

Universidad de Belgrano

Apellido y Nombres del Alumno: RE Javier Alberto

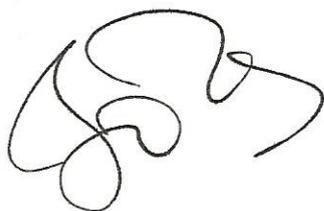
Carrera: Ingeniería en Informática

Número de Matrícula: 9135

Título de la Tesina: Creación de Presentaciones con una herramienta desarrollada en Flash MX 2004

Apellido y Nombres del Tutor: CAMPO PIOMBI Federico Germán

Firma del Alumno

A handwritten signature in black ink, consisting of several loops and a long horizontal stroke extending to the right.

Agradecimientos

Quiero agradecer a mi tutor por haber trabajado conmigo todo este tiempo.

Además a los docentes y compañeros de la Facultad de Ingeniería y Tecnología Informática.

También quiero agradecer a mi familia y amigos y amigas por estar junto a mí en mis malos y buenos momentos.

Resumen

En este documento se explica cómo se crea una aplicación de autoría y visualización de presentaciones, accesible a través de un navegador de internet, utilizando un enfoque bottom-up para su desarrollo.

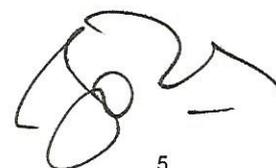


Índice

1. Primera Parte.....	8
1.1 Introducción	8
1.1.1 Planteamiento y Contexto del Problema	8
1.1.2 Situación Actual	9
1.1.3 Trabajos Relacionados	9
1.2 Objetivo.....	9
1.3 Justificación	9
1.4 Delimitaciones.....	10
1.4.1 Relacionadas con la metodología	10
1.4.2 Relacionadas con la funcionalidad de la aplicación.....	10
1.5 Organización del documento	10
1.6 Conceptos Fundamentales.....	10
1.6.1 Rich Internet Applications.....	11
1.6.2 Requerimientos para Rich Internet Applications	11
1.6.3 Macromedia Flash Player.....	11
1.6.4 Macromedia Flash MX 2004	12
1.7 Hipótesis de Trabajo.....	12
1.8 Metodología del Trabajo	12
2. Segunda Parte: Desarrollo de la Aplicación.....	13
2.1 Creación de Presentaciones.....	13
2.1.1 Presentaciones	13
2.1.2 Menú Principal	13
2.1.2.a Manejo de errores.....	17
2.1.2.1 Nueva	18
2.1.2.2 Cargar.....	18
2.1.2.2.a Progreso de carga de imágenes de una presentación	19
2.1.2.3 Carga de Filminas de otra presentación.....	20
2.1.2.4 Ayuda.....	21
2.1.2.5 Borrar una presentación	22
2.1.2.5.a Manejo de errores	23
2.1.2.6 Guardar.....	23
2.1.2.7 Animación	26
2.1.2.7.a Transparencia	27
2.1.2.7.b Agrandar hacia abajo	27
2.1.2.7.c En diagonal de arriba hacia abajo de izquierda a derecha.....	28

2.1.2.7.d En línea recta de izquierda a derecha	28
2.1.2.7.e En línea recta de arriba hacia abajo	28
2.1.2.7.f En diagonal de abajo hacia arriba de izquierda a derecha.....	28
2.1.2.8 Fondo.....	28
2.1.2.8.a Paleta de colores.....	30
2.1.2.8.b Manejo de errores	30
2.1.2.9 Borrar Filminas	30
2.1.2.10 Insertar Filminas	31
2.1.2.11 Insertar imagen.....	32
2.1.2.11.a Progreso de carga.....	34
2.1.2.11.b Error de carga	34
2.1.2.11.c Finalización de carga.....	34
2.1.2.12 Insertar texto.....	34
2.1.2.12.a Mover el cuadro de texto.....	35
2.1.2.12.b Establecer las propiedades del texto	36
2.1.2.12.b.1 Paleta de colores	36
2.1.2.12.c Destruir un cuadro de texto	36
2.1.2.12.d Cambiar el tamaño del cuadro de texto	36
2.1.2.12.e Manejo de errores	36
2.1.2.13 Cambiar la posición de una filmina.....	36
2.1.2.14 Impresión de Filminas	37
2.1.2.15 Ir A	40
2.1.2.16 Previsualizar	41
2.1.2.17 Acerca De	42
2.1.3 Comunicación remota	43
2.1.3.a Borrar una presentación	43
2.1.3.b Guardar una presentación.....	43
2.1.3.c Listar las imágenes disponibles para cargar en una filmina.....	44
2.1.3.d Listar las presentaciones disponibles para cargar	44
2.2 Visualización de Presentaciones.....	44
2.2.a Manejo de errores	45
2.2.1 Acerca De.....	45
2.2.2 Ayuda.....	45
2.3 Mensajes de error	46
2.4 Hallazgos	47
2.4.a Carga de imágenes	47
2.4.b Caché del Navegador.....	47
2.5 Pruebas.....	48

3. Tercera Parte	49
3.1 Conclusión	49
3.2 Contribuciones	49
3.3 Líneas Futuras de Desarrollo	49
Glosario.....	50
Bibliografía	51
Apéndice A.....	52
Diagramas de Ideas (brainstormings)	52
A.1 Diagrama 1.....	52
A.2 Diagrama 2.....	54
A.3 Diagrama 3.....	55
A.4 Diagrama 4.....	56
Apéndice B.....	57
Apéndice C	59
C.1 Impresión de texto en modo óptimo para gráficos vectoriales	59
C.2 Impresión de texto en modo óptimo para mapas de bits	59
C.3 Impresión de una imagen en modo óptimo para gráficos vectoriales	60
C.4 Impresión de una imagen en modo óptimo para mapas de bits	60
C.5 Impresión de una imagen y texto en modo óptimo para gráficos vectoriales	60
C.6 Impresión de una imagen y texto en modo óptimo para mapas de bits.....	61

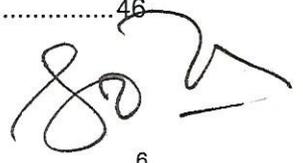


Índice de Figuras

Figura 1: Representación gráfica de una presentación.....	13
Figura 2: Presentación 1.....	20
Figura 3: Presentación 2.....	21
Figura 4: Presentación 2 después de insertar las filminas de Presentación 1.....	21
Figura 5: Posibilidades en fondos para una filmina.....	29
Figura 6: Presentación antes de borrar una filmina.	31
Figura 7: Presentación después de borrar una filmina.....	31
Figura 8: Presentación antes de insertar una filmina.	32
Figura 9: Presentación después de insertar una filmina.	32
Figura 10: Intercambio de datos entre el diálogo de impresión del sistema operativo (e impresora) y la aplicación.	38
Figura 11: Ajuste de la filmina al área imprimible del papel (o no) e impresión en modo vectorial o mapa de bits..	39

Índice de Capturas de Pantalla

Captura de pantalla 1: Menú principal.....	14
Captura de pantalla 2: Menú principal (parte superior).....	15
Captura de pantalla 3: Menú principal (parte media).....	16
Captura de pantalla 4: Menú principal (parte inferior).....	17
Captura de pantalla 5: Carga de presentaciones.....	19
Captura de Pantalla 6: Ayuda de la aplicación.....	22
Captura de pantalla 7: Borrar una presentación.	23
Captura de pantalla 8: Guardar la presentación de ejemplo "exposiciones", con un comentario.	24
Captura de pantalla 9: Animación de la filmina.	27
Captura de pantalla 10: Fondo de la filmina.....	29
Captura de pantalla 11: Inserción de una imagen en una filmina.	33
Captura de pantalla 12: Texto a insertar y sus propiedades.....	35
Captura de pantalla 13: Cambiar la posición de una filmina.	37
Captura de pantalla 14: Impresión de filminas.	38
Captura de pantalla 15: Ir a filmina.....	41
Captura de pantalla 16: Previsualización de una presentación.	42
Captura de pantalla 17: Acerca de.	43
Captura de pantalla 18: Menú principal del proyector de filminas.....	44
Captura de pantalla 19: Ayuda del proyector de filminas.....	46



Captura de pantalla 20: Mensaje de error de ejemplo.47



Handwritten signature in black ink, consisting of stylized cursive letters.

1. Primera Parte

1.1 Introducción

1.1.1 Planteamiento y Contexto del Problema

A la hora de realizar una presentación, convencionalmente se crea un archivo en una herramienta determinada para tal fin (por ejemplo Microsoft Powerpoint) que puede incluir por ejemplo imágenes y texto con formato. El archivo en cuestión una vez creado se almacena en cualesquiera de los medios que se conocen y se traslada al equipo a utilizar, por ejemplo, para su presentación, modificación o ambos. Para estas tareas, este equipo debe contar con una versión del software mencionado para poder recuperarse y presentarse.

Se propone que el usuario podría crear, modificar, exponer y compartir sus presentaciones utilizando una Rich Internet Application.

Con respecto al término Rich Internet Application (RIA): *"the term Rich Internet Application (RIA) is a classification coined by Macromedia [...] defined as those Flash movies that allow the user to accomplish some sort of task or enable a transaction from the Flash movie to an external remote data source."* (el término Rich Internet Application es una clasificación creada por Macromedia, definida como una categoría de aplicaciones realizadas en la herramienta Flash que permiten al usuario hacer una tarea determinada o ejecutar una transacción desde esta aplicación hacia un origen de datos remoto).¹

Las Rich Internet Applications son ejecutadas en un navegador cliente que cuenta con el Macromedia Flash Player instalado.

Según un censo de marzo del 2005 publicado por Macromedia, el 98.3% de las personas que navegan por internet tienen el Macromedia Flash Player instalado y disponible para utilizar con su navegador web.²

En las aplicaciones convencionales para realizar presentaciones el usuario no tiene la posibilidad de compartir su presentación, ya sea con el fin de que otros usuarios puedan acceder a la misma desde diferentes ubicaciones, o con la intención de mantenerla disponible para presentarla sin necesidad de transportarla en un soporte convencional (CD, diskette, etc.)

En el terreno de Rich Internet Applications el usuario no cuenta con una herramienta de autoría de este tipo. También es dependiente de la plataforma en que está trabajando, sea ésta, por ejemplo, Windows, o Macintosh.

¹ Robert REINHARDT; Joey LOTT, *Flash MX 2004 ActionScript Bible*, Estados Unidos, Wiley Publishing, Inc., 2004. Página 5.

² Macromedia. Macromedia Flash Player Statistics, *"Macromedia Flash content reaches 98.2% of Internet viewers"*. Septiembre, 2004.
<http://www.macromedia.com/software/flash/survey/>.



1.1.2 Situación Actual

Se toma como situación actual una lista de sitios publicados en Macromedia Site of The Day³.

Los sitios extraídos están realizados con la herramienta Flash.

Son las Rich Internet Applications que se consideran como estado del arte.

1.1.3 Trabajos Relacionados

Se realiza en el año 2003 para la cátedra Habilitación Profesional 1 un trabajo práctico en grupo utilizando la herramienta Flash MX. El trabajo (denominado "Comunidad Musical") consiste en una aplicación compuesta de un foro, un mixer de música y un reproductor de música. El usuario que desea utilizar esta aplicación se registra y forma parte de la "Comunidad Musical". Siendo miembro puede escribir mensajes en el foro para compartir con otros miembros, combinar loops de música para crear canciones con el mixer y luego utilizar el reproductor de música para escucharlas. También puede compartir canciones para que éstas sean accesibles a otros miembros de la comunidad.

1.2 Objetivo

Desarrollar una herramienta que ofrezca funcionalidad al usuario para crear, modificar y mostrar sus presentaciones así como también una versión en modo proyector de la misma que sólo permita visualizar presentaciones ya creadas.

Estas herramientas serán accesibles utilizando cualquier navegador de internet que posea instalada la versión 7 (o superior) del Macromedia Flash Player.

1.3 Justificación

Las Rich Internet Applications que se conocen (a través del Macromedia Site of the Day) no le ofrecen al usuario la posibilidad de crear una presentación, es decir, crear un archivo que sea un conjunto de *filminas*, para luego agregar texto e imágenes a las mismas. Es decir, el usuario no tiene la posibilidad de aprovechar una conexión a internet y un navegador para mantener su

³ Macromedia. Showcase: Site of the Day Archive, "Best of 2003". Septiembre, 2004.

http://www.macromedia.com/cfusion/showcase/index.cfm?event=sotdarchive&year=2003&loc=en_us

Macromedia. Showcase: Site of the Day Archive, "January, 2004". Septiembre, 2004.

http://www.macromedia.com/cfusion/showcase/index.cfm?event=sotdarchive&year=2004&month=1&loc=en_us

presentación en un sitio remoto, ser autor de la misma, accederla y exponerla utilizando este medio.

1.4 Delimitaciones

1.4.1 Relacionadas con la metodología

Se toma un subconjunto de sitios de internet (ver 1.1.2) como estado del arte.

1.4.2 Relacionadas con la funcionalidad de la aplicación

La aplicación a desarrollar podrá ser utilizada por un único usuario a fines de prueba.

La aplicación a desarrollar ofrece funcionalidades para el desarrollo de presentaciones limitadas. Por limitadas se entiende con capacidad menor a la de una herramienta comercial: el formato de archivos que manejará será único, ya sea para guardar las presentaciones, como en la inserción de contenido externo (tipo de archivos jpg para las imágenes).

El usuario no contará con la opción de cargar sus propias imágenes para agregar a sus *filminas*, ni plantillas predeterminadas, ni la opción de crear gráficos con datos externos, ni incorporar audio, o video.

El usuario tiene que estar conectado a internet para utilizar la aplicación.

Podrá visualizar correctamente su presentación si tiene instaladas las tipografías Arial, Arial Black, Comic Sans MS, Courier New, Georgia, Impact, Times New Roman, Trebuchet MS y Verdana.

1.5 Organización del documento

En la siguiente parte del documento se describirá el desarrollo de la aplicación.

En la última parte se expondrán los hallazgos y resultados de las pruebas y conclusiones.

En el apéndice A se encuentran los diagramas de ideas.

En el apéndice B, un diagrama resultante de la metodología empleada.

En el apéndice C, pruebas de impresión utilizando la aplicación.

Las palabras escritas en cursiva se encuentran explicadas en el glosario.

1.6 Conceptos Fundamentales

Las Rich Internet Applications conocidas hasta el momento, permiten a los usuarios compartir recursos entre ellos, como por ejemplo audio, video, imágenes, etc. como único propósito, indistintamente del uso que pueda darle el usuario a esta posibilidad.



Estas aplicaciones no poseen un carácter de "herramienta de autoría", que a diferencia de la aplicación que aquí se propone, permite a los usuarios desarrollar contenido y compartirlo con un fin concreto (almacenamiento remoto, o publicación para un público determinado).

Por el contrario, la herramienta de visualización de presentaciones aquí propuesta (como un módulo separado de la herramienta de creación) posee las características de las Rich Internet Applications convencionales, donde el usuario sólo accede al contenido de manera pasiva, sin la posibilidad de poder modificarlo.

Los conceptos fundamentales son tomados de (X) en la bibliografía (Rich Internet Applications, Requerimientos para Rich Internet Applications y Macromedia Flash Player únicamente).

1.6.1 Rich Internet Applications

Las aplicaciones denominadas Rich Internet Applications se ejecutan del lado del cliente para ofrecer posibilidades de interactividad al navegar en internet.

Combinan la experiencia del usuario de aplicaciones de escritorio con aplicaciones web.

La tecnología de "rich clients" hace posible que se construyan estas aplicaciones.

Del lado del "rich client" están los archivos que se acceden vía protocolos estándar para navegar en internet (en este caso de extensión swf, a utilizar con el Macromedia Flash Player).

Las Rich Internet Applications dependen de páginas web y no las reemplazan, sino que las extienden, en términos de funcionalidad disponible para el usuario.

1.6.2 Requerimientos para Rich Internet Applications

Requiere tres elementos que se explican a continuación.

- Tecnología de "rich client": esta tecnología está del lado del cliente. Relacionada con la capacidad de procesamiento de la computadora del mismo.
- Tecnología del servidor: mecanismo para conectar los "rich clients", a la lógica y los datos.
- Herramientas de desarrollo: para crear Rich Internet Applications.

La herramienta Flash MX 2004 es una herramienta de desarrollo de Rich Internet Applications.

La tecnología que posibilita la ejecución de la aplicación desarrollada en el "rich client" es el Macromedia Flash Player.

La tecnología de servidor utilizada es un servidor web estándar que ejecute ASP o PHP.

1.6.3 Macromedia Flash Player

Tecnología de "rich client". Se instala en la PC del usuario como un plug-in para el navegador de internet, y es el encargado de la ejecución de los archivos SWF generados con Macromedia Flash.

Este control es de distribución gratuita y masiva.

1.6.4 Macromedia Flash MX 2004

Es una herramienta de autor que permite desarrollar contenido enriquecido para la web: diseñar y animar sitios web con alto contenido multimedia (video, imágenes, audio, etc.) y además posee un lenguaje de programación propio denominado Actionscript. Este lenguaje permite desarrollar aplicaciones interactivas e integrarlas en la web.

