



**BENEMÉRITA UNIVERSIDAD
AUTÓNOMA DE PUEBLA**

**FACULTAD DE CIENCIAS DE LA
COMPUTACIÓN**

**“SISTEMA MULTIMEDIA PARA
LA PROGRAMACION EN
LENGUAJE C”**

**TESIS PARA OBTENER EL
TITULO DE:**

**LICENCIADA EN CIENCIAS
DE LA COMPUTACIÓN**

**PRESENTA:
NOHEMI VAZQUEZ ZUVIETA**

**ASESOR:
M.C. EUGENIA ERICA VERA
CERVANTES.**

H. PUEBLA DE ZARAGOZA, DICIEMBRE DE 2009.



Agradecimientos

A ti Hijo Mío:

Ya que con tus simples y muy sencillas palabras "Hola Mamá Mimi", me cargo siempre de energía para sacar adelante mi vida y la tuya, además de brindarme positividad para lograr mis proyectos actuales y los futuros. Gracias por soportarme y gracias por todo tu cariño hijo mío, Te amo y siempre te amare Jahir Nee.

A mis Padres:

Gracias por todo su Esfuerzo, Apoyo y su Ayuda, ya que sin estos elementos fundamentales, nunca lo hubiera logrado.

A mi Asesora:

Gracias profesora por todo su tiempo y su paciencia para soportarme, gracias por todos los conocimientos adquiridos ya que siempre le estaré agradecido.

A mi familia y amigos:

También les dedico este trabajo a mi familia y a todos mis amigos que de alguna manera directa o indirectamente siempre confiaron en mí, Gracias.



INDICE

INTRODUCCION.....	4
CAPITULO I “Marco Teórico”.....	7
1.1 Ingeniería de Software	7
1.1.1 Factores de Calidad y productividad	7
1.1.2 Modelos de Madurez de Capacidad (CMM)	10
1.2 Software Educativo	14
1.2.1 Tipos de Software Educativo.....	15
1.2.2 Ingeniería de Software Educativo.....	16
1.2.2.1 Metodología para el desarrollo de software educativo.....	16
1.2.2.2 Análisis de necesidades educativas	16
1.2.3 Funciones y Características de software educativo	18
1.2.4 Funciones y Características de software educativo	22
1.3 Multimedia	25
1.3.1 Componentes de Multimedia.....	25
1.3.2 Etapas de desarrollo de un producto multimedia.....	29
1.3.3 ¿Dónde y porque utilizar multimedia?	30
1.4 Lenguaje de Programación	31
1.4.1 Lenguajes de Desarrollo Multimedia	32
1.4.2 Herramientas Multimedia	32
CAPITULO II “Análisis y diseño educativo”	34
2.1 Determinación de estructura del Módulo de Aprendizaje	34
2.1.1 Descripción y características	34
2.1.1.1 Temática:	34
2.1.1.2 Propósito General:	34
2.1.1.3 Nivel de Usuario:.....	35
2.1.2 Objetivos Educativos.....	35
2.1.3 Usuarios destinatarios.....	36
2.2.1 Contenidos de Aprendizaje.....	36
2.2.2 Descripción de Actividades por Subtema y Operaciones Mentales...37	
2.2.2.1 Registro de información	38
2.2.2.1.1 Actividades, medios y fuentes de información.....	38
2.2.2.2 Lista de preguntas y respuestas por subtema.....	39
2.2.3 Cuadro de decisiones de aprendizaje.....	40
2.2.4 Diagrama de flujo de aprendizajes	40
2.3 Estrategias de enseñanza-aprendizaje y actividades iterativas (tipo), Duración y Actividades Mentales:	33
2.4 Motivación:.....	34



CAPITULO III “Análisis y Diseño del Sistema”	42
3.1 Planteamiento del problema	42
3.2 Especificación de Requerimiento	42
3.3 Guión literario	44
3.4 Guión Técnico	46
3.4 Diagramas	54
CAPITULO IV “Desarrollo del Sistema”	56
4.1 Herramientas de Software	56
4.1.1 Sound Forge 4.5.....	56
4.2 Director 8.5	57
4.2.1 Escenario o “Stage”	57
4.2.2 Reparto o “Cast”	58
4.2.3 Canales de sonido	60
4.2.4 Partitura	61
4.2.5 Animación en Director	62
4.2.6 Lenguaje Lingo.....	65
CAPITULO V “Pruebas”	69
5.1 Pruebas	69
5.2 Conclusiones	70
5.3 Perspectivas	70
BIBLIOGRAFIA	72



Introducción

La comunicación es un factor muy importante para los seres vivos, ya que es una necesidad de cada individuo el tener que transmitir ideas, sentimientos y conocimientos.

¿Qué es la Comunicación?

Es la transmisión de información de una persona a otra, o de un elemento a otro.

Para poder darse una comunicación, devén de existir 3 elementos muy importantes en ella, los cuales son: El Emisor, Medio y un Receptor.

La primera comunicación que existió entre los hombres fue a base de signos y gestos que expresaban intuitivamente determinadas manifestaciones con sentido propio.

Los signos eran por medio de algún soporte como el papel, piedras, la madera, etc. Para poder expresar alguna idea, sentimiento, situación, acontecimiento, etc.

Posteriormente se dio la comunicación hablada la cual esta constituida por las Letras, Palabras y/o Frases.

Consecutivamente también se le conoce como medio de comunicación a las Señales de humo y los Destellos con espejos.

A continuación se da la Comunicación por cable la cual se interpreta por El telégrafo que permite comunicaciones a través de cables y El teléfono que permite comunicación por voz a larga distancia.

Inmediatamente surge la Radio y Televisión, las cuales evolucionan por mucho la comunicación entre las personas.

Actualmente se da ahora la comunicaron entre las Computadoras y Telecomunicaciones, en donde en la década de los sesenta comienza a surgir la idea de incorporar las computadoras en las comunicaciones de datos a cierta distancia.



Las necesidades de ahora en la actualidad necesitan que se desarrollen nuevas herramientas para un mayor rendimiento y eficacia en el flujo de la información para que esta misma sea además consultada y mantenida constantemente actualizada conforme valla avanzando el tiempo para que siempre de un mejor rendimiento y eficacia al momento de manipularla.

Todo esto se toma en cuenta porque se pretende desarrollar una alternativa para una institución como esta, a la cual se le desarrollara un sistema a Analizar e Implementar. Este proyecto surgió por la necesidad de automatizar el proceso de inscripciones a partir de los pobres recursos que se tenían para este proceso. De tal manera que el objetivo principal de este sistema es el de automatizar el proceso que se sigue para controlar el flujo de la información que se sigue a través de este proceso a la vez tan corto y a la vez tan largo como lo es el de las inscripciones a cursos nuevos y especialidades en esta institución tan prestigiada.

La información que se almacenara en la base de datos tendrá como la descripción, de todos los alumnos que se inscribirán al instituto al tomar algunos de sus cursos que se iniciaran en algún trimestre por empezar estando estos listos desde que termina un trimestre.

También se encontrara la información de todos los cursos que se imparten en el Instituto, ya sea a alumnos externos por llamarlos de alguna manera o también a los alumnos que toman cierto curso(s), únicamente y no de alguna especialidad como lo mencionaremos en los próximos temas. También se encuentra la información de los salones, grupos, horarios y una gran cantidad de información como la de los maestros que se encuentran impartiendo clases en el instituto, y esta información puede ser de tipo personal, así como de su último grado de estudios y su actual preparación académica que se almacena también en la base de datos para un mejor sueldo dependiendo precisamente de esta preparación.



Toda esta gran cantidad de información anteriormente era manipulada manualmente y algunas operaciones como la de las consultas tenían que llevarse a cabo en mucho papel, con lo cual se perdía mucho tiempo, papel, esfuerzo y resultaba muy poco práctico, el a veces realizarlo en Word. También es importante mencionar que cuando se requería hacer una consulta y que ya mencionamos que éstas están hechas en Word, se recurría inmediatamente a los archivos almacenados en alguna carpeta de las muchas que se iban generando con el transcurso del tiempo, ya sea por año, por trimestre, por mes y por ramas (computación, inglés, francés, matemáticas, electrónica, etc.). y cuando se consultaba la información a la vez esta tenía que estar actualizando constante y manualmente por algún cambio que surgiera por parte obviamente de los alumnos. Y bueno, lo que trato de decir es que la información que se manejaba de esta manera era muy pesada trabajarla de esta manera, y es por eso que este Sistema trata de mejorar todo esto y sin necesidad de buscar en archivos por archivos, ahora tendremos la posibilidad de que solamente eligiendo de qué manera queremos ver la información, al momento de ser procesada, ya sea por alguna fecha en especial o por alguno de los que controlaran la información (recepción, profesores y/o directivos).



CAPITULO I “Marco Teórico”

1.1 Ingeniería de Software

Con la aparición en la década de los 60 de nuevos sistemas multiusuarios surgió un problema que se ha ido acrecentando a medida que han surgido nuevos sistemas cada vez más sofisticados, este problema es la transportabilidad de las antiguas aplicaciones a los nuevos sistemas y el mantenimiento de las mismas. Junto a este problema hay que decir, que es cada vez mayor la demanda de aplicaciones ya existentes con la incorporación de nuevas facilidades.

Como consecuencia, desde comienzos de la década de los 60 se hizo patente la necesidad de amalgamar y agrupar los muy distintos puntos de vistas existentes a la hora de enfrentarse con la realización de proyectos de envergadura. En un intento, de conseguir una relativa unificación de criterios, comenzó a surgir y a desarrollarse la disciplina a la que se dedica esta asignatura, “la Ingeniería de Software”, y que puede definirse como:

“La disciplina tecnológica que trata de la producción y mantenimiento sistemático de productos software que se desarrollan y mantiene a lo largo del tiempo y con un coste estimado”

1.1.1 Factores de Calidad y Productividad

Existen una serie de factores que influyen para lograr la excelencia en la calidad y productividad, ayudando a realizar dicha medida; los cuales son:

La capacidad individual.- En ella interviene la competencia del individuo y su familiaridad con el área de la aplicación.

La comunicación entre los miembros del equipo.- Es un factor importante, ya que el trabajo en la mayor parte de las ocasiones no es individual y debe integrarse con el que ha sido desarrollado por otros miembros del equipo.

La complejidad del producto.- Esta parte depende del tipo de aplicación a desarrollarse y de que tan difícil estimación, ya que muchas veces hasta la fase de desarrollo no es posible comprender en toda su perspectiva las complicaciones que conlleva su realización.

Utilización de una nota adecuada.- Es de gran importancia para facilitar la comunicación entre las partes involucradas (incluido el usuario).



Empleo de métodos sistemáticos.- Es importante que se empleen técnicas que sean bien conocidas por los integrantes del equipo de desarrollo de la aplicación; también es fundamental que estas técnicas se empleen de manera sistemática sobre todas las aplicaciones de características semejantes con objeto de facilitar el análisis de costo y tiempo, así como para poder observar la trayectoria profesional de los miembros del equipo.

Conocer el tiempo disponible.- Este factor está vinculado a otros anteriores, ya que es básico conocer el tiempo que puede aportar cada miembro del equipo y en que plazo, sobre todo en función de las tareas a realizar y de la mejor o peor productividad de determinados miembros en cada una de ellas.

Existencia de facilidades y recursos externos.- Es una parte determinante, en la medida en que se conozcan productos o herramientas (automáticas o no) que faciliten las labores de desarrollo e integración de la aplicación.

En mayor medida cuando se conocen aplicaciones parecidas de fácil transporte y modificación que puedan servir de base a la que hay que realizar. En el proceso de desarrollo de aplicaciones educativas, al igual que en los modelos, métodos y técnicas que provee la Ingeniería de Software para poner en práctica y lograr el mejoramiento de la calidad del proyecto y una mayor satisfacción de los usuarios. En particular trataremos las guías de la norma ISO.

El modelo de referencia del proceso de producción en un desarrollo de sistema de información, denominado normalmente "ciclo de vida", identifica y describe cada una de las fases en que se puede descomponer el proceso de producción del proyecto, la naturaleza y el contenido de los trabajos a realizar, así como los distintos productos y componentes a obtener y entregar como resultado del software elaborado. El establecimiento de modelos de referencia forma parte de las metodologías de desarrollo.

Los modelos más representativos de ciclo de vida, podrían ser:

-Modelo secuencial (o "en cascada"). Es el modelo más sencillo y corresponde a una ordenación del ciclo de vida del proyecto en fases ordenadas de modo secuencial. En este modelo es necesario finalizar una fase para empezar la siguiente por lo que es adecuado para muchos tipos de proyectos. La división del proceso de producción en fase se puede realizar con distintos niveles de detalle en función de las peculiaridades de cada proyecto concreto. Según el tamaño del proyecto (grande, pequeño) se pueden realizar las tareas de las fases o no.

-Modelo de desarrollo modular. Se podría completar como una variante de los modelos secuenciales adaptados al caso particular de sistemas de información que posean una naturaleza manifiestamente modular, es decir, que puedan descomponerse en módulos desarrollables con un notable grado de independencia.



-Modelo de desarrollo por evolución de prototipos. Este modelo parte de la consideración de que el diseño de un sistema puede (y debe) realizarse de forma "incremental", es decir, definiendo y añadiendo al sistema sucesivos niveles de funcionalidad a otros previamente implantados, mediante un proceso de desarrollo cíclico.

-Modelo en espiral. Es el resultado de una aproximación al problema basada en los riesgos y no en la documentación o el código. El modelo en espiral puede incluir la mayor parte de los modelos anteriores como casos particulares y, además, proporciona elementos de juicio para decidir qué combinación de los modelos anteriores se adapta mejor a una situación de elaboración de software dada.

Este modelo fue desarrollado por B. Boehm, básicamente, la idea es Desarrollo Evolutivo, usando el Modelo de Cascada para cada etapa; está orientada a evitar riesgo de trabajo. No define en detalle el sistema completo a la primera.

El Modelo Espiral mejora el Modelo de Cascada enfatizando la naturaleza iterativa del proceso de diseño. Eso introduce un ciclo de prototipo iterativo. En cada iteración, las nuevas expresiones que son obtenidas transformando otras dadas son examinadas para ver si representan progresos hacia el objetivo.

Las fases del Modelo Espiral son:

- 1.- Análisis de Riesgos.
- 2.- Diversos Prototipos.
- 3.- Simulación y Modelos.
- 4.- Conceptos de Operación.
- 5.- Requerimientos de Software. Validación de Requerimientos.
- 6.- Diseño, validación y verificación de Software.
- 7.- Detalles de Diseño.
- 8.- Implementación de código.
- 9.- Diversos test para el Código (unificación, integración, aceptación e implementación).
- 10.- Planes de integración, requerimientos y ciclos de vida.



Estos puntos son sólo algunas sugerencias que se han adoptado por muchas compañías de software para tratar de producir software de calidad. Por la importancia de esta producción existen compañías de calidad que han diseñado estándares para este desarrollo de software, y uno de estos modelos es el “Modelo de Madurez de Capacidades”.

1.1.2 Modelos de Madurez de Capacidad (CMM)

CMM es un modelo que incluye prácticas de planeación, ingeniería, administración de desarrollo y mantenimiento de software. El propósito de CMM es guiar a las organizaciones a darle madurez a sus procesos de producción, consiguiendo así calidad en el producto.

CMM es una estrategia de mejora que señala las deficiencias dentro de las organizaciones, no es una solución rápida, pero sí proporciona una guía para avanzar hacia la calidad del producto que en este caso es software.

La estructura de CMM cuenta con cinco niveles de madurez. Definimos niveles de madurez como: “Plataforma bien definida desde la cual podemos obtener un proceso maduro de software y definimos madurez de un proceso de software como: “el punto hasta el cual un determinado proceso es explícitamente definido, administrado, medido, controlado y efectivo”. Para alcanzar cada uno de estos niveles de madurez, existen metas intermedias llamadas KPAs. Estas KPAs varían en cantidad y forma en cada una de las capas. Cada uno de los niveles de madurez del CMM nos indicará que tan capaz es cierto proceso de software para solucionar un problema, obteniendo mediciones podremos, con mayor precisión, determinar el resultado del próximo proyecto que se presente.

