emplea una filosofía diferente a la planificada; se necesitaría reestructurar, perdiendo en tiempo.

Como se señaló con anterioridad, describe la navegación del producto a través de un diagrama de flujo y la utilización de media y sus tipos en tablas. No se centra en la especificación de la estructura al nivel de programación, llevando un nivel elemental el análisis y diseño; con la obtención de medias y la descripción de un proceso verbal fuerza al programador a llevar adelante la confección de módulos que pueden resultar complejos. Carece de herramientas de sostén para la descripción del proceso de implementación.

Visto de esta forma, Multimet no es el proceso ideal a desarrollar, aunque aporta una visión inicial de cómo proyectarse.

- **RMM (Relationship Managment Methodology).**

El método RMM fue la primera metodología para el diseño de multimedia; si bien se trataba ésta de una versión con múltiples limitaciones que al ser detectadas dieron lugar a una versión extendida, ERMM. Se trata, probablemente, del único método para hipermedia que parece cubrir todo el ciclo de desarrollo, desde el estudio de factibilidad hasta la evaluación del sistema, aunque sólo propone actividades y productos concretos para las fases de análisis y de diseño.

La clase de aplicaciones para la cual RMM es más adecuada, corresponde a las que presentan una estructura regular para un dominio de interés, en donde hay clases de objetos, relaciones definibles entre éstas clases, y múltiples instancias de objetos dentro de cada clase. Muchas aplicaciones hipermediales satisfacen estos requerimientos, como por ejemplo, catálogos de productos, aplicaciones hipermediales frontales (front-end) para

bases de datos tradicionales o aplicaciones legadas. Considerando que muchas aplicaciones hipermediales de este tipo poseen datos volátiles que requieren actualizaciones frecuentes, se hace necesario disponer de medios que permitan automatizar y agilizar los desarrollos iniciales y los subsecuentes procesos de actualización.

El análisis se realiza por medio de un diagrama entidad-relación en el que sólo se permiten relaciones con cardinalidades $1 \rightarrow 1$ o $1 \rightarrow N$ y tampoco es posible establecer relaciones reflexivas.

Con respecto al diseño, una de las características más relevantes de este método es que éste se hace tanto de forma ascendente como descendente, ofreciendo una interesante manera de realizar una verificación. El diseño descendente empieza con la construcción de un diagrama de aplicación descendente, que es un esquema de las unidades de presentación (equiparables a ventanas) y de los enlaces que existen entre las mismas. A continuación, se compone cada una de esas unidades partiendo de las entidades del diagrama E-R, generando los denominados *m-slices*. En los *m-slices* se especifican los contenidos, enlaces, herramientas de navegación y funciones asociadas a cada unidad. El diseño ascendente toma como punto de partida los *m-slices* y genera un nuevo diagrama de aplicación ascendente que contrasta con el descendente.

Para el resto de las fases, estudio de factibilidad, implementación, pruebas y evaluación, no se propone ningún tipo de guías.

RMM constituye una metodología tentadora para el desarrollo del proceso por el desglose de las fases de la producción y la incorporación de diagramas para el diseño de la presentación, el comportamiento dinámico y la estructura de la navegación. No obstante, su uso óptimo se basa en las aplicaciones de catálogo de productos y aplicaciones hipermediales frontales para bases de datos tradicionales o aplicaciones legadas, por poseer una alta volatilidad de la información. En el lado opuesto del espectro, un trabajo artístico puede tener una estructura bastante difusa en la cual no se observen cambios frecuentes a través del tiempo, haciendo de RMM poco aplicable. Una multimedia educativa, entra en consideración de acuerdo al uso de los datos para su ejecución, fácilmente identificables son las plataformas de muestra de contenidos, mientras que las didácticas de contenido residente, se ven afectadas por el poco o nulo uso de esta metodología para su representación ingenieril.

- UML y RUP

UML oficialmente se presenta 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, enfocando a los constructores de herramientas en producir características más útiles.

Para modelar un sistema complejo se requieren múltiples modelos donde cada uno representa una vista del sistema; estos modelos se complementan entre sí y es esta la razón de la existencia de varios artefactos en UML que modelan diferentes aspectos del sistema, desde las vistas lógicas y físicas hasta los aspectos dinámicos, estáticos y funcionales. El grado de precisión con que se representa un modelo varía de acuerdo al grado de complejidad, detalle o abstracción que se desee representar o presentar ya sea al usuario o al equipo de trabajo.

A través de la historia se han desarrollado varios modelos de proceso de software siendo RUP uno de dichos paradigmas creado a partir de las mismas técnicas de modelado que originaron productos eficientes y descrito a través de UML, orientado por tanto a la visión objeto. Al igual que cualquier notación, el proceso unificado actúa como un modelo que puede adaptarse a cualquier tipo de proyecto y empresa y que basa su desarrollo en ciclos que al concluir originan una versión del producto. Cada ciclo consta de cuatro fases: Inicio, Elaboración, Construcción y Transición llevando a cabo el cumplimiento de los objetivos propuestos y en su culminación el alcance de un hito siendo respectivamente: objetivos del ciclo de vida, arquitectura del ciclo de vida, funcionalidad operativa inicial y la versión del producto. Cada fase a su vez consta de varias iteraciones que definen los tiempos de ejecución, tratan los riesgos más importantes y se suceden de forma incremental a una etapa superior de producción garantizando el coste de riesgo del producto al de una iteración, manejando de forma más visible y progresiva los resultados, e identificando de manera creciente las necesidades y requisitos totales del usuario y el sistema.

RUP es centrado en los modelos y diagramas como vehículo de comunicación más expresivo de las descripciones en el lenguaje natural, tratando de minimizar su uso y el de las especificaciones textuales del sistema. Es guiado por casos de uso, el instrumento para validar y fundamentar la arquitectura del software, aspecto en quien se centra cada modelo a través de sus proyecciones del análisis y el diseño.

El gráfico que representa el RUP incluye las cuatro fases importantes que son: la iniciación, elaboración, construcción y transición, las cuales muestran que para producir

una versión del producto en desarrollo se aplican todas las actividades de ingeniería pero con diferente énfasis; en las versiones preliminares, como además indica la intuición, hay más interés en actividades de modelado del negocio, requisitos, análisis y diseño; conforme se producen versiones la atención pasa a las actividades de implementación, pruebas y despliegue.

OMMMA – L

Muchos lenguajes de modelado han sido propuestos para la especificación del proceso de desarrollo de aplicaciones multimedia, aunque aún no existe un estándar que cubra todos los aspectos relacionados con el comportamiento dinámico e interactivo asociado a las interfaces gráficas para una generalización de herramientas, productos y procesos.

En medio de esta búsqueda de una modelación adecuada, el Lenguaje de Modelado Orientado a objetos de Aplicaciones Multimedia (OMMMA - L) 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) [véase Anexos I Fig.7] para la interfaz de usuario.

El paradigma MVC es un modelo de arquitectura conocido en el desarrollo de aplicaciones orientadas a objetos que distinguen un componente *modelo* sosteniendo la funcionalidad del núcleo y los datos, un componente *vista* para mostrar la información al usuario y un componente *controlador* para manipular los eventos de interacción. Un mecanismo de propagación de cambios asegura la consistencia entre el modelo y la interfaz visual.

Una multimedia está formada por su estructura lógica, que incluye los objetos de aplicación del dominio y los objetos de media asociados, una presentación espacial, un comportamiento temporal dado en los requisitos de ejecución en tiempo real y los objetos de media de funcionamiento continuo, y el control interactivo, que tiene lugar en el manejo de eventos de interacción con el usuario.

2.6 Conclusiones.

Una vez vista las características de OMMMA – L, se argumenta su aplicación exitosa partiendo de la idea de que no es un lenguaje nuevo, sino una extensión del UML que se imparte en la universidad, por lo que no es necesario aprenderlo, sino interpretar las características extendidas, centrados a la lógica de funcionamiento de una multimedia, que

es por lo general, sencilla. Muestra análisis similares a otras metodologías potentes como RMM y no se especializa en una clasificación de producto, sino que generaliza a través del uso de la semántica original de UML. Es robusto y altamente descriptivo, refleja el proceso en todas sus etapas y hereda de RUP el ciclo de vida basado en iteraciones y el flujo de trabajo iterativo e incremental, centrado en casos de uso y en la arquitectura.

Por su parte, RUP modela, centrado en la arquitectura del software, una serie de artefactos que permite la división de responsabilidades dentro de la ingeniería, la construcción precisa de la estructura del producto y un mecanismo de producción iterativa e incremental que dividen en pasos dentro el flujo de trabajo garantizando el crecimiento.

Por tanto se decide basar la propuesta de este trabajo en la descripción de un proceso basado en RUP, y utilizando OMMMA – L para la especificación del producto.

Capítulo III

Descripción de la Solución Propuesta.

3.1 Introducción.

En el presente capítulo se describe y modela la solución propuesta para nuestro sistema, el cual tiene como objetivo, facilitar el aprendizaje de la asignatura de programación orientada a objetos, así como servir de apoyo al profesor a la hora de impartir los contenidos de dicha asignatura.

3.2 Situación problémica.

En nuestra universidad la asignatura de programación I se apoya en los servicios de red para la distribución de tele conferencias y demás materiales informativos que permiten que los estudiantes puedan tener accesos a gran cantidad de información y contenido para el mejor aprendizaje de la asignatura, aunque en ocasiones los resultados no son los esperados debido a problemas que se presentan como son: demoras en la actualización de las tele conferencias, y de las clases en formato digital, problemas de redes que impiden la descarga de los contenidos. Estos problemas han influido de manera negativa en la promoción de los resultados de la asignatura.

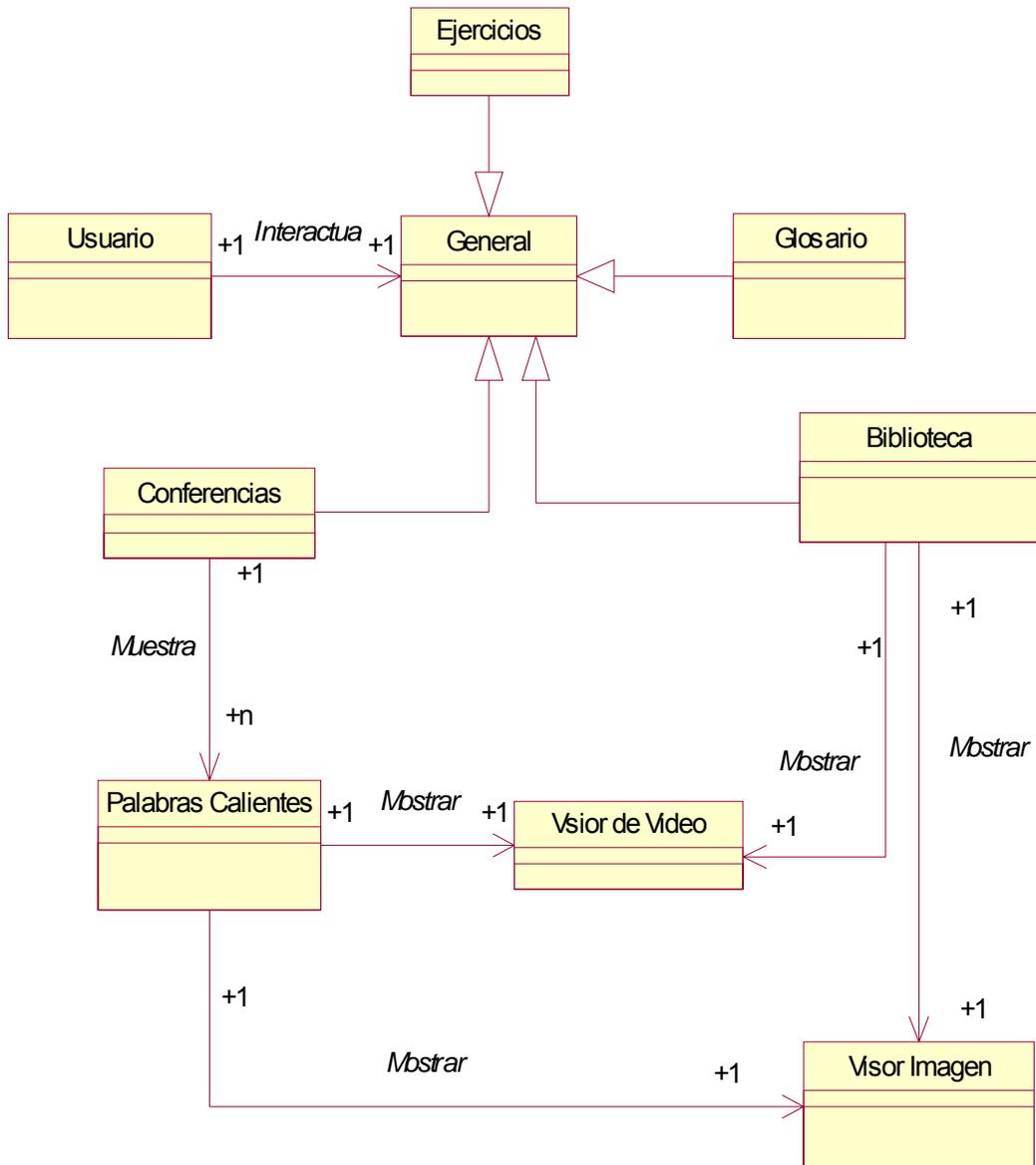
3.3 Objeto de automatización.

La multimedia implementada pretende automatizar de manera parcialmente dinámica todos los métodos usados en la docencia en P1, acorde a los temas que se imparten, tele conferencias, conferencias, ejercicios y términos propios de la asignatura. Con el desarrollo de nuestro sistema tratamos de darle solución a estos problemas, logrando centralizar los contenidos, hacerlos de manera amena y que el estudiante se sienta motivado, además, lograr que el uso de la red disminuya para acceder al contenido.

3.4 Modelo del sistema.

A partir de este punto se comienza a modelar el sistema que se va a construir. Para ello se identifican sus requisitos, tanto funcionales como no funcionales, y se modelan los funcionales en términos de casos de uso del sistema.

3.4.1 Modelo de dominio.



3.4.

2 Requisitos Funcionales.

R.1 Ver conferencias.

- 1.1 Mostrar al usuario conferencias por temas.
- 1.2 Cada conferencia debe mostrar las palabras calientes.
- 1.3 Cada palabra calientes debe ser capaz de mostrar el visor correspondiente.

R.2 Ver ejercicios.

2.1 Mostrar al usuario los ejercicios y sus respuestas.

2.2 Mostrar al usuario los ejercicios propuestos.

R.3 Ver Glosario.

3.1 Muestra al usuario los términos usados en la asignatura y sus significados.

R.4 Ver Biblioteca.

4.1 Muestra al usuario las teleconferencias a través de un visor de video.

4.2 Muestra una galería de imágenes.

4.3 Permite ver además algunas animaciones.

R.5 Controlar Sonido.

5.1 Permite al usuario controlar el sonido de fondo.

R.6 Mostrar Ayuda.

6.1 Muestra al usuario la ayuda del sistema.

R.7 Modificar el contenido.

7.1 El profesor accede a los documentos que contienen el contenido.

7.2 Realiza los cambios pertinentes.

7.3 Guarda los cambios realizados.

7.4 El sistema muestra los cambios realizados en el contenido.

R.8 Imprimir contenido.

8.1 El usuario selecciona imprimir un contenido.

8.2 El sistema muestra ventana de impresión.

8.3 El sistema realiza la impresión.

3.4.3 Requisitos no Funcionales.

Apariencia o interfaz externa:

- Diseño sencillo, permitiendo la utilización del sistema sin mucho entrenamiento.
- Diseño encuadrado para resoluciones de 800x600, pero preparado para verse en otras resoluciones.

Requerimientos mínimos.

Requiere de Flash Player 8 para su correcto funcionamiento. Los requerimientos de hardware mínimos exigidos para Flash Player 8 sobre Windows son: Procesador Intel Pentium III de 450 MHz (o equivalente) y versiones posteriores y 128 MB de RAM. Para Macintosh son: 500 MHz PowerPC G3 y versiones posteriores y 128 MB de RAM.

Compatibilidad.

Windows	
Plataforma	Navegador
Windows 98	Microsoft Internet Explorer 5.x, Netscape 4.7, Netscape 7.x y superior, Firefox 1.x, Mozilla 1.x, AOL 8 u Opera 7.11
Windows Me	Microsoft Internet Explorer 5.5, Netscape 4.7, Netscape 7.x y superior, Firefox 1.x, Mozilla 1.x, AOL 8 u Opera 7.11
Windows 2000	Microsoft Internet Explorer 5.x, Netscape 4.7, Netscape 7.x o superior, Firefox 1.x, Mozilla 1.x, CompuServe 7, AOL 8 u Opera 7.11
Windows XP	Microsoft Internet Explorer 6.0, Netscape 7.x, Mozilla 1.x, Firefox 1.x, CompuServe 7, AOL 8 u Opera 7.11

Macintosh	
Plataforma	Navegador
Mac OS X 10.1.x, Mac OS X 10.2.x, Mac OS X 10.3.x o Mac OS X 10.4.x	Microsoft Internet Explorer 5.2, Netscape 7.x y superior, Mozilla 1.x, Firefox 1.x, AOL 7, Opera 6 o Safari 1.x y superior

Linux (x86, 32-bit)

Plataforma	Navegador
RedHat Enterprise Linux WS v. 3	Mozilla 1.2 y superior, Netscape 7.1
RedHat Linux 9	Mozilla 1.2 y superior, Netscape 7.1
Sun Java Desktop System 1.0	Mozilla 1.2 y superior, Netscape 7.1

Confiabilidad.

Se realizaron las validaciones necesarias para que en caso que la persona que la estuviera actualizando cometiera algún error esto no provoque un incorrecto funcionamiento de la misma.

3.4.4 Actores.

Los actores de un sistema son agentes externos: aquellas personas o sistemas que interactúan con él. A continuación se describen en la siguiente tabla.

Tabla 1. Actores del sistema.

Actor	Justificación
Usuario	Generaliza a todos los usuarios del sistema. Realiza las operaciones comunes a todos ellos.
Estudiante	Es a quien va dirigido el sistema
Profesor	Además de las operaciones comunes, es el único capaz de modificar o corregir algún error existente en el contenido mostrado.