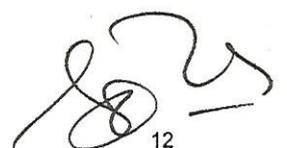
1.7 Hipótesis de Trabajo

Es posible, con la herramienta Flash MX 2004, desarrollar una aplicación de creación y visualización de presentaciones accesible desde un navegador de internet.

1.8 Metodología del Trabajo

La metodología a seguir consiste en:

- Descomponer el conjunto de ideas usando diagramas de ideas (brainstormings) para identificar aquellas que se van a considerar, y aquellas que se van a descartar. Con esto se analiza el problema. Ver Apéndice A.
- Definir la funcionalidad de la herramienta en base a los diagramas de ideas.
- Crear los componentes de acuerdo a la funcionalidad para integrar en forma bottom-up, al mismo tiempo que se describe cada uno, los hallazgos durante la práctica y características de cada componente. Ver Apéndice B y Segunda Parte.
- Integrar los componentes.
- Revisar la funcionalidad.
- Testear la aplicación online en distintos navegadores.
- Elaborar el documento para presentar.
- Elaborar la presentación.



2. Segunda Parte: Desarrollo de la Aplicación

2.1 Creación de Presentaciones

2.1.1 Presentaciones

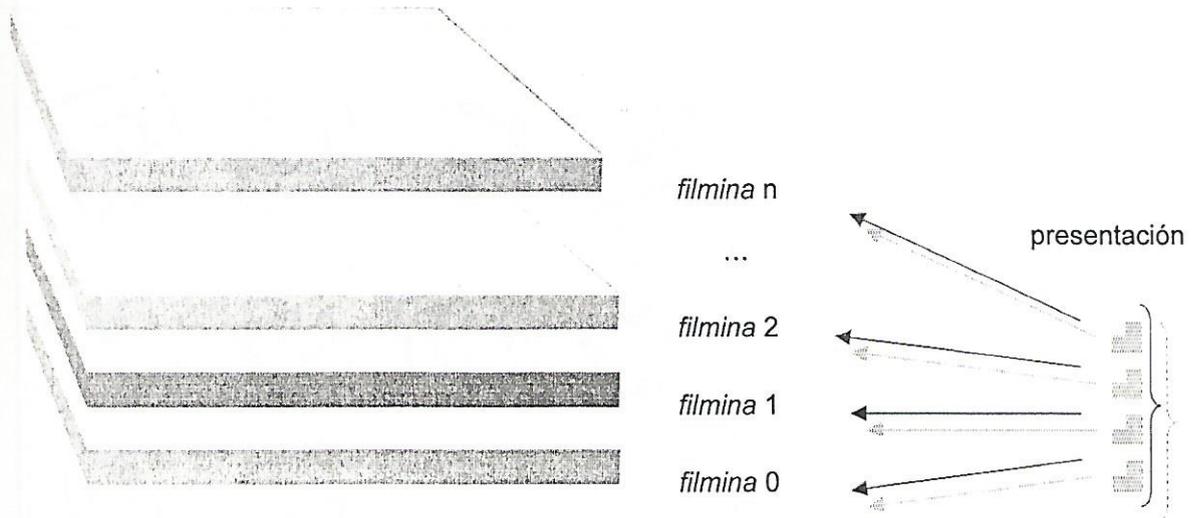


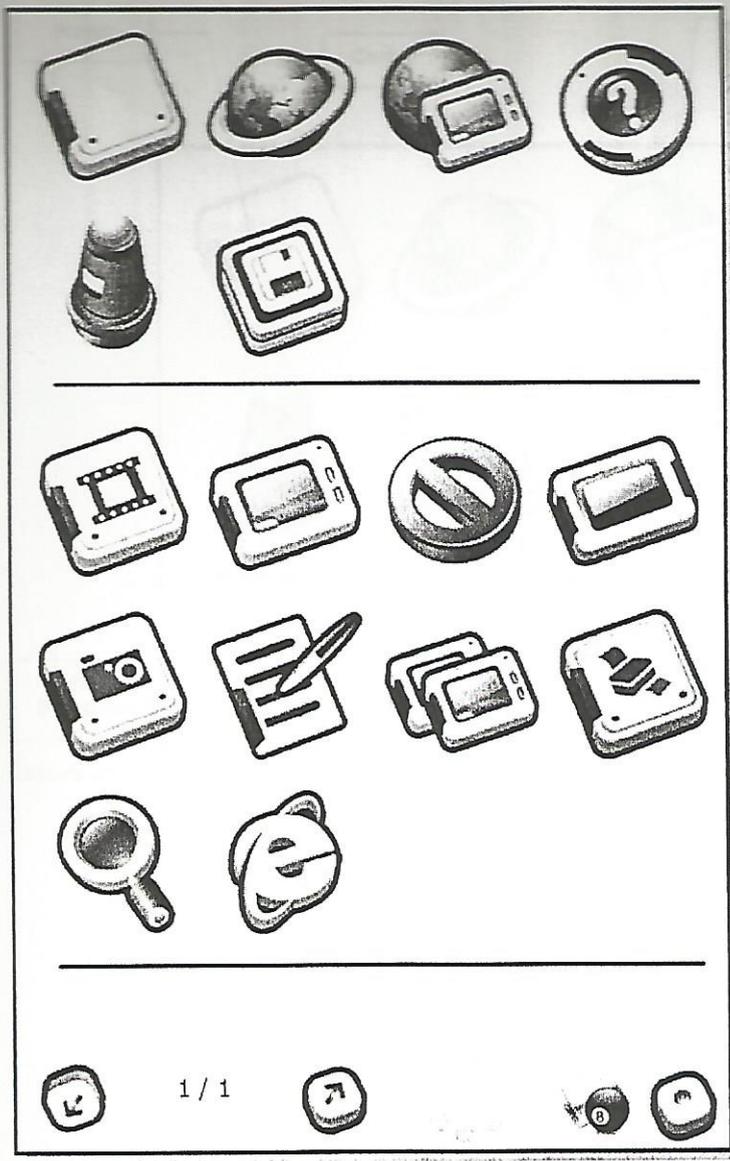
Figura 1: Representación gráfica de una presentación

La presentación está representada en la aplicación por un arreglo ordenado (según la creación, con orden modificable a cargo del usuario) con referencias a los *movieClips* que la componen, y un índice, que es un número, en el rango $[0; longitud \text{ de la presentación}]$ o un valor nulo, en el caso que la misma esté vacía, que representa la *filmina* activa que se está mostrando.

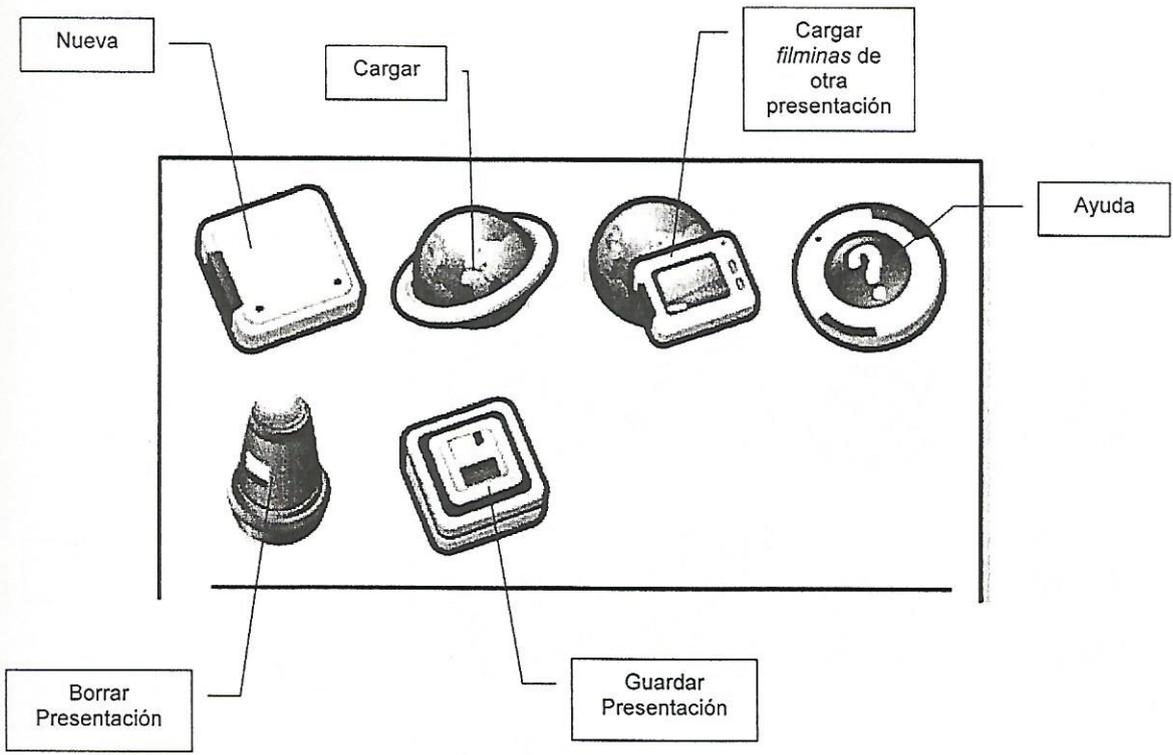
En la Figura 1, el arreglo está representado por los cuatro cuadrados encerrados en una llave.

2.1.2 Menú Principal

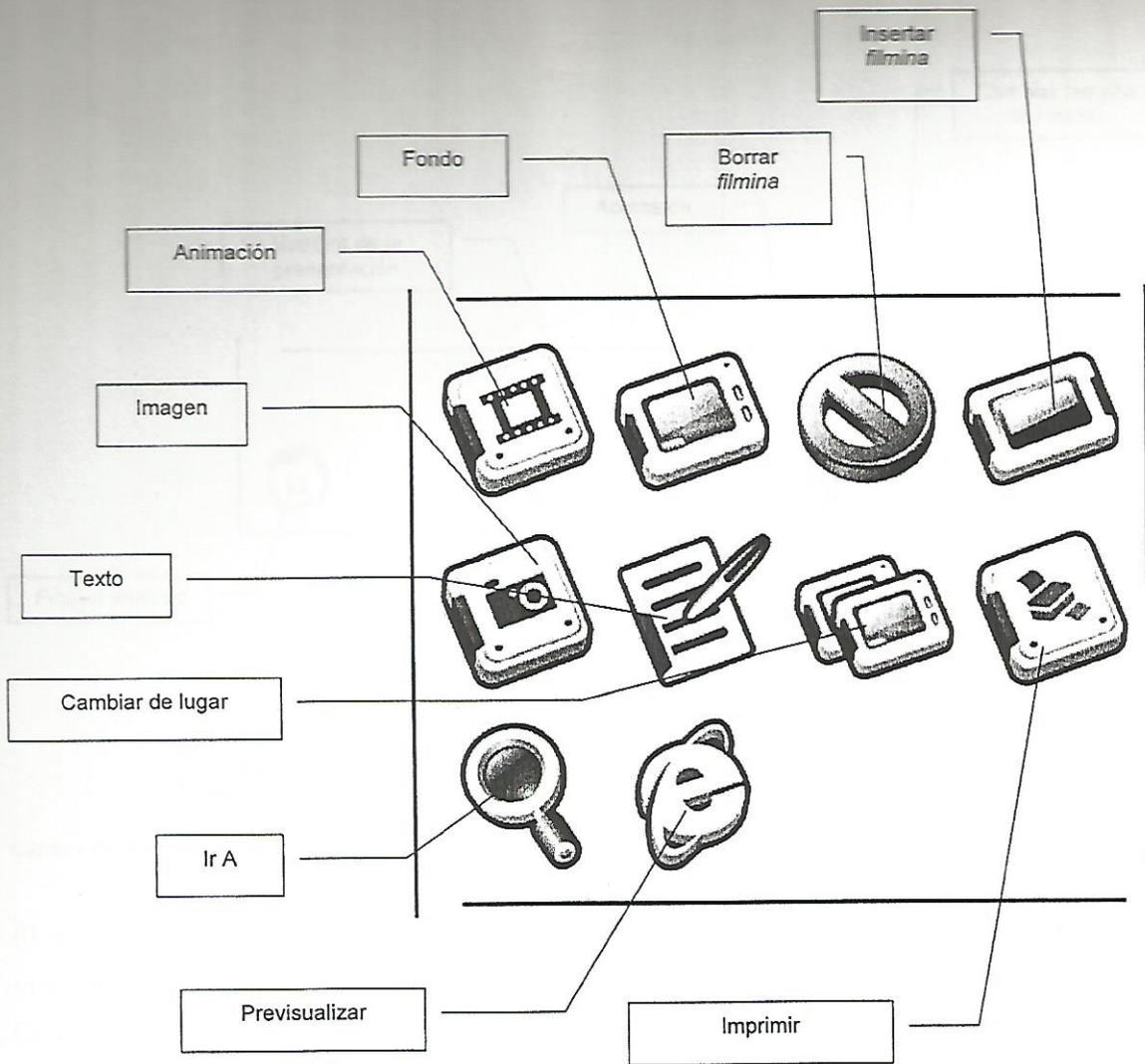
A continuación se muestran capturas de pantallas del menú principal, de creación de presentaciones.



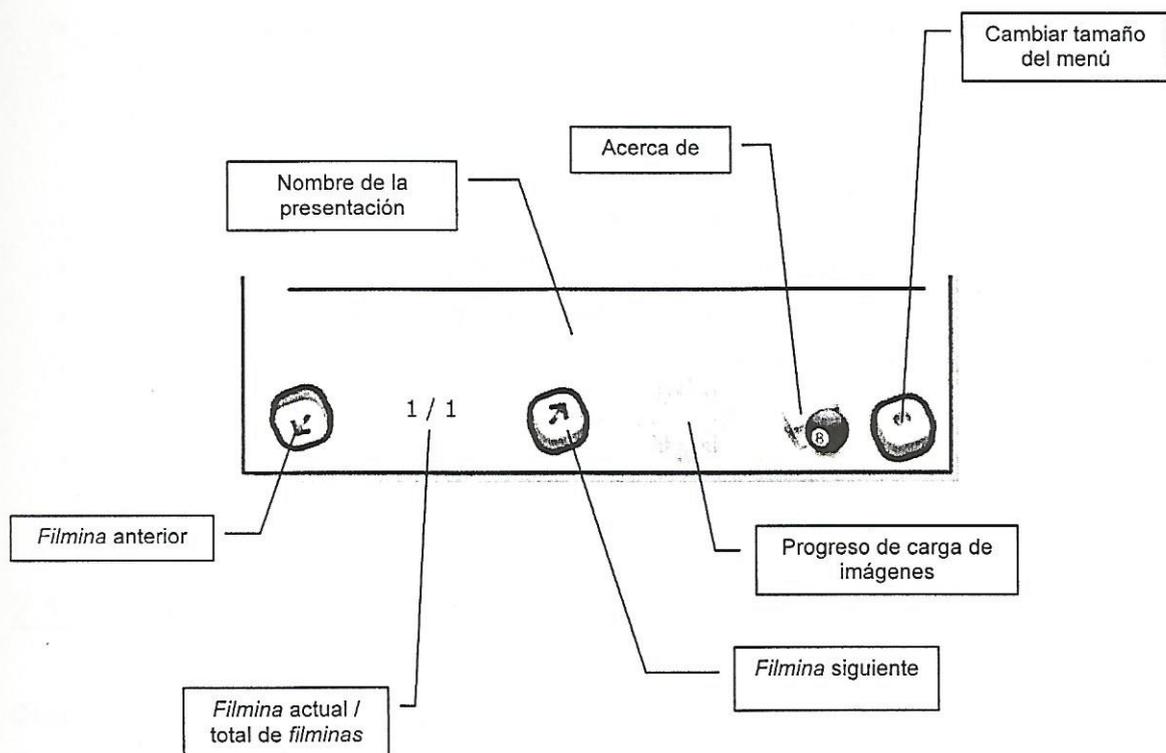
Captura de pantalla 1: Menú principal



Captura de pantalla 2: Menú principal (parte superior)



Captura de pantalla 3: Menú principal (parte media)



Captura de pantalla 4: Menú principal (parte inferior)

El menú principal ofrece acceso a todas las funciones disponibles, incluyendo aquellas para navegar una presentación existente.

Consiste en un *movieClip* con *movieClips anidados*, cada uno con una imagen asociada dependiendo de cada opción.

Haciendo click en cada sub *movieClip* se invoca a una función que carga el *movieClip* correspondiente a cada opción.

También contiene un *movieClip* que permite arrastrarlo.

Es posible cambiar el tamaño del mismo a 50% de su tamaño real, su tamaño real, o 150% su tamaño real.

Para mostrar u ocultar el menú principal, se utiliza la Barra Espaciadora.