De este modo las cinco capas del CMM se describen a continuación:

- **Nivel Inicial:**
Es el primer nivel (figura 1) indica el punto de partida, las empresas se encuentran en este nivel de calidad de producción de software cuando sus procesos no tienen control y se hacen explícitamente para cada proyecto. Este nivel se caracteriza porque todo lo que se sabe del proceso es que recibe entrada de datos y mediante un proceso prácticamente desconocido, por no tener documentación, administración, medición, etc. Obtener una salida de datos.



Figura 1 Nivel 1 de CMM Inicial

- Nivel Repetible:
Para alcanzar este nivel la organización debe empezar a documentar los procesos y guardar información sobre los mismos. De este modo este nivel lo podemos ver como la figura 2, en la cual encontramos que ahora el proyecto se divide en procesos que aunque no sabemos bien qué hacen, si obtenemos salidas esperadas y podemos generar políticas para el desarrollo del software. Al término de cada proceso podemos tener una revisión donde se podrá detectar si el proceso está funcionando. Resumiendo la capacidad de software de una organización de nivel 2 tiene disciplina para documentar el proyecto de software e involucre planeación y seguimiento.

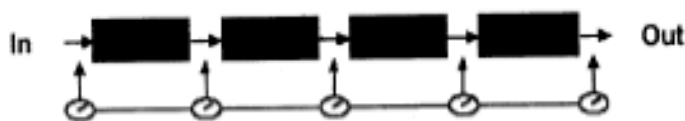


Figura 2 Nivel 2 de CMM Repetible

- Nivel 3 Definido:
En este nivel el proceso de software es un estándar y se puede aplicar a todos los proyectos venideros. Lo que antes eran procesos oscuros se convierten en procesos documentados y la cantidad de accesos para conocer cómo confiabilidad como se muestra en la figura 3.
Algunos de los puntos clave de este nivel son:
- Los procesos se implementan y actualizan para ayudar a los administradores de proyectos y staff técnico a desempeñarse más efectivamente.
- Para estandarizar se ocupan prácticamente de ingeniería de software.
- Un grupo dentro de la organización está encargado de las actividades del proceso de software.



- La organización cuenta con un programa de capacitación para que todos los mineros de la organización cuenten con el conocimiento y las habilidades requeridas para desempeñar completamente sus roles.
- El proceso de software estándar de la organización se amolda a cada proyecto para así determinar el proceso de software definido del proyecto (su propio proceso de software) éste contendrá información acerca de cómo saber que ya está listo el producto, entradas, estándares y procedimientos para desarrollar el trabajo, mecanismo de verificación, salida y un criterio de terminación.

El nivel 3 es estándar y consiste gracias a la ingeniería de software y la administración de proyectos.



Figura 3 Nivel 3 de CMM Definido

- Nivel 4 administración:
En este nivel lo importante es que contamos con medición del progreso y de los problemas. Hacemos uso todos los datos recolectados y los convertimos en información relevante para identificar y corregir problemas. Cuando alcanzamos este nivel, el cliente puede tener un entendimiento medible de la capacidad y complejidad de cada proceso, aun antes de que empiece el proyecto. En este nivel se tiene una retroalimentación del proyecto como se muestra en la figura 4. Este nivel es cuantificable y predecible gracias a que el proceso, el producto y los servicios son medidos y trabajan dentro de un límite.

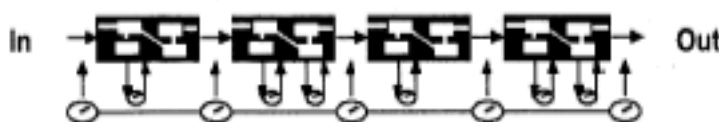


Figura 4 Nivel 4 de CMM Administrado



- Nivel 5 Optimizado:

Cuando la organización alcanza este nivel es porque se dedica a mejorar constantemente su proceso. Esto con el fin de encontrar y corregir defectos en el mismo. Si un defecto es encontrado, es revisado o substituido, como se observa en la figura 5, el objetivo de esto es prevenir y evitar la recurrencia de estos defectos.

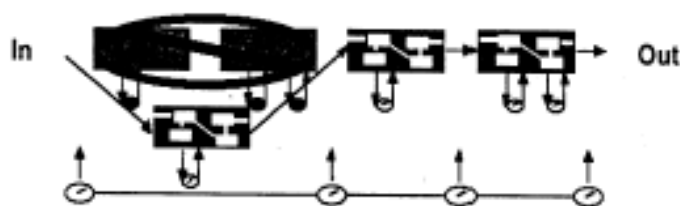


Figura 5 Nivel 5 de CMM Optimizado

Para superar un nivel de madurez y alcanza el siguiente nivel dentro de CMM debemos cumplir con un número determinado de KPAs. El único nivel que requiere KPAs es en nivel Básico. Es entonces a partir del nivel Repetible que se definen conjuntos de KPAs que (figura 6) están organizados en tres categorías:

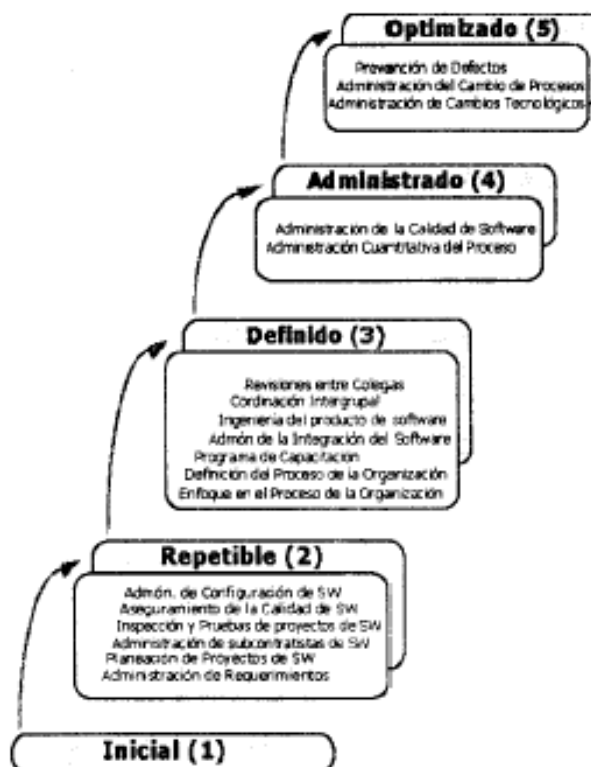


Figura 6- KPAs por Nivel de Madurez



- Gerencia: Planeación del proyecto de software, administración, etc.
- Organización: Capacitación e infraestructura.
- Ingeniería: Análisis de requerimientos, diseño, codificación, pruebas, etc.

Para poder alcanzar estas metas intermedias, CMM nos proporciona un conjunto de prácticas que se describen en cinco grupos según CMM V1.1:

- Compromiso: Es el conjunto de acciones que la organización debe cumplir para asegurar la calidad del proceso.
- Habilidades Necesarias: Describen lo que debe cumplir la organización para poder implementar un proyecto.
- Actividades Realizadas: Nos indica las actividades, roles y procedimientos para poder llevar a cabo la implementación de alguna KPA.
- Medición y Análisis: Incluye ejemplos de matices para lograr las mediciones que son tan necesarias para control de procesos.
- Verificación e implementación: Abarca revisiones y auditorias para asegurar la calidad del software.

1.2 Software Educativo

Con frecuencia no se hace diferencia entre los términos como software educativo, programa educativo y programa didácticos, de este modo son usados como sinónimos para designar genéricamente los programas para computadora creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje.

Según Pérez Marqués, de la Universidad de Barcelona, en los términos anteriores se engloban todos los programas que han estado elaborados con fin didáctico, desde los tradicionales programas basados en los modelos conductistas de los programas experimentales de Enseñanza Inteligente Asistida por Ordenador (EIAO), que, utilizando técnicas propias del campo de los Sistemas Expertos y de la Inteligencia Artificial en general, pretenden imitar la labor tutorial personalizada que realizan los profesores y presentan modelos de representación del conocimiento en consonancia con los procesos cognitivos que desarrollan los alumnos.



No obstante según esta definición, más basada en un criterio de finalidad que de funcionalidad, se excluyen del software educativo todos los programas de uso general en el mundo empresarial que también se utilizan en los centros educativos con funciones didácticas o instrumentales como por ejemplo; procesadores de textos, gestores de bases de datos, hojas de cálculo, editores, gráficos... Estos programas, aunque puedan desarrollar una función didáctica, no han estado elaborados específicamente con esta finalidad.

1.2.1 Tipos de Software Educativo

Uno de los criterios, por los cuales se clasifican los software's educativos, es el grado de control que el programa tiene sobre las actividades de los alumnos y la estructura de su algoritmo. De este modo distinguimos la siguiente clasificación:

- Programa Tutoriales:
- "Pretenden que, a partir de unas informaciones y mediante la realización de ciertas actividades previstas de antemano, los estudiantes pongan en juego determinadas capacidades y aprenden o refuercen unos conocimientos."
- Bases de Datos:
- "Proporcionan unos datos organizados, en un entorno estático, según determinadas criterios, y facilitan su exploración y consulta selectiva".
- Simuladores:
- "Presentan un modelo o entorno dinámico (generalmente a través de gráficos o animaciones interactivas) y facilitan su exploración y modificación a los alumnos que pueden realizar aprendizajes inductivos o deductivos mediante la observación y la manipulación de la estructura subyacente"
- Constructores:
- "Son programas que tienen un entorno programable. Facilitan a los usuarios unos elementos simples con los cuales pueden construir elementos más complejos o entornos".
- Programa herramienta:
- "Son programas que proporcionan un entorno instrumental con el cual se facilita la realización de ciertos trabajos generales de tratamiento de la información: escribir, organizar, calcular, dibujar, transmitir, captar datos etc."



1.2.2 Ingeniería de Software Educativa

La enseñanza asistida por computadora se ha convertido en una rama de investigación importante de la inteligencia artificial. Francisco Ruedas (Ruedas, 92) menciona algunas técnicas, empleadas para desarrollar software educativo, tales como representación del conocimiento, sistema experto, redes neuronales y procesamiento de lenguaje natural.

1.2.2.1 Metodología para el desarrollo de software educativo

En cuanto a metodología de desarrollo, varios autores han tratado el tema, por ejemplo Alvaro Galvís. Del cual: "Ingeniero de Software Educativo", es una referencia bastante compleja y es una buena guía para el desarrollo del software. En esencia se conserva los grandes pasos o etapas de un proceso sistemático para desarrollo de materiales (análisis, diseño, desarrollo, prueba y ajuste, implementación). Sin embargo, en este caso se da particular énfasis a los siguientes aspectos: la solidez del análisis, como punto de partida: el dominio de teorías sustantivas sobre el aprendizaje y la comunicación humana, como fundamento para el diseño de los ambientes educativos computarizados; la evaluación permanente y bajo criterios predefinidos, a lo largo de todas las etapas del proceso, como medio de perfeccionamiento continuo del material; como base para el mantenimiento que requerirá el material a lo largo de su vida útil.

1.2.2.2 Análisis de necesidades educativas

Todo MEC (Material educativo computarizados) debe cumplir un papel relevante en el contexto donde se utilice. Su incorporación a un proceso de Enseñanza/Aprendizaje no se puede deber simplemente a que el MEC sea del agrado de una sola persona o un grupo de ellas. Deben existir razones mucho más importantes y sobre todo estudiadas para que un MEC se utilice en alguna área.

A diferencia de las metodologías asistemáticas, donde se parte de ver de qué soluciones disponemos para luego establecer para qué sirve, de lo que se trata acá es de favorecer en primera instancia el análisis de qué problema o situaciones problemáticas existen, sus causas y posibles soluciones, para entonces si determinar cuáles de estas últimas son aplicables y pueden generar los mejores resultados.



Esta metodología entonces pretende responder a las preguntas: ¿cómo identificar las necesidades o los problemas existentes?, ¿qué criterios usar para llegar a decidir si amerita una solución computarizada? ¿con base en qué, decidir si se necesita un MEC y qué tipo de MEC conviene que sea, para satisfacer una necesidad dada?

Es por lo anterior que esta metodología se centra en:

- Consulta a fuente de información apropiadas e identificación de problema.
- Análisis de posibilidades causas de los problemas detectados.
- Análisis de alternativas de solución.
- Establecimiento del papel del computador en la solución del problema.



1.2.3 Modelos de desarrollo de software educativo multimedia

El método propuesto para realizar software multimedia está basado en un modelo de ciclo de vida evolutivo que parte de una caracterización de la situación del problema y pasa sucesivamente por seis fases como se muestra en la figura 7. Con base a las evoluciones resultantes del prototipo desarrollado, se pueden ir realizando ajustes sucesivos hasta obtener el producto deseado. Las fases se describen a continuación:

- Fase 1: Descripción del Problema.
En esta fase se inicia con la descripción detallada del problema, a partir de la idea que tiene el cliente del producto que quiere obtener. Se hace también un perfil de los usuarios potenciales en términos de sus niveles educativos, cultural, socioeconómico, edad y el ambiente en el que se usara el software. Toda esta información se debe verificar con el cliente y plasmarse en un Documento de definición de proyecto como se muestra en la figura 8.



Figura 7 Fases del Método de Desarrollo Multimedia



- Fase 2: Definición de requerimiento y Términos de desarrollo
Con base al documento que se realiza al final de la fase anterior, se definen los requerimientos funcionales y los no funcionales en conjunto con el cliente y se establecen los criterios de aceptación del proyecto. Con esta información redactamos el Documento de Requerimientos. Se deben hacer estimaciones de tiempo y costo para las diferentes fases del proyecto y con esto generamos un documento Plan de Desarrollo del Producto. Una parte importante de índole legal es la redacción del contrato y éste se debe basar en los tres documentos anteriores (figura 9).
- Fase 3: Diseño educativo.
En esta fase es de vital importancia la colaboración multidisciplinaria entre analistas, pedagogos, sociólogos, etc. Después de este análisis se establecen los temas que se tratarán y la forma de abordarlos. Para esta parte se sugiere hacer el análisis objetivo y subjetivo, la estructura de los temas y sub-temas, las actividades que realiza el usuario y los mecanismos de evaluación (figura 10).
- Fase 4: Diseño Computacional
En esta fase se establece la estructura general de los módulos y sub-módulos que componen el software. Con ayuda de los diseñadores gráficos se establecen los formatos que se utilizarán. Para cada actividad descrita se hace el diseño de las unidades de información a presentar y también se realizan los enlaces con otras unidades. El producto final de esta fase es el Documento de Diseño Computacional (figura 11).



Figura 8 Descripción del Problema

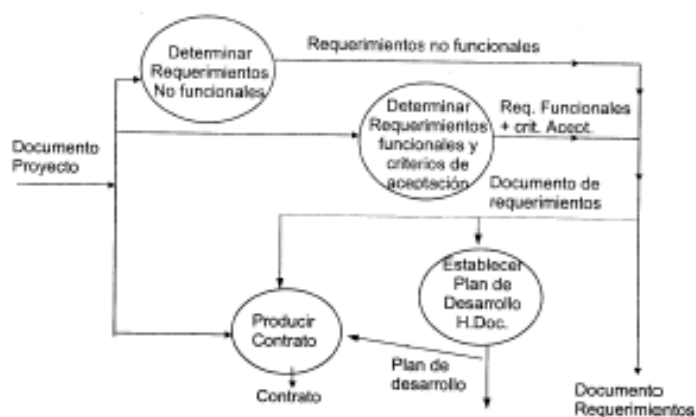


Figura 9 Definición de Requerimientos

- Fase 5: Producción.

En esta fase se producen los ítem de información, es decir, se hacen la digitalización, edición y almacenamiento del sonido, imágenes, video, etc. De este modo se obtiene en la primera iteración un producto en prototipo que se ira mejorando con la reiteración de estas fases, hasta obtener un software de buena calidad (figura 12).

