



FES Aragón



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN

**“INFORME DE ACTIVIDADES DEL DIPLOMADO ‘PLANEACIÓN Y
CONSTRUCCIÓN DE SITIOS WEB’”**

**Tesis en Modalidad de Seminarios y Cursos de Actualización
y Capacitación Profesional**

PARA OBTENER EL TÍTULO DE

Ingeniero en Computación

PRESENTA

Juan Manuel Martínez Reyes

Asesora, Ing. Silvia Vega Muytoy

MARZO 2009



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A mí Madre, familia, amiguitas y amiguitos.

Gracias por soportarme tantos años.

ÍNDICE

ÍNDICE	3
INTRODUCCIÓN	6
CAPITULO 1 INFORME DEL DIPLOMADO “PLANEACIÓN Y CONSTRUCCIÓN DE SITIOS WEB”	7
1.1 CONCEPTOS Y TECNOLOGÍAS QUE INTERACTÚAN EN LA CONSTRUCCIÓN E IMPLEMENTACIÓN DE SITIOS WEB.	7
1.2 USABILIDAD Y ACCESIBILIDAD	8
1.2.1 Accesibilidad	8
1.2.2 Usabilidad	11
1.3 FUNDAMENTOS DE DISEÑO GRÁFICO PARA LA CONSTRUCCIÓN DE SITIOS WEB	13
1.3.1 Diagramación	13
1.3.2 Imagen	14
1.3.3 Tipografía	15
1.3.4 Color	16
1.4 PLANEACIÓN DE SITIOS WEB	17
1.4.1 Equipo de Trabajo	18
1.4.2 Estructura de una Página Web	19
1.4.3 Publicando el Sitio Web	20
1.4.4 Mantenimiento y Actualización	21
1.5 DIGITALIZACIÓN DE MEDIOS	21
1.5.1 Escáneres	21
1.5.2 Digitalización de Audio	22
1.5.3 Convirtiendo de Analógico a Digital	24
1.5.4 Digitalización de Video	26
1.5.5 Principales Formatos para el Web	30
1.6 CONSTRUCCIÓN DE INTERFACES PARA SITIOS WEB CON PHOTOSHOP CS2 Y FIREWORKS 8	30
1.6.1 Creación y Edición	31
1.6.2 Rollovers y Encabezados	34
1.6.3 Usando Fireworks 8	35
1.6.4 Creando Animaciones GIF	36
1.6.5 Creación y edición de Rollovers y Menús Desplegables	37
1.7 CRECIÓN DE PÁGINAS WEB CON XHTML Y CSS	37
1.7.1 XHTML	38
1.7.2 Atributos principales de las etiquetas	39

1.7.3	Etiquetas	40
1.7.4	Hojas de Estilo (CSS)	42
1.7.5	Selectores	43
1.7.6	Propiedades para el Texto.	45
1.8	DESARROLLO DE ANIMACIONES PARA SITIOS WEB CON FLASH 8	47
1.8.1	Línea de Tiempo, Capas y Fotogramas	48
1.8.2	Creación y Edición de Animaciones	49
1.8.3	Símbolos, Instancias y Elementos de Biblioteca	51
1.8.4	Audio y Video	52
1.9	ACTIONSRIPT	52
1.9.1	Acciones	53
1.9.2	Comportamientos	54
1.9.3	Control de la línea de tiempo	56
1.9.4	Control de objetos	57
1.9.5	Componentes	58
1.10	INTEGRACIÓN Y MANTENIMIENTO DE SITIOS WEB CON DREAMWEAVER 8	59
1.10.1	Creación y Elementos Web	60
1.10.2	Formularios	63
1.10.3	Insertar Objetos Media	64
1.10.4	Plantillas	65
1.10.5	Publicación	65
1.11	JAVASCRIPT	66
1.11.1	Programando en JavaScript	67
1.11.2	Estructuras de Control	68
1.11.3	Clases y objetos	69
1.11.4	Objetos del navegador	70
1.11.5	Uso de JavaScript con Marcos Incorporados	71
1.12	USO DE BASES DE DATOS EN SITIOS WEB	72
1.12.1	MySQL	74
1.12.2	Crear una Base de Datos y Tablas	76
1.12.3	Inserción de Datos	77
1.12.4	Selección de Datos	77
1.12.5	Borrar Datos	78
1.13	PHP	79
1.13.1	Programando Scripts en PHP	79
1.13.2	Estructuras de Control	81
1.13.3	Matrices	82
1.13.4	Formularios	83
1.13.5	Conexión con MySQL	84

1.14	TEMAS ACTUALES PARA EL DESARROLLO DE SITIOS WEB	86
1.14.1	Características de los CMS	86
1.14.2	Drupal	87
CAPÍTULO 2 INFORME DE DESARROLLO DE PROYECTO PERSONAL		
“ENTER”		91
2.1	FASE DE PREPRODUCCIÓN	91
2.1.1	Definiendo el Sitio Web	91
2.1.2	Fases de Preproducción	92
2.1.3	Layout	94
2.2	FASE DE PRODUCCIÓN	94
2.2.1	Creación de Contenidos	94
2.2.2	Utilización de Tecnologías	101
2.3	FASE DE POSTPRODUCCIÓN	102
CONCLUSIONES		103
BIBLIOGRAFÍA		105

INTRODUCCIÓN

El presente trabajo es un resumen objetivo de las actividades llevadas a cabo durante el Diplomado de Planeación y Construcción de Sitios Web, así como también del desarrollo del proyecto personal que se realizó durante dicho diplomado. Cuya meta principal es conocer y llevar a la práctica los elementos tecnológicos, metodológicos y conceptuales necesarios para la planeación y construcción de sitios Web.

El avance tecnológico que se ha visto en la última década en temas como las tecnologías de la información, las telecomunicaciones e Internet, han acercado a un sector cada vez más grande de la población a la información y a los servicios de telecomunicaciones, en la actualidad existen muchísimas aplicaciones, sistemas de información y servicios basados en tecnologías Web, los cuales prácticamente seguirán creciendo e impactando en diversos ámbitos y actividades sociales y laborales de la gente.

Es por eso que la planeación, desarrollo e implantación actual de sitios web ha requerido el uso e integración de diversas tecnologías, herramientas y metodologías; además la conformación de equipos de trabajo multidisciplinarios, integrado principalmente por profesionales tanto de las áreas de tecnologías de la información y programación de sistemas, como de diseño y artes visuales, quienes en conjunto, han logrado el desarrollo de sitios web eficaces y eficientes.

Durante el diplomado se han obtenido conocimientos tanto teóricos como prácticos que permiten realizar una excelente planeación, construcción y mantenimiento de los sitios web. Por lo tanto puedo aplicar los conceptos de diseño gráfico, así como también conocer las bases para tener un buen manejo de usabilidad y accesibilidad, que facilitaran la navegación del sitio para distintos tipos de usuarios.

La metodología para la creación de contenidos y formatos para Web es imprescindible en la actualidad, mostrar a los usuarios una variedad de contenido actual y moderno, mantendrá una atención en los proyectos realizados.

Conociendo las herramientas adecuadas de programación de los diversos lenguajes que mencionare durante este trabajo, son de gran ayuda para construir sitios dinámicos que van más allá de las páginas y diseños planos que antes se solían realizar, creando un vínculo entre desarrollador web y usuario; mediante el desarrollo de animaciones, foros, libros de visitas, formularios, desarrollo de correos electrónicos, e información bien organizada en bases de datos.

Finalmente sólo me queda mencionar que la descripción en este trabajo se ha resumido a una parte teórica principalmente y un poco en descripción práctica, ya que sería demasiado extenso la descripción de todo lo visto en el diplomado.

CAPITULO 1 INFORME DEL DIPLOMADO “PLANEACIÓN Y CONSTRUCCIÓN DE SITIOS WEB”

1.1 CONCEPTOS Y TECNOLOGÍAS QUE INTERACTÚAN EN LA CONSTRUCCIÓN E IMPLEMENTACIÓN DE SITIOS WEB.

El objetivo de este primer módulo es identificar todos los elementos tecnológicos y de desarrollo para crear e implementar páginas o sitios web; así como sus servicios, evolución de los medios de transmisión, técnicas de desarrollo y todos los beneficios que ofrece la red.

En un principio se definieron los conceptos de Internet, WWW, y un poco de su historia. Internet se conoce como la red de redes, una gran interconexión de redes de computadoras, usando protocolos de comunicación como lo es el TCP/IP (Protocolo de Control de Transferencia / Protocolo de Internet) el más importante de todos los protocolos de comunicación que influye en internet, ya que enlaza las redes de computadoras y permite la comunicación entre éstas, el HTTP (Protocolo de transferencia de Hipertexto) que es indispensable para acceder a las páginas web, FTP (Protocolo de transferencia de archivos), SMTP (Protocolo de transferencia de correo simple) y POP (Post Office Protocolo) ambos usados para el envío y recepción de correos electrónicos.

La Internet se origina desde el desarrollo de la ARPANET, una red que se creó a finales de los años 60's entre tres universidades de Estados Unidos. El concepto de la WWW (World Wide Web) es uno de los muchos servicios que se sustentan a partir del Internet como lo es también el correo electrónico, los chats, las descargas, transferencias y compartición de archivos; por mencionar algunos de los muchos servicios que se ejecutan a diario en línea.

Se comprendieron los modelos y tecnologías en que están basadas las comunicaciones como es el modelo OSI (modelo de referencia de Interconexión de Sistemas Abiertos) desarrollado por la Organización Internacional para la Estandarización (ISO), que comprende 7 niveles de desarrollo para un sistema de comunicaciones: físico, enlace de datos, red, transporte, sesión, presentación y aplicación.

En general este módulo ha servido de introducción para conocer de una manera rápida los conceptos y el funcionamiento de las diversas tecnologías involucradas en el diseño de sitios web, así como también los diversos grupos, equipos de trabajo y analistas encargados de llevar a cabo las estructuras, evaluaciones y diseño general de los proyectos web.

1.2 USABILIDAD Y ACCESIBILIDAD

El tema de la accesibilidad y la usabilidad en el diseño de las páginas web puede resultar a veces muy ambiguo, y un tanto peleado entre sí al tratar de diseñar un sistema que sea de fácil comprensión para los usuarios y por lo tanto su usabilidad deba ser mayor; además de que su accesibilidad trate de ser universal, para que todos aquellos usuarios con capacidades diferentes pueden acceder a un sitio web sin problemas.

1.2.1 Accesibilidad

En la web, el contenido de este medio es accesible cuando puede ser usado por alguien con discapacidad. Esto implica proporcionar flexibilidad para acomodarse a las necesidades de cada usuario y a sus preferencias y/o limitaciones.

El Consorcio de World Wide Web (W3C) ha establecido ciertas pautas que garantizan que la mayoría de los usuarios puedan usar y entender la información en Internet.

Algunas de las discapacidades que son atendidas por estas pautas son:

- ❖ Usuarios ciegos.
- ❖ Usuarios con ceguera al color.
- ❖ Usuarios con visión débil que no pueden leer texto pequeño.
- ❖ Usuarios sordos.
- ❖ Usuarios con deficiencia auditiva.
- ❖ Usuario que usan dispositivos distintos al ratón.
- ❖ Usuarios con discapacidades de control motor.
- ❖ Epilepsia fotosensible.

Aunque la mayoría de estas discapacidades son desatendidas por la mayoría de los programadores y diseñadores web, se ha tratado de difundir más su implementación para el buen desarrollo de sitios en la red.

A su vez estas disciplinas o recomendaciones de diseño tanto en usabilidad y accesibilidad están siendo contempladas en los estándares internacionales o locales. Existen dos tipos de estándares:

- ❖ Estándares iure: Son generados por un comité con estatus legal y están avalados por un gobierno o institución, en este caso hay dos importantes la ISO (International Organization for Standardization) y el ANSI (American National Standards Institute).
- ❖ Estándares de Facto: Son estándares que nacen a partir de productos de la industria y el uso generalizado los convierte en referente.

El W3C es el organismo encargado de producir estándares para la Internet, creado en 1994, y cuyo propósito general es conducir y desarrollar el máximo potencial de la Web promocionando el uso de protocolos comunes y asegurando la evolución de la misma.

Dentro de este organismo existe la WAI (Iniciativa para la Accesibilidad de la WEB), es un rubro muy importante que persigue que los contenidos dentro de la Red sean cada vez más accesibles, se enfoca en cinco áreas diferentes:

- ❖ Tecnología
- ❖ Directrices
- ❖ Herramientas
- ❖ Formación y difusión
- ❖ Investigación y desarrollo

La WAI contiene tres herramientas para solucionar estas necesidades de accesibilidad.

Web Content Accessibility Guidelines (WCAG, Directrices de Accesibilidad para el Contenido Web)

Encargado de estudiar los escenarios usuales que pueden ocasionar problemas para los usuarios discapacitados. Por ejemplo: que las imágenes de un sitio no pueden ser totalmente accesibles para algunos usuarios.

Authoring Tool Accessibility Guidelines (ATAG, Directrices de Accesibilidad de la Herramienta de Autoría)

El objetivo general de este apartado es desarrollar las herramientas autorizadas sean accesibles para los autores, independientemente de su discapacidad, y que se produzca un contenido accesible por defecto.

User Agent Accessibility Guidelines (UAAG, Directrices de Accesibilidad del Agente Usuario)

La UAAG se enfoca en la navegación con teclado, opciones de configuración, las características del software, como pueden ser sintetizadores de voz, enfocadas a usuarios con discapacidades visuales, auditivas, físicas, cognitivas y neurológicas.

El desarrollo del diplomado se ha enfocado más en el seguimiento y la guía de la WCAG, que contiene 14 pautas para que el diseño de un sitio sea accesible:

1. Proporcione alternativas equivalentes para el contenido visual y auditivo
2. No se base sólo en el color
3. Utilice marcadores y hojas de estilo y hágalo apropiadamente.

4. Identifique el idioma usado.
5. Cree tablas que se transformen correctamente.
6. Asegúrese de que las páginas que incorporan nuevas tecnologías se transformen correctamente.
7. Asegure al usuario el control sobre los cambios de los contenidos tiempo-dependientes.
8. Asegure la accesibilidad directa de las interfaces de usuario incrustadas.
9. Diseñe para la independencia del dispositivo.
10. Utilice soluciones provisionales. Por ejemplo: texto en lugar de imagen.
11. Utilice las tecnologías y pautas W3C.
12. Proporcione información de contexto y orientación
13. Proporcione mecanismos claros de navegación.
14. Asegúrese de que los documentos sean claros y simples.

Cada pauta tiene asignado tres diferentes tipos de prioridad, con un rango de mayor importancia la prioridad 1, teniendo que satisfacer las necesidades de dicha prioridad o en caso contrario un grupo significativo de usuarios no podría comprender el sitio web.

Durante el transcurso del diplomado se usaron diversas herramientas para verificar la accesibilidad de algunos sitios populares, se hace una breve descripción de dichas herramientas:

- ❖ **TAW:** Es un software que sirve para evaluar de forma rápida y precisa un sitio, con la ventaja de que es automático.
- ❖ **A-prompt:** Es una herramienta especializada en identificar y corregir errores de accesibilidad, por ejemplo en la sintaxis del código, ayuda a disminuir los índices de deficiencia de accesibilidad.
- ❖ **JAWS:** Es el lector de pantallas por excelencia y más popular en el medio. Comprueba que los sitios sean comprensibles para los usuarios de visibilidad nula o débil.
- ❖ Además se hace uso de diversos navegadores para probar la viabilidad de los sitios en varias plataformas.
- ❖ Finalmente se realiza una revisión y corrección manual, de esta forma se asegura que el código sea correcto y no sólo “se engañe” a los verificadores automáticos.

TAW

Se instala la extensión de TAW en el navegador Firefox ya que es el único navegador que permite instalar el software de TAW, gracias a su gran flexibilidad. De esta forma se despliega una barra de herramientas en el navegador Firefox, que permitirá revisar automáticamente los errores de prioridad, y un resumen detallado de la forma en cómo solucionar estos casos.

A-Prompt

Este software también permite verificar un archivo de HTML, para revisar sus fallas en cuanto a accesibilidad, y corregirlas en base a las decisiones que se le indiquen, cabe destacar que para que este trabajo rinda verdaderos frutos se debe tener un buen conocimiento de las pautas que se mencionaron anteriormente, si no es posible que no se corrija ningún error en absoluto.

Emulación

Finalmente no hay mejor forma de revisar un sitio, que emulando los proyectos en diferentes ambientes de trabajo y visualización, por ejemplo: usando capacidades de hardware y software diferentes, tipografías, resoluciones de pantalla, lectores de pantalla; en caso de que el sitio sea visitado por personas de visibilidad débil o nula.

1.2.2 Usabilidad

La palabra usabilidad significa “facilidad de uso”. Dentro del diseño web, la usabilidad, implica observar a los usuarios que se interesan en las tareas y meditar entre el diseño y las necesidades de los usuarios finales, asegurando que éstos puedan conseguir los objetivos del producto, sin importar que sea.

Por otro lado Jakob Nielsen, pionero en la difusión de la usabilidad, sugiere que la usabilidad es un término multidimensional. Indica que un sistema usable debe poseer los siguientes atributos: Capacidad de aprendizaje, eficiencia en el uso, facilidad de memorizar, tolerante a errores y subjetivamente satisfactorio.

Dentro del diseño web, son diferentes disciplinas que han desarrollado el crecimiento de la usabilidad:

- ❖ Diseño de la Interfaz del Usuario: Este es el diseño de cualquier interfaz que el usuario observe ya sea desde su computadora o algún producto.
- ❖ Interacción Humano – Computadora: Esta disciplina radica en el estudio en como los humanos interactúan y responden a ciertos estímulos dentro del diseño o la interfaz.
- ❖ Diseño Gráfico de la Interfaz del Usuario: Corresponde al diseño específico para las interfaces gráficas del usuario conocidas como GUI.

Cualquier diseñador que se enfrente a la tarea de mejorar su usabilidad tiene que estudiar a fondo estas tres disciplinas y trabajar con ellas en combinación.

Así se obtienen tres principios básicos para la buena usabilidad:

- ❖ Facilidad de aprendizaje
- ❖ Flexibilidad
- ❖ Robustez

Beneficios de la usabilidad

- ❖ Reducción de los costos de aprendizaje.
- ❖ Disminución de los costos de asistencia y ayuda al usuario.
- ❖ Optimización de los costos de diseño, rediseño y mantenimiento de los sitios.
- ❖ Aumento de la tasa de conversión de visitantes a clientes del sitio web.
- ❖ Mejora la imagen y el prestigio del sitio web.
- ❖ Mejora la calidad de vida de los usuarios del sitio ya que reduce su estrés, incrementa la satisfacción y la productividad.

Son 10 puntos que menciona Nielsen para que un sitio web pueda ser usable:

1. Visibilidad del estado del sistema: Es una forma en que puede haber una retroalimentación con el usuario, por ejemplo: a través de una barra de estado.
2. Adecuación entre el sistema y el mundo real: El sistema debe contener el lenguaje que el usuario habla, tratando de evitar los tecnicismos propios del sistema, usando frases y números.
3. Control y libertad del usuario: Permite rehacer las cosas que el usuario haya cometido por error.
4. Consistencias y estándares: Usando ciertos lineamientos a los que el usuario es familiar.
5. Prevención de errores: Evitar los mensajes de error, es mejor informar al usuario con diversas opciones.
6. Reconocer mejor que recordar: Hacer visibles objetos, acciones y opciones, para que el usuario recuerde con facilidad las áreas o temas a los que se refieren dichas opciones.
7. Flexibilidad y eficiencia de uso: Utilizando los aceleradores o atajos que el usuario común realmente no conoce, pueden hacer que un usuario avanzado o uno común pueden recordar fácilmente, ajustando una respuesta para cada combinación.
8. Estética y diseño minimalista: Se recomienda mantener un criterio de información básica e importante para evitar redundancia o información innecesaria que compita con temas relevantes.
9. Ayudar a los usuarios a reconocer, diagnosticar y solucionar errores: Los mensajes de errores deben presentarse de una forma coherente y sin usar código que el usuario común no reconocerá, indicando con precisión el problema y sugerir una forma de solución.

10. Ayuda y documentación: Usando una buena documentación que sea clara, concisa y breve se puede llegar a tener un buen sistema que informe al usuario sobre distintas dudas o aspectos del sistema.

En conclusión la usabilidad, en el diseño web, es la disciplina que estudia la forma de diseñar sitios web para que los usuarios puedan interactuar con ellos de la forma más fácil, cómoda e intuitiva posible.

Algunas técnicas para la evaluación de la usabilidad son: la realización de cuestionarios en línea, test de usuarios, simulación cognitiva (predicción de los problemas de uso), entrevistas a usuarios (individual o en grupo), etc.

1.3 FUNDAMENTOS DE DISEÑO GRÁFICO PARA LA CONSTRUCCIÓN DE SITIOS WEB

En este módulo del diplomado se revisan algunos de los conceptos básicos para el diseño de sitios web, aunque puedan resultar muy simples son de gran importancia y elementales para la armonización en el proyecto.

Formato de la página

La resolución de pantalla se entiende como la cantidad de pixeles que se despliega en un monitor para visualizar algo, a mayor cantidad de pixeles será menor el tamaño de las imágenes conllevando una mejor nitidez y calidad de las imágenes.

Existen dos resoluciones que se usan con frecuencia para el diseño web: 800 x 600 y 1024 x 768, siendo la primera la que se considera como estándar, aunque ya comienza a ser reemplazada por la segunda, gracias al avance y comercialización de monitores que admiten mayores resoluciones.

Ahora bien, considerando que un sitio web será visualizado en navegadores que incluyen barras de herramientas tanto superiores, inferiores y laterales, se tendrá un área segura de visualización de 700 x 415 pixeles.

1.3.1 Diagramación

Esta se refiere a crear una composición que sea armoniosa, eficiente, funcional, práctica y estética a su vez, sin que ninguno de estos elementos estén peleados entre sí, aunque a veces pueda resultar complicado, se puede llegar a tener estos elementos trabajando en conjunto.

Existen dos tipos de composiciones en el diseño web:

- ❖ Composición Simétrica: Se entiende por simetría a la relación proporcional o de medidas entre el conjunto y partes que integran un todo, considerando una relación centrada entre los elementos de la página.
- ❖ Composición Asimétrica: Se le llama también informal, libre y dinámica; esta composición se basa en dar un equilibrio específico a un solo eje, pero de forma descentrada, así se concentra la atención del usuario hacia una sola dirección.

Usando la diagramación en los diseños de sitios web, para formar un esqueleto en que basar y colocar los objetos dentro del producto, se crean retículas con columnas y renglones, medidos con precisión.

1.3.2 Imagen

No es necesario enfatizar que las imágenes son de un alto impacto visual para el diseño, son de suma importancia si se quiere llamar la atención con el diseño. Pero es importante destacar la buena selección de las mismas, que vayan acorde con el sitio, en cuanto a tipografías, colores, marcos, etc.

Imagen Digital

La resolución promedio en que trabajan los monitores comunes es de 72 pixeles por pulgada, por esta situación es recomendable trabajar las imágenes digitales a esta resolución a mayor resolución el archivo tendrá mayor tamaño en almacenamiento.

Imagen Vectorial

Las imágenes vectoriales están basadas en vectores que siguen medidas y patrones matemáticos, y conservan sus características originales si son escaladas a diferentes tipos de tamaño.

Imagen Bitmap

Compuestas por pixeles que representan valores de color y forma individual y grupal. Existen varios tipos de imágenes que son utilizadas como estándar para el uso en diseño web, éstas son:

- ❖ JPEG (Joint Photographics Expert Group, Grupo de Expertos en Conjunto Fotográfico), con una variedad de 16 millones de colores diferentes, por lo tanto contienen una buena calidad, el archivo se puede comprimir, pero no contiene transparencias.

- ❖ GIF (Graphics Interchange Format, Formato de Intercambio Gráfico), es el formato más utilizado en la web, permite usar 256 colores y permite transparencias, y es usado para imágenes que contengan poco colorido.
- ❖ PNG (Portable Network Graphics, Gráficos de Red Portables) es el formato que maneja 16 millones de colores diferentes y 256 tipos diferentes de transparencia en imágenes.

1.3.3 Tipografía

La tipografía es un elemento muy importante, ya que jerarquiza los textos dependiendo de sus características, además de ser un elemento más para la estética del sitio. Existen 5 familias tipográficas diferentes:

- ❖ Romanas (Patines triangulares)
- ❖ Egipcias (Patines rectangulares)
- ❖ San Serif (sin patines)
- ❖ Script (caligráficas)
- ❖ Display (o de ornamento)

Además de que pueden ser agrupadas o justificadas de diversas maneras: centrada, banda izquierda, banda derecha, o de bloque. Pero siempre es importante elegir un buen tipo de letra para que sea totalmente visible y legible.

En un documento HTML existen 5 tipos de letras por default que son usadas por los navegadores:

- ❖ Arial, Helvética, sans-serif
- ❖ Times New Roman, Times Serif
- ❖ Courier New, Courier, Mono
- ❖ Georgia, Times New Roman, Times
- ❖ Verdana, Arial, Helvética, sans-serif

CCS (*Cascading Style Sheets, Hojas de Cascada de Estilos*)

Las hojas de estilo sirven para dar un formato de apariencia a documentos HTML, como puede ser su tipografía, bloques, bordes, cuadros, fondos, posición de ciertos elementos, etc.

Se pueden definir internamente en el código del documento HTML o guardarse externamente para hacerla llamar por uno o más documentos.

Texto como Imagen

Esta técnica no es muy viable pero es recomendable cuando se desea utilizar una tipografía específica como elemento de diseño estético, más cuando se trata de elementos como título, subtítulos, encabezados de texto y botones principalmente. A este texto se le puede agregar un sinfín de efectos para lograr un mayor impacto.

1.3.4 Color

El color forma parte fundamental en el diseño web, con el que se puede enfatizar, jerarquizar, diferenciar y destacar diversas áreas de contenido; además de producir un sensación psicológica, física o emotiva al usuario.

También se considera importante la combinación de dichos colores para que armonicen en el sitio web, en conjunto con la tipografía y el color de la misma, imágenes, fondos, y marcos u otros objetos como los botones, etc. Para generar un buen contraste entre los elementos y lograr una excelente legibilidad.

Color en la WEB

Se le conoce como profundidad de color a la cantidad de colores diferentes que puede tener un píxel, y que se despliegan simultáneamente en pantalla, a mayor profundidad de color va a ocupar más memoria el archivo de imagen.

Paleta Web 216: Se refiere a 216 colores que comparten en común las paletas de los sistemas Windows y Macintosh, más otros 40 adicionales, que varían de una plataforma a otra. También es conocida como browser-safe palette (paleta segura para navegadores) asegura que estos colores se mostraran sin ningún problema en navegadores o exploradores de diferentes plataformas. Los colores de esta paleta están basados en combinaciones matemáticas de RGB (Red Green Blue, Rojo Verde Azul).

Color en HTML

HTML permite agregar colores a diversos elementos dentro del código, estos colores son declarados por medio de valores hexadecimales. Cada componente se extiende del *00* al *FF* en Hexadecimal, correspondiendo al *0* a *255* en los valores decimales. Siendo el valor *00 00 00* Negro (*0 0 0* Negro en RGB) y *FF FF FF* Blanco (*255 255 255* Blanco en RGB).