En las capturas de pantalla, los botones tienen el siguiente significado (con la aplicación ejecutándose hay ayuda al mover el Mouse sobre los mismos):



2.1.2.a Manejo de errores

Se muestran errores en los siguientes casos:



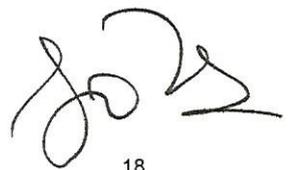
- Se quiere insertar un cuadro de texto y no hay una *filmina* en la presentación
- Se quiere insertar una imagen y no hay una *filmina* en la presentación
- Se quiere borrar una *filmina* y no hay una *filmina* en la presentación
- Se quieren establecer las propiedades de una *filmina* y no hay una *filmina* en la presentación
- Se quiere imprimir una presentación y ésta está vacía
- Se quiere previsualizar una presentación y ésta no está guardada
- Se quiere borrar una presentación y ésta no está guardada
- Se quiere establecer la animación para una *filmina* y no hay una *filmina* en la presentación
- Se quiere ir a una *filmina* y no hay una *filmina* en la presentación

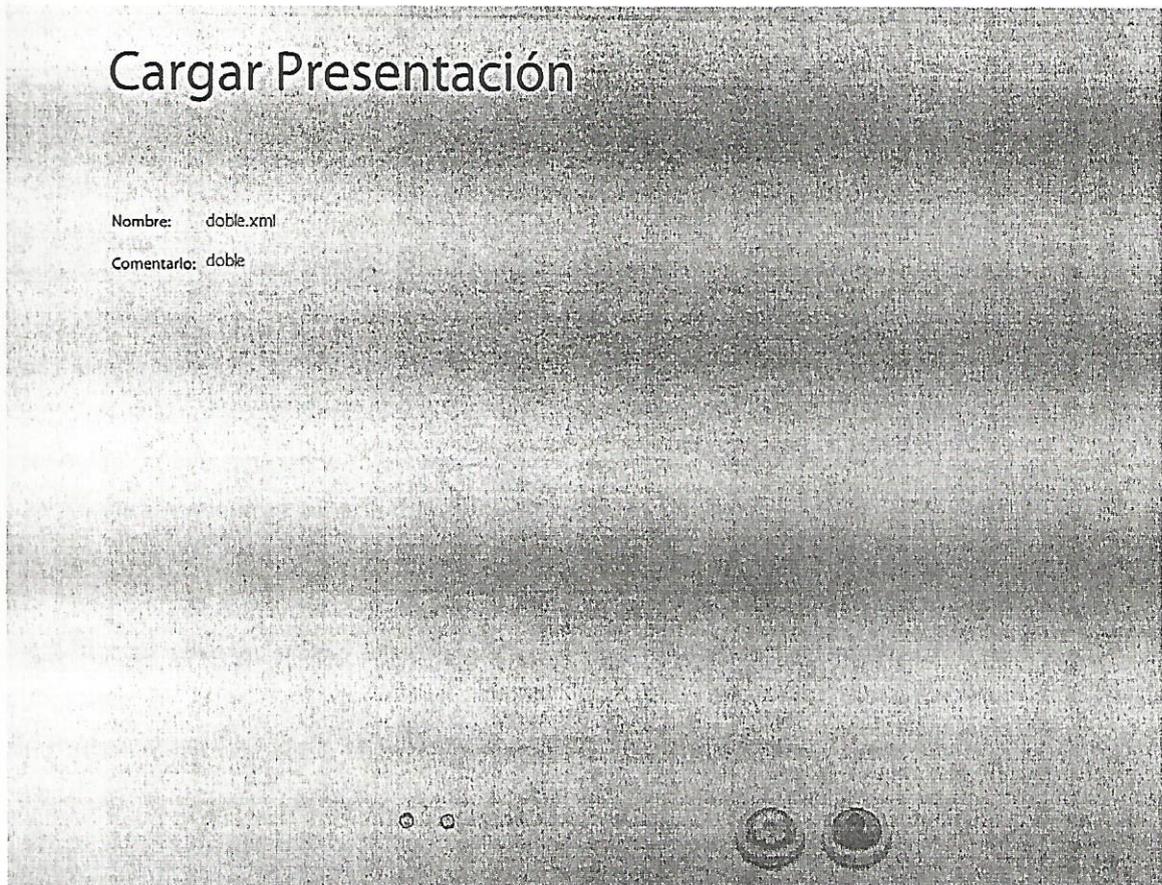
2.1.2.1 Nueva

Crea una presentación nueva y vacía.

Si hay una presentación cargada (esté ésta guardada o no) se desaloja de la memoria, sin pedir confirmación al usuario.

2.1.2.2 Cargar





Captura de pantalla 5: Carga de presentaciones.

Cargar una presentación significa tomar una presentación previamente guardada con esta aplicación y reconstruirla.

Para ese fin se utiliza una función que se invoca para realizar todas las acciones necesarias. Ésta toma un objeto XML, recorre sus nodos, e invoca otras funciones para reconstruir la presentación guardada.

2.1.2.2.a Progreso de carga de imágenes de una presentación

La carga de imágenes de una presentación es monitoreada con un objeto que recibe notificaciones frente a diferentes eventos.

Estos eventos son: comienzo de carga, progreso de carga, error de carga y finalización de carga.

El objeto tiene asociada una función que muestra un indicador en el menú principal que es un *movieClip*.

La función es la misma en los diferentes eventos.

Esta función hace que se muestre la animación, en cada uno de los diferentes eventos.

El *movieClip* tiene asociadas las acciones para mostrar la animación.

Mientras que no se hayan finalizado de cargar todas las imágenes, se muestra la animación del *movieClip*, es decir, se ejecutan esas acciones.

Cuando se terminaron de cargar todas las imágenes de la *filmina*, la animación cesa de ejecutarse automáticamente.

El objeto que monitorea la carga de imágenes se registra con los objetos que se utilizan para la carga de imágenes de la clase *MovieClipLoader* de Flash MX 2004.

2.1.2.3 Carga de Filminas de otra presentación

Para cargar *filminas* de otra presentación se invoca al *movieClip* para carga de presentaciones. Como se están agregando *filminas* a la presentación existente, no se descarga previamente la presentación que está en la memoria.

Las *filminas* se insertan a partir de la *filmina* activa en adelante.

Ejemplo: cargar las *filminas* de Presentación 1 en la posición 1 de Presentación 2.

En Presentación 1 (Figura 2) se puede ver que la *filmina* 1 y *filmina* 2 fueron cambiadas de lugar por el usuario. Esto se afirma porque los números de *filmina* no coinciden con los subíndices del arreglo con las referencias.

En Presentación 2, (Figura 3) la *filmina* 0, 1 y 2 fueron cambiadas de lugar.

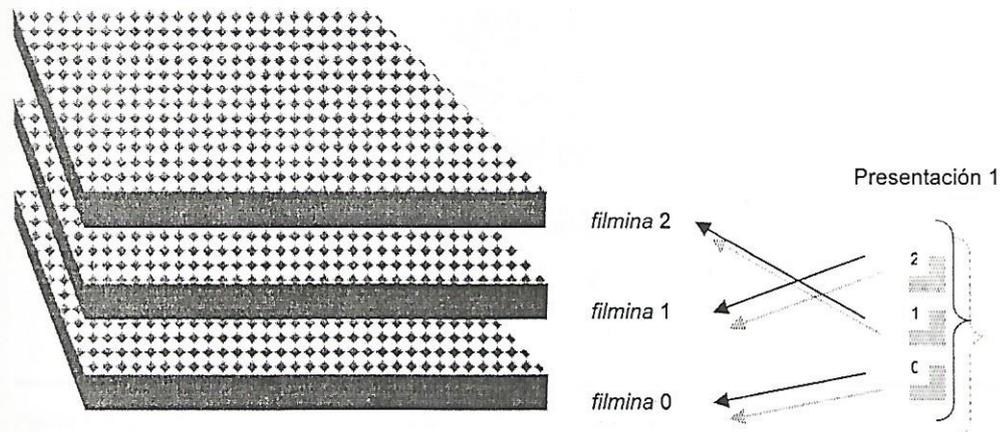


Figura 2: Presentación 1.

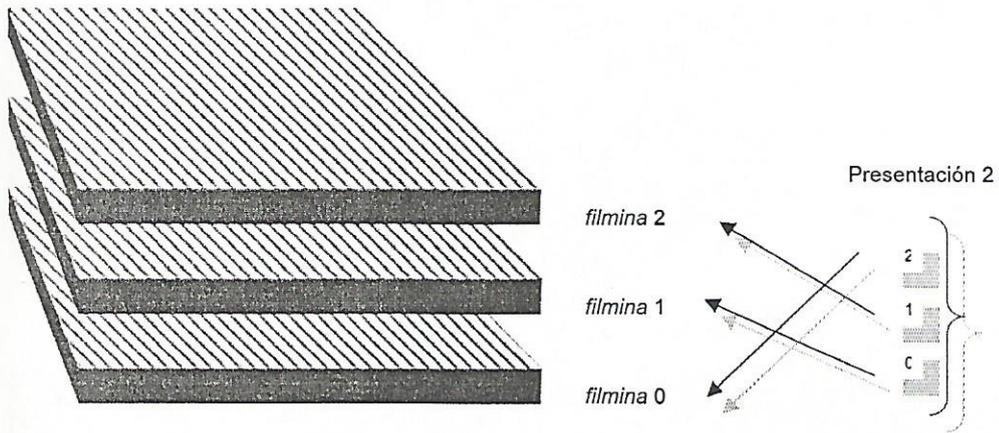


Figura 3: Presentación 2.

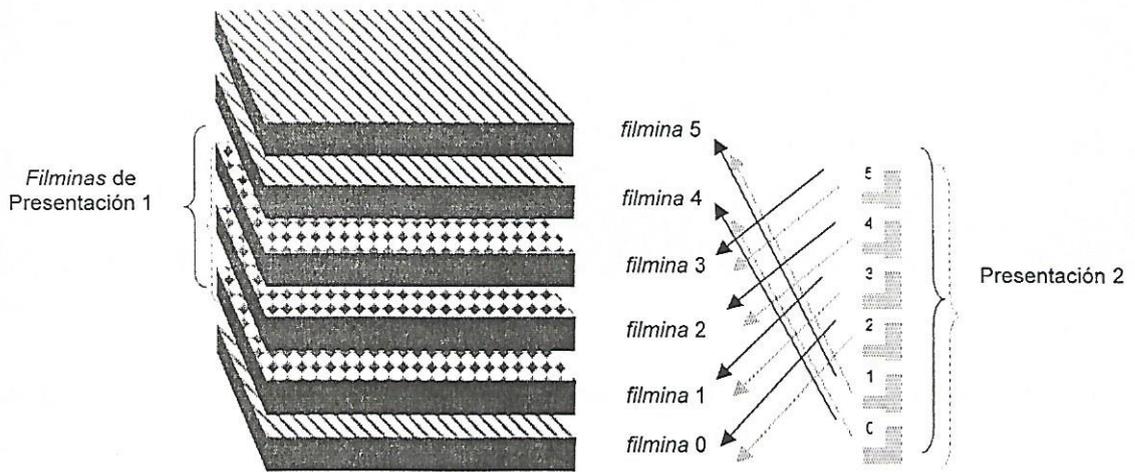


Figura 4: Presentación 2 después de insertar las filminas de Presentación 1.

2.1.2.4 Ayuda

Ayuda

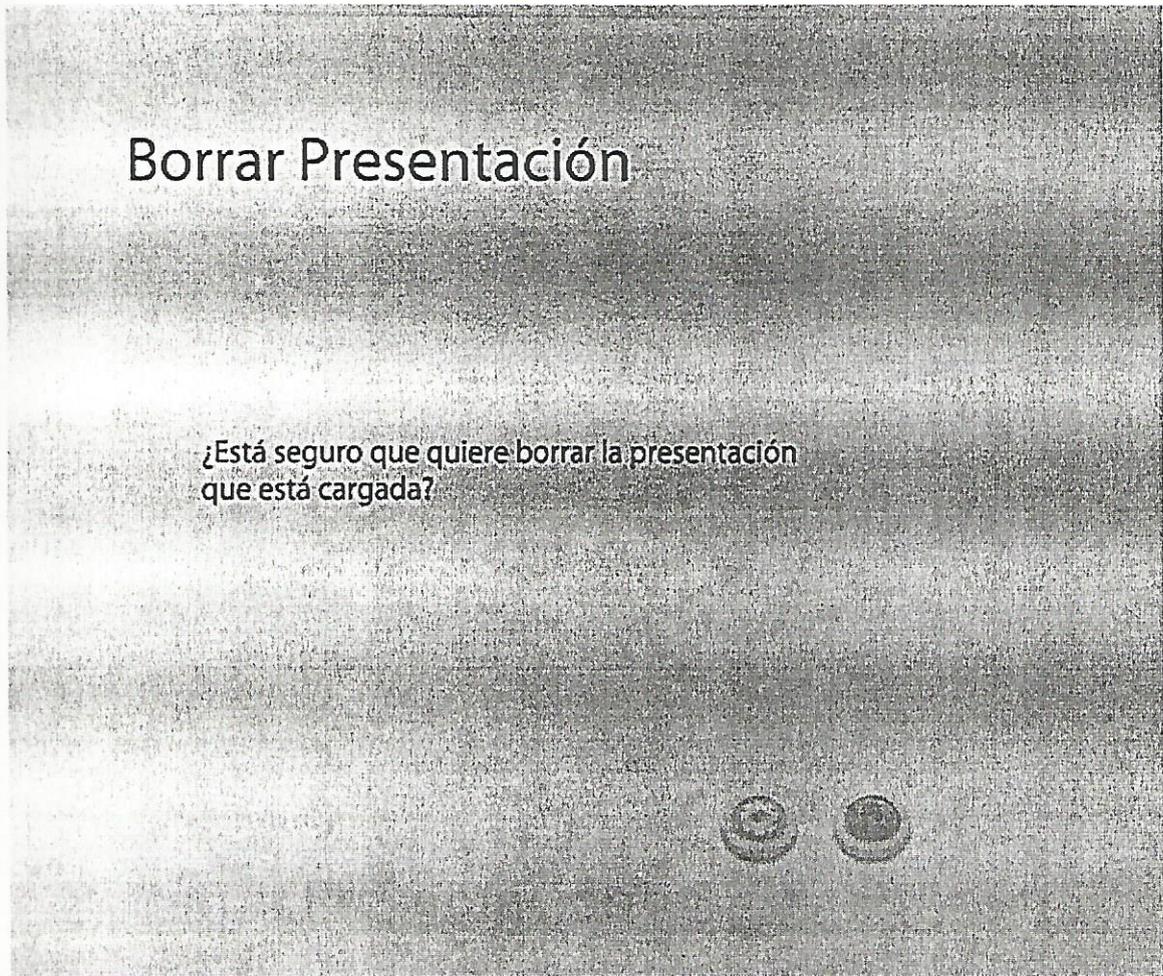
vacía la presentación que haya creada o cargada.
muestra el diálogo de apertura de presentaciones existentes.
permite insertar todas las filminas de una presentación existente.
Se permite insertar filminas de la versión guardada de la presentación que se ha cargado.
elimina la presentación actual.
almacena la presentación existente y muestra el vínculo para previsualización.
No se permite almacenar una presentación vacía.
muestra el diálogo para establecer la animación para la filmina actual. Esta animación se utiliza en modo proyector.
muestra el diálogo para establecer el fondo de la filmina actual. Éste puede ser una imagen o un color sólido. Si se tiene una imagen cargada como fondo y se elige un relleno sólido se reemplaza automáticamente.
borra la filmina actual.
crea una filmina nueva en la posición actual.
muestra el diálogo para carga de imágenes en la filmina actual.
Inserta un cuadro de texto con un texto por defecto. Para cambiar el texto y sus propiedades, presione el botón anaranjado que aparece moviendo el mouse sobre el texto. Para desplazar el cuadro de texto, presione el botón amarillo que aparece moviendo el mouse sobre el texto.
mueve una filmina a otra posición en la presentación.
muestra el diálogo para imprimir toda la presentación o un intervalo.
estira el menú automáticamente a 3 diferentes tamaños.
vá a una filmina de la presentación.

Captura de Pantalla 6: Ayuda de la aplicación.

La ayuda muestra breves descripciones con respecto a cada una de las opciones disponibles en el menú principal.

2.1.2.5 Borrar una presentación





Captura de pantalla 7: Borrar una presentación.

Se puede acceder a esta opción desde una presentación cargada.

La presentación cargada se toma como activa y es la que va a ser borrada.

La misma tiene que haber sido guardada previamente.

Luego de borrada, se limpian las *filminas* correspondientes a la misma de la memoria.

El usuario tiene ahora la posibilidad de abrir otra presentación o crear una nueva.

2.1.2.5.a Manejo de errores

La presentación a borrar tiene que haber sido guardada previamente.

Si esto no se cumple se muestra un mensaje de error.

Si no se pudo ejecutar el comando para borrar la presentación, se muestra un mensaje de error.

2.1.2.6 Guardar

Guardar Presentación

Nombre:

Comentario:

Captura de pantalla 8: Guardar la presentación de ejemplo "exposiciones", con un comentario.

Guardar una presentación significa convertir la misma en su equivalente en XML, tomando ciertos datos que luego se consideran para reconstruirla.

Esto significa tomar el contenido de la misma (residente en la memoria) y generar un correspondiente documento XML para su posterior recuperación.