3.4.5 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 de seguir. (Como se muestra en el Anexo II fig.1)

3.4.7 Diagrama de casos de uso del sistema.

Los casos de uso son fragmentos de funcionalidad del sistema. En ellos se describe la secuencia determinada de eventos que realiza un actor en interacción con la aplicación. En Anexos II, Fig.2 se muestran los diagramas casos de uso del sistema.

3.4.8 Descripción de los casos de uso.

En este epígrafe se describen los casos de uso del sistema.

Tabla. 2 Descripción de CU. Ver Conferencias.

Caso de Uso	01 Ver Conferencias
Actores	Usuario
Resumen	El caso de uso se inicia cuando el usuario accede en el menú general a la opción conferencias, Este a su vez muestra conferencias organizadas por temas en la cual se muestran las palabras calientes contenidas en cada conferencia.
Propósito	Permitir al usuario ver y estudiar el contenido mostrado.

Referencias	R1
Precondiciones	El usuario ha ejecutado la multimedia.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario ejecuta la multimedia.	
	2. Muestra menú general, que incluye conferencias, ejercicios, glosario y biblioteca.
3. accede a la opción conferencias.	4. Muestra conferencias organizadas por temas.
	5. finaliza el caso de uso.

Tabla.3 Descripción de CU. Ver Accionar palabras calientes.

Caso de Uso	02 Accionar palabras calientes
Actores	Usuario
Resumen	El caso de uso se inicia cuando el usuario da clic sobre las palabras calientes contenidas en las conferencias y el sistema muestra los visores de video o imágenes correspondientes a la palabra accionada.
Propósito	Permitir al usuario ver imágenes y animaciones que facilitan el aprendizaje del contenido tratado.
Referencias	R1 1.3
Precondiciones	El usuario accionado una palabra caliente.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario acciona palabra caliente	
	2. Muestra visor de video o imágenes de acuerdo a la palabra accionada.
3. El caso de uso finaliza cuando el usuario cierra el visor.	

Tabla.4 Descripción de CU. Ver Ayuda.

Caso de Uso	03 Ver Ayuda
Actores	Usuario
Resumen	El caso de uso se inicia cuando el usuario da clic sobre el botón ayuda.
Propósito	Permitir al usuario ver la ayuda del sistema.
Referencias	R6
Precondiciones	El usuario da clic en el botón ayuda
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario da clic sobre el botón ayuda	
	2. Muestra ayuda del sistema.
3. El caso de uso termina cuando el	

usuario cierra la ayuda.	
--------------------------	--

Tabla.5 Descripción de CU. Controlar Sonido.

Caso de Uso	04 Controlar Sonido.	
Actores	Usuario	
Resumen	El caso de uso se inicia cuando el usuario da clic sobre el botón sonido.	
Propósito	Permitir al usuario escuchar o detener el sonido de fondo.	
Referencias	R5	
Precondiciones	El usuario a dado clic sobre el botón sonido.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El usuario da clic sobre el botón sonido.		
	2. Ejecuta o detiene el sonido de fondo.	
	3. Finaliza el caso de uso	

Tabla.6 Descripción de CU. Ver Biblioteca.

Caso de Uso	05 Ver Biblioteca.	
Actores	Usuario	
Resumen	El caso de uso se inicia cuando el usuario accede en el menú general a la opción biblioteca, Este a su vez muestra tele clases organizadas por temas, además de una galería de imágenes y animaciones.	
Propósito	Permitir al usuario ver, escuchar y estudiar el contenido mostrado.	
Referencias	R1	
Precondiciones	El usuario ha ejecutado la multimedia.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El usuario ejecuta la multimedia.		
	2. Muestra menú general, que incluye conferencias, ejercicios, glosario y biblioteca.	
3. accede a la opción conferencias.	4. Muestra conferencias organizadas por temas.	
	5. finaliza el caso de uso.	

Tabla. 7 Descripción de CU. Ver Conferencias.

Caso de Uso	06 Ver Conferencias	
Actores	Usuario	
Resumen	El caso de uso se inicia cuando el usuario accede en el menú general a la opción conferencias, Este a su vez muestra conferencias organizadas por temas en la cual se muestran las palabras calientes contenidas en cada conferencia.	
Propósito	Permitir al usuario ver y estudiar el contenido mostrado.	
Referencias	R4	

Precondiciones	El usuario ha ejecutado opción biblioteca.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El usuario ejecuta la multimedia.		
	2. Muestra menú general, que incluye conferencias, ejercicios, glosario y biblioteca.	
3. Accede a la opción biblioteca.	4. Muestra tele clases organizadas por temas, además de una galería de imágenes y animaciones.	
5. Ejecuta la opción tele clases.		
	6. El sistema muestra un visor de video donde se muestra la tele clases correspondiente.	
	7. Finaliza el caso de uso.	

Nota: Al ejecutar la opción galería de imágenes o las animaciones el sistema responde mostrando los visores correspondientes a esta acción.

Tabla. 8 Descripción de CU. Ver Ejercicios.

Caso de Uso	07 Ver Ejercicios.	
Actores	Usuario	
Resumen	El caso de uso se inicia cuando el usuario accede en el menú general a la opción ejercicios, Este a su vez muestra ejercicios y algunas respuestas.	
Propósito	Permitir al usuario ejercitarse en los contenidos tratados.	
Referencias	R2	
Precondiciones	El usuario ha ejecutado la opción ejercicios.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El usuario ejecuta la multimedia.		
	2. Muestra menú general, que incluye conferencias, ejercicios, glosario y biblioteca.	
3. accede a la opción ejercicios.	4. Muestra ejercicios y algunas respuestas.	
	5. finaliza el caso de uso.	

Tabla. 9 Descripción de CU. Ver Glosario.

Caso de Uso	08 Ver Glosario.	
Actores	Usuario	
Resumen	El caso de uso se inicia cuando el usuario accede en el menú general a la opción glosario, donde se muestran un grupo de términos utilizados en la asignatura con su significado.	
Propósito	Permitir al usuario conocer el significado de estos términos.	
Referencias	R3	
Precondiciones	El usuario ha ejecutado la multimedia.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	

1. El usuario ejecuta la multimedia.	
	2. Muestra menú general, que incluye conferencias, ejercicios, glosario y biblioteca.
3. accede a la opción glosario.	4. Muestra algunos términos utilizados en la asignatura con sus significados..
	5. finaliza el caso de uso.

Tabla. 10 Descripción de CU. Imprimir Contenido.

Caso de Uso	09 Imprimir Contenido.	
Actores	Usuario	
Resumen	El caso de uso se inicia cuando el usuario accede en el menú general a las opciones conferencias, glosario o ejercicios. Donde se muestra el contenido seleccionado y le da clic a al botón imprimir.	
Propósito	Permitir al usuario ver y estudiar el contenido mostrado.	
Referencias	R8	
Precondiciones	El usuario ha ejecutado la opción conferencias, glosario o ejercicios.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El usuario ejecuta la multimedia.		
	2. Muestra menú general, que incluye conferencias, ejercicios, glosario y biblioteca.	
3. Accede a la opción conferencias, ejercicios o glosario.	4. Muestra el contenido seleccionado.	
5. Da clic en el botón imprimir.	6. Imprime el contenido.	
7. Finaliza el caso de uso.		

Tabla. 11 Descripción de CU. Modificar Contenido.

Caso de Uso	10 Modificar Contenido.	
Actores	Profesor	
Resumen	El caso de uso se inicia cuando el profesor accede a los documentos que contienen los contenidos y realiza las modificaciones necesarias.	
Propósito	Permitir al profesor actualizar el contenido que se muestra en la multimedia.	
Referencias	R7	
Precondiciones	El usuario ha ejecutado la multimedia.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El profesor accede a los archivos que contienen la información que se desea modificar.		
2. Realiza los cambios deseados y guarda dichos cambios.		
3. Ejecuta la multimedia.	4. Muestra menú general, que incluye conferencias, ejercicios, glosario y biblioteca.	

5. Accede al contenido modificado.	6. Muestra los cambios realizados.
	7. Finaliza el caso de uso.

3.5 Diagrama de clases del modelo de objeto.

En la confección del diagrama que da al fin una estructura de entidades para estudio inicial del software, se utilizaron tres prototipos agregados al proceso. Diremos que un objeto puede ser de tipo *escenario* cuando representa un conjunto de pantallas que muestran una información a través de objetos con similar funcionalidad, de tipo *aplicación* cuando agrupa elementos de media y aúna sus funcionalidades como una entidad y de *media* cuando se hace referencia a sonido, texto, imágenes, animaciones, video. Partiendo de la navegación expandida del producto y especificada a través de la especificación de casos de uso, se reconocieron las entidades principales. (Ver Anexos II fig.3)

3.6 Conclusiones.

En este capítulo se hizo una descripción de la propuesta de solución a través de la modelación de dominio: la identificación de los actores y los casos de uso correspondientes; el planteamiento de los requisitos funcionales y no funcionales de la aplicación que se va a desarrollar y la modelación de la misma en términos de casos de uso de sistema. A partir de este punto se puede comenzar a construir el sistema que constituye la propuesta de solución.

Capítulo IV.

Construcción de la Solución Propuesta.

4.1 Introducción.

En el presente capítulo se diseña la propuesta de solución, se especifican los principios de diseño gráfico para multimedia, apoyándonos en los diagramas de presentación y los diagramas de clases correspondientes a esta fase.

4.2 Patrones de diseño.

En medio de la búsqueda de una modelación adecuada, el Lenguaje de Modelado Orientado a objetos de Aplicaciones Multimedia (OMMMA - L) se lanza como una propuesta de extensión de UML para la integración de especificaciones de sistemas multimedia basados en el paradigma orientado a objetos, En el diseño de la aplicación se tuvo en cuenta principalmente el patrón MVC (Modelo Vista Controlador)(Ver Anexos II Fig.4), el cual se basa en la interacción de una clase vista para la interfaz de usuario, una clase controladora, la cual controla las solicitudes del sistema y una clase modeladora es la encargada de dar respuesta a las solicitudes de la clase controladora.

4.1 Modelo análisis.

4.1.1 Diagrama de presentación.

Este es un artefacto nuevo dentro del lenguaje UML, es específico de OMMMA – L, y sirve, como se explicó en la fundamentación teórica, 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. Aunque UML especifica propuesta de interfaz de usuario en sus requisitos no funcionales, no es un aspecto de fuerte medición, ni consideración en el análisis de la arquitectura del software. Un diagrama de clases es utilizado para la representación de este diagrama, teniendo en cuenta el modelo de objetos preliminar creado.

En Anexos II Fig. 5 se muestra el diagrama de presentación del escenario general.

4.4.2 Diagrama de Clase de Análisis.

Los diagramas de clases son el núcleo de un modelo de aplicación orientado a objeto, y describen el modelo estático presentado en MVC. Esencialmente consisten en clases y asociaciones que describen la estructura de objeto y las posibles interrelaciones estructurales. Para representar la lógica de aplicación y los tipos de media del modelo estático, representamos un diagrama de clases en dos partes vinculadas: una jerarquía de definición de tipos de media que recoge la composición de las medias utilizadas en el diseño de un software para una herramienta de autor determinada, y el modelo lógico de la aplicación que comprende clases y sus asociaciones para describir el modelo del dominio. (Ver Anexos II Fig.6)

4.4.3 Diagrama de Clases de Diseño.

Se modela utilizando UML como metodología, el cual permite fácilmente ver las relaciones establecidas entre las clases y comprender como se gestionan las solicitudes, que se realizan al sistema. (Véase Anexos II Fig.7)

Se deben corresponder las clases de nuestro diseño con las de la herramienta de autor empleada, para poderlas identificar a la hora de programar. OMMMA – L propone en cada diagrama de clases elaborado, adicionar la jerarquía de media de la herramienta y enlazar a través de relaciones las clases del tipo correspondientes. Una forma secundaria de llevar adelante esta correspondencia es sustituir en los estereotipos de las clases utilizadas las de la herramienta. Se muestra la jerarquía de las clases de interfaz y la correspondencia en el modelo de clases de la aplicación.

4.4.4 Diagrama de Colaboración.

Los diagramas de colaboración explican gráficamente las interacciones existentes entre las instancias o clases de modelo de estas. El punto de partida de las interacciones es el cumplimiento de las poscondiciones de una operación determinada y muestran las interacciones en formato de grafo o red. (Véase Anexos II Fig.8)

4.4.5 Diagrama de Secuencia.

Los diagramas de secuencia son usados para especificar el comportamiento predefinido de una aplicación multimedia. Además de mantener su semántica original en el paso de mensajes para las interacciones. Estos se generan a partir de los diagramas de colaboración. (Véase Anexos II Fig.9) Muestra las interacciones a través de un cerco o muro.

4.5 Etapa de prueba.

En la etapa de pruebas de esta segunda iteración la ingeniería se volcó en los casos de uso de prueba, centrando las bases para un buen proceso de comprobación de fallos funcionales y estéticos de la multimedia para las correcciones internas e intermedias; así como las versiones finales entregadas a terceros.

El trabajo con OMMMA – L termina para esta etapa, al menos las extensiones, pues para las fases presente y venidera, los artefactos y sus semánticas se mantienen invariantes. Se reitera en este punto que el lenguaje propuesto centra su trabajo en la descripción de las etapas de análisis y diseño.

4.6 Modelo de implementación.

4.6.1 Diagrama de componentes.

La declaración de cada uno de los módulos de ejecución se representó en este diagrama, en analogía con la estructura que se le fue moldeando desde el diagrama de navegación en la fase de análisis. (Ver Anexos II Fig.10)

4.7 Modelo de Prueba.

Se ejemplifica este proceso a través de los casos de pruebas del sistema para el escenario *Conferencias*, que basa su funcionamiento en el *escenario General* y que se especifica en el caso de uso *ver conferencias*. En la estrategia de pruebas se especificaron los

diferentes tipos de pruebas, haciendo marcado énfasis en las de solidez, referidas a la estética visual de la interfaz.

4.7.1 Estrategia de pruebas

Para el producto no se automatiza ningún caso de prueba, todos serán llevados a cabo de forma manual.

4.7.2 Pruebas de instalación

Ejecutar la multimedia en los sistemas operativos MAC, Win98, Win2000, WinXP y las distribuciones de Linux RedHat y Debian.

4.7.3 Pruebas de solidez.

- Correspondencia entre el tema tratado, el texto y el resto de los medios que aparecen en cada pantalla contra el guión de contenidos o medias.
- Revisión ortográfica de los textos de la multimedia.
- Calidad de los medios que se muestran: sonido, imágenes, incluyendo diferentes resoluciones de pantalla.
- Cumplimiento de las pautas trazadas por diseño.

4.8 Conclusiones

En este capítulo se modeló detalladamente la solución propuesta, a través del modelo de diseño, de implementación y de despliegue. Se analizaron los principios de diseño de la interfaz de la multimedia, los estándares de codificación a emplear durante la fase de construcción y se hizo una breve descripción del modelo de pruebas que se propone realizar al sistema para comprobar que cumpla los requerimientos que le dieron origen.

Capítulo V

Estudio de Factibilidad.

Introducción.

En el presente capítulo se hace un análisis de los costos y los beneficios que generará el desarrollo de la aplicación y la relación entre ellos, con el objetivo de demostrar la factibilidad del proyecto, así como la conveniencia de su explotación en la universidad.

5.1 Estimación de costo.

Existen diversos modelos para estimar los costos de los proyectos de software. Entre ellos uno de los más aceptados internacionalmente en la actualidad es el modelo COCOMO v2 (del inglés Constructive Cost Model). Este modelo expresa el esfuerzo de desarrollo en términos de Personas Mes (cantidad de tiempo que una persona dedica a trabajar sobre el proyecto de desarrollo durante un mes). Finalmente se hace una estimación del costo monetario del proyecto teniendo en cuenta el tiempo estimado de desarrollo, la cantidad de personas involucradas y el salario de estas personas (Ver ANEXO III).

En el caso de POO Interactivo, al ser una multimedia a desarrollar en el marco de la Universidad de las Ciencias Informáticas, no se hace necesario calcular el costo monetario, aunque se realizó, pues la elaboración del sistema no supone gastos adicionales por concepto de salario para la institución, ya que cuenta con el capital humano necesario (estudiantes y profesores vinculados al proyecto) y no necesita contratar otro personal. Por otra parte la universidad cuenta con la infraestructura tecnológica necesaria para llevar a cabo el proyecto.

5.2 Beneficios tangibles e intangibles.

POO Interactivo es un sistema que por su flexibilidad puede adaptarse a varios lenguajes de programación que se impartan con un plan de estudio similar al que se ajustó cuando fue diseñado.

Por otra parte, su implantación disminuye considerablemente el uso de la red y servidores, por ejemplo las conferencias y ejercicios de la asignatura las tendrán todas sin necesidad de utilizar la red en caso de que esta por algún motivo no este funcionando, entre otras.

Al lograr una mayor ayuda a la hora de impartir la docencia se logra que el profesor pueda dedicar tiempo quizás a otros estudiantes menos aventajados o cualquier labor docente en que los estudiantes puedan prescindir de él en alguna medida y esta necesidad pueda

solucionarla la aplicación. Es bueno resaltar que la multimedia se hizo con el propósito de ayudar al profesor a impartir la docencia, nunca con el de sustituirlo, algo que hasta el momento es imposible.

Se logra un aumento considerable en la comprensión de las clases al dar la posibilidad de enriquecer las conferencias con imágenes y animaciones, por ejemplo para simular algún algoritmo u otro proceso que el profesor crea es un poco abstracto a la hora de impartirlo, por ejemplo.

5.3 Conclusiones.

Del estudio realizado en este capítulo, a partir del análisis de los beneficios que aporta a la erradicar lo mas posible los problemas que se presentan en la docencia en la asignatura P1, se concluye que es factible el desarrollo de esta aplicación, por todos los aportes económicos y sociales que genera.

Recomendaciones.

Para versiones futuras de POO Interactivo se recomienda darle a la aplicación la interactividad que realmente requiere, cosa tales como ejercicios de marcar, de enlazar; métodos para que el estudiante pueda medir sus conocimientos y le de un estado del resultado de sus respuestas. También se debe mejorar en su dinamismo. Esto se podrá alcanzar al crear los menús de forma dinámica, obteniéndose los datos necesarios como: cantidad de botones y título del botón, a partir de un fichero con estructura de XML. De esta forma aseguramos que la persona asignada para darle mantenimiento a la multimedia no se le haga necesario tener conocimientos de Macromedia Flash, solo necesitara tener conocimientos básicos de HTML y XML para con actualizar los ficheros contenedores de datos y con esto será suficiente para alargarle su vida útil. Creemos que con el estado de nuestra actual versión sería provechoso que la utilizaran estudiantes que por determinadas razones no se encuentran en la escuela y tengan necesidad de recibir la asignatura. Se les podría facilitar la aplicación con toda la información necesaria en un DVD.