A partir de estos conceptos básicos de Diseño Gráfico, se toman en cuenta las siguientes recomendaciones para los proyectos:

1. Identificar formato y utilizar área segura de pantalla
2. Utilizar algún tipo de composición.

3. Tipografía acorde con el tema y concepto del sitio.
4. Colores que contrasten, sean legibles, coherentes con el diseño y concepto del sitio.
5. El diseño gráfico general del sitio debe reflejar y cumplir con los objetivos del sitio WEB.

1.4 PLANEACIÓN DE SITIOS WEB

En este módulo se revisan los modelos y técnicas en las que se basa la planeación de sitios web, desde plantear el tipo de página a realizar, tipos de usuarios que visitaran dicha página, esquemas y borradores del sitio, técnicas de medición de tiempo para el desarrollo del proyecto, mantenimiento, etc.

Planear, implica decidir, en el presente, las acciones que habrán de ejecutarse en el futuro. La planeación puede definirse como un proceso anticipado de asignación de recursos (personas, bienes, dinero y tiempo) para el logro de fines determinados.

Para planear un sitio WEB se debe formular algunas preguntas básicas, en base a las respuestas se determina el curso de acciones indicado para el desarrollo del proyecto.

- ❖ ¿Para qué necesitamos el sitio Web?
- ❖ ¿Para qué necesita el sitio Web el público?
- ❖ ¿Qué es lo que buscan las personas en mi empresa/entidad?
- ❖ ¿Qué información de la empresa/entidad es útil para el público?
- ❖ ¿Qué imagen de la empresa/entidad quiero proyectar en mis usuarios?
- ❖ ¿Qué servicios puedo proporcionar a través del sitio?
- ❖ ¿Cómo puedo facilitar la adquisición de mis productos y servicios con el sitio Web?
- ❖ ¿Quiénes están involucrados en la realización del proyecto?

A partir de estas preguntas se identifica el objetivo del sitio web, el público objetivo al que está orientado el trabajo, la tecnología necesaria para crear y mantener en línea el sitio, y el tema principal del sitio web.

Definiendo estos parámetros se tendrá que hacer una evaluación del costo y mantenimiento que tendrá el sitio, con el fin de hacer un considerable estudio minucioso del desarrollo del sitio, hay que tener en cuenta los servicios de desarrollo, todo el personal que interviene en la realización del trabajo, así se asignará una cotización o presupuesto al cliente para su aprobación.

1.4.1 Equipo de Trabajo

Es necesario tener los recursos humanos suficientes y específicos para cada área de trabajo indispensable, y que de tal forma formen en conjunto un equipo de trabajo exitoso para la elaboración del proyecto, de aquí se obtienen los miembros de este equipo:

Líder de proyecto: Es el encargado de coordinar y plantear los requerimientos que se utilizarán para el diseño del sitio web. Es el principal encargado del sitio web, llevará un control de fechas de entrega, con el fin de prever errores o modificaciones, además está encargado de informar al cliente o superiores del avance del trabajo.

Diseñadores Gráficos: Encargado de hacer los bocetos, layouts, propuestas de diseño, desarrollo de plantillas e integración del sitio en código HTML; trabaja en conjunto con los programadores y los gestores de contenido.

Programadores: Encargado de generar los algoritmos matemáticos que requiera el sitio, así como funciones matemáticas, bases de datos y subsistemas del proyecto.

Gestores de Contenido: Son los encargados de escoger el contenido necesario que se publicará en el sitio, analizan y estudian la información proporcionada por el usuario, así escogemos los tópicos del sitio, la estructura y los niveles de navegación.

Web Master: Es el encargado de todo el sitio web, ya que es quien tiene todos los códigos o passwords para hacer modificaciones en el sitio.

Finalmente se realizará una investigación detallada para definir los contenidos que se incluirán en el sitio web, identificando contenidos, los tipos de usuario, etc. Una vez definidos los tópicos, se procede a definir la estructura del sitio, en la que se identificarán las secciones, funcionalidades y sistemas de navegación que lo integrarán.

Árbol de Contenido

Se debe intentar que sean las secciones menos posibles, así se enfocará la atención del usuario en pocas áreas, se recomienda que las secciones sean entre 5 y 7 (esto puede variar dependiendo del tamaño del sitio). Los niveles de contenido no deben sobrepasar más de tres, para que el usuario tenga facilidad de encontrar lo que busca.

Sistema de Navegación

Ya establecidos los contenidos, niveles y secciones, se da paso a desarrollar el sistema de navegación que será la forma en cómo los usuarios ingresarán y navegaran dentro de las diversas áreas del sitio. Así se obtiene:

- ❖ Menú General: Debe estar presente en todo el sitio, ya que permitirá siempre el acceso a todas las áreas del sitio.
- ❖ Pie de Página: Aquí puede variar un poco la información, pero generalmente contiene, el nombre de la entidad, teléfonos, dirección física, correo electrónico y el copyright del sitio.
- ❖ Rutas de acceso: Es un listado que aparece en la parte superior de cada página y que muestra el trazado de páginas que hay entre la página principal del sitio hasta la que se encuentra actualmente. Cada lista debe tener una liga a las páginas dependientes de las mismas.
- ❖ Botón Home, Mapa de Sitio, Contacto, etc: Botones de acceso a diversas partes del contenido, pueden variar dependiendo de layout, o el diseño de sitio.
- ❖ Buscador: En caso de que el sitio lo requiera, así el usuario puede encontrar más fácilmente lo que busca.

1.4.2 Estructura de una Página Web

Diseñando un layout, que es un dibujo lineal que describe los componentes de cada una de las pantallas del sitio, se puede verificar cada uno de los elementos y donde estará colocado.

Otra herramienta de utilidad son los bocetos de diseño, que es la creación de dibujos digitales acabados en la forma que tendrá la página principal, siguiendo los lineamientos designados de colores, layout, y elementos de contenido.

Una vez escogido de entre los diversos bocetos de diseño, se hace un borrador de página que es un diseño del boceto seleccionado que ya contendrá código HTML, esto se hace con el objetivo de hacer una prueba de cómo funcionara el sitio usando ligas reales.

Finalmente sólo queda generar las páginas del sitio usando HTML e integrarlas. Aquí todo el equipo de desarrollo entra en acción para generar todos los contenidos, digitalización de imágenes, diseño gráfico, programación, etc.

Metodologías de Seguimiento

Para tener un excelente control de los tiempos de desarrollo y tiempos de entrega es necesario utilizar alguna herramienta para estos casos; aquí se usa la gráfica de Gantt y las minutas, reuniones de control y reportes de avance; la gráfica de Gantt se compone de una lista donde se anotan las actividades del proyecto de principio a fin, se indica el tiempo de duración estimada de dicha actividad, y las actividades dependientes.

1.4.3 Publicando el Sitio Web

Antes de publicar el sitio hay que evaluar que el proyecto cumpla con los requerimientos que el cliente pidió, así como realizar una serie de pruebas de validación con tal de que el sitio no contenga errores, como es la verificación de contenidos, la validación de los códigos en HTML, verificación de vínculos rotos, consistencia en la diagramación, validación de los CSS, comprobar la resolución para la cual se diseñó el sitio, y verificarla en diversos navegadores, validación de los formularios (en caso de que se hayan incluido), pruebas de seguridad como puede ser: protección interna del sitio, protección contra robots, manejo de privacidad, etc. Y finalmente se hacen pruebas de carga, que se refiere a la capacidad máxima que tiene un servidor Web, para atender a un conjunto de usuario de manera simultánea.

Entrega de documentación

Según lo estipule el convenio con el cliente, los desarrolladores web pueden o no entregar la documentación del sitio, aunque generalmente así se hace, y esto puede ser: el código fuente, que se refiere a todos los programas y aplicaciones construidas; elementos gráficos, todas las imágenes que se utilizaron para la construcción del sitio; capacitación, que es una demostración para que el usuario sepa cómo funcionan de forma general el sitio.

1. Sitio Web funcionando (dependiendo del hosting)
2. Login y password de la cuenta.
3. Código fuente de todas las aplicaciones construidas (previo acuerdo).
4. Documentación completa de los elementos del Sitio Web.
5. Elementos gráficos del Sitio Web, originales (cuando sea necesario).
6. Capacitación en los elementos necesarios para utilizar en el sitio.
7. Manual de usuario del sistema o sistemas desarrollados.

Seguimiento

El seguimiento se refiere a estar pendiente del funcionamiento del sitio web, hay diversas formas de checarlo:

- ❖ Hits: Hit es una medida que se registra en un archivo cada vez que el servidor Web presenta un elemento a quien lo solicita.
- ❖ Páginas vistas o visitadas: Se refiere a la cantidad de páginas que han sido solicitadas dentro del sitio.
- ❖ Sesiones de usuario: es el número de personas que ha visitado el sitio web.
- ❖ Archivo Log: Dentro del servidor Web donde está alojado el sitio se genera un archivo de texto donde se registra cada operación que ha sido realizada por un usuario en el sitio web.

- ❖ Presencia del sitio en buscadores: se verifica la presencia del sitio a partir de los buscadores más populares, ya que estos concentran la mayor parte del tráfico.
- ❖ Buscar por nombre
- ❖ Enlaces en otros sitios

1.4.4 Mantenimiento y Actualización

Cabe destacar esta última área de suma importancia, ya que mediante la renovación de contenidos y actualizaciones se mantendrá al público objetivo en contacto directo y con su debida atención en el proyecto. Mediante el mantenimiento se podrán corregir errores o prevenirlos, enfocándose principalmente en tres áreas: contenidos, diseño y programación.

1.5 DIGITALIZACIÓN DE MEDIOS

El propósito de este módulo fue proporcionarnos los conocimientos y herramientas suficientes para comprender los métodos para digitalizar imágenes, video y audio, e incluirlos en los proyectos, fue un módulo teórico práctico.

Dependiendo del tipo de material que se desee digitalizar, se escogerá el tipo de dispositivo requerido y el método para lograrlo, por ejemplo: para digitalizar imágenes se requiere del scanner, para la digitalización de audio se requiere de una tarjeta de sonido que pueda recibir el audio, por ejemplo un micrófono. Para la digitalización del video se requiere de una tarjeta de captura de video o un dispositivo externo de captura.

Cabe mencionar que para realizar este tipo de trabajo, de captura y edición de video, se requiere de un buen equipo de cómputo que ayudará a agilizar el proceso y completarlo de una forma rápida y eficaz.

1.5.1 Escáneres

El scanner es un periférico diseñado para registrar caracteres escritos o gráficos en forma de fotografías o dibujos, impresos en una hoja de papel, de esta forma se facilita su introducción en la computadora convirtiéndolos en información binaria que será comprensible para ésta. En el scanner la imagen se rastrea mediante un elemento semiconductor sensible a la luz (LDR). La señal emitida por estos semiconductores se traduce a la computadora mediante una secuencia de bytes, de esta forma es almacenada en la memoria de la PC. Para poder identificar los colores utilizan sensores con filtros coloreados por los colores primarios del modelo RGB (red, Green y Blue) rojo, verde y azul. De esta forma, un determinado pixel puede tener un 100% color rojo, un 30% verde y 0% de azul, de esta combinación sale un pixel de color naranja oscuro.

El reconocimiento de Caracteres

Existe también un programa de reconocimiento óptico de caracteres (OCR) que permiten convertir palabras impresas en texto editable en un programa procesador de textos. La base de este tipo de tratamiento es suponer que la computadora lee la imagen que contiene texto y reconozca las letras en él para poder ser editadas. De tal forma que el programa lee esas imágenes digitales y busca conjuntos de puntos que se asemejen a letras, a caracteres. Dependiendo de la complejidad de dicho programa entenderá de una manera aproximada los tipos de letra, llegando incluso a interpretar de una manera errónea los caracteres, mantener el formato original (columnas, fotos entre el texto) o a aplicar reglas gramaticales para aumentar la exactitud del texto) o aplicar reglas gramaticales para aumentar la exactitud del texto.

La resolución de un Scanner

El scanner está formado por un conjunto de detectores luminosos que se mueven a lo largo de la imagen. La distancia entre estos detectores luminosos determina la resolución horizontal del scanner, este valor se mide en dpi, y de ellos dependerá la calidad de la imagen. La mayoría de los escáneres modernos alcanzan una resolución de 800 hasta 2,400 dpi. Sin embargo, estos datos no siempre son fiables. Muchos fabricantes se sirven de un truco para crear una mayor resolución llamado interpolación. La interpolación consiste en crear artificialmente un punto suplementario entre cada dos puntos atribuyéndole como valor la media de los puntos de ambos lados, esto a nivel software. Lo que se consigue con esto es suavizar la imagen aumentando la resolución aparente.

Formatos de imagen

Finalmente se debe elegir el formato de la imagen a guardar para visualizarlo en web o para su uso y tratamiento en algún programa de edición de imagen. Como anteriormente se mencionó los principales son el GIF, JPG, PNG, etc; cada uno con sus características especiales.

1.5.2 Digitalización de Audio

Una señal de audio analógica es la versión electrónica del surco de un disco. Un micrófono convierte las variaciones de presión del aire de las ondas de sonido en un voltaje variable denominado señal de audio. El audio digital es la representación del sonido en una secuencia de números que se corresponde al nivel de una señal a un nivel predeterminado

Muestreo

Para convertir una señal analógica en digital, el voltaje se mide a intervalos regulares y se le asigna un valor numérico a través de un convertidor de analógico a digital. El muestreo se realiza miles de veces por segundo, el valor de cada muestra se redondea al número entero más próximo, o entero, y se convierte a binario.

Velocidad de muestreo y resolución

Esta velocidad de muestreo hace referencia a las veces por minuto que se mide el nivel de la misma. El audio de un CD tiene una velocidad de 44,100 veces por segundo (44.1 KHz). Las cintas de audio digital admiten velocidades de muestreo de 33, 44.1 y 48 KHz. La velocidad de muestreo debe ser al menos el doble de alta que cualquier frecuencia que se produzca. La resolución de una señal digital es el número de valores enteros exclusivos disponibles para representar el nivel de voltaje de una señal analógica. La resolución se especifica mediante el número de bits utilizados para almacenar cada muestra. El número de bits determina el intervalo de valores que se pueden asignar a cada muestra; cuando más bits se utilicen, mayor será el número que se puede representar.

El audio de un CD utiliza 16 bits por muestra (una resolución de 16 bits), lo que proporciona 65,536 (2 elevado a la potencia 16) valores enteros posibles. El término velocidad de bits hace referencia a la cantidad de bits que se utilizan por segundo para representar la señal. La velocidad de bits del audio digital se expresa en miles de bits por segundo (Kbps), y está directamente relacionada con el tamaño de archivo y la calidad del sonido. Las velocidades de bits más bajas generan archivos de menor tamaño pero de mala calidad; las velocidades de bits más altas devuelven una mayor calidad pero archivos de mayor tamaño.

Compresión

El audio digital en su forma básica no está comprimido, lo que significa que una tarjeta de sonido o un programa de reproducción pueden acceder directamente a la información. El principal inconveniente del audio sin comprimir es que ocupa mucho espacio. La mayor parte de los datos contiene información redundante que se puede expresar de forma más eficaz. Se puede sustituir los patrones más habituales que aparecen con mayor frecuencia en un archivo asignándoles códigos breves.

Se han desarrollado diversos CODECS de compresión sin pérdida específicamente para audio pero están limitados en una compresión máxima de 2 a 1. La compresión con pérdida funciona mediante la eliminación de la información redundante e innecesaria, la calidad de sonido percibida varía en función de factores como la velocidad de bits, la complejidad de la música y el tipo de software de codificación.

MP3 es una forma de compresión con pérdida que puede reducir el tamaño de un archivo de audio digital sin comprimir hasta la décima parte, al tipo que conserva un elevado nivel de calidad, la mayoría de los usuarios no apreciará diferencias de calidad. La utilización de mono en lugar de estéreo no ahorra demasiado espacio, ya que la información estéreo se puede comprimir con mayor eficacia que otros tipos de audio.

En algunas aplicaciones, la combinación de una menor velocidad de muestreo y de bits genera archivos más reducidos de calidad aceptable. Por ejemplo, la creación de un archivo MP3 de grabación de voz a 128 kbps y una velocidad de bits de 44,1 KHz, sería inútil; como la voz humana se puede reproducir con una menor velocidad de muestreo, bastaría con una velocidad de bits de 32 kbps y una velocidad de muestreo de 22,05 KHz, sin perder demasiada calidad.

Requisitos de Ancho de Banda

Los requisitos de ancho de banda de una señal digital son los mismos que su velocidad en bits, independientemente de que la señal este comprimida o no. Por ejemplo, para transmitir un archivo codificado a 128 kbps sin perder calidad, se necesitaría una conexión a Internet con una velocidad mínima de 128 kbps. Para transmitir audio sin comprimir con calidad de CD, se necesitaría una velocidad de conexión de al menos 1,411 Kbps.

1.5.3 Convirtiendo de Analógico a Digital

Si se desea realizar numerosas grabaciones, conviene hacerlo en un programa completo de grabación y edición como Sound Forge o Adobe Audition para Windows o Peak para Mac. Que permitirán ver una forma de onda, una representación gráfica del audio. Tras ello, se puede buscar visualmente problemas de grabación como, por ejemplo, niveles de señal bajos, acoples o exceso de silencio al inicio o al final.

Estos programas ofrecen funciones más avanzadas que los de los programas de grabación gratuitos o incorporados, sobre todo en lo que respecta a reducción de ruidos, edición de formas de onda, ecualización y aumento de la duración. Antes de comenzar a grabar, se tendrá que crear un nuevo archivo y especificar la velocidad de muestreo, la resolución y el número de canales.

En algunos programas de grabación se tendrá que seleccionar estos parámetros al crear el archivo, mientras que en otros establecen automáticamente en valores predeterminados a menos que se especifique lo contrario.

Editar la grabación

La mayoría de las grabaciones pueden editarse para ajustar el volumen y recortar el silencio de los extremos. Las grabaciones de fuentes analógicas también se benefician de la eliminación de ruidos. Utilizando un programa de grabación que incluye un editor de forma de onda, se puede reducir el zoom para ver la forma de onda completa. Buscar exceso de silencio, acoples o un nivel de señal bajo. Las grabaciones de audio desde fuentes analógicas son proclives a la presencia de ruido en sus señales. Los programas antes mencionados contienen también detectores de estas fuentes de ruido en las señales, eliminando parcialmente estas muestras.

Encabezados y metadatos

En muchos tipos de archivo utilizados para audio digital, la parte inicial del archivo contiene datos adicionales denominados encabezados, que definen la estructura de los datos de audio que vienen detrás. Cuando se abre el archivo, el programa primero lee el encabezado, que le indica cómo interpretar el resto de los datos.

Los metadatos, datos sobre los datos de audio, pueden aparecer por detrás del encabezado o encontrarse en otras partes del archivo, un ejemplo habitual de metadatos es la etiqueta ID3 que puede formar parte de un archivo MP3.

Formatos sin pérdida

Los formatos sin pérdida almacenan audio digital sin pérdida alguna información. Algunos como PCM, almacenan únicamente los datos de audio sin procesar y sin compresión, mientras que otros utilizan técnicas de compresión sin pérdida para crear archivos con la mitad de tamaño que los archivos PCM. Los archivos sin pérdida son la solución perfecta para archivar material que puede ser necesario modificar o volver a codificar más adelante.

Formatos con pérdida

Debido al aumento de la potencia de los equipos informáticos, se han empezado a utilizar para aplicaciones adicionales, que requieren mayor espacio de almacenamiento tanto para programas como para datos. Los formatos sin pérdida han ayudado en cierto modo, pero en el caso del audio, la compresión sin pérdida puede reducir los archivos sólo a la mitad de su tamaño original.

Formato AIFF, es un archivo de sonido típico de Macintosh y estaciones de trabajo Silicon Graphics. Los datos contenidos son un código de pulso basado en PCM, haciendo que los archivos sean más largos y usen menos compresión que un MP3.

Formato WAV, es un formato de archivo originario de Microsoft Windows 3.1, y es el formato para almacenar sonidos más utilizado por los usuarios de Windows, lo flexible de este formato lo hace muy usado para el tratamiento de sonido pues puede ser compresado y grabado en distintas calidades y tamaños. La ventaja más grande es la de su compatibilidad para convertirse en varios formatos por medio del software adecuado, un ejemplo de ellos es pasar de WAV a MP3.

Formato MP3, técnicamente es MPEG Audio Layer III, es un formato de compresión de audio digital, que comprime los archivos a una décima parte de su tamaño original aproximadamente, pero con una calidad cercana a la del CD.

La calidad de sonido del MP3 y su pequeño tamaño lo hecho muy popular en Internet, su algoritmo se basa en la forma de escuchar que tiene el oído humano, pues las frecuencias que quedan fuera de la audición no son registradas en el archivo (las mayores de 20 KHz y las menores de 20 hz), esto se traduce en archivos mucho más pequeños, sin una pérdida de la calidad del sonido.

El formato de Real Audio tiene un alto grado de compresión con tamaños de archivo más pequeños que MP3. Permite descargar archivos de canciones completas en un periodo de tiempo razonable. Dado que los archivos se pueden reproducir en flujo desde un servidor Web normal, los visitantes pueden comenzar a escuchar el sonido antes de que el archivo se haya descargado por completo. Los visitantes deberán descargar e instalar la aplicación auxiliar o plug-in RealPlayer para reproducir estos archivos.

El formato QuickTime es un formato de audio y video desarrollado por Apple Computer. QuickTime está incluido con los sistemas operativos Apple Macintosh, y lo utilizan la mayoría de las aplicaciones de Macintosh que emplean audio, video o animación.

El formato midi (Interfaz digital de instrumentos musicales) es un formato de música instrumental. Los archivos MIDI son compatibles con numerosos navegadores y no precisan ningún plug-in. Un archivo MIDI no se puede grabar y deben sintetizarse en una computadora con hardware y software especiales.

1.5.4 Digitalización de Video

La edición de video digital ha crecido gracias al paso que hubo de las antiguas cámaras de video analógicas, al nuevo sistema de grabación digital. Esto facilitó mucho la edición de video, ya que no había que realizar una conversión de analógico a digital, sino que el paso era de digital a digital con la ventaja de no haber pérdida de calidad en imagen y sonido. La facilidad de paso que hay a un sistema de edición antes de volcar el video a un formato definitivo de distribución, ésta es una de las razones por las que el éxito de la venta de videocámaras digitales ha crecido tanto, hasta llegar a ser el formato estándar.

Formatos de codificación de video

Formato PAL (Phase Alternating Line), es el nombre con el que se designa al sistema de codificación empleado en la transmisión de señales de televisión analógica en color en la mayor parte del mundo. El sistema de color PAL se usa habitualmente con un formato de video de 625 líneas por cuadro (un cuadro es una imagen completa, compuesta de dos campos entrelazados) y una tasa de refresco de pantalla de 25 cuadros por segundo. Algunos de los países donde se utiliza el formato PAL son España, Suecia, Grecia, China, Singapur, Angola, Nigeria y Nueva Zelanda.

Formato NTSC, es un sistema de codificación y transmisión de televisión analógica desarrollado en Estados Unidos en torno a 1940, y que se emplea en la actualidad en la mayor parte de América y Japón, entre otros países. El formato consiste en la transmisión de 30 cuadros de video por segundo y 525 líneas de resolución.

Formato SECAM (Sequentiel Couleur avec Memorie en francés o Color secuencial con memoria) es un sistema para la codificación de televisión en color analógica utilizado por primera vez en Francia. Es históricamente la primera norma de televisión en color europea. Este sistema trabaja con 819 líneas de resolución y dibuja las líneas en vertical y no en horizontal, como lo hacen PAL y NTSC. Otros países donde se utiliza son Hungría, Bulgaria, Mónaco, Polonia, Rusia y Ucrania.

Modos de Captura de Video Digital y Analógica

Se podría decir que se va a capturar el video de forma digital pero no es así ya que el video digital no se captura, se transfiere. Para realizar una transferencia se recomienda realizar un check list de lo siguiente antes de abrir el programa de captura:

Proteger la cinta contras grabación, esto porque algunas cámaras tienen la opción DV IN que podría superponer o borrar el material de la cinta original.

Revisar el tipo de cable, es necesario contar con el cable IEEE 1394 o FireWire o i.Link, que es un estándar multiplataforma para entra/salida de datos en serie a gran velocidad.

Tarjeta o puerto FireWire en la computadora, no hay problema si el puerto es de 4 ó 6 pines, ya que la captura y la edición es exactamente igual. Si la captura se realiza desde una cámara digital no tiene que haber pérdida de calidad, y lo que habrá en el disco duro una vez finalizado el proceso tiene que ser una copia exacta de lo que hay en la cámara de video.

Captura Analógica

El video se degrada con el tiempo. Al capturarlo en la computadora queda digitalizado, y por tanto el deterioro se detiene una vez que se tienen las imágenes en la computadora, pero mejorar la calidad de una señal deteriorada es en la práctica muy difícil. El Material analógico tiene, además, el potencial problema de la desincronización entre imagen y sonido. Esto no sucede siempre, y cuando parece tampoco hay una explicación clara, aunque normalmente se da en vídeos largos (de más de media hora de duración).

El Dazzle DVC, que significa Digital Video Creator, es un sencillo aparato que permite introducir en la computadora cualquier video procedente de diferentes formatos, Súper Video (S- Video) y video compuesto, a través de un puerto USB 2.0

Renderización

La palabra renderización es una adaptación al castellano del vocablo inglés rendering y que define un proceso de cálculo complejo desarrollado por una o varias computadoras destinado a generar un imagen 3D o video. Cuando se trabaja en un programa de diseño 3D o de edición de video por computadora, no es posible visualizar en tiempo real el acabado final deseado de una escena compleja ya que esto requiere una potencia de cálculo demasiado elevada. Por lo que se opta por crear el entorno con una forma de visualización más simple y técnica y luego generar el lento proceso de renderización para conseguir los resultados finales deseados.

Formato AVI (Audio Video Interleave) es el formato estándar de video digital, su funcionamiento es muy simple pues almacena la información por capas, guardando una capa de video seguida por una de audio. Cuando capturamos videos hacia nuestra computadora llega en formato AVI. Puede generar archivos muy grandes y de difícil manejo.

Formato WMV (Windows media Video) es una de las últimas propuestas de Microsoft que funciona con el Windows Media Player, ha tenido gran impulso debido al XP y que viene integrado en dicho sistema operativo. También tiene una opción para streaming que viene incluida en el Windows 2000 Server.

Formato MOV o QuickTime, desarrollado por Apple requieren de una player especial que es el QuickTime Player para visualizarlos. Ofrece dos alternativas de servidores Web. El Darwin Streaming Server y el QuickTime Server, ambos para plataformas Mac. Su códec es muy utilizado para presentar películas cortas, aunque su tamaño es más pesado que otros formatos.

Formato RM o Real Player, la compresión al formato RM permite definir qué cantidad de bits por segundo dedicamos al audio y cuantos al video. Real en los pasados años ha sido

muy utilizado para streaming de audio en diversos medios. También tiene una propuesta de video llamada Real Video, requiere de su propio player que el Real Player y para hacer streaming requiere del Real Server.

Formato MPEG-4, es un códec estándar internacional de video creado especialmente para la Web, es un algoritmo de compresión que codifica datos de audio y video optimizando su calidad de almacenamiento, codificación y distribución en redes. El estándar está diseñado para entregar video con calidad DVD (MPEG-2) en enlaces de baja velocidad y archivos de menor tamaño.

Formato FLV o Flash Video, es un archivo de video que contiene datos codificados de audio y video para enviarlos a través de Flash Player. Por ejemplo, si tuviera un archivo de video de QuickTime o de Windows Media, debería utilizar un codificador (como Flash 8 Video Encoder) para convertir el archivo de video en un archivo FLV.

