

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN



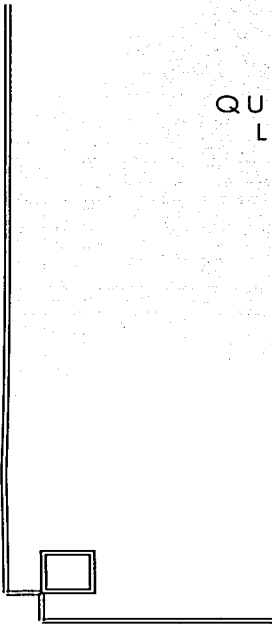
SISTEMAS DE INFORMACIÓN ORIENTADOS A LA WEB  
CASO ESPECÍFICO: "AVISO DE OCASIÓN DEL PORTAL UNAM"

TESIS PROFESIONAL  
QUE PARA OBTENER EL TÍTULO DE:  
LICENCIADO EN INFORMÁTICA

P R E S E N T A N  
SONIA CAROLINA MADRIGAL LOYOLA  
MARÍA DE LOS ANGELES SÁNCHEZ ZARAZÚA

A S E S O R  
L.I. LUZ MARÍA RAMÍREZ ROMERO

México, D.F. 2002





Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**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.

## AGRADECIMIENTOS

### Angeles

"A LA FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN Y A LA  
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO"

Por darme los conocimientos, valores y herramientas necesarias para desarrollarme profesionalmente y para ser mejor persona.

"A SONIA"

Lo más importante por tu amistad, por compartir conmigo la realización de esta tesis, por tu positivismo, por alentarme en los momentos que más lo necesitaba, por tus consejos, por escucharme, por la confianza que para compartir conmigo tus alegrías, tristezas y logros. Y por todos los momentos que pasamos juntas en la Facultad.

"A MIS PADRES"

Por su amor, su confianza, y por enseñarme a ser perseverante, constante y a esforzarme para lograr mis objetivos. Principalmente por haber hecho el esfuerzo y darme la oportunidad de estudiar y terminar una carrera profesional.

"A MIS HERMANOS"

Por su amor, paciencia, apoyo y fe que tienen en mí.

"A MIS TÍOS Y PRIMOS"

Mil gracias por el apoyo y amor que siempre me han brindado y por estar siempre presentes en todos los momentos de mi vida.

"A HEIDI, VERENA, CLAUDIA, ROSALBA, BETO Y OCTAVIO"

Por su amistad, por hacerme reír y pasar momentos muy agradables, por hacerme participe de su vida y por todo el apoyo que nos brindaron para la realización de esta tesis.

**Sonia**

**"A LA UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO"**

Por brindarme tanto como estudiante pero, principalmente, como ser humano.

**"A MI MAMÁ Y MI PAPÁ"**

Origen de mi ser y de la naturaleza de muchos de mis actos, que han procurado ayudarme y apoyarme con el amor que como padres su instinto les ha dictado, para lograr esta meta tan trascendental para mí. Gracias por estar siempre a mi lado en todo momento.

**"A MIS HERMANITOS"**

Gilo, Luis y Nachito, saben cuanto los quiero y todo lo que significan para mí, es por eso que gran parte de lo que hago se sustenta en nosotros, lo que somos y nuestra unión como hermanos.

**"A MI MAMÁ CLARITA, MI PAPÁ NACHO,  
MI MAMÁ LOLA Y MI PAPÁ ANDRES"**

Porque el cariño que me han entregado a lo largo de mi vida, ha sido muchas veces lo que me ha ayudado a seguir adelante.

**"A MIS TIAS, TIOS, PRIMOS Y PRIMAS"**

Porque su apoyo y amor más que de tíos, tías, primos o primas, siempre ha sido de hermanos.

**"A ANGELES"**

Quiero que sepas que de todas las cosas tan hermosas que me ha regalado la Universidad, la amistad que me permitió construir contigo es algo que con nada podré pagarle. Ha sido muy grato compartir todos esos momentos que sólo la escuela te da con una persona como tú. Siempre estaremos juntas sin importar la distancia.

Era obligado cerrar con broche de oro nuestro paso por la Facultad y, que mejor, que trabajando unidas en este proyecto ¿no lo crees?.

"A OCTAVIO, KARLA, MELISSA, DIANA, ELI Y ROSALBA"

Me siento muy afortunada de conocer personas tan valiosas como Uds., siempre llevo presentes todos los momentos tan especiales y únicos (sobre todo las risas) que hemos compartido y que, seguramente, todos extrañamos.

# ÍNDICE

## Capítulo 1. Introducción

<b>1.1 GENERALES</b>	<b>1</b>
<b>1.2 DEFINICIÓN DEL PROBLEMA</b>	<b>1</b>
<b>1.3 OBJETIVOS</b>	<b>2</b>
<b>1.3.1 OBJETIVO GENERAL</b>	<b>2</b>
<b>1.3.2 OBJETIVOS ESPECÍFICOS</b>	<b>2</b>

## Capítulo 2. Internet

<b>2.1 ORIGEN</b>	<b>3</b>
<b>2.2 CÓMO FUNCIONA INTERNET</b>	<b>13</b>
<b>2.2.1 TCP/IP</b>	
<b>(TRANSPORT CONTROL PROTOCOL/INTERNET PROTOCOL)</b>	<b>13</b>
<b>2.2.1.1 HISTORIA</b>	<b>13</b>
<b>2.2.1.2 CONCEPTO Y ARQUITECTURA</b>	<b>16</b>
<b>2.2.1.3 CARACTERÍSTICAS</b>	<b>17</b>
<b>2.2.1.4 PROTOCOLOS</b>	<b>18</b>
<b>2.2.1.5 CÓMO FUNCIONA TCP/IP</b>	<b>19</b>
<b>2.2.1.6 IP</b>	<b>19</b>
<b>2.2.1.7 TCP</b>	<b>24</b>
<b>2.2.1.8 CAMBIOS EN TCP/IP</b>	<b>27</b>
<b>2.2.2 IP VERSIÓN 4</b>	<b>28</b>
<b>2.2.3 LA NUEVA VERSIÓN IP (IP VERSIÓN 6, IPV6, IPNG)</b>	<b>30</b>
<b>2.3 SERVICIOS DE INTERNET</b>	<b>33</b>
<b>2.3.1 SERVICIOS DE RED</b>	<b>33</b>
<b>2.3.1.1 SERVICIOS DE ENTREGA DE PAQUETES SIN CONEXIÓN</b>	<b>33</b>
<b>2.3.1.2 SERVICIOS DE TRANSPORTE DE FLUJO CONFIABLE</b>	<b>33</b>
<b>2.3.2 SERVICIOS DE APLICACIÓN</b>	<b>34</b>
<b>2.3.2.1 SESIÓN REMOTA</b>	<b>34</b>
<b>2.3.2.2 TRANSFERENCIA DE ARCHIVOS</b>	
<b>(FILE TRANSFER PROTOCOL, FTP)</b>	<b>35</b>
<b>2.3.2.3 SECURE SHELL (SSH)</b>	<b>37</b>
<b>2.3.2.4 GRUPOS DE INTERÉS (NEWS)</b>	<b>39</b>
<b>2.3.2.5 LISTAS DE DISTRIBUCIÓN</b>	<b>39</b>
<b>2.3.2.6 TRANSMISIÓN DE CHARLAS EN INTERNET</b>	
<b>(INTERNET RELAY CHAT, IRC)</b>	<b>40</b>
<b>2.3.2.7 CORREO ELECTRÓNICO (E-MAIL)</b>	<b>40</b>
<b>2.3.2.8 SERVICIOS DE BÚSQUEDA DE INFORMACIÓN</b>	<b>42</b>
- ARCHIE	<b>42</b>
- WAIS	<b>43</b>