Los datos relevantes para el armado del árbol son:

- El nombre de la presentación
- Un comentario acerca de la presentación, que ingresa el usuario en el momento de guardarla
- La cantidad total de *filminas* de la presentación
- El total de objetos por *filminas* de la presentación
- La posición de los cuadros de texto dentro de cada *filmina*
- El tamaño de los cuadros de texto dentro de cada *filmina*
- El formato del texto adentro de cada uno de ellos, incluyendo tipografía, color y alineación, y el texto en sí
- Datos sobre si cada *filmina* de la presentación tiene una imagen de fondo, o un relleno de fondo
- La imagen de fondo o color de fondo, según corresponda



- El nombre y posición de cada imagen en cada *filmina*
- La animación elegida para cada una de las *filminas*

El documento XML que representa una presentación, tendría este aspecto:

```
<presentacion nombrePresentacion="nombre" comentario="comentario" cantSlides="x">
  <slide0 imagenFondo = "imagen" cantObjetos="y" animacion="animacion">
    <texto0 ... >
      Este es el texto correspondiente
    </texto0>
    <imagen0 ...>
      <nombre de la imagen>
    </imagen0>
  </slide0>
  <slide1 colorFondo="color" ...>
    ...
  </slide1>
  <slide2 ...>
    ...
  </slide2>
  ...
</presentacion>
```

Todos los datos salvo el texto ingresado por el usuario en los cuadros de texto y las imágenes son guardados como atributos en cada uno de los nodos.

A continuación se muestra un ejemplo de una presentación, con dos *filminas*. En la primera hay un cuadro de texto con el texto "Título" en el mismo, con tipografía "Georgia" en color blanco. En la segunda hay un cuadro con el texto "Contenido de la Filmina 1", con tipografía "Georgia" en color negro.

Ambas *filminas* tienen un color de fondo blanco.

```
<presentacion nombrePresentacion="pruebaDeTexto" comentario="prueba de texto"
cantSlides="2">
  <slide0 colorFondo="16777215" animacion="0" cantObjetos="1">
    <texto0 bullet="false" align="left" color="16777215" size="16" underline="false"
italic="false" bold="false" font="Georgia" _width="118" _height="52" _y="146"
_x="159"> T%EDtulo </texto0>
  </slide0>
  <slide1 colorFondo="16777215" animacion="0" cantObjetos="1">
    <texto0 bullet="false" align="left" color="0" size="10" underline="false" italic="false"
bold="false" font="Georgia" _width="167" _height="37" _y="100" _x="100">
Contenido%20de%20la%20Filmina%201 </texto0>
```

</slide1>

</presentacion>

Notar los datos respectivos al formato del texto usados como atributos: bullet, align, color, size, underline, italic, bold y font. `_x` y `_y` son la posición del cuadro de texto en el *Stage*. `_width` y `_height` es el tamaño del cuadro de texto.

Al texto de los cuadros de texto se le aplica la función `escape` y `unescape` para contemplar el caso en que se ingresen caracteres especiales.

Observar también dos datos importantes, que son la cantidad total de *filmintas* y la cantidad total de objetos en una *filminta*, que sirven como condición de parada, cuando se toma un documento XML generado por la aplicación y se obtiene la presentación que éste representa.

A éstos se los ubica como atributos en el nodo raíz y en el nodo principal de cada *filminta*.

El valor de color que se almacena es el equivalente decimal de ese color. En este caso 16777215_{10} es $FFFFFF_{16}$ (blanco) y 0_{10} es 0_{16} (negro).

Una presentación con tres *filmintas* sin texto ingresado tendría el siguiente aspecto:

```
<presentacion nombrePresentacion="tres" comentario="tres" cantSlides="3">
```

```
  <slide0 colorFondo="16777215" animacion="0" cantObjetos="0"/>
```

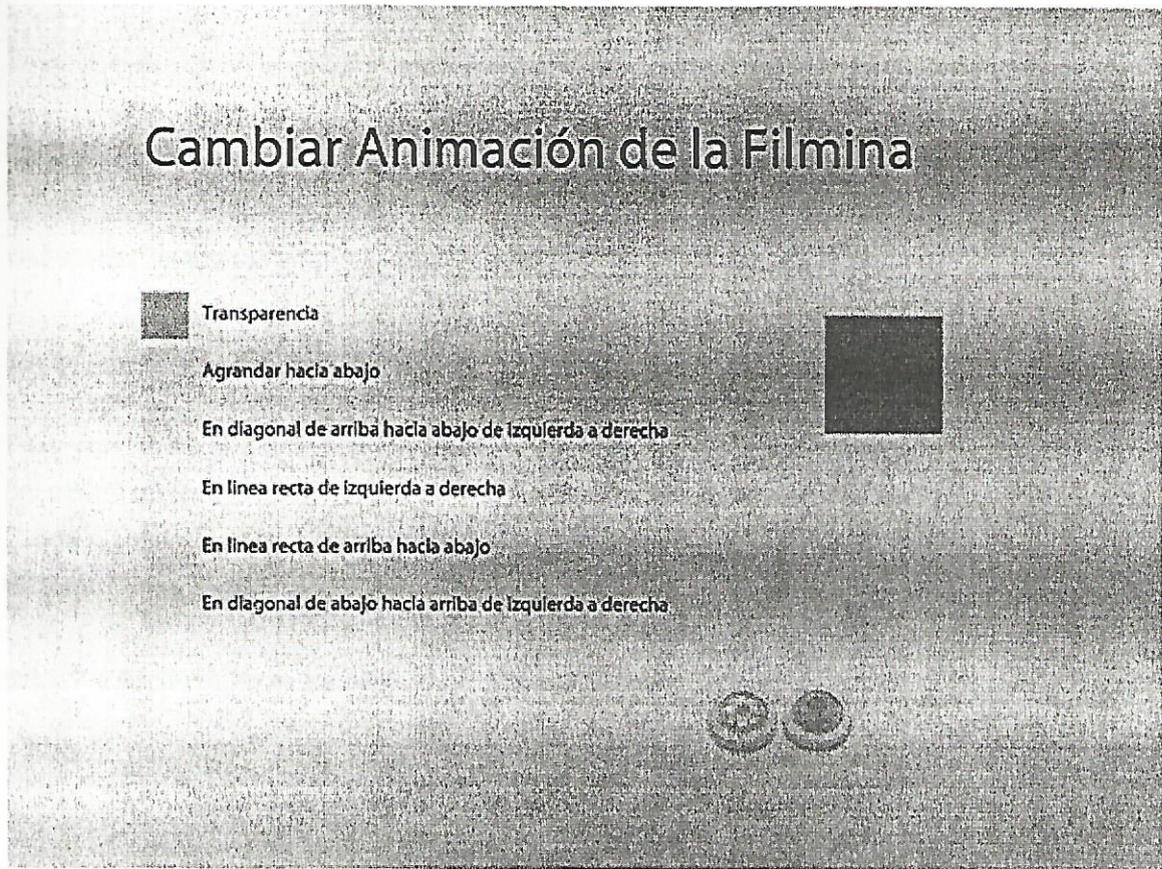
```
  <slide1 colorFondo="16777215" animacion="0" cantObjetos="0"/>
```

```
  <slide2 colorFondo="16777215" animacion="0" cantObjetos="0"/>
```

```
</presentacion>
```

2.1.2.7 Animación





Captura de pantalla 9: Animación de la filmina.

La animación elegida para cuando se muestra la *filmina* que se está editando se guarda en una variable.

Cuando se utiliza el proyector (ver 2.2), se consulta el valor de esta variable y se realiza la animación elegida por el usuario.

En todos los casos la animación se ejecuta en un intervalo igual a 1 segundo.

La misma se lleva a cabo con el método *tween*.

El usuario tiene seis animaciones disponibles para elegir.

2.1.2.7.a Transparencia

Para comenzar la animación, la propiedad *_alpha* de la *filmina* se establece en 0.

Luego, usando *tween*, se disminuye su transparencia hasta llegar a tomar el valor 100.

2.1.2.7.b Agrandar hacia abajo

Para comenzar la animación, el tamaño en x e y de la *filmina* se establece 0.

Luego, usando *tween*, se aumenta el valor en x e y hasta llegar a tomar el valor 100.

2.1.2.7.c En diagonal de arriba hacia abajo de izquierda a derecha

Para comenzar la animación, la posición en x de la *filmina* se disminuye y la posición en y de la misma se aumenta, de modo que se encuentre no visible en el *Stage*.

Luego, usando *tween*, se aumenta la posición en x y se disminuye la posición en y para que sea visible, quedando centrada en el *Stage*.

2.1.2.7.d En línea recta de izquierda a derecha

Para comenzar la animación, la posición en x de la *filmina* se disminuye, de modo que se encuentre no visible en el *Stage*.

Luego, usando *tween*, se aumenta la posición en x para que sea visible, quedando centrada en el *Stage*.

2.1.2.7.e En línea recta de arriba hacia abajo

Para comenzar la animación, la posición en y de la *filmina* se aumenta, de modo que se encuentre no visible en el *Stage*.

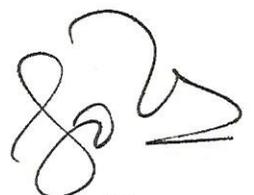
Luego, usando *tween*, se disminuye la posición en y para que sea visible, quedando centrada en el *Stage*.

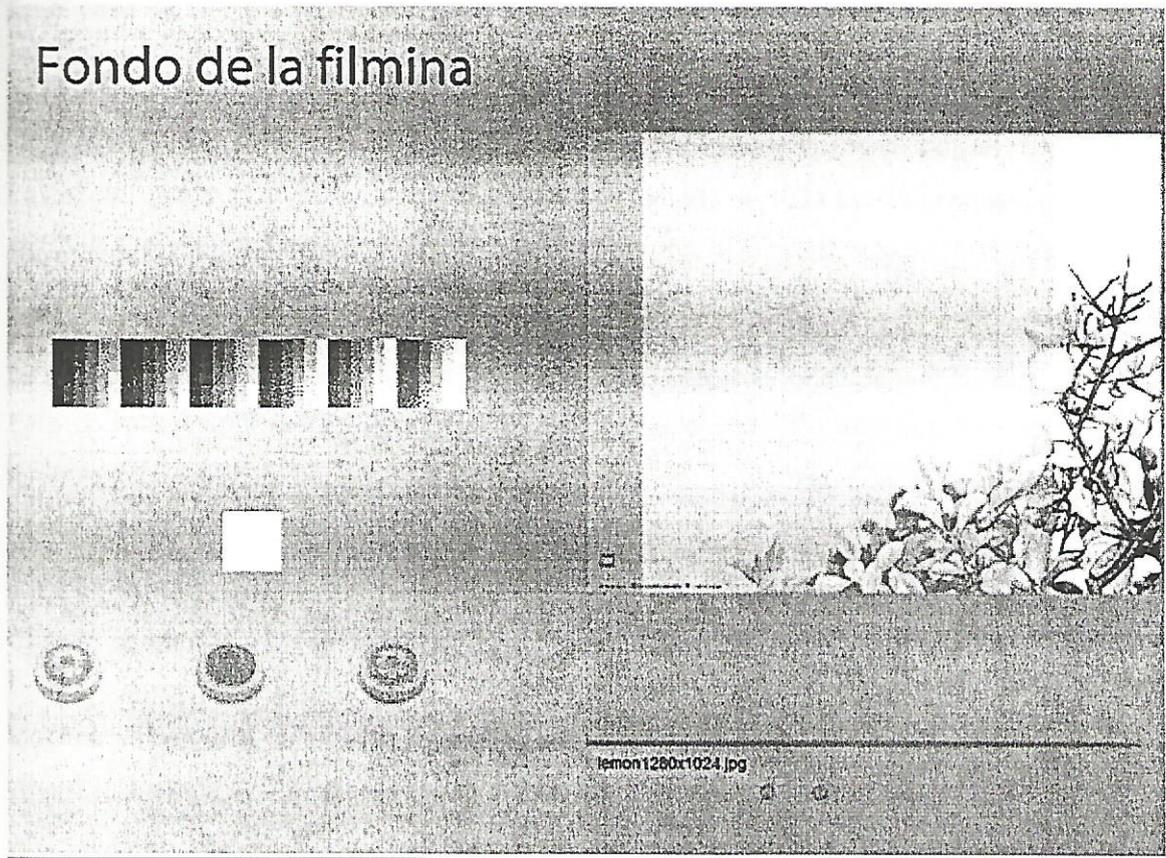
2.1.2.7.f En diagonal de abajo hacia arriba de izquierda a derecha

Para comenzar la animación, la posición en x e y de la *filmina* se disminuyen, de modo que se encuentre no visible en el *Stage*.

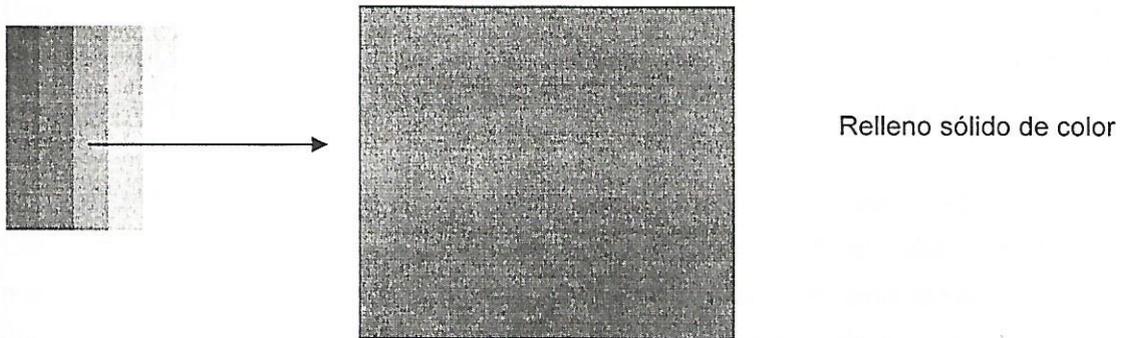
Luego, usando *tween*, se aumentan las posiciones en x e y, para que sea visible, quedando centrada en el *Stage*.

2.1.2.8 Fondo

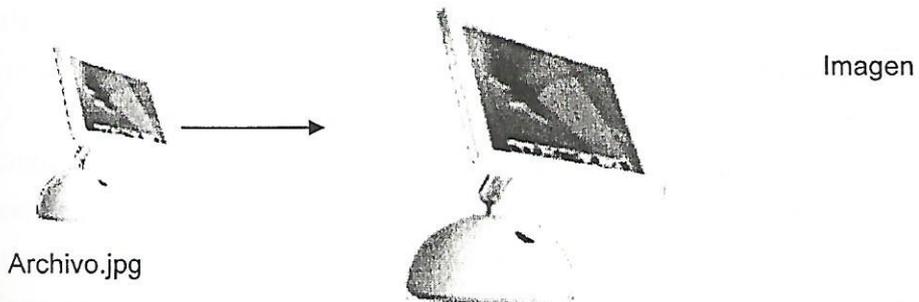




Captura de pantalla 10: Fondo de la filmina



Relleno sólido de color



Archivo.jpg

Imagen

Figura 5: Posibilidades en fondos para una filmina.

El fondo de la *filmina* puede ser un relleno sólido de color o bien una imagen.

En el caso de elegirse un fondo relleno de color sólido, se utilizan métodos de Flash MX 2004 para dibujar el rectángulo de fondo.

En el caso de elegirse una imagen, se utiliza un *movieClip* vacío dependiente de la *filmina* para cargarla.

La imagen se ajusta en forma no proporcional a un tamaño de 1024 por 768 pixeles.

2.1.2.8.a Paleta de colores

La paleta de colores se utiliza para establecer el color sólido de fondo de una *filmina*.

Está compuesta por 6 subpaletas con 36 colores cada una. En total son 216 colores. Se denomina Web Safe.

2.1.2.8.b Manejo de errores

Cuando se abre una presentación y se está tratando de cargar una imagen de fondo para una *filmina* determinada, y ésta no se encuentra, o se comienza a cargar pero no es posible terminar de cargarla, se muestra un mensaje de error.

2.1.2.9 Borrar Filminas

Dependiendo la posición en la presentación, el valor del índice, se utilizan tres métodos de manipulación de arreglos de Flash MX 2004, de acuerdo a si el mismo se encuentra al final, medio o principio de la misma, respectivamente. Estos son splice, shift y pop. Se explican a continuación.

Con una presentación vacía (el índice con valor nulo) no se realiza ninguna acción.

Con una presentación no vacía, con el índice al final del arreglo, se quita un elemento del mismo, utilizando el método pop. Este método retorna el último elemento del arreglo y lo quita del mismo (en este caso la referencia al último *movieClip*, *filmina*, de la presentación). Esta referencia se almacena en una variable temporaria.

Con una presentación no vacía, con el índice en una posición diferente en el arreglo de *filminas* {0 , longitud de la presentación-1}, se utiliza el método splice de Flash MX 2004 (el cual también se usa para insertar elementos) de la siguiente forma: splice (índice, 1). Esto quiere decir que a partir del índice, se borra 1 elemento. Previo a esto se almacena el elemento arreglo[índice] en una variable temporaria para luego quitar el *movieClip* de memoria.

Con una presentación no vacía, con el índice al principio del arreglo, se utiliza el método shift. Este método retorna el primer elemento del arreglo y lo quita del mismo (en este caso la referencia al primer *movieClip*, *filmina*, de la presentación). Esta referencia se almacena en una variable temporaria.



Luego se quitan de memoria los *movieClips* (almacenados en las variables temporarias en cada caso).

En todas las situaciones, después del borrado, se verifica que el arreglo que representa la presentación no esté vacío después de aplicar cada uno de los correspondientes métodos. Si queda vacío, el índice toma el valor nulo.