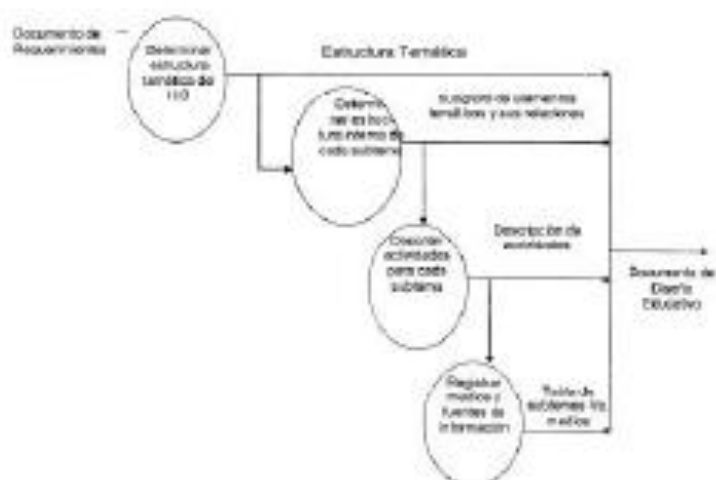


Figura 10 Diseño Educativo



Figura 11 Diseño Computacional

- Fase 6: Evaluación del producto.
El producto es evaluado por grupos diseñadores, usuarios finales, psicólogos, etc. Para verificar si cumple con todos los requisitos planteados en la fase 2, se elabora una lista de modificaciones que se deben aplicar realizando nuevamente las 5 fases anteriores hasta obtener un producto final de calidad (figura 13).



Figura 12 Fase de Producción

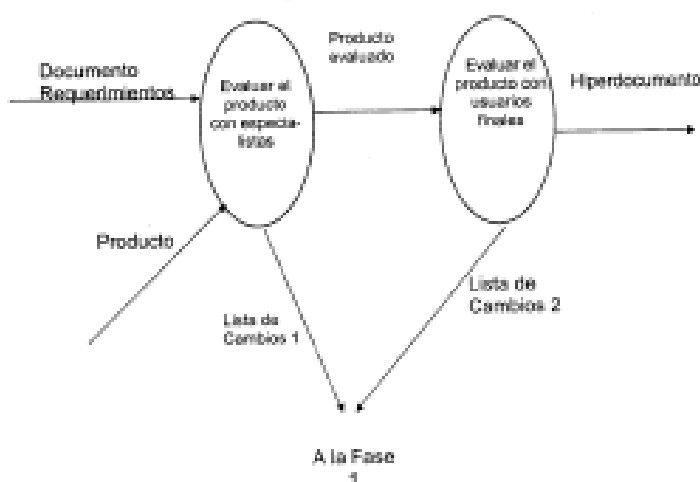


Figura 13 Evaluación del Producto

1.2.4 Funciones y Característica de software educativo.

Aunque existen varias maneras de hacer software educativo, para que éste sea considerado como tal, se deben cumplir ciertas características como son:

- El material presentado debe ser elaborado con fines didácticos.
- Evidentemente debe ser utilizando una computadora.
- Deben de ser interactivos para captar la atención del usuario y de este modo agilizar el aprendizaje y el aprendizaje del usuario.
- No debe ser necesario tener estudios avanzados de computación para poder hacer uso y obtención de los beneficios de estos software's.
-



Así pues, podemos diferenciar tres partes básicas en este tipo de programa. La parte del software, a través de la cual el programador de éste puede establecer un diálogo con el usuario y de este modo lograr la interactividad, se conoce como interfase. Esta parte del software, a su vez, tiene dos partes que se conocen como:

- Interfase:
 - Programa-Usuario: Facilita la transmisión de la información desde la computadora hasta el usuario. Esto se hace con pantalla de información, impresiones, voz, videos, etc.
 - Usuario-Programa: Facilita la retroalimentación desde el usuario hasta la computadora, a través del teclado, ratón, micrófonos, lectoras y cualquier otro periférico de entrada que existe para las computadoras.

Otra parte importante del software es la información y cómo está almacenada. Normalmente esta información que consta de imágenes, texto, videos, sonidos, etc. Está almacenada en una base de datos para que su recuperación sea rápida.

- Base de Datos:
 - Datos de tipo texto: Información del tipo alfanumérica.
 - Datos Gráficos: Dibujos, fotografías, secuencias de videos, animaciones, etc.
 - Sonidos: Música, fragmentos de partituras, MIDI, etc.
 - Modelos de Comportamiento: Representan la dinámica de algunos sistemas, existen los físicos-matemáticos con leyes perfectamente definidos por grafos y tablas de comportamiento.

La tercera parte se refiere a cómo, cuándo y de qué forma se le presenta la información al usuario. A esta parte se le llama Motor Algoritmo:

- Motor o Algoritmo:

Esta parte del software es la que gestiona las acciones que el usuario realiza y está preparado para responder a éstas, comunicándose con la base de datos y la interfase. Se distinguen cuatro tipos de algoritmos:

 - Lineal: Cuando la secuencia de las actividades no varía y es una sola.
 - Ramificado: Cuando puede existir más de un camino según sea la entrada que proporcione el usuario.



- Tipo entorno: El usuario puede elegir qué hacer y cuándo hacerlo.

Este entorno puede ser del tipo:

- Estático: El usuario sólo puede consultar la información sin modificar la estructura.
 - Dinámico: Si el usuario puede modificar el estado de los elementos que conforman el entorno.
 - Programable: Este se da cuando el usuario es capaz de construir los entornos.
 - Instrumental: Cuando se tiene un motor de inferencias.
- Sistema Experto: Cuando se tiene un motor de inferencias.

Las diferentes funciones que se le puede dar a un software educativo, tiene mucho que ver con la calidad del material que presentan, la claridad del mismo y la utilidad que le dé el profesor. Podemos categorizarlos en:

- Función informativa:
Los programas que entran en esta característica son los tutoriales, simuladores y las bases de datos.
- Función Instructiva:
Orientan y regulan el aprendizaje de los usuarios y condicionan el tipo de aprendizaje que se realiza.
- Función Motivadora:
Tratan de atraer la atención de los usuarios, focalizar los aspectos de mayor interés.
- Función Evaluadora:
Las funciones de evaluación pueden ser de tipo implícita, cuando el estudiante encuentra de forma explícita, cuando es el programa el que presenta informes sobre el desempeño del usuario.
- Función Investigadora:
Esta función la realizan los software's a los cuales se les modifican valores y la salida cambia, como son los simuladores y los programas constructores.
- Función Expresiva:
Los que realizan estas funciones son los que nos permiten crear formas de expresión, como los lenguajes de programación, editores de texto.

Los software's educativos pueden caer dentro de varias de estas funciones al mismo tiempo, es decir, estas funciones no son excluyentes.



1.3 Multimedia

En una computadora existen muchas formas de presentar información para la persona que la está utilizando. De este modo el usuario del equipo puede recibir información de este último en forma de texto, es decir, en el monitor se le presenta un escrito tal y como si estuviera leyendo un libro. Se puede también, en una computadora, escuchar música, relatos, efectos de sonido, etc. Por supuesto, puedo tener un álbum fotográfico, caricaturas, dibujos y demás formas de representación gráfica. Entre otras cosas más, se puede obtener videos a través de un equipo de cómputo, de este modo las computadoras nos permiten presentar información de diferentes formas y medios. Al intercalar todos estos, y otros, componentes en una misma presentación, programa o software, obtenemos el concepto de "multimedia".

De este concepto se desprenden otros más, que dependen de la forma en que el usuario del software se relaciona o interactúa con éste. Así obtenemos el concepto de "multimedia lineal" y "multimedia interactiva". Se considera un "software multimedia lineal" cuando el usuario se sienta frente a la computadora y simplemente ve pasar todo el material desde el principio hasta el final y no interviene para nada en el transcurso del software. Cuando el software permite que el usuario controle algunos elementos y el momento en que éstos deben presentarse, entonces hablamos de "software multimedia interactivo". Si además de ser interactivo se proporciona una estructura de elementos ligados, a través de los cuales el usuario puede navegar, esto es, el usuario puede brincar de un concepto a otro, de un tema a otro, de una imagen a otra, etc. Entonces tenemos un "software hipermedia".

1.3.1 Componentes de Multimedia

Como ya vimos, las capacidades multimedia se conforman de componentes más pequeños que por sí solos son capaces de transmitir información desde la computadora para el usuario de ésta. Estos componentes son:

- **Texto:**
El texto en las computadoras nos permiten, no sólo la expresión semántica de las oraciones que escribimos, si no que también nos permite utilizar diferentes tipos de letras con los cuales podremos enfatizar la idea que queremos comunicar. Los tipos de letras son familias de caracteres gráficos que incluyen varios tamaños y estilos de letras.



Una fuente es una colección de caracteres con un solo tamaño y un solo estilo. “Los tipos de letras más comunes son negritas e itálica (cursiva o bastardillas). Otros atributos de tipo de letras, como el subrayado y el delinado de caracteres, pueden agregarse con su programa de computadora. Los tamaños de letras se expresan casi siempre en puntos: un punto es .0138 pulgadas o cerca de 1/72 de pulgadas. El tamaño es la distancia desde la parte de arriba de las letras mayúsculas hasta la parte de debajo de las letras minúsculas como la g y la y. Helvética, Times y Courier son tipos de letras. Times de 12 puntos itálica es una fuente”.

- **Sonido:**
El sonido es un fenómeno físico que puede representar mediante una curva la cual tiene ciertas características como son amplitud y frecuencia, a esta curva se le llama oscilación. A la cantidad de oscilaciones por cada unidad de tiempo se le conoce como frecuencia. La unidad de frecuencia es el hercio (hz). El sonido en el aspecto digital tiene dos características importantes que hacen que el sonido dentro de una computadora sea de mejor calidad. Estas características son la velocidad de muestreo es la frecuencia con la que se toman las muestras y el tamaño de la muestra es la calidad de información tomada por cada muestra. Normalmente en una computadora se manejan muestras de frecuencia de 44.1 Khz. 22.0 Khz. Y 11.025 Khz. Y tamaños de muestras de 8 o 16 bits. Existe también un estándar llamado MIDI (Music Instrumenta Digital Interfase). En el caso de los archivos MIDI, no se contiene el dato como tal, más bien el archivo le comunica al sintetizador el tomo, volumen, timbre, dinámica de impulso y el instrumento que deberá tocar la nota.
- **Imágenes:**
Las imágenes son representaciones gráficas, y en el caso de las computadoras es muy importante distinguir entre los gráficos de mapa de bits y los gráficos vectoriales. En los gráficos de mapa de bits, la información se guarda en forma de matriz de píxeles, que contienen la información sobre el color y la posición de cada punto del gráfico. Y en los gráficos de vectores, la información está formada por círculos, cuadrados, líneas y demás figuras geométricas. A diferencia de muchas otras áreas técnicas, como los sistemas de video y los disquetes, los formatos y el usuario se encuentra ente el viejo y conocido problema de la importancia y la exportación.



Esto puede resultar irritante, pero cuando se profundiza en la edición o el almacenamiento de imágenes, se podrá comprobar que existen tantos campos diferentes de aplicación que probablemente no será inteligente imitarse a un solo formato de imagen.

Los formatos de imágenes más comunes son PCX, TIFF, EPS, GIF, TGA y BMP. Estos formatos se diferencian dentro si por la forma de almacenamiento, el espacio que ocupa y la forma de tratar los colores.

- Gráficos de vectores o de puntos: Según la forma en que se guardan las imágenes, se diferencia entre dos tipos de formato. **Importantes:** En los formatos de vectores no se guardan puntos individuales, sino objetos. Mientras que, por ejemplo, una circunferencia se compone de muchos puntos, en un gráfico de vectores sólo se guardan las informaciones sobre la forma (circunferencia). Localización, tamaño y color. No puede establecer una ventaja clara para uno de los tipos de formatos. Un gráfico de vectores tiene la ventaja de que su tamaño puede modificarse libremente, sin que ello implique pérdidas de calidad, mientras que un gráfico de puntos presenta irregularidades en sus contornos, cuando se agrada demasiado. Por otra parte, los gráficos de puntos tienen la nada despreciable ventaja de que con filtros para crear efectos borrosos u otro tipo de filtros, es posible eliminar puntos de la imagen, para lograr modificaciones. Los gráficos de vectores no resultan, por ello, nada adecuados para la edición de Imágenes. Por motivo, a continuación nos referimos sólo a los principales formatos gráficos de puntos, con la excepción del formato EPS, que ocupa un lugar especial entre los formatos gráficos.
- El formato PCX: Es el más ampliamente difundido entre los gráficos de puntos es el PCX, el cual fue desarrollado por la firma ZSoft para su programa de dibujo Paintbrush. La ventaja de que este formato PCX pueda ser leído por casi todos los programas (en especial de Windows) resulta muy relativa para los profesionales de la edición de Imágenes, dado que en el formato PCX no son posibles representaciones TrueColor con resolución por encima de 1024x768 puntos. Lo máximo posible es una representación de 8 bits. A pesar de eso, el formato PCX está cerca de ser el estándar en los ordenadores personales.



- El formato TIFF: Es el formato gráfico más versátil (Tagges Image File Format). Lamentablemente, esta versatilidad también tiene el inconveniente de que existen muchos tipos de formatos TIFF. Esto puede significar que una imagen guardada en un formato TIFF no puede ser leída por otro programa que soporta otro tipo de TIFF. Como una imagen DIN A4 digitalizada a 300 ppp en formato TIFF, con una densidad de color de 24 bits (TrueColor), ya ocupa 42 MB de espacio, se han desarrollado diferentes métodos de compresión para este formato.

El más conocido es el método LZW de la firma Aldus (fabricante de photoStylerPagemaker). Los archivos comprimidos de esa forma pueden ser leídos por muchos programas, también de Macintosh.

- El formato GIF: Este formato puede utilizarse, incluso con más amplitud que el formato TIFE, en diferentes sistemas de ordenadores. Es una lástima que este formato sólo pueda representar 256 colores (8 bits). El formato CIF fue desarrollado por la firma de CompuServe.
- El formato TGA: Es un formato para guardar y convertir imágenes en TrueColor tanto en el PC como en el Macintosh. No está demasiado difundido en España, pero en América ocupa una posición importante.
- El Formato EPS: Puede ser leído por casi todos los procesadores de texto, pero generalmente no puede ser representado en la pantalla. En formato EPS se guardan imágenes que pueden ser reproducidas desde el proceso de texto en una impresora láser de PostScript. El formato EPS se utiliza con frecuencia como formato intermedio entre diferentes formatos de vectores (las impresoras PostScript trabajan en una forma muy similar a los vectores) y también es muy común en la autoedición o en la edición de imágenes. Para que pueda ser visualizado en pantalla, generalmente va acompañado de una imagen TIFF de menor resolución.



- El formato BMP: Es un formato de mapa de bits que guarda las imágenes según un diseño de bits. El principal campo de aplicación del formato BMP está en Windows. Con el tiempo se han desarrollado muchos formatos estándares BMP existe el también casi idéntico formato DIB o RDIB, el cual es utilizado por la extensión multimedia de Windows.
- Animación:
“La animación es posible debido a un fenómeno biológico conocido como persistencia de la visión. Un objeto que ve el ojo humano permanece mapeado en la retina por un breve tiempo. Esto hace posible que una serie de imágenes que cambian rápidamente, una tras otra, parezcan mezclarse creando la ilusión de movimiento. En otras palabras, si usted cambia el lugar o la forma de objetos lo suficiente rápido percibirá los cambios como movimientos”.
- Video:
El video se puede ver como “fotografías en movimientos”, a las cuales se les puede agregar sonido. En realidad lo que nosotros percibimos como video no es otra cosa que la construcción de 30 cuadros por segundo, en el caso de la televisión y casi 48 cuadros por segundo en el cine. Existen tres estándares para producir video que son NTSC, PAL y SECAM. Y para formatos de video digital, el más famoso y usado es el formato JPEG.
- Hipertexto:
Cuando tenemos una gran cantidad de información escrita en las computadoras, podemos indéxala vincular todos los elementos para provocar que la consulta de la información sea del tipo de organización de la información tenemos hipertexto.