2.3.2.9	SERVICIO INTEGRAL DE INFORMACIÓN WORLD WIDE WEB	43
2.4	INTERNET2 (I2)	44

### Capitulo 3. WORLD WIDE WEB

3.1	HISTORIA	51
3.2	MODELOS DE DESARROLLO WEB	54
3.2.1	ARQUITECTURA CLIENTE/SERVIDOR	54
3.2.1.1	CARACTERÍSTICAS	54
3.2.1.2	COMPONENTES	55
3.2.1.3	LA ESTRUCTURA DE UNA APLICACIÓN Y SU RELACIÓN CON LA ARQUITECTURA CLIENTE/SERVIDOR	58
3.2.1.4	MODELOS DE LA ARQUITECTURA CLIENTE/SERVIDOR	61
	- SISTEMA CLIENTE/SERVIDOR DE DOS NIVELES O CAPAS (TWO THIERED)	61
	- SISTEMA CLIENTE/SERVIDOR DE TRES NIVELES O CAPAS (THREE THIERED)	61
3.2.1.5	TECNOLOGÍAS DE DESARROLLO	61

### Capitulo 4. Sistemas de Información Administrativos

4.1	CONCEPTOS BÁSICOS	68
4.1.1	SISTEMAS NATURALES	68
4.1.1.1	SISTEMAS FÍSICOS	68
4.1.1.2	SISTEMAS VIVIENTES	69
4.1.2	SISTEMAS HECHOS POR EL HOMBRE	69
4.2	SISTEMAS AUTOMATIZADOS O SISTEMAS DE INFORMACIÓN	69
4.2.1	ELEMENTOS DE UN SISTEMA DE INFORMACIÓN	69
4.2.2	ACTIVIDADES BÁSICAS DE UN SISTEMA DE INFORMACIÓN	70
4.2.3	SISTEMAS DE INFORMACIÓN ADMINISTRATIVOS	70
4.2.3.1	OBJETIVOS	70
4.2.3.2	CLASIFICACIÓN	71
4.2.4	LOS SISTEMAS DE INFORMACIÓN EN LA ESTRATEGIA DE LA EMPRESA	73
4.2.5	SISTEMAS DE INFORMACIÓN ORIENTADOS A LA WEB	74
4.2.5.1	CONCEPTO	74
4.2.5.2	REQUERIMIENTOS	74
4.2.5.3	ROL DE USUARIO	75
4.2.5.4	CARACTERÍSTICAS	75
4.2.5.5	EVOLUCIÓN	75
4.2.5.6	TECNOLOGÍAS WEB	77
4.2.5.7	CLASIFICACIÓN	80

	- <b>BUSCADORES</b>	<b>80</b>
	- <b>COMERCIO ELECTRÓNICO</b>	<b>83</b>
	- <b>PORTALES</b>	<b>90</b>
<b>4.2.5.8</b>	<b>FUTURO DE LOS SISTEMAS WEB</b>	<b>94</b>
	- <b>DESARROLLO DE APLICACIONES EN INTERNET2</b>	<b>94</b>
<b>4.2.5.9</b>	<b>POR QUÉ TODO TIENE A INTERNET</b>	<b>95</b>
<b>4.2.6</b>	<b>METODOLOGÍAS DE DESARROLLO DE SISTEMAS</b>	<b>96</b>
<b>4.2.6.1</b>	<b>ANÁLISIS MODERNO ESTRUCTURADO DE EDWARD YOURDON</b>	<b>97</b>
<b>4.2.6.2</b>	<b>METODOLOGÍA DE DESARROLLO ORIENTADO A OBJETOS</b>	<b>97</b>

## Capítulo 5. Caso Específico: "Aviso de Ocasión del Portal UNAM"

<b>5.1</b>	<b>INTRODUCCIÓN</b>	<b>99</b>
<b>5.2</b>	<b>ANÁLISIS</b>	<b>102</b>
<b>5.3</b>	<b>DISEÑO</b>	<b>119</b>
<b>5.4</b>	<b>DESARROLLO</b>	<b>126</b>

<b>Conclusiones</b>	<b>127</b>
---------------------	------------

## Apéndices

- A. PROGRAMAS FUENTE**
- B. PANTALLAS DEL SISTEMA**
- C. GLOSARIO**
- D. REFERENCIAS**



# 1. INTRODUCCIÓN

## 1.1 Generales

Hoy en día, vivimos en un mundo sumergido en un constante y acelerado devenir, lo que nos hace testigos de los avances que las tecnologías de comunicación y computación han tenido. Esto lo podemos constatar en la gran cantidad de gente que hace uso de una de sus aplicaciones llamada INTERNET.

Internet ha venido a revolucionar la vida del hombre, lo que le ha permitido hacer innovaciones en la manera de llevar a cabo ciertas actividades, tanto individuales, como aquellas que efectúa siendo parte de un todo: LA ORGANIZACIÓN.

Dentro de la organización Internet se ha desarrollado en varias fases. En sus inicios, se fue haciendo importante para ellas el disponer de algún tipo de presencia en este medio, con lo que aparecieron una multitud de pequeñas páginas corporativas en las que se mostraba algún tipo de información sobre la organización.

Posteriormente, la sola presencia en Internet dejó de ser suficiente y se fue haciendo necesario introducir nuevos elementos y ofrecer algún tipo de servicio, así como, una mejora en los contenidos. Actualmente, algunas organizaciones prácticamente permanecen como en esos primeros momentos y otras, empiezan a ver clara la necesidad de ser más ambiciosos respecto a las posibilidades de Internet.

Para las organizaciones que cuentan con este medio, ya no basta con "estar" sino que es preciso "hacer". Ya no sirve esperar que vengan los clientes, sino que se hace necesario ir a buscarlos y conseguir mantenerlos fieles a los espacios en Internet y a las organizaciones.

## 1.2 Definición del Problema

Evidentemente Internet evoluciona de forma sin precedentes, sin embargo, es importante señalar que su entusiasta adopción por parte de algunas organizaciones, obedece más que nada, a considerarlo simplemente parte de una moda. Por lo anterior, se ha podido observar que existe una utilización reducida o inadecuada de este medio.

En resumen, el saber que Internet ofrece un gran potencial no significa que se aproveche o se tenga idea de cómo orientar el mismo, de manera que ayude a cumplir con los objetivos propios de la organización que ha optado por Internet.