Conclusiones

En este trabajo se llegó a la conclusión de la necesidad de implementar nuestro sistema debido a deficiencias que se presentaron en el aprendizaje de la asignatura de programación I, deficiencias que fueron descritas en el capítulo I. Utilizando para ello un mecanismo de fácil manipulación y que pueda brindar al estudiante y al profesor un método para facilitar el aprendizaje y para impartir un contenido. Por lo cual se optó por la creación de una multimedia clasificada como tutorial, interactiva que muestre información de manera amena y de fácil navegación.

Se llegó a la conclusión también de que la metodología idónea para llevar a cabo el proceso de desarrollo es RUP (Proceso Unificado de Rational) y dentro de ella OMMMA-L como una extensión, la cual permite la modelación del proceso de desarrollo de un software multimedia.

Se definieron los requerimientos del sistema, tanto funcionales como no funcionales, y posteriormente se estructuró el modelo de casos de uso del sistema, describiéndose cada caso de uso para una mejor comprensión de la funcionalidad que brindan.

Se diseñó el sistema, a través de diagramas de clases aplicando el modelo vista controlador (MVC),

Finalmente, se analizaron los beneficios y costos que genera el sistema, se compararon y se concluyó que el proyecto es factible.

En este trabajo se demostró la importancia y facilidades que brinda nuestro sistema para el aprendizaje de la asignatura de programación orientada a objetos, haciendo uso de las tecnologías que se han puesto en nuestras manos. Permitiendo a su vez interactuar de forma amena y sencilla con toda la información que ponemos a su disposición en esta multimedia. Esperando que cumpla con su objetivo en el mejoramiento de su cultura integral y formación.

Referencia Bibliográfica

Fuente electrónica.

[1] Curso de Postgrado Las tecnologías informáticas en la construcción del conocimiento y la gestión del aprendizaje.

Curso 2005-2006

[2] Departamento de Técnicas de Programación. Plan de Trabajo Metodológico. Curso 2005/06 - 1er Semestre

Sitios WEB.

[3] LA TECNOLOGIA EDUCATIVA. EL USO DE LAS NTIC EN LA EDUCACION

[Disponible en:

<http://cepes.uh.cu/bibliomaestria/tendenciaspedagogicas/libros/TENDE2.doc.>]

[4] Dr. Pere Marquès Graells, Ventajas e inconvenientes del multimedia educativo, 2000 (última revisión: 9/06/03). [Disponible en:

<http://dewey.uab.es/pmarques/ventajas.htm#inicio>].

[5] Maricopa Community College. Multimedia Authoring. Maricopa Center for learning and Instruction (MCLI). [Disponible en: <http://www.mcli.dist.maricopa.edu/authoring/lang.html>]

[6] Ramos Bravo, Juan L. Los sistemas multimedia en la enseñanza. [Disponible en:

<http://www.ice.upm.es/wps/lbr/documentacion/libros/sistemul.pdf>]

[7] Vázquez Enrique, Laritza. Moreno Ruiz, Rafael. Desarrollo de contenidos digitales.[Disponible en:

<http://redescolar.ilce.edu.mx/redescolar/cursos/sepacomputo/descondigi.pdf>]

[8] Valencia Santacruz, Liliana P. Herramientas de autor. Criterios de selección. Madrid, 1998. [Disponible en: <http://www.um.es/ead/red/9/sga.pdf>]

[9] Zapata, Miguel. Brecha digital y educación a distancia a través de redes. Funcionalidades y estrategias pedagógicas para el E-Learning. [disponible en: www.um.es/fccd/anales/ad08/ad0815.pdf]

[10] Curso de Flash 8. [Disponible en: <http://www.aulaclic.es/flash8/index.htm>]

[11] Sauer, Stefan-Engels Gregor, MVC-Based Modeling Support for Embedded Real-Time Systems

[12] Isakowitz, T., Stohr, E. A. and Balasubramanian, P. «RMM: a methodology for structured hypermedia design». Comm. of the ACM, 38(8), 34-44.

[13] Isakowitz, Tomás - Stohr, Edward A. - Balasubramanian, P. RMM: Metodología para el Diseño Estructurado de Hipermedios

[14] Sauer, Stefan-Engels Gregor; Extending UML for modeling of multimedia applications.

[15] Sauer, Stefan-Engels Gregor; UML-Based behavior Specification of Interactive multimedia Applications

ANEXO I

Fig.1 Ambiente de trabajo del Asymetrix Toolbook

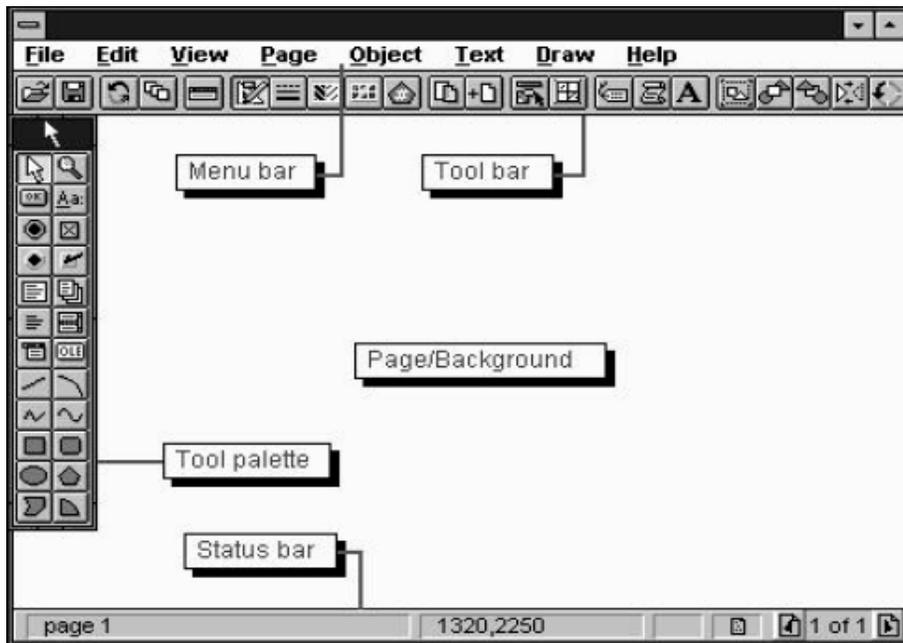


Fig. 2 Ambiente de trabajo de Visual Basic.

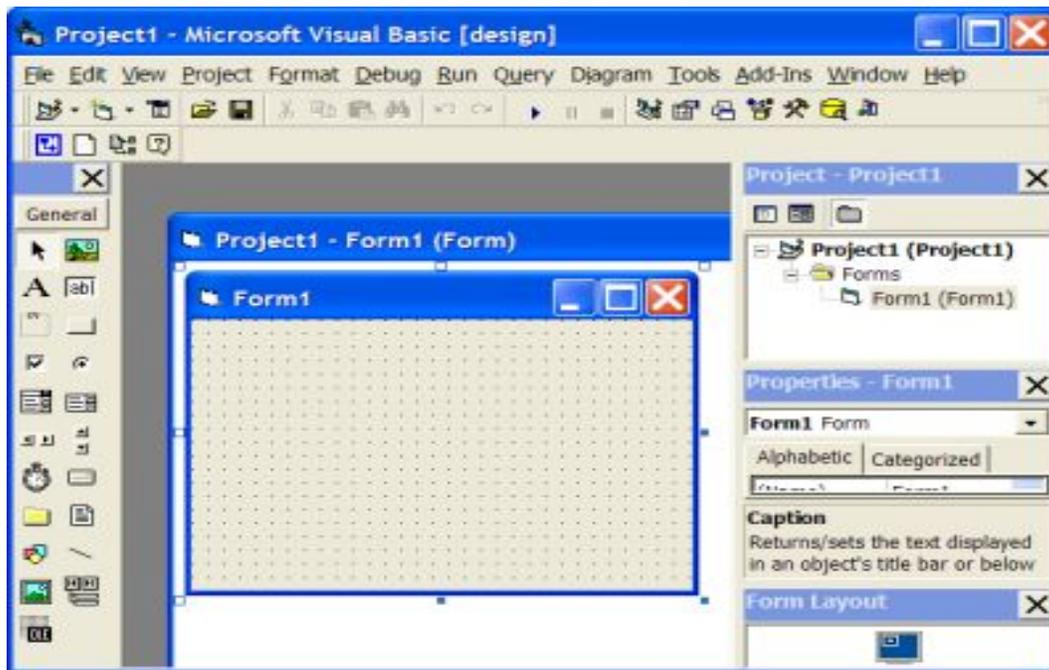


Fig. 3 Ambiente de trabajo de Authorware.

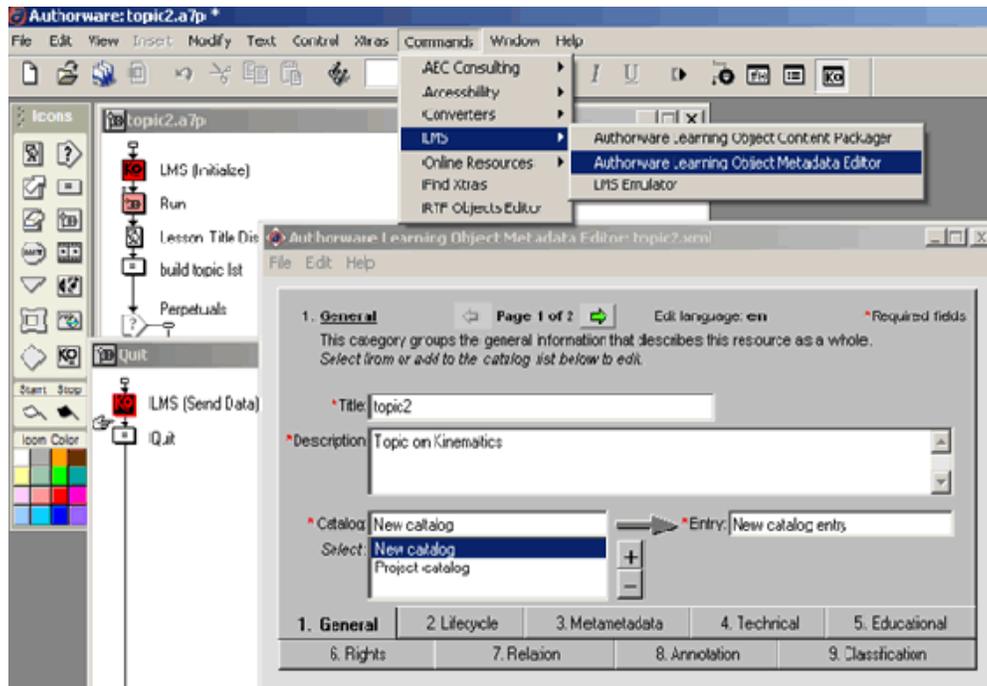


Fig. 4 Paleta de iconos Disponible en Authorware.

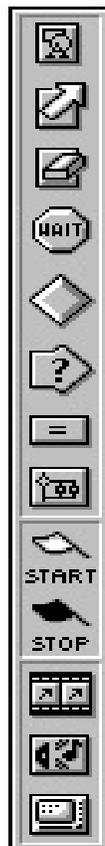


Fig. 5 Ambiente de trabajo de Macromedia Flash.

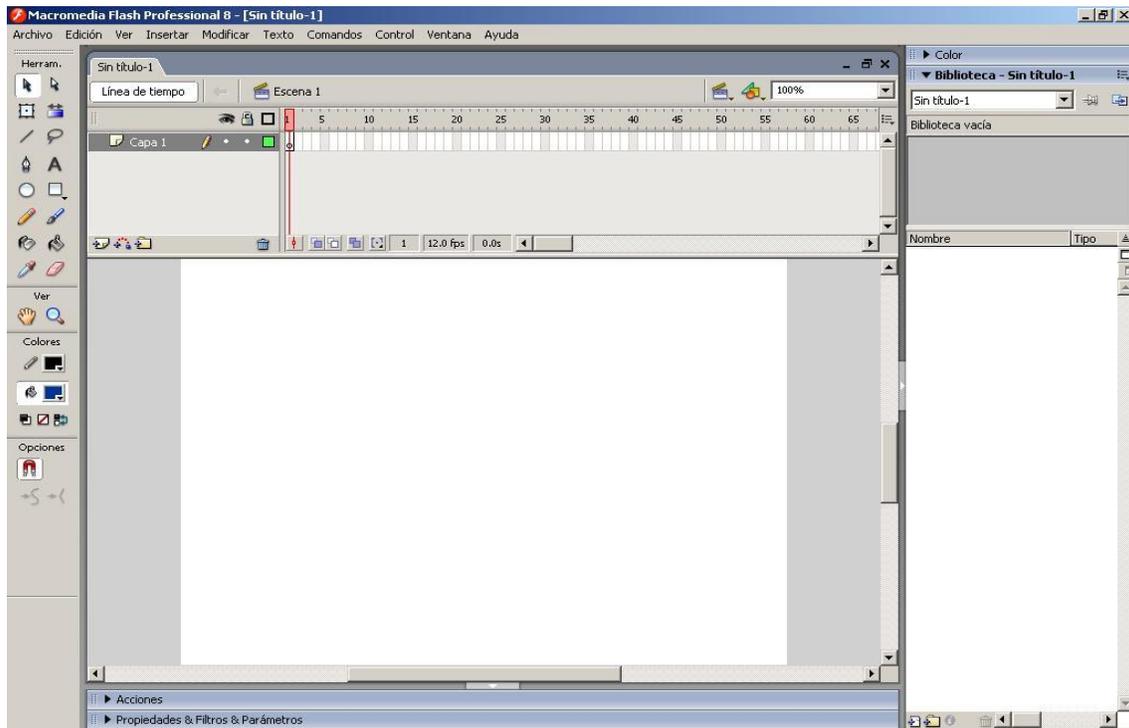


Fig. 6 Inspector de Propiedades

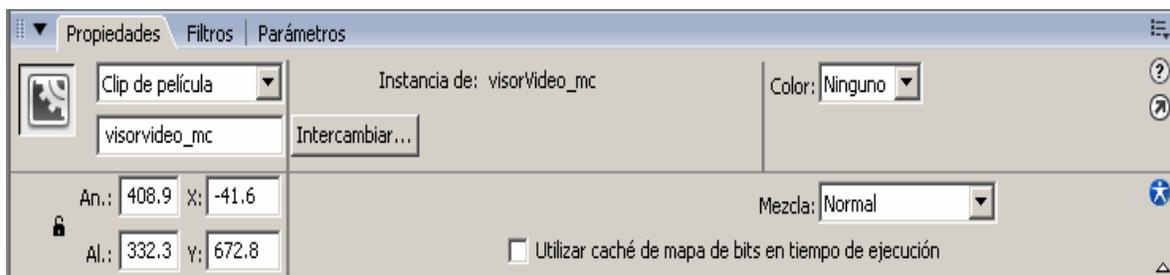


Fig. 7 Modelo Vista Controlador (MVC).

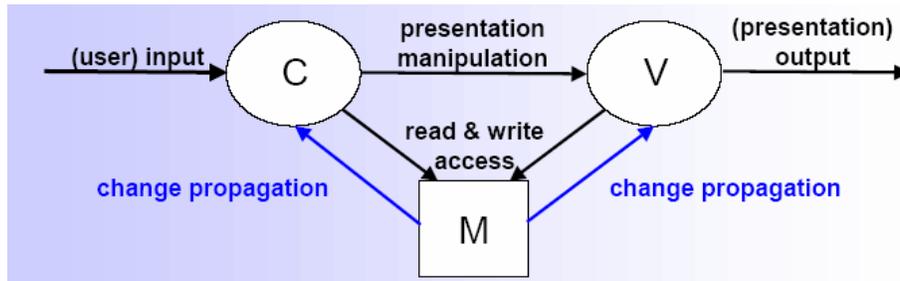
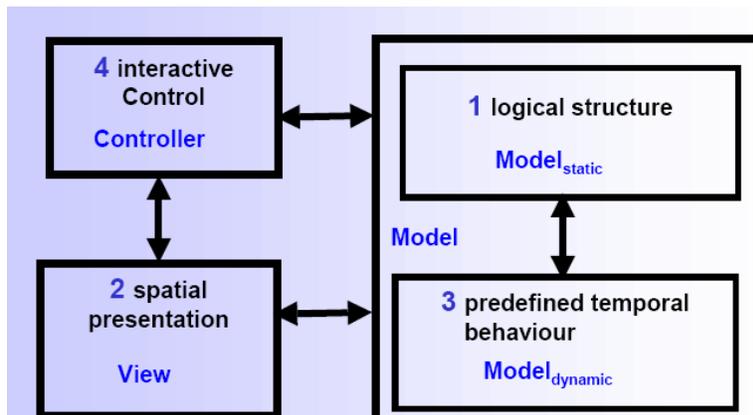
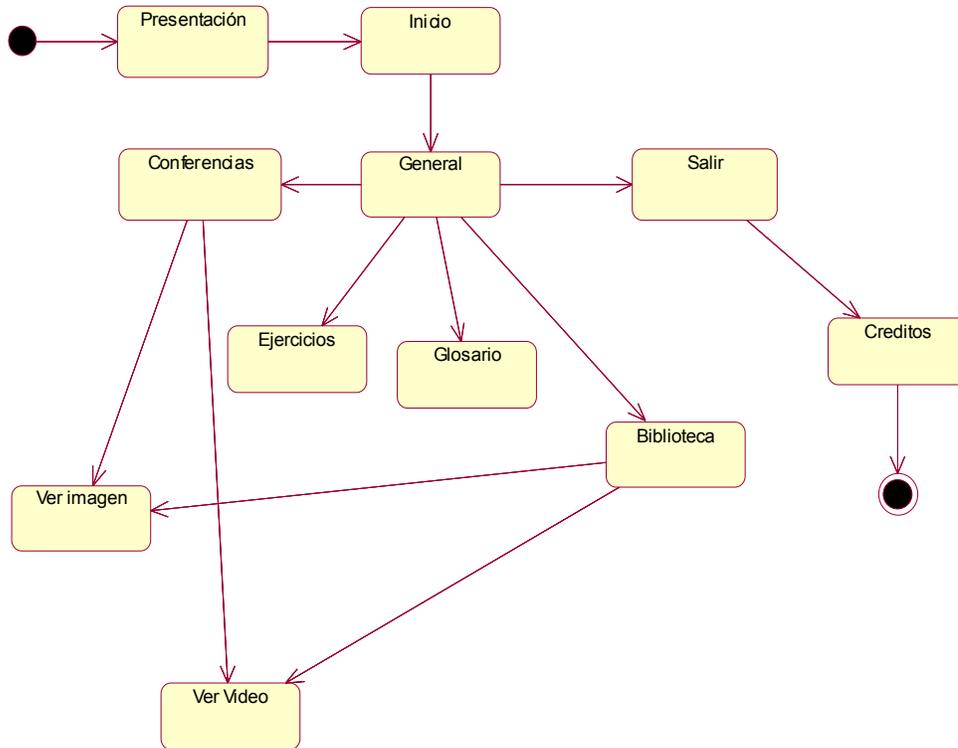


Fig.8 Extensión del MCV (MCV_{mm}).