1.3.2 Etapas de desarrollo de un producto multimedia.

Para desarrollar un producto multimedia se deben considerar varios factores, algunos de ellos tienen que ver con el aspecto económico y financiero. Para llevar a buen término un proyecto de multimedia se deben alcanzar y superar tres fases.



- Fase de orientación:
En esta fase se establecen los objetivos que se desean alcanzar con el producto. Se puede empezar por escribir en un papel lo que se espera del producto. ¿Qué objetivos persigues?, ¿Qué se aportará al comprador del producto?, ¿Sólo se va a entretener al comprador o se le brindará una enseñanza? Una vez que hayamos contestado estas y otras preguntas podemos entonces dedicar esfuerzos a la realización del producto.
- Fase de producción:
En esta fase se trata de sacar el mayor rendimiento posible al trabajo, por lo general los conceptos y la calidad del producto están relacionados con el tiempo disponible y con el marco financiero. Es básico generar un guión en el cual se encontrarán todos los pasos a seguir para la elaboración del producto. Se deben seleccionar adecuadamente las herramientas que se utilizará para que el proyecto se cumpla en tiempo y en dinero. Es necesario recopilar y organizar todo el material que se usará durante la ejecución del software.
- Fase de distribución:
Esta fase es importantísima para que el proyecto llegue a ser un éxito, es necesario analizar los medios y métodos de distribución que podremos usar, ya que existe gran variedad de forma en las cuales podemos presentar un proceso multimedia. Aquí hay que volver a tomar en cuenta el aspecto financiero del proyecto.

1.3.3 ¿Dónde y por qué utilizar multimedia?

En general, se ha comprobado que la multimedia hace que los que la usan tengan un mayor porcentaje de retención de la información que se presenta que en material de otras características también presentados por computadora.



Ahora la multimedia se encuentra en todos los ámbitos, en las oficinas a diario las personas realizan presentaciones con texto, audio y video para ayudar a que jefes puedan tomar las mejores decisiones de negocios. En nuestra vida cotidiana, nos enfrentamos a “Kiosco multimedia” en restaurantes, museos, tiendas, etc. Que nos ayudan a encontrar lo que buscamos. Pero sin duda en el área en donde más se pueden aprovechar las capacidades de la multimedia es el área educativa. Existen ya varios programas multimedia dirigido a niños desde los 3 años, hasta llegar a enciclopedias para todo público y programas especializados para áreas de licenciaturas.

Las opciones de los programas multimedia en la educación son infinitas, ya que con estos software´s se puede mostrar gran cantidad de información, pasado desde animación Infantil sugerente para aprendizaje de lectura, escritura, matemáticas, etc. ; hasta llegar a reproducir videos complejos sobre cualquier clase de Información, como podría ser: operaciones médicas, viajes espaciales, vida natural, etc. Además la multimedia no puede dar la facilidad de la simulación, que nos ayuda a proporcionar conocimientos del tipo de manejo de maquinaria peligrosa, cara o difícil de conseguir.

Por todo lo anterior, sólo podemos concluir que los programas multimedia deben y pueden ser usados para casi todas las actividades que se pueden realizar con una computadora.

1.4 Lenguaje de Programación.

Como se sabe, una computadora puede realizar operaciones simples a gran velocidad, pero para que una computadora pueda realizar dichas tareas, se necesita poder indicarle el algoritmo a seguir.

Un ALGORITMO es una secuencia ordenada, finita e Inequívoca de pasos a seguir para resolver un determinado problema. Es importante el hecho de que sea una secuencia ordenada porque cuando queremos resolver un determinado problema, tenemos que efectuar los pasos en un cierto orden a no ser que queramos obtener un resultado totalmente diferente al esperado, Por ejemplo, en una receta, que es un algoritmo para preparar un determinado platillo, se si altera el orden de los pasos, nunca obtendremos el platillo deseado. Por otro lado, también es importante recalcar que es una secuencia finita de pasos. Un algoritmo debe terminar en algún momento determinado, ya que de nada nos servirá un algoritmo que nos resolverá el problema dentro de mil años.



Cuando es una computadora la encargada de ejecutar un determinado algoritmo, dicho algoritmo deberá ser expresado en forma de un PROGRAMA de computadora,, el cual consiste de un conjunto de instrucciones que la computadora pueda entender y posteriormente ejecutar. Para esto uno debe usar un LENGUAJE DE PROGRAMA para escribir un programa. A la actividad de expresar un algoritmo en forma de programa se le denomina PROGRAMACION.

1.4.1 Lenguaje de Desarrollo Multimedia

Los Lenguajes de autor poseen multitud de herramientas visuales que permite crear objetos de un modo sencillo y cómodo. Son herramientas cuya característica más representativa se encuentra en su símbolo de potencias y facilidad de manejo, algo que permite que cualquier usuario, sin ningún conocimiento previo de las herramientas de desarrollo, pueda llegar a crear aplicaciones sin estar obligados a escribir ni una línea de código.

1.4.2 Herramientas Multimedia

Para poder realizar programas multimedia, se utilizan varios software's especializados para generar, modificar y mejorar cada uno de os componentes que forman la multimedia. De este modo, es necesario utilizar por lo menos un software especializado en sonido digital, otro software para manejo de imágenes, uno ovarios más para manejo de texto, animación y video.

Aunque todas estas herramientas son sumamente importantes, la herramienta realmente indispensable para generar software multimedia, son los llamados "sistemas de autor". Por lo general estas herramientas no nos sirven para modificar ninguno de los componentes de multimedia, pero si no permiten manipular su aparición y duración durante la vida de nuestro software multimedia.

"Se organizan las diferentes herramientas (o sistemas) de desarrollo en grupo, basándose en la presentación que utilizan para dar una secuencia y organizar los elementos y eventos multimedia".



- Herramientas basadas en tarjetas o paginas:
En estos sistemas de desarrollo los elementos se organizan como páginas de un libro o como una pila de tarjetas. Se puede disponer de miles de páginas o tarjetas en un libro o pila. Estas herramientas son adecuadas cuando gran parte del contenido consiste en elementos que pueden verse individualmente, como las páginas de un libro o como las tarjetas de un fichero.
- Herramientas basadas en íconos controlados por eventos:
En estos sistemas de desarrollo los elementos de multimedia y las señales de Interacción (eventos) se organizan como objetos en un marco estructural o proceso. Las herramientas basadas en Iconos controladas por eventos simplifican la organización de su proyecto y siempre despliegan diagramas de flujo de actividades junto con vías de bifurcación.
- Herramientas basadas en tiempo y de presentación:
En estos sistemas de desarrollo los elementos y eventos se organizan a lo largo de una línea de tiempo con resolución tan altas como un treintavo de segundo.

Los sistemas de desarrollo de multimedia ofrecen uno o más de los siguientes enfoques:

- Programación visual con señalamientos e íconos.
- Programación con lenguajes de guiones.
- Programación con herramientas tradicionales como Basic o C.
- Herramientas de desarrollo de documentos.

Una de estas herramientas que trabajan con guiones es “Director de Macromedia” y es la elegida para este trabajo.



CAPITULO II - "Análisis y diseño educativo"

2.1 Determinación de estructura del Módulo de Aprendizaje.

En este capítulo, presentamos en análisis y diseño educativo del módulo de aprendizaje en el cual está centrado el presente trabajo. Aquí encontraremos también, los objetivos educativos, el tipo de usuarios al que nos dirigiremos, las actividades a realizar, los tipos de conocimiento y otros aspectos más.

2.1.1 Descripción y características.

2.1.1.1 Temática:

Cuando hablamos de computadoras, muchos de nosotros no alcanzamos a comprender que éstas no funcionan por sí solas y que todo aquello que ellas realizan es mandado o "programado" por el ser humano. De este modo lo que es todavía más complicado de entender para las personas que no se dedican a esto cómo es que podemos ordenarles a las máquinas lo que deben hacer y cómo deben responder a las acciones que nosotros introducimos en ellas. Así es como llegamos a la explicación de lo que es un lenguaje de programación y sus características. Una de estas formas de comunicación se conoce como "Lenguaje de programación en C". Como cualquier otro tipo de conocimiento, hay que seguir ciertas reglas para adquirirlo y debemos cumplir algunos tópicos importantes para completar dicho conocimiento. Pensando en todo lo anterior, se planea realizar este software multimedia que enseñe las bases de la programación en lenguaje C a los estudiantes de computación de la Benemérita Universidad Autónoma de Puebla.

2.1.1.2 Propósito General

Los alumnos que utilicen este software deberían ser capaces, una vez que lo concluyan, de entender, explicar y aplicar los conceptos relacionados con los principales métodos de ordenamiento, así como su Implementación. Este conocimiento será transmitido mediante una variedad de actividades, como pueden ser: cuestionarios, actividades de arrastrar, asociación, entre otras.



2.1.1.3 Nivel de Usuario:

El tipo de usuario a los que está dirigido este software es un grupo muy específico de gente que debe cumplir con los perfiles que solicita la carrera de Ciencias de la Computación de la BUAP. Entre los requisitos aparece la edad, que va desde los 18 a los 28 años, un nivel de estudios medio superior, facilidad para las Ciencias Exactas, un alto grado de Inclinación por las actividades que requieren análisis e Iniciativa.

2.1.1.4 Tipo de Actividades

- a) Se realizará un software TUTOR NO DIRECTIVO, el cual, contendrá una amplia explicación de ejemplos y conceptos necesarios para aprender las bases principales de la programación, disponibles a través de animaciones, sonidos, textos y representaciones gráficas.
- b) Se elaborará una guía didáctica, la cual orientará al usuario en las actividades de aprendizaje, para navegar en forma idónea el software educativo multimedia.

2.1.2 Objetivos Educativos:

El software educativo que realizaremos, tiene como contenido un tema relacionado con la materia de Programación Básica en Lenguaje "C" y tiene un conjunto de objetivos fijos y claros que el estudiante de primer semestre de la carrera de Ciencias de la Computación de la BUAP.

El tema central del software es el tópico de ordenamiento de registros o archivos en lenguaje "C", de este modo podemos identificar claramente los siguientes objetivos de aprendizaje que el alumno deberá obtener al terminar de revisar el material que se le presentará al usar este software.

- Introducción a lenguaje "C"
- Funciones de Entrada y Salida
- Proposiciones

De este modo los objetivos claros del software son: que el alumno conozca y comprenda los objetivos anteriores.



2.1.3 Usuarios destinatarios.

Este tutorial está diseñado y estructurado para cierto grupo de usuarios que deben tener un grado de conocimientos generales sobre el equipo de computó, ya que en ninguna parte del software se dan Instrucciones de cómo deben manejarse los dispositivos de entrada y salida necesarios para obtener los resultados esperados durante el uso del software. El tipo de información que ya debe manejar la persona que desee usar este tutorial consiste en los conocimientos básicos de conjuntos matemáticos y el manejo de algunos conceptos fundamentales de programación como son los arreglos, variables y

sentencias de control de un programa escrito en lenguaje "C". Los usuarios ideales para este software son los estudiantes de la materia de Programación Básica en Lenguaje "C" de la Licenciatura en Ciencias de la Computación de la Facultad de Computación de la BUAP.

2.2.1 Contenido de Aprendizaje.

Dado que el software a desarrollar en este trabajo es un software de carácter educativo, es de vital importancia el asunto de los tipos de aprendizaje que el usuario o alumno puedan adquirir con este programa. Lo anterior nos obliga a definir cuáles son los aprendizajes relevantes de cada uno de los temas y subtemas. Existen varios métodos para realizar este diagnóstico de diferenciación entre un aprendizaje relevante de uno secundario. Uno de estos métodos consta de tres pasos que son:

- a) Seleccionar las ideas principales o esenciales de cada uno de los temas o subtemas.
- b) Determinar los procesos cognoscitivos deseables para cada idea esencial.
- c) Señalar el tipo de aprendizaje.



2.2.1.3 Descripción de Actividades por Subtema y Operaciones Mentales.

Aprendizajes Relevantes

Ideas Esenciales

Definición y características de un programa

1. El usuario aprenderá el concepto de lenguaje c y las características que los distinguen.

Proceso Intelectual. Comprensión de idea.

Aprendizaje. Es un aprendizaje esencial.

Pregunta 1.

Proceso de la programación

2. El usuario aprenderá los pasos para la elaboración de un programa.

Proceso Intelectual. Aplicación de la idea.

Aprendizaje. Es un aprendizaje indispensable.

Pregunta 2.

Aprendizaje antecedente.

El usuario debe saber que es lenguaje c.

Proceso intelectual. Repetición de idea.

Aprendizaje. Es un aprendizaje indispensable.

Funciones de entrada.

- 3.- El usuario aprenderá funciones de entrada para la aplicación en lenguaje c.

Proceso Intelectual. Analizar la idea.

Aprendizaje. Es un aprendizaje indispensable.

Pregunta 3.

Aprendizaje antecedente.

3.1 El usuario debe saber que es una función de entrada.

Proceso intelectual. Repetición de idea.

Aprendizaje. Es un aprendizaje indispensable.

Funciones de salida.

- 4.- El usuario aprenderá funciones de salida para la aplicación en lenguaje c.

Proceso Intelectual. Analizar la idea.

Aprendizaje. Es un aprendizaje indispensable.

Pregunta 4.



Aprendizaje antecedente.

4.1 El usuario debe saber que es una función de salida.

Proceso intelectual. Repetición de idea.

Aprendizaje. Es un aprendizaje indispensable.

Funciones de salida.

4.- El usuario aprenderá funciones de salida para la aplicación en lenguaje c.

Proceso Intelectual. Analizar la idea.

Aprendizaje. Es un aprendizaje indispensable.

Pregunta 4.

Aprendizaje antecedente.

4.1 El usuario debe saber que es una función de salida.

Proceso intelectual. Repetición de idea.

Aprendizaje. Es un aprendizaje indispensable.

Proposiciones

5.- El usuario aprenderá que proposiciones son utilizadas para el flujo de la información en el lenguaje c.

Proceso Intelectual. Analizar la idea.

Aprendizaje. Es un aprendizaje indispensable.

Pregunta 5.

Aprendizaje antecedente.

5.1 El usuario debe saber que es una proposición.

Proceso intelectual. Repetición de idea.

Aprendizaje. Es un aprendizaje indispensable.

2.2.2.1 Registro de Información

2.2.2.1.1 Actividad, medios y fuente de información

La estructura que se presenta en el sistema es ramificado y dinámico; cuenta con elementos motivadores como mensajes, también cuenta con puntajes en cada ejercicio para poder evaluar al alumno; se pretende que no sea extenso, a la vez que sea ameno y agradable al estudiante.

Se tratará de mantener entretenido al estudiante que las pantallas y la información no sean aburridas para que se interese por el sistema.



2.2.2.2 Lista de preguntas y respuestas por subtema.

Cap. I Introducción a lenguaje C

1.- ¿Qué es lenguaje C?

- a) Es un conjunto de pasos que tiene que realizar un programador.
- b) Es un lenguaje de programación de sistemas.
- c) a y b
- d) Ninguna de las anteriores

2.- ¿Qué es un programa?

- a) Es un lenguaje de programación de sistemas.
- b) Es un conjunto de instrucciones, ordenes dadas a la máquina que producirán una determinada tarea.
- c) Es un conjunto de pasos que tiene que realizar un programador.
- d) Ninguna de las anteriores

3.- ¿Cuáles son los procesos de la programación?

- a) Definición del problema, diseño del algoritmo, codificación del programa, depuración y verificación.
- b) Definición del problema, diseño del algoritmo, depuración y verificación.
- c) Es un conjunto de instrucciones, ordenes dadas a la máquina que producirán una determinada tarea.
- d) Ninguna de las anteriores

4.- ¿Qué es una función de entrada?

- a) Permite leer datos entre archivos y dispositivos.
- b) Permite escribir datos entre archivos y dispositivos.
- c) a y b
- d) Ninguna de las anteriores

5.- ¿Qué es una función de salida?

- a) Permite leer datos entre archivos y dispositivos.
- b) Permite escribir datos entre archivos y dispositivos.
- c) a y b
- d) Ninguna de las anteriores

6.- ¿Qué son las proposiciones?