## **1.3 Objetivos**

### **1.3.1 OBJETIVO GENERAL**

Podemos decir que es necesario que la Organización analice a fondo las razones que justifiquen la adopción de Internet, puesto que ello contribuirá a que se explote correctamente esta herramienta y, lo más importante, es que coadyuvará de sobremanera a la evolución de la propia entidad.

Concretamente, para una organización como lo es la Universidad Nacional Autónoma de México, una gran parte del desarrollo de Internet se orienta a ofrecer a la comunidad universitaria servicios.

Por lo anterior, el presente proyecto tiene como finalidad mostrar un ejemplo con el "Aviso de Ocasión", de lo que actualmente se realiza en la UNAM como parte de su incursión a este nuevo mundo: Internet y el aprovechamiento de sus servicios.

### **1.3.2 OBJETIVOS ESPECÍFICOS**

- ✓ Realizar un estudio sobre el surgimiento de Internet, su funcionamiento, servicios, evolución y tendencias.
- ✓ Mostrar, de manera específica, cuál es el más importante de los servicios que brinda Internet y, en qué radica dicha importancia.
- ✓ Presentar las características de los sistemas, enfocándonos a los sistemas de información administrativos.
- ✓ Señalar que son los sistemas de información Web, ventajas que brindan a la empresa y perspectivas.
- ✓ Determinar el papel que juega actualmente Internet en el contexto organizacional.
- ✓ Enfocarnos sobre un servicio específico que ofrece el Portal UNAM: "Aviso de Ocasión".

## 2. INTERNET

Debido a su magnitud e importancia la historia de Internet es algo compleja y extensa, con diversas personas y hechos involucrados durante varios años. Esto se debe, principalmente, a su acelerado y continuo desarrollo y a que Internet no es una tecnología única - como en el caso de la imprenta o la televisión donde se puede especificar exactamente su inventor - sino que, se basa en múltiples tecnologías enlazadas para conformar lo que actualmente conocemos como tal.

### 2.1 Origen

Ante el lanzamiento del *Sputnik 1*, primer satélite artificial, el 4 de octubre de 1957 y del *Sputnik II* el 3 de noviembre del mismo año por parte de la Unión Soviética, una de las reacciones inmediatas del gobierno de los Estados Unidos fue la creación de organismos que intentasen superar la ventaja tecnológica adquirida por los soviéticos pero, ante todo, recuperar la hegemonía mundial.

Así, en 1958, se crea la Agencia para Proyectos de Investigación Avanzada (*Advanced Research Projects Agency, ARPA*)<sup>1</sup> como una agencia militar dependiente del Departamento de Defensa (*Department of Defense, DoD*) y, cuyo objetivo, era ayudar a mantener la superioridad tecnológica de los E.U. y estar prevenida frente a nuevos avances tecnológicos de enemigos potenciales, desarrollando para ello, creativas ideas.

La organización interna del ARPA estaba dividida en departamentos que trabajaban en diferentes campos pero que no disponían de medios propios de investigación, siendo su actividad principal el financiamiento de cualquier iniciativa, tanto pública como privada, considerada útil para futuras aplicaciones en materia de defensa. Los proyectos que financiaba no tenían un resultado militar inmediato, incluso algunos nunca llegaron a tener una utilidad bélica; sin embargo, los fondos que aportaba el ARPA, dotaban de una extraordinaria infraestructura a las organizaciones beneficiadas (normalmente universidades o centros estatales de investigación) permitiéndoles con ello, realizar grandes proyectos que, de otra manera, no hubiesen sido llevados a cabo<sup>2</sup>.

A principio de los años 60, los responsables del ARPA, se dieron cuenta de la utilidad que las computadoras y las redes de comunicaciones iban a tener para la defensa nacional en un futuro inmediato. Así, el DoD de manera paralela, encargó a la *RAND Corporation* (una de las primeras organizaciones dedicadas a realizar estudios en materia de defensa), el análisis sobre el reciente

---

<sup>1</sup> Hoy en día, el ARPA se ha desprendido de la parte no militar y ha cambiado su denominación a Agencia de la Defensa para Proyectos de Investigación Avanzada (*Defense Advanced Research Projects Agency, DARPA*), funcionando con los mismos objetivos que el día de su fundación.

<sup>2</sup> Esta es una de las razones por las que, la creación de Internet, ha estado estrechamente vinculada con los entornos militares norteamericanos, ya que, si bien es cierto que los fondos eran de origen militar, las iniciativas y los creadores materiales provenían de los entornos académicos del país.

atentado sufrido a las comunicaciones (infraestructura vital para el ejército en un conflicto bélico) y las posibles soluciones al mismo. En dicho atentado, se hicieron estallar tres torres de comunicaciones en el estado de Utah, presuntamente por miembros de un grupo denominado Ejército Republicano Americano; tras este ataque, parte del sistema de comunicaciones norteamericano quedó interrumpido por un tiempo indeterminado por las autoridades y puso en evidencia la vulnerabilidad del mismo.

La otra acción que efectuó el DoD, fue abrir una línea de Investigación sobre Sistemas de Mando y Control (*Command and Control Research, CCR*) que se desprendió del ARPA, y que fue encabezada por Joseph Carl Robnett Licklider quien era un psicólogo formado en el Instituto Tecnológico de Massachusetts (*Massachusetts Institute of Technology, MIT*), con amplia formación en materias de ingeniería y que se encontraba realizando trabajos sobre el tema en una compañía denominada *Bolt Beranek and Newman Ltd. (BBN)*.

Licklider dedicó dos años a este propósito y convirtió el CCR en la Oficina para las Tecnologías de Procesado de la Información (*Information Processing Techniques Office, IPTO*) dentro del ARPA y, cuyo objetivo era, investigar técnicas para procesar la información y buscar mejores maneras de usar las computadoras, es decir, visualizar más allá de su uso inicial que era simplemente el de grandes máquinas calculadoras.

Licklider fue el primer director del IPTO en el periodo de 1962 a 1964, entre otras cosas, aportó dos ideas fundamentales para el nacimiento de la red:

- ✓ Las computadoras iban a ser, por encima de otras consideraciones, equipos de comunicaciones.
- ✓ Predijo la necesidad de las redes de computadoras como medio para compartir recursos entre grupos de trabajo.

En 1966, tras la salida de Licklider del IPTO, este organismo apoyaba a varios centros de investigación, muchos de los cuales, estaban dotados de costosas instalaciones, normalmente grandes computadoras que iban creciendo de forma alarmante desde el punto de vista económico.

Independientemente del número de computadoras que se instalasen, siempre había científicos que reclamaban servicios que no prestaban las suyas propias, pero que si lo hacían algunas de otros centros, además, se llegaba a presentar duplicación de esfuerzos dentro de la comunidad de investigadores. Esta situación estaba empezando a adquirir dimensiones de problema, por lo que, el entonces director del IPTO Robert Taylor, decidió retomar las ideas de Licklider y empezar a plantearse seriamente la posibilidad de crear algún sistema que permitiese compartir todos los recursos del IPTO, entre las personas que participaban en los diferentes programas del mismo, independientemente del lugar donde se encontrase su centro de trabajo.