ANEXOS II

Fig.1 Diagrama de Navegación.



Diagramas de CU del Sistema.
Fig.2 Diagrama de CU Ver Contenido.

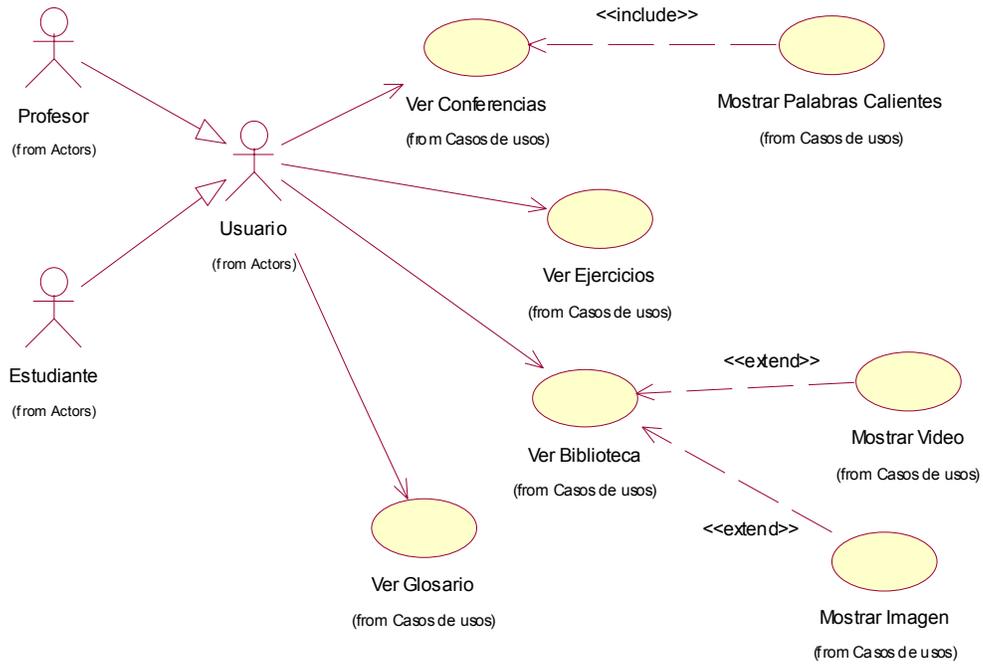


Fig.2.1 Diagrama de CU Accionar Palabras Calientes.

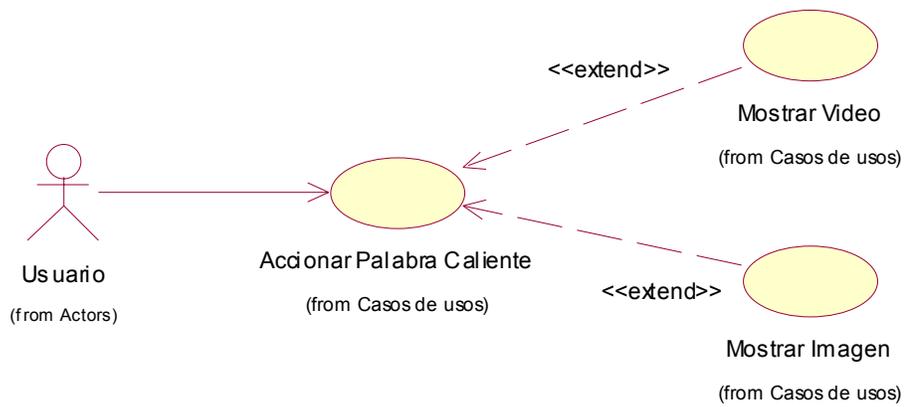


Fig.2.2 Diagrama de CU Controlar Sonido, Imprimir Contenido y Ver Ayuda.

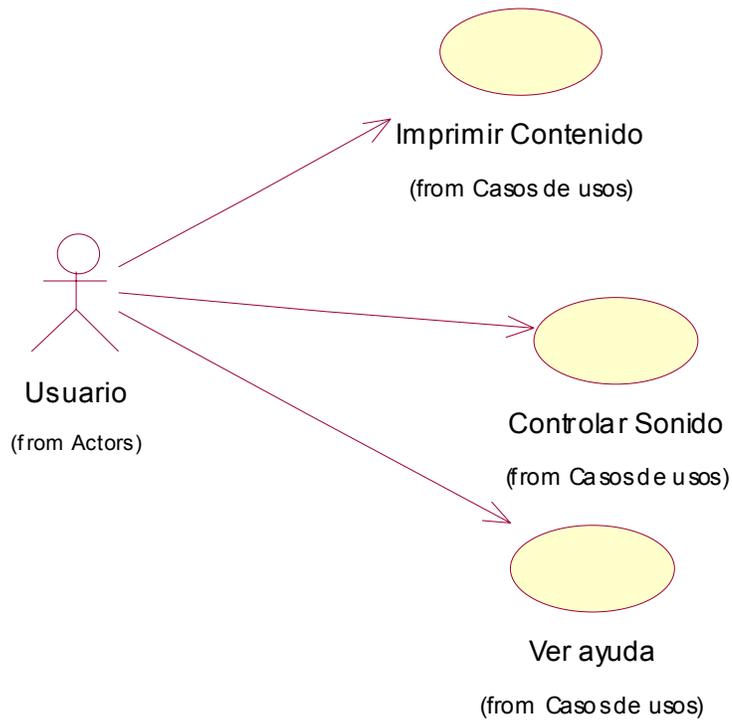


Fig.2.3 Diagrama de CU Modificar Contenido.

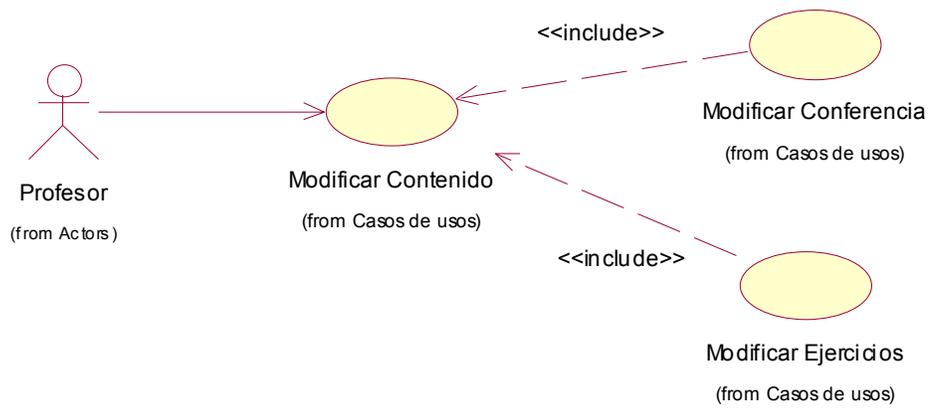


Fig.3 Diagrama de Clases del Modelo de Objetos.

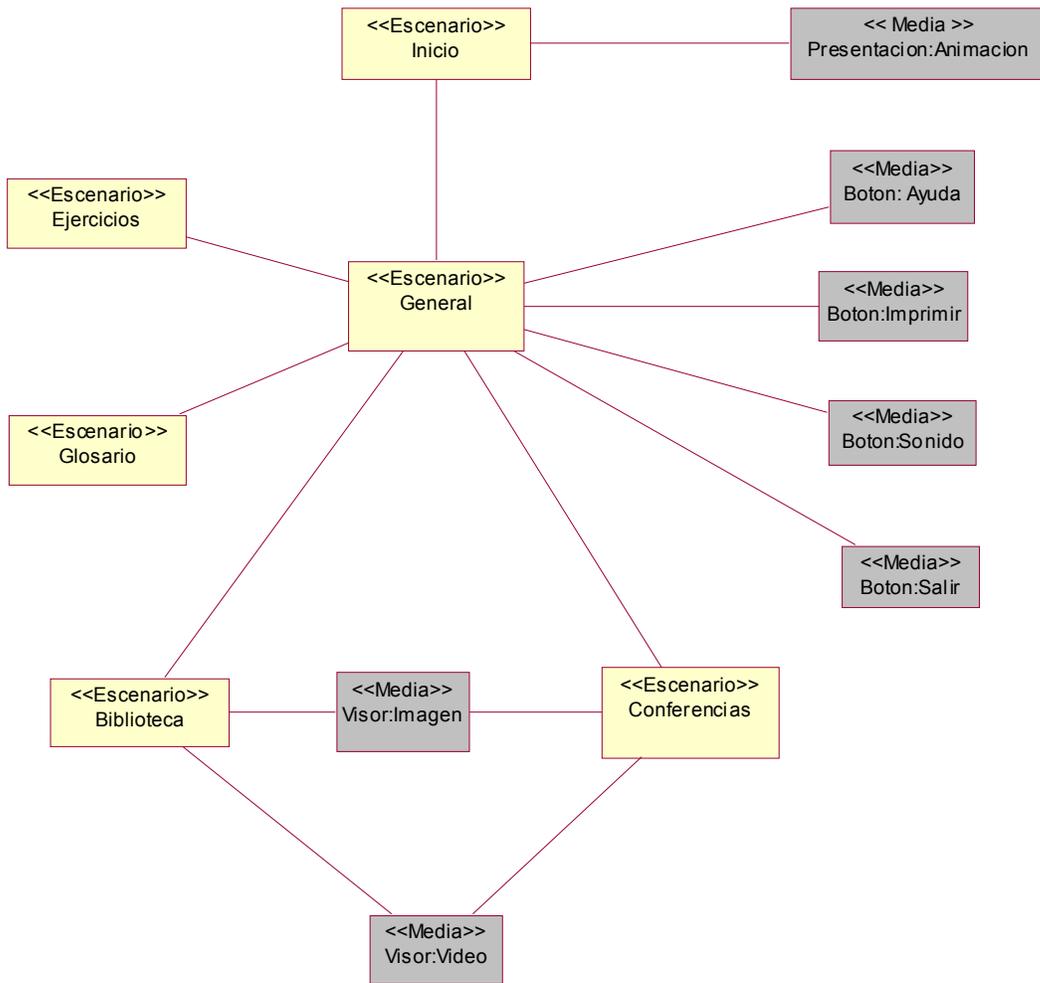


Fig.5 Diagrama de Presentación del Escenario General.

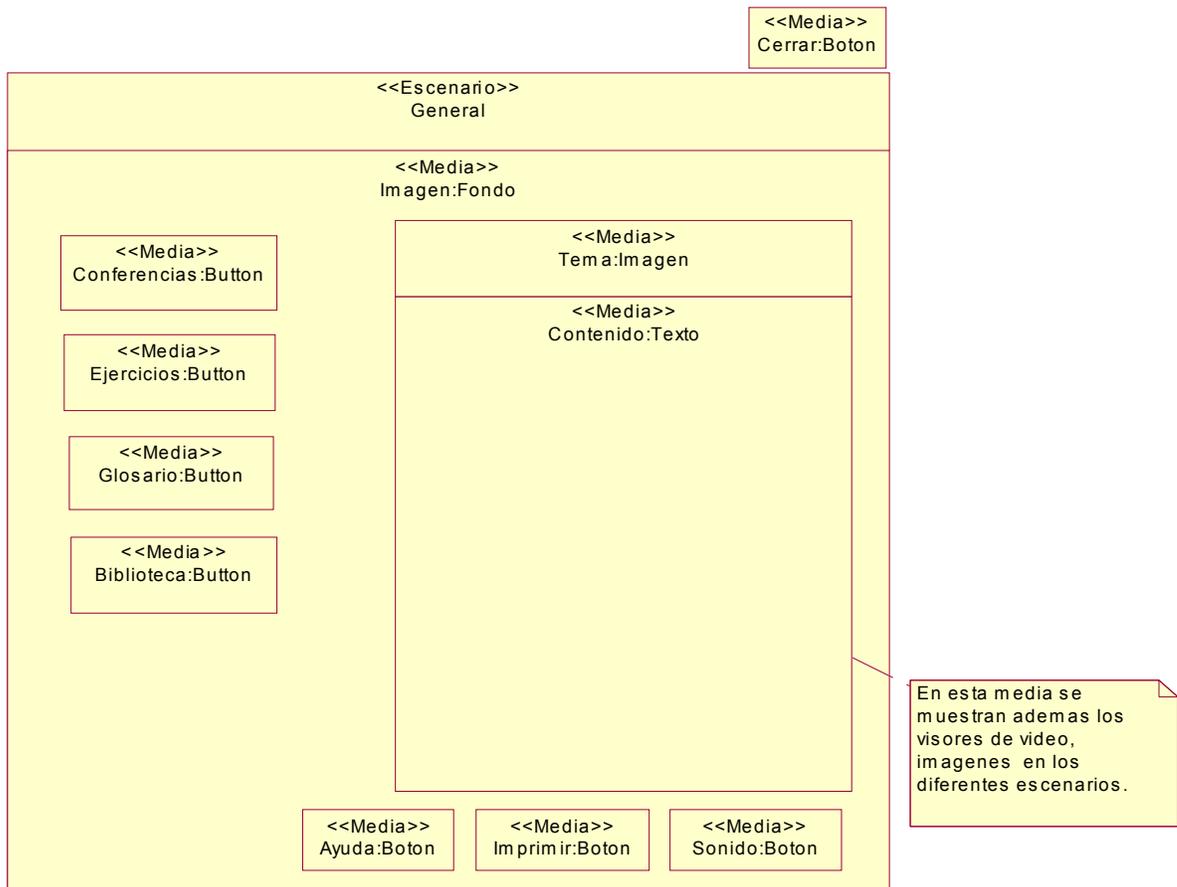
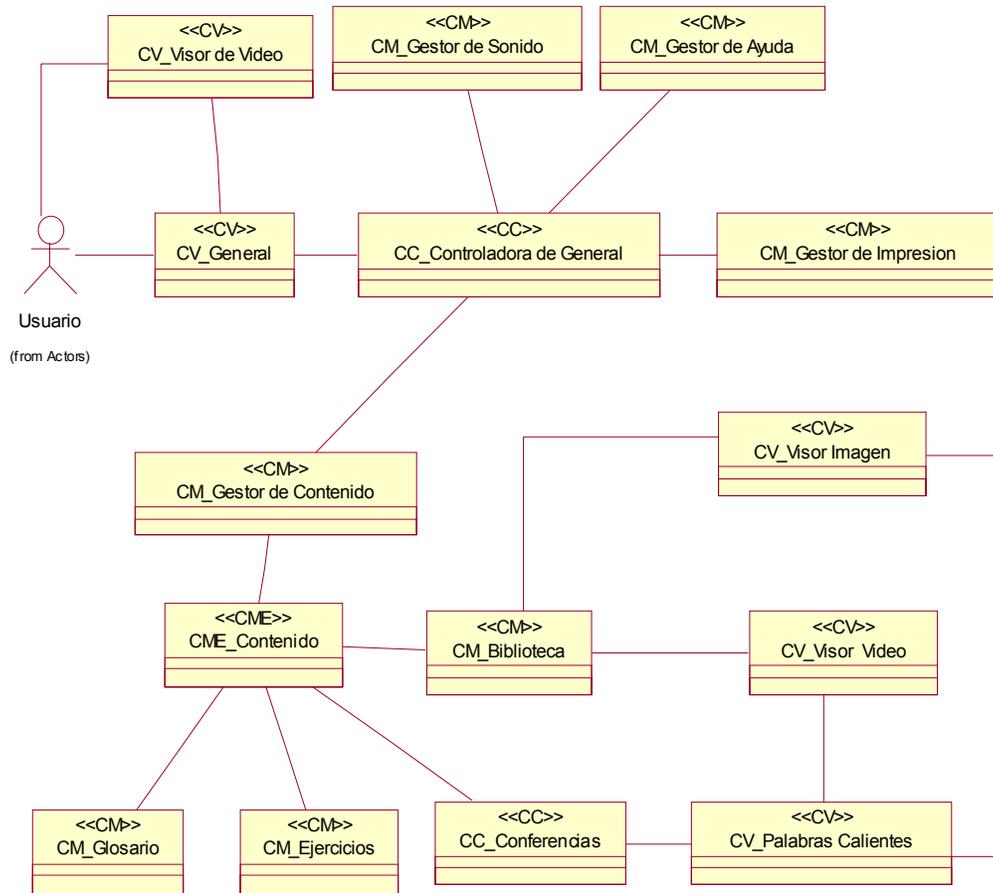


Fig.6 Diagrama de Clases de Análisis General.



Diagramas de Clases de Diseño.

Fig.7 Diagrama de Clases de Diseño de Escenario General.