- a) Son las que controlan el flujo y la ejecución de un programa.
- b) Son las que controlan el flujo del hardware.
- c) a y b
- d) Ninguna de las anteriores



2.2.3 Cuadro de decisiones de aprendizaje

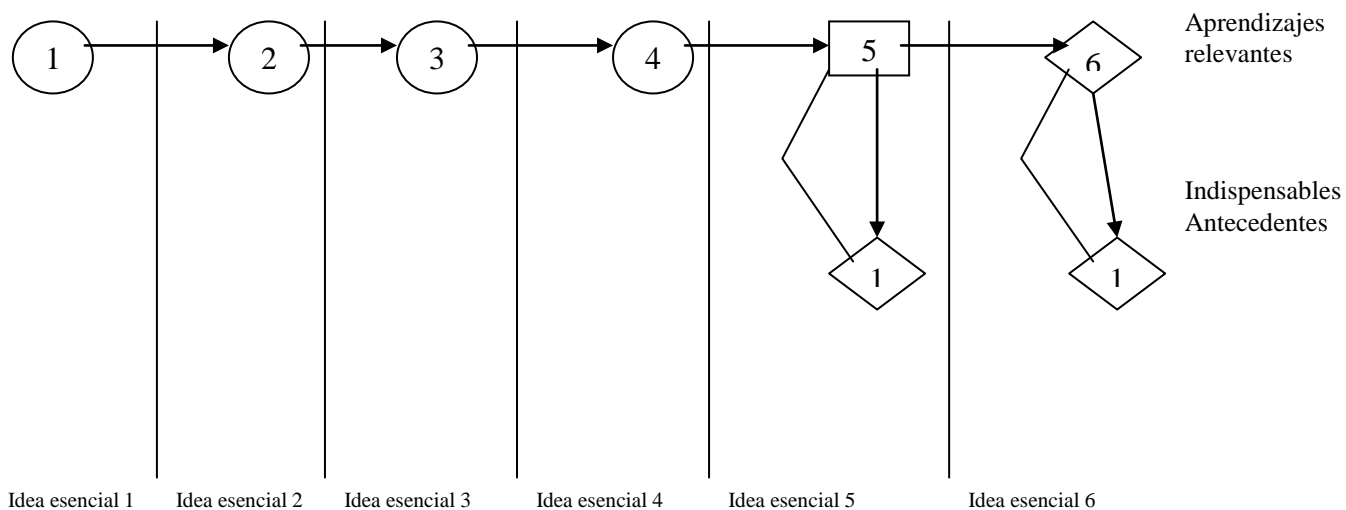
A continuación en la tabla 2.2.3, se presenta el cuadro de decisiones que representa los tipos de aprendizaje que estarán presentes en el sistema.

IDEA	Proceso intelectual									
	I	E	A	R	C	Re	Ej	In	Ap	An
1	1				1					
2	2								2	
2.1			1	1						
3	3									3
3.1	3			3						
4	4									4
4.1	4			4						
5	5									5
5.1	5			5						

Tabla 2.2.3 Cuadro de decisiones de aprendizaje.

2.2.4 Diagrama de flujo de aprendizajes

En la siguiente figura se muestra el diagrama de flujo referente a los tipos de aprendizaje.





2.2.1.3 Estrategias de Enseñanza-aprendizaje y actividades iterativas (tipo), Duración y Actividades Mentales:

Para lograr que el usuario o alumno de este material educativo adquiriera los conocimientos necesarios, recurriremos a las siguientes estrategias de enseñanza-aprendizaje:

- Presentación audio-visual del material
- Actividades de arrastrar y soltar
- Lectura y ejemplos
- Animaciones.

De la variedad de actividades mentales que existen en las estrategias de enseñanza-aprendizaje, se ha identificado que sólo se usaran 4 de estas actividades, las cuales son suficientes para que el usuario del software adquiriera los conocimientos que se pretenden.

2.2.1.4 Motivación

Cuando hablamos de aprendizaje, debemos entender que cualquier proceso de aislamiento del conocimiento conlleva cierta cantidad de tiempo, si por cualquier circunstancia este proceso de aislamiento se retrasa o se nos complica, requerimos de algún método que vuelva a lograr que el individuo que trata de terminar este proceso de aislamiento mantenga o tenga nuevamente la inquietud de culminar el proceso.

De este modo los aspectos de motivación en cualquier proceso de aprendizaje son muy importantes para lograr que el usuario mantenga la atención sobre el material que se aprende, y en el caso de que el aprendiz se encuentre desganado, será sumamente importante que el aspecto motivacional sea el indicado para que no sea contraproducente.

Si el aspecto de motivación es especialmente importante en un proceso de aprendizaje en el que existe un maestro y un alumno, es más importante aun, cuando este proceso de aprendizaje lo tratamos de llevar entre un equipo de computó y un alumno. Por lo anterior y debido a que este software de aprendizaje está dirigido a un usuario común, de nivel medio superior, es básico que el aspecto motivacionales provoquen una reacción negativa debido a que la motivación pueda parecer en este software con un nivel de conocimientos inferiores al que está buscando. Así pues, el aspecto de motivación que nosotros utilizaremos será, simplemente, frases de aliento al terminar o realizar un ejercicio correctamente.



CAPITULO III “Análisis y Diseño del Sistema”

3.1 Planteamiento del problema.

Con este progreso de la tecnología y las teorías de aprendizaje se hace notorio que debemos aprovechar todos estos avances y proporcionar a las nuevas generaciones mayor cantidad de Información y de herramientas para que logren aprender los temas o tópicos que se pretende conozcan, sea cual sea la materia, nivel, disciplina o tema del que se trata.

De este modo debemos buscar la manera de realizar material de apoyo para los estudiantes de las distintas áreas y niveles. Una de las formas más eficientes de generar material educativo e calidad y utilidad es usar la tecnología multimedia, la cual está ya comprobado, que estimula al aprendizaje, debido a que captura la atención de más de uno de los sentidos y no como lo hacen los métodos convencionales de aprendizaje.

La cantidad de alumnos que presentan problemas para aprender los tópicos de la materia de Programación Básica de la Licenciatura en Ciencias de la Computación de la BUAP, nos lleva a tratar de producir un material educativo multimedia que se acople a la petición al programa de estudios y además, sirva como material de apoyo para los estudiantes y profesores de este nivel educativo. En particular el material que cubrirá este software multimedia es el tópico de “Sistema Multimedia para la Programación en Lenguaje C”.

3.2 Especificación de Requerimientos.

El software educativo multimedia, deberá contener la información necesaria para cubrir adecuadamente, y en el nivel educativo que corresponde a la Benemérita Universidad Autónoma de Puebla, el tópico de “Sistema Multimedia para la Programación en Lenguaje C” correspondiente al primer semestre de la carrera de Ciencias de la Computación.

El contenido didáctico del software educativo, está protegido al plan de estudios de la BUAP y es en su totalidad el material que se presenta en el apéndice A de este documento.

El usuario de este software tendrá siempre a la mano la ayuda del programa, misma que será temática, es decir, dependiendo de la zona en la que nos encontramos, será la ayuda que nos proporcione y también se podrá desde ahí acceder a cualquier otra parte del apoyo.



Dado que es un software educativo, el usuario podrá repasar cuantas veces quiera la sección del tema que no haya comprendido, así mismo, podrá pasar al siguiente tema, si así lo desea, aún cuando en las actividades de calificación no se obtenga el mínimo recomendado de conocimientos para continuar con los temas siguientes.

Los colores, ubicación, fuente, dibujo, videos y demás materiales multimedia utilizados para conformar la interfaz del usuario, no podrán ser modificados ni alterados por el usuario, esta interfaz será estándar durante todo el programa, permitiendo así, que el usuario del software se concrete en el contenido educativo del programa.

Se incluirán actividades de arrastrar y soltar, completar, rompecabezas, preguntas y respuestas, relacionar y elección múltiple. Todas estas actividades son con la finalidad de que el alumno asimile los conocimientos que se pretenden.

Para la realización de este trabajo necesitaremos por lo menos una computadora con capacidades multimedia y software de desarrollo multimedia como lo es Director de "Macromedia". La intersección de especialistas en educación, diseño gráfico y computación, es vital para el correcto desarrollo del software. Con estos especialistas el trabajo puede realizarse en 5 meses.

Para poder ejecutar este programa con propiedad, la computadora destino debe tener monitor o dispositivo compatible con la resolución de 800 x 600 píxeles y 16 bits de color, debe contar con un lector de CD-ROM de por lo menos 12x de velocidad, un sistema operativo compatible con Windows 9x o posterior, se requiere también 64 MB en RAM o más y un procesador Pentium 133 Mhz. O compatible. Si el equipo tiene un hardware superior al que se menciona aquí, será mucho mejor y el software tendrá un mejor desempeño.

El trabajo será conocido cuando el experto en educación y el experto en el tema de " Sistema Multimedia para la Programación en Lenguaje C" coincidan en que se cumple con los objetivos trazados para este proyecto.



3.3 Guión Literario

El software multimedia que presento, será distribuido en CD, Para trabajar con este software, sólo es necesario introducir el disco en el lector de la computadora, si el equipo tiene activada la opción de inicio automático, el software se iniciará solo; de lo contrario, debemos ejecutar el archivo de nombre de "Sistema Multimedia para la Programación en Lenguaje C" por medio de hacer "doble-click" con el botón izquierdo del "Mouse".

De inmediato aparece la pantalla de introducción, la cual presenta una animación con el escudo de la "Benemérita Universidad Autónoma de Puebla", y se escuchan efectos de sonido, posteriormente, aparecerá la pantalla de bienvenida, que muestra la siguiente información por medios escritos y auditivos:

Institución que presenta el software
Tema presentado
Título del tema

La ejecución del software continúa y nos presenta la pantalla de entrada en la que se debe proporcionar un nombre para continuar, el software no seguirá su ejecución mientras el recuadro asignado para teclear un nombre permanezca vacío o con la leyenda "Ingrese nombre aquí". Una vez que introducimos el nombre del alumno en la casilla adecuada debemos 5 botones de acción, mismos que ya no desaparecen durante el resto de la ejecución del software. Estos botones son:

Salir
Ayuda

La pantalla que el software nos presenta a continuación, se denomina el menú principal en la cual se incluyen los datos del software como contenidos, temas a tratar, objetivos, etc. En esta parte, aparecen otros botones de acción que son:

Menú principal
Siguiendo
Anterior



En este momento, el usuario del software podrá elegir cualquiera de los temas que se le presenta del lado izquierdo de la pantalla, una vez elegido el tema, la pantalla cambia relucéndose el sector izquierdo para dejar más amplio el sector derecho de la misma, en este sector derecho se presenta de modo multimedia el contenido del tema que se ha escogido aprender. Dicho tema será presentado y durante esta presentación se darán ejemplos y toda la teoría necesaria para que al terminar del mismo, el alumno sea capaz de realizar y terminar con prontitud y exactitud los ejercicios que se encuentran al finalizar. En todo momento el usuario contará con el botón de la ayuda, misma que es contextual; es decir, irá variando dependiendo del punto en que nos encontramos. Así contará con botones de navegación que sirven para avanzar, retroceder, dentro del tema o sub-tema que se presenta, estos botones de navegación son distintos a los botones llamados “Anterior” y “Siguiete”.

Al pasar en Mouse por cualquiera de los temas que se presentan:

Introducción a lenguaje C
Funciones de entrada y salida
Propósitos
Ejercicios

Se despliega el menú con las opciones que tiene cada tema, cada una de estas opciones, al pasar el Mouse por encima, genera un sonido y los textos cambian de color para indicar la selección. Una vez seleccionada la opción a estudiar, el software presenta mediante, texto, movimiento y sonido todo el material necesario hasta llegar a la sección de ejercicios para dicho tema.

Cada una de las secciones de ejercicios lleva el registro de los problemas que se resolvieron y aquellos que no fueron resueltos, este registro es borrado cuando el software se detiene y es reiniciado.



3.4 Guión Técnico

El Software tiene solamente el CD, el cual no es editable.
Dentro de la estructura de este CD aparece en la raíz:

- Ultimo. Exe

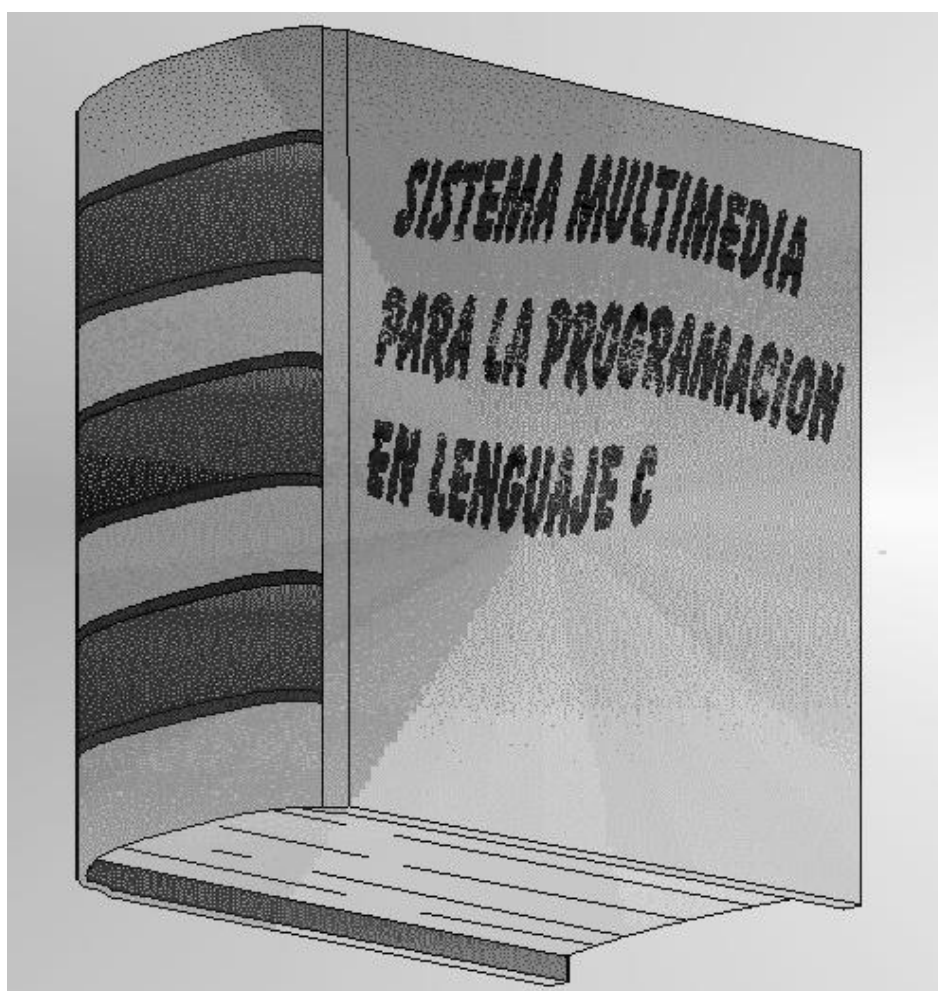
El archivo “Ultimo.exe” tiene el software que se muestra durante la ejecución del programa.

Una vez ejecutado el programa, lo primero que aparece es una animación de dos escudos de la BUAP, en color y fondo naranja. Estos escudos hacen un recorrido por la pantalla, uno empezando de atrás hacia delante y después empieza de abajo hacia arriba y colocarse al centro.