La idea de compartir recursos estaba planteada. Desde ese momento se empezó a construir el término *networking* que intentaba describir un grupo de computadoras intercomunicadas, formando

una red de tal manera que pudiesen compartir recursos. Inicialmente, estos recursos eran básicamente *software* y aún estaba lejos la idea de que se incluyera el *hardware*.

La primera necesidad de Robert Taylor en ese momento era pensar en quién podría encargarse en desarrollar el proyecto, ya que, era una idea totalmente innovadora, prácticamente, en materia de comunicaciones entre computadoras, no había nada hecho, solamente se habían desarrollado pocos experimentos de conexión de equipos entre sí.

Una de las pocas personas que había trabajado en este tema era Larry Roberts que, en aquel momento, trabajaba en los laboratorios Lincoln (*Lincoln Lab*). Dada la relación que existía entre el ARPA y los Laboratorios Lincoln, Roberts fue enviado por sus jefes a trabajar con Taylor y se convirtió en la primera persona en la historia de Internet contratada específicamente para trabajar en la red.

En abril de 1967, Taylor y Roberts se presentan en el Congreso Anual de Responsables de Investigación que ARPA organizaba en la Universidad de Michigan, con el objeto de presentar la idea y proponer una serie de medidas encaminadas a crear una coyuntura para empezar a trabajar.

La propuesta de Roberts y Taylor era la creación de un sistema de comunicaciones que interconectara las distintas supercomputadoras financiadas por ARPA, a fin de poder extender su uso y así poder rentabilizar al máximo la inversión realizada. El cómo y el cuándo hacer esto, era aún una incógnita pero se empezó a hablar del tema en forma generalizada y, como consecuencia, aparecieron los primeros problemas de carácter técnico como el siguiente: Dado que existían muy pocas posibilidades, principalmente por las necesidades técnicas a la hora de hacer la adquisición, de que todas las computadoras se comprarán al mismo fabricante, y ante el hecho de que cada fabricante usaba su propio conjunto de normas y se trabajaba distinto con las computadoras de cada uno, se veía en la necesidad de encontrar una manera de estandarizar la forma de trabajar con todas estas computadoras.

Por lo anterior, se creó un grupo de trabajo denominado Grupo de Comunicaciones (*Communication Group*), encargado de buscar alguna salida al problema, planteándolo como sigue:

1. Construir una red de datos físicamente, es decir, definir los canales de comunicaciones que se iban a utilizar para intercomunicar las diferentes computadoras, qué velocidades de transmisión usarían y su costo.
2. Conectada cada computadora con la red descrita anteriormente, ¿qué sistema operativo<sup>3</sup> usaría y qué protocolo<sup>4</sup> sería el elegido para su intercomunicación?

---

<sup>3</sup> Conjunto de programas que administran los recursos del sistema para su mejor aprovechamiento. Hay un administrador para cada uno de los recursos (Administrador de Memoria, Administrador de Dispositivos de Entrada/Salida, Administrador de Almacenamiento, Administrador de Información, Administrador del Procesador y, Administrador de Protección y Tratamiento de Errores) y, cada administrador es un programa. Por lo tanto, el sistema operativo es un conjunto de programas.

La técnica de solución consistía en nombrar un coordinador del grupo, en el caso anterior el Grupo de Comunicaciones, responsable de recopilar y coordinar los trabajos aportados por todos sus miembros y generar un documento denominado *draft*<sup>4</sup> (borrador) que era distribuido a todos los miembros para ser estudiado y para que cada uno aportara y corrigiera lo necesario. Tres meses después, se organizó una nueva reunión en ARPA con el fin de exponer la respuesta a la problemática anteriormente expuesta.

El primer punto se solucionó adoptando como soporte de comunicación líneas telefónicas que estarían permanentemente conectadas a las computadoras; este hecho, alejaba al sistema a construir, de una utilización al más puro estilo militar y lo acercaba a los modelos informáticos utilizados en los entornos académicos norteamericanos. Para el segundo punto, se ideó la construcción de un dispositivo que tuviera dos partes diferentes:

- ✓ La primera parte sería conectada a las líneas de comunicaciones y, por lo tanto, al resto de la red estaría compuesta por equipos electrónicos y un protocolo de comunicaciones idéntico en todos los sitios, hacer fluir por él la información y, posteriormente, cerrarlo una vez finalizada la transmisión.
- ✓ La segunda parte, sería específica para cada uno de los sitios y se construiría en función de las computadoras allí instaladas que, desde entonces, comenzaron a ser conocidas como *hosts*<sup>5</sup> (anfitrión).

Con lo anterior, se tenía una visión más clara: ARPA debía proporcionar a la futura red una infraestructura mínima compuesta por líneas de comunicación, y por la parte común de los dispositivos descritos anteriormente que comenzaron a ser llamados IMP<sup>7</sup> (*Procesador de Interfaz de Mensaje/Interface Message Processor*).

A finales de 1967, ARPA asigna al Instituto de Investigaciones de Stanford (*Stanford Research Institute, SRI*) un contrato para poner en orden todas las ideas manejadas hasta el momento al aplicar la técnica *draft* y, escribir unas especificaciones que sirvieran como base al desarrollo de la futura red (incluyendo *software* y *hardware*).

---

<sup>4</sup> Conjunto de normas de diseño electrónico y de programación que permiten abrir un canal de comunicación entre dos computadoras

<sup>5</sup> Esta técnica ha sido ampliamente utilizada en el desarrollo de Internet y es la que ha permitido hablar del concepto de "comunidad" en el desarrollo de muchos de los puntos de la misma

<sup>6</sup> Es una computadora que, mediante la utilización de los protocolos TCP/IP, permite a los usuarios comunicarse con otros sistemas anfitriones de una red. Los usuarios se comunican utilizando programas de aplicación.

<sup>7</sup> Es el antecedente de lo que hoy denominamos ruteador. En diciembre de 1968 se designó a la compañía *Bolt Beranek and Newman Ltd (BBN)*, establecida en Cambridge (Massachusetts), como contratista para la construcción e integración de los primeros IMP. BBN ha representado una inagotable fuente de recursos humanos e ideas en el desarrollo de Internet tal y como la conocemos en la actualidad

Roberts introdujo algunas modificaciones y lo dejó listo para ser presentado y discutido en el Congreso Anual de Responsables de Investigación del ARPA, en junio de 1968.

El 3 de junio de dicho año, Robert Taylor, basándose en la propuesta de Larry Roberts, presentó al director del ARPA un plan denominado *Resource Sharing Computer Network* (Red de Computadoras para Compartir Recursos), que contenía el resultado de los trabajos realizados en el último año y que constituía la primera propuesta para construir la red.