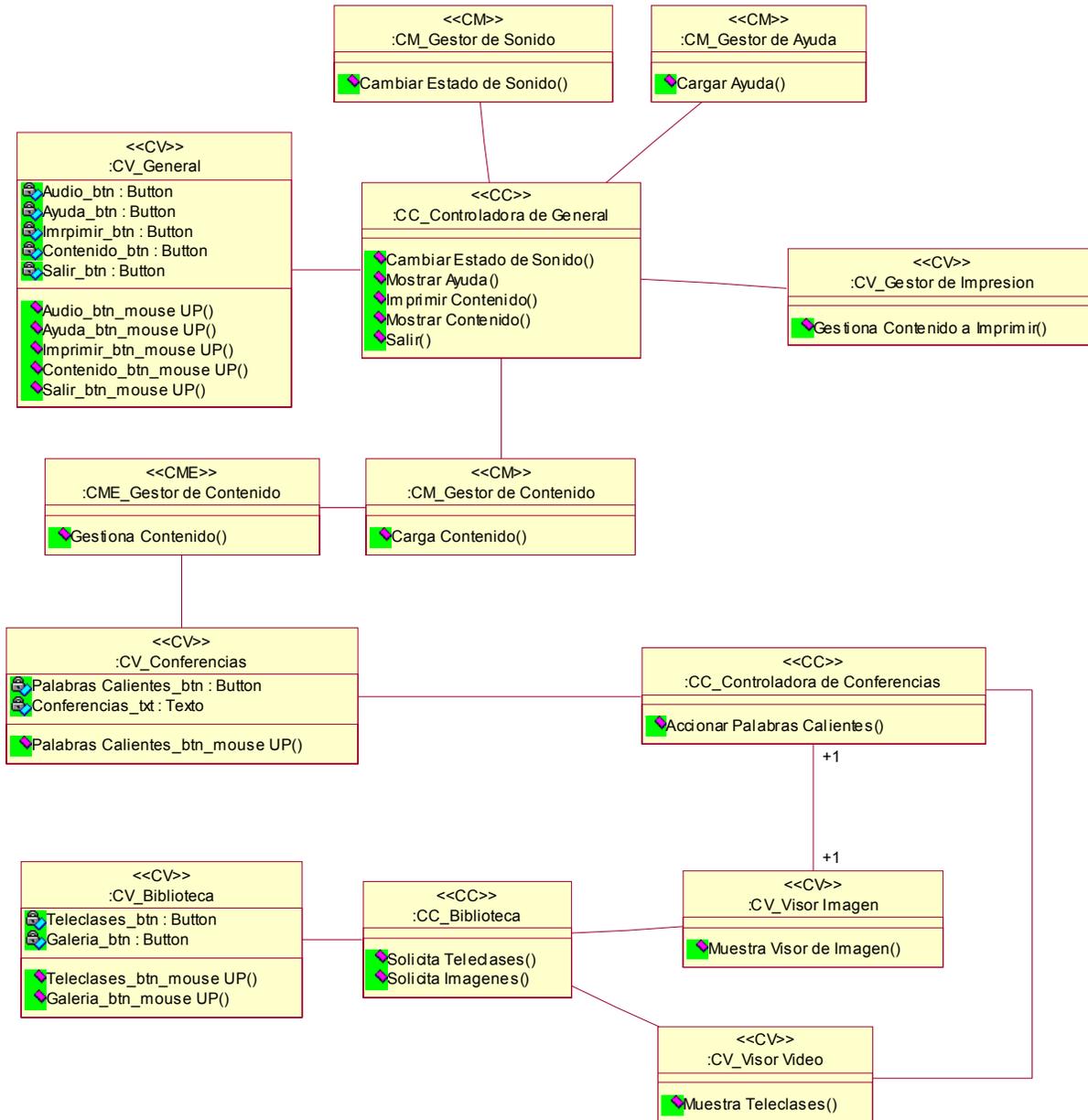


Fig.7.1 Diagrama de Clases de Diseño. Mostrar Conferencias.

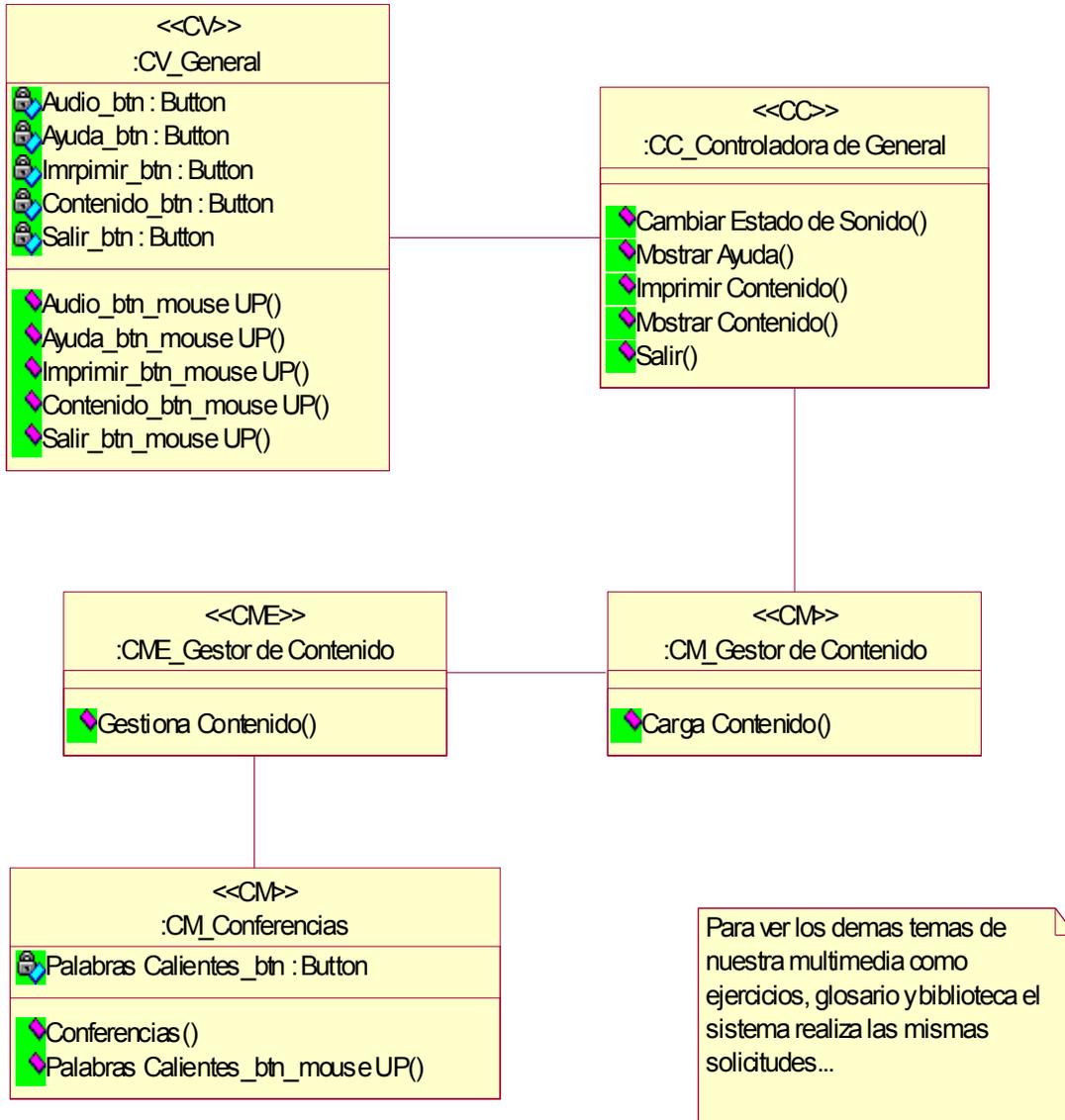


Fig.7.2 Diagrama de Clases de Diseño. Accionar palabras Calientes.

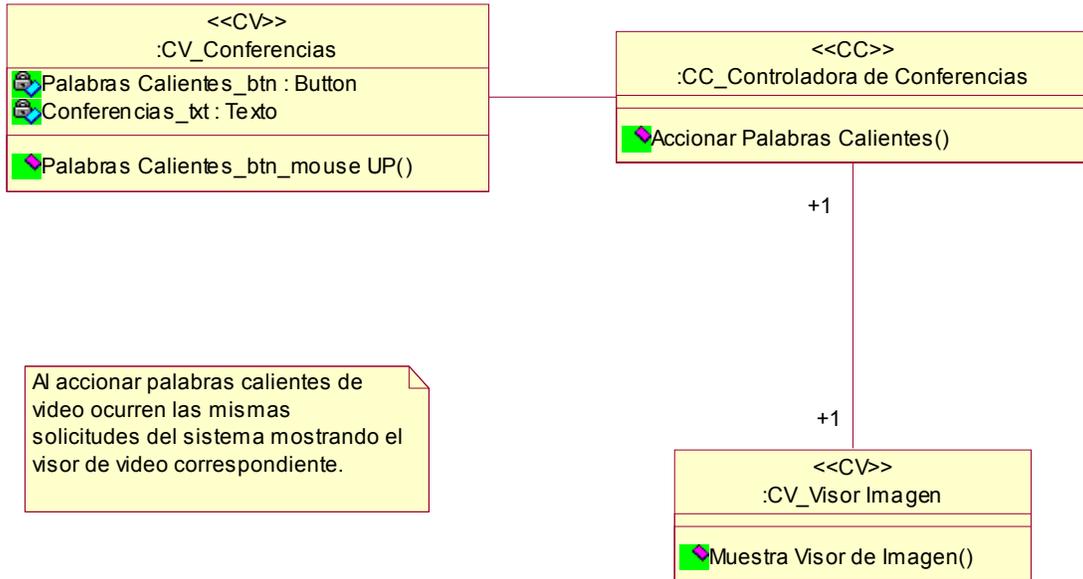


Fig.7.3 Diagrama de Clases de Diseño. Controladora de Sonido.

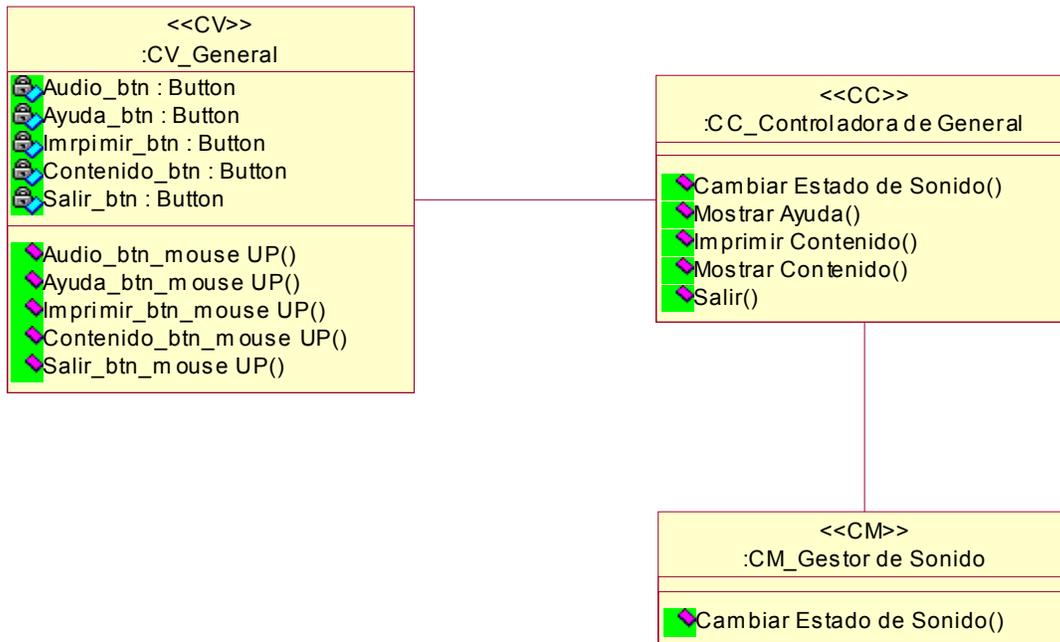


Fig.7.4 Diagrama de Clases de Diseño. Mostrar Ayuda.

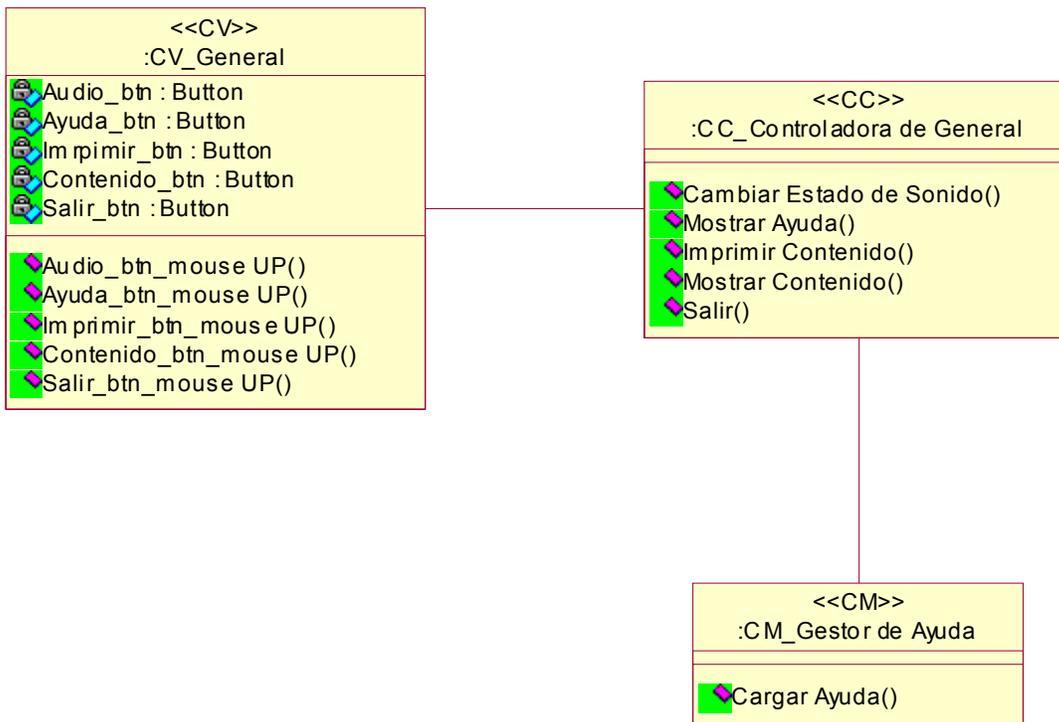


Fig.7.5 Diagrama de Clases de Diseño. Mostrar Video.

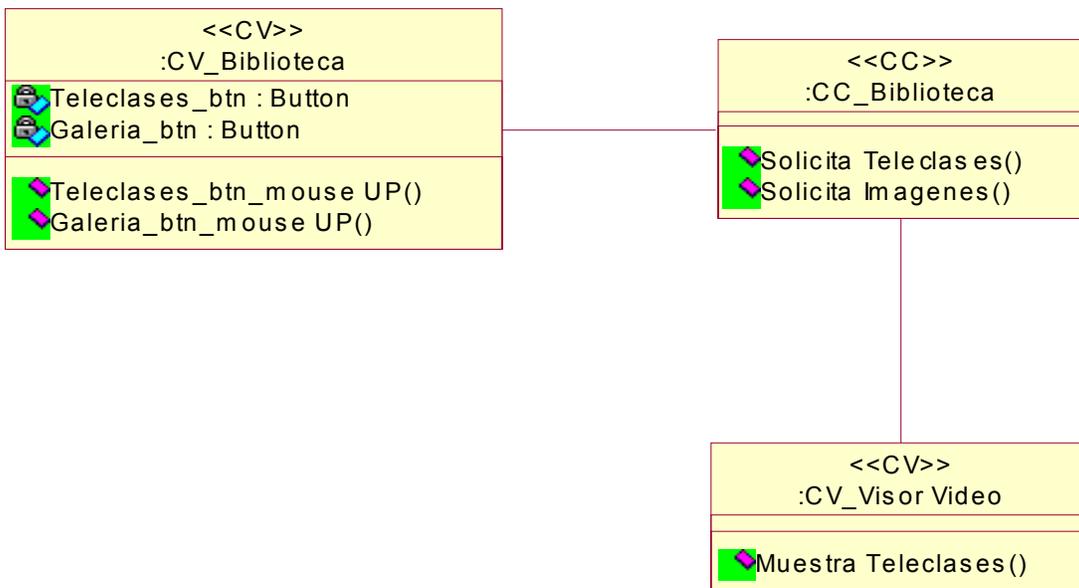


Fig.7.6 Diagrama de Clases de Diseño. Imprimir Contenido.

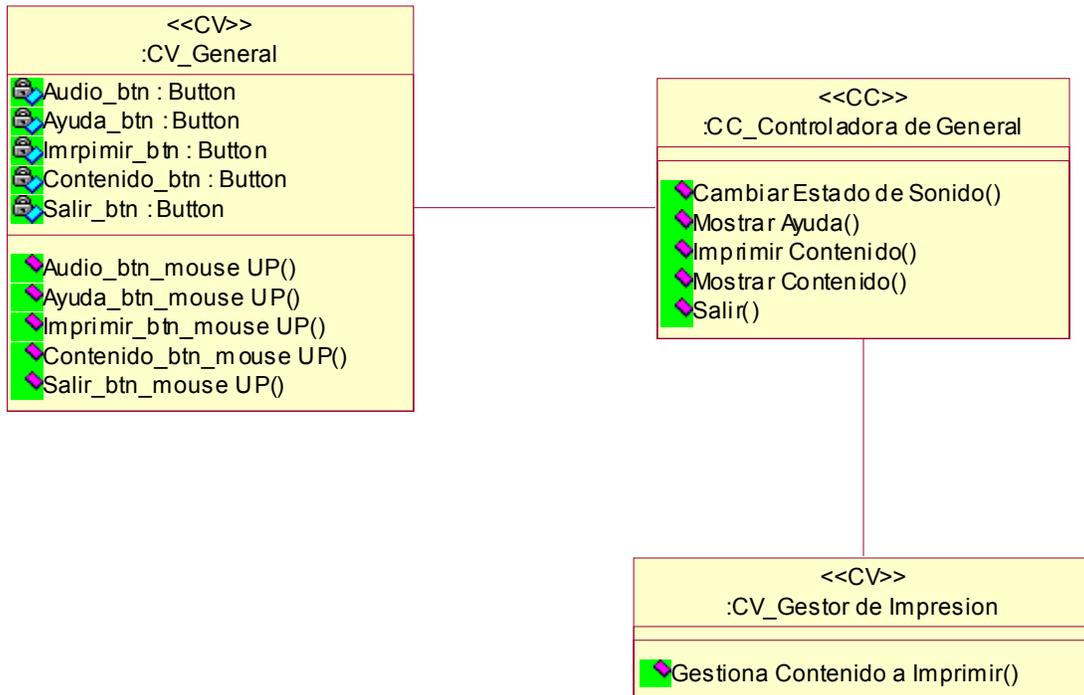


Diagrama de Colaboración.