Ejemplo: borrar una *filmina* en la posición 2 de una presentación.

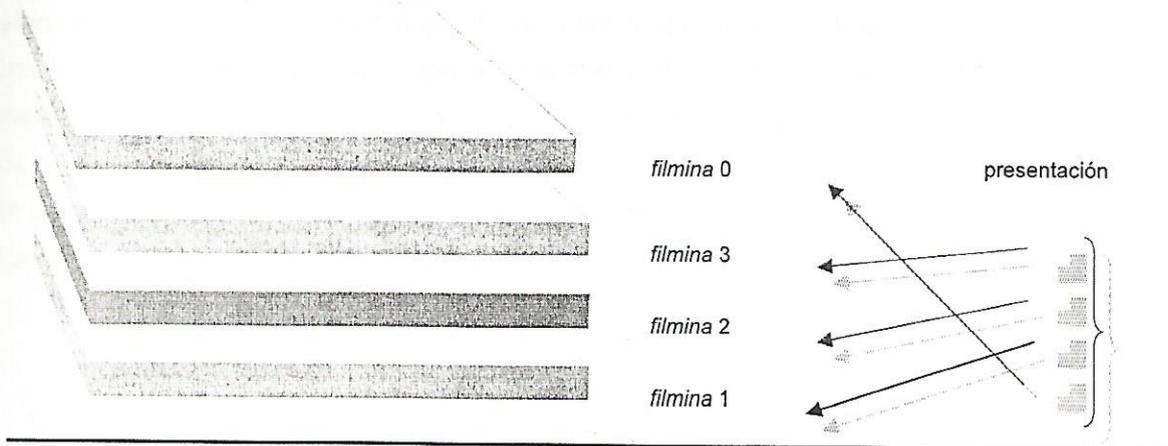


Figura 6: Presentación antes de borrar una filmina.

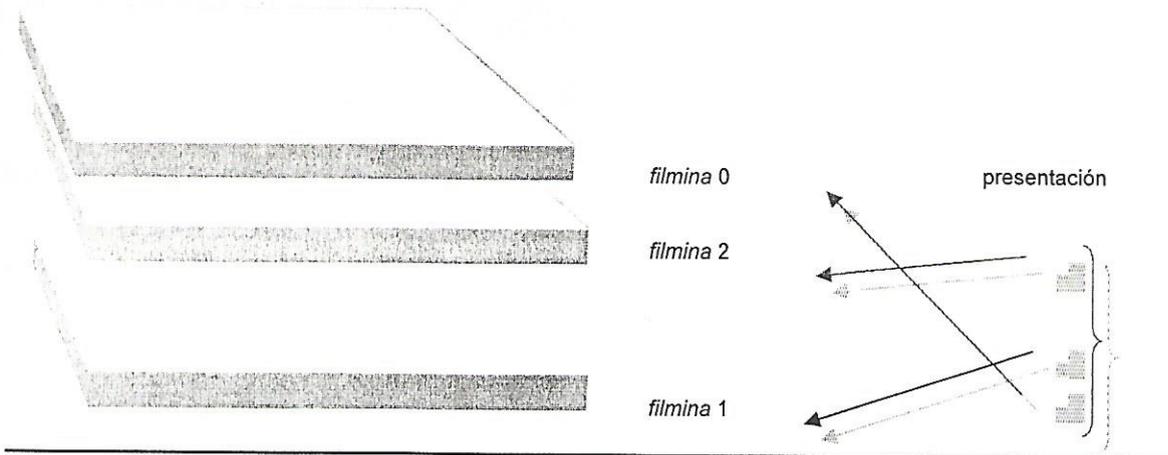


Figura 7: Presentación después de borrar una filmina.

2.1.2.10 Insertar Filminas

Se crea un *movieClip* nuevo, luego, dependiendo la posición en la presentación, el valor del índice, se utilizan tres métodos de manipulación de arreglos de Flash MX 2004, de acuerdo a si el mismo se encuentra al final, medio o principio de la misma, respectivamente. Estos son *push*, *splice* y *unshift*. Se explican a continuación.

La presentación vacía se reconoce por el valor del índice de *filmina* activa (con valor nulo). En ese caso se utiliza el método *push* de Flash MX 2004 para insertar una nueva *filmina*. El nuevo índice pasa a tomar el valor 0.

Con una presentación no vacía, con el índice al final del arreglo, se utiliza push para insertar un nuevo elemento, y el nuevo índice es la propiedad length del arreglo -1 (la *filmina* recién creada, cuya referencia se inserta al final del arreglo).

Con una presentación no vacía, con el índice en el medio del arreglo se utiliza splice (índice actual, 0, nueva *filmina*), donde índice actual es el valor del índice en este momento, donde se está insertando. El valor 0 indica que no se están borrando elementos, y nueva *filmina* es la referencia a insertar dentro del arreglo. El valor del índice no se modifica.

Con una presentación no vacía, con el índice al principio del arreglo, se utiliza el método unshift, que inserta un nuevo elemento al principio del arreglo. El valor del índice se mantiene (0).

En todos los casos, se muestra la *filmina* recién creada.

Ejemplo: insertar una nueva *filmina* en la posición 0 de una presentación.

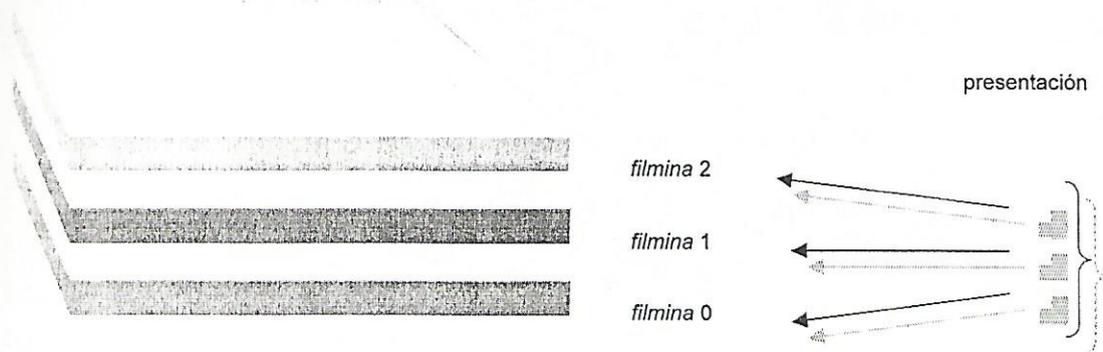


Figura 8: Presentación antes de insertar una *filmina*.

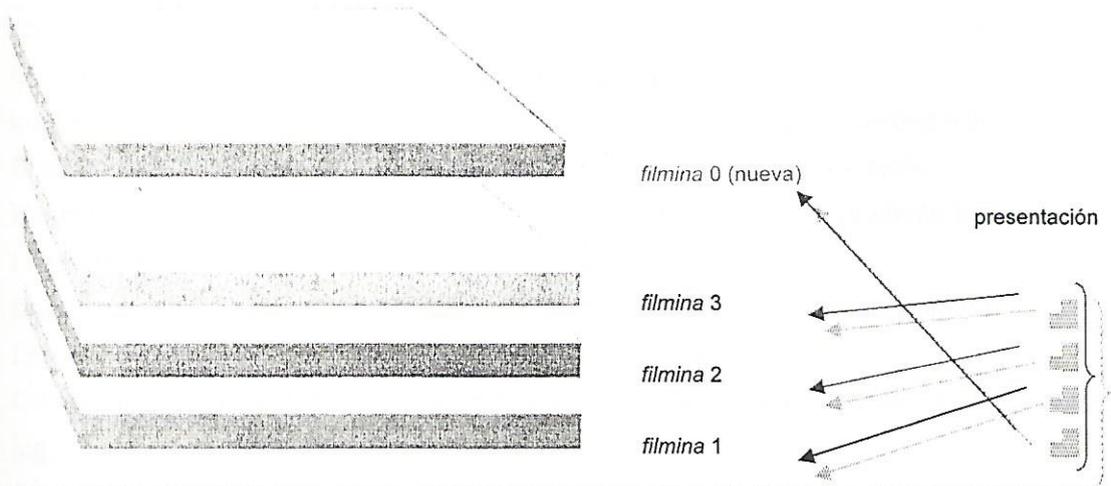


Figura 9: Presentación después de insertar una *filmina*.

2.1.2.11 Insertar imagen

Carga de Imágenes



Captura de pantalla 11: Inserción de una imagen en una filmina.

La lista de imágenes disponibles se obtiene a través de la ejecución de un archivo de comandos .php.

Con la creación de un objeto LoadVars se ejecuta este archivo.

LoadVars devuelve con el método sendAndLoad la lista de imágenes disponibles.

Si es imposible cargar la lista de imágenes se muestra un mensaje de error.

Por convención, la lista de imágenes se guardan en el archivo de la siguiente forma:

Imagen=url1;url2;url3

Donde url1, url2 y url3 representan una ruta para encontrar los archivos de imágenes.

Los archivos de imágenes son imágenes de formato JPG.

Con la herramienta Flash MX 2004 no es posible cargar imágenes de otro formato que no sea jpg.

Ejemplos de valores válidos de url: imagen.jpg, <http://www.sitio.com/img/imagen.jpg>

Si la url no es válida, se muestra un mensaje de error.

Las imágenes a insertar en una *filmina* se previsualizan en el *movieClip* de inserción de imágenes.

Este *movieClip* está creado en tiempo de diseño y tiene un tamaño de 500 por 500 pixeles.

Las imágenes que se previsualizan se ajustan proporcionalmente a este *movieClip*.

Las imágenes se cargan utilizando un objeto de la clase *MovieClipLoader* de Flash MX 2004.



Con el método `loadClip` se carga la imagen, y se indica el destino donde cargarla.

El destino es el *movieClip* de previsualización.

Para el objeto *movieClipLoader* que se está utilizando se definen acciones para los eventos de: progreso de carga, error de carga y finalización de carga.

Estas acciones son funciones que se asocian a un objeto genérico.

Este objeto genérico "escucha" estos eventos.

Se registra este objeto con el objeto *movieClipLoader*.

2.1.2.11.a Progreso de carga

A medida que se obtiene información con respecto a la cantidad de bytes de la imagen que se está cargando, se cambia el tamaño del *movieClip* que es la barra de progreso de carga.

2.1.2.11.b Error de carga

Si hay error en la carga por url no válida, se muestran mensajes correspondientes.

El código para mostrar los mismos se encuentran dentro de esta función.

2.1.2.11.c Finalización de carga

Al finalizar la carga de la imagen, ésta se ajusta proporcionalmente al *movieClip* que se utiliza de destino.

Si no se cambiara el tamaño del *movieClip* luego de la carga de la imagen, éste tendría el tamaño de la imagen que se cargó.

En la parte inferior a la vista previa es posible visualizar el progreso de carga de cada una de las imágenes.

En la parte inferior a la vista previa también se encuentran los controles de navegación para recorrer la lista de imágenes recuperada con el método `sendAndLoad`.

Si la imagen que se está previsualizando se quiere insertar en la *filminta*, se presiona el botón para aceptar, y la misma se inserta en la *filminta*.

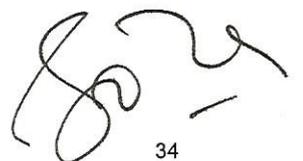
Para insertar una imagen se agrega al arreglo de objetos de la *filminta* una referencia a la imagen insertada.

Luego se crea un *movieClip* que contiene controles para moverlo y destruirlo, y contiene a su vez la imagen.

El movimiento y destrucción de este *movieClip* se realiza con dos *movieClips* anidados en el mismo con acciones asociadas.

Si se quiere cancelar la inserción de imágenes, se presiona el botón para salir de esta opción.

2.1.2.12 Insertar texto





Captura de pantalla 12: Texto a insertar y sus propiedades.

Cuando se quiere insertar texto en una *filmina*, se presiona el botón de menú principal para crear un cuadro de texto.

Se crea un cuadro de texto que por defecto tiene el texto "Inserte su texto aquí".

El objeto cuadro de texto es, en conjunto, un *movieClip* que contiene 4 *movieClips* anidados, y un cuadro de texto no seleccionable, donde se visualiza el texto con el formato elegido.

Los 4 *movieClips* se utilizan como botones.

Uno de ellos sirve para desplazar el cuadro de texto sobre el área de la *filmina*.

Otro sirve para acceder a establecer de propiedades del texto.

Otro sirve para destruir el cuadro de texto y quitarlo del arreglo de objetos activos para esa *filmina*.

Otro sirve para cambiar el tamaño del mismo.

2.1.2.12.a Mover el cuadro de texto

Un *movieClip* recibe los eventos de mouse para desplazar el cuadro de texto.

Al hacer click sobre el mismo, se trae al *movieClip* padre al frente, es decir, a una *profundidad* mayor que los otros objetos.

Se lo empieza a arrastrar cuando se lo clickea, y se lo deja de arrastrar cuando se suelta el botón del mouse.

2.1.2.12.b Establecer las propiedades del texto

Las propiedades a establecer del texto son su tipografía, color y tamaño.

2.1.2.12.b.1 Paleta de colores

La paleta de colores se utiliza para establecer el color del texto.

Está compuesta por 6 subpaletas con 36 colores cada una. En total son 216 colores. Se denomina Web Safe.

2.1.2.12.c Destruir un cuadro de texto

El *movieClip* se destruye y se quita del arreglo de objetos activos de la *filmina* actual.

2.1.2.12.d Cambiar el tamaño del cuadro de texto

Al hacer click en el botón para cambiar el cuadro de texto, se define una función, donde se chequea la posición del mouse.

Sabiendo la posición del mouse se arrastra el *movieClip* para el cambio de tamaño.

También se cambia de tamaño el cuadro de texto dentro del *movieClip* cuadro de texto.

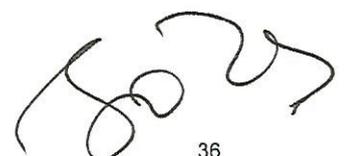
Al soltar el *movieClip* para cambiar el tamaño de cuadro de texto se deja de arrastrarlo.

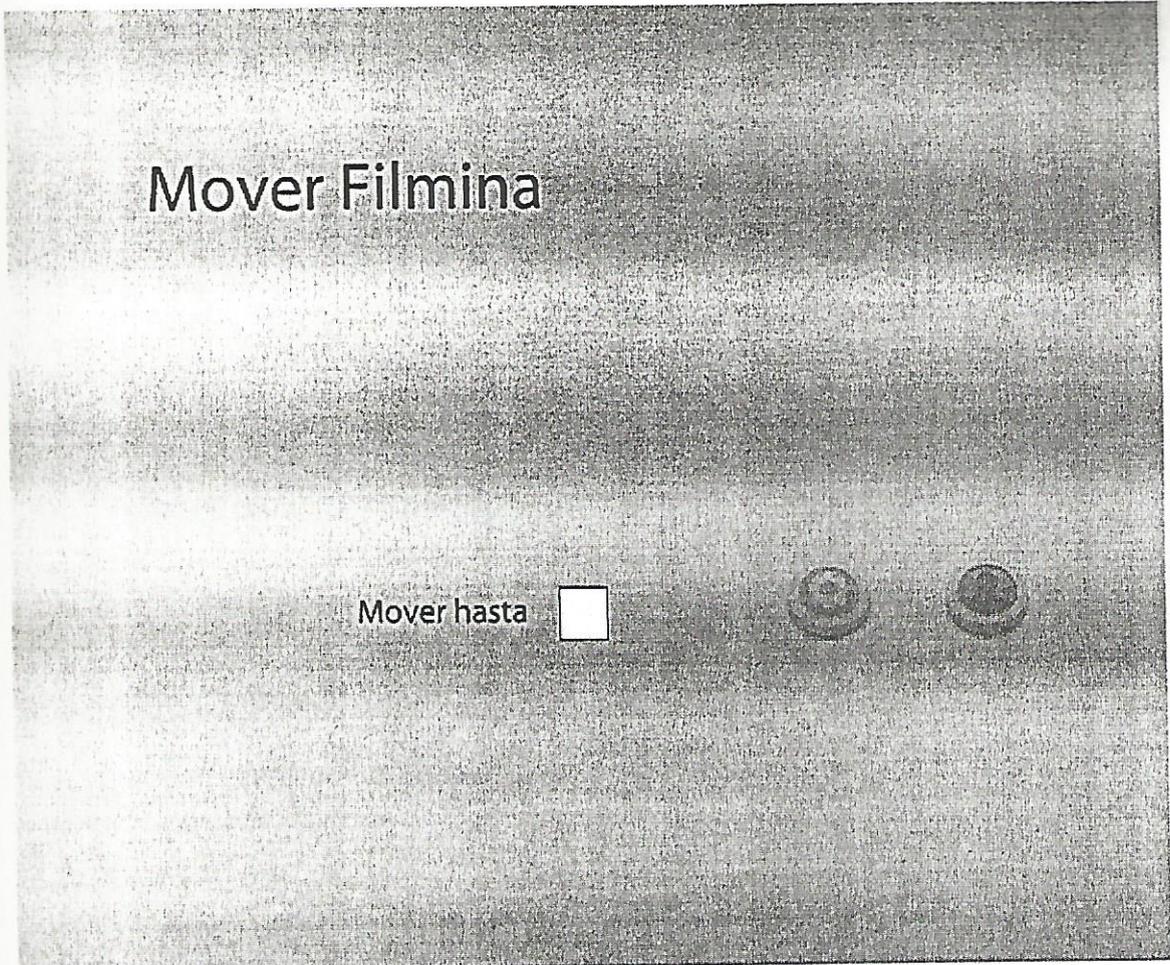
También se destruye la función que verifica constantemente la posición del mouse para ahorrar tiempo de procesador y porque no es más necesaria.