El día 21 del mismo mes y año, ARPA da su visto bueno y asigna 500 000 dólares a la construcción de su red de datos denominada ARPANETwork, abreviadamente ARPANET, tal como ha sido conocida desde entonces. Teniendo una idea medianamente clara de qué había que hacerse y lo que es más importante, con una provisión de fondos económicos para poder empezar, a finales de julio ARPA distribuye un pliego de condiciones a 140 contratistas especificando todos los detalles técnicos para la interconexión de 4 centros de cómputo que deberían quedar plenamente en operación, abriendo la posibilidad de extender la red hasta un total de 17 centros.

Dichos centros (v figura 21) fueron elegidos con gran cuidado teniendo en cuenta ciertas consideraciones; debían ser centros con programas en curso patrocinados por ARPA y directamente relacionados con el mundo académico e investigación.

Paralelamente a la selección de estos centros, comenzó otro trabajo de vital importancia: Necesitaban dar algún contenido a la infraestructura que iban a construir, ya que, una red no sólo era un conjunto de cables que conectan computadoras sino que debía disponer de herramientas que dieran alguna utilidad a estos equipos. Por esa razón, ARPA decidió encomendar esta responsabilidad a los cuatro centros mencionados y, de esta manera, involucrar directamente a la comunidad científica.

NODO	CENTRO	SIGLAS	FECHA
1	Universidad de California en Los Angeles	UCLA	30 de Agosto
2	Instituto de Investigaciones de Stanford	SRI	1 de Octubre
3	Universidad de California en Santa Bárbara	UCSB	1 de Noviembre
4	Universidad de UTAH	UTAH	Diciembre

Figura 2.1 Primeros Centros en ARPANET

Roberts decidió dar un giro al Grupo de Comunicaciones (formado hasta aquel entonces y desde el congreso de abril de 1967 por representantes de muy diversos centros) nombrándolo *Network Working Group* (NWG) y se conformó por representantes de los centros elegidos inicialmente. Para coordinar el grupo, nombró al autor del informe entregado meses atrás por el SRI, Elmer Shapiro.

Dentro de las primeras reuniones que efectuó el *Network Working Group*, no se tenía una idea clara de lo que se estaba haciendo, sin embargo, de ahí salieron ideas trascendentales en el posterior

desarrollo de la red, entre ellas, el concepto de sesión remota (v. capítulo 2.3.2.1), la transferencia de archivos (v. capítulo 2.3.2.2) y el correo electrónico (v. capítulo 2.3.2.7); conceptos habituales en nuestros días pero, en aquella época, no existía el mínimo intercambio de información entre computadoras.

En los primeros días de 1969 Frank Heart, jefe del grupo de desarrollo del BBN, presentó una primer propuesta de lo que debería ser el IMP. Dado el tamaño de las computadoras en ese momento, decidieron buscar una lo más reducida posible; se eligió la DDP-516 de la compañía *Honeywell*, la minicomputadora<sup>8</sup> más potente del mercado en esa época, pesaba 400 Kg (de tamaño similar al de una máquina actual de refrescos), con 24Kb de memoria, 16 puertos de entrada/salida. Junto con la DDP-516 se proyectó un módem<sup>9</sup> con un tamaño similar a la minicomputadora. El proyecto proveía distribuir los equipos junto con una terminal teletipo<sup>10</sup> y un lector de tarjetas perforadas.

En febrero de 1969, se organizó la primera reunión entre los fabricantes del IMP (BBN) y el NWG (proveedor de ideas para el uso de los IMP); ahí se pudo apreciar que la primera de las partes sólo se había preocupado en cómo hacer circular los *bits* a la mayor velocidad posible y de la forma más segura mientras que los segundos, sólo habían pensado en las aplicaciones finales; es decir, se dejó a un lado los protocolos y las normas a seguir para intercambiar la información.

Con este panorama, la gente del BBN y del NWG decidió reunir notas y apuntes, producto de las reuniones para crear documentos de trabajo que sirvieran como base de discusión que seguirían sólo dos reglas básicas: Todo el mundo podía decir todo cuanto quisiera, situándose todos a un mismo nivel jerárquico y, Todo cuanto se escribiera fuera discutible y modificable, por lo tanto, nada sería oficial.

Uno de los estudiantes del grupo, Steve Crocker, redactó un primer documento al que llamó *Host Software* y, para dar mayor peso a la idea de "no oficial", decidió llamar a este tipo de documentos *Peticion para Comentarios (Request For Comments, RFC<sup>11</sup>)*. Esta técnica consistía en

---

<sup>8</sup> *Computadora multiusuario concebida para cubrir necesidades de una empresa o de un departamento de la misma. Una minicomputadora es más poderosa que una computadora personal pero menos que un mainframe.*

<sup>9</sup> *Módem: Palabra inglesa resultante de unir los términos MODulador-DEModulador. Es un dispositivo que permite conectar la computadora a una red informática por medio de la red telefónica. En el proceso de emisión, el dispositivo modula la información y en el proceso de recepción demodula la señal recibida y reconstruye la original.*

<sup>10</sup> *Método de representación de caracteres en un monitor en el cual, los caracteres se generan y envían uno a uno a la pantalla de un monitor; luego, a medida que se reciben, la pantalla se va llenando línea a línea. Una vez llena, la pantalla se recorre hacia arriba para acomodar las nuevas líneas de caracteres que aparecen en la parte inferior.*

<sup>11</sup> *Esta abreviatura que se ha convertido en una leyenda dentro de Internet. El primer RFC (RFC-1) tiene fecha del 7 de abril de 1969. El modelo actual de los RFC, constituye el estándar de muchos de los protocolos usados comúnmente y está basado en la publicación de documentos definitivos, manteniéndose los debates paralelamente en foros.*



proponer un tema a debate y, a través de los RFC, los integrantes del BBN y del NWG presentaban sus ideas particulares en forma de comentarios, críticas o elogios; a su vez, estas ideas eran respondidas a través de otros RFC, es decir, la apertura de un tema generaba una serie de documentos numerados. Esta práctica se abandonó debido al volumen de los debates abiertos que hacía casi imposible el manejo de tal cantidad de información.

Finalmente, en 1969, se inició la entrega e instalación de los IMP a cada uno de los centros elegidos. Así, el 1º de septiembre, BBN entregó oficialmente el primer IMP a la UCLA. En 72 horas el IMP estaba conectado a la supercomputadora<sup>12</sup> Sigma-7 corriendo un sistema operativo llamado *SEX*, la electrónica de interconexión la desarrolló Mike Wingfield, un estudiante de posgrado y, el *software*, Crocker en colaboración con Vinton Cerf y Jonathan B. Postel. La primera conexión entre la computadora central y el IMP se probó con éxito; para poder hablar de una red sólo faltaba instalar otro IMP e intentar la intercomunicación.

En el mes de octubre del mismo año, BBN realizó la segunda entrega de un IMP a Shapiro y sus colaboradores del SRI en Stanford, conectándolo a una supercomputadora SDS940 casi de manera inmediata. El 21 de noviembre de 1969, se realizó la primer conexión entre las computadoras situadas en Los Angeles y Stanford efectuándose la primer sesión remota de la historia. Durante el mismo mes se entregó el tercer IMP a UCSB, donde fue conectado a una computadora IBM 360/75 y al mes siguiente se hizo lo mismo en la Universidad de Utah conectándose a una computadora modelo PDP-10.