Software para captura de Video

Pinnacle Studio: Este programa permite de una manera muy fácil editar video a partir de la cámara de vídeo digital o fuente de entrada analógica en función del hardware con que cuente el equipo de cómputo. Las principales características de Pinnacle Studio son:

- ❖ Traspaso del vídeo a la computadora.
- ❖ Selección de las mejores escenas.
- ❖ Ordenación y clasificación de los clips de video.
- ❖ Agregar efectos, títulos, música de CD o su propia voz.
- ❖ Grabación en DVD del proyecto final.

Estos procesos se pueden ejecutar de una forma muy sencilla desde que apareció el video digital.

Sony Vegas: Con esta aplicación de edición de video, se puede realizar la reconstrucción de eventos de audio y video en tiempo real con la ventaja que no se requiere render durante la edición. Al mismo tiempo, se puede detectar y reparar eventos de audio/video, cuando los eventos no están alineados, es decir, el usuario puede identificar problema de sincronización de un vistazo y corregir la alineación. Soporte de ambiente de multiprocesadores. Toma ventaja de arquitecturas de procesamiento paralelo, reduciendo significativamente los tiempos de render en varios procesadores.

Captura e importación de distintos formatos de medios

- ❖ Imágenes en BMP, GIF, JPG y PSD.
- ❖ Video en AVI, DV, SWF, MPEG1 Y MPEG 2.
- ❖ Audio en WAV, MP3, AIF, CD Y PCA.

- ❖ Extracción de audio de discos compactos.

Efectos y transiciones

- ❖ Contiene más de 185 transiciones editables para 2D y 3D
- ❖ Aproximadamente 300 efectos de video editables.
- ❖ Animación de textos, títulos y créditos en scroll.
- ❖ Edición multicámara.

1.5.5 Principales Formatos para el Web

La compresión tiene la ventaja añadida de exigir un menor ancho de banda para el transporte de los datos de video, por lo que se usa intensivamente en las emisiones vía Internet (el streaming). Lo ideal, dada la limitación actual del espacio disponible en los medios de grabación, es optar por un compromiso entre la compresión con pérdida, la calidad final del vídeo, y el espacio que ocupará el archivo. Cuanto mayor sea la compresión, más información se desechará del video original, lo que repercutirá en la calidad final de las imágenes obtenidas.

Para comprimir se emplean los llamados códecs, palabra que surge de los términos codificador-decodificador, y su finalidad es reducir información codificándola para que se pueda transmitir en menos tiempo, y que además ocupe menos espacio en el disco duro.

Hay dos métodos para codificar o ahorrar espacio; la compresión temporal donde se analiza cada fotograma o frame y se guarda la diferencia con el cuadro que lo precede, y la compresión espacial, que elimina los datos de los píxeles que no cambian de un fotograma a otro.

1.6 CONSTRUCCIÓN DE INTERFACES PARA SITIOS WEB CON PHOTOSHOP CS2 Y FIREWORKS 8

El objetivo de este módulo fue conocer y utilizar las funciones y comandos que ofrece el software Photoshop como una gran herramienta en el tratamiento, creación y modificación de imágenes a nivel de diseño gráfico de alta calidad visual.

En primer lugar aprendimos a reconocer el entorno de trabajo de la aplicación, como son las barras de propiedades, menús, herramientas, estado, y las agrupaciones de paletas; y cada uno de los comandos que contiene dichas barras y sus funciones.

- ❖ La barra de herramientas contiene diversos comandos que permiten seleccionar, pintar, modificar y visualizar imágenes.
- ❖ La barra de propiedades muestra los atributos de los objetos seleccionados.

- ❖ Los paneles de paletas agrupan diferentes aplicaciones que se pueden guardar o modificar para su rápido acceso, con la cualidad de que pueden ocultarse para no estorbar la visualización de los documentos.
- ❖ La barra de estado muestra las propiedades del documento, como dimensiones, peso y la herramienta que se esté usando en ese momento.

1.6.1 Creación y Edición

Imágenes

Es muy importante conocer que al obtener cierto tipo de imágenes conlleva algunos aspectos que se deben tomar en cuenta, como por ejemplo una pixelación (como comúnmente se le conoce a la distorsión de una imagen al verla en forma de cuadros).

Se debe definir correctamente el tamaño de imagen final en el que se va a trabajar. Además el origen de las imágenes puede venir de diferentes fuentes a considerar como fotografías digitalizadas, imágenes procedentes de videos, mediante escáneres, etc.

Photoshop puede guardar los archivos de imágenes en diferentes formatos, algunos de los cuales ya se han mencionado como son: el GIF, JPEG, BMP, y PNG. El formato nativo de Photoshop es el PSD, que guarda una copia íntegra de todos los elementos involucrados en el tratamiento de la imagen.

Las selecciones son zonas parciales de una imagen delimitadas, por su forma o color, para realizar sobre ellas todo tipo de modificaciones y transformaciones. Este tipo de selecciones están delimitadas por la forma en que se describen: marco, marco elíptico, marco de fila única, marco de columna única, lazo y lazo poligonal; existe una herramienta llamada Varita mágica o el comando de gama de colores, que también son grandes herramientas de selección específicas.

Por ejemplo la varita mágica permite seleccionar áreas de color de la imagen en función del valor de tolerancia específico. A mayor tolerancia, mayor número de píxeles seleccionados, pues el valor de tolerancia entre los píxeles adyacentes es mayor.

Transformar una selección permite aumentar o reducir la selección de manera precisa, según las características del área que se quiera obtener. La transformación libre es la más clásica, permite modificar la selección a lo alto y ancho individualmente de los valores al mismo tiempo.

Las selecciones también se pueden rotar, escalar, invertir, sesgar, distorsionar, o hacer perspectivas de ellas, rellenarlas de cierto color, recortarlas, suavizar, etc.

Realizar ajustes globales rápidos en una imagen

Una forma rápida de hacer correcciones y realizar ajustes de una forma global, es usando los comandos Brillo/Contraste, niveles automáticos, contraste automático, variaciones y color automático; que aunque no sean del todo exactos, pueden proporcionar un ahorro de tiempo considerable cuando se trata de modificar imágenes para que tengan una apariencia más natural.

- ❖ Los niveles permiten corregir la gama tonal y el equilibrio de color de una imagen ajustando los niveles de intensidad de las sombras, los medios tonos y las iluminaciones de la imagen.
- ❖ Los niveles automáticos regulan las variaciones de iluminaciones y sombras; definiendo los píxeles más claros y los más oscuros de cada canal de color como blanco y negro y redistribuyendo proporcionalmente los valores de los píxeles intermedios, ajustando cada canal de color individualmente, se pueden eliminar o introducir dominancias de color.
- ❖ El contraste automático ajusta la mezcla y el contraste global de los colores de una imagen RGB automáticamente. Asignando los píxeles más claros y más oscuros de la imagen al blanco y el negro, las iluminaciones aparecen más claras y las sombras más oscuras.
- ❖ Color automático ajusta el contraste y el color de una imagen, neutralizando los medios tonos y corta los píxeles blancos y negros.
- ❖ Brillo/contraste permite realizar ajustes sencillos en la gama tonal de una imagen, realiza el mismo ajuste en todos los píxeles de la imagen.
- ❖ Tono/saturación permite ajustar el tono, la saturación y la luminosidad de toda la imagen o de componentes individuales de una imagen.
- ❖ Variaciones ajusta el equilibrio de color, el contraste y la saturación de una imagen mostrando miniaturas alternativas.

Capas

Las capas son las diferentes secciones en las que se puede dividir un documento. El trabajo de capas se realiza independientemente una de otra, de tal forma que no se ven afectadas; esto permite modificar el objeto u objetos comprendidos en la capa sin afectar a los diferentes objetos contenidos en el resto de las capas.

De esta forma se facilita mucho trabajar los documentos, a partir de una selección se puede crear una capa, las capas se pueden mover simultáneamente, duplicar y copiar, así como cambiar su opacidad o una mayor transparencia. Según la opacidad establecida, los píxeles de una capa y otra se relacionarán de un modo específico, pudiendo producir efectos de veladuras y transparencias.

Además existen diversos modos de mezcla de color en las capas, como pueden ser fusionando un color base con un color de mezcla, aclarando u oscureciendo los colores dependiendo de si el color de mezcla es más claro o más oscuro que un gris neutro, multiplicando o dividiendo los colores según sea la intensidad del color de mezcla, aplicando el tono del color de mezcla sin variar la saturación y la luminosidad del color de base, modificando la saturación y el tono de los píxeles del color de base sin variar su luminosidad; por mencionar algunos, pero existen muchos más.

Así como existen los modos de mezcla de color, existen algunos estilos de capas específicos, que dan un efecto especial a las capas, por ejemplo: creando sombras internas entre la capa, produciendo un resplandor exterior o interior, superposición de color, trazo, etc. Es muy importante conocer estas herramientas ya que de ellas se lograrán obtener muchos efectos impresionantes para dar más detalle al trabajo.

Texto

En Photoshop también se puede generar texto y editarlo de tal forma que es posible escalarlo o modificarlo sin que pierda sus propiedades, ya que está basado en algoritmos matemáticos y vectores, de tal forma que no importa que tan grande se haga no resulta píxeleado, conservando sus bordes nítidos, independientemente de la resolución.

Al crear texto, una nueva capa de texto se añade a la paleta Capas. De esta forma se puede realizar varias de las transformaciones de capas antes mencionadas. Rasterizar un texto convierte la capa de texto en una capa normal y hace que su contenido sea imposible de modificar como texto.

El suavizado de un texto permite generarlo con bordes suaves rellenando parcialmente los píxeles del borde. Como consecuencia, los bordes del texto se fusionan con el fondo.

Deformar la capa de texto permite distorsionarlo para ajustarlo a una variedad de formas; por ejemplo, se puede deformar texto con la forma de un arco o de una onda. El estilo de deformación que selecciona es un atributo de la capa de texto, puede cambiar en cualquier momento el estilo de deformación de una capa y así cambiar la forma general de la deformación. También se puede aplicar un formato de párrafo de los más conocidos, definiendo opciones que se pueden aplicar a párrafos enteros, como alineación, sangría y espacio entre líneas de texto.

Mascarillas

Las máscaras permiten aislar y proteger áreas de una imagen conforme se aplican cambios de color, filtros al resto de la imagen. Al hacer selección en una parte de la imagen, la parte no seleccionada es un área con máscara o protegida frente a

modificaciones. Se usan también para modificaciones complejas de imágenes como la aplicación gradual de color o efectos de filtro a una imagen.

El modo de máscara rápida permite editar cualquier selección como máscara sin utilizar la paleta Canales y al tiempo que se ve la imagen. Así se tiene la ventaja de poder utilizar casi todas las herramientas o los filtros del programa para modificar la máscara.

Efectos y Filtros Especiales

Existen muchísimos efectos especiales que se pueden aplicar a las capas o selecciones en los proyectos, se seleccionará el adecuado dependiendo del mensaje que se desee comunicar, existen diversos tipos, por ejemplo: artísticos, de desenfoque, enfoque, texturas, trazos de pincel, pixeleado, etc.

Mapas de Imágenes

Image Ready es una herramienta extra integrada a Photoshop CS2 para generar los mapas de imágenes, que se trata de utilizar una misma imagen como espacio único para colocar varios hipervínculos, los cuales pueden enlazar con otras páginas web, direcciones de correo, etc.

Se pueden crear áreas de mapas de imagen, cada área está vinculada a una dirección URL, se pueden crear áreas circulares, poligonales o rectangulares, que sirvan como punto de inicio para crear los hipervínculos contenidos en la imagen.

GIF's Animados

El GIF animado es una secuencia de imágenes, o cuadros. Donde cada cuadro varía visiblemente del anterior, creando la ilusión de movimiento, cuando los cuadros se visualizan de forma rápida.

Para crear cuadros de animación, se usan las paletas Animación y Capas. Cada cuadro representará una configuración diferente en la paleta Capas. Estas animaciones pueden configurarse de diferentes formas, variando la rapidez de visualización, intercalando cuadros diferentes, y finalmente guardarlas para un formato de Web.

1.6.2 Rollovers y Encabezados

Rollovers

También con Image Ready se crean los rollovers, que es un botón o una imagen de una página web que cambia cuando el usuario mueve el puntero sobre él o hace clic en el mismo. Así puede cambiar el botón mismo o puede aparecer otro elemento de la página.

Al guardar una imagen optimizada, que contenga estado de rollover siempre se deberá especificar, en el tipo de exportación como HTML e imágenes, para que el programa genere el código resultante para este tipo de acciones. Image Ready añade código JavaScript al archivo HTML para especificar los estados del rollover, y cada estado de rollover se guarda como un archivo de imagen diferente.

Encabezados

El uso principal de los encabezados en el diseño web, es básicamente para proyectar la estética, en cuanto a diseño de gráficos, colores y formas se refiere, a propósito de identificar el sitio web. En ellos generalmente se contiene la identidad gráfica, logo o imagen de la empresa en específico o del proyecto que se realiza, como puede ser la tipografía, el color, y empleo de imágenes simbólicas de dicho evento.

La composición de los encabezados debe apoyarse mediante el contraste visual, de manera que pueda guiar al usuario visualmente de toda la información contenida en la página, es decir, donde se ubican las áreas de información y/o contenidos, el área de menús, la navegación, etc. Se debe proporcionar equilibrio dentro de los diversos elementos que componen el encabezado, porque permitirá balancear y asegurar la claridad, en toda la información presente en el sitio. Finalmente las dimensiones del encabezado dependerán del concepto gráfico, guardando la proporción de las demás áreas del proyecto.

Salvando Imágenes para el Web

Finalmente se aprendió a usar el cuadro de diálogo Guardar para Web, que se usa para optimizar y previsualizar ilustraciones optimizadas. Escogiendo así los formatos adecuados, dependiendo del uso de la imagen, y su peso específico, calidad, y tamaño. Ya sea si se usa como transparencia, diseño de fondo, una imagen del sitio, si contiene vínculos web, si se requiere mucha calidad, etc.

1.6.3 Usando Fireworks 8

Esta parte del módulo en específico fue un tanto desatendida, justificando este acto por el hecho de que varias de las herramientas usadas en Photoshop son similares a Fireworks, y se considero sólo manejarlo de una forma más superficial y ejemplificado especialmente para el diseño de rollovers, botones, menús desplegables, etc. Se explico de una forma rápida el entorno de trabajo de la aplicación, destacando sus menús, panel de herramientas, grupo de paneles, y el inspector de propiedades; en conjunto, cada uno de los comandos principales.

De igual forma que en el módulo anterior se comenzó a trabajar el tratamiento de imágenes exclusivas para web, ajustando sus valores y atributos, modificando las propiedades generales y estableciendo las pautas adecuadas para la visualización en los navegadores.

Se trabajó con las capas en Fireworks que son los componentes básicos de una ilustración, que se van dibujando en distintas hojas transparentes superpuestas, esto facilita muchísimo el trabajo y se necesitan muchísimas capas para realizar una composición compleja de una imagen, ajustando efectos, diseños, y dibujos.

1.6.4 Creando Animaciones GIF

Los GIF, como se mencionó anteriormente, es un formato especial de imagen usado con mayor frecuencia en la web. Para realizar una animación en formato GIF se crea una serie de imágenes que se puedan reproducir en secuencia, así se crea la ilusión de movimiento o representar algunos trucos de animación, como hacer aparecer o desaparecer objetos de una imagen.

Es importante mencionar que contiene las mismas limitantes de colores y sobras que el GIF original, o sea 256 colores, y se debe tomar en cuenta el número de gráficos que contenga la animación, considerando el tamaño del archivo y el tiempo de descarga del mismo.

Para hacer una animación en GIF en Fireworks se usa el panel de Fotogramas para controlar las animaciones. Se selecciona cada fotograma para así modificar la imagen en él y crear el efecto deseado. Además de esta aplicación, el Papel Cebolla también le permite trabajar con múltiples fotogramas a la vez. Finalmente se ajusta la duración de tiempo entre cada fotograma y el número de veces que se repetirá la animación (bucle).

Creación de Mapas de Imagen con Fireworks

En Fireworks también se trabaja con dos elementos importantes para web, como son las zonas interactivas, que son áreas especiales que añadidas a una imagen hacen que se pueda hacer clic en una región determinada cuando aparece en un navegador. Se puede agregar opciones que al dar clic en dicha área se dirija a una URL predefinida. A grandes rasgos son códigos especiales colocados sobre una imagen y que proporcionan una área de comportamientos en JavaScript.

La otra herramienta son las Divisiones, que son renglones a las que se les pueden adjuntar comportamientos. Sin embargo a diferencia de las zonas interactivas, las divisiones rompen una imagen en diferentes áreas, además sirve como extra para la exportación de los archivos y su optimización por separado.

Utilización de Botones

Con la herramienta de Editor de Botones, se podrá definir el aspecto que tendrá un botón cuando el usuario haga clic sobre él, y puede proporcionarle hasta cuatro aspectos distintos para un mismo botón.

Así se puede modificar cada estado:

- ❖ Arriba: Es el aspecto original del botón cuando se accede a la página.
- ❖ Sobre: Es el aspecto del botón cuando se coloca el cursor por encima del botón.
- ❖ Abajo: Es el aspecto del botón cuando el usuario ha pulsado el botón.
- ❖ Sobre y abajo. Cuando un botón aparece en estado Abajo, se puede cambiar su aspecto utilizando esta opción. Un botón que muestra su aspecto abajo puede cambiar otra vez cuando el usuario le pase el ratón por encima.

1.6.5 Creación y edición de Rollovers y Menús Desplegables

El rollover es uno de los efectos más comunes en la web, una imagen de la página se cambia por otra cuando el usuario pase el cursor por encima de un gráfico. Se usa la herramienta de división para crear el efecto de rollover, que cambia el aspecto de un objeto de un fotograma reemplazándolo por un objeto en un segundo fotograma. Ahora se pueden hacer cambios en los fotogramas subsecuentes como modificar el color de texto, efectos luminosos, sombras o la ubicación del objeto. Los últimos pasos para crear los rollovers incluyen insertar las divisiones y adjuntar el comportamiento. Se adjunta el comportamiento a la división y Fireworks se encarga de generar el código en JavaScript para que provoque el efecto.

Un menú emergente funciona como una barra de menús que va mostrando diversas opciones conforme se posicione el cursor y haciendo clic, lo cual llevara a diferentes URL. Aunque para desarrollar este tipo de menús se debe tener en cuenta que genera mucho código, y que puede disminuir la calidad de la navegación en el sitio web.

1.7 CRECIÓN DE PÁGINAS WEB CON XHTML Y CSS

En sus inicios las páginas web estaban hechas completamente en HTML (HyperText Markup Language), que se compone de dos herramientas preexistentes para su desarrollo: El concepto de hipertexto, conocido como link o ancla, el cual permite conectar dos o más elementos entre sí, de manera no secuencial; y el SGML (Standardized Generalized Markup Language), que sirve para colocar etiquetas o marcas en un documento que indica cómo debe verse la página web.

Las etiquetas y sus componentes

Las etiquetas son palabras o abreviaturas sencillas en inglés, que se encuentran contenidas entre paréntesis angulares, e indican al navegador como debe verse lo que contienen las mismas, con ayuda de los atributos que modifican el contenido de las etiquetas y la visualización en pantalla. La mayoría de las etiquetas tienen un inicio y un cierre que afectan al contenido que está en el interior de ellas, el inicio se declara con el nombre de la etiqueta dentro de los paréntesis angulares, más sus atributos si los tiene. Después va el contenido y el cierre que va precedido de una diagonal con el nombre de la etiqueta dentro de los paréntesis angulares.

```
<p class="texto">Esto es un texto</p>
```

1.7.1 XHTML

XHTML (Extensible HyperText Markup Language) es un lenguaje de marcas de hipertexto extensible, éste requiere que el código se escriba en el formato exacto y que todas las etiquetas se cierren y aniden correctamente. XHTML está basado completamente en HTML 4.01 y respeta las reglas de XML.

Diferencias de XHTML con HTML

1. Se tiene que hacer la declaración de DOCTYPE
2. Los nombres de las etiquetas y atributos deben estar en minúsculas
3. Los elementos XHTML deben estar anidados de manera coherente
4. Todos los atributos de las etiquetas en XHTML deben tener un valor y estar entre comillas
5. Todos los elementos de XHTML deben cerrarse

Estructura Básica de un documento XHTML

1. Una línea de código que contiene información sobre el tipo de documento de XHTML
2. Etiqueta de inicio de documento html <html>
3. Una sección de cabecera delimitada por la etiqueta <head> </head> Asimismo el elemento <title> debe ser el primer elemento dentro de <head>
4. Todo el contenido del documento, se especifica dentro de la etiqueta <body> </body>
5. Etiqueta de cierre del documento </html>

Caracteres especiales

Tanto en HTML y XHTML, existen caracteres especiales que no puede interpretar el navegador, con sólo escribirlos a través del teclado en el documento, sino que, necesita de cierta sintaxis para poder hacerlo. Por ejemplo si se quiere escribir Árbol la sintaxis

para escribir Á es `Á` o `Á`; en este caso se escribiría así `<p>Árbol</p>` o `<p>Árbol</p>`. En los caracteres especiales se encuentran las vocales mayúsculas y minúsculas acentuadas, la ñ, algunos signos especiales, etc.

La *sintaxis* ` ` indica un espacio sin retorno de línea (non-breaking space), es útil para cuando se quiere dar un espacio en blanco.

1.7.2 Atributos principales de las etiquetas

- ❖ **lang**: identifica el idioma en que está escrito el documento y mantiene compatibilidad con versiones anteriores
- ❖ **xml:lang**: identifica el idioma en que está escrito el documento
- ❖ **xmlns**: identifica un namespace que es una colección de nombres única y universalmente aplicable a Internet, a través de un URI
- ❖ **dir**: Controla la dirección del texto del contenido de la etiqueta (ltr y rtl)
- ❖ **profile**: Especifica un identificador uniforme de recursos del perfil de los metadatos
- ❖ **base**: Indica un URL principal, para todos los URL relativos al documento
- ❖ **link**: Define un vínculo entre el documento actual con otro dentro de la sección `<head>`
- ❖ **meta**: Proporciona información adicional al documento.
- ❖ **script**: Especifica un script dentro del documento
- ❖ **style**: Sirve para definir los estilos que puede tener un documento
- ❖ **title**: Especifica un título identificador para el documento y es indispensable que lo contenga la página Web
- ❖ **name**: Especifica un nombre para identificar el valor del atributo content
- ❖ **content**: Es el valor que representa el atributo name.
- ❖ **charset**: Especifica la codificación de caracteres utilizada en el documento destino
- ❖ **http-equiv**: Indica un nombre para el par nombre/valor y da instrucciones al servidor para incluir esos datos en la cabecera del documento.
- ❖ **scheme**: Define un formato para interpretar el valor de los atributos en el contenido.
- ❖ **alink**: Establece el color de los vínculos activos del documento
- ❖ **background**: Especifica la ruta de una imagen para que aparezca de fondo en la página.
- ❖ **bgcolor**: Especifica un color de fondo para el documento.
- ❖ **class**: Indica un nombre de una clase, para aplicarle un estilo a la etiqueta `<body>`
- ❖ **id**: Se usa para rotular la sección de un documento, para hacer referencia a hipervínculos, hojas de estilos y programas si se desea.
- ❖ **text**: Define un color para el texto del documento.
- ❖ **vlink**: Indica el color para las ligas que ya fueron visitadas.
- ❖ **cite**: Permite identificar la fuente de la cita.
- ❖ **border**: Indica el ancho del borde.
- ❖ **cellspacing**: Sirve para especificar un espacio entre celdas.

- ❖ **cellpadding**: Sirve para especificar un espacio entre celda y el contenido de la misma.
- ❖ **summary**: Sirve para colocar una breve descripción de la tabla.
- ❖ **width**: Especifica el ancho de la tabla.
- ❖ **accesskey**: Permite tener acceso a un elemento de un documento a través del teclado, con la combinación de teclas Alt o Ctrl dependiendo la plataforma, y la tecla definida.
- ❖ **href**: Se utiliza para especificar el URL de un hipervínculo.
- ❖ **tabindex**: Se utiliza para indicar un orden de navegación de los vínculos de un documento con la tecla Tab.
- ❖ **target**: Indica en donde se debe visualizar el contenido del hipervínculo.
- ❖ **alt**: Especifica el texto alternativo que aparece en lugar de la imagen en los navegadores.
- ❖ **height**: Define el alto de la imagen.
- ❖ **longdesc**: Permite asociar una descripción larga de una imagen a través de un archivo.
- ❖ **src**: Especifica la ruta del archivo de imagen.
- ❖ **width**: Define el ancho de la imagen.
- ❖ **cols**: Define el número de columnas
- ❖ **rows**: Indica el número de renglones.
- ❖ **frameborder**: Indica si aparece o no los bordes del frame (0 o 1)
- ❖ **noresize**: Define que el usuario no podrá modificar el tamaño del marco.
- ❖ **marginheight**: Indica la altura del margen con respecto a los bordes del marco.
- ❖ **marginwidth**: Indica el ancho del margen con respecto a los bordes del marco.
- ❖ **scrolling**: Sirve para indicar si se usan o no las barras de desplazamiento en el frame (yes, no o auto).

1.7.3 Etiquetas

- ❖ **<html>** Delimita un documento XHTML </html> Atributos: lang, xml:lang, xmlns, dir
- ❖ **<head>** Especifica la cabecera del documento XHTML </head> Atributos: dir, lang, profile, base, link, meta, script, style, title.
- ❖ **<meta />** Sirve para especificar información adicional del documento. Atributos: name, content, charset, http-equiv, scheme.
- ❖ **<body>** Delimita el inicio y el final del cuerpo del documento </body> Atributos: alink, background, bgcolor, class, dir, ir, link, style, text, title, vlink.

Elementos a nivel bloque

Los elementos a nivel bloque dan la estructura principal de los documentos, y pueden contener elementos del mismo tipo, elementos en línea y datos de caracteres. Los elementos a nivel bloque son: h1, p, ol, ul, table y div.

- ❖ **<h1>,<h2>,...,<h6>** Estas etiquetas denominadas de encabezado, sirven para dar jerarquía y diferenciar títulos, subtítulos y determinadas secciones del documento. Atributos: class, dir, id, style, title.
- ❖ **<p>** Sirve para definir un párrafo de texto </p> Atributos: class, dir, id, style, title.
- ❖ **<blockquote>** Define un bloque de cita y lo representa con un margen izquierdo y derecho </blockquote> Atributos: cite, class, dir, id, style, title.
- ❖ **<pre>** Determina un segmento en el que el navegador mostrará el texto exactamente con los espacios y tabuladores definidos en el documento </pre> Atributos: class, dir, id, style, title.
- ❖ **** Sirve para crear una lista numerada u ordenada alfabéticamente. Atributos: class, dir, id, style, title.
- ❖ **** Define un elemento dentro de una lista ordenada y sin ordenar Atributos: class, dir, id, style, title.
- ❖ **** Sirve para crear una lista que no es numerada ni lleva un orden Atributos: class, dir, id, style, title.
- ❖ **<table>** Sirve para definir una table </table> Atributos: border, class, cellspacing, cellpadding, dir, id, style, summary, title, width.
- ❖ **<caption>** Sirve para indicar un título o leyenda para la tabla, y la diferencia con el atributo title, es que el título de la etiqueta <caption> si se visualiza en pantalla </caption> Atributos: class, dir, id, style, title.
- ❖ **<tr>** Sirve para crear una fila dentro de la table </tr> Atributos: class, dir, id, style, title.
- ❖ **<th>** Sirve para identificar los datos de la table </th> Atributos: class, dir, id, style, title.
- ❖ **<td>** Sirve para colocar los datos de las celdas </td> Atributos: class, dir, id, style, title.
- ❖ **<dl>** Crea una lista de definición </dl> Atributos: class, dir, id, style, title.
- ❖ **<dt>** Indica un término de la lista de definición </dt> Atributos: class, id, dir, style, title.
- ❖ **<dd>** Especifica la explicación de un término de lista de definición. </dd> Atributos: class, dir, id, style, title.
- ❖ **<div>** Esta etiqueta permite crear bloques o secciones dentro de un documento para dar una estructura </div> Atributos: class, dir, id, style, title.