2.1.2.12.e Manejo de errores

Si se intenta crear un cuadro de texto sin una *filmina* activa (con una presentación vacía) se muestra un mensaje de error.

2.1.2.13 Cambiar la posición de una filmina





Captura de pantalla 13: Cambiar la posición de una filmina.

Para llamar a esta opción es necesario que la presentación no esté vacía.

Se utiliza una variable de tipo *movieClip* como auxiliar para almacenar en forma intermedia la *filmina* en la posición origen para moverla a la posición destino.

2.1.2.14 Impresión de Filminas

Imprimir

Imprimir toda la presentación

Imprimir desde hasta

- Impresión optimizada para Imágenes
- Ajustar filmina al área imprimible del papel

Captura de pantalla 14: Impresión de filminas.

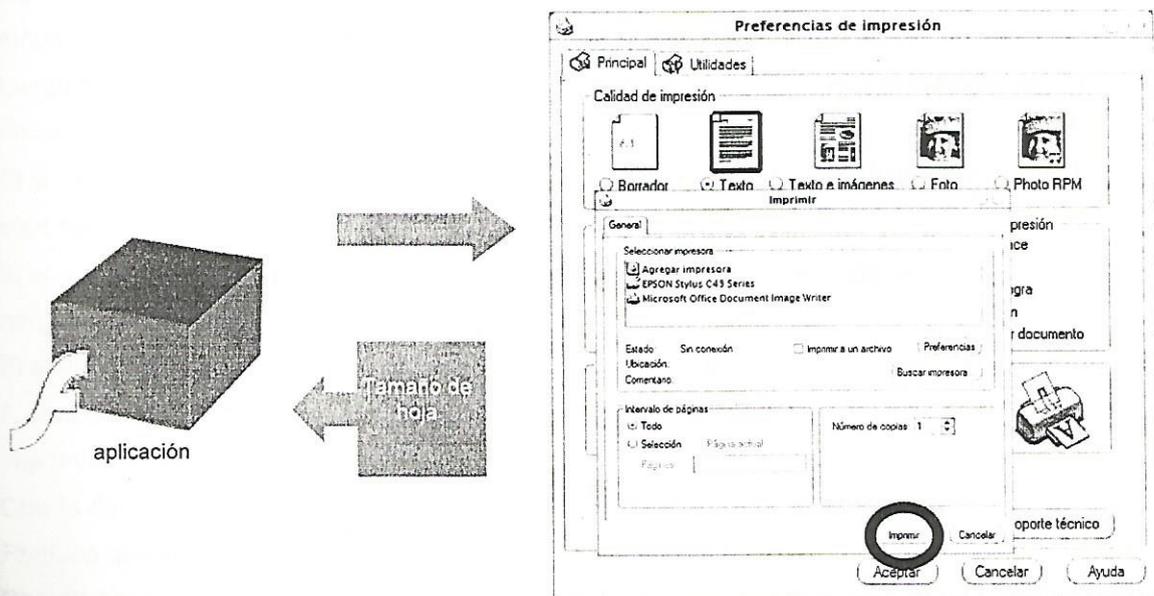


Figura 10: Intercambio de datos entre el diálogo de impresión del sistema operativo (e impresora) y la aplicación.

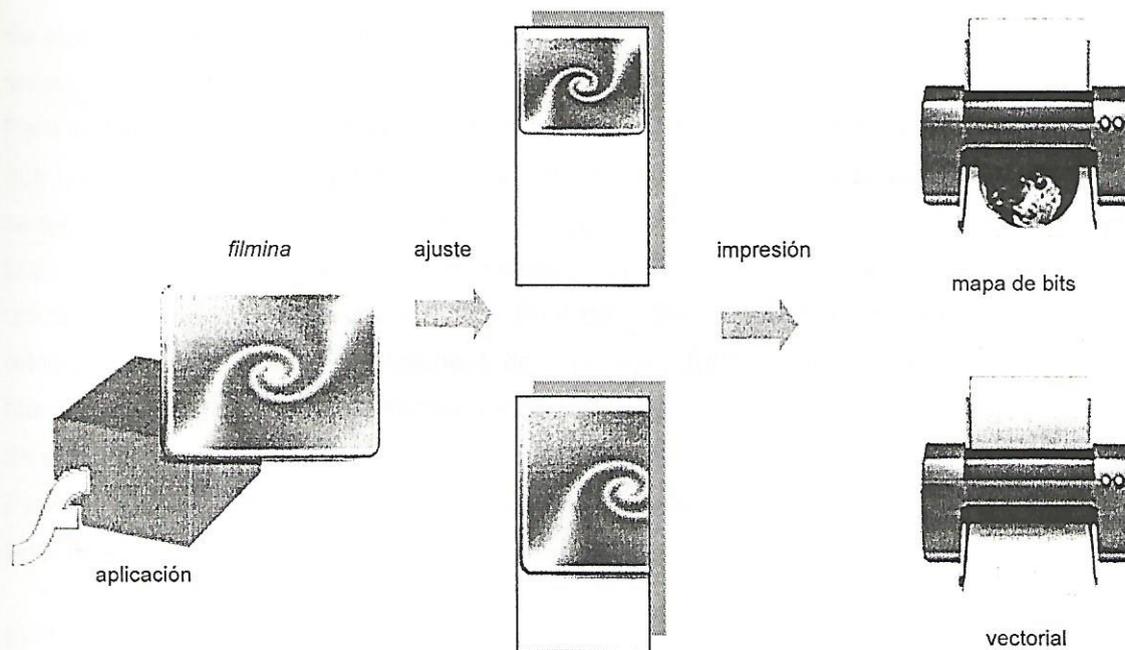


Figura 11: Ajuste de la filmina al área imprimible del papel (o no) e impresión en modo vectorial o mapa de bits..

Se ofrece la posibilidad de imprimir un intervalo de *filminas* o la presentación entera.

En ambos casos se utiliza una misma función para enviar las *filminas* a imprimir a la impresora.

Dentro de la función se crea una instancia del objeto *PrintJob* de Flash MX 2004.

Con la llamada al método *start* de la clase *PrintJob*, se invoca al cuadro de diálogo del sistema operativo con las impresoras disponibles, para que se elija dónde se imprimirá.

También, para una impresora determinada, está la opción de elegir la calidad de impresión y tamaño de hoja a utilizar. Esto depende del panel de establecimiento de propiedades de cada impresora, según los drivers instalados en la máquina que se está utilizando la aplicación, modelo de impresora y sistema operativo.

Luego que se eligió la impresora el usuario presiona OK. En este caso el método *start* retorna un valor booleano igual a *true*.

Si el usuario cancela la selección y establecimiento de propiedades de la impresora, el método *start* retorna *false*.

Si el valor retornado por el método *start* es *false*, se muestra un mensaje de error y no se envía ninguna *filmina* a imprimir.

Si el valor retornado por el método *start* es *true*, se procede.

A continuación se chequea si el usuario quiere imprimir ajustando el área de impresión al área imprimible del papel que eligió.

Con la llamada al método *start*, presionando el usuario OK, se instancian atributos del objeto *PrintJob* que se está utilizando para imprimir.

Dos de esos atributos son el ancho y alto de la zona a imprimir (es decir, el ancho y alto de la hoja, quitándole los márgenes predeterminados del papel).

En base a estos atributos se procede de la siguiente forma.

Se toma el ancho y alto de cada *filmina* a imprimir, y se los ajusta proporcionalmente (o no) al ancho y alto del área imprimible.

Para ajustar los valores del área a enviar a imprimir, se modifica el tamaño de cada *filmina* para que quepan en la misma, y luego de enviar a imprimir, se vuelven a su valor original. Además se quita el borde de los cuadros de texto para que no se impriman.

Luego que están cambiadas las propiedades del *movieClip filmina*, se envía a imprimir utilizando el método `send` de la clase *PrintJob*. Dentro del mismo interviene la variable relacionada con la calidad seleccionada de impresión. Esta puede ser óptima para mapas de bits, u óptima para gráficos vectoriales (por ejemplo, *filminas* que contengan sólo texto y relleno de color sin imágenes).

Para apreciar las diferencias entre las distintas calidades de impresión en diversos casos, ver Apéndice C.

El método `send` retorna un valor booleano igual a `true` si la página se envió satisfactoriamente a imprimir y `false` en caso contrario.

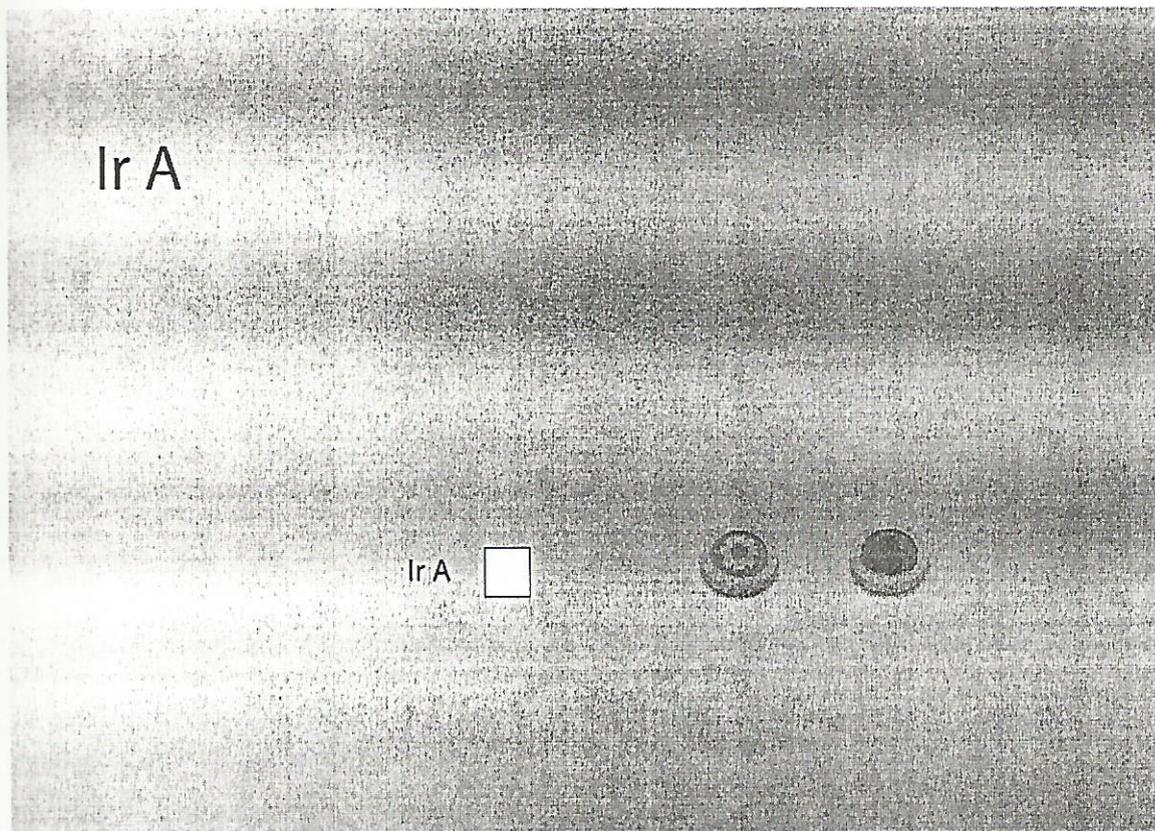
Se muestra un mensaje si ocurrió un error.

Si no ocurrió un error, se incrementa una variable, que al final de mandar a imprimir todas las *filminas* se chequea.

Si éste es igual a cero, no se pudieron enviar las *filminas* a imprimir y hubo un error. En este caso se muestra un mensaje.

Luego de encolar el trabajo completo a imprimir, se destruye el objeto de la clase *PrintJob* que se utilizó.

2.1.2.15 Ir A



Captura de pantalla 15: Ir a filmina.

Dada una presentación no vacía, cambia la *filmina* activa (visible).

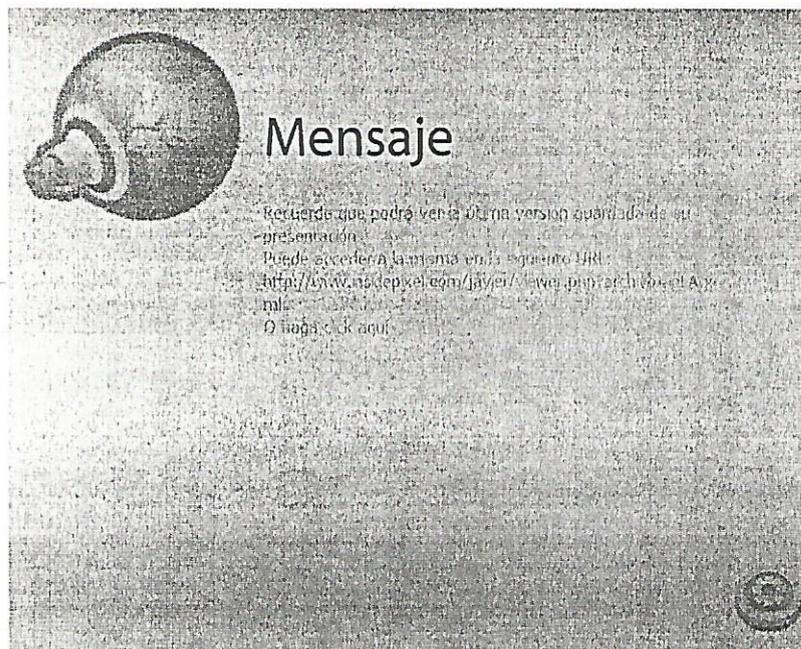
En el cuadro de texto de este *movieClip* se ingresa la posición de la *filmina* que se desea visualizar.

Si el número ingresado está en el rango [1;cantidad de *filminas*], se oculta la *filmina* actual y se muestra la *filmina* deseada.

2.1.2.16 Previsualizar

Dirección

Hipervínculo (abre
en ventana de
navegador)



Captura de pantalla 16: Previsualización de una presentación.

Cuando un usuario guarda su presentación o presiona el botón de previsualizar en el menú principal, se le presenta un mensaje (ver captura de pantalla 16) con un cuadro de texto seleccionable con la dirección de su presentación y uno no seleccionable con un hipervínculo a la misma.

Tiene la opción de seleccionar el texto de la dirección y pegarlo (por ejemplo si quiere compartir la dirección del hipervínculo de su presentación con otro usuario) o directamente hacer clic sobre el hipervínculo que dice "O haga clic aquí" (en cuyo caso se abre una ventana de navegador de Internet con el proyector de *filminas*). Ver 2.2.

2.1.2.17 Acerca De

A handwritten signature or set of initials in black ink, located in the bottom right corner of the page. The signature is stylized and appears to consist of several loops and a final horizontal stroke.

Acerca De

Prototipo de aplicación para creación y
visualización de presentaciones
Universidad de Belgrano
Año 2005
Alumno: Javier Re

Captura de pantalla 17: Acerca de.

2.1.3 Comunicación remota

Se solicita la ejecución de las siguientes acciones desde la aplicación, en un servidor remoto.
Se llaman con un objeto de la clase *LoadVars*.

2.1.3.a Borrar una presentación

Se recibe un nombre de archivo a borrar.

Se borra el mismo.

Esta acción puede ser exitosa o no.

El resultado de la acción se asocia a una variable que se retorna y chequea para mostrar un mensaje de error en la aplicación si es necesario.

2.1.3.b Guardar una presentación

Se reciben: el nombre de archivo a guardar y la cadena con la representación en XML de la presentación, enviados desde la aplicación.

Se abre el archivo con el nombre recibido, se escribe la cadena, y se cierra el archivo.

El resultado de las acciones se asocian a una variable que se retorna y chequea para mostrar un mensaje de error en la aplicación si es necesario.



2.1.3.c Listar las imágenes disponibles para cargar en una filmina

Se ejecuta una lectura de directorio con filtro.