Los cuatro primeros puntos de la red sirvieron, entre otras cosas, para proporcionar una plataforma de pruebas donde perfeccionar los equipos existentes e inmediatamente se les unieron los IMP número 5, 6 y 7 instalados en las oficinas de BBN, en el MIT y en *RAND Corporation* respectivamente. Estos tres nuevos nodos estuvieron disponibles durante los meses de marzo y abril de 1970.

Con este hecho, no mejoraron los diseños anteriores, sin embargo, los nodos<sup>13</sup> del MIT y de BBN estaban en la Costa Este de los Estados Unidos, es decir, por primera vez un nodo estaba realmente alejado (a casi 5 000 Km de distancia). La tecnología utilizada era básicamente la misma que en los casos anteriores, una línea telefónica con capacidad para transmitir 56 Kbps (Kbits por segundo) proporcionada por AT&T. Lo sorprendente era que, a pesar de su longitud, el rendimiento de esta línea no era funcionalmente menor que el de las anteriores.

A mediados de año, se instalaron dos nodos en SDC y en la Universidad de Harvard (los números 8 y 9 respectivamente). Durante el tercer cuatrimestre se realizaron las instalaciones de los Laboratorios Lincoln y de la Universidad de Stanford. A finales de año, se terminaría con la instalación

---

<sup>12</sup> Refinada y costosa computadora diseñada para ejecutar cálculos complejos a la máxima velocidad permitida por los avances tecnológicos actuales. Las supercomputadoras se usan en investigación científica.

<sup>13</sup> Nombre genérico que se le da a cualquier dispositivo hardware con su correspondiente software que se encuentra conectado a una red y que es accesible desde la misma.

de los números 12, 13, 14 y 15 en la Universidad de Urbana en Illinois, en la Universidad de Case Western, en la Universidad de Carnegie-Mellon y en la primera compañía privada en conectarse, Burroughs.

Aunque la red empezaba a extenderse a lo largo y ancho del país, seguía quedando un asunto pendiente: la creación de líneas redundantes<sup>14</sup> de comunicación (primera motivación para la construcción de la red) que permitieran sobrevivir en caso de producirse algún tipo de desastre en la infraestructura de comunicación existente.

Por esta razón, 1971 se dedicó a la instalación de un mayor número de líneas de comunicaciones en sitios donde ya existían. Durante esta etapa y, desde el punto de vista del *hardware*, se consiguieron progresos substanciales, uno de ellos fue la mejora de los IMP encaminada a poder trabajar con dispositivos multipuerto, es decir, con capacidad de ser conectados a más de una computadora. Pese a todo esto, para conectarse a la red se necesitaba disponer de una gran computadora con un costo muy elevado al igual que el mantenimiento.

Ante esto, se planteó la necesidad de desarrollar algún equipo que, conectándolo a simples terminales<sup>15</sup>, permitiera acceder a la red sin la necesidad de disponer de un gran equipo de cómputo. La solución al problema se dió con la creación de los denominados TIMP (Terminal IMP), ideados para ser conectados por un lado a la red y, por el otro, a un gran número de terminales en lugar de a una computadora, permitiendo el acceso múltiple de usuarios. Actualmente a los descendientes de estos dispositivos se les denomina Terminales de Acceso Remoto y siguen utilizándose ampliamente. Esto representó otro avance en la extensión del uso de la red, al permitir conectarse y, por lo tanto, compartir sus recursos, con un elevado número de usuarios a un costo reducido.

Pese a todo lo anterior, se estaba perdiendo gran parte del interés inicial en la red; ARPANET se empezó a ver como algo muy caro y condicionado a que sólo un número reducido de personas, tuviera acceso real. Por lo que Larry Roberts contrata a Alex McKenzie y lo coloca al frente del recién creado Centro de Control de la Red (*Network Control Center, NCC*) y, por otro, hace del conocimiento público las bondades de ARPANET, las innegables utilidades y aplicaciones que proporcionaría a la comunidad científica americana.

De esta manera, se lleva a cabo la Conferencia Internacional sobre Comunicaciones por Computadora (*International Conference on Computer Communications, ICC*) en el hotel *Hilton* de Washington D.C. con la asistencia de organizaciones extranjeras, resultando todo un éxito. Una de las

---

<sup>14</sup> El concepto de línea redundante consiste en unir dos puntos no sólo por una única línea, sino por varias que realicen la conexión por caminos diferentes para, así de esta manera, aumentar las posibilidades de supervivencia en caso de fallo.

<sup>15</sup> Terminal: Pequeño equipo informático formado normalmente por un monitor y un teclado, que se sitúa bastante lejos de la computadora central (*host*) al que está conectado y desde donde se pueden pedir servicios a la computadora central.

aportaciones de dicha conferencia fue, la creación del Grupo Internacional de Trabajo de Redes (*International Network Working Group*, INWG) ante la necesidad de avances en las tecnologías de redes y, en 1973 se lograría el primer enlace intercontinental con la *University College of London* (Londres) y el *Royal Radar Establishment* (Noruega).

Por otra parte, el tipo de líneas de comunicación (aquellas instaladas entre los sitios donde se montaron los primeros IMP que proporcionó la compañía AT&T) que se utilizaron durante los primeros años de la implantación de la red, originó una maraña de cables y, ante ello, la necesidad de crear algún tipo de infraestructura que permitiera "limpiar" la red y, por otra parte, proporcionar enlaces de muy alta capacidad que sustituyeran a las pequeñas líneas existentes.

Con ese fin, la Fundación Nacional de Ciencias de los E.U. (*National Science Foundation*, NSF) creó la Oficina de Cómputo Avanzado (*Office of Advanced Scientific Computing*, OASC) que comenzó a trabajar en dos proyectos simultáneamente:

- ✓ La creación de un grupo de centros de supercómputo.
- ✓ La creación de una infraestructura encargada de comunicarlos entre sí, para proporcionar acceso a terceros usuarios, dejando la posibilidad de comunicar el conjunto de pequeñas redes académicas dispersas por todo el país.

En 1984 el NSF fundó los centros de supercómputo de John Von Neumann en Princeton, el Centro de Supercómputo de San Diego (*San Diego Supercomputing Center*) en la Universidad de California en San Diego (UCSD), el Centro Nacional para Aplicaciones de Supercómputo (*National Center for Supercomputing Applications*) en la Universidad de Illinois, el Centro de Teoría (*Theory Center*) en Cornell y el Centro de Supercómputo (*Supercomputing Center*) de Pittsburgh. La NSF instaló líneas de comunicación entre estos equipos, con el fin de conseguir un óptimo uso de los recursos a imagen y semejanza de lo realizado por ARPA a finales de los años 70.