Elementos a nivel en línea

Los elementos a nivel en línea representan los detalles del documento, y no pueden contener elementos a nivel bloque, sólo pueden contener otros elementos a nivel en línea y datos de caracteres.

- ❖ **<a>** Sirve para crear conexiones o enlaces a cualquier tipo de archivo y dirección electrónica. Atributos: accesskey, charset, class, dir, id, href, name, style, tabindex, target, _blank, _parent, _self, _top, title.

- ❖ **<abbr>** Indica la abreviatura de una palabra o una frase más grande </abbr> Atributos: class, id, lang, style, title.
- ❖ **<acronym>** Indica que el texto es una abreviatura formada con la primera letra de cada letra de un nombre o una frase como IBM, HTML, o XML por mencionar algunos ejemplo. </acronym> Atributos: class, id, lang, style, title.
- ❖ **
** Añade un salto de línea en los párrafos. Atributos: class, clear, id, style, title.
- ❖ **** Indica al navegador que debe enfatizar al texto que se encuentra contenido en la etiqueta y lo hace mostrándolo en cursivas Atributos: class, dir, id, style, title.
- ❖ **** Sirve para indicar al navegador que apliqué mayor énfasis a un palabra o texto, y lo hace poniendo este en negritas Atributos: class, dir, id, style, title.
- ❖ **<i>** Sirve para mostrar el texto en cursivas </i> Atributos: class, dir, id, style, title.
- ❖ **** Pone en negritas una letra o texto, que se encuentre contenido dentro de esta etiqueta Atributos: class, dir, id, style, title.
- ❖ **** Delimita una parte del contenido de un texto para aplicarle un estilo Atributos: class, dir, id, style, title.
- ❖ **** Sirve para insertar una imagen dentro de un documento Atributos: alt, class, dir, id, height, longdesc, src, style, title, width.
- ❖ **<frameset>** Define una sección de frames o marcos en el documento </frameset> Atributos: border, class, cols, id, rows, style, title.
- ❖ **<frame>** Esta etiqueta solo puede aparecer dentro de la etiqueta <frameset>, y permite indicar un url para cada sección que se delimite con el frame o marco </frame> Atributos: class, frameborder, id, noresize, name, marginheight, marginwidth, scrolling, src, style, title.

1.7.4 Hojas de Estilo (CSS)

La otra parte con la que trabaja HTML y XHTML son las CSS (cascading style sheets) u hojas de estilo en cascada y sirven principalmente para controlar la apariencia visual de las páginas web. Los estilos se componen por un selector, propiedad y valor principalmente. La declaración del estilo se hace dentro de los símbolos de llaves {}.

Ejemplo:

```
h1 {
color: #000000;
font-size: 20px;
text-align:center
}
```

Los estilos de un documento se pueden aplicar de manera interna o externa. De manera interna, se pueden declarar de dos formas:

1. Al inicio del documento dentro de la etiqueta `<head></head>`
2. El estilo se declara dentro de las etiquetas, utilizando el atributo `style`, y entre comillas se indica el selector, propiedad y valor.

Dentro de la página o sitio web, los estilos externos hacen referencia a una hoja de estilo externa, que se declara y almacena afuera del sitio web. Y para vincularla al documento XHTML se puede hacer de las siguientes maneras:

1. Utilizando la etiqueta `<link>` en la sección de `<head>`, especificando el URL del lugar en el que se encuentra la hoja de estilo almacenada, como se muestra en el siguiente ejemplo:
2. `<link href="estilo.css" rel="stylesheet" type="text/css"/>`

Con importar se pueden combinar y aplicar en una misma página, estilos provenientes de distintos documentos. Si se importan varias hojas de estilo, éstas se aplican en el orden en que son importadas. Para importar una hoja de estilo se utiliza la instrucción `@import` dentro de la etiqueta `<style>`, como se muestra en el ejemplo:

```
<style type="text/css"> @import url ("estilo.css"); </style>
```

Ventajas de usar CSS

- ❖ Una de las ventajas es que cuando se cambia la hoja que se aplica a un sitio completo, la hoja cambia la apariencia de todas las páginas sin tener que estar modificando una por una.
- ❖ Con el uso de los CSS el código en el documento disminuye y ahorra espacio en archivo.
- ❖ Los CSS pueden aplicarse de varias maneras y combinarse formando una cascada de estilos con la información de cada una.
- ❖ CSS se puede usar con otros lenguajes de programación como JavaScript para tener mayor interacción y efectos dinámicos en las páginas.

1.7.5 Selectores

Los selectores, forman parte fundamental de los estilos y su principal función es indicar sobre que parte de un documento HTML o XHTML se debe aplicar el estilo, existen varios tipos de selectores:

Selector de Clase: consta de un punto, seguido por el nombre de la clase que se desea crear. Ejemplo: `.nombre`

Se declara en la hoja de estilo como:

```
.pregunta{font-weight:bold; font-style:italic}
```

En el código HTML y XHTML, se aplica

```
<p class="pregunta"> Este es un ejemplo </p>
```

Todos los párrafos de la clase "pregunta" aparecerán con texto en itálicas negritas para diferenciarlos de los párrafos normales.

Selector Tipo o Etiqueta: son los que identifican a un tipo de elemento dentro de los que conforman el código HTML y XHTML. Es decir, todas las etiquetas pero sin los paréntesis angulares <>, por ejemplo: h1 selecciona todos los elementos <h1> de la página y la siguiente regla identifica a los elementos <h1> de la página y los alinea centralmente: *h1{text-align:center}*. Los selectores de tipo tienen la ventaja de que con un simple cambio en la hoja de estilo pueden modificar el aspecto de todos los elementos de ese tipo en todas las páginas del sitio.

Selector Avanzado ID: Los selectores de ID funcionan de manera similar al de clase, pero sólo pueden aplicarse a un elemento de la página. Quiere decir que si hay un elemento que tiene asignado el atributo *id="principal"* no podrá haber otro ID con igual valor. La sintaxis en la hoja de estilo es similar a la de clase, solamente que en lugar un punto se usa el numeral #, como en el siguiente ejemplo: *#titulo 1 h1{text-align:center}; #uno {font-style: italic}*. En el primer caso, aquel elemento h1 con un *id="titulo1"*, indica que el texto será centrado. En el segundo, cualquier elemento que tenga asignado el atributo *id="uno"* irá en itálicas.

Selector Universal: se identifica con un asterisco * y sirve para representar a cualquier elemento de la página, y es útil cuando se desea que las páginas mantengan determinadas características de manera general, por ejemplo: ** { text-align:center; }* Este código se está indicando que todo el texto que se escriba, tenga una alineación centrada.

Pseudo-Clases

Se refieren a los estados de un elemento y se caracterizan porque llevan dos puntos después de éste y tienen nombres predeterminados.

:link y *:visited* se conocen como pseudo clases de vínculo y solamente se pueden aplicar a elementos ancla *:hover*, *:active* y *:focus* se conocen como pseudo clases dinámicas y se pueden aplicar a cualquier elemento.

Pseudo elementos

Sirven para que algunas partes tengan un formato específico.

- ❖ **:first-line:** Permite aplicar un estilo determinado a la primera línea de un párrafo
- ❖ **:first-letter:** selecciona la primera letra de un párrafo para aplicarle un estilo.
- ❖ **:before** y **:after:** sirven para indicar un estilo antes o después de un elemento determinado.

Cascada: Cuando existe más de una hoja de estilo en un documento, el navegador dará prioridad descendente a través de los siguientes estilos: autor, usuario y navegador.

Herencia: La herencia se da por los ascendentes y descendentes. Es como un diagrama de navegación, por ejemplo, si *body* se le asigna un color de texto, elementos como `<p>` que se encuentren dentro de *body* tendrán el mismo color.

Medidas

Usando medidas se facilita mucho la visibilidad de los documentos HTML o XHTML en pantalla, ya sea usando medidas absolutas o relativas:

Absolutas: Indican valores exactos y no guardan ninguna relación con otra propiedad de medida. Son: in (pulgadas), cm (centímetros), mm (milímetros), pt (puntos), pc (picas 1 pic = 12 puntos) y px (pixel).

Relativas: Los valores relativos indican una medida en relación a otra propiedad de medida. Son em (equivale al ancho de las letras mayúsculas de la fuente relevante), ex (equivale a la altura de la letra minúscula de la fuente relevante), y el porcentaje % (1% = 6 px).

1.7.6 Propiedades para el Texto.

- ❖ **font-family:** Sirve para indicar la tipografía del texto. (Arial, Helvetica, sans-serif, Georgia, Times New Roman, Times, Serif).
- ❖ **font-size:** Especifica el tamaño de letras.
- ❖ **font-size-adjust:** Ajusta la relación del aspecto de la fuente para su mejor visualización.
- ❖ **font-stretch:** Especifica que los caracteres de las palabras se condensan o expandan.
- ❖ **font-style:** Se usa para dar una apariencia al texto inclinada.
- ❖ **font-variant:** Permite seleccionar una variante de la fuente seleccionada.
- ❖ **font-weight:** Permite poner el texto en negritas.
- ❖ **letter-spacing:** Pone un espacio adicional entre las letras.

- ❖ **line-height**: Sirve para especificar un espacio entre líneas de texto.
- ❖ **text-align**: Sirve para indicar la alineación del texto.
- ❖ **text-decoration**: Sirve para decorar el texto.
- ❖ **text-indent**: Especifica una sangría para la primera línea de un párrafo.
- ❖ **text-shadow**: Da al texto un efecto de 3d con el uso de sombras.
- ❖ **vertical-align**: Indica una posición relativa de un texto con relación a la línea que lo contiene.
- ❖ **word-spacing**: Añade un espacio las palabras.
- ❖ **text-transform**: Sirve para convertir el texto o palabras en mayúsculas y minúsculas.

Propiedades de color y fondo.

- ❖ **color**: Especifica un color para el texto.
- ❖ **background-color**: Se usa para indicar un color para el fondo.
- ❖ **background-image**: Indica una imagen para el fondo del elemento.
- ❖ **background-attachment**: Sirve para especificar cómo se va a acoplar una imagen de fondo en la pantalla.
- ❖ **background-position**: Indica la posición de la imagen de fondo.
- ❖ **background-repeat**: Indica si se desea que la imagen se repita o no.

Propiedades de caja.

Cada elemento de un documento puede estar contenido en una caja rectangular, y esta se rodea con 3 cajas más: el relleno, el borde y el margen.

- ❖ **position**: Indica la posición de las cajas.
- ❖ **clear**: Define donde colocar al elemento cuando se encuentra con la propiedad float.
- ❖ **float**: Define el espacio de visualización donde se colocara un elemento flotante y provoca que el texto fluya alrededor de alguna manera establecida.
- ❖ **height**: Define la altura de visualización de la etiqueta asociada.
- ❖ **width**: Indica el ancho de la etiqueta asociada.
- ❖ **border-color**: Se utiliza para definir un color de borde.
- ❖ **border-width**: Indica el ancho del borde.
- ❖ **border-style**: Define la apariencia de los bordes.
- ❖ **border-top, border-bottom, border-right, border-left**: Define el borde superior, inferior, derecho e izquierdo respectivamente.
- ❖ **margin-left, margin-right, margin-top, margin-bottom**: Indica el margen izquierdo, derecho, superior e inferior respectivamente del elemento de la parte exterior de los bordes.

- ❖ **padding-left, padding-right, padding-top, padding-bottom:** Definen un relleno en la parte izquierda, derecha, superior e inferior respectivamente entre el contenido y el borde.

Propiedades de las listas ordenadas y no ordenadas.

- ❖ **list-style-image:** Define una imagen para marcar el elemento de la lista.
- ❖ **list-style-position:** Indica la posición del elemento de la lista.
- ❖ **list-style-type:** Define la apariencia de cómo se deben de presentar las listas numeradas y no numeradas.

1.8 DESARROLLO DE ANIMACIONES PARA SITIOS WEB CON FLASH 8

Flash es una de las herramientas por excelencia para diseño web de animaciones, con un grado de complejidad que puede ser manejado por principiantes hasta programadores o diseñadores avanzados. Con él se puede enriquecer los proyectos web en un grado multimedia de buen agrado para los usuarios, integrando audio, video, imágenes e interactividad. Con flash se pueden diseñar animaciones, películas interactivas, cd's interactivos, y es un buen editor de vectores.

La interfaz de trabajo es similar a Fireworks, ya que ambos son desarrollos de Macromedia, contiene comandos, barras de menús, barras de herramientas, grupo de paneles, inspector de propiedades, visualizador de línea de tiempo, capas, etc.

Lo primero que se debe tomar en cuenta es el escenario donde se desarrollara la animación. El escenario es el área rectangular donde se coloca el contenido gráfico. Ajustando las propiedades del documento se podrán cambiar las medidas del escenario, considerando la resolución total de los monitores estándar, y una resolución parcial dentro de los navegadores convencionales.

La velocidad de reproducción es otro punto importante a mencionar, lo ideal para el barrido de una computadora en una animación o en CD es de 15 cuadros por segundo esta armonizado con el barrido y ocupa menos recursos de memoria y procedimientos, pero cuando esta velocidad es pensada en Internet por la optimización de la transferencia de datos, la velocidad ideal es de 12 FPS. En el caso particular de un banner publicitario de Internet la velocidad puede llegar a dos cuadros por segundo.

Se cuentan con diversas ayudas para tener una mejor visualización del documento, usando los zooms apropiados, las reglas para tener una relación de distancia entre objetos, y la utilización de guías que son referencias lineales que cruzan el escenario. Estas líneas sirven para acomodar los objetos en determinada alineación y componer de

manera más eficiente. También se puede mandar a llamar una cuadrícula que es un reticulado que pone una red de cuadros o rectángulos a la medida que se desee.

Importación de elementos para su tratamiento

Flash reconoce gran variedad de formatos de mapas de bits y vectoriales, que al hacer importados se agregaran automáticamente a la biblioteca del documento. Al importar sonido Flash reconoce el formato WAV (sólo en Windows), AIFF (sólo en Macintosh) y MP3. Existen dos tipos de sonido en Flash: sonidos de evento y flujos de sonido. Un sonido de evento debe descargarse por completo antes de empezar a reproducirse y continúa haciéndolo hasta que se detiene completamente. Los flujos de sonido empiezan a reproducirse en cuanto se ha descargado información suficiente para los primeros fotogramas y sincronizan el sonido con la línea de tiempo para reproducirlo en un sitio Web.

Se debe seleccionar opciones de compresión para controlar la calidad y el tamaño del sonido en las películas al exportarlas. Se puede seleccionar las opciones de compresión para sonidos determinados, mediante el cuadro de diálogo Propiedades de sonido, o definir valores para todos los sonidos de la película en el cuadro de diálogo Configuración de publicación.

También se pueden importar películas de video de Flash, según el formato de video y el método de importación que se elija, se puede publicar el video como película Flash (archivo SWF) o como película QuickTime (archivo MOV).

1.8.1 Línea de Tiempo, Capas y Fotogramas

El tiempo es básico para la comprensión del movimiento dentro de la animación, se usa la línea de tiempo, que se encuentra encima del escenario, y con ella se organiza y controla el contenido de una película a través del tiempo, en capas y fotogramas.

También controla la duración de la animación y la velocidad de la reproducción. Se puede decir que la línea de tiempo es el progreso de los elementos a través del tiempo virtual asignado a cada elemento.

La información de estado de la línea de tiempo situada en la parte inferior indica el número del fotograma seleccionado, la velocidad de fotogramas actual y el tiempo transcurrido hasta el fotograma actual.

Capas

Las capas son varias bandas de película apiladas unas sobre otras, cada una de las cuales contiene una imagen diferente que aparece en el escenario. Desde ella se controla

su distribución, color de entorno permisos de edición y visualización. Las capas son muy importantes ya que de ahí se separaran los movimientos de los elementos y su visualización en el escenario.

Las capas aparecen a la izquierda de la línea de tiempo. Se recomienda asignar un nombre a la capa para mejor reconocimiento de los elementos. La posición de las capas en la línea de tiempos da su modo de visualización en planos. Es decir la capa en la parte superior de la línea de tiempo será el primer plano y así hasta llegar al fondo. Esto se puede cambiar simplemente seleccionando y arrastrando las capas a la posición deseada.

Las propiedades de mezcla sirven para combinar los colores de los objetos en la capa superior contra las inferiores, de tal manera que se puede experimentar con los colores desarrollando nuevos a partir de algunos ya establecidos. Se pueden sumar colores, sustraerlos, sombrearlos, hacer opuestos, perforar brillos, su modo de quemado, superposición de colores. Igual se pueden añadir efectos básicos de capa que son similares a Photoshop como son el blur, bevel, shadow y ajustes de color.

Fotogramas

En la línea de tiempo se trabaja con fotogramas, los cuales indican la duración de los elementos del proyecto sus cambios o transformaciones y el periodo que van a durar estas. Existen varios tipos, como el cuadro clave, el cuadro vacío, y el cuadro de transición. Los fotogramas clave son aquellos en los que tiene lugar un cambio importante en algún objeto de la animación (modificando su posición, cambiando de forma, tamaño, etc.). El fotograma vacío es sólo para dejar un espacio en el elemento o para crear un cuadro donde se le dará un uso de programación. El fotograma de transición no cuenta con ningún cambio pero existe, son fotogramas localizados entre dos cuadros clave.

Entre los distintos cambios que se pueden hacer en los fotogramas es eliminarlos, moverlos, alargarlos, copiarlos o convertirlos de un fotograma clave a un fotograma normal.

1.8.2 Creación y Edición de Animaciones

Para crear una animación en un proyecto de Flash, es necesario modificar el contenido de fotogramas sucesivos. Se puede hacer que un objeto se desplace a los largo del escenario, aumente o disminuya de tamaño, gire, cambie de color, desaparezca o aparezca progresivamente. Pudiendo hacer que estos cambios funcionen simultáneamente.

Existen dos métodos de animación en Flash: la animación interpolada y la animación fotograma por fotograma. En la animación interpolada, se crean los fotogramas inicial y

final, y Flash crea los fotogramas intermedios entre estos dos puntos. El programa se encarga de variar el tamaño, la rotación, el color u otros atributos del objeto uniformemente entre los fotogramas inicial y final para crear la ilusión de movimiento. En la animación de fotograma por fotograma, la imagen se va creando y modificando individualmente en cada fotograma.

Además de ahorrar tiempo se puede ahorrar peso ya que en esta forma se reduce la información vectorial, simplemente Flash calcula los cambios. En la interpolación de movimiento, se definen propiedades tales como la posición, el tamaño y la rotación de una instancia, un grupo o un bloque de texto en un instante específico, y después en estas propiedades se pueden cambiar en otro momento.

La animación por interpolación de forma es otro método más para animar, similar a la transformación, cuando una forma parece transformarse en otra. En la interpolación de forma, se dibuja un objeto en un instante específico y después se modifica o se dibuja otro objeto en otro instante. El programa se encarga de interpolar los valores o formas de los fotogramas intermedios para crear la animación, calculando el cambio de una forma a otra simplemente la condición es que sean vectores y por cálculos matemáticos lleva unos a otros.

Consejos de forma

Los consejos de forma identifican puntos que deben corresponder en las formas inicial y final. Por ejemplo, si se está interpolando el dibujo de una cara mientras cambia de expresión, se puede marcar cada ojo con un consejo de forma. En lugar de que la cara se convierta en una figura totalmente diferente durante el cambio, los ojos siguen siendo reconocibles y cambian de manera independiente durante el proceso.

Los consejos de forma se pueden identificar fácilmente tienen letras de la a hasta z que corresponden de la forma inicial a la final, y aparecen de color amarillo en los fotogramas clave iniciales, verde en los fotogramas clave finales, y rojos cuando no se encuentran en un curva.

Capas en animación

Las capas son elementos que ayudaran a crear una sensación de profundidad, pudiendo animarlas y editarlas independientemente, así se podrán apilarlas de formas diferentes para crear efectos; existen diferentes tipos de capas: Capas de animaciones, que se usan para contener elementos que sólo serán animados. Capa normal, es la capa por default que proporciona flash al iniciar un documento. Capa Guía, ésta se usará como referencia para el trabajo.

Las capas de guía de movimiento permiten dibujar trazados a lo largo de los cuales se animan instancias, grupos y bloques de texto interpolados. Se pueden vincular varias capas a una capa de guías de movimiento para hacer que varios objetos sigan el mismo trazado.

Uso de Mascaras

Las mascararas son una capa que determina la parte que se verá en el escenario. Es decir la capa ocultará el contenido del escenario que está inmerso en ella. Los elementos dentro de ella sólo podrán ser visualizados cuando pasen por la zona que se enmascarilló previamente. Un elemento de mascara puede ser una forma rellena, un objeto de tipo, una instancia de un símbolo de gráfico o un clip de película. Varias capas pueden agruparse bajo una misma capa de máscara para crear efectos sofisticados.

Para crear efectos dinámicos se puede animar una capa de máscara. Para una forma rellena utilizada como máscara, se utiliza interpolación de movimiento de formas; para un objeto de tipo, una instancia de un gráfico o un clip de película, se utiliza interpolación de movimiento. Si se utiliza una instancia de clip de película como máscara, se puede animar la máscara a lo largo de un trazado de movimiento.

1.8.3 Símbolos, Instancias y Elementos de Biblioteca

Un símbolo es un gráfico, un botón o un clip de película que se crea en Flash, se puede crear una única vez y poder utilizarlo cuantas veces se quiera. Cada símbolo tiene su propia línea de tiempo, y se puede hacer las mismas modificaciones que se hacen sobre la línea de tiempo normal del documento.

Una instancia es una copia de un símbolo ubicada en el escenario o anidada en otro símbolo. Esta instancia puede diferir a su símbolo en color, tamaño o función.

Símbolos Gráficos: se usan para las imágenes estáticas y para crear animaciones reutilizables ligadas a la línea de tiempo principal.

Símbolos de Botón: se usan para crear botones interactivos que respondan a las pulsaciones y desplazamientos del ratón, o a otras acciones.

Símbolos de Clip de Película: tienen sus propias líneas de tiempo de varios fotogramas, independientes de la línea de tiempo principal. Se encuentran dentro de una línea de tiempo principal que contiene elementos interactivos como controles, sonidos e incluso otras instancias de clip de película.

Símbolos de fuentes: se usa para exportar fuentes y utilizarlas en otros documentos de Flash.

Biblioteca: La biblioteca de Flash es el sitio virtual donde el programa almacena la información de los elementos, símbolos, sonidos, imágenes y cambios hechos en un elemento. En ella se pueden cargar los elementos y clonarlos infinitas veces sin sufrir cambio alguno.

1.8.4 Audio y Video

Se puede añadir sonido al proyecto, se añade a la biblioteca asignándolo a una capa y definiendo las opciones de los controles de sonido en el inspector de propiedades. Se recomienda colocar cada sonido diferente en capas individuales.

En flash se pueden importar videos en formato MOV, AVI, MPEG u otros. Los clips de video en formato FLV pueden impostarse directamente a Flash.

Para seleccionar el nivel de compresión que debe aplicarse al video importado se debe elegir un nivel de compresión, se usa el asistente de importación de video para visualizar estas opciones; se puede comprimir en función al ancho de banda o calidad de video.

Concluyendo se puede decir que Flash es un programa muy poderoso que proporciona las herramientas suficientes para realizar animaciones para los sitios web, o agregarle mucha interactividad o multimedia a el proyecto.

1.9 ACTIONSCRIPT

ActionScript, es el lenguaje de creación de comportamientos de Macromedia Flash, que permite agregar interactividad a los proyectos, además de agregarle elementos funcionales y desarrollo de aplicaciones propias para Internet. Proporciona elementos, como acciones, operadores y objetos, que se agrupan en comportamientos que indican a la película lo que debe hacer; comanda la película a fin de que los eventos, como presionar los botones del ratón y las teclas, activen estos scripts.

En Flash, el panel de Acciones sirve para programar comportamientos con ActionScript. Los comportamientos o scripts se crean eligiendo las opciones de los menús y las listas o con el panel se introduce texto directamente en el panel Script. En ambos modos, existen consejos sobre códigos que ayudan a realizar acciones e insertar propiedades y eventos. Una vez que se disponga de un script, se puede adjuntar a un botón, a un clip de película o a un fotograma para crear la interactividad que necesite.

1.9.1 Acciones

Para agregar una acción a un documento de Flash, se debe asociar a un botón, a un clip de película o a un fotograma en la línea de tiempo es decir darle el comportamiento a un objeto. El panel Acciones permite seleccionar, arrastrar y soltar, reorganizar y eliminar acciones.

Utilización de la sintaxis de ActionScript

ActionScript sigue reglas gramaticales y de puntuación que determinan que caracteres y palabras se utilizan para dar significado a una sentencia y el orden en que se deben escribir. En ActionScript, se utiliza un punto y coma para finalizar una sentencia, por ejemplo: `stop ();`

También se utiliza un punto (.) para indicar las propiedades o métodos relacionados con un objeto o un clip de película. Se utiliza para identificar la ruta de destino a un clip de película, variable, función u objeto. Una expresión que utiliza una sintaxis con punto empieza con el nombre del objeto o clip de película seguido de un punto y termina con el elemento que desee especificar.

El punto en las rutas sirve para llegar a determinado objeto. Si un objeto se encuentra dentro de una película se debe de indicar que película, y en donde, por ejemplo: `proyecto.pelicula.objeto`

La sintaxis con punto también utiliza alias especiales el guión bajo es uno de ellos para indicar el nivel dentro de la ruta el guión bajo es la gramática correcta, ejemplo: `_root` y `_parent`. El alias `_root` se refiere a la línea de tiempo principal. Puede utilizar el alias `_root` para crear una ruta de destino absoluta. Por ejemplo, la siguiente sentencia llama a la función `has esto` en el clip de película `functions` en la línea de tiempo principal:

```
_root.functions.hasesto();
```

Se puede utilizar el alias `_parent` para referirse a un clip de película en el que está anidado el clip de película actual. También se puede utilizar `_parent` para crear una ruta de destino relativa. Por ejemplo, si el clip de película `dog` se encuentra anidado dentro del clip de película `animal`, la siguiente sentencia de la instancia `dog` indica a `animal` que se detenga:

```
_parent.stop();
```

Las sentencias de ActionScript se agrupan en bloques con llaves (`{ }`), como se muestra en el siguiente ejemplo:

```
on(release) {  
    myDate = new Date();  
    currentMonth = myDate.getMonth();  
}
```

El fin de una sentencia se indica mediante un punto y coma.

```
column = passedDate.getDay();  
row = 0;
```

Aun cuando omita el punto y coma final, Flash compilará el script correctamente. No obstante, al crear scripts se recomienda utilizar el punto y coma final.