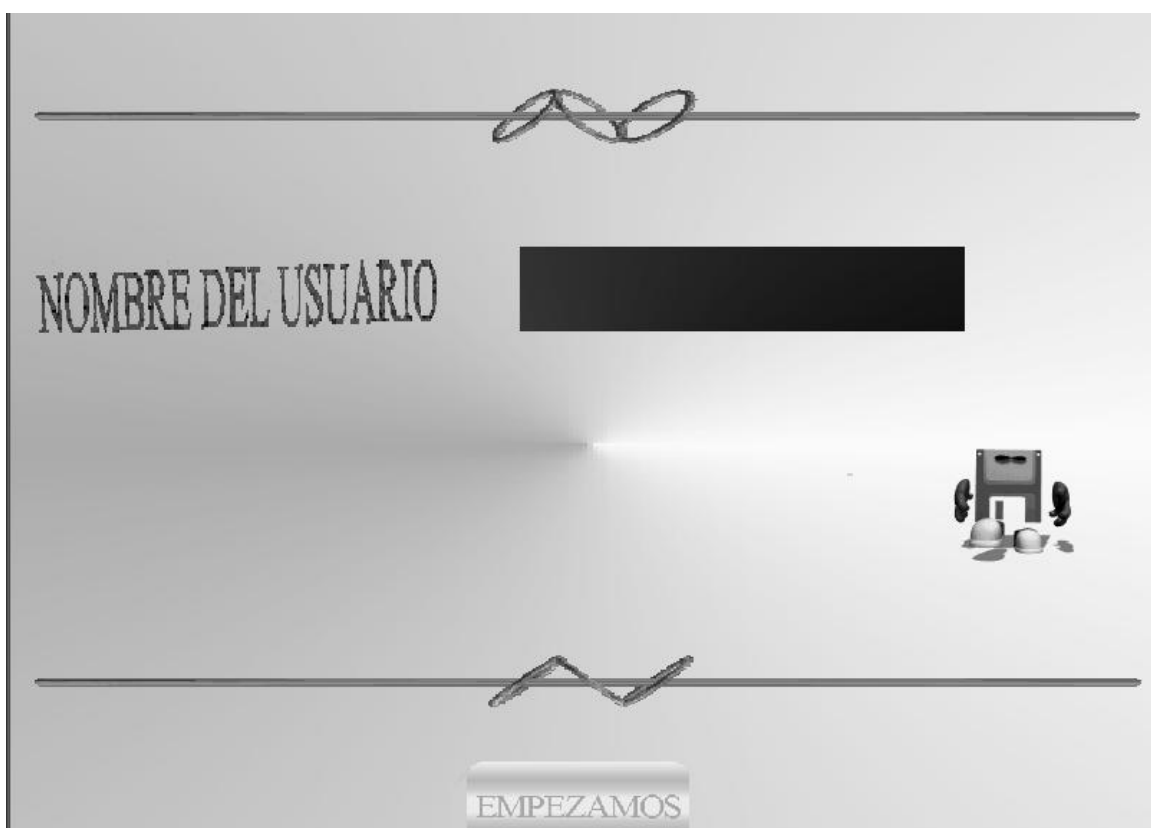


En este momento, la pantalla en cuestión cambia para mostrar el nombre del software que se va a utilizar.



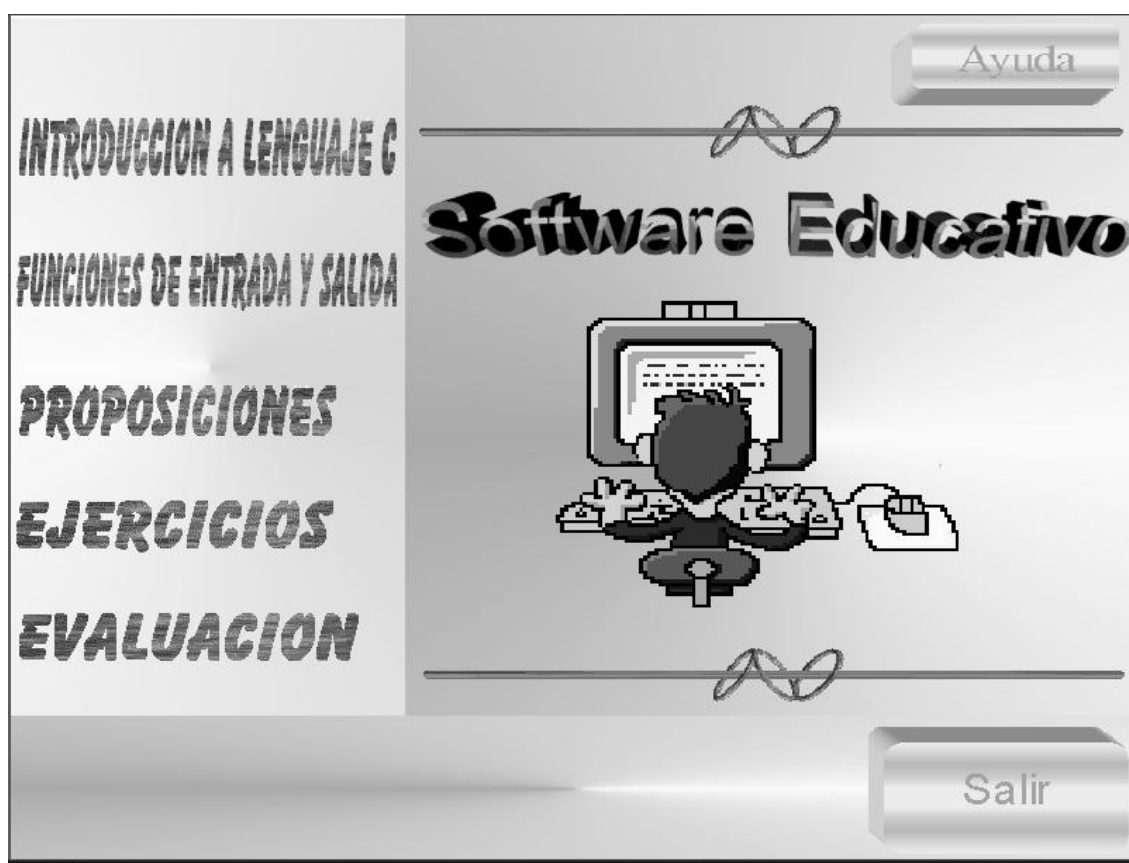


En este momento, la pantalla en cuestión cambia para mostrar el nombre del software que se va a utilizar.





Esta es la parte del software donde debemos introducir el nombre del usuario, para poder continuar y una vez escrito el nombre de usuario debemos activar el botón de empezar para poder continuar.



Ahora es cuando tenemos la pantalla que llamamos “menú principal”, esta pantalla es llamada así, ya que es desde aquí donde controlamos el flujo que tendrá el software. En esta pantalla tenemos los 4 temas que presenta este sistema multimedia de aprendizaje y los botones de control de navegación, empezar, ayuda y botón salir.

En esta pantalla al pasar por cada uno de los temas del menú, con el Mouse automáticamente cambia el puntero para activar un tema.



Esta ventana muestra las definiciones de cada tema que forman la programación en lenguaje en lenguaje "C" como se muestra a continuación:

INTRODUCCION A LENGUAJE C

DEFINICION DE UN PROGRAMA

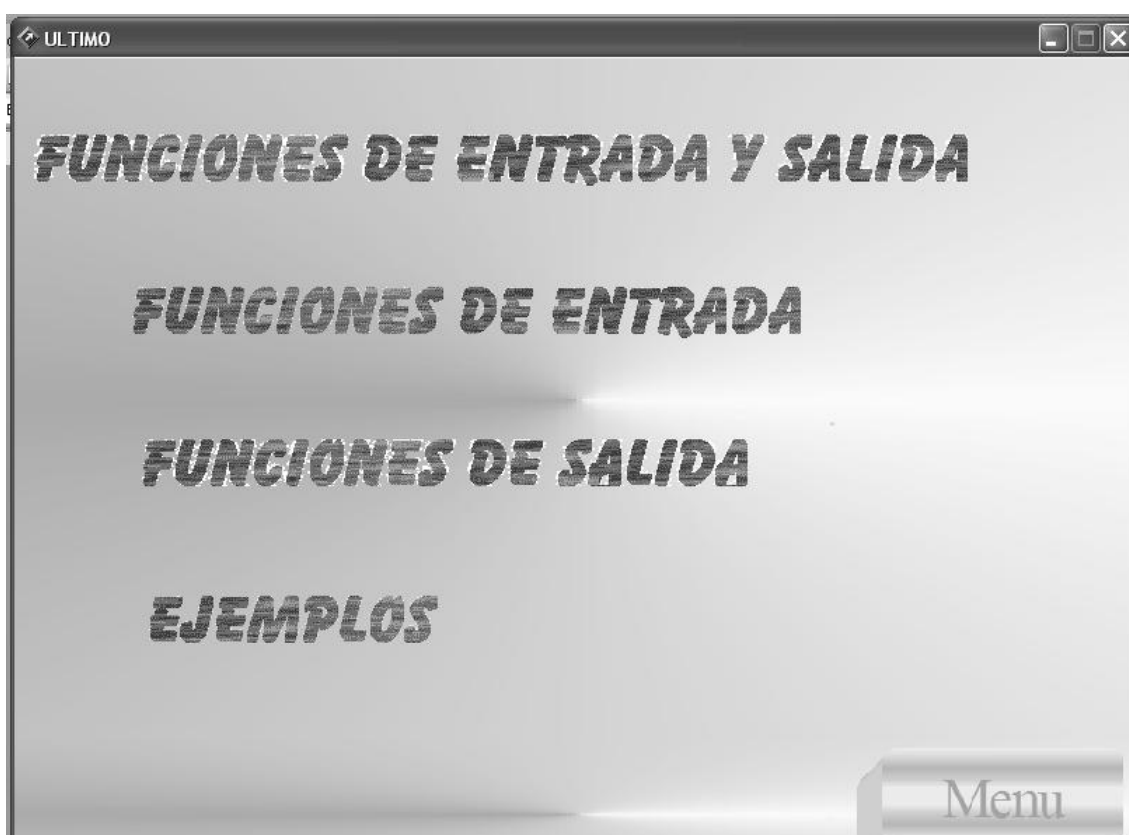
PROCESO DE LA PROGRAMACION

PARTES CONSTITUTIVAS DE UN PROGRAMA

Menu

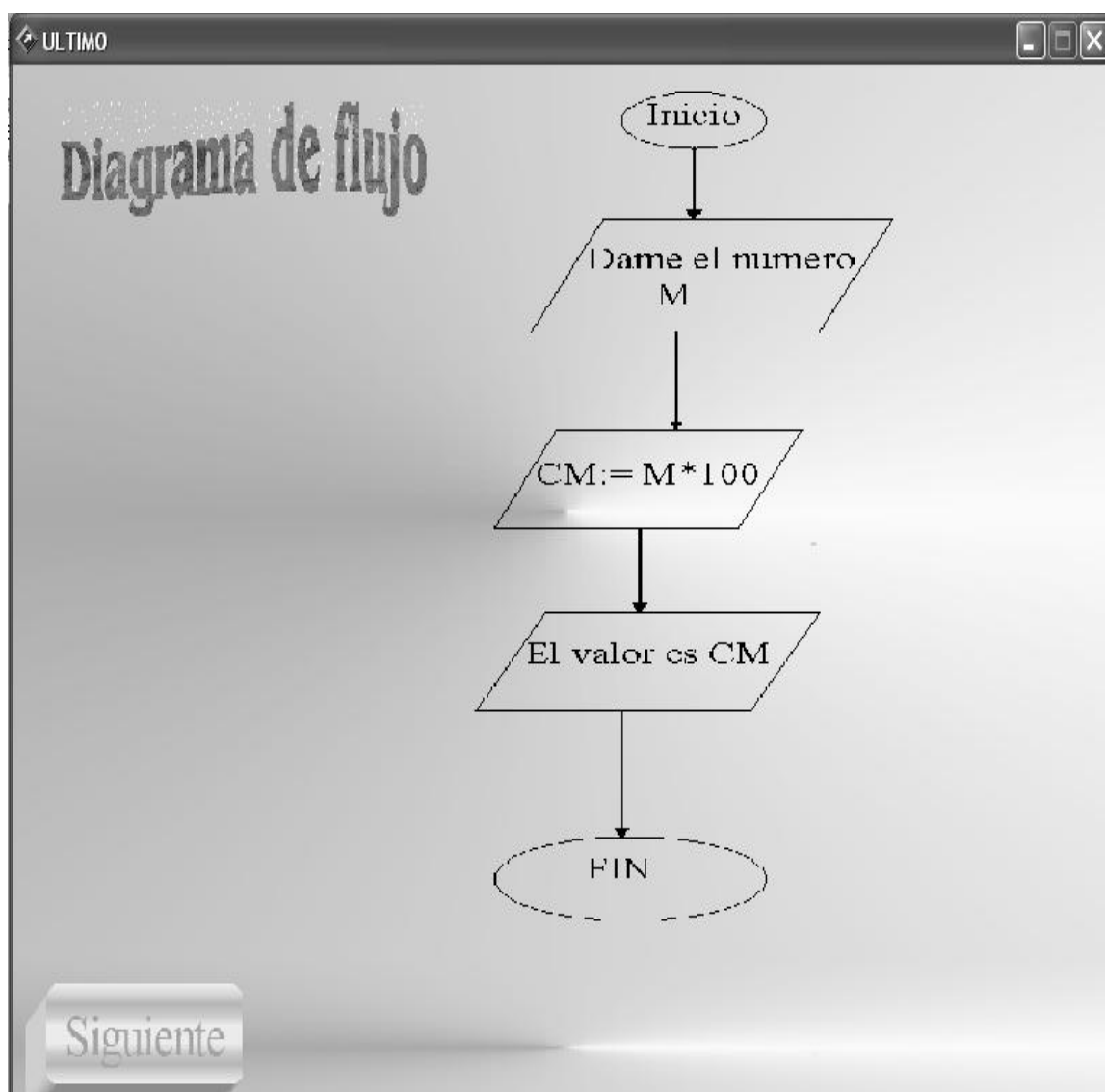


Se define a continuación los tipos de funciones de entrada y salida en lenguaje "C" así como también mostrar algunos ejemplos sobre estas funciones.





Dentro de los ejemplos se muestra un diagrama de flujo como se muestra a continuación que es convertir metros a centímetros:



:



Después del diagrama de flujo desarrollamos el programa que es el siguiente

```
ULTIMO
```

Programa

```
#include<stdio.h>

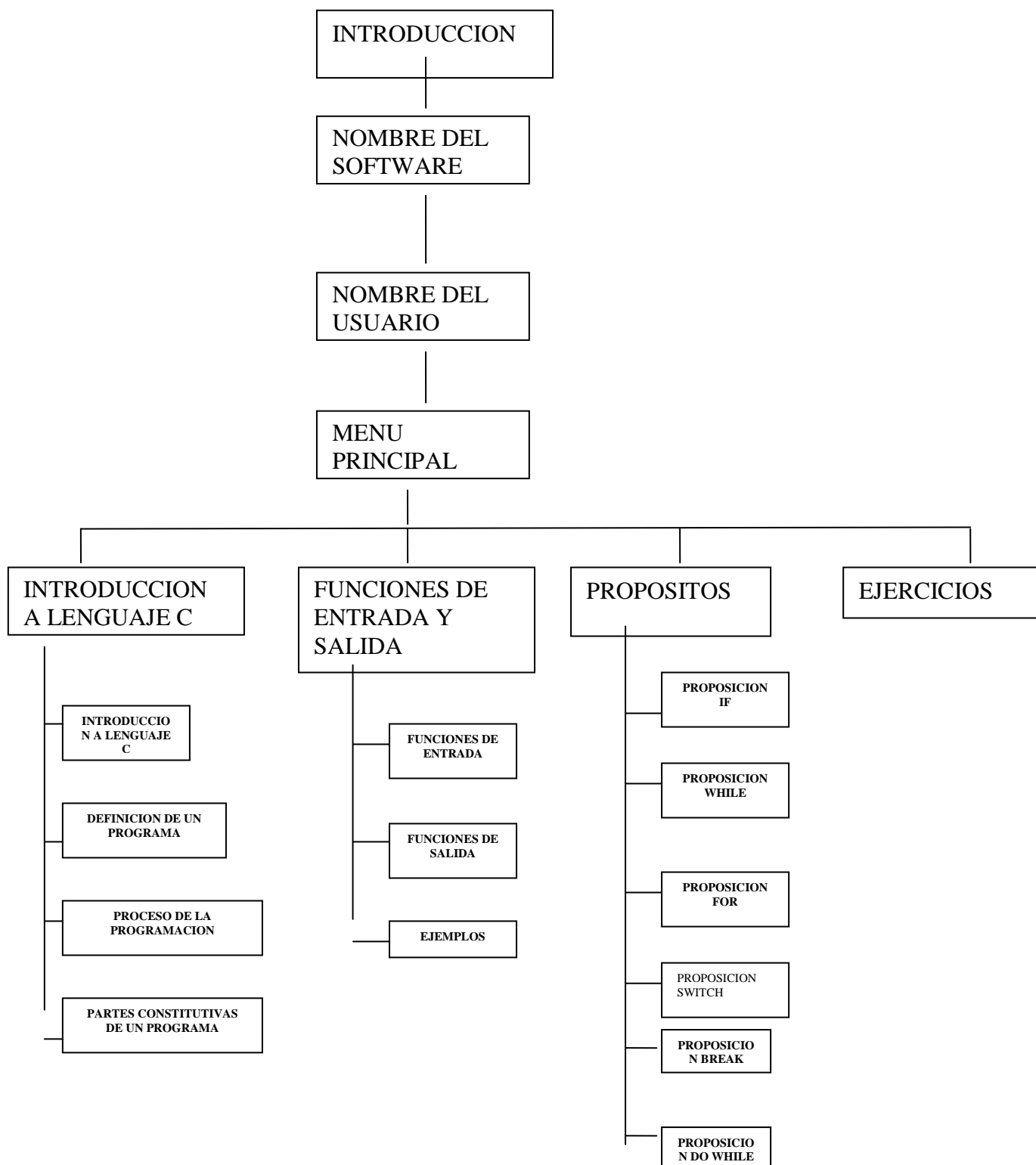
int main(void)
{
    int M,CM;
    printf("\n Dame el número que quieres convertir:");
    scanf("%d",M);
    CM:=M*100;
    printf("\n El valor de los centí metros es%d:",CM);
}
```

Regreso Menu



3.4 Diagramas

Para comprender cómo funciona este sistema, tenemos a continuación los diagramas que nos dicen cómo está diseñada la estructura de cada una de las partes o módulos de los que consta.