En la filosofía del proyecto, figuraba el ofrecer a toda la comunidad americana estos centros que, al contrario de los integrados en ARPANET, estaban totalmente dedicados a la investigación de cualquier universidad del país y no sólo para aquellas que realizaran proyectos útiles a la defensa. Con este fin, se ofreció una conexión a las numerosas redes regionales que, con carácter académico intercomunicaban los diferentes centros de los E.U. El resultado de esta mezcla fue la creación de la red académica americana, denominada NSFNET. Los primeros centros instalados fueron los de Illinois y Cornell.

Después de los primeros años de la implantación de esta infraestructura, se produjo un aumento en la utilización de las líneas tal que dejaron la red prácticamente colapsada. Para solucionar este problema en noviembre de 1987, la NSF asignó un contrato a la compañía *Merit Network, Inc.* (compañía proveedora de servicios de Internet en Michigan E.U.) para que, junto con IBM y MCI (importante compañía americana de telecomunicaciones) rediseñaran completamente NSFNET y la

dotara de *backbones*<sup>16</sup>. Al proyecto, se incorporó el *National Center for Atmospheric Research* (NCAR) de Colorado.

Ocho meses después de la asignación del contrato, las líneas estaban instaladas e interconectaban un total de 170 redes de menor tamaño, administradas desde un centro de control instalado en el Centro de Operaciones de la Red (*Network Operation Center, NOC*) proporcionado por Merit. El gran número de operaciones diarias de esta red pusieron de manifiesto que los catorce trabajadores de la plantilla del NSF dedicados a la red, eran insuficientes para mantener en explotación una infraestructura que crecía sin parar.

Esta situación no pasó desapercibida para IBM que, desde años atrás, colaboraba con Merit en la infraestructura del *backbone* NSFNET. Como resultado de esta situación, a finales de 1989 e inicios de 1990, IBM decide realizar una propuesta a Merit y a MCI a fin de crear una corporación sin fines de lucro con el objetivo de proponer a la NSF la ampliación de las líneas existentes.

El nombre propuesto para dicha corporación fue el de *NewNet* y pretendían crear un *backbone* de propiedad y gestión privada y colocar como primer cliente al propio NSFNET, utilizando sus fondos para su construcción. A finales de junio, Merit realiza una propuesta formal al NSF donde se les invita a aceptar a la nueva compañía como explotadora de la infraestructura existente en sustitución de Merit.

Bajo la aparente claridad de la oferta, se escondía una realidad mucho más compleja. Se estaba gestando el nacimiento de la primera carretera de información por donde, en un futuro inmediato, se esperaba circulara una exorbitante cantidad de información de todo tipo, de carácter tanto público como privado, con fines de investigación y de negocio para grandes instituciones y para usuarios residenciales.

La concesión del proyecto supuso la privatización de la red y el acceso a la misma de todo tipo de usuarios. Se había colocado la primera piedra para la construcción de la Internet que conocemos actualmente. Posteriormente, MCI, IBM y Merit firman los papeles de constitución de la nueva sociedad en Delaware que adopta un nuevo nombre *Advanced Network and Services, Inc.* (ANS).

Desde la puesta en marcha del NSFNET, hubo desacuerdo por parte de varias organizaciones por el uso comercial del *backbone*, por lo que, el 14 de febrero de 1991 se creó la Asociación de Intercambios Comerciales en Internet (*Comercial Internet Exchange, CIX*). Un año después, en enero de 1992, se crea la *Internet Society*<sup>17</sup> (ISOC).

---

<sup>16</sup> *Backbone: Centro primario de conexión en un sistema de red distribuido. Todos los sistemas que se conectan al backbone pueden, a la vez, conectarse entre sí.*

<sup>17</sup> *Organización internacional para la interconexión de sistemas abiertos y la Internet. Sus miembros pueden ser individuos y organizaciones comerciales, gubernamentales o sin fines de lucro*

El CIX tenía la ventaja de ser independiente del NSFNET. La NSF tomó conciencia de que su negocio no estaba en el mantenimiento del *backbone* y diseñó un nuevo sistema basado en la experiencia observada con los CIX e inventó los Puntos de Acceso a la Red (*Network Access Points, NAP*); por lo que, el 30 de abril de 1995 la NSFNET vuelve a transformarse en una red de investigación.

Al mismo tiempo, comienzan a aparecer centros comerciales, bancos y pizzerías en Internet y, aparecen los famosos Proveedores de Servicios de Internet (*Internet Service Provider, ISP*), quienes por una cuota permiten al público en general conectarse a Internet a través del módem de la computadora y la línea telefónica común.

En el año 2000 existían más de 320 millones de usuarios alrededor del mundo, de los cuales, 15 millones eran de habla hispana.

## **2.2 Cómo funciona Internet**

Hasta el momento se han revisado conceptos históricos sobre Internet, ahora toca el turno de conocer uno de sus aspectos más importantes: Su funcionamiento, tema del presente capítulo.

### **2.2.1 TCP/IP (Transport Control Protocol/Internet Protocol)**

#### **2.2.1.1 HISTORIA**

En el mundo de la computación escuchamos frecuentemente el término TCP/IP y no es extraño, ya que es el protocolo asignado para la integración de redes tipo WAN (Red de Área Extensa/*Wide Area Network*) y, por consiguiente, de Internet.

Cuando se inició este experimento (finales de la década de los 60), fue conducido por el Departamento de Defensa de los Estados Unidos, por muy pocas universidades y, bajo la batuta de DARPA. El experimento fue un éxito y surgió de él una pila de protocolos relacionados entre sí con un solo objetivo: Conectividad entre diferentes tipos de computadoras. Rápidamente se convirtió en el estándar para redes de tipo Unix<sup>18</sup> y se extendió por todo el mundo su uso.

Son varias las asociaciones y organismos que intervienen en la regulación de TCP/IP, entre ellas podemos identificar a la ISOC como la más importante; ya que es la responsable de aprobar tecnologías que aplican a dicho protocolo.

---

<sup>18</sup> *UNIX: Sistema operativo multitarea (puede ejecutar varios programas simultáneamente) y multiusuario (puede gestionar a varios usuarios simultáneamente). Se desarrolló en 1969 en los laboratorios Bell (por Brian W. Kernighan y Ken Thompson), y aunque al principio se diseñó para el PDP-11, una máquina de Digital, ahora se ejecuta en gran cantidad de plataformas con microprocesadores distintos, haciéndolo un sistema multiplataforma, y provocando por tanto que un programa en código máquina ejecutable en una plataforma en UNIX no tenga por qué ser ejecutable en otra*

A continuación, revisaremos conceptos de vital importancia para la comprensión de TCP/IP.

### **PROTOCOLO DE COMUNICACIONES**

**Definición.** Es un estándar que proporciona las reglas para la comunicación; contiene los detalles referentes a los formatos de los mensajes, describe cómo responde una computadora cuando llega un mensaje y especifica de qué manera una computadora maneja un error u otras condiciones anormales.