Al definir una función, los parámetros deben incluirse entre paréntesis, al llamar una función, se incluyen los parámetros que se deseen pasar a la misma entre paréntesis, ejemplo:

```
myFunction ("Steve", 10, true);
```

Puesto que ActionScript no distingue entre mayúsculas y minúsculas, no deben utilizarse nombres de variable que coincidan con objetos de ActionScript incorporados. Así, por ejemplo no puede especificarse lo siguiente:

```
date = new Date ();
```

En el panel Acciones se deben utilizar comentarios para agregar notas a los scripts. Los comentarios son de gran utilidad para realizar un seguimiento de las acciones que se han deseado llevar a cabo, para proporcionar muestras y, en el caso de que se trabaje en un entorno de trabajo en equipo, para pasar información a otros desarrolladores. Estos comentarios serán útiles para saber cómo se llega hasta este punto o si alguien más los ve, saber cómo se desarrollo el comportamiento sin que las palabras afecten el mismo. Los comentarios se dan por dos diagonales, por ejemplo: *//esto fue un comentario*

1.9.2 Comportamientos

ActionScript sirve para crear scripts que le indican a Flash la acción que debe llevar a cabo cuando ocurra un evento. Algunos eventos que desencadenan un script se producen cuando la cabeza lectora llega a un fotograma, cuando se carga o descarga un clip de película, o cuando el usuario hace clic en un botón o presiona una tecla del teclado.

Los scripts pueden constar de una sola acción, como indicar a una película que se detenga, o bien de una serie de acciones, como primero comprobar una condición y, a continuación, realizar una acción.

En modo normal, los scripts se crean seleccionando elementos de la caja de herramientas Acciones, la lista situada a la izquierda del panel Acciones. También se puede seleccionar acciones en el menú emergente del botón de adición (+). La caja de herramientas Acciones separa los elementos en categorías como Acciones, Propiedades y Objetos, además proporciona una categoría Índice que muestra todos los elementos ordenado alfabéticamente. Al hacer clic en un elemento una vez, la descripción de éste aparece en la esquina superior derecha del panel. Al hacer doble clic en un elemento, éste aparece a la derecha del panel Script.

Ventana de comportamientos

Existen también comportamientos predeterminados en una ventana fuera de las acciones de ActionScript estos comportamientos ya están preconfigurados para obtener comportamientos sencillos sobre objetos varios. En esta ventana sólo hace falta seleccionar el objeto y aplicarle tal o cual comportamiento, al momento de aplicarlo aparecerá en líneas de programación dentro del panel de acciones.

Flash permite utilizar los botones situados sobre el panel Script para buscar y reemplazar texto, establecer y eliminar puntos de corte de depuración, ver los números de línea e insertar rutas de destino; también permite utilizar el menú de salto y el botón fijar del panel Script.

Además se puede comprobar la existencia de errores en la sintaxis, formatear código de forma automática y utilizar consejos sobre códigos para que resulte más sencillo completar la sintaxis. Además la función de balance de puntuación ayuda a establecer los pares de paréntesis, corchetes o llaves.

Asignación de acciones a un fotograma

Para que una película lleve a cabo una acción cuando la cabeza lectora llegue a un fotograma de la línea de tiempo, debe asignar una acción al fotograma. Por ejemplo, para crear un loop en la línea de tiempo entre los fotograma 20 y 10, se debe agregar una acción al fotograma 20 que envíe la cabeza lectora al fotograma 10: *gotoAndPlay (10);*

Es recomendable colocar todas las acciones de fotograma en una capa denominada Acciones; de ese modo, siempre podremos encontrar las acciones en la línea de tiempo. Después de haber asignado una acción se comprueba si funciona.

Eventos de Ratón

Para que una película lleve a cabo una acción cuando se hace clic en un botón o se sitúa el cursor sobre él, se debe asignar una acción a dicho botón. Es necesario asignar acciones a una instancia del botón; las demás instancias del símbolo no se ven afectadas.

Cuando se asigna una acción a un botón, hay que anidarla dentro de un controlador y especificar los eventos de ratón o teclado que desencadenarán la acción. Cuando se asigna una acción a un botón en modo Normal, el controlador *on* se inserta de forma automática y se puede elegir un evento de una lista. También se puede utilizar eventos del objeto *Button* de ActionScript para ejecutar scripts cuando se produce un evento de botón.

Asignación de acciones a un clip de película

Cuando se asigna una acción a un clip de película, se debe anidar dentro de un controlador *onClipEvent* y especificar el evento de clip que desencadena la acción. Cuando se asigna una acción a un clip de película en el modo Normal, el controlador *onClipEvent* se inserta automáticamente. Se puede elegir un evento en una lista, para ejecutar script también se puede utilizar eventos del objeto *MovieClip* y del objeto *Button* de ActioScript.

1.9.3 Control de la línea de tiempo

Para ir a una escena o a un fotograma específico de la película, se utiliza la acción *goto* “ve a”, cuando la película salta a un fotograma, se pueden elegir parámetros que reproducen la película a partir del nuevo fotograma (el predeterminado) o la detienen en éste, la acción *goto* aparece en la caja de herramientas Acciones como dos acciones: *gotoAndPlay* y *gotoAndStop*. El primero ira al fotograma y seguirá hasta que encuentre un script de detenerse *Stop ()*, el segundo ira y se detendrá donde se le indico. La película también puede saltar a una escena y reproducir un fotograma especificado o el primer fotograma de la escena siguiente o de la anterior.

Si se adjunta la acción a un fotograma, en el panel Script aparece el código siguiente:

```
stop ();
```

Si se adjunta a un botón, la acción se incluye de forma automática en un controlador *on* (evento de ratón):

```
on (release){  
stop();  
}
```

Si se adjunta a un clip de película, la acción se incluye de forma automática en un controlador *onClipEvent*:

```
onClipEvent (load){  
stop();
```

Si la acción es reproducir una película será como en los casos contrarios, pero sustituyendo la acción *stop ()*; por *play()*;

Salto a otros URL (a otras páginas)

Para abrir una página web en una ventana de navegador o para pasar datos a otra aplicación en un URL definido, puede utilizar la acción *getUrl*. Por ejemplo, se puede disponer de un botón que lleve a un nuevo sitio Web, o bien enviar datos a un script CGI para procesarlo igual como se procesaría un formulario HTML.

1.9.4 Control de objetos

En la creación de scripts orientados a objetos, la información se organiza en grupos denominados clases. Se puede crear varias instancias de una clase, denominadas objetos, para utilizarlas en los scripts. Se puede crear las propias clases y utilizar las clases incorporadas de *ActionScript*. Al crear una clase, es preciso definir todas las propiedades (características) y métodos (comportamientos) de cada objeto que ésta crea. Los objetos en *ActionScript* pueden ser meros contenedores de datos, o estar representados gráficamente en el escenario como clips de película, botones o campos de texto. Todos los clips de película son instancias de la clase incorporada *MovieClip*, y todos los botones son instancias de la clase incorporada *Button*. Para definir una clase, se debe crear una función especial denominada función constructora (constructor).

En la creación de scripts orientados a objetos, las clases pueden recibir propiedades y métodos de ellas mismas, de acuerdo a un orden determinado; esta función se denomina herencia. La herencia puede utilizarse para ampliar o redefinir las propiedades y métodos de una clase. Una clase que hereda propiedades y métodos de otra clase recibe el nombre de subclase. Una clase que pasa propiedades y métodos a otra clase recibe el nombre de superclase. Una clase puede ser a la vez subclase y superclase.

Utilización de variables en un script

Se debe declarar una variable en un script antes de poder utilizarla en una expresión. Si se utiliza una variable que no ha sido declarada el valor de la variable será *undefined* (indefinido) y su script generará un error.

Los operadores son signos matemáticos que ejecutan respuestas, las acotan o las reducen a un objetivo previsto. En *ActionScript* se puede usar los operadores de comparación, valores numéricos, valores alfanuméricos para concatenación de cadenas, o booleanos.

ActionScript permite de igual forma trabajar con sentencias de control de variables como los arreglos, que permiten almacenar información por índices. Además se usa también sentencias de control como los loops y bucles, condicionales, funciones básicas y avanzadas.

Rutas y niveles de objetos en flash

Para llegar a ciertas instancias se debe mencionar detalladamente en donde se encuentran el elemento por eso existen niveles para llegar a llamar diferentes objetos, propiedades o scripts. Por ejemplo:

_root Es la raíz de la película, la línea principal de la película madre.
_parent Es el objeto contenedor de otro, es decir el nivel superior del objeto, pero no es la línea principal.

1.9.5 Componentes

Los componentes son clips de película complejos con parámetros definidos (que se establecen durante la edición de documentos) y un conjunto exclusivo de métodos. ActionScript que permiten definir parámetros y opciones adicionales en tiempo de ejecución. Los componentes reemplazan y amplían los clips inteligentes presentados en versiones anteriores de Macromedia Flash. El programa incluye pocos componentes de interfaz de usuario de Flash como *CheckBox*, *ComboBox*, *ListBox*, *PushButton*, *RadioButton*, *ScrollBar* y *ScrollPane*, algunos componentes para la visualización de video y audio, etc.

Preparación de películas flash para la Web

El proyecto ya terminado tiene varias posibilidades de ser exportado para la web, una de ellas en la opciones de publicación se encuentran las diferentes opciones de exportación tanto de el "SWF" como de la pagina HTML, encontraremos los parámetros de tamaño de documento, su alineación en HTML; así como la integración de flash con la ayuda de otros editores de HTML como Dreamweaver. Para esto se tiene el menú de comandos File/publish settings, herramienta que permitirá proteger el trabajo, así como reducir y/o optimizar sus elementos para mejorar o aligerar el producto.

La reproducción de una película Flash en un navegador Web requiere un documento HTML que active la película y especifique la configuración del navegador. Este documento se genera de forma automática con el comando Publicar, desde los parámetros HTML de un documento de platilla.

Exportar flash en otros formatos

Es posible exportar a flash en otros formatos pero no es muy común, se puede exportar el proyecto a secuencias de imagen jpeg, png, gif animados, películas mov, etc.

Flash es una herramienta poderosa y ligera sus capacidades como herramienta son enormes, siempre y cuando se tenga una buena imaginación y ganas de querer hacerlo. El uso de ActionScript le dará al proyecto, mayor potencial y mayor control en la elaboración de aplicaciones Flash.

1.10 INTEGRACIÓN Y MANTENIMIENTO DE SITIOS WEB CON DREAMWEAVER 8

El propósito de este módulo es conocer el uso de las herramientas de Dreamweaver para facilitar la gestión e integración del sitio web. Dreamweaver es un gran editor de código HTML, con el se puede manejar y editar de una manera manual o automática con ayuda de muchas herramientas que permitirán crear las páginas de forma rápida. Además de que trabaja con otras aplicaciones Web dinámicas a partir de bases de datos, empleando lenguajes de servidor como ASP, ASP.NET, ColdFusion Markup Language, JSP y PHP, además interactúa con otras herramientas de Adobe como Fireworks y Flash.

Entorno de Trabajo

En primer lugar existen dos opciones para trabajar en Dreamweaver, la vista de Diseñador o la vista de Programador, o se puede tener una tercera vista, que es una combinación de las dos.

De esta forma se identifican los principales componentes del programa: la barra de menús, barra insertar, barra de herramientas de documento, grupos de paneles, selector de etiquetas, inspector de propiedades, y la ventana de trabajo.

La Barra de Menús, permite acceder a diferentes opciones del programa como abrir archivo, copiar y pegar, insertar imágenes, etc.

La Barra Insertar, presenta 8 barras de herramientas con botones y opciones independientes que permiten la inserción de objetos como texto, capas, imágenes, elementos Flash, formularios, etc.

La Barra de Herramientas de documento contiene herramientas que permiten proporcionar diferentes vistas de la ventana documento, el título de documento, ayudas visuales, administrador de archivos, validadores de formato, etc.

La Barra de Estado muestra las herramientas que sirven para la visualización del documento e información adicional del documento que se está generando como tamaño y tiempo de descarga.

El Grupo de Paneles permite tener un control de todos los atributos, capas, los estilos, códigos, etiquetas, comportamientos y archivos que se usan en el desarrollo del proyecto.

El Inspector de Propiedades permite examinar y editar los atributos del elemento seleccionado en la página. Las propiedades del inspector cambian dependiendo del objeto seleccionado.

Panel de Resultados se encuentra en la parte inferior del inspector de propiedades y sirve para controlar mejor las páginas y sitios Web, con ayuda de las herramientas de Buscar, Referencia, Validación, Revisión del navegador destino, Verificador de vínculos, Informes de datos, Registro FTP y Depuración del Servidor.

La Definición de un Sitio Web

Consiste en identificar y organizar los archivos que van a conformar el proyecto web, lo que permitirá un mejor control de los documentos que lo conformarán, así como el acceso remoto al servidor donde se alojara el sitio. Se usa la opción *Menú Sitio>Administrar Sitios>Nuevo>Sitio*. Desde ahí se puede definir el nombre para el sitio, la URL, definir tecnologías de servidor, la carpeta donde se guardaran los archivos localmente y opciones de conexión al servidor.

Manejo de Archivos: Se recomienda guardar los distintos tipos de archivos en carpetas independientes, para su fácil manejo y ubicación, así se tienen imágenes, animaciones, audios y videos en diferentes carpetas, además si se usan demasiados documentos HTML que pertenecen a un mismo sitio pero con temas distintos, también es recomendable agruparlos en diversas carpetas.

1.10.1 Creación y Elementos Web

Documento

Para crear un documento basta con ir a *Menú>Archivo>Nuevo>Página Básica>HTML>Crear*. De aquí se obtiene un documento HTML en blanco, para desarrollar la página Web.

Meta Tags

Dreamweaver puede incluir etiquetas para la sección head, que sirve para la identificación y descripción de las Páginas Web, a través de *Barra de Herramientas HTML>Meta*.

- ❖ La Etiqueta Meta registra información sobre la página actual, como la codificación de caracteres, el autor, copyright, o palabras clave.
- ❖ La Etiqueta Palabras Clave sirve para que los robots de motores de búsqueda detecten automáticamente y recopilen la información de dicha etiqueta y se indexen estos datos en sus bases de datos.
- ❖ La etiqueta descripción sirve para indicar un párrafo del contenido y tema del cual trata la página web.
- ❖ Actualizar sirve para especificar al navegador cuando debe volver a cargar la página o pasar a otra distinta después de un tiempo determinado.
- ❖ La etiqueta Base sirve para establecer un URL que sirva de referencia para todas las rutas de la página relativas al documento.
- ❖ Vinculo define una relación entre el documento actual y otro activo.

Propiedades de la Página

Se pueden modificar varias opciones básicas del diseño del documento a través del *Menú>Modificar>Propiedades de la Página*. Estas opciones pueden ser:

- ❖ Aspecto: Se refiere a la apariencia visual que tendrá la página (fuentes, tamaños, colores).
- ❖ Vínculos: Presenta las propiedades que pueden tener los vínculos normales, activos, de sustitución y los visitados a través de el tamaño, fuente o color.
- ❖ Encabezados: Permite indicar la fuente, tamaño y color de letra, de los encabezados del documento.
- ❖ Título/Codificación: Indica el título, tipo de documento y codificación que se le puede aplicar a la página.
- ❖ Imagen de Rastreo: Es una imagen que aparecerá de fondo, y se puede utilizar como referencia para seguir un diseño específico.

Para guardar el documento se usa el *Menú Archivo>Guardar Como*, por omisión, se guardará con extensión .htm, de no especificarse lo contrario. Con la opción F12, se puede tener una vista previa del documento HTML en el navegador.

Tablas y Celdas de Diseño

El uso de estos objetos, permitirán tener más control sobre las estructuras complejas del diseño, que requieren mayor precisión y dinamismo en la inserción y acomodo de texto e imágenes. Se crean a partir de la Barra de herramientas de diseño al dar clic sobre Diseño para activar tablas y celdas, o con *Menú Insertar>Objetos de diseño>Tabla de diseño*. Las tablas se muestran en un color de línea verde y las celdas en color de línea azul.

Convertir tablas en capas: Si se desea convertir tablas a capas, éstas deben de tener un color de fondo y se hace a través de *Menú Modificar>Convertir>Tablas en capas*.

Capas

Las capas funcionan también como contenedores de información, como texto, imagen, animaciones y video. Se pueden crear a partir de Barra de herramientas Diseño, dando clic en el botón correspondiente, o a través del *Menú Insertar>Objetos de Diseño>Capa*. Para modificar las capas sólo basta con seleccionar la deseada y cambiar sus atributos en el inspector de propiedades. Para convertir las capas a tablas, es importante verificar que las capas no se encuentren encimadas o que rebasen el tamaño de la ventana, para realizar la conversión se usa el *Menú Modificar>Convertir>Capas a Tablas*, y a continuación se pueden realizar ciertos ajustes como evitar la superposición de tablas, usar cuadrículas, etc.

Estilos

Para agregar y aplicar un estilo en Dreamweaver es necesario seleccionar el texto u objeto al que se va a cambiar de aspecto y luego recurrir al *Menú texto>Estilos CSS>Nuevo*, o también se puede hacer a través del *Panel CSS>Estilos CSS* y dar clic en el botón *Nueva Regla CSS*. A continuación se podrá elegir el tipo de selector que se desee. En el *Panel CSS*, en la pestaña de *Estilos CSS* al seleccionar un estilo se activaran las propiedades de éste y permitirá editar cualquiera directamente en pantalla, sin necesidad de abrir otras ventanas de edición. Las hojas de estilo externas son los documentos que contienen las reglas de apariencia que se aplicaran al proyecto HTML o XHTML y que se almacenan fuera del documento con la extensión *.css*. Para aplicar la hoja de estilo externa se hace a través de vincular e importar desde el panel CSS, o haciendo clic en el icono adjuntar hoja de estilos, donde se debe especificar la ruta del CSS externo para adjuntarlo.

Imágenes

Las imágenes se insertan a partir de la *Barra de herramientas Común*, dando clic en el icono de imagen, o con el *Menú Insertar>Imagen*. Después se tiene que indicar la ruta donde se localiza la imagen, se puede cambiar las propiedades de la imagen, desde el inspector de propiedades. Para crear una imagen de sustitución (*RollOver*), se necesita dos archivos de imágenes, una será la principal, y la imagen secundaria es la que aparecerá al pasar el puntero sobre la imagen principal. Ambas imágenes deben tener el mismo tamaño; dichas imágenes responderán al evento *onMouseOver*. Se puede hacer a través de la *Barra de herramientas Común*, dando clic en el icono Imagen de Sustitución del menú emergente, o a través del *Menú Insertar>objetos de imagen>Imagen de sustitución*.

Vínculos

Los vínculos se pueden hacer a través de texto e imágenes principalmente. Se aplican para tener conexión a una dirección electrónica externa, con otra página dentro del sitio, dentro de un mismo documento y con archivos de cualquier tipo. Existen vínculos a rutas absolutas y relativas, las absolutas especifican la ruta completa del documento, incluido el protocolo que se debe usar; las rutas relativas son más adecuadas para utilizar vínculos locales en la mayoría de los sitios web (ej: ../mapaext.html). También se pueden manejar vínculos a correo electrónico y los vínculos de mapa de imagen, que es una zona que consta de regiones o zonas interactivas, que cuando el usuario da clic en cualquiera de esas zonas, se realiza una acción. Otro tipo de vínculo es conocido como ancla, y es un vínculo dentro de un mismo documento, se crea a partir del *Menú Insertar>Anclaje con nombre*.

Marcos

Un marco es una zona de una ventana de navegador que puede mostrar un documento HTML o XHTML independiente de lo que muestra en el resto de la ventana. El objetivo de esto es dividir una página Web en múltiples páginas HTML. Un sitio compuesto de tres marcos consta realmente de al menos cuatro documentos HTML o XHTML distintos. Entre las ventajas de utilizar marcos se distingue que el navegador de un usuario no tendrá que volver a cargar los gráficos de navegación para cada página. Cada marco dispone de su propia barra de desplazamiento cuando el contenido es demasiado grande para una ventana, permite al usuario desplazarse independientemente por cada marco. Los marcos se crean desde el *Menú Archivo>Nuevo Documento>General>Conjunto de Marcos*, seleccionando la estructura de marcos que se quiera utilizar.

1.10.2 Formularios

Un formulario permite hacer el envío de información desde el cliente hacia el servidor Web. A través de una serie de controles, el usuario puede introducir información que es enviada hacia el servidor para su procesamiento. Los formularios tienen diversas funciones, como realizar consultas a bases de datos, llenar cuestionarios, solicitar información sobre un tema, inscribirse en una lista de correo, libro de visitas, etc.

La estructura de los formularios es:

1. Etiqueta `<form>`, que incluyen el URL de la secuencia de comandos que procesará el formulario y el método que se utilizará para enviar los datos a un servidor.
2. Campos de formulario, como campos de texto, menús, casillas de verificación y botones de opción.
3. El botón Enviar, que manda los datos de la secuencia de comandos al servidor.

Para insertar los elementos del formulario en un documento de Dreamweaver, se hace a través de la barra de herramientas formularios. Es importante recordar que cada elemento que se inserte en el formulario, mostrará las propiedades que los constituyen y se podrán modificar según las necesidades que se consideren útiles para su funcionamiento. Entre los diversos elementos que cuentan los formularios están: campos de texto, campo oculto, área de texto, casilla de verificación, botón de opción, grupo de opciones, lista/menú, menú de salto, campo de imagen, campo de archivo, botones, etiqueta y juego de campos.

Comportamientos DHTML

Los comportamientos permiten la interacción del usuario con la página o sitio Web, con el fin de cambiar alguna propiedad o ejecutar determinadas tareas. El comportamiento funciona con evento y una acción. Los eventos son funciones llamadas de JavaScript que provocan la ejecución o no de acciones. Las acciones son comando o reglas que modificarán a un objeto. Dreamweaver tiene una lista de comportamientos que se obtienen a través del *Panel Inspector de Etiquetas>Comportamientos*. Entre los principales comportamientos se encuentran: Abrir ventana del navegador, arrastrar capas, cambiar propiedad, carga previa de imágenes, comprobar navegador, comprobar plug-in, controlar shockwave o flash, controlar sonido, definir texto, establecer imagen como barra de navegación, intercambiar imagen, ir a URL, llamar Javascript, línea de tiempo, mensaje emergente, menú de salto, mostrar menú emergente, mostrar-ocultar capas, restaurar imagen intercambiada, validar formulario, etc.

1.10.3 Insertar Objetos Media

Los objetos media son las animaciones, audios y videos que se pueden insertar en una página o sitio web para dar interactividad y reforzar el contenido del sitio en caso que se necesite.

Audio: Los formatos más comunes para insertar audio son el wav y mp3. Existen varias formas de insertar audio: agregando un audio a través de una liga o link, con la etiqueta *<bgsound>*, por medio del comportamiento "controlar sonido", a través del *Menú>Insertar>Media>Plug-in*, etc.

Video: Los formatos que se pueden incrustar en una página web son: avi, rm, mov y mpeg4. Se puede agregar video a través de una liga o link y por medio del *Menú Insertar>Media>Plug-in*.

Flash: El formato de archivo Flash Video (.flv) es un archivo que contiene datos codificados de audio y video para enviarlos a través de Flash Player. Para insertar el video se puede hacer a través del *Menú Insertar>Media>Flash Video*.

Animaciones: Se puede incrustar películas o animaciones de Flash a través del *Menú Insertar>Medio>Plug-in*. Al insertar una película de Flash, esta se mostrará con un fondo gris y con un icono de la letra F. Y se podrá hacer cambios en sus atributos en el inspector de propiedades.

También se puede incrustar películas y animaciones, a través del *Menú Insertar>Medio>Shockwave*.

1.10.4 Plantillas

Las plantillas son documentos que contienen zonas editables y no editables, las primeras se pueden modificar de forma libre, las segundas no se pueden modificar en un documento normal, sólo en el archivo plantilla, y son zonas que utilizan elementos constantes que se mostrarán en la mayoría de las páginas de un sitio, esos elementos constantes pueden ser logotipos, encabezados, imágenes, títulos, textos o simplemente la estructura general que permanecerá siempre en las páginas del sitio como un elemento común. Es importante señalar que cuando se crea una plantilla es necesario tener definido o configurado un sitio.

Ya que se identificaron las zonas, hay que seleccionar la tabla, capa o etiqueta div que contiene todo lo que si se va a poder editar, y dar clic en la *Barra de herramientas comunes>Plantillas>Región editable*. Hay que especificar un nombre para la zona seleccionada, y ésta cambiara de color y mostrara un borde de color verde agua, junto con una etiqueta del nombre que se indico para identificar la zona editable. Para guardar la plantilla hay que ir al *Menú Archivo>Guardar como plantilla*, donde se mostrara una ventana que indica el sitio al que pertenece, descripción y nombre, y se almacenará con la extensión .dwt.

Una vez que se ha marcado las zonas editables y diseñado la plantilla, ésta se puede aplicar a nuevos documentos. Esto se hace a través desde el *Menú Archivo>Nuevo>activar la ventana Plantillas* y se selecciona la que se quiere aplicar a un documento y se da clic en crear.

1.10.5 Publicación

Ya que se tiene definido el sitio, es importante verificar éste, con ayuda de la revisión de vínculos rotos e informes, para corroborar que todo esté funcionando correctamente, para su publicación en la Web.

Informes del Sitio: Cuando se prueba el sitio, se pueden compilar y generar informes para varios atributos HTML por medio del Menú Informes. Esta opción permite comprobar los vínculos externos, las etiquetas de fuentes anidadas combinables, el texto alternativo que

falta, las etiquetas anidadas repetidas, las etiquetas vacías que pueden borrarse y los documentos sin título. Puede comprobar y solucionar este tipo de problemas HTML en los documentos seleccionados o en todo un sitio antes de publicarlo. Para crear un informe hay que ir al Menú Sitio>Informes. Ya que se realizó el informe se van a mostrar los resultados, en la pestaña de Informes de sitios.

Revisar ligas rotas: Se puede hacer esta revisión a través del *Menú Sitio>Comprobar vínculos en todo el sitio*. Y se activará en panel de Resultados la opción de Verificador de vínculos. En el cuadro de opciones *Ver*, se puede hacer la comprobación de vínculos rotos, vínculos externos y archivos huérfanos.

Buscar y reemplazar: Es una herramienta muy útil, que agiliza la búsqueda de datos para modificarlos en un documento, carpeta y sitio. Se usa a través del *Menú Edición>Buscar y reemplazar*.

Información Remota y Publicación del Sitio.

La información remota, se refiere a todo el contenido que se encuentra almacenado y que se puede guardar en el servidor Web, y que será publicado y disponible en el WWW. En el servidor debe estar creada la carpeta *public_html* que es el lugar donde se almacenarán las páginas Web, para que se puedan mostrar y ser consultadas por los usuarios. Para publicar los archivos basta con seleccionar todos los archivos y carpetas y arrastrarlos a la ventana del sitio remoto en la carpeta destinada para el sitio.

1.11 JAVASCRIPT

JavaScript es un lenguaje de programación que permite escribir instrucciones estructuradas y secuenciales para indicarle a la computadora que tareas se desea que realice. Permite la declaración de variables, la manipulación de expresiones aritméticas y lógicas, el uso de estructuras de control, la declaración de funciones y, como en lenguajes orientados a objetos, permite la declaración y manipulación de objetos. La sintaxis de este lenguaje es muy similar del lenguaje C/C++ y en cuanto al manejo de objetos es similar a Java.