Fig.8 Diagrama de Colaboración: Ver Conferencias.

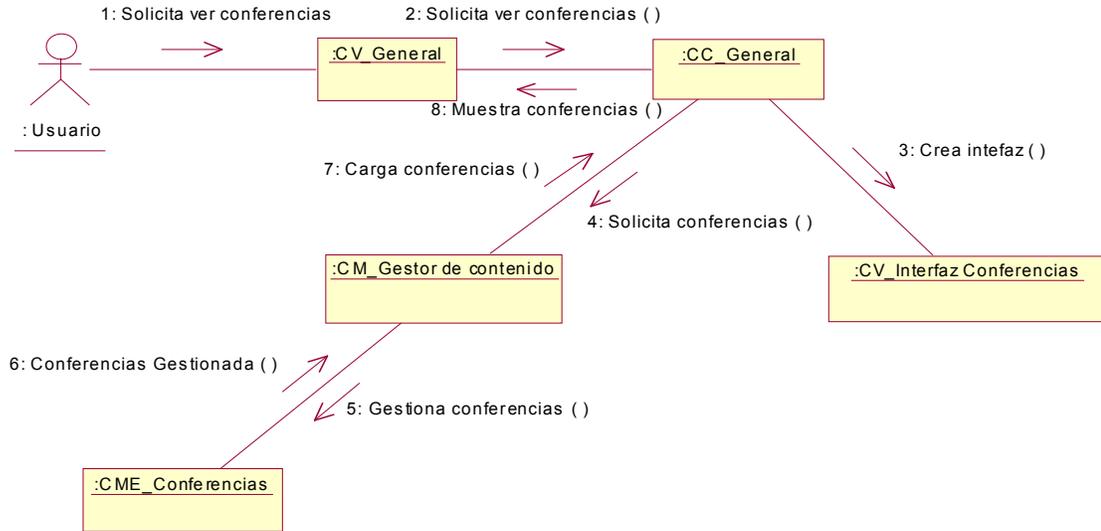


Fig.8.1 Diagrama de Colaboración: Mostrar Palabras Calientes.

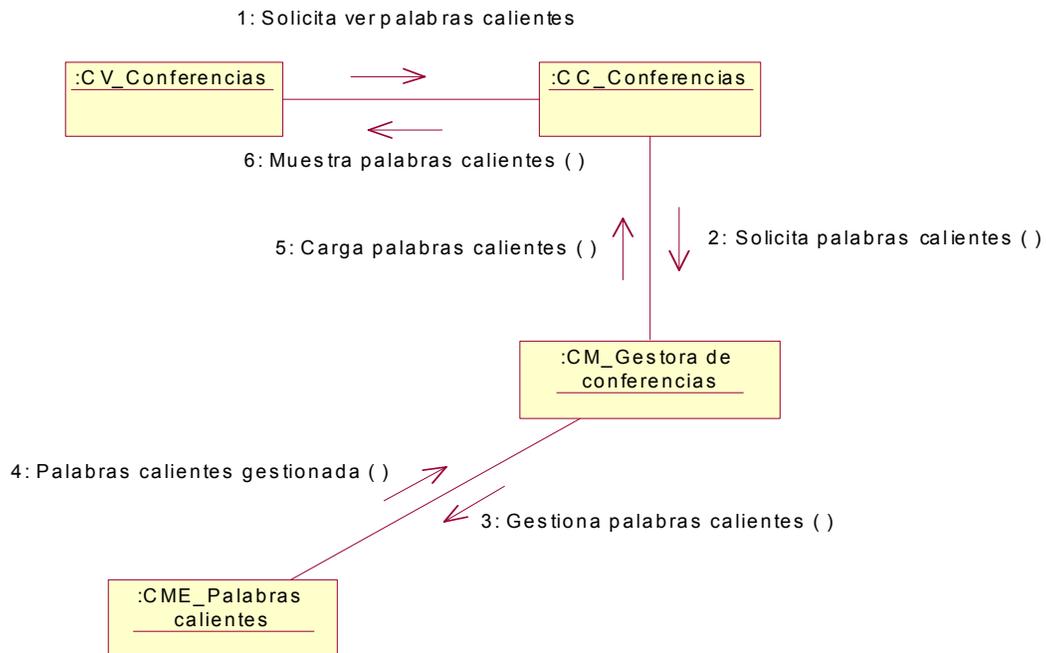


Fig.8.2 Diagrama de Colaboración: Accionar Palabras Calientes.

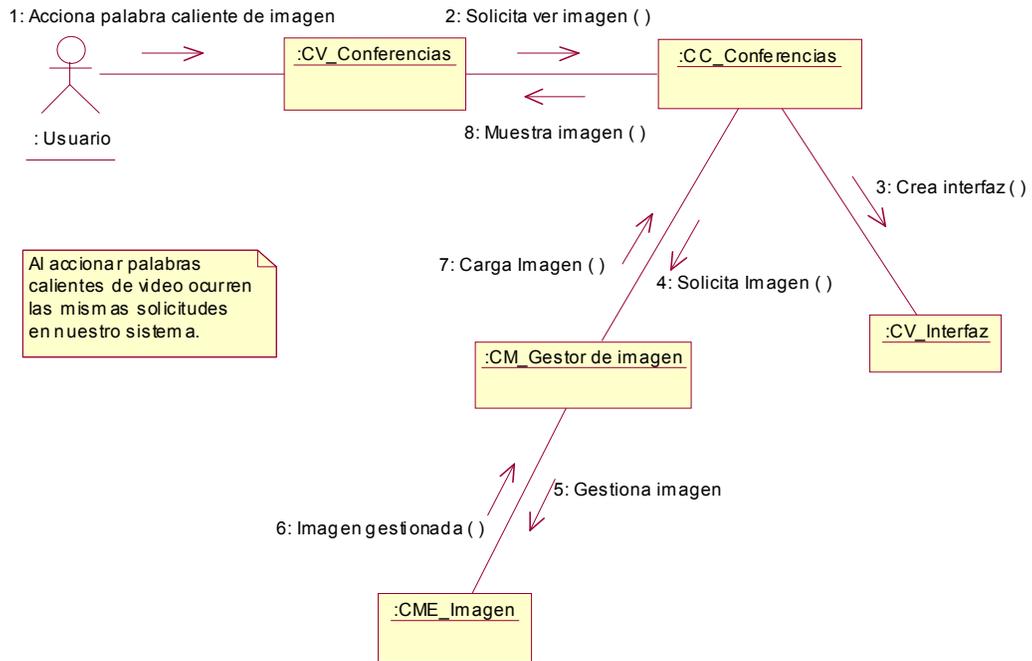


Fig.8.3 Diagrama de Colaboración: Mostrar Ayuda.

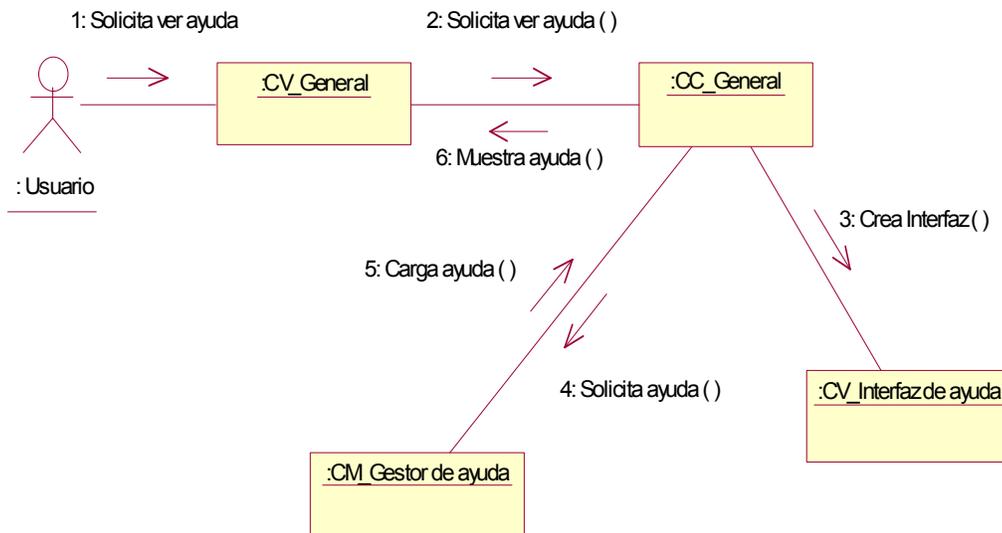


Fig.8.4 Diagrama de Colaboración: Controlar Sonido.

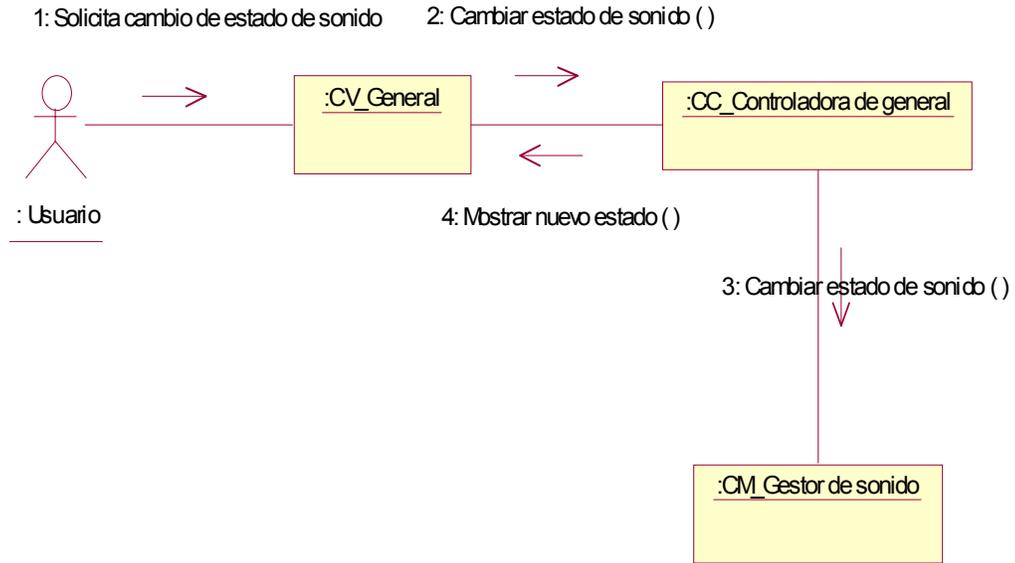


Fig.8.5 Diagrama de Colaboración: Imprimir Contenido.

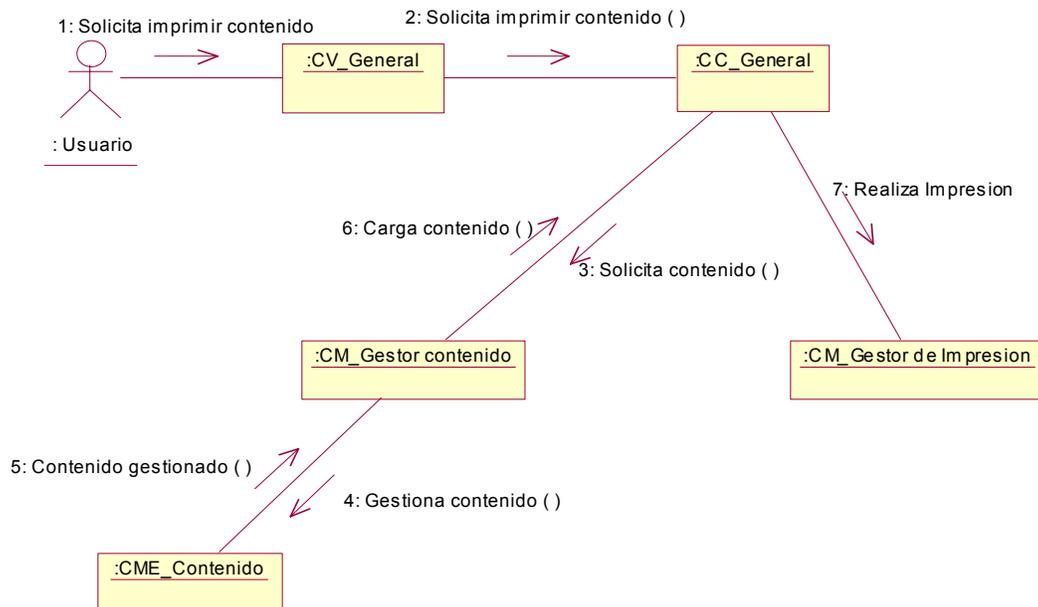
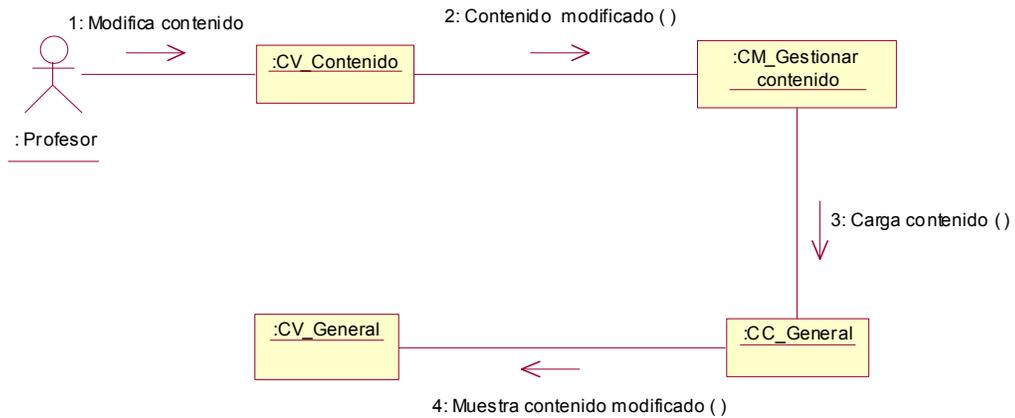


Fig.8.6 Diagrama de Colaboración: Mostrar Video.



Fig.8.7 Diagrama de Colaboración: Modificar Contenido.



Diagramas de Secuencias.

Fig.9 Diagrama de Secuencia: Ver Conferencias.

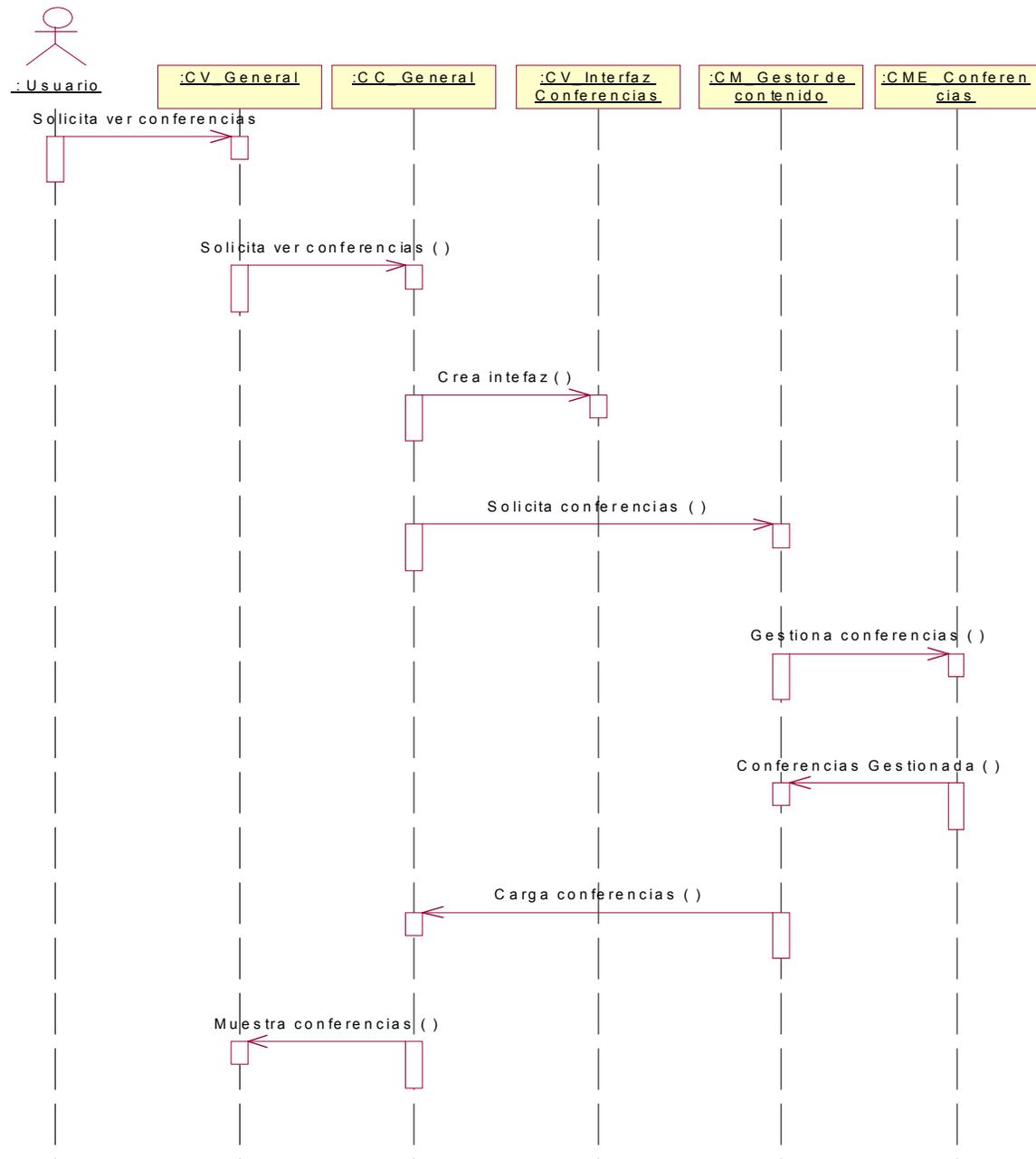


Fig.9.1 Diagrama de Secuencia: Mostrar Palabras Calientes.

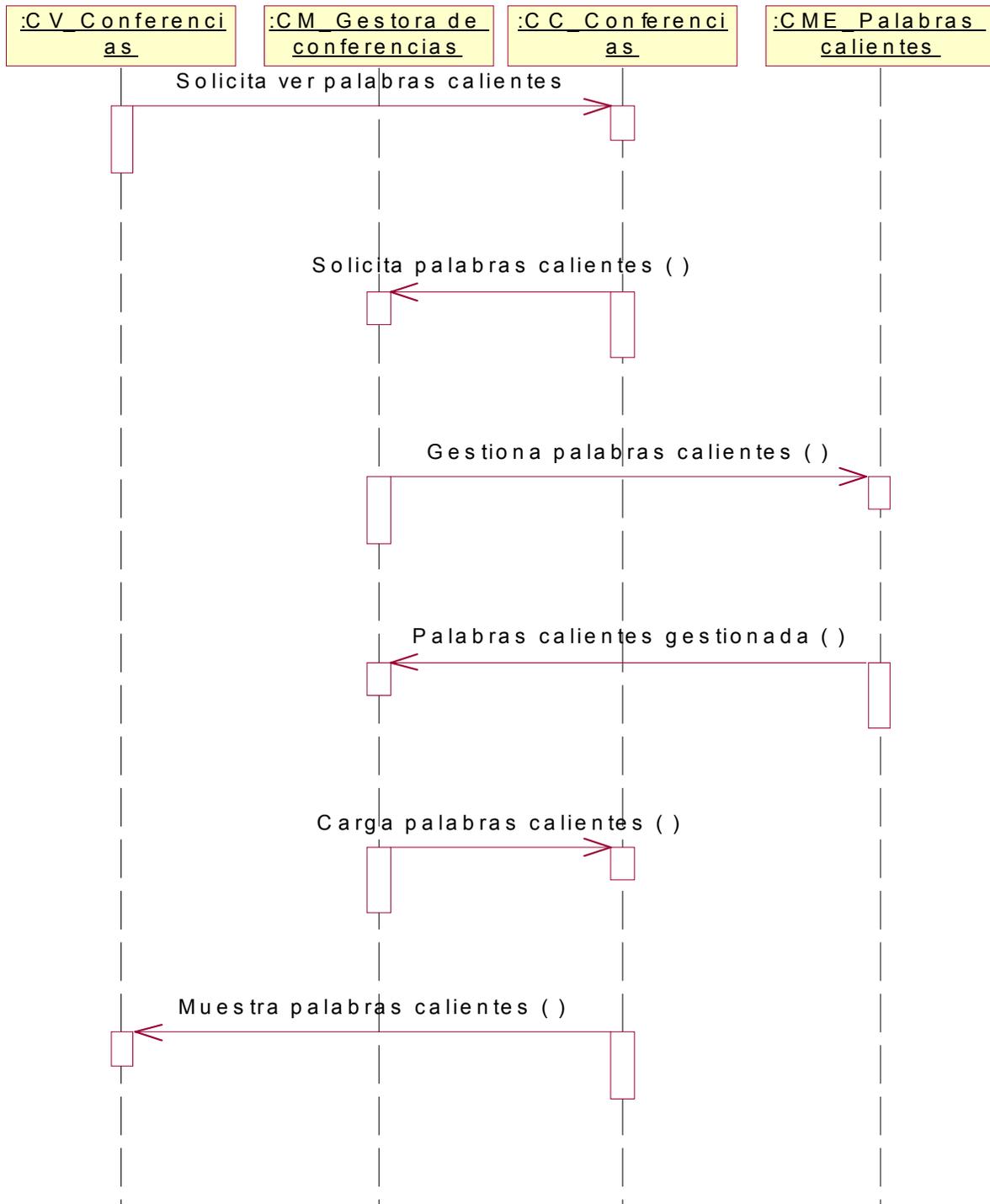


Fig.9.2 Diagrama de Secuencia: Accionar Palabras Calientes.

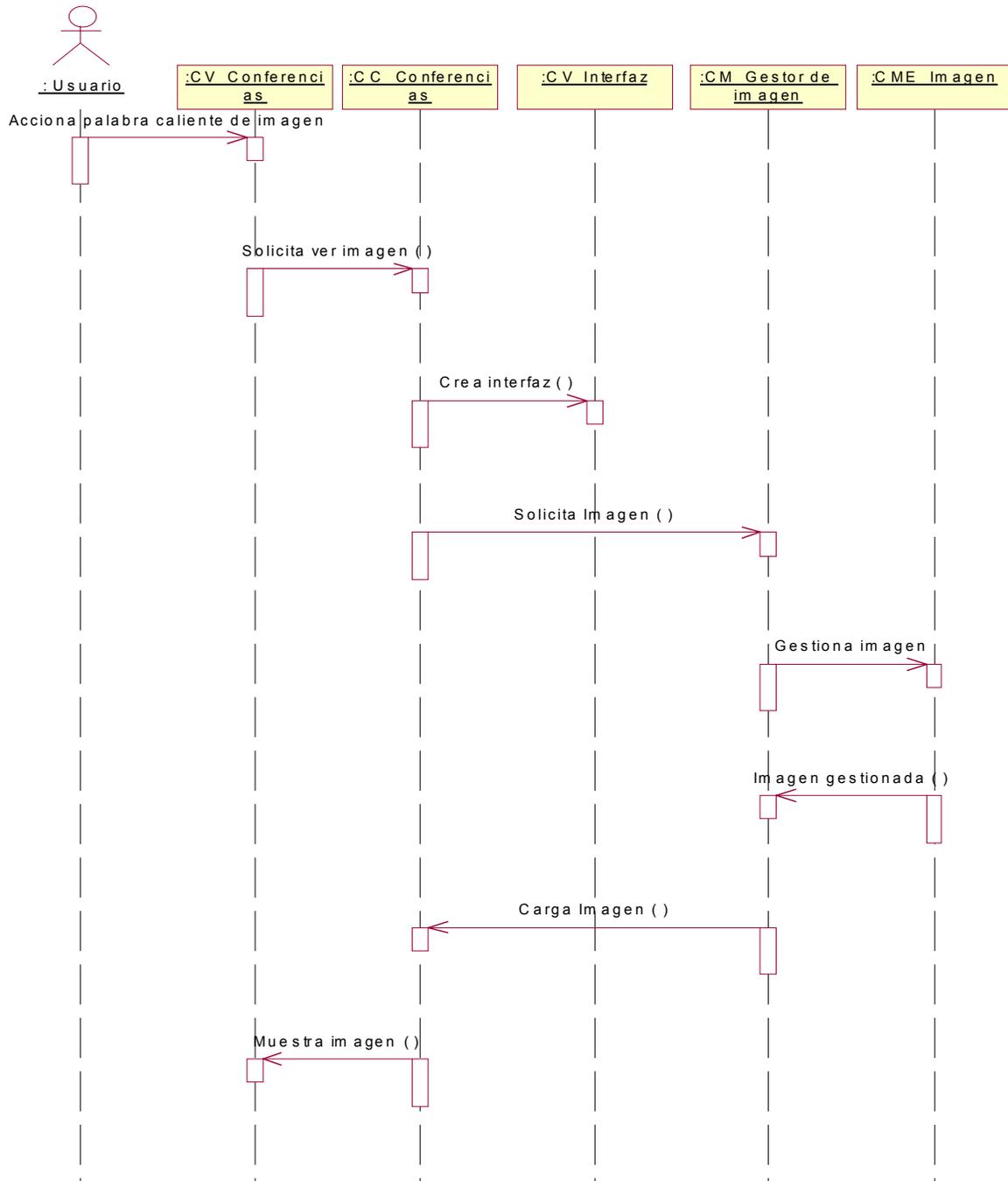


Fig.9.3 Diagrama de Secuencia: Mostrar Ayuda.

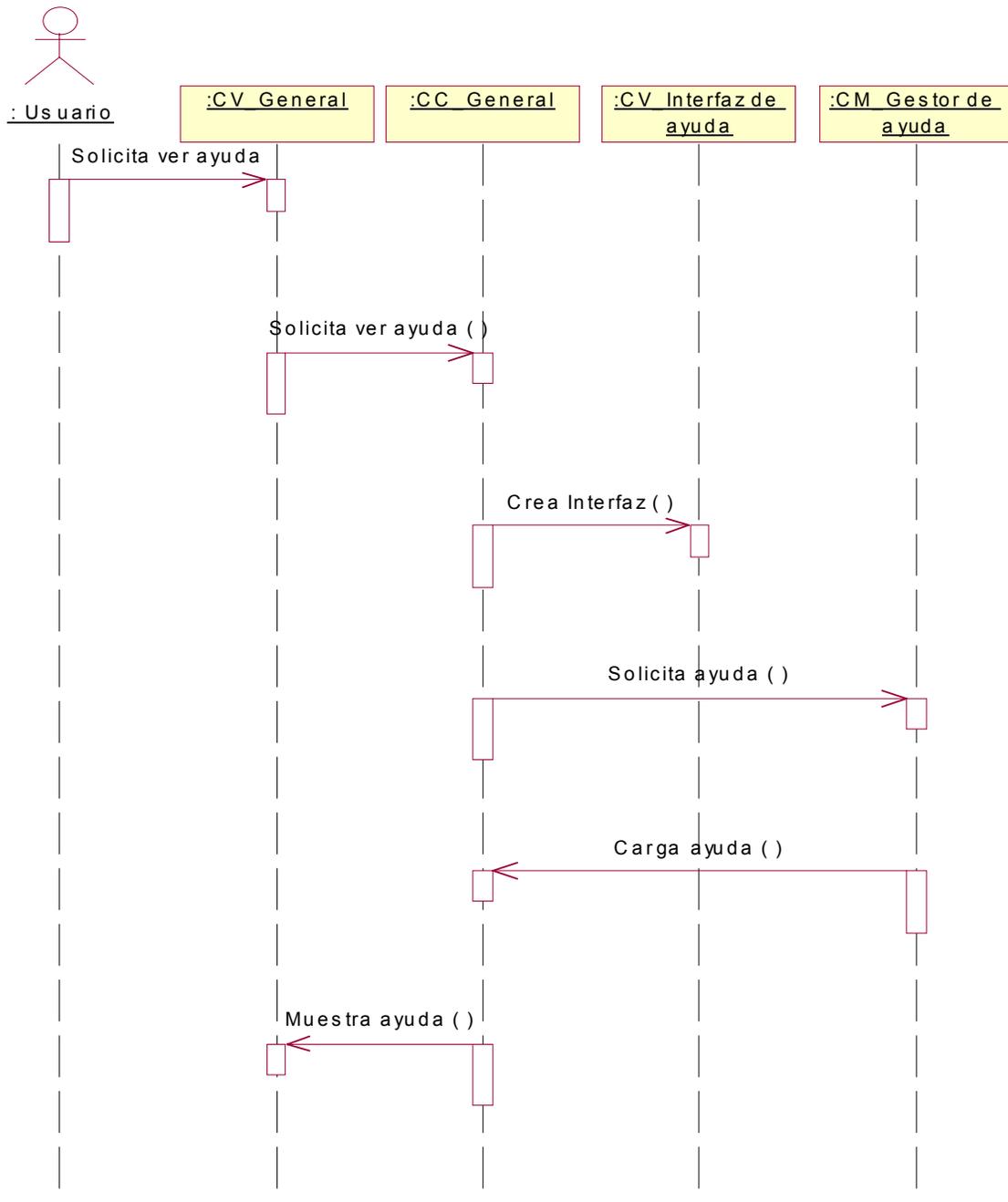


Fig.9.4 Diagrama de Secuencia: Controlar Sonido.

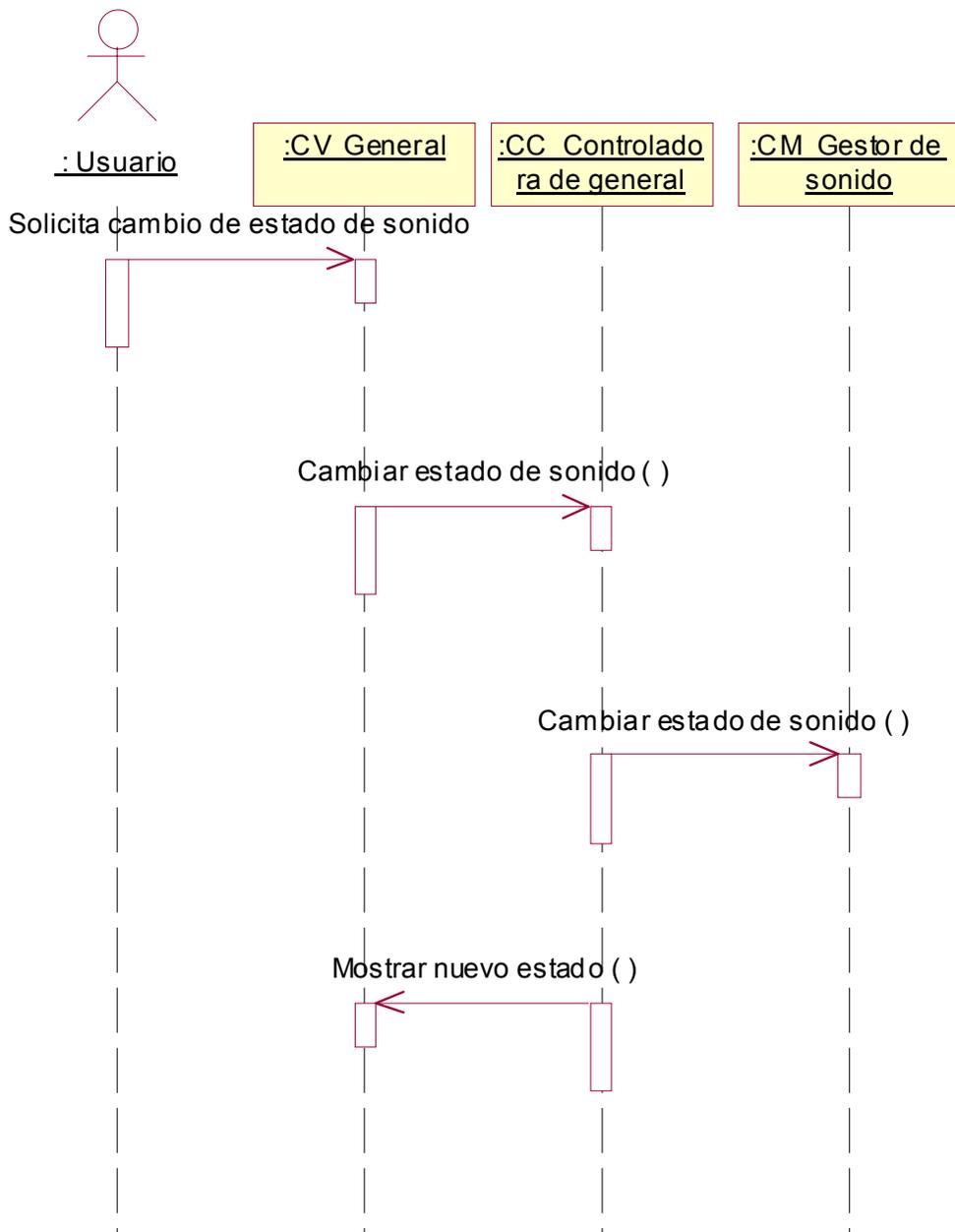


Fig.9.5 Diagrama de Secuencia: Imprimir Contenido.

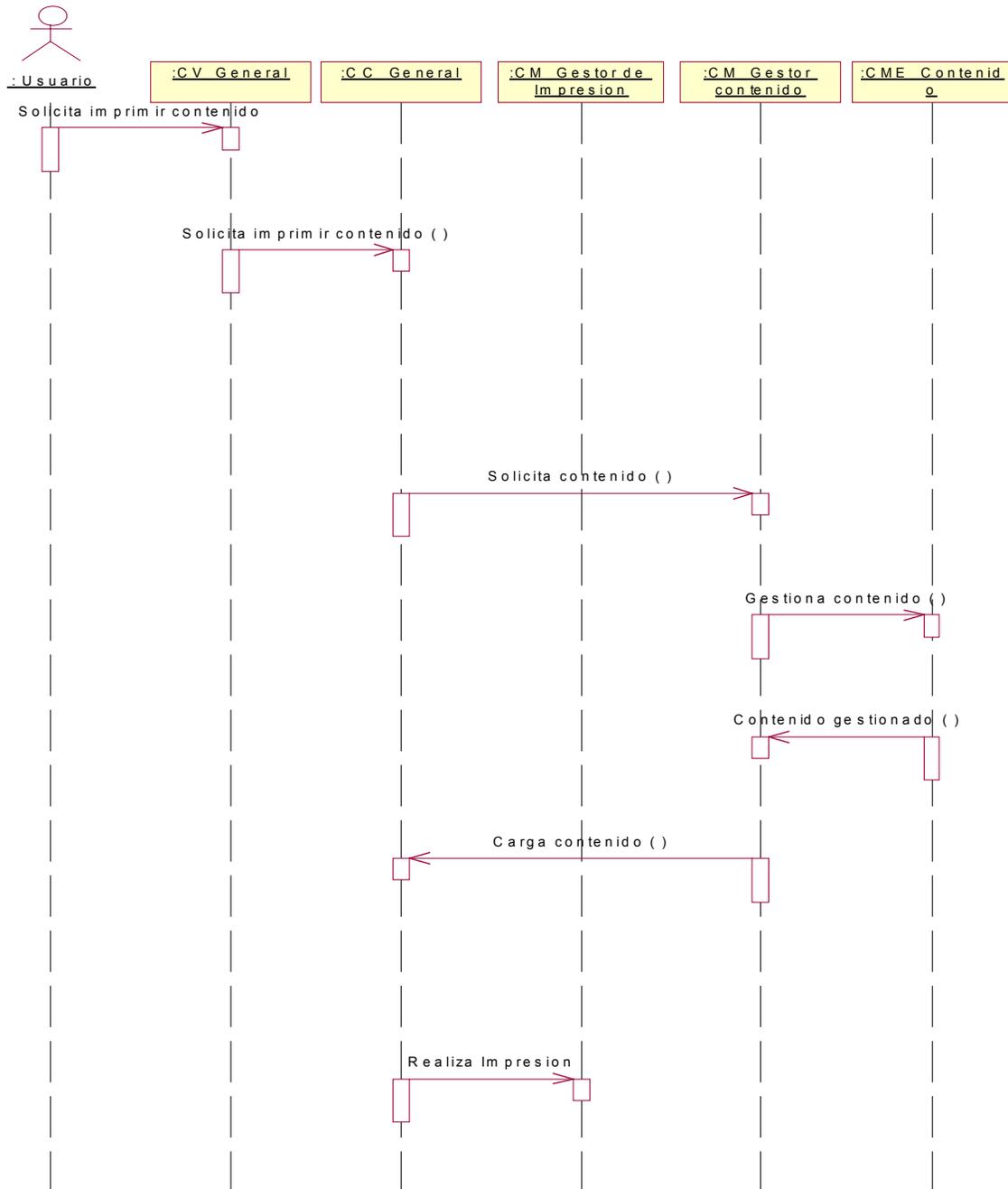


Fig.9.6 Diagrama de Secuencia: Mostrar Video.

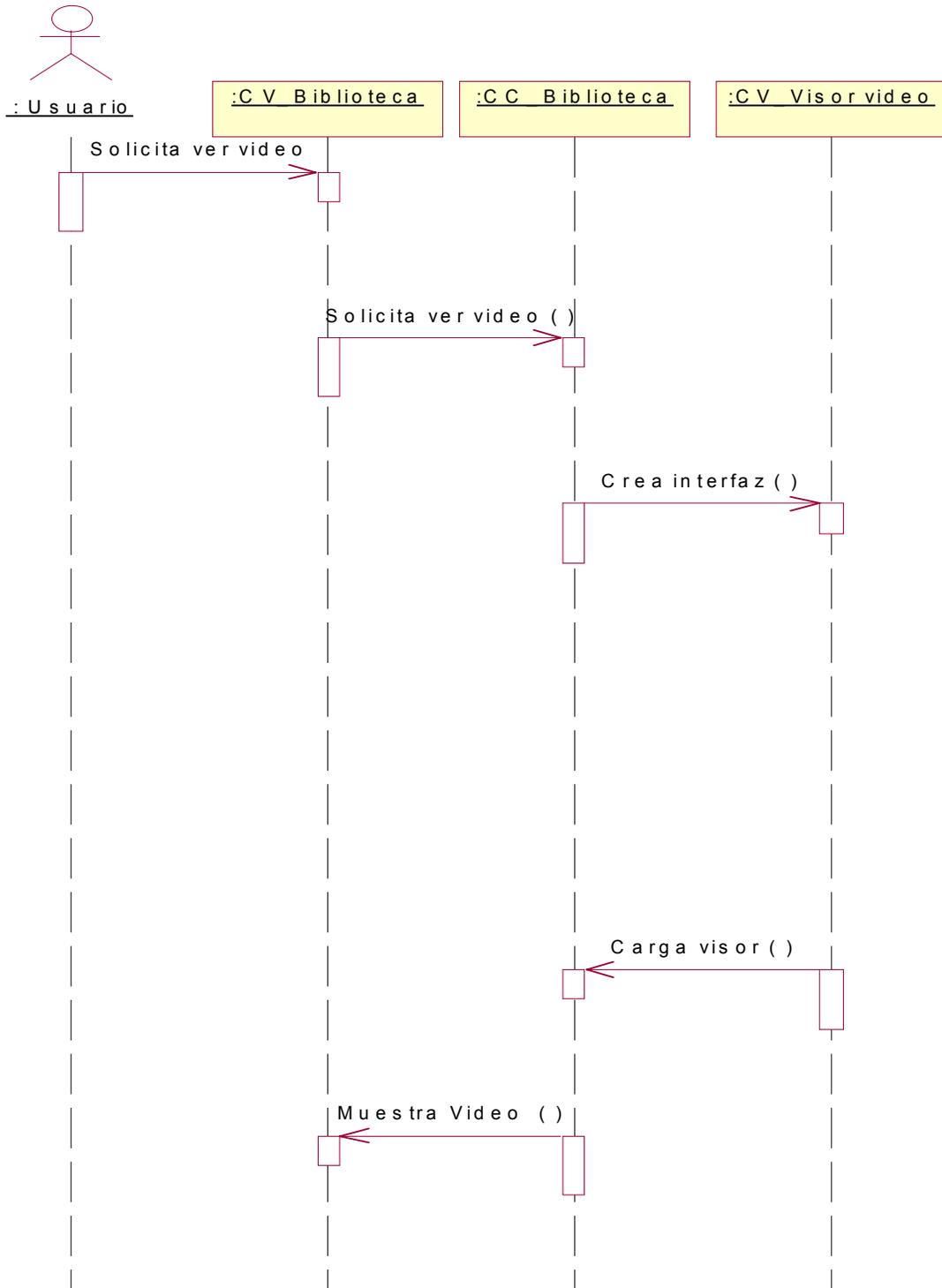
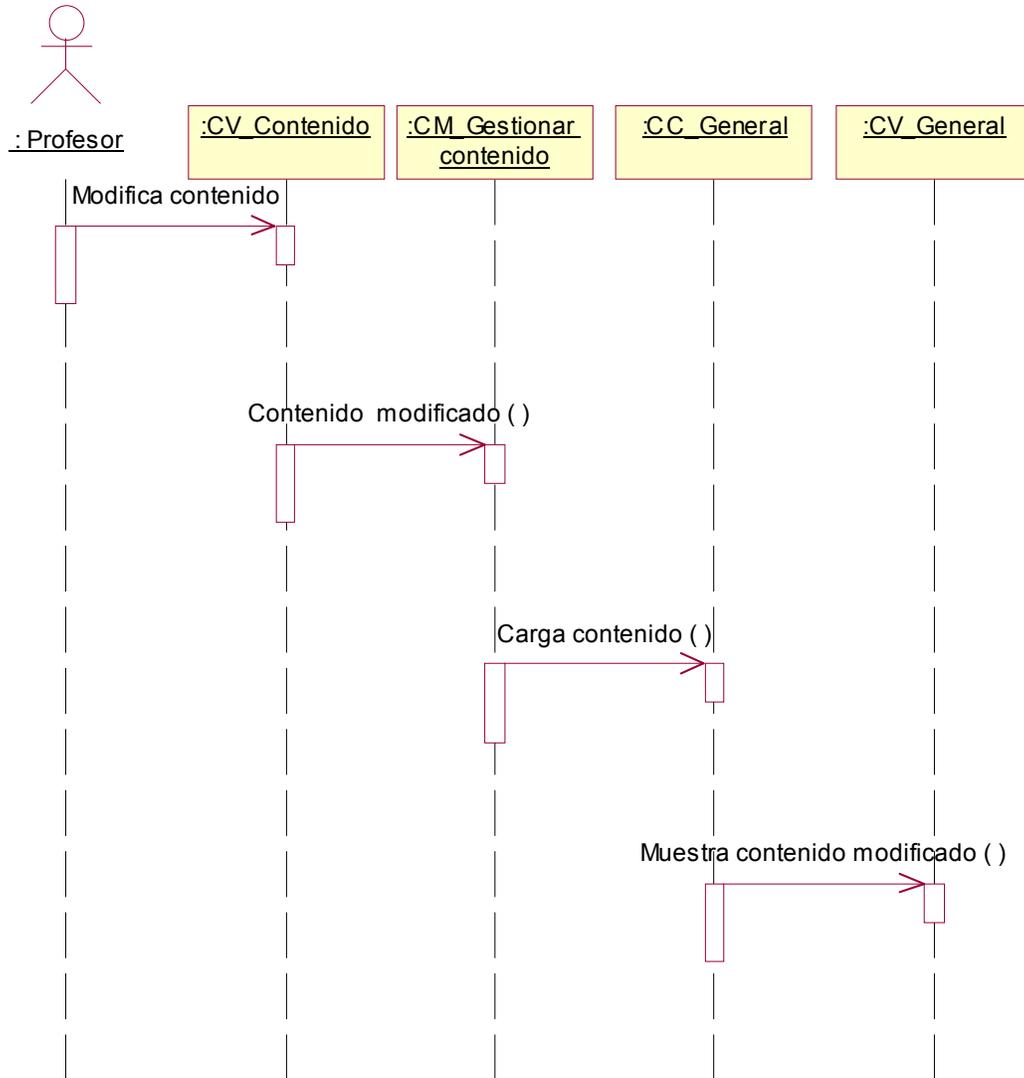


Fig.9.7 Diagrama de Secuencia: Modificar Contenido.



Diagramas de Componentes.

Fig.10 Diagrama de Componentes.

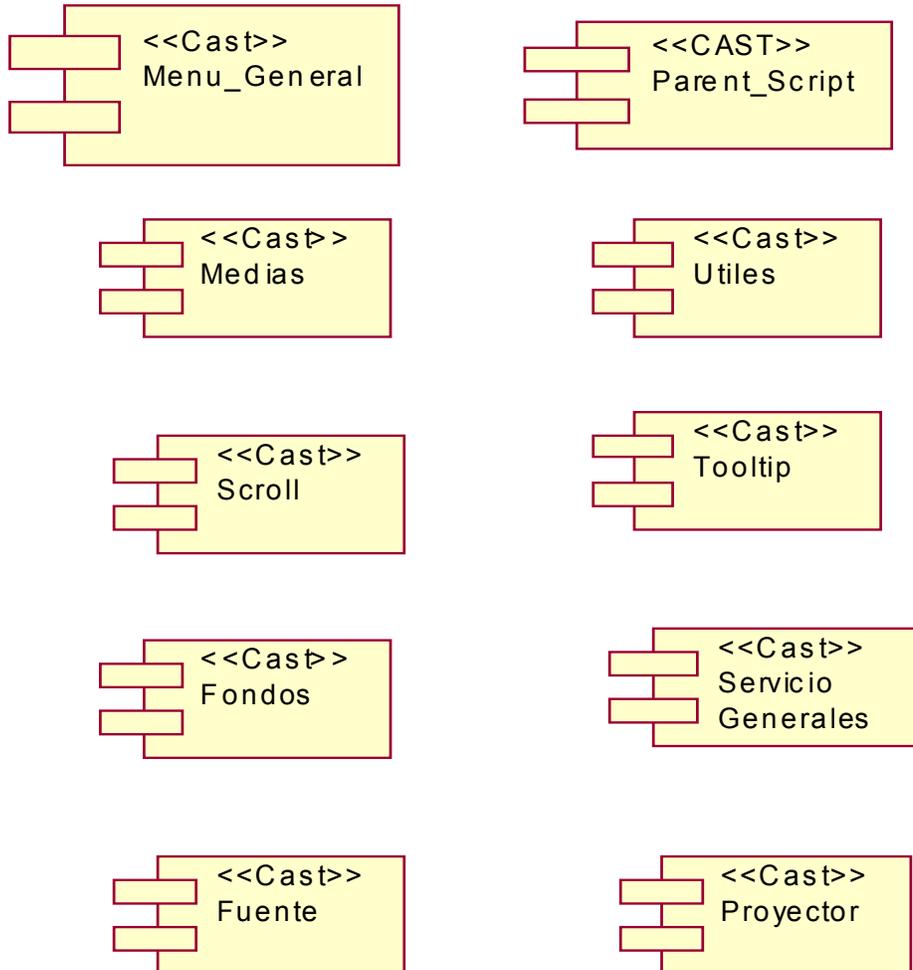
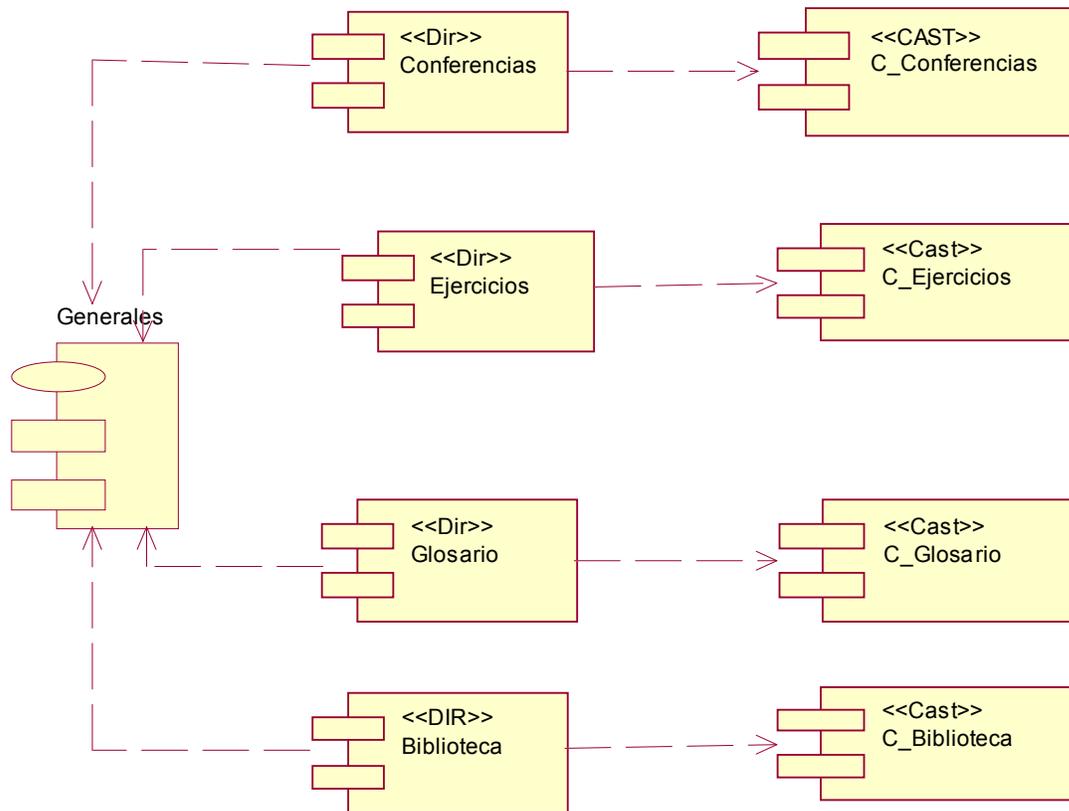


Fig.10.1 Diagrama de Componentes Generales.



ANEXO III
COCOMO V2

Peticiones

Nombre de la petición	Cantidad de ficheros	Cantidad de Elementos de datos	Clasificación(Simple, Media y compleja)
Mostrar conferencia	16	16	Alto
Mostrar clase práctica	16	16	Alto
Mostrar glosario	1	1	Bajo
Mostrar teleclase	16	16	Alto

• **Puntos de Función desajustados**

Elementos	Simple	X Peso	Medios	X Peso	Complejos	X Peso	Subtotal de puntos de función
Ficheros lógicos internos	0	7	0	10	0	15	0
Ficheros de interfaces externas	0	5	0	0	0	0	0
Entradas externas	0	3	0	4	0	6	0
Salidas externas	0	4	0	5	0	7	0
Peticiones	1	3	0	4	48	6	291
Total							291

➤ Cálculo de las instrucciones fuentes, esfuerzo, tiempo de desarrollo, cantidad de hombres y costo.

Características	Valor
Puntos de función desajustados	291
Lenguaje	10 HTML 90 Action Script
Instrucciones fuentes por puntos de función	47 HTML 44 Action Script
Instrucciones fuentes	12891

Factores	Complejidad	Valor	Justificación
PREC	Alta	2.48	Hay precedencia.
FLEX	Muy Alta	2.03	La flexibilidad debe ser alta.
TEAM	Nominal	5.48	El equipo está cohesionado en alguna medida.
RESL	Nominal	4.24	Hay buena fortaleza en la arquitectura.
PMAT	Bajo	6.24	La madurez es baja.
Total		25.2	

Parámetro	Atributo	Clasificación	Valor	Justificación
Producto	RELY	Bajo	0.92	En caso de fallo habrá pérdidas bajas.
	DATA	Nominal	1.00	La relación de datos en bytes contra líneas de código está entre 10 y 100.
	DOCU	Nominal	1.00	Depende de las necesidades del ciclo de desarrollo.
	CPLX	Baja	0.73	La complejidad del producto es baja.
	RUSE	Nominal	1.07	La reusabilidad del código es óptima para otra aplicación similar.
Plataforma	TIME	Nominal	1.00	Usa menos del 50% de los recursos disponibles.
	STOR	Alto	1.05	Requiere de un espacio no muy pequeño de almacenamiento.
	PVOL	Bajo	0.87	Hay cambios de hardware y software cada 12 meses.
Personal	ACAP	Nominal	1.00	Capacidad nominal de los analistas.
	AEXP	Alto	0.88	Se requiere de programadores con experiencia.
	PCAP	Bajo	1.15	Difícil el trabajo en equipo.
	PEXP	Bajo	1.09	Poca experiencia en la plataforma.
	LTEX	Bajo	1.09	Baja experiencia con Action Script.
	PCON	Bajo	1.12	Se rota poco el personal.
Proyecto	TOOL	Nominal	1.00	Buen uso de herramientas.
	SCED	Nominal	1.00	Buen estimado en el calendario de desarrollo
	SITE	Nominal	1.00	Se ha usado el correo electrónico y el Chat para interactuar con los desarrolladores.
$\prod_{i=1}^n EM_i$			0.88400464	

$$PM = A \times Size^E \times \prod_{i=1}^n EM_i$$

$$E = B + 0.01 \times \sum_{j=1}^5 SF_j$$

$$E = 0.91 + 0.01 \times 25.2 = 0.22932$$

$$PM = 2.94 \times (12.891^{0.22932}) \times 0.88400464 = 4.670948 \approx 5$$

$$TDEV = C \times (PM)^F$$

$$F = D + 0.2 \times 0.01 \times \sum_{i=1}^5 SF_i$$

TDEV es el tiempo de desarrollo en meses, si se supone que se trabaje 152 horas al mes.

$$C = 3.67 \quad D = 0.28$$

$$F = 0.28 + 0.2 \times 0.01 \times 25.2 = 0.3304$$

$$TDEV = 3.67 \times 5^{0.3304} = 6.2460542 \approx 6$$

La cantidad de personas en el equipo debe ser: $PM/TDEV = 5 / 6 \approx 1$

Pero al ser 2 los integrantes del equipo se recalcula el tiempo de desarrollo de esta forma:

$$PM/2 = 5 / 2 = 2.5 \approx 3$$

•Cálculos

Cálculo de:	Valor
Esfuerzo	5 Pers-mes
Tiempo de desarrollo	6 mes
Cantidad de hombres	1
Costo	\$1325
Salario medio	\$225
RCPX	1.13
RUSE	1.07
PDIF	1
PREX	1
FCIL	1
SCED	1