Un aspecto importante, es que permite llevar a cabo la comunicación por computadora de manera independiente de cualquier *hardware* de red, es decir, un protocolo de comunicaciones permite especificar o entender la comunicación de datos, sin depender de un conocimiento detallado de una marca de *hardware* de red en particular.

El hacer a un lado los detalles de bajo nivel de la comunicación ayuda a mejorar la productividad de muchas maneras, como son:

- ✓ Los programadores no necesitan aprender o recordar tantos detalles sobre una configuración de *hardware* en particular, por lo que se pueden crear con rapidez nuevos programas
- ✓ Los programas no se encuentran restringidos a una sola arquitectura de computadora o a un solo tipo de *hardware* de red, por lo que no se necesitan cambiar cuando las computadoras o las redes se reconfiguran
- ✓ Puesto que los programas de aplicación son independientes del *hardware* subyacente, pueden proporcionar comunicación directa entre un par arbitrario de computadoras; por lo que los programadores no necesitan desarrollar versiones específicas de *software* de aplicación para mover y traducir los datos entre cada par de computadoras posibles.

En resumen, las funciones básicas de un protocolo de comunicación son: iniciar, mantener y terminar una comunicación.

### **MODELO DE REFERENCIA DE INTERCONEXIÓN DE SISTEMAS ABIERTOS (REFERENCE MODEL OF OPEN SYSTEMS INTERCONNECTION, OSI)**

El Modelo OSI, es un modelo de referencia para establecer las características que deben cumplir dos o más equipos que se conectan entre sí. Nace de la necesidad de homogeneizar la conexión entre redes. Cabe destacar que OSI no es una arquitectura particular, porque no especifica los detalles de los niveles sino, los estándares de ISO<sup>19</sup> que existen para cada nivel o capa que lo componen.

---

<sup>19</sup> Organización Internacional para la Estandarización (Internacional Standards Organization). Es una organización que ha definido un conjunto de protocolos diferentes, llamados protocolos ISO/OSI. Esta organización de carácter voluntario fue fundada en 1946 y es responsable de la creación de estándares internacionales en muchas áreas, incluyendo la informática, la ecológica y las comunicaciones.

Las capas del modelo OSI son siete (v. figura 2.2). Cada una es considerada como un módulo independiente, constituido por protocolos que particionan información para la comunicación entre computadoras.

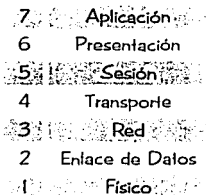


Figura 2.2 Capas del Modelo OSI

**Nivel Físico:** Se trata de la comunicación a nivel de interfaz de *hardware*. Las especificaciones de este nivel se refieren a las conexiones físicas, incluyendo los conectores, la configuración y las características de los voltajes en los cables.

**Nivel de Enlace de Datos:** Dado que el *hardware* entrega sólo un flujo de *bits*, este nivel debe definir el formato de las tramas<sup>20</sup>. En otras palabras se encarga de darle significado a los voltajes.

**Nivel de Red:** Contiene funciones que completan la interacción entre la computadora y la red. Este nivel define la unidad básica de transferencia a través de la red e incluye el concepto de ruteo<sup>21</sup>. Es decir, proporciona los mecanismos necesarios y protocolos para encaminar los paquetes de información para que lleguen a su destino.

**Nivel de Transporte:** Proporciona confiabilidad punto a punto<sup>22</sup> regulando el flujo de información, también puede proporcionar un transporte confiable, asegurando que los datos lleguen sin errores y en secuencia.

**Nivel de Sesión:** Parecido al nivel de transporte pero, proporciona un servicio administrativo que maneja el establecimiento y liberación de una conexión entre dos entidades de presentación. Las sesiones son establecidas cuando un proceso de aplicación pide acceso a otro proceso del mismo origen.

**Nivel de Presentación:** Está proyectada para incluir funciones que muchos programas de aplicación necesitan cuando utilizan la red. Los ejemplos comunes incluyen rutinas estándar que comprimen el texto o convierten las imágenes gráficas en flujos de *bits* para su transmisión a través de la red.

<sup>20</sup> Se refiere a la unidad de datos que viaja de una computadora a otra.

<sup>21</sup> Es el proceso de seleccionar una ruta sobre la cual se enviarán los paquetes de datos. Este proceso se inicia en un host TCP/IP, cuando envía paquetes IP a un ruteador y, su complementa con un ruteador IP.

<sup>22</sup> Comunicación entre un programa de aplicación y otro.

Nivel de Aplicación: Incluye programas de aplicación que utilizan la red. Por ejemplo: el correo electrónico, programas de transferencia de archivos, etc.

### 2.2.1.2 CONCEPTO Y ARQUITECTURA

TCP/IP no es un único protocolo, sino que es un conjunto de protocolos que cubren los distintos niveles del modelo OSI y son utilizados por todas las computadoras conectadas a Internet, de manera que éstas puedan comunicarse entre sí.

Se debe tomar en cuenta que a Internet se conectaban computadoras de clases muy diferentes y con *hardware* y *software* incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. Es aquí donde se encuentra una de las grandes ventajas del TCP/IP, pues este protocolo se encargará de que la comunicación entre todas ellas sea posible. TCP/IP es compatible con cualquier sistema operativo y con cualquier tipo de *hardware*.

Los dos protocolos más importantes son el TCP (*Transport Control Protocol*) y el IP (*Internet Protocol*), que son los que dan nombre al conjunto. La arquitectura del TCP/IP consta de cinco niveles o capas (v. figura 2.3) que, se encargan de realizar una función o tarea específica dentro de una red y también puede incluir diferentes protocolos.

**Aplicación:** Correspondiente con los niveles OSI de aplicación, presentación y sesión. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico SMTP (v. capítulo 2.3.2.7), transferencia de archivos FTP (v. capítulo 2.3.2.2), sesión remota TELNET (v. capítulo 2.3.2.1), entre otros. Interactúan con uno o más protocolos de transporte para enviar o recibir datos, en forma de mensajes o bien en forma de flujos de bytes.

**Transporte:** Correspondiente con el nivel de transporte del modelo OSI. Los protocolos de este nivel, tales como TCP y UDP, se encargan de manejar los datos y proporcionar la fiabilidad necesaria en el transporte de los mismos. Para hacer esto, se tiene el lado de recepción enviando acuses de recibo<sup>23</sup> de retorno y el lado de envío, retransmitiendo los paquetes perdidos. Se divide el flujo de datos que se está enviando en pequeños fragmentos (por lo general conocidos como paquetes) y pasa cada paquete con una dirección de destino, hacia la siguiente transmisión.

La capa de transporte debe aceptar datos desde varios programas de usuario y enviarlos a la siguiente capa. Para realizar esto se agrega información adicional a cada paquete, incluyendo códigos que identifican qué programa de aplicación envía y qué programa de aplicación debe recibir, así como, una suma de verificación. La computadora receptora utiliza la suma de verificación para comprobar que no haya sufrido alteraciones el paquete durante su transmisión y utiliza el código de destino para identificar el programa de aplicación al que debe entregar.

---

<sup>23