Para empezar a programar en JavaScript, no se necesita más que un editor de textos simples que guarde archivos HTML en formato ASCII o UNICODE y un navegador con soporte para JavaScript donde se puedan probar los programas escritos.

Ventajas

Dependerá básicamente de los objetivos que se le dé al sitio Web. Algunos de los usos comunes son:

- ❖ Validación de campos en un formulario HTML antes de enviarlos a un servidor Web.
- ❖ Mostrar ventanas emergentes al usuario.
- ❖ Detectar la presencia de Plug-ins como el Adobe Flash o el Real Player.
- ❖ Animaciones simples del tipo Roll Over.
- ❖ Juegos simples.
- ❖ Manipulación de cookies.
- ❖ Generación de HTML desde un programa JavaScript.

1.11.1 Programando en JavaScript

A los programas escritos en JavaScript se les dice guiones o scripts los cuales se insertan en secciones de documentos HTML o bien, en archivos aparte con la extensión “js” para indicar que son archivos con programas de JavaScript. Para empezar a escribir un programa se utiliza la etiqueta de apertura `<script>` y `</script>` para cierre. Dentro del encabezado del documento HTML se incluyen diferentes etiquetas como el título y los estilos en cascada, ahí se puede incluir el programa en JavaScript. Ejemplo:

```
<script lenguaje="javascript" type="text/javascript">
document.write("hola, mundo");
</script>
```

Los comentarios son anotaciones que se usan en los programas para documentar su comportamiento. Para hacer un comentario dentro del programa basta con usar `//` para comenzar a hacer el comentario de una sola línea; también se puede hacer un bloque comentario con `/*` y escribir el comentario de varias líneas y cerrarlo usando `*/`

JavaScript distingue entre mayúsculas y minúsculas, y siempre es recomendable (y a veces obligatorio) colocar punto y coma al final de cada sentencia. Los tipos de datos que puede manejar JavaScript son datos numéricos, cadena de caracteres, datos de tipo lógico, objetos, tipos undefined y null.

Variables

En JavaScript las variables pueden nombrarse al gusto del desarrollador siguiendo las siguientes reglas:

1. El primer símbolo del nombre de variable debe ser alfabético, un guión bajo (`_`) o un signo de pesos (`$`).
2. Los siguientes símbolos pueden ser alfanuméricos, guiones bajo (`_`) o el signo de pesos (`$`) excepto el punto (`.`).

3. No deben emplearse las palabras clave o reservadas del lenguaje como nombres de variables.

Operadores Numéricos: suma +, resta -, multiplicación *, división /, módulo %, incremento ++, decremento –

Operadores para cadenas de caracteres: Concatenación +, “Hola” + “mundo”

Operadores lógicos: AND &, OR |, NOT !, XOR ^.

Operadores Relacionales: menor que <, menor igual que <=, mayor que >, mayor igual que >=, igual que ==, diferente de !=

Declaración de variables: Se emplea la palabra clave var, después el nombre de la variable y opcionalmente se le asigna un valor. Ejemplos:

```
var nombre = “ “;  
var edad = 0;  
var fecha = null;
```

1.11.2 Estructuras de Control

Nos permiten modificar el flujo dentro de un programa. Existen básicamente dos tipos: Las condicionales y las iterativas. Las condicionales cambian el flujo del programa de acuerdo al cumplimiento de una condición y las iterativas realizan un mismo conjunto de instrucciones varias veces hasta que se cumple una condición.

Las estructuras de control condicional en JavaScript son de dos tipos. Una de ellas evalúa una condición y elige uno de dos caminos posibles. La otra compara una variable con un conjunto de valores y elige el camino que corresponde al valor con el cual se compara. Dentro de las estructuras de control condicional esta *if* y *switch*.

Las estructuras de control iterativas realizan un bloque de instrucciones varias veces, a estas repeticiones se les conoce como iteraciones. Existen dos tipos de estructuras iterativas, aquellas en las que se conoce el número de iteraciones y aquellas en las que se desconoce y depende del cumplimiento de una condición. Entre las estructuras iterativas se encuentran el *for*, *while* y *do-while*.

Creando funciones en JavaScript

El programa permite crear funciones con la palabra clave *function*. A éstas se les debe asignar un nombre que concuerde con el proceso que realizan, una función es similar a un subprograma, puede contener declaraciones de variables y estructuras de control. Las

funciones a menudo reciben parámetros de entrada los cuales modifican la salida de la función. Se le llama salida a aquello que la función “devuelve” y las funciones en JavaScript “devuelven” valores con la palabra clave *return*. Ejemplo:

```
function nombreFuncion (parámetros){  
    instrucciones;  
    return valorDeRetorno;  
}
```

En un documento HTML las funciones JavaScript se deben declarar en el encabezado. También pueden declararse en archivos con extensión “js” los cuales se invocarán en el documento que necesite las funciones. Aunque las funciones también pueden ser declaradas dentro del cuerpo del documento, es recomendable mantenerlas declaradas en el encabezado logrando con esto un programa más fácil de leer y comprender. Una vez declaradas y funcionando correctamente las funciones pueden ser invocadas con diversos parámetros y en cualquier parte de los programas.

Los archivos con funciones también se les conocen como documentos JavaScript, no deben contener etiquetas HTML solamente declaraciones de funciones en JavaScript. Para utilizar las funciones de estos archivos en un documento HTML se crean las etiquetas `<script>` y se les agrega la propiedad *src* la cual tendrá como valor la ruta donde se encuentran los documentos JavaScript. Ejemplo:

```
<script language="JavaScript" src="promedio.js"></script>
```

Ámbito de las variables

El ámbito de las variables se refiere a los bloques en los que una variable es conocida y conserva su valor, así existen variables de ámbito global y variables de ámbito local. Las variables de ámbito global, también llamadas globales, son declaradas dentro de etiquetas `<script></script>` y su valor será conocido en cualquier bloque de código que aparezca después de su declaración y podrá usarse sin problemas. Las variables de ámbito local también llamadas locales son aquellas que se declaran dentro de funciones, su valor sólo se conserva dentro de la función donde fue declarada.

1.11.3 Clases y objetos

Aun cuando los términos objeto y clase se usan de forma indistinta son conceptos diferentes. Una clase es considerada como una plantilla para crear objetos mientras que los objetos son instancias con dicha plantilla. La plantilla indica las características que tendrá un objeto, así como las propiedades.

La clase Array: permite crear arreglos de variables u objetos. Un arreglo consiste en espacios contiguos en la memoria donde se almacenan los elementos del arreglo. Algunas funciones de la clase array son: *pop*, *push*, *sort*, *reverse*, *shift*, *unshift*, *join*, *toString*, *valueOf*. Para emplear un elemento del arreglo se necesita conocer su índice.

La clase Math: permite aplicar diversas formulas sobre variables numéricas, por ejemplo elevar un número a una potencia, obtener el máximo y el mínimo entre dos números, obtener la raíz cuadrada, etc. También define algunas constantes por ejemplo, el valor de PI, la constante de Euler, etc.

Constructores: Los constructores permiten crear instancias de una clase las cuales se pueden asignar a variables de JavaScript. Para invocar el constructor de una clase se emplea la palabra clave *new* seguida del nombre de la clase.

La clase Date: permite obtener la fecha actual del equipo del usuario, manipular los datos de una fecha para obtener información como el día del mes, el día de la semana, la hora, los minutos, etc. También permite crear fechas y manipularlas. Para crear una instancia de la clase Date se invoca a su constructor.

La clase String: Las cadenas de símbolos también pueden ser manipulados como objetos, por lo tanto existen constructores para crear cadenas y también existen métodos para manipular las cadenas. Algunos de estos métodos permiten convertir cadenas en arreglos, otros permiten cambiar los símbolos de la cadena de minúsculas a mayúsculas y viceversa. La creación de cadenas no requiere estrictamente la invocación de constructor, es posible crear cadenas declarando una variable y asignándole una cadena de símbolos encerrados entre comillas. Esta se comportará como una instancia de la clase String y podrá hacer uso de los métodos de clase.

1.11.4 Objetos del navegador

Los objetos del navegador son aquellos que reflejan las operaciones que se pueden realizar con el entorno del navegador. Estos objetos son el navegador, las ventanas, el documento y el historial. Estos objetos también tienen propiedades y métodos a diferencia de los objetos vistos anteriormente, éstos ya han sido creados cuando se abre algún documento en el navegador.

El objeto Window: Representa o hace referencia a la ventana del navegador, cada ventana podría tener dentro de uno o varios frames, también tienen asociado un nombre. Los métodos de una ventana permiten abrir ventanas emergentes de alerta, de confirmación, para pedir un dato e incluso es posible que la ventana principal abra nuevas ventanas con características especiales por ejemplo, ventanas de un tamaño arreglado, con o sin barras de herramientas o de desplazamiento.

El objeto *Historial*: El historial del navegador guarda una lista de los sitios que se han visitado. Usando JavaScript es posible moverse en el historial hacia atrás y hacia adelante. El siguiente ejemplo muestra la sintaxis para moverse hacia atrás empleando un número negativo: `window.history.go(-1);`

Dado que el historial se guarda como una lista, imagine que ésta ha sido numerada y mediante el número -1 se le indica que la ventana abra la dirección que se encuentra una posición atrás de la actual, dicho de otra manera, la posición actual menos uno. De la misma forma se puede indicarle que “se mueva” hacia delante con un número positivo.

El objeto *Location*: permite obtener información acerca de la dirección del sitio. Incluso es posible cambiar la dirección para crear un efecto de redireccionamiento automático a otras páginas.

El objeto *Navigator*: proporciona, mediante sus propiedades, datos sobre el navegador que se emplea. Las propiedades que contiene dependen del navegador que se use y en algunos casos también dependerá de la versión de los mismos.

El objeto *Document*: Representa el documento que se despliega en el navegador y está compuesto de diversos objetos agrupados en arreglos los cuales representan diversas entidades que puede contener un documento HTML, por ejemplo, tiene un arreglo con las imágenes que contiene el documento, otro arreglo contiene los formularios del documento, un arreglo más contendrá los vínculos del documento. También cuenta con algunas propiedades como el color de fondo, el título, etc.

Formularios de un documento: Están compuestos por campos de texto, listas, botones, etc. Cada uno de estos elementos es representado en JavaScript como componentes de un formulario y de acuerdo a su tipo tendrán diferentes propiedades y métodos.

Comportamientos Predefinidos en Adobe Dreamweaver: Los comportamientos permiten la interacción del usuario con la página o sitio web, para cambiar alguna propiedad o ejecutar determinadas tareas. El comportamiento funciona con un evento y una acción. Los eventos son funciones llamadas de JavaScript, que provocan la ejecución o no de acciones. Las acciones son comandos o reglas que modificarán a un objeto.

1.11.5 Uso de JavaScript con Marcos Incorporados

Un marco incorporado es un objeto inmerso en un documento HTML que permite el despliegue de un URL. Es como tener una página incrustada en el documento, esta página incrustada no afecta del todo al documento que la contiene. Las propiedades de un marco incorporado pueden ser manipuladas usando JavaScript por ejemplo para cambiar el URL, para modificar el tamaño del marco, etc. El documento que contiene al

marco puede obtener una referencia a él mediante la función “*getElementById*” del objeto document, enviando como parámetro el identificador del marco deseado.

Para cambiar el contenido de un marco incorporado basta con modificar el atributo “*src*” del objeto. Aunque no es necesario JavaScript para cambiar el URL que se muestra dentro del marco pues basta con emplear el atributo “*target*” de las etiquetas para crear hipervínculos. Los hipervínculos invocaran a una función de JavaScript que modificará el atributo “*src*” de la etiqueta marco. Esta función ha de declararse en el encabezado del documento HTML. Al pulsar los hipervínculos se mostrará en el marco el URL seleccionado.

También es posible modificar el documento contenido en el marco. Para obtener la referencia al documento puede emplearse una de dos propiedades, una de ellas es “*contentDocument*” y la otra es “*document*” del objeto “*contentWindow*”. La elección depende del navegador ya que no todos permiten la manipulación del documento mediante la misma propiedad. Es importante mencionar que solamente es posible manipular los documentos que pertenezcan al mismo dominio del sitio que contiene el marco por razones de seguridad. También es posible escribir en el documento que se encuentra incorporado en el marco. Para ello deben utilizarse las funciones para abrir y cerrar el documento, con la instrucción “*open*” se abre el documento y con la instrucción “*close*” se cerrará. Una vez abierto se emplea la función “*write*” para escribir dentro del documento.

Menús en JavaScript

Existen diversas técnicas para simular menús en aplicaciones JavaScript, todas las técnicas emplean como base JavaScript y hojas de estilo para crear un menú. También se puede crear funciones para ocultar y mostrar los menús, y darles una apariencia más agradable a la vista cambiando los atributos de las tablas diseñadas en HTML con los CSS, como los colores, fondos, tamaños, etc.

1.12 USO DE BASES DE DATOS EN SITIOS WEB

Un Sistema Manejador de Base de Datos (Data Base Management System: DBMS) es aquél que proporciona aislamientos de datos, para actuar como un intermediario entre la vista de una aplicación y la estructura física de almacenamiento. Un DBMS es un conjunto de rutinas, funciones, métodos de acceso, áreas de trabajo, almacenamiento y control, requeridas para el tratamiento del manejo de información bajo el concepto de bases de datos. Un DBMS extrae datos de los programas y permite diferentes vistas lógicas de los mismos datos, los cuales son independientes de la forma en que se distribuya físicamente la información en disco. Un problema importante que se presenta al usuario es el empleo efectivo de esos sistemas; sin embargo, un elemento vital para lograr una buena

utilización y respuesta de un DBMS es el diseño de la base de datos; la cual, es una metodología que a partir de un conjunto de elementos de información brinda una estructura flexible para un DBMS.

Objetivos de un DBMS

En términos generales, algunos de los objetivos principales del DBMS son: eliminar la redundancia, mantener la consistencia de los datos, resolver los problemas de concurrencia y regular el acceso a los datos. La administración de las bases de datos fusiona la necesidad que tiene el usuario por técnicas más complejas para la manipulación de los datos con la creciente tecnología de las computadoras. Un DBMS hace más fácil la puesta al día y la modificación de las bases de datos existentes, reduce de modo substancial el espacio necesario para almacenar y procesar información, su uso mejora significativamente el manejo de bases de datos en comparación con los métodos comunes de manipulación de archivos.

Elementos de un DBMS

- ❖ **DDL (Data Definition Language).** Es el medio para definir el contenido y formato de la Base de Datos.
- ❖ **DD (Data Dictionary).** Contiene metadatos, es decir, datos de los datos.
- ❖ **DML (Data Manipulation Language).** Lenguaje que permite la manipulación de los datos.

Conceptos Básicos

- ❖ **Datos:** Los datos son una representación del mundo real; únicamente identificarán la porción que ejemplifique un hecho particular de interés para una o más aplicaciones.
- ❖ **Información:** Es el resultado obtenido de la transformación de los datos para una persona dada con un fin determinado o para satisfacer un requerimiento específico.
- ❖ **Entidad:** Es una persona, lugar, cosa, evento o concepto acerca del cual se desea registrar información. Ejemplos: clientes, proveedores, inventarios, empleados, etc.
- ❖ **Atributos:** Son las características básicas que identifican a la entidad (campo de datos).
- ❖ **Valor de los datos:** Es la información o los datos mismos contenidos en cada campo de estos valores que toman los campos de datos, pueden ser cuantitativos, cualitativos o descriptivos, dependiendo de la manera como los campos de datos (atributos) describan a la entidad.
- ❖ **Registros:** Es una colección de datos relacionados, referentes a una entidad.
- ❖ **Archivos de Datos:** Los registros de datos forman un archivo de datos. Así, un archivo es un conjunto de registros similares.

1.12.1 MySQL

Es el generador de bases de datos por excelencia que permite a PHP y a Apache trabajar de forma conjunta para acceder y mostrar datos en un formato legible en un navegador. Se trata de un servidor SQL (Lenguaje de consulta estructurado) diseñado para grandes cargas y que puede procesar consultas muy complejas. Como sistema de bases de datos relacional, MySQL permite combinar multitud de tablas diferentes para optimizar la eficacia y la velocidad.

Modelo Relacional

El DBMS (Relational Model of Data for Large Shared Banks) relacional consiste en una colección de tablas, cada una de las cuales está asignada con un nombre único, cada una de éstas representa una entidad del mundo real y a su vez se identifica por filas y columnas. Una fila comprende una relación entre un conjunto de valores. Las columnas, campos o atributos de una tabla pueden arreglarse en cualquier orden sin afectar el significado de los datos. Una tabla siempre tiene un campo o grupo de campos que sirven como llave única; debido a que dos registros no pueden ser idénticos, aquella que se elija como llave única de identificación se denomina llave primaria. Si en un conjunto de tablas que constituyen una base de datos relacional un atributo sirve como llave primaria de una tabla y aparece como campo de otra, entonces se define como llave foránea.

Modelo Básico

Un DBMS utiliza un modelo de datos para definir la estructura fundamental de los mismos. Un modelo de datos expresa las entidades y sus relaciones y es la herramienta usada para representar la organización conceptual de los datos.

- a) **Entidad:** Una entidad es una persona, lugar, evento u objeto identificado en forma única y del cual se registra información y que además cae dentro del alcance del sistema, por lo que éste debe mantener, correlacionar y desplegar información. Las entidades son sustantivos y son representadas por medio de las tablas, pueden ser tangibles (empleados, alumnos, piezas, artículos o lugares) o intangibles (un suceso, un nombre de actividad, la cuenta de un cliente o un concepto abstracto).
- b) **Tabla:** Es un arreglo de dos dimensiones compuesto por renglones y columnas. Una columna contiene un tipo de datos y un renglón incorpora similar tipo de información; ambos, contienen datos pertenecientes a una entidad.
- c) **Atributos:** Son las características propias de la entidad, la misma posee atributos básicos que la caracterizan. Los atributos se modelan como columnas

de la entidad. La forma de diferenciar entidades es por medio de atributos, y cada una de ellas debe tener por lo menos un atributo diferente.

- d) **Tupla:** Colección ordenada de uno o más elementos de datos que forman un registro. Es el renglón n de la tabla.
 - e) **Grado:** Son los diferentes valores que pueden tomar la tabla con respecto al dominio, es decir, la amplitud de la tabla en cuanto al número de atributos.
 - f) **Dominio:** Es la colección de valores de los cuales uno o más atributos obtienen sus valores reales.
 - g) **Llave Primaria:** Es un atributo o un conjunto de atributos (clave compuesta) que identifican en forma única una instancia de una entidad. Debe ser estática y no volátil es decir no sujeta a cambios. La relación entre dos entidades se especifica por la llave primaria de la otra entidad (llave foránea fk).
 - h) **Llaves Foráneas:** Es la llave primaria de una tabla y al mismo tiempo forma parte de otra tabla únicamente como atributo.
 - i) **Asociaciones:** Una asociación es la unión o enlace entre dos o más entidades (u otras asociaciones), las cuales se encuentran dentro del alcance del sistema, y por ello el sistema debe mantener, correlacionar y desplegar información.
1. **Asociación Uno a Uno (1:1)**, las ocurrencias de una entidad se pueden relacionar sólo a una ocurrencia de la otra entidad.
 2. **Asociación Uno a Muchos (1:M)**, se dice que una relación entre entidades es uno a muchos, si las ocurrencias de una entidad están relacionadas con diversas ocurrencias de otra entidad.
 3. **Asociación Muchos a Muchos**, ocurre cuando se asocian una ocurrencia en una entidad, con muchas ocurrencias en la otra entidad y viceversa.

Normalización

La normalización es una técnica para el diseño de bases de datos, representativa del tipo de métodos que toman de entrada una lista de campos y las asociaciones entre ellos. La normalización de datos constituye una metodología para arreglar campos en tablas, de manera que se elimine la redundancia entre los campos no llave y donde cada una de las tablas resultantes se ocupa de un área de conocimiento.

SQL es un lenguaje de nivel de conjuntos, las instrucciones SQL definen el conjunto de respuesta que será devuelto.

- ❖ **Data Definition Language (DDL):** Crear tablas, borrar vistas, alterar macros.
- ❖ **Data Manipulation Lenguaje (DML):** Seleccionar filas, insertar, actualizar, eliminar.
- ❖ **Data Control Lenguaje (DCL):** Otorgar para actualizar tablas, revocar para eliminar vistas, entregar para seleccionar macros.

MySQL es un sistema gestor de base de datos relacionales, que además ofrece compatibilidad con PHP, Perl, C y HTML, y funciones avanzadas de administración y optimización de bases de datos para facilitar tareas habituales. Implementa funcionalidades Web, permitiendo un acceso seguro y sencillo a los datos a través de Internet. Este sistema gestor de base de datos incluye capacidades de análisis integradas, servicios de transformación y duplicación de datos y funciones de programación mejoradas.

Se puede decir que MySQL es un sistema cliente servidor de administración de bases de datos relacionales diseñado para el trabajo tanto en los sistemas operativos Windows como en los sistemas UNIX/LINUX. Además, determinadas sentencias de MySQL pueden ser embebidas en código PHP y HTML para diseñar aplicaciones Web dinámicas que incorporan la información de las tablas de MYSQL a la página Web.

1.12.2 Crear una Base de Datos y Tablas

Para crear una base de datos usamos la sentencia *CREATE DATABASE* y el nombre que se le asignará, como en el siguiente ejemplo:

```
mysql> CREATE DATABASE ESCUELA;
```

Una vez creada la base de datos con el comando *USE*, permite ingresar dentro de la base de datos que se creó.

```
mysql> USE ESCUELA;
```

Después se puede utilizar el comando *SHOW TABLES*, que permite mostrar el contenido de tablas que ya existen dentro de la base de datos.

```
mysql> SHOW TABLES;
```

Las tablas a considerar de la base de datos ESCUELA, son: DEPARTAMENTO, CLASE, ESTUDIANTE, MATRICULA, CLAUSTRO, y PERSONAL. Se usa la sentencia *CREATE TABLE* para generar las tablas y definir los tipos de datos para cada columna que conformarán las tablas.

```
mysql> CREATE TABLE departamento (  
-> Dept CHAR(4) not null,  
-> Dedif CHAR(2) not null,  
-> Ddespacho INTEGER not null,  
-> Dchfno CHAR(3),  
-> PRIMARY KEY (Dept));
```

CHAR se utiliza para los campos que usan texto en sus registros, e INTEGER es para definir números enteros. Cuando se definen valores no vacíos se utiliza NOT NULL. Para confirmar que la tabla fue creada, se usa la sentencia DESCRIBE.

```
mysql> DESCRIBE departamento;
```

1.12.3 Inserción de Datos

La inserción de datos se realiza una vez creada la tabla, a través de la sentencia *INSERT* y *LOAD DATA*

```
INSERT INTO departamento VALUES ('THEO','HU',200,'10');
```

Los valores de tipo cadena deben estar entre comillas simples a diferencia de los tipos numéricos que no se requiere las comillas. Si se declara un campo de tipo no null es un requisito indispensable no dejar valores nulos ya que no se podrá ingresar valores a la tabla.

Si se parte de una tabla vacía, se puede crear un archivo de texto “departamento.txt” que contenga los registros de los campos definidos, con la sentencia de *CREATE TABLE* definiendo cada tabulador como una nueva columna y los saltos de línea es el separador de registros. Para cargar el contenido del archivo en la tabla departamento, se utiliza el siguiente comando:

```
mysql> LOAD DATA LOCAL INFILE “departamento.txt” INTO TABLE departamento;
```

La sentencia *LOAD DATA* permite insertar todos los datos de la tabla por medio de un separador de columnas (tabulador), y saltos de línea es otro separador de registros.

1.12.4 Selección de Datos

SELECT es la sentencia SQL de uso más frecuente para crear un juego de registros que recuperen la información de una base de datos. *SELECT* extrae las columnas especificadas de una o varias tablas de bases de datos y las utiliza para crear un juego de registros o una nueva consulta. La sintaxis básica es:

```
SELECT NombreColumna FROM NombreTabla WHERE CondicionBusqueda.
```

Permite seleccionar todos los datos de una tabla.

```
SELECT * FROM departamento;
```

Se puede seleccionar solo registros particulares de una tabla.

```
SELECT * FROM departamento WHERE Ddespacho=300;
```

Se puede también utilizar condicionales lógicos AND y OR, dentro de las consultas a la base dentro de sus tablas.

```
SELECT * FROM departamento WHERE Ddespacho=100 AND Dedif="SC";
```

La consulta anterior usa el operador lógico AND. Hay También un operador lógico OR:

```
SELECT * FROM departamento WHERE Ddespacho=100 OR Dedif="SC";
```

Dentro de una tabla, se puede hacer consultas a una columna particular usando el nombre de dicha columna en la que estamos interesados. Si son más de una columna se puede utilizar comas para separarlas dentro de una misma sintaxis.

```
SELECT Dept, Dchfno FROM departamento;
```

Cuando se realizan selecciones de datos, se puede ver que los registros son mostrados sin ningún orden en particular. Cuando los registros son muchos se requiere ordenarlos para su fácil manejo. Para ordenar los registros, se tiene que usar la cláusula *ORDER BY*.

```
SELECT Dept, Ddespacho FROM departamento ORDER BY Dept;
```

Para modificar los registros existentes en una tabla se utiliza la sentencia *UPDATE*, que tiene la siguiente forma:

```
UPDATE nombre_tabla  
SET columna a cambiar  
WHERE registros a actualizar;
```

La cláusula *WHERE*, ayuda a hacer una previa consulta de los registros contenidos en la tabla, sin llegar a modificar registros que no se deseaban, causando errores futuros.

```
UPDATE frutas SET Nombre_frutas= 'Naranjas';
```

1.12.5 Borrar Datos

La sentencia *DELETE* tiene la siguiente forma:

```
DELETE FROM nombre_tabla  
WHERE registros a eliminar;
```

La cláusula *WHERE* que especifica los registros que se quiere eliminar es opcional, pero si le quitamos eliminaremos todos los registros de la tabla.

```
DELETE FROM frutas WHERE Tipo:frutas='Am';
```

1.13 PHP

PHP (HyperText Preprocessor, Preprocesador de Hipertexto) es un lenguaje de programación scripting que permitirá generar contenidos de forma dinámica del lado del servidor web. El programa PHP es ejecutado en el servidor y el resultado es enviado al navegador. Al ser un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, porque es independiente del browser, sin embargo para que las páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

Para programar en PHP se agrega dentro del código HTML las instrucciones de PHP usando las etiquetas: `<?php ?>` o `<? ?>` o mediante `<script lenguaje="php">` `</script>`

El funcionamiento de las páginas en PHP es el siguiente: el navegador del cliente solicita el documento PHP, luego la solicitud llega al servidor y este localiza el documento, ejecuta el intérprete de PHP y e interpreta el código, una vez ejecutado el código se genera el resultado en HTML y lo devuelve al servidor para que éste lo envíe al cliente, finalmente se muestra el resultado en el navegador del cliente.

1.13.1 Programando Scripts en PHP

Scripts

Teniendo ya instalados PHP y MySQL, y el servidor Apache, se puede realizar la primera prueba de un script en PHP.

```
<html>  
<body>  
<?php  
$myvar = "Primer script en PHP n";  
//Esto es un comentario  
echo $myvar;  
?>  
</body>  
</html>
```

Se guarda el script con la extensión php y se carga al servidor. Para visualizarlo se ingresa en la URL apropiada y se observa el texto "Primer script en PHP". Todas las líneas que se encuentren entre los delimitadores de PHP deben acabar en punto y coma

(;), excepto las sentencias de control (if, switch, while, etc). Las líneas que comienzan con // serán comentarios del programa únicos para esa línea; si se requiere comentar un bloque completo se usara /* - */.