Los cuatro primeros bloques se ejecutan en secuencia uno tras otro y sin la posibilidad de regresar, a menos que sea reiniciado el software. El primer bloque, a introducción, está dividido en dos partes, la primera donde se presentan dos estudios de la BUAP con movimientos y sonidos, la segunda parte es la bienvenida, que presenta el escudo de la BUAP de forma estática y la voz del narrador explica de qué se tratará el software. Este bloque tiene una duración de 255 “frames”.

El bloque dos es el encargado de mostrar el nombre del software que se está presentando.

El bloque tres es el encargado de registrar al usuario con la finalidad de poder generar un reporte de su desempeño. Durante este bloque, el espacio asignado para registrar al usuario, aparece con la leyenda “ingrese el nombre del usuario”, es importante decir que el software no continúa su ejecución mientras no se introduzca un valor diferente a “vacío” o el valor por default que es “ingrese el nombre del usuario”.

El bloque cuatro es en si el bloque del cual parten todos los otros bloques, dado que él contiene el “menú principal” a partir del cual el usuario escoge el tema que quiera estudiar.

3.5 Diseño de la interfaz.

Al diseñar las pantallas del programa, como se muestra a continuación se busca que la persona que está utilizando el software Educativo Multimedia se familiarice con el programa. Desde el momento en que ve la primera pantalla se busca que la interacción que tenga con todos los elementos de ella, reaccionen de la misma manera cada vez que se utilicen, independientes del espacio- tiempo en que se requiera.



CAPITULO IV “Desarrollo del Sistema”

4.1 Herramientas de Software

Para la realización de este sistema educativo fue necesaria la utilización de diferentes paquetes de software especializados en retoque digital, manejo de audio, manejo de multimedia, etc. El uso correcto de esta paquetería desemboca en una calidad óptima del producto a realizar.

4.1.1 Sound Forge 4.5

Este software de edición de sonidos es de uso que le demos es de nivel bajo, pero al mismo tiempo muy completo e incluye una gran cantidad de opciones para gente experta en sonido. Este paquete de software nos permite grabar sonido en varios formatos y en varias calidades, por ejemplo:

- Formato wav
- Formato voc
- Formato avi
- Formato asf
- Calidad de 44,100 KHz 8 bits, stereo
- Calidad de 44.100 KHz 4 bits, mono
- Calidad de 22 KHz 8 bits stereo

La manipulación del sonido mediante “Sound Forge” es muy flexible, ya que nos permite grabar en dos canales de audio o en uno sólo, si grabamos en dos canales podemos modificar independientemente cada uno de los canales. Podemos también unir dos o más sonidos diferentes, subir o bajar volumen y además, tenemos una gran variedad de efectos de audio como:

- Efectos Eco
- Efecto de aviones.
- Invertir Sonidos
- Sonidos Robotizar
- Acelerar Sonido
- Bajar la velocidad



Para lograr la edición exacta del audio podemos realizar un “zoom” a la parte de la onda de sonido que queramos, de tal manera que el sonido sea cortado, añadido, aumentado o modificado en el momento justo.

4.2 Director 8.5

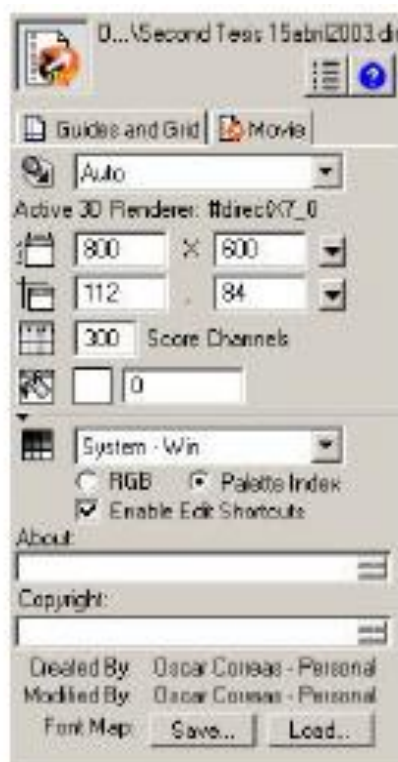
Los programas de autor, como Director de Macromedia utilizan la analogía de dirigir una película o una obra de teatro para simplificar el proceso e incluso aluden a una producción multimedia como si fuera una película. El software de autor se utiliza para crear sonido, video, texto, gráficos y animaciones complejas partiendo de cero, así como para crear una obra literativa basándose en los enlaces que unen dichos elementos, empleados como “miembro del reparto” para producir la película.

4.1.2 Escenario o “Stage”

La pantalla vacía en la que se colocan los miembros del reparto se llama “escenario”. El diseñador selecciona los elementos visuales: gráficos de fondo, botones en pantalla, íconos animados, etc. Y utiliza el ratón para situarlos en la posición adecuada; los miembros del reparto que presentan acontecimientos, como las secuencias de video y los efectos sonoros, se colocan en una línea de flujo que sincronizan las entradas, los movimientos en el escenario y las salidas. Por ejemplo, si el diseñador quiere mover un elemento por el escenario mediante el ratón, con la línea de flujo se registrarán de forma automática el lugar hacia donde debe moverse.



El escenario también tiene propiedades que podemos configurar para que el software se ejecute como queremos que lo haga. Entre estas propiedades está el tamaño de la ventana en la que se visualizará el software. La cantidad de cuadros que tiene, color, paleta de colores, etc.



4.2.2 Reparto o "Cast"

Los miembros del reparto pueden ser de tipo importados, es decir, los creados con otras herramientas multimedia y no pueden ser editados en Director. El otro tipo de miembros de reparto son los creados por las propias herramientas de Director y por ende, pueden ser modificados desde el propio Director. Los miembros de reparto de tipo importados son:

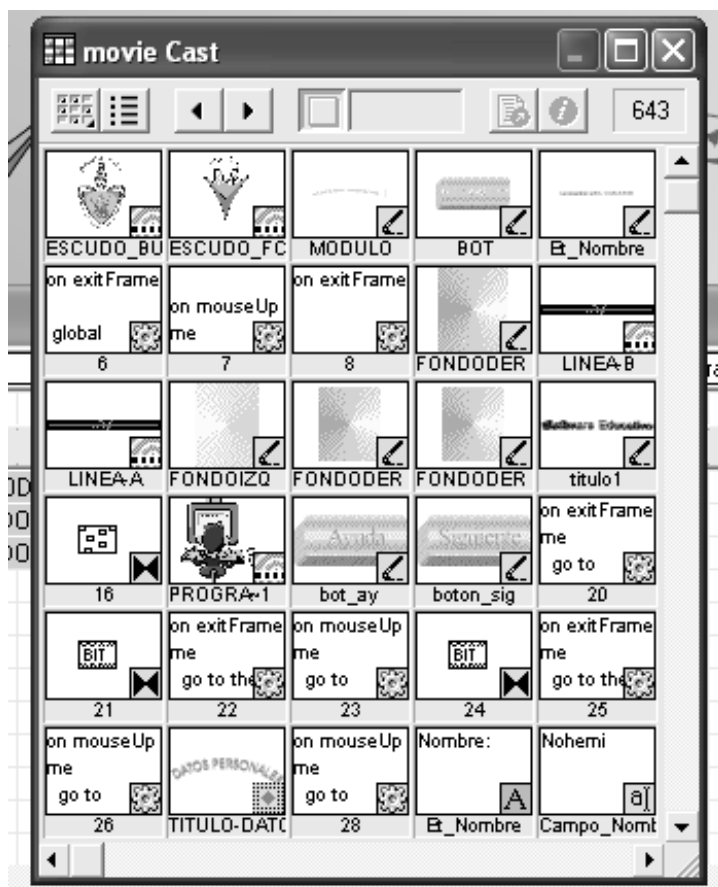
- Imágenes: en formato bmp, jpg, psd, Pict, MacPaint, tiff, eps, gif, png, ing, pcd, pcx y wmf.
- Texto: Archivos de extensión rtf y txt. Así como cualquier archivo de texto de editores externos.
- Animación: en formato fic, fil, pics, Srapbootit.



- Sonidos: de tipo aiff, wav y sonidos del sistema 7 para mac.
- Video Digital: archivo avi o mov.
- Repartos Externos.
- Películas de Director y de Flash.
- Paletas de color: en formato pal de Windows.
- Archivos de cualquier tipo Importados con enlaces OLE (elementos enlazados): documentos de Word, gráficos de Lotus, etc.

Los miembros de reparto de tipo creados:

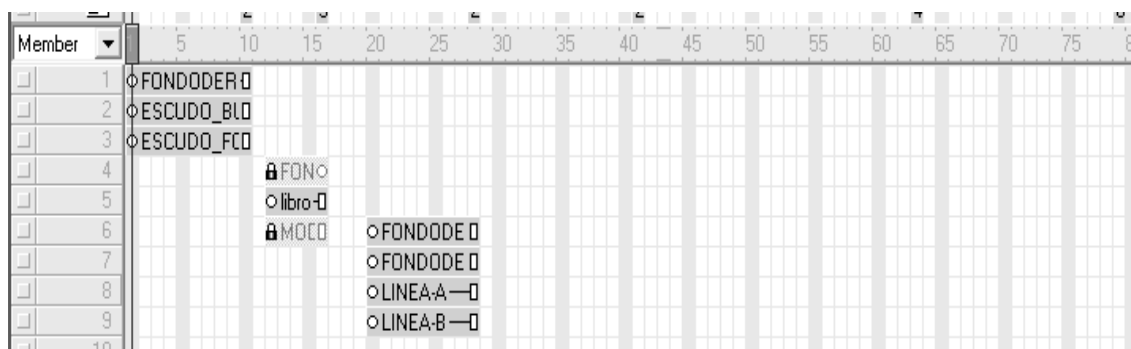
- Botones.
- Campos de texto.
- Formas o Shapes.
- Guiones o Scripts.
- Formas Vectoriales.
- Comportamientos.
- Transiciones.





4.2.3 Canales de sonido

En director 8.5 existen dos formas de manejar el sonido: arrastrando algún miembro del reparto que sea de audio hasta el “score”, justo en alguno de los dos canales que están disponibles.



Sin embargo podemos tener mucho mejor mayor control sobre el sonido que reproduce nuestra película si usamos el lenguaje lingo. Por este motivo, esta sistema sólo utiliza los canales de audio del “store” durante la introducción de lingo. Algunos de los comandos que se utilizan son:

```
-puppetSound whichChannel, whichCastMember  
-Sound (whichChannel)_stop()  
-soundBusy(whichChannel)
```

Se cuenta también con la propiedad “volumen”, misma que puede tomar un valor entre 0 y 256. Con esta propiedad podemos bajar, subir o suspender el volumen de algún audio. La sintaxis es:

```
-soundBusy (whichChannel)
```



4.2.4 Partitura

Una parte importante de la creación de películas en Director es la partitura que es aquella ventana desde la que se controla y coordina la interpretación de los personajes en el escenario. La partitura está dividida en pequeños cuadros definidos por franjas horizontales y verticales, las franjas horizontales se denomina Canales (channels) y las verticales Cuadros (Frames). Los canales albergan los distintos personajes colocados en el escenario, indican el modo en que a parecen en de él, el tiempo que permanece y su función en el mismo; los frames o cuadrados marcan la sucesión del tiempo en la película, ya que cada cuadro conforma una imagen estática, que al suceder a otro cuadro y ser sucedido por un nuevo cuadro, componen la animación (recordemos la metáfora del cine).

La partitura de Director 8 posee 999 canales, lo que quiere decir que podremos tener hasta 999 personajes en un mismo cuadro en la película. El número de frames está limitado a 150, aunque sí colocamos un personaje en el cuadro 150, la partitura de Director nos dará otros 150 frames. El desarrollo de la acción está marcado por la cabeza de lectura, que es la que indica qué cuadro estamos visualizando en la pantalla en cada momento. Podemos situar la cabeza de lectura en cualquier punto de la partitura, pulsando un clic sobre un frame, entonces, la cabeza de la lectura, que es la que indica qué cuadro estamos visualizando en la pantalla en cada momento. Podemos situar la cabeza de lectura en cualquier punto de la partitura, pulsando un clic sobre un frame, entonces, la cabeza lectora se desplaza al punto seleccionado y el frame marcado se rellena de negro quedando, así, seleccionado.

Cuando situamos un miembro del reparto en la partitura, éste se coloca en un canal determinado, ocupando un número de frame's, que son 28 por defecto; esto significa que este miembro, que ahora al pasar del reparto a la partitura se le llama "sprite". Estará presente por 28 fotogramas y después desaparecerá.

Los canales ocupados indican el modo en que los personajes se sitúan en el escenario: el personaje situado en el canal 1 se encuentra en el escenario por debajo del que ocupa el canal 2 que, a su vez, está por debajo del 3, y así sucesivamente. Es decir, si nuestra película tiene un fondo que ocupa todo el escenario para que este fondo no tope ningún personaje, deberá situarse en el canal 1.



Existen canales especiales que reciben el nombre de “canales de efectos”, que se sitúan en la parte superior de la ventana de la partitura. Estos canales especiales tienen tipo de elementos muy determinados que son:

- Canal de tiempo: Contiene información sobre el tiempo de la película, como por ejemplo. Cambia la velocidad de la película (en frames por segundo), detenerla durante algún tiempo o a la espera de alguna acción.
- Canal de paleta: Almacena la información sobre los cambios en las paletas de color de la película.}
- Canal de transiciones: Nos permite crear transiciones entre un frame y otro, como por ejemplo, difundidos, desplazamientos, disoluciones, etc.
- Canal de sonido: estos son dos y almacenan sonido que puede ejecutarse al mismo tiempo..
- Canal de guión: Aquí se tienen los scripts que se ejecutan automáticamente cuando la ejecución de la película llega a un determinado cuadro.

4.2.5 Animación en Director

La posibilidad de animar los personajes en la pantalla con una gran facilidad es una de las grandes ventajas de “Director”. La exploración de las capacidades de este software depende de que conozcamos todo lo que se puede hacer. En este caso existen diferentes tipos de animaciones con “Director”.