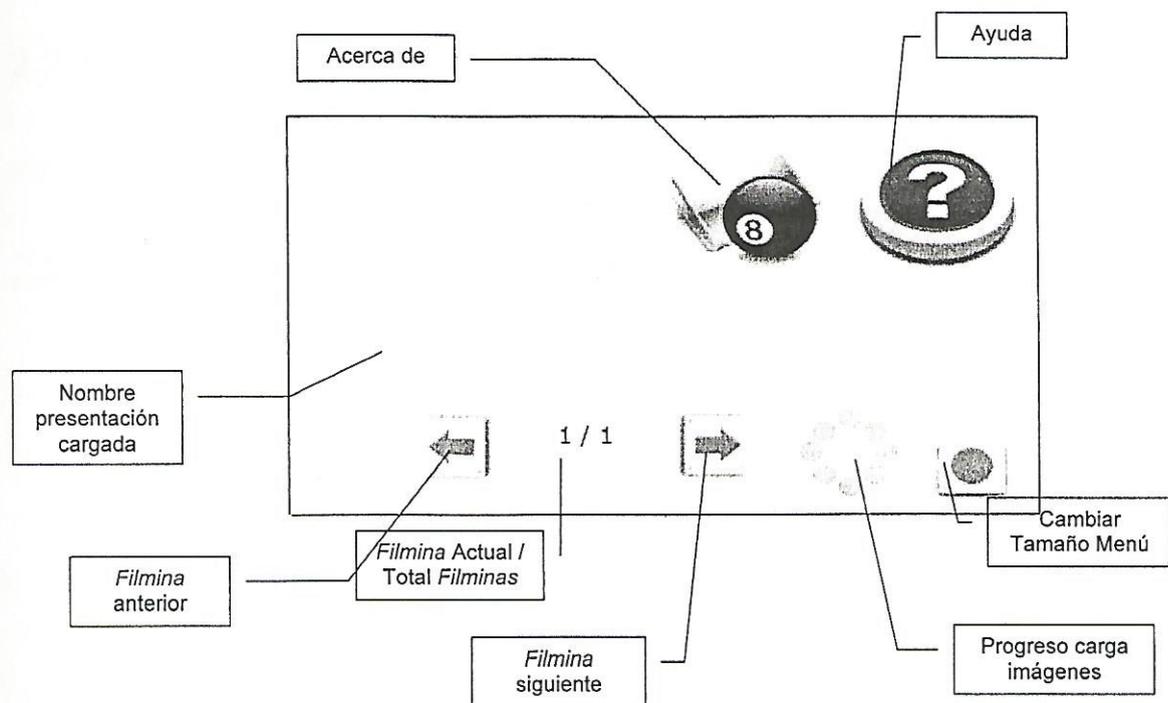
Se retorna la lista de archivos con extensión jpg que se encuentran en el directorio de imágenes, separados con punto y coma.

2.1.3.d Listar las presentaciones disponibles para cargar

Se ejecuta una lectura de directorio con filtro.

Se retorna la lista de archivos con extensión xml que se encuentran en el directorio de presentaciones, separados con punto y coma.

2.2 Visualización de Presentaciones



Captura de pantalla 18: Menú principal del proyector de filminas

La previsualización de presentaciones se realiza en el proyector de *filminas*.

El proyector de *filminas* es la segunda parte de la aplicación, que permite visualizar presentaciones ya creadas.

La presentación que se carga está en la dirección para llamar al proyector.

En el proyector se toma la variable de la dirección. Luego esta variable es accesible desde el mismo y se utiliza para cargar la presentación que se va a ver.

El proyector ofrece la posibilidad de navegar una presentación existente.

Las capacidades de agregado/quitado y edición de *filminas* no se encuentran disponibles.

Al iniciarse se carga una presentación y se va a la primer *filmina*.

Para navegar la presentación, se usan los botones del menú para tal fin, o bien las teclas de flecha derecha o avpág para avanzar una *filmina*, y flecha izquierda o repág para retroceder una *filmina*.

Para mostrar u ocultar el menú se utiliza la barra espaciadora.

Para comenzar el modo de reproducción, se utiliza la tecla Enter.

Cuando se pasa de una *filmina* a otra se efectúa la animación seleccionada en tiempo de diseño.

2.2.a Manejo de errores

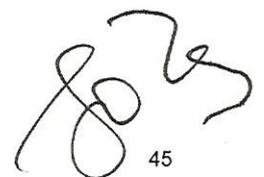
El manejo de errores de carga de imágenes para fondos o imágenes en cada *filmina* es igual al manejo de errores en modo creación de presentaciones.

Se muestra un error en caso de no haber encontrado la presentación a cargar.

2.2.1 Acerca De

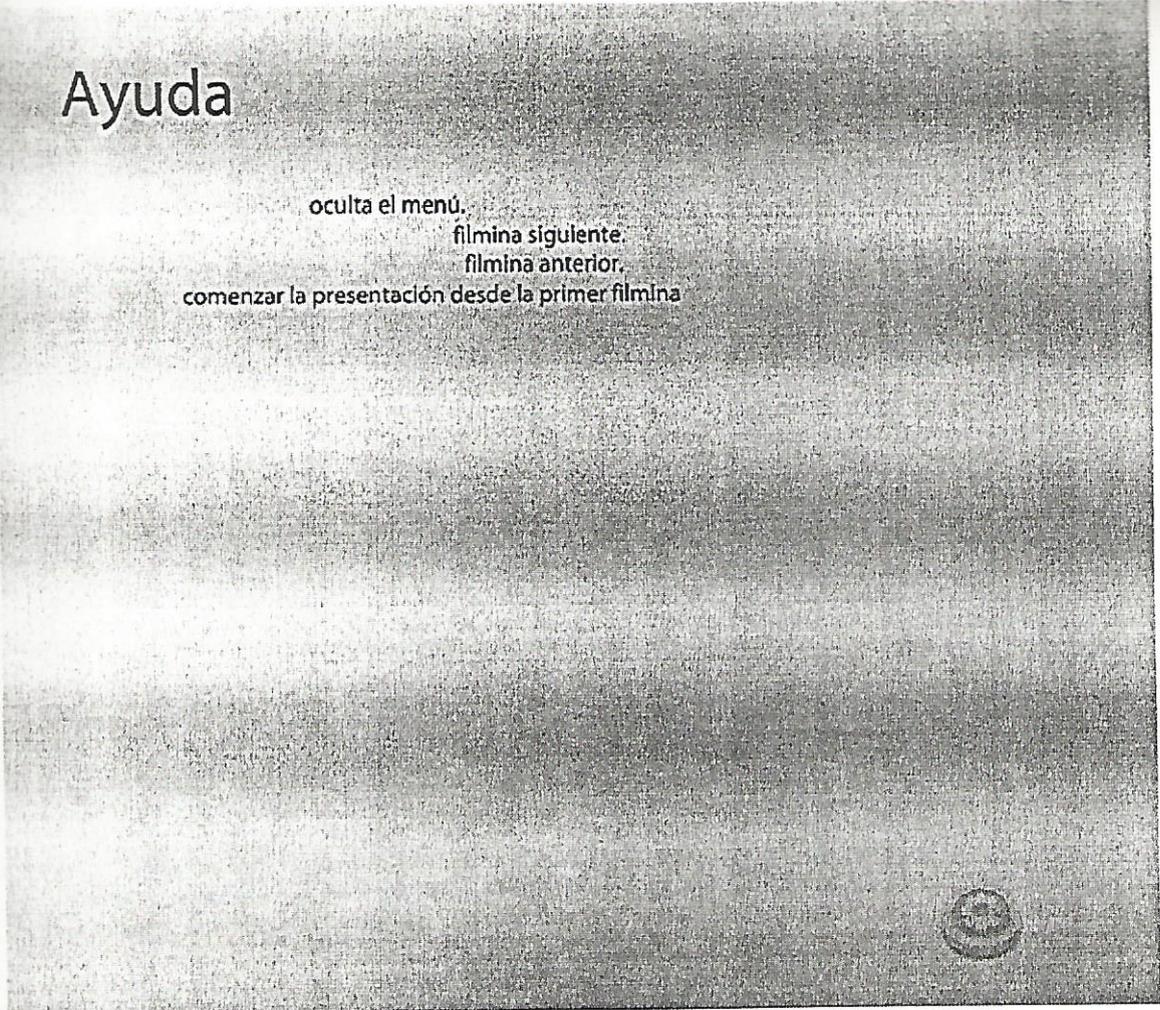
Ver 2.1.2.17.

2.2.2 Ayuda



Ayuda

oculta el menú.
filmina siguiente.
filmina anterior.
comenzar la presentación desde la primer filmina

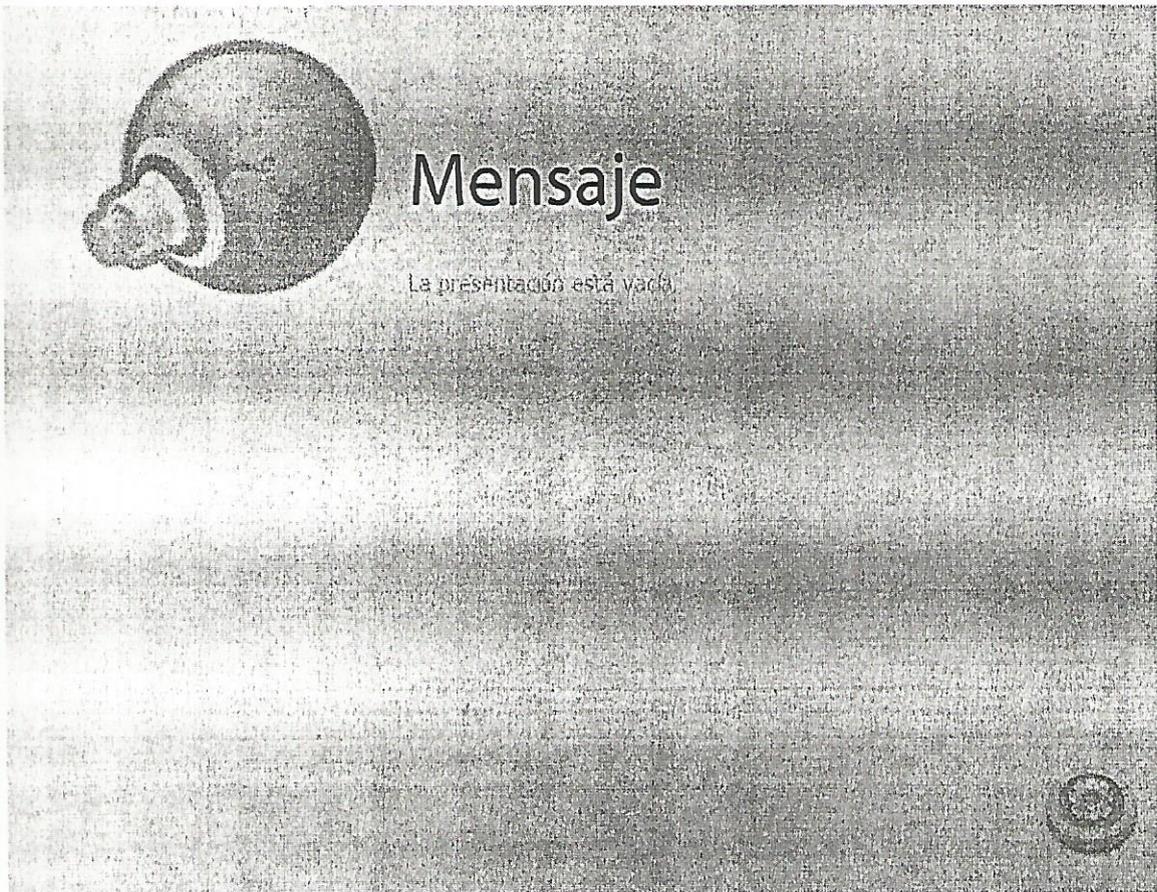


Captura de pantalla 19: Ayuda del proyector de filminas

2.3 Mensajes de error

Los mensajes de error se muestran en pantalla de la siguiente forma:

A handwritten signature or set of initials in black ink, located in the bottom right corner of the page.



Captura de pantalla 20: Mensaje de error de ejemplo.

El texto descriptivo del error varía en cada caso, según el error que haya ocurrido.

2.4 Hallazgos

A continuación se describen los hallazgos, relacionados con la utilización de la herramienta Flash, en el desarrollo de la aplicación.

2.4.a Carga de imágenes

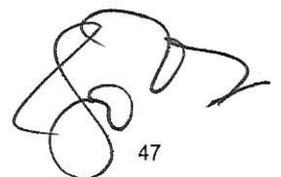
El manejo de errores de carga de imágenes se tuvo que limitar a la posibilidad de no encontrar una imagen para cargar debido a una dirección no válida.

Se contemplaba el escenario en que la imagen se comience a cargar y no se finalice su carga. Esto ocasionaba la aparición de mensajes de error, cuando estos realmente no ocurrían.

Con respecto a los métodos para cargar imágenes, se utiliza el método `loadClip` de la clase `MovieClipLoader`, en vez del método `loadMovie`.

Con el último no es posible capturar los errores de carga.

2.4.b Caché del Navegador



Cuando se invoca a un archivo de comandos, se llama al mismo junto con una variable que toma un valor siempre distinto (número aleatorio).

Se tuvo que recurrir a esta solución porque después de la primer llamada, el archivo con la respuesta remota queda almacenada en el caché del navegador.

Si se actualiza la lista de presentaciones o imágenes y ésta se quiere visualizar mientras se está ejecutando la aplicación, la misma tiene un valor no actualizado (tiene el valor de la primera vez que se ejecutó, almacenado en caché).

2.5 Pruebas

La aplicación se testeó en los navegadores Internet Explorer 6.0, Mozilla Firefox 1.0.6, Opera 8.02 y Netscape Browser 8.0.3.3 en plataforma Windows XP con Service Pack 2, con conexión de banda ancha, en una computadora Pentium 4 con 512 MB de RAM.

Se crea una presentación con 1100 *filminas*. Todas las *filminas* tienen una imagen como fondo.

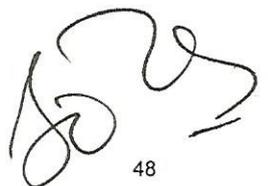
Se guarda y luego se carga esta presentación. Aparece un error de carga de imágenes.

Luego se verifican todas las *filminas* de la presentación, y las imágenes se encuentran todas cargadas.

En el momento de imprimir, el equipo utilizado para la aplicación deja de responder.

Se supone que esta última falla está relacionada con la cantidad de *filminas* enviadas a imprimir.

Luego se realiza otra prueba. Se crean una presentación con 10 *filminas* y se guarda. Luego se accede al modo proyector para visualizarla. No se encuentran errores.



3. Tercera Parte

3.1 Conclusión

La aplicación sirve a su propósito a pesar de las limitaciones impuestas en la funcionalidad.

En forma satisfactoria se crea una Rich Internet Application de autoría de contenido (compuesta por otra, necesaria para la visualización del contenido creado).

El usuario cuenta con una herramienta de autoría de presentaciones, puede crearlas, editarlas, imprimirlas, exponerlas, compartirlas y borrarlas.

3.2 Contribuciones

Se contribuye creando una aplicación que se ejecuta en un navegador de internet, que tiene funcionalidad de herramienta de oficina, que no sólo permite al usuario el almacenamiento de contenido del que pueda disponer desde cualquier computadora con conexión a internet, sino también la posibilidad de compartir este contenido con terceras partes.

3.3 Líneas Futuras de Desarrollo

Se podría agregar funcionalidad a la aplicación, por ejemplo: posibilidad de incrustar video y audio en cada *filmina*, hacer animaciones para cuadros de texto en modo proyector, cargar imágenes de otro formato de archivo, cargar otras aplicaciones realizadas en Flash MX 2004 como objetos de *filmina*, cargar *filminas* en otro formato de archivo (en lo posible, guardado con una herramienta de tipo comercial), crear diferentes cuentas para diferentes usuarios, donde cada uno cree, edite y visualice sus presentaciones, darle a cada usuario la posibilidad que administre las imágenes para incluir en las *filminas* de sus presentaciones.



Glosario

_alpha: transparencia. Valor que va desde 0 (completamente transparente) a 100 (completamente opaco).

Anidado: un movieClip contenido dentro de otro.

Filmina: diapositiva o slide. Elemento básico que constituye una presentación.

movieClip: objeto de Flash MX 2004.

loadMovie: método para cargar imágenes en formato jpg en un movieClip de destino.

LoadVars: clase utilizada para la transferencia de datos entre una aplicación desarrollada en Flash MX 2004 y un servidor.

MovieClipLoader: clase para manejar la carga de imágenes.

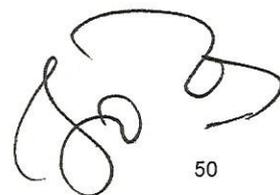
Padre: un movieClip que contiene a otro se llama padre de este otro.

PrintJob: clase utilizada para impresión.

Profundidad: ubicación de un movieClip en el eje z. Determina si un movieClip va a ser mostrado en frente o atrás de otro.

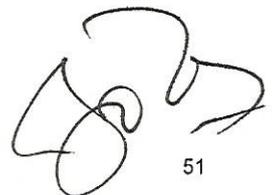
Stage: lugar de la aplicación donde se colocan texto y otros elementos como imágenes o movieClips.

Tween: método para hacer animaciones.



Bibliografía

- I. REINHARDT, Robert; LOTT, Joey, Flash MX 2004 ActionScript Bible, Estados Unidos, Wiley Publishing, Inc., 2004. (e-book)
- II. MOOCK, Colin, ActionScript for Flash MX: The Definitive Guide, Segunda Edición, Estados Unidos, O'Reilly & Associates, 2003.
- III. LOTT, Joey, Actionscript Cookbook, Estados Unidos, O'Reilly & Associates, 2003.
- IV. JACOBSON, Dov; JACOBSON, Jesse, Flash and XML, A Developer's Guide, Estados Unidos, Addison-Wesley, 2001.
- V. Macromedia. Flash MX 2004 Actionscript Reference Guide. Estados Unidos, 2003. (e-book)
- VI. Macromedia. Getting Started with Flash. Estados Unidos, 2003. (e-book)
- VII. Macromedia. Learning Flash. Estados Unidos, 2003. (e-book)
- VIII. Macromedia. Using Components. Estados Unidos, 2003. (e-book)
- IX. Macromedia. Using Flash. Estados Unidos, 2003. (e-book)
- X. Macromedia. Developing Rich Internet Applications with Macromedia MX 2004. Estados Unidos, 2003. (e-book)



Apéndice A

Diagramas de Ideas (brainstormings)

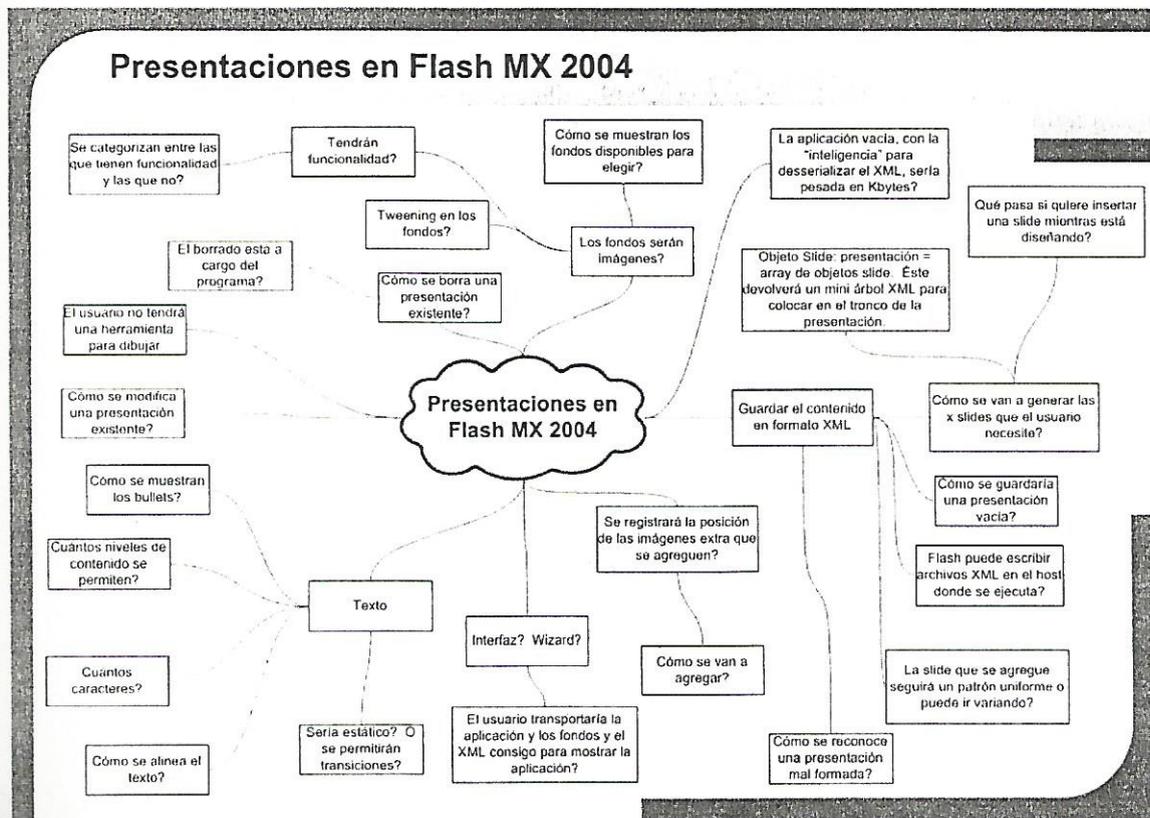
En forma meramente indicativa se utilizan leyendas en los diagramas.

Se enfatizan ciertas ideas sobre otras, aplicando una leyenda sobre el rectángulo que las representa.

Una idea puede ser descartada, luego revisada, y más tarde reconsiderada y agregada, o bien una idea previamente considerada y aceptada puede descartarse en una revisión posterior.

- Idea que se considerará preponderante
- Idea que se considerará menos importante
- Idea que se descartará
- Idea cuya solución o implementación no se conoce

A.1 Diagrama 1



En este primer diagrama es posible observar las ideas al comienzo del desarrollo de la aplicación.

Se utiliza el término "slide" como sinónimo de la palabra *filmina*.

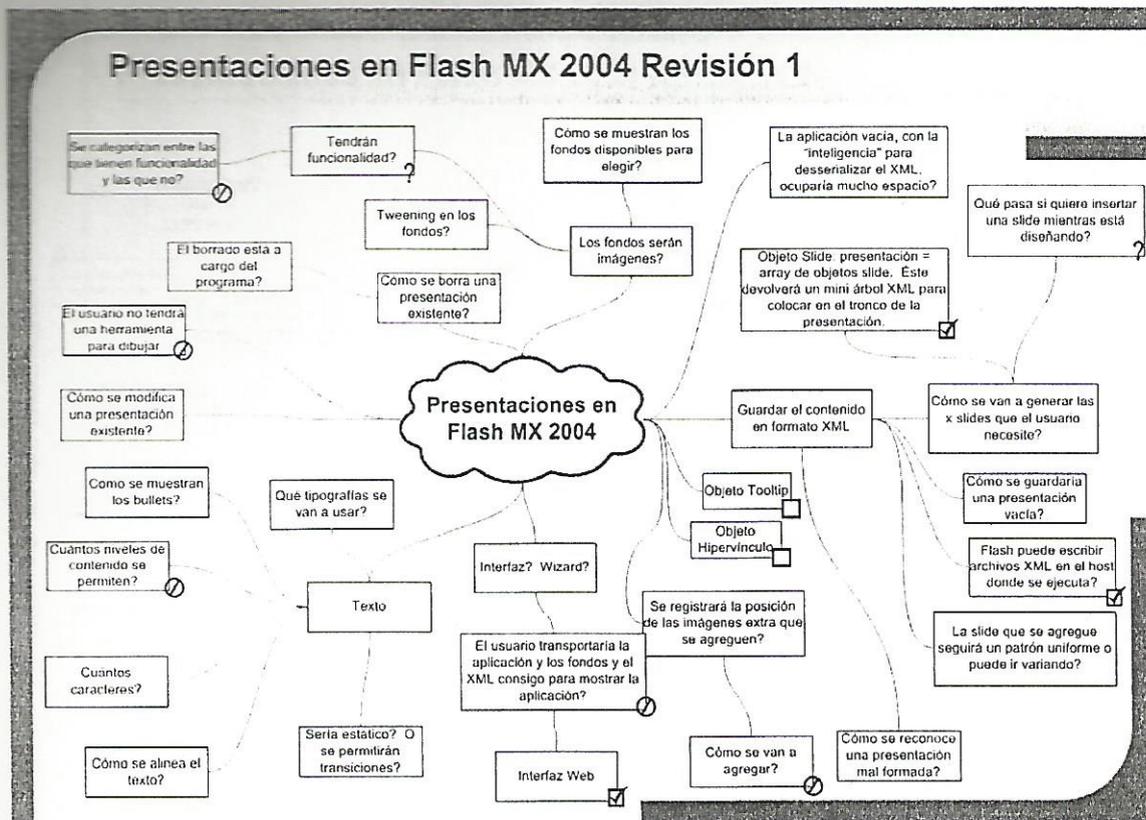
Algunas ideas que luego no se consideran:

- Usar un wizard o asistente para crear *filminas* en la aplicación.
- Los fondos tienen funcionalidad (no son imágenes estáticas y es posible interactuar con los mismos).
- Varios niveles de contenido, es decir, varias jerarquías de viñetas (o bullets) para insertar en el texto.
- Uso de plantillas (templates).
- Texto con animación.

Algunas ideas que luego se consideran:

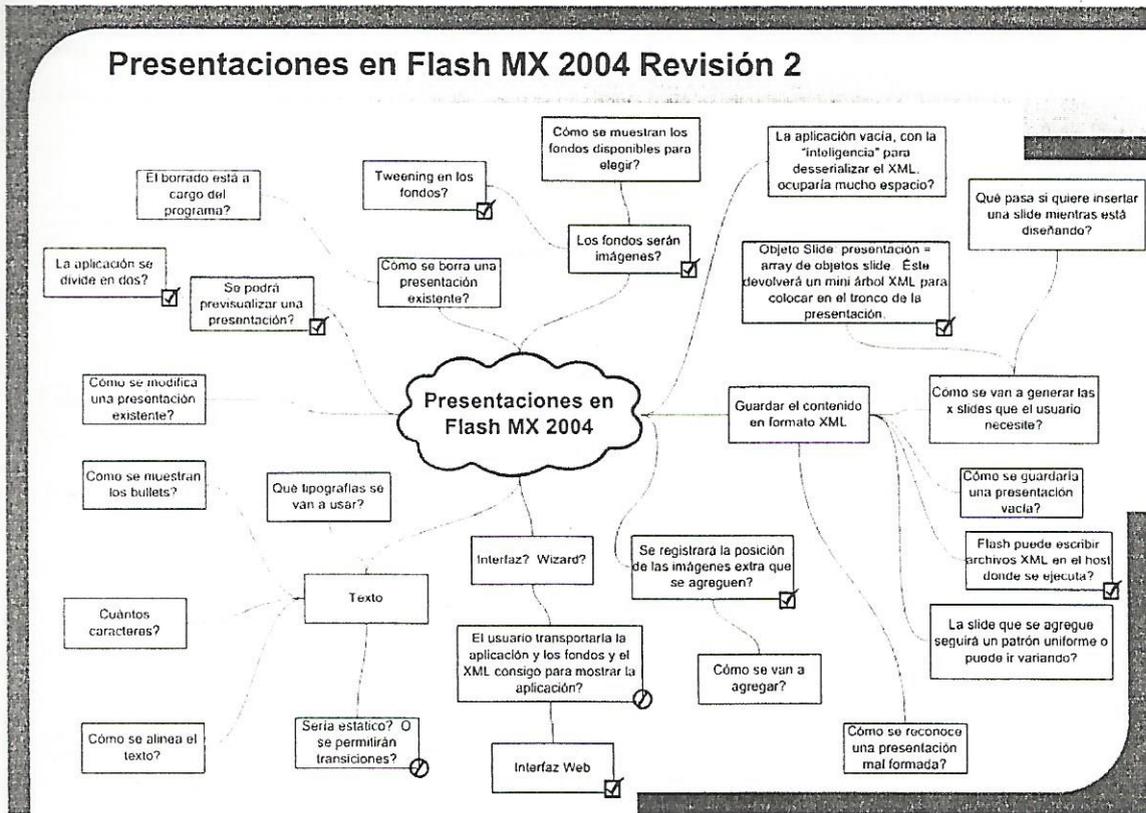
- El contenido se guarda en formato XML.
- Se utiliza animación para las *filminas*.
- El usuario no tendrá una herramienta para dibujar.
- La aplicación y sus componentes no necesitan ser portados por el usuario.



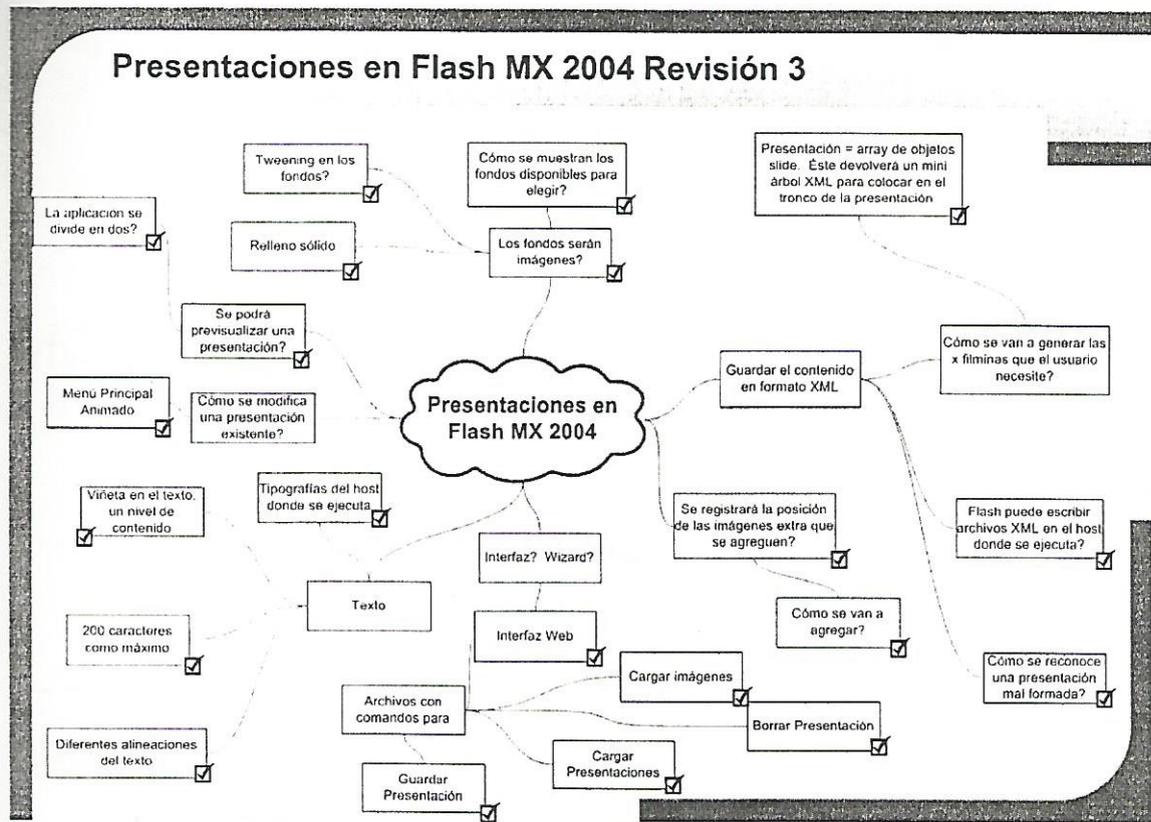


En este diagrama están destacadas las ideas acerca del trabajo con objetos de tipo tooltip (ayuda al mover el Mouse sobre un objeto insertado en una *filmína*) y objeto hipervínculo (cuadro de texto insertado en una *filmína* con un hipervínculo). Ambas son consideradas en esta única revisión.

A.3 Diagrama 3



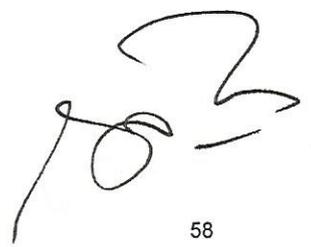
A.4 Diagrama 4



En esta última revisión es posible apreciar solamente aquellas ideas que se consideran para el desarrollo de la aplicación.

En este diagrama se muestra cada hito, tarea o componente desarrollado para conformar la aplicación.

Ver Segunda Parte.



Apéndice C

Las siguientes imágenes fueron capturadas utilizando como impresora el software Microsoft Office Document Imaging versión 11.0.1897.0.

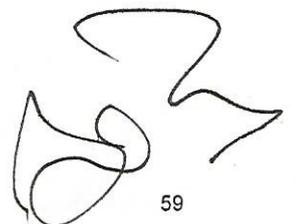
Las mismas están agrandadas un 500% y recortadas.

C.1 Impresión de texto en modo óptimo para gráficos vectoriales

ste es un t
ueba Es

C.2 Impresión de texto en modo óptimo para mapas de bits

ste es un t
ueba Es



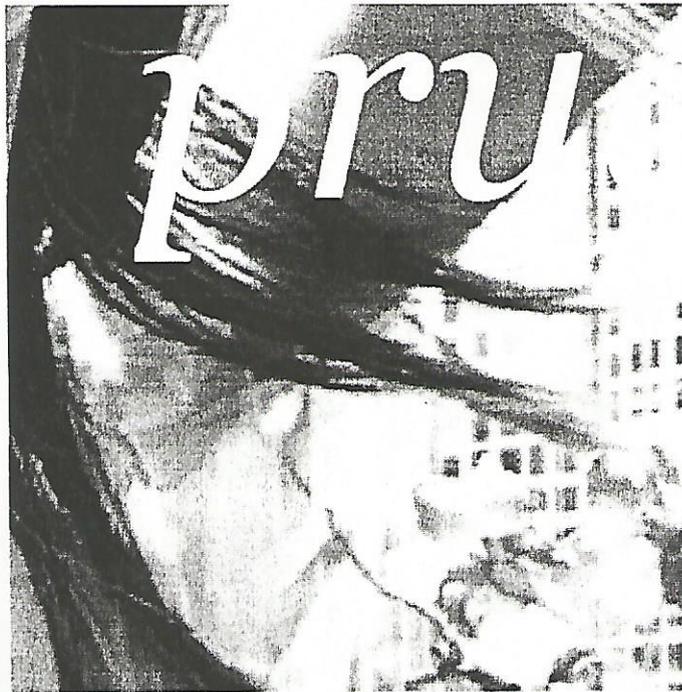
C.3 Impresión de una imagen en modo óptimo para gráficos vectoriales



C.4 Impresión de una imagen en modo óptimo para mapas de bits



C.5 Impresión de una imagen y texto en modo óptimo para gráficos vectoriales



C.6 Impresión de una imagen y texto en modo óptimo para mapas de bits



John