Para que se envíe el mensaje escrito se usa la instrucción *echo*, también se puede usar *printf*. Al final de haber escrito el mensaje agregamos una *n*, esto sirve para indicarle al navegador que se trata de una nueva línea.

Variables

En PHP (al igual que en JavaScript) no hace falta declarar la variable ya que simplemente anteponiendo el caracter \$ al nombre de la variable se está indicando que es una variable.

Otro hecho que cabe destacar a la hora de programar en PHP y la declaración de variables es que PHP es un lenguaje "CASE SENSITIVE" es decir que diferencia entre mayúsculas y minúsculas y debido a esta razón no sería lo mismo *\$miVariable=valor*; que *\$MiVaRiABle=valor*; ya que PHP lo interpretaría como dos variables completamente diferentes.

Los tipos de datos posibles que puede almacenar una variable son los siguientes: enteros positivos o negativos (integer), números decimales o de coma flotante (double), cadenas de texto (string), valores booleanos (true o false), arreglos (array) y object que es un tipo de dato complejo.

Convertir tipos

PHP realiza la conversión de tipos de acuerdo a los operandos y del operador. Por ejemplo si se quisiera sumar la cadena '20' y el número 20 PHP convertiría la primera variable a numérica y de esa forma se sumarían 20+20=40.

Existen dos funciones especiales del propio lenguaje las cuales permiten saber el tipo de variable que se están usando y también pueden convertir el tipo de variable:

- ❖ **gettype()** Recibe el tipo de variable que es .
- ❖ **settype(\$variable,'tipo variable')** Transforma el tipo de variable del modo actual a el modo que se le introduzca.
- ❖ **Constantes:** Para definir constantes se usará la siguiente sintaxis: define("nombre de la constante", valor);

Operadores

Operadores aritméticos: suma (+), resta (-), multiplicación (*), división (/), resto de dividir dos valores (%), incremento (++), decremento (--).

Operadores de asignación: asignación de valor (=), realiza la suma de la derecha con la izquierda y la asigna a la derecha (+=), realiza la resta de la derecha con la izquierda y la asigna a la derecha (-=), realiza la multiplicación de la derecha con la izquierda y la asigna a la derecha (*=), realiza la división de la derecha con la izquierda y la asigna a la derecha (/=), se obtiene el resto y se asigna (%=), concatena el valor de la izquierda con la derecha y lo asigna a la derecha (.=).

Operadores Lógicos: Negación (!), y (and), o (or), verdadero si alguno de los dos es true pero nunca ambos (xor), true si ambos lo son (&&), true si alguno lo es (||).

Operadores Condicionales: Comprobar si dos números son iguales (==), comprobar si dos números son diferentes (!=), mayor que, devuelve true en caso afirmativo (>), menor que, devuelve true en caso afirmativo (<), mayor o igual (>=), menor o igual (<=).

1.13.2 Estructuras de Control

Al igual que los demás lenguajes de programación, PHP contiene sus estructuras de control para realizar acciones a partir de decisiones, bucles, condiciones, etc. Se usan *IF*, *SWITCH*, *FOR*, *WHILE*, *DO WHILE*, *BREAK* y el *CONTINUE*.

Funciones

Las funciones deben ser colocadas siempre antes de realizar la llamada a la función. La sintaxis de una función es la siguiente:

```
function nombre(parámetros){  
    instrucciones de la función  
}
```

Para llamar a la función sería de la siguiente forma: nombre(parámetros)

Un hecho relevante que cabe destacar es que las variables que se declaren dentro de la función sólo existirán o tendrán dicho valor dentro de la función.

Existen casos en los cuales no se sabe el número de parámetros que se le pasarán a la función y en estos casos se debe usar las funciones creadas al efecto como son:

- ❖ **func_num_args()** Número de parámetros que se le han pasado a la función.
- ❖ **func_get_args()** Devuelve un elemento de los que forman la lista de argumentos.

Cargando Código desde otro Archivo

Existen ocasiones donde es necesario usar el mismo código en diferentes scripts, de varias formas se pueden guardar cierto código en un archivo para ser cargado en estos scripts cuando se necesiten. En PHP existen dos funciones:

```
require("archivo");
```

Cuando se incluye un archivo con `require` el intérprete abandona el modo PHP y entra en modo HTML, una vez abierto el fichero lo incluye hasta su última línea y posteriormente abandona el modo HTML para volver a posicionarse en el modo PHP. Su principal diferencia con `include()` es que no permite la carga condicional.

```
include("archivo");
```

Permite la carga condicional, es decir, que se puede cargar un archivo u otro según si se cumple o no una condición.

1.13.3 Matrices

En PHP las matrices o arreglos son de suma importancia para almacenar fatos similares. Un array es un conjunto de variables agrupadas bajo un único nombre. Cada variable dentro de la matriz se denomina elemento. Dentro de la misma matriz pueden existir variables de diferentes tipos y no es necesario que sean todas del mismo tipo. Existen dos tipos diferentes de arreglos:

- ❖ **Indexada:** Aquella cuyo acceso a los elementos se realiza por la posición que ocupan dentro de la estructura (se inician siempre desde la posición 0). Ejemplo:
\$amigos[0]
- ❖ **Asociativa:** Es aquella en la que los elementos están formados por pares clave-valor y el acceso se realiza proporcionando una determinada clave. Ejemplo:
\$amigos['edad']

Para crear matrices en PHP existen dos formas:

De forma implícita, que consistiría en indicarle el elemento (ya sea proporcionando su posición o su clave).

Ejemplo: \$nombres[0]='Javier';

En caso de no indicarle una posición el array tomara el valor siguiente al último valor introducido. *Ejemplo: \$nombres[]='Lucas' // tomaría como valor 1 ya que lo ultimo introducido era 0.*

Mediante array() en el cual se le asignan los elementos como parámetros. En caso de matriz indexada toman la posición que ocupan en la creación de la matriz, mientras que los de la matriz asociativa se les asignan su valor mediante "=>".

Ejemplo: `$amigo=array('Nombre'=>'Jose','Direccion'=>'Neopatria 21');`

Cabe destacar que PHP no sólo se limita a la existencia de matrices por sí solo sino que existen matrices de matrices, o lo que es lo mismo, matrices multidimensionales.

Ejemplo: `$amigos[2]['Pedro']`

Para acceder al contenido de una matriz indexada el recorrido se realizara mediante un bucle y para ello se debe saber el número de elementos totales que posee la matriz. Para ello se hace uso de la función `count(variable)` donde `variable` representa la variable de la que se quiere obtener el número de elementos. Si `variable` es una matriz devuelve el número de elementos que tiene, devuelve 1 si solo tiene un elemento (aunque no sea matriz) y 0 si no tiene ningún valor.

Otra función que permite saber el número de elementos es `sizeof(matriz)`. Para acceder a los elementos de una matriz asociativa se debe usar la función `each()` que recupera el par formado por clave y valor y además avanza una posición de puntero. Su sintaxis es `each(matriz)` y los valores que devuelve la matriz asociativa son los siguientes: 0 (nombre de la clave), 1 (valor asociativo de la clave), `key` (nombre de la clave), `value` (valor asociado a la clave).

La función que realiza el constructor `list(variable1,variable2...variableN)` es asignar los valores del elemento actual de una matriz a las variables indicadas como parámetro.

Cuando se trata de matrices indexadas la navegación es sencilla ya que tan sólo basta acceder al elemento que se desea mostrar, pero al tratarse de alguna matriz asociativa no se puede aplicar el mismo tratamiento. Para ello existen un conjunto de funciones prefabricadas que permiten realizar multitud de acciones: `reset(matriz)`; el puntero interno vuelve a la primera posición, `end(matriz)`; el puntero interno va a la última posición, `next(matriz)`; el puntero va al elemento siguiente, `prev(matriz)`; accede al elemento anterior, `current(matriz)`; devuelve el contenido del elemento actual.

1.13.4 Formularios

Los Formularios forman parte del lenguaje HTML. Todo formulario comienza con la etiqueta `<FORM ACTION="lo_que_sea.php" METHOD="post/get">`. Con `ACTION` se indica el script que va procesar la información que se recoge en el formulario, mientras

que *METHOD* indica si el usuario del formulario va a enviar datos (post) o recogerlos (get). La etiqueta *<FORM>* indica el final del formulario.

A partir de la etiqueta *<FORM>* vienen los campos de entrada de datos que pueden ser: Cuadro de texto, cuadro de texto con barras de desplazamiento, casilla de verificación, botón de opción, menú desplegable, botón de comando, campo oculto, etc.

Este último tipo de campo resulta especialmente útil cuando que remos pasar datos ocultos en un formulario.

Todos los tipos de campo tienen un modificador llamado *name* , que no es otro que el nombre de la variable con la cual se obtendrán los datos en el script indicado por el modificador *ACTION* de la etiqueta *FORM* , con *value* establecemos un valor por defecto.

1.13.5 Conexión con MySQL

MySQL es uno de los gestores de bases de datos más usados en entornos en los cuales se emplea PHP, ya que PHP dispone de numerosas funciones que se compaginan perfectamente con MySQL. La forma genérica de obtener información de tablas en Mysql es la siguiente:

- ❖ Conexión con el gestor
- ❖ Preparación de la consulta SQL.
- ❖ Ejecución de la consulta.
- ❖ Procesamiento del resultado obtenido en el cursor
- ❖ Liberación de recursos (esta es opcional, aunque es recomendable).
- ❖ Cierre de la conexión con el gestor.

Para realizar éstas y otras muchas más cosas se dispone de las siguientes funciones:

Función	Descripción
mysql_connect("host","usuario","password")	Establece la conexión con el servidor. Recibe el host y el usuario y contraseña con el que debe conectar.
mysql_select_db("base de datos",conexión)	Selecciona la base de datos sobre la cual se va a trabajar
mysql_query(consulta,conexión)	Ejecuta la consulta SQL indicada como primer parámetro.
mysql_num_fields(cursor)	Devuelve el número de atributos que figuran en el cursor que se le pasa como parámetro y en el que se almacena el resultado de la consulta

mysql_fetch_row(cursor)

Avanza a la siguiente posición de la fila en cursor. Devuelve un array que contiene en sus celdas cada uno de los valores de los atributos de la fila.

mysql_free_result(cursor)

Libera los recursos asociados al cursor.

mysql_close(conexion)

Cierra la conexión establecida con mysql_connect.

Una de las ventajas que proporciona la altísima integración que PHP y MYSQL tienen es la existencia de funciones que permiten al programador acceder a las diferentes estructuras que conforman la base de datos. Algunas de las funciones son:

Función

Descripción

mysql_list_dbs(conexion)

Devuelve en un cursor los nombres de las bases de datos disponibles en el servidor al que se haya conectado con mysql_connect

mysql_list_tables(base_datos,conexion)

Devuelve en un cursor los nombres de las tablas disponibles en la base de datos.

mysql_tablename(cursor,numero_fila)

Devuelve el nombre de la tabla o base de datos en la que está el cursor indicado

mysql_field_name(cursor,numero_col)

Devuelve el nombre del campo cuyo índice se pasa como segundo parámetro

mysql_field_type(cursor,numero_Col)

Devuelve el tipo del campo cuyo índice se pasa como segundo parámetro

mysql_field_len(cursor,numero_col)

Devuelve la longitud del campo cuyo índice se pasa como segundo parámetro

mysql_field_flags(cursor,numero_col)

Devuelve una serie de indicativos correspondientes a características del atributo cuyo índice se pasa como segundo parámetro

mysql_affected_rows(conexion)

Devuelve el número de filas afectadas por una actualización o borrado

mysql_change_user(usuario,password)

Cambia de usuario

mysql_create_db(basedatos)

Crea una base de datos con el nombre pasado por parámetro

mysql_drop_db(basedatos)

Elimina la base de datos pasada por parámetro

mysql_insert_id(cursor)

Devuelve el valor generado para un AUTOINCREMENT

1.14 TEMAS ACTUALES PARA EL DESARROLLO DE SITIOS WEB

Los administradores de contenidos desde hace años han cambiado los procesos de creación y consumo de información. Conocidos en inglés como CMS (Content Management System) permiten gestionar de forma uniforme, accesible y cómoda un proyecto Web. Las principales funciones de un administrador es facilitar la creación, presentación y distribución de los contenidos. Antes del concepto CMS, los desarrollos dependían de grupos especializados. Hoy día, una persona sin conocimientos técnicos puede obtener resultados similares a un profesional.

El funcionamiento de los CMS se reduce a rellenar formularios y activar opciones, siendo el sistema el que hace automáticamente las tareas, en algunos casos se apoya de un editor HTML WYSIWYG (what you see is what you get, lo que ves es lo que obtienes), lo que facilita aún más la edición de los contenidos, la apariencia y la publicación de recursos. Así, la principal característica de estas herramientas es brindar al usuario la capacidad de poder ser administrador de su proyecto, sin el requisito de saber lenguajes de programación o tener conocimientos específicos de herramientas de desarrollo.

Actualmente existen una gran cantidad de CMS, destacando aquellos que cuentan con una comunidad organizada y amplia. De forma general se dividen en dos grupos, lo que son ofrecidos como un servicio y aquellos que necesitan ser instalados en un servidor. La introducción de los CMS trajo como consecuencia nuevas relaciones entre los usuarios y la información. Y de forma natural también se crea una nueva posibilidad de acceder a los contenidos, el RSS (Really Simple Syndication) permite revisar las actualizaciones sin necesidad de visitar cada sitio de forma individual; permitiendo ver los recursos de centenares.

1.14.1 Características de los CMS

Escalabilidad: La escalabilidad es un aspecto sumamente importante, a través de ella el sistema que se esté desarrollando podrá adecuarse a las futuras necesidades. La forma como los SGC cubren este punto es con la separación de los contenidos, presentación, estructura y manejo de diversos estándares como CSS, XML, WAI, RSS.

Estabilidad: La mayoría de los SGC cubren este aspecto, ya que al abarcar los puntos anteriores y estar en constante actualización, permiten la solución y la disponibilidad de las nuevas funcionalidades, los SGC se clasifican en Blogs, Foros, Portales y Wiki. Así también se clasifican en dos grandes tipos: los comerciales y los gratuitos, con licencia de código abierto.

Seguridad: En internet, los sitios o portales pueden sufrir distintos tipos de ataques que afectan la información que contienen o la estabilidad del propio sistema. Por ello, la mayoría de los sistemas SGC poseen diferentes niveles de seguridad configurables, es

decir, proporcionan mecanismos de autenticación que ayudan a establecer distintos niveles de acceso a los contenidos, acceso a contenidos públicos, accesos a contenidos restringidos, acceso al mantenimiento de los contenidos, acceso administrativo.

El CMS debe proporcionar un ambiente que permita a uno o varios autores generar sus documentos o artículos, de manera amigable. El propósito es trabajar la información con el aspecto real que tendrá, pero manteniendo ciertas restricciones que permitan conservar un formato uniforme, para que se adopte el mismo estilo de página. Normalmente, esto lo cumplen la mayoría de los CMS, con editores como: WYSIWYG (lo que ves es lo que obtienes).

También, debe contar con soporte para ser visto desde cualquier navegador de Internet, como Internet Explorer, Netscape, Opera, Mozilla, etcétera. El tamaño de la página debe limitarse para garantizar que los tiempos de carga sean aceptables para los usuarios. En ese sentido, es necesario especificar los métodos de acceso: módem, cable, etcétera, así como las tecnologías requeridas del lado del cliente: Java, Javascript, Flash, etc., para poder ver el sitio.

Ventajas al utilizar CMS

- ❖ Ofrece flexibilidad y escalabilidad en el diseño para su actualización, mediante la inclusión de nuevos módulos, que permiten adaptarse a las necesidades del proyecto.
- ❖ Permite un control de acceso, importante para el buen registro de los usuarios.
- ❖ Permite manejar con facilidad importantes cantidades de información.
- ❖ Incorpora un manejo adecuado de la administración, el mantenimiento y la actualización de todo el sitio, a través de la distribución del trabajo en varios usuarios, con privilegios explícitos para diferentes áreas.
- ❖ Incorpora un manejo sencillo de información dinámica.

1.14.2 Drupal

Drupal es un sistema de gestión de contenido modular y muy configurable, es un programa de código abierto, con licencia GNU/GLP, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la Web, y un énfasis especial en la usabilidad y consistencia de todo el sistema. El diseño de Drupal es especialmente idóneo para construir y gestionar comunidades en Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar diferentes sitios.

Características

Autenticación de usuarios: Los usuarios pueden registrar e iniciar sesión de forma local o utilizando un sistema de autenticación externo como Jabber, Blogger, LiveJournal u otro sitio Drupal. Para su uso en una intranet, Drupal se puede integrar con un servidor LDAP.

Permisos basados en roles: Los administradores de Drupal no tienen que establecer permisos para cada usuario. En lugar de eso, pueden asignar permisos a un “rol” y agrupar los usuarios por roles.

Control de versiones: El sistema de control de versiones de Drupal permite seguir y auditar totalmente las sucesivas actualizaciones del contenido: que se ha cambiado, la hora y la fecha, quién lo ha cambiado, y más. También permite mantener comentarios sobre los sucesivos cambios o deshacer los cambios recuperando una versión anterior.

Objetos de Contenido (Nodos): El contenido creado en Drupal es, funcionalmente, un objeto (Nodo). Esto permite un tratamiento uniforme de la información, como una misma cola de moderación para envíos de diferentes tipos, promocionar cualquiera de estos objetos a la página principal o permitir comentarios –o no- sobre cada objeto.

Plantillas (Templates): El sistema de temas de Drupal separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio web. Se pueden crear plantillas con HTML y/o con PHP.

Agregador de noticias: Drupal incluye un potente Agregador de noticias para leer y publicar enlaces a noticias de otros sitios web. Incorpora un sistema de caché en la base de datos, temporización configurable.

Soporte de Blogger API: La API de Blogger permite que un sitio Drupal sea actualizado utilizando diversas herramientas, que pueden ser “herramientas web” o “herramientas de escritorio” que proporcionen un entorno de edición más manejable.

Independencia de la base datos: Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una “capa de abstracción de base de datos” que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.

Multiplataforma: Drupal ha sido diseñado desde el principio para ser multi-plataforma. Puede funcionar con Apache o Microsoft IIS como servidor web y en sistemas como Linux, BSD, Solaris, Windows y Mac OS X. Por otro lado, al estar implementado en PHP, es totalmente portable.

Administración vía Web: La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional.

Análisis, Seguimiento y Estadísticas: Drupal puede mostrar en las páginas web de administración informes sobre referrals (enlaces entrantes), popularidad del contenido, o de cómo los usuarios navegan por el sitio.

Comentarios Enlazados: Drupal proporciona un potente modelo de comentarios enlazados que posibilita seguir y participar fácilmente en la discusión sobre el comentario publicado. Los comentarios son jerárquicos, como en un grupo de noticias o un foro.

Encuestas: Drupal incluye un módulo que permite a los administradores y/o usuarios crear encuestas on-line totalmente configurables.

Control de congestión: Drupal incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques dependiendo de la carga del servidor. Este mecanismo es totalmente configurable y ajustable.

Sistema de Caché: El mecanismo de caché elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor.

Módulos: Los módulos definen las funcionalidades del sitio. En administrar>módulos se activan o desactivan, en administrar> control de acceso se habilitan los permisos para diferentes roles de usuarios. En administrar > opciones > nombre del módulo se configuran diversas opciones de cada módulo.

Taxonomía: Es la capacidad de categorizar los contenidos del sitio. Se gestionan desde administrar > categorías, donde se pueden crear diferentes “vocabularios”. Cada sistema de categorización es un vocabulario, y cada uno de estos contiene una lista de términos. Cada vocabulario se puede asignar a diferentes tipos de contenido. Por ejemplo, un vocabulario para los foros, otro para las imágenes, otro para las páginas, etc. De esta manera, al editar cada contenido, aparecerá la lista de términos para seleccionar el que hay que asociar al contenido.

Bloques: Los bloques son las cajas visibles en diversas regiones del sitio: lateral izquierdo, derecho, encabezado, final del contenido, pie de página. Algunos bloques vienen predefinidos por los módulos activos. Se pueden escribir bloques personalizados en administrar bloques > añadir bloque, los bloques se activan mediante administrar > bloques. En la configuración de cada bloque se puede determinar su visibilidad, en que páginas aparecerá o en cuales no; también pueden tener otras opciones configurables dependiendo del módulo a que correspondan.

Menús: En administrar > menús se pueden definir menús para facilitar la navegación por el sitio, existe un menú navegación predefinido y se pueden añadir nuevos menús. Cada menú se compone de una lista de opciones y cada opción tienen un título y una descripción que aparece cuando se pasa el puntero sobre el título; también se puede determinar si aparece expandido o colapsado y cuál es su elemento padre. El peso permitirá ordenar las opciones del menú.

Temas: Existen diversos temas que están disponibles en <http://drupal.org/project/themes>. Es importante tener presente la versión de Drupal y la del tema para evitar incompatibilidades. Esta recomendación aplica también a los módulos y a las traducciones. Se descarga el tema deseado, se descomprime y la carpeta resultante se copia en el servidor bajo la carpeta '/themes'. Los temas se configuran en administrar > temas.

El desarrollo de las tecnologías de la información y la comunicación (TIC's) no solo brinda nuevas formas de comunicación como Internet, sino que también proporciona herramientas más amigables, para crear y administrar de manera eficiente, la publicación de información en la red. Los CMS son herramientas que permiten crear, administrar, actualizar y dar mantenimiento a un sitio web de un modo sencillo. Para explotar estas necesidades del desarrollo que se llevará a cabo.

CAPÍTULO 2 INFORME DE DESARROLLO DE PROYECTO PERSONAL “ENTER”

1.1 FASE DE PREPRODUCCIÓN

A continuación se describe la metodología aprendida durante los primeros módulos específicamente Fundamentos de Diseño Gráfico y Planeación de Sitios Web, que darán las bases para establecer los primeros bosquejos y definir los objetivos para la creación del sitio Web, así como también del uso de los diversos elementos tecnológicos que usarán para el proyecto.

1.1.1 Definiendo el Sitio Web

Selección del Tema

La selección del tema se debe a mi actual empleo, trabajo en un cibercafé desde hace año y medio. El negocio es familiar y ha estado en funcionamiento desde hace 8 años aproximadamente, y contaba con una página web diseñada por mi hermano, la cual no era actualizada apropiadamente y estaba un poco mal organizada y desarrollada. Es por ese motivo que me decidí a hacer un rediseño total del sitio, agregando más funciones, servicios y novedades.



Ejemplo del Sitio Web anterior

Objetivos del Sitio

- ❖ El sitio pretende ser un medio de difusión publicitario dentro y fuera del cibercafé.
- ❖ Tiene que ser bastante funcional puesto que será página de inicio en las computadoras del negocio.
- ❖ Accesible desde cualquier plataforma y navegador.
- ❖ La información debe ser de fácil comprensión para cualquier tipo de usuario, aunque algunos temas son más enfocados a personas jóvenes.
- ❖ Principalmente el sitio debe ser muy amigable.

Público Objetivo

El público objetivo es el tipo de usuario al que está enfocada la página:

- ❖ La página estará orientada a ser visitada por todo tipo de usuarios desde niños hasta personas mayores.
- ❖ Algunas partes del contenido estarán más enfocadas a la gente joven, adolescentes y niños.

Usuario Lineamientos

La barra de navegación siempre va a estar visible en cada sección de la página, tanto el encabezado y el buscador en él también estarán siempre visibles, así como el cintillo al final de la página con la información de Copyright, contacto, etc.

1.1.2 Fases de Preproducción

1. **Escoger el tema:** Obviamente no habría página web sin sentido o propósito en específico sin el tema que debe tratar, en este caso es sobre un cibercafé que administro, cuyo nombre es ENTER, de ahí el nombre del proyecto.
2. **Investigación:** A partir del tema del sitio, investigue sitios de internet acerca de cibercafé, en primer lugar debo decir que en México y América Latina existen muy pocos sitios acerca de ello, sólo se encontraron algunos pocos con un buen diseño o funcionalidad. Se encontraron bastantes sitios norteamericanos, europeos y asiáticos, con un gran diseño y apariencia visual agradable, de todos estos sitios se tomaron ideas para implementar.
3. **Escoger los temas que incluirá el sitio:** En este punto se seleccionaron los principales tópicos que se incluirán en el sitio.
 - a) **Inicio o Bienvenida:** Bienvenida a la página.
 - b) **Servicios:** Descripción de los servicios y costos que maneja el negocio.
 - c) **Cafetería:** Zona de productos alimenticios o bebidas que se ofrecen en el negocio.
 - d) **Galería:** Galería fotográfica de las instalaciones, clientes, y/o eventos.

- e) **Foro:** Foro de discusión para uso de los clientes, donde se pretende resolver dudas o cuestionamientos de informática o acerca del negocio.
- f) **Juegos:** Área destinada al ocio y el entretenimiento, donde se hace promoción de un evento en específico del negocio.
- g) **Accesorios:** Lista completa de los accesorios o consumibles que se ponen a la venta al público, de esta área se desprenden otras ligas a cada tipo de consumible en específico.
- h) **Nosotros:** Zona de contacto principal del negocio, contiene el apartado de nosotros, ubicación, contacto y el libro de visitas.

JUSTIFICACIÓN: Se seleccionaron todos estos tópicos puesto que son los más importantes y atractivos para el giro del negocio, no hay duda acerca de que en el futuro próximo incluya más elementos o servicios que deban tener un apartado específico, como la inclusión de secciones especiales para cada tipo de servicio, realización de eventos, asesorías, o un foro de información más completo y dinámico.

- 1) **Selección de tecnologías a implementar:** En este punto se visualizo a futuro que elementos se iban a utilizar para cada sección (flash, php, javascript, etc.), aunque en un principio desconociera realmente que funciones específicas me brindaría cada tecnología, pero poco a poco se fueron descubriendo sus funciones, y obviamente cambiaba el desarrollo para cada sección del sitio.
- 2) **Realización de bocetos:** Se realizaron bocetos simétricos para el sitio, porque se considero que sería lo más viable para el esquema visual que se planteaba y la temática a seguir.
- 3) **Realización del Layout:** a partir de lo realizado en los bocetos se diseñaron varios Layouts cambiando los colores o el acomodo de los elementos, aunque sin perder el orden simétrico, se definieron los tamaños de las áreas de contenido, en base a un esquema visual aceptable de una resolución de pantalla de 800 * 600 pixeles, el Layout definitivo a usar cambio radicalmente en cuanto a colores originales que me había propuesto usar en un principio. Pero se mantuvo los tamaños y la organización original.

JUSTIFICACIÓN: Se considero que el diseño simétrico era el que daba más espacio y libertad de interacción con área de contenido.

- 4) **Definición de tiempos de desarrollo:** Aunque para esta etapa no se hizo un gran énfasis en la práctica, era muy importante por el hecho de obtener las cosas a tiempo y no terminar con un mal trabajo. Afortunadamente se cumplieron los objetivos clave aunque con alguna demora, pero a tiempo para entregar el proyecto.

Justificación de la Preproducción

Todos los elementos aquí mencionados fueron básicos para definir el destino final de la página, los elementos que necesitaba aprender para la producción, contenido a crear, formas u ordenamientos de cajas o áreas para el contenido.