En el primer tipo de animación, que es el más sencillo, consiste en el desplazamiento en un mismo personaje por la pantalla desde un punto inicial a un punto final. Este tipo de animación es ahora con “director 8.5” algo muy fácil de hacer, gracias a una nueva propiedad que se llama “tweening”. En Director 8.5”, los “sorties” poseen la propiedad denominada “tweening Automatico”, lo que determina esta propiedad es que el punto inicial y el punto final del “sprite” están enlazados de tal modo que las modificaciones realizadas en cualquiera de ellos afectan al resto de los “frames” intermedios. Los pasos para realizar este tipo de animación son:

- Colocamos el personaje en el cuadro de inicio y canal de la partitura que nos interese.
- Aspiramos el personaje a una posición adecuada.
- Seleccionamos el “frame” final y arrastramos el “sprite” a la posición que queremos que se mueva.



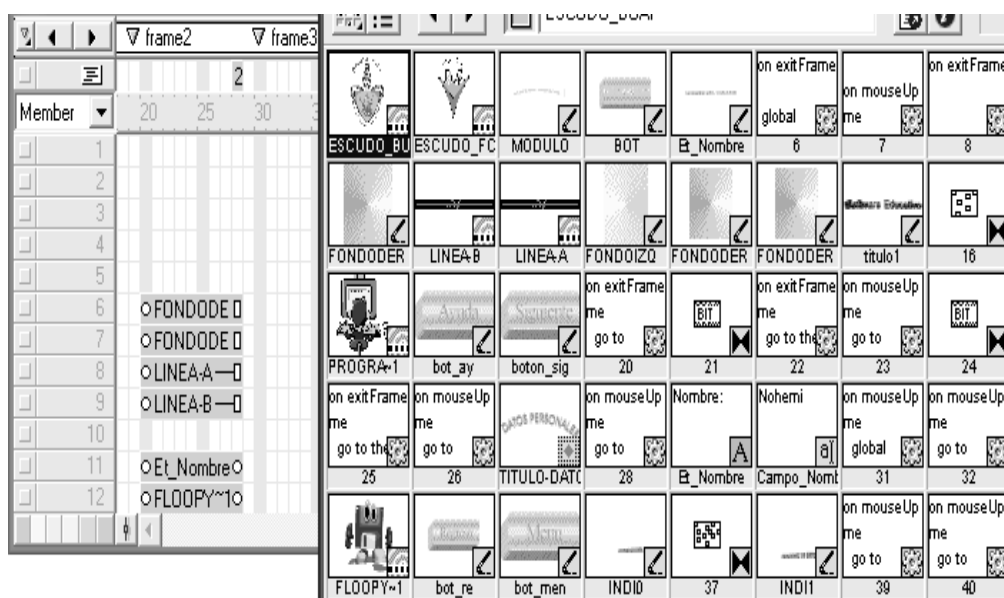
El segundo tipo de animación consiste en la incorporación al escenario de distintos personajes que se simulan no sólo en varias fases de una acción. Esta técnica es la misma que se utiliza en los dibujos animados, en donde se tiene un primer dibujo y se van sobreponiendo otros con cambios pequeños de tal modo que al ver todo junto aparece el movimiento. Para esta técnica de animación lo que debemos hacer es importante todos los archivos que formen parte de la animación que queremos realizar y colocarlos en orden dentro de la ventana “cast”. Una vez hecho esto, esta técnica se puede hacer de tres formas:

-Forma Artesanal: Consiste en colocar uno a uno todos los miembros de 1 reparto en la partitura ocupan frames consecutivos, de este modo, al pasar la cabeza lectora aparecerá la animación.



-Forma “cast to time”. Esta técnica consiste en unir todos estos personajes en uno sólo y después llevarlo al escenario como solamente un personaje. Para hacerlo, necesitamos seleccionar todos los miembros del reparto que van a pertenecer a la animación. Esto se hace con la ayuda de la tecla “shift” que nos permite seleccionar un conjunto consecutivo de miembros del reparto. Ahora seleccionamos la opción “cast to time” del menú a modificar.

-Forma “Space to time”. Esta técnica tiene como resultado, al igual que las técnicas anteriores, la creación de un único “sprite” de los distintos personajes, solo que en este caso, los personajes quedan separados por “frames claves”, mismos que nos dan flexibilidad al momento de modificar el “sprite” resultante.



El tercer tipo de animación es el de mayor exactitud, ya que consiste en los pasos en la grabación de los pasos de la animación, mientras la realizamos sobre el escenario. La técnica consiste en grabar la posición del “sprite” (en posición, tamaño, mezcla, colores, etc.) grabar su nuevo estado y así sucesivamente. Para grabar una animación paso a paso debemos hacer:



- Mostrar el panel de control, desde el menú Windows, o teclear "ctrl. +2"
- Colocamos en el escenario todos los personajes que intervendrán en la animación y seleccionamos los "sortees" de aquellos que vamos a animar.
- Situamos la Cabeza lectora en el "frame" donde deseamos iniciar la grabación.
- Desde el menú de control, escogemos la opción "sto.Recording"
- Modificamos el o los personajes en el escritorio.
- Pulsamos el botón "stepForward" del panel de control.
- Realizamos un nuevo cambio en los personajes y pulsamos de nuevo "stepForward".
- Repetimos los dos pasos anteriores hasta completar la animación.
- Escogemos de nuevo la opción "stepForward" para dar por finalizada la animación.
- Escogemos de nuevo la opción "StepForward" para dar por finalizada la animación.

Podemos combinar o escoger qué tipo de animación nos conviene para nuestro sistema multimedia.

4.2.6 Lenguaje Lingo

Lingo es un lenguaje de programación. Es una forma de hablar a la computadora proporcionando comandos y haciéndole preguntas. La clave para escribir aplicaciones más potentes con director de su lenguaje Lingo. En director, podemos hablar con el entorno del programa: con el guión, el reparto y el escenario. Lingo es lenguajes de programación parecido al Inglés, todos los comandos funciones y otras palabras claves en director son palabras, grupos de palabras o abreviaturas en Inglés. Lingo es un tipo de miembro del reparto lo que significa que cuando tenemos código Lingo para contrarrestar películas, se guardan miembros del reparto llamado escritorios. Los escritorios son miembros del reparto que contiene un fragmento del texto formado por código en lenguaje Lingo. Dichos escritos forman parte del reparto junto con los mapas de bits, sonidos y formas de nuestra película en director. En algunos casos, se pueden guardar estos escritos en el guión de la película. Lingo es un lenguaje de programación muy potente, de casi todos los casos que se pueden hacer en director, casi ninguna de ellas se pueden hacer sin utilizar Lingo.



Otro elemento clave en Lingo y en todo lenguaje de programación es el uso de las variables. Las variables son zonas de almacenamiento para ciertos valores que utilizaremos posteriormente.

El lenguaje Lingo también tiene lo que llamamos rutina de tratamiento, es un conjunto de instrucciones en Lenguaje Lingo agrupadas para realizar una tarea específica. De este modo este conjunto de instrucciones puede ser llamado la cantidad de veces que sea necesario. Además, podemos nombrar a estos sucesos de director. A este conjunto de instrucciones les llamamos generalmente funciones.

El lenguaje Lingo también proporciona instrucciones para el control del flujo que se conoce como instrucciones de control. A continuación presentamos algunas de estas instrucciones de control:

La sentencia de control IF THEN ELSE no sirve para hacer una bifurcación en el código, es decir, se hace una pregunta que en lenguaje natural podría interpretarse como:

Si pasa algo entonces realizar cierta actividad sino realizar otra actividad.

Otra instrucción de control se llama CASE, ésta es útil cuando la decisión debemos otro y si sucede la opción se realizará una tercera acción y si no sucederá nada de lo anterior realizará una cuarta acción.

Tenemos también en lenguaje Lingo las instrucciones adecuadas para repetir una tarea varias veces. La instrucción que nos permite hacer esto es la instrucción REPEAT WHILE, esta instrucción le dice al lenguaje Lingo que repita ciertas acciones hasta que se cumpla alguna condición.

Una de las cosas más importantes del lenguaje Lingo es que permite navegar en la película proporcionando los comandos de navegación que nos ayudarán a decidir si la película que tenemos, avance, regrese, vaya un sitio específico, etc. La instrucción que nos permite hacer todo esto es la instrucción GO TO.

Están también los guiones que consisten en una serie de instrucciones escritas en lenguaje "lingo" que son ejecutadas cuando se interacciona con alguno de los personajes de la película o bien cuando se llega a un cuadro (frame) específico de la película.



Estas instrucciones que contienen los “scripts” son órdenes sencillas que nos permiten: Intercambiar un “sprite” por otro, o modificar las propiedades de cierto “sprite”, o hacer que la película se dirija a otro “frame” que no es el que le sigue en orden consecutivo o hacer que se repita algún “frame”, S etc.

Existen cuatro tipos diferentes de “scripts” que se catalogan dependiendo del lugar donde son asignados:

- Guión Miembro del reparto: Es aquél que va asociado a un miembro del reparto y no se pueden separar, de este modo cada vez que se utiliza este actor en el reparto, el guión va implícito en él. Los personajes con guión asociado llevan un pequeño ícono en la esquina inferior izquierda de su muestra en la ventana de Cast.
- Guión de “Sprite”: El guión de este tipo está asociado a un “sprite” en particular, es decir, un miembro del reparto que han sido situado en la partitura. Este script se ejecuta en el momento en el que la película pasa por el “sprite” que contiene el guión. Son de gran utilidad cuando vamos a usar el mismo miembro del reparto en varios cuadros de la película y en cada cuadro tendrá una función diferente, esto no se lograra con los guiones de miembro de reparto, ya que en estos el guión viaja con el miembro del reparto a donde éste va.
- Guión de cuadro: Este se almacena en algún canal del guión y se ejecuta cuando la cabeza de reproducción pasa por él, sin Importar que haya o no “sprites”.
- Guión de Película: Este tipo de guión se ejecuta durante toda la película, Independientemente del recorrido de la cabeza lectora de la misma.

Veamos ahora algunos ejemplos de “sorites” utilizados en la realización de este sistema multimedia:

Este código se encarga de verificar que la respuesta a la pregunta echa sea correcta y la variable se coloca en 1 de lo contrario queda en cero.



```
on mouseUp me
global var
num="stdio.h"

if member("Campo_tiro").text=num then
sound(1).play(member("29ex"))
var=1
go to "frame49"
member("Campo_tiro1").text=" "
else
sound(1).play(member("30mal"))
var=0
go to "frame48"
end if
--go to "frame45"

end
```

El siguiente código activa el botón de “siguiente” para indicar así al usuario que puede continuar.

```
on mouseUp me
go to "frame18"
end
```

El código a continuación es el asociado con el botón de nombre “empezar”. Este botón tiene cierto comportamiento que se definió por este código. Cuando el Mouse pasa por encima del botón se ejecuta el código apropiado para que el botón cambie de forma y se escucha la música de fondo .

```
on mouseUp me
global Var_Nombre

if member("Campo_Nombre").text<>" " then
Var_Nombre=member("Campo_Nombre").text
go to "frame3"
else
go to the frame
end if
end
```



CAPITULO V “Pruebas”

5.1 Pruebas

Para obtener la versión final de este sistema multimedia, se realizaron pruebas en distintas etapas del desarrollo del mismo. Las pruebas que realizamos consistieron en pedir a 10 usuarios de edades entre 17 y 25 años que ejecutaran el sistema multimedia.

Se les pidió a estos usuarios que comentaran cualquier dificultad, por pequeña que esta fuera, que el sistema les presentara. Además se les pidió su opinión con respecto al contenido de la aplicación, aunque alguno de estos usuarios desconocían por completo el tema.

La retroalimentación obtenida de estos usuarios, fue parte importante para detallar el sistema y generar la versión final del mismo.

Las primeras etapas del desarrollo, los usuarios encontraron problemas de continuidad en el sistema. Por ejemplo: al realizar una acción que el diseño no tenía contemplada el sistema cayó en un estafo del cual no habrá forma de salir, a no ser que el proceso generado por la aplicación fuere detenido vía la tecla escape (Esc).

Para varios de los usuarios que probaron el sistema, la voz original del narrador no era agradable y además no resultaba con la claridad necesaria para que el usuario comprendiera el tema expuesto. De tal modo que fue necesario cambiar esta voz que resultará clara y comprensible.

En general los comentarios sobre los colores y seriedad de la “Interfase”, resultaron positivos ya que un 90% opinaron que el ambiente presentado es adecuado para el perfil para el cual fue diseñado.



5.2 Conclusiones

Siguiendo el Modelo en Espiral, fue posible identificar los elementos claves para la realización del sistema, ya que en cada interacción del modelo se fueron obteniendo progresos significativos los cuales nos llevaron al objetivo final que es el de un software útil para el proceso de la educación multimedia.

El tema de la enseñanza, el aprendizaje y el proceso de transmisión de cualquier conocimiento es sumamente complicado. Tratando de alcanzar los mejores resultados, los expertos en el tema han encontrado que la computadora es una gran ayuda, no sólo por la tecnología de punta, sino también por la intriga que genera para la gente el hecho de estar trabajando con computadora que para muchos, todavía son un reto.

Con la ayuda de los paquetes de software para el manejo de video, audio, Imágenes, texto y programación multimedia, se logró desarrollar un sistema de aprendizaje multimedia, que colabore y sirva de apoyo al maestro en la enseñanza del alumno de lo que son los métodos de ordenamientos en lenguaje C.

Se desarrolla una "interfase" de usuario, muy sencilla e intuitiva con colores, texto, imágenes y audio solemne que le transmite al usuario del software el nivel educativo que se trata de alcanzar.

Con el desarrollo de este trabajo obtenemos un buen sistema de apoyo multimedia para la enseñanza de los métodos del ordenamiento en lenguaje C, debido a que esta basado en procesos de calidad e software y en proceso de la teoría de aprendizaje.

5.3 Perspectivas

Creo que el software será de gran utilidad para el docente que imparte los conocimientos sobre el tema sistema multimedia para la programación en lenguaje c. Le ayudara a que los alumnos tengan otra perspectiva sobre el tema y tengan una herramienta extra en su educación.

Para mejorar los logros con este sistema multimedia de enseñanza, se le podrían agregar funciones de actualización del contenido, sobre todo hablando de los ejercicios. Se podría anexar una base de datos para controlar y sacar estadísticas del uso y aprovechamiento de los alumnos sobre el sistema.

Con la ayuda de diseñadores gráficos, seria muy útil mejorar las animaciones, Imágenes, y texto presentados, con la finalidad de darle mayor vista al software.

Un punto importante podría ser, que el sistema tuviera conexión a “Internet” que le da al alumno. La posibilidad de consultar una mayor calidad de términos y acceder a las páginas de la facultad donde se tiene mucho material desarrollado por la plantilla docente.



BIBLIOGRAFIA

[1] Vera Erica, Carrillo Maya, Céron Carmen, (2002), Software Educativo Multimedia, México, Manual.

[2] Vaughan Tay (1994), Todo el Poder de Multimedia, (2 ed.), México, McGraw-Hill.

[3] Freter Harald, Paulissen Dirk, (1994), El gran libro de Multimedia, México, Marcombo.

[4] Tenenbaum Aarón, Langsam Yadidyah, Augenstein Moshe, (1993), Estructura de Datos en C, México, Prentice Hall.

[5] Galvis Panqueva Alvaro, (1994), Ingeniería de Software Educativo, Ediciones Uniandes.

[6] Rueda Francisco, (1992), La inteligencia artificial, sus principios básicos y sus aplicaciones educativas, Aparece en Memorias del Congreso Colombiano de Informática Educativa.

[7] <http://www.ei.uvigo.es/~jcmoreno/is/analisis/software.html>

[8] Burgos Danel, De-León Luz, (2000), Director 8.x/Lingo Práctico. Guía de Aprendizaje, España, Osborne McGraw Hill.

[9] http://www.cs.buap.mx/~provis/modelos/modelos_educativos.html