1.1.3 Layout



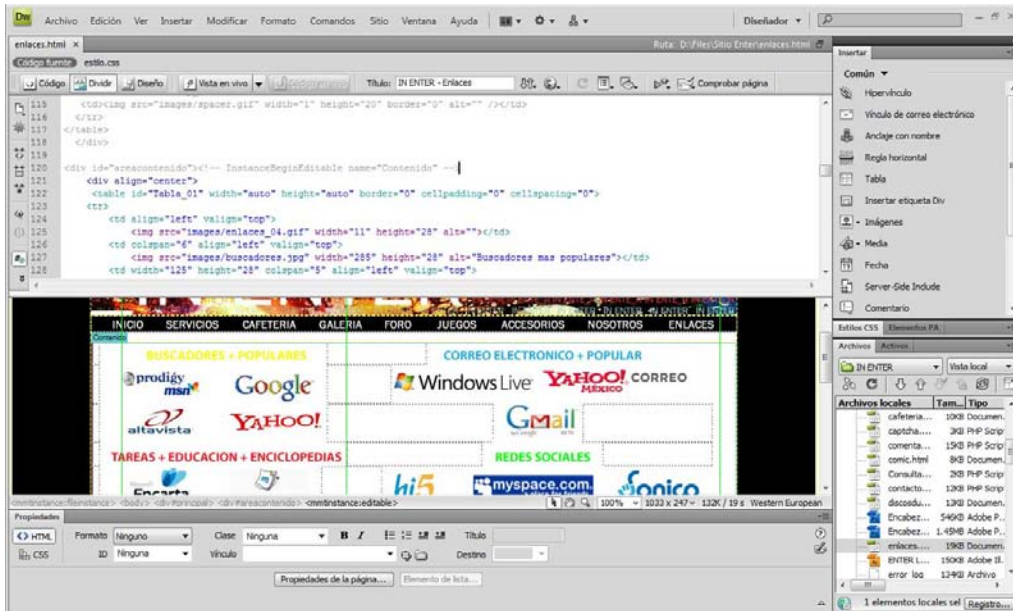
Este es el diseño final del Layout de la página principal, para las demás páginas, simplemente se eliminaron las barras laterales, y el área Contenido Principal, ocupa todo el ancho de la página.

1.2 FASE DE PRODUCCIÓN

A continuación se describirán brevemente los métodos y desarrollos que se hicieron para cada sección del proyecto.

1.2.1 Creación de Contenidos

- a) **Búsqueda de imágenes en la red:** Logotipos principalmente, se usaron para la página de enlaces. Seleccionando los logotipos a usar, se hizo un retoque de imagen de cada uno de ellos, esencialmente en tamaño y color. Posteriormente se acomodaron dentro de Illustrator para hacer un esquema de la página de enlaces. Desde Illustrator se puede seleccionar la opción de Salvar como HTML, lo que permitirá crear automáticamente el código HTML y las imágenes para después ser incorporadas a Dreamweaver para anexarlo a una página con la plantilla realizada.

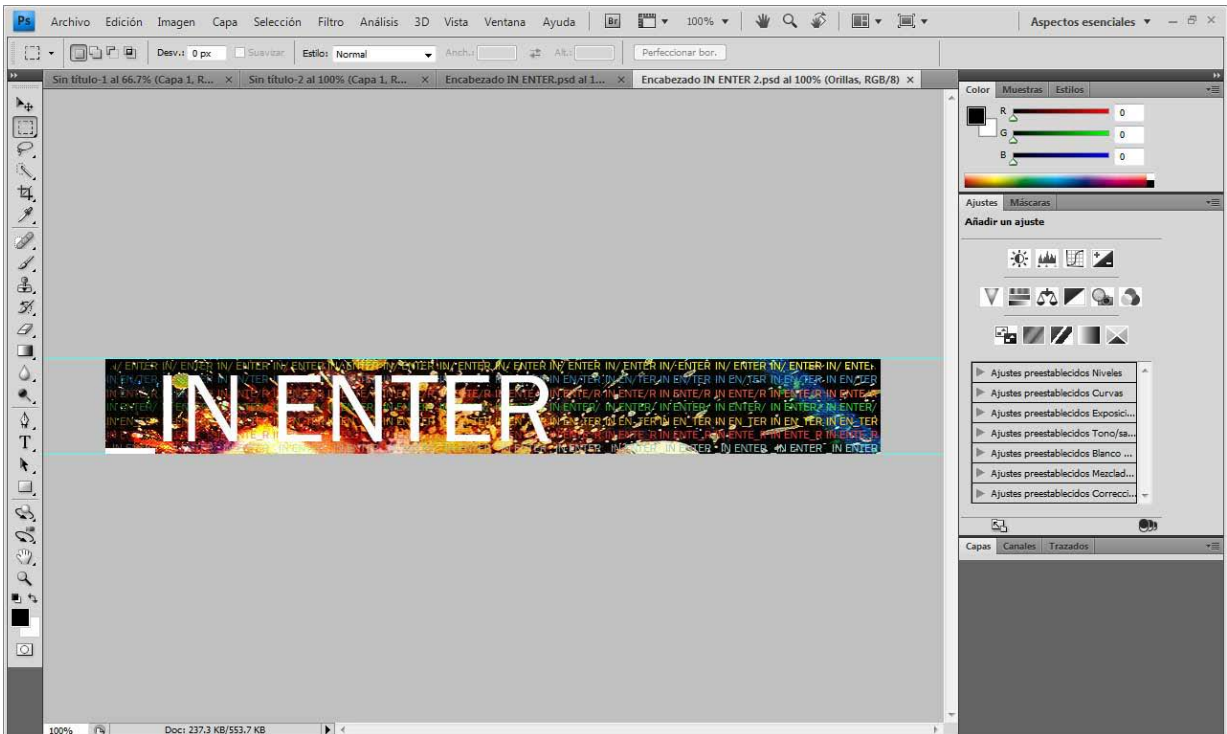


Integración del código e imágenes en Dreamweaver para la página de Enlaces



Página de Enlaces: Ejemplo de manejo de logotipos.

- b) **Encabezado:** En un principio quería hacer el encabezado sencillo, y sin mucho estilo visual, pero finalmente me decidí hacer un diseño atractivo usando Photoshop, manejando estilos y efectos, y opacidades diferentes. Usando también técnicas de superposición de imágenes en diversas capas y modificando las opacidades de las mimas y lograr efectos adecuados.



Diseño de Encabezado en Photoshop

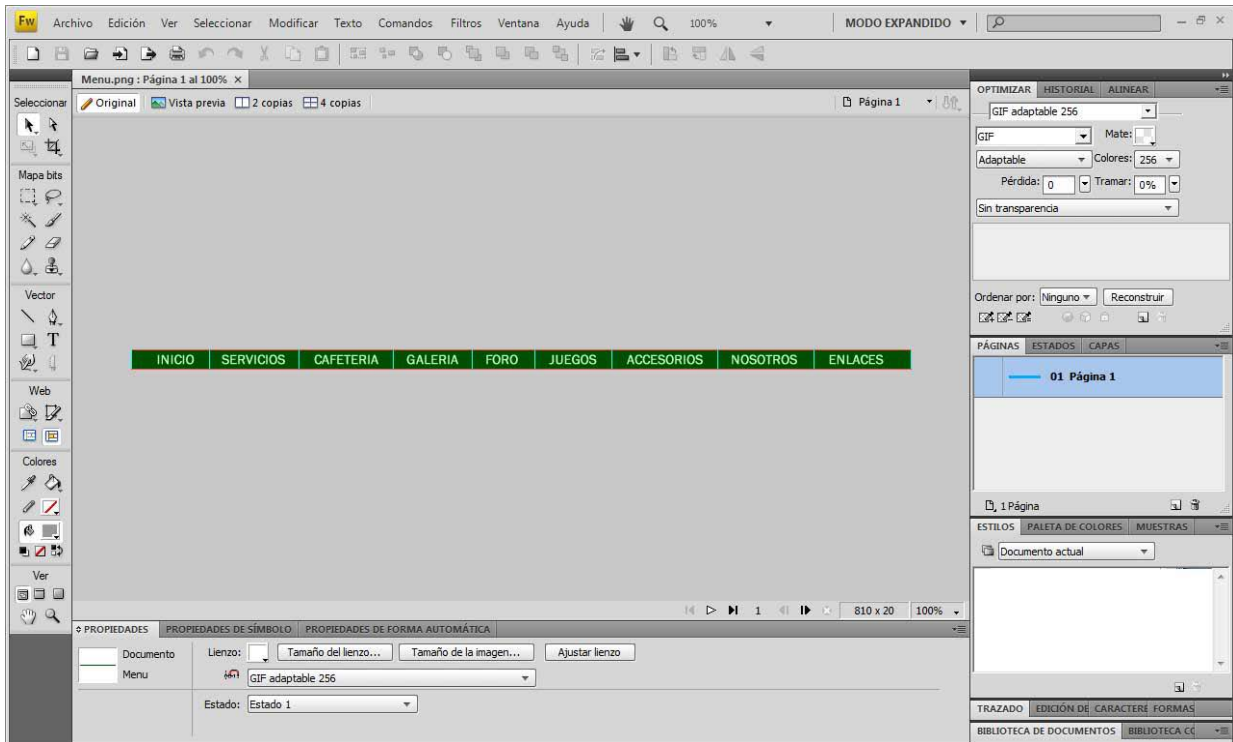


Primer Encabezado

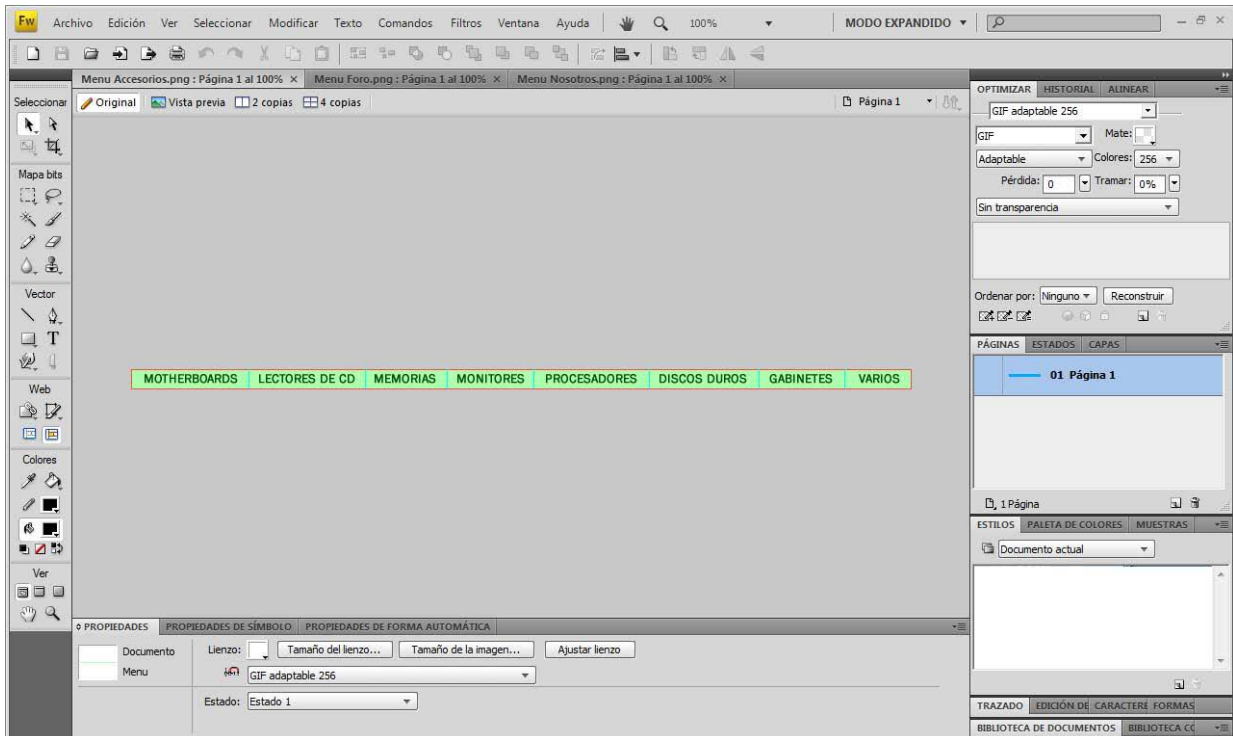


Encabezado Actual

- c) **Menú principal y Submenús:** Para el menú principal y los submenús (foro, accesorios y contacto) se utilizó fireworks, usando efectos de rollovers.

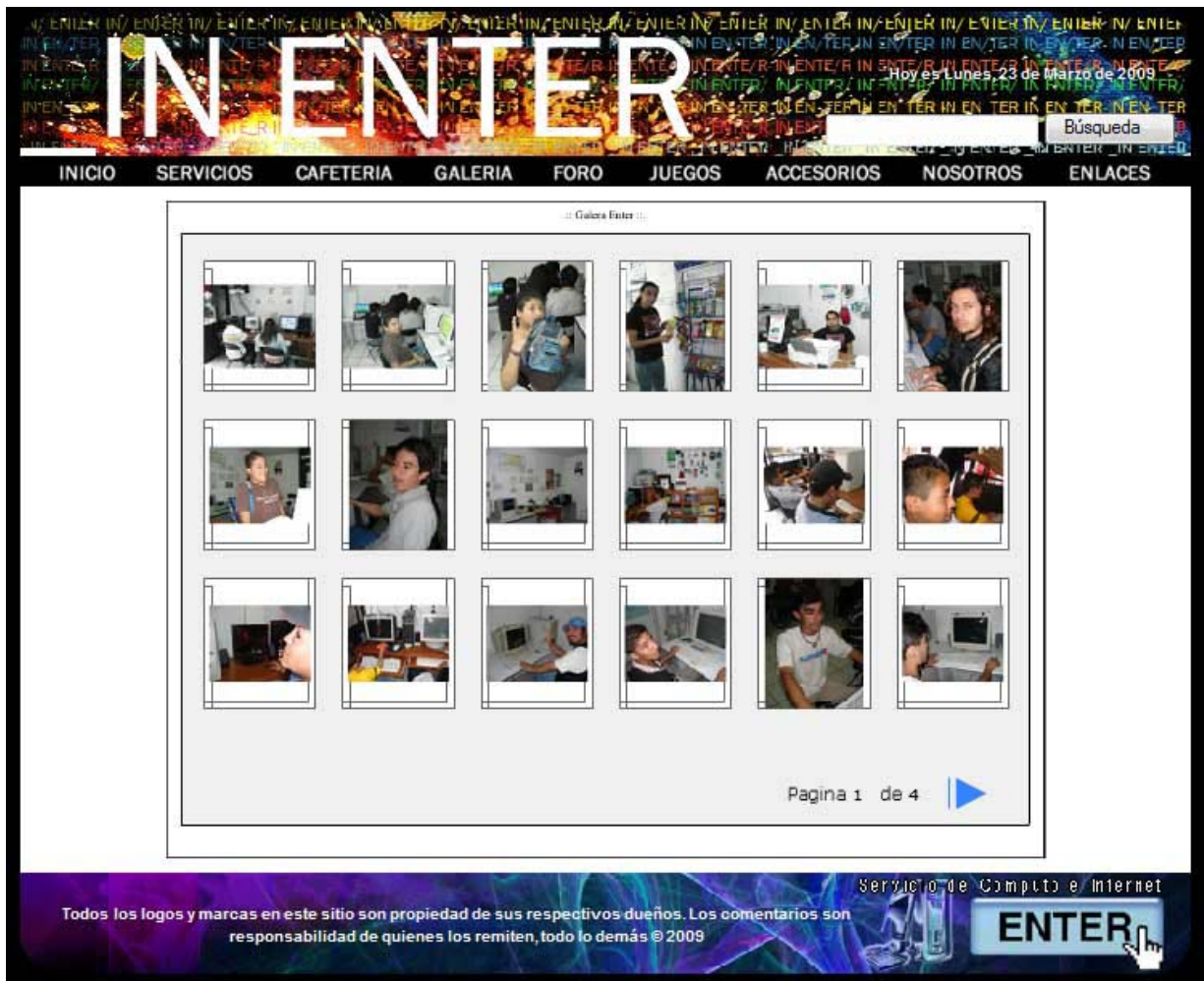


Menú Principal: Ejemplo de desarrollo en Fireworks con efecto de rollover.



Submenú Accesorios: Ejemplo de desarrollo en Fireworks con efecto de rollover.

- d) **Manejo de imágenes:** En cada área del sitio se usaron cierto tipo de imágenes ya sea fotográficas, gráficos o algún otro elemento visual que siempre se retocaba o rediseñaba en Photoshop, desde las fotografías de la galería, hasta algunos botones, o ilustraciones pequeñas, etc.
- e) **Galería Fotográfica:** Esta hecho completamente en Flash usando programación avanzada en ActionScript, las fotografías están retocadas en Photoshop.



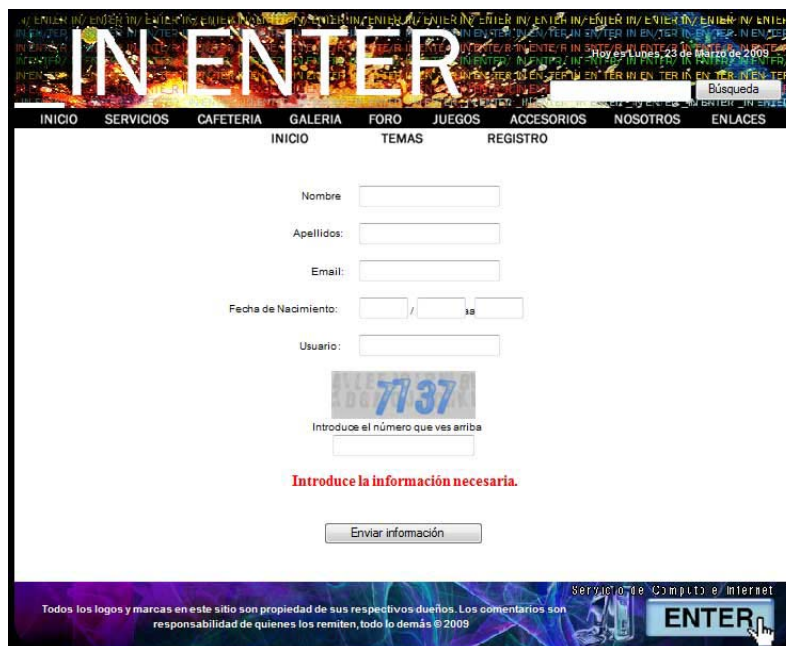
Página de Galería: Ejemplo de desarrollo en Flash, con programación en ActionScript y retoque de imagen en Photoshop.

- f) **Mapa de ubicación:** Este mapa se realizo en Illustrator, posteriormente se importo a Flash, para animarlo y darle efectos de zoom, restauración y arrastre. Usando programación en ActionScript.



Página de Ubicación: Ejemplo de uso de Illustrator, Flash y ActionScript.

- g) **Foro:** El foro esta hecho en php y MySQL, usando funciones de Javascript, para validar los formularios, así mismo de programación avanzada en PHP para la validación y seguridad del mismo. Este mismo foro fue el que se desarrollo para el modulo 13 de PHP.



Página de Registro a Foro: Ejemplo de desarrollo en PHP y MySQL.



Página de Juegos: Ejemplo de desarrollo de edición de video y audio en Vegas 6 y Adobe Audition.

1.2.2 Utilización de Tecnologías

- a) **Editores de texto:** Se uso principalmente Word y el bloc de notas, para realización de escritos y la edición del código HTML, Javascript, PHP y MySQL.
- b) **Illustrator:** Se uso para la realización del Layout principal, y el Mapa de ubicación del negocio, y la estructura principal de la página de enlaces.
- c) **Photoshop:** Se uso principalmente para el retoque de imágenes, reedición de las mismas, y ajuste de tamaños.
- d) **Fireworks:** Se uso para el desarrollo de los submenús con efecto de rollover.
- e) **Editor HTML:** Dreamweaver, para todas las páginas del sitio. Además de que se creó una plantilla especial que servía de base para las demás páginas del sitio.
- f) **Flash:** Se uso para el diseño de la galería fotográfica, minisitio del área de juegos, y mapa de ubicación del local.
- g) **PHP y MySQL:** Base de datos y comunicación para el foro, libro de visitas, y contacto.
- h) **Javascript:** Validación de formularios del foro, libro de visitas y contacto. También se utiliza para la programación en HTML para cargar las imágenes de rollover y algunas funciones especiales, como la fecha en todas las páginas.

1.3 FASE DE POSTPRODUCCIÓN

Antes de colocar el sitio en línea se debe verificar si no existen ligas rotas, ya sea documentos HTML, objetos flash, imágenes, videos, o cualquier tipo de archivo que se use, para la imágenes se debe verificar que cumplan los requerimientos para discapacidades de debilidad visual o nula. Al poner el sitio en línea se hace usando las herramientas vistas en el Módulo 2 Accesibilidad y Usabilidad, se verifican usando TAW y A-prompt. Además de probar el sitio en diferentes navegadores como Internet Explorer versiones 6 y 7, Mozilla Firefox, Safari y Opera, para corregir errores de visibilidad, acomodo de tablas o cajas, etc.

El tiempo aproximado de desarrollo fue de 3 meses, tomando en cuenta las 3 fases de producción. Esto debido al grado de complejidad de programación del foro, la edición de video, la animación y programación de la galería fotográfica, la implementación de todos los HTML que se realizaron, etc.

Calcular el costo para la realización de un proyecto no es tarea fácil, muchas veces depende del tipo de cliente para el que se está trabajando, y la magnitud del proyecto, esto se refiere a la cantidad de páginas que se necesitan, los niveles de profundidad, si se requiere algún sistema especial, bases de datos, animaciones web, integración de audio y video, etc. El costo aproximado del proyecto en cuestión se calcula en \$30,000 pesos, debido a los sistemas empleados, las animaciones en Flash, y la cantidad de páginas realizadas.

En los meses posteriores se debe dar un seguimiento adecuado al sitio, en cuanto a información actualizada, mantenimiento, seguimiento minucioso de las áreas de mayor interés, etc. Y realizar los cambios pertinentes.

CONCLUSIONES

El diplomado de Planeación y Construcción de Sitios Web me ha dejado una experiencia grata y muy constructiva. Los módulos se cumplieron a su debido tiempo y los temas vistos en clase han sido de gran interés general. La metodología usada ha sido la adecuada, aunque algunos puntos se hubieran extralimitado en su momento, ha sido lo correcto para la enseñanza de los temas posteriores.

He aprendido todo el proceso que conlleva diseñar una página web, los estudios previos que serán los cimientos para el diseño, la diagramación, escoger colores, visualizar el público objetivo, y los objetivos principales de la página. Este proceso se llevara a cabo en coordinación con los clientes que pidan el proyecto. Es muy importante establecer los tiempos de desarrollo del trabajo, para dar un tiempo aproximado de término del mismo. Ya una vez establecidas estas bases, comenzamos a trabajar y crear contenido que sea avalado y verificado por el cliente como idóneo para el sitio. Los contenidos deben ser claros y concisos con los objetivos que se han establecido. Las herramientas que se han utilizado durante el diplomado son de gran ayuda para desarrollar completamente el sitio, aunque en la actualidad existan nuevas versiones de dichos programas, es importante darle seguimiento a las nuevas tecnologías de software aplicados al diseño web.

Ya creados los contenidos para la página se va armando la estructura del sitio, es importante tener una buena organización de material en carpetas acorde al tipo y temática de las mismas, sin abusar demasiado también en crear demasiados subniveles de organización, esto puede ser un factor adverso para la navegación del sitio. El armado de la página se ha realizado completamente en Dreamweaver, con la ayuda de estilos CSS y las funciones apropiadas de JavaScript, se puede dar una apariencia visual uniforme a todas las páginas y ahorrar mucho tiempo de desarrollo y programación.

Editar los contenidos de forma idónea permitirá darle dinamismo al proyecto, haciendo uso de las herramientas avanzadas, tanto de Photoshop como Fireworks, se puede lograr efectos sorprendentes que permitirán llamar la atención del usuario, ciertamente sin abusar de los mismos, una imagen demasiado complicada puede ser confusa para los usuarios. El mismo caso se aplica para los videos y audios, en la actualidad existen muchísimos programas editores, pero mis favoritos son los usados en clase: Sony Vega y Adobe Audition, los cuales he usado apropiadamente en la práctica de diseño y edición de video, y para el sitio del proyecto personal, en el video de la presentación de la sección de juegos. Además del uso de Flash y la programación en ActionScript que permiten dar una gran apariencia visual y dinámica al sitio a través de animaciones y videos.

Por último el uso de PHP y MySQL ayudan en estar en contacto con el cliente y usuarios a través de formularios, foros, contactos, etc. Con ayuda de JavaScript y PHP se le puede dar seguridad al ingreso de datos, evitando que usuarios malintencionados logren saturar

los sitios o falsear información. Un buen diseño de base de datos ahorrara muchísimo tiempo de programación al momento de realizar consultas a nuestra base de datos, u obtener la información adecuada para ellas.

Finalmente, el desarrollo del proyecto ha sido de gran beneficio personal tanto laboral como educativo, el sitio puede encontrarse en dos direcciones, en la página del Centro Coapa UNAM <http://zeus.coapa.unam.mx/~reyes/sitioenter/index.html>. o en el sitio oficial del negocio <http://www.in-enter.net> .

BIBLIOGRAFÍA

- ❖ McKelvey Roy, GRAFICOS PARA EL HIPERESPACIO. Edit. Mc Graw-Hill, España, 2001.
- ❖ Weinman Lynda, DISEÑO DE IMAGENES PARA LA WEB. Edit Anaya – Multimedia, España, 2000.
- ❖ Gonzalez Mariano, DISEÑO DE PAGINAS WEB. Edit. McGraw-Hill, España, 2001.
- ❖ Underhahl Keith, DIGITAL VIDEO FOR DUMMIES. Edit. Wiley Publishing, Estados Unidos, 2003.
- ❖ Gradias Michael, TODO SOBRE DIGITALIZACIÓN DE IMÁGENES. Edit. Marcombo, España, 1998.
- ❖ McClelland Deke, LA BIBLIA DE PHOTOSHOP CS. Edit. Anaya, España, 2004.
- ❖ Schulze Patti, MACROMEDIA FIREWORKS MX. Edit. Berkeley, Estados Unidos, 2003.
- ❖ Musciano, Chuck, HTML Y XHTML. Edit. Anaya Multimedia, España, 2001.
- ❖ Schmitt, Christopher. CSS Hojas de estilo en cascada para el diseño Web. Anaya Multimedia. España, 2006.
- ❖ Issi Camy, Lázaro. FLASH MX. Anaya Multimedia, España, 2004.
- ❖ Paniagua Navarro, Antonio. FLASH MX. Anaya Multimedia, España, 2004.
- ❖ Pascual, Francisco. MACROMEDIA DREAMWEAVER MX 2004. Alfaomega, México, 2004.
- ❖ Peña de San Antonio, Oscar. DREAMWEAVER MX 2004. Anaya Multimedia. España, 2004.
- ❖ Zakas, Nicholas. PROFESIONAL JAVASCRIPT PARA DESARROLLADORES WEB. Anaya Multimedia Wrox, 2005.
- ❖ Manger, Jason. FUNDAMENTOS DE JAVASCRIPT. McGraw-Hill Osborne, 1997.
- ❖ DuBois, Paul. PROGRAMACIÓN MYSQL. Anaya Multimedia, Madrid, 2005.
- ❖ Pérez, Cesar. MYSQL PARA WINDOWS Y LINUX. Ra-ma, Madrid, 2007.
- ❖ www.drupal.org.
- ❖ www.drupal.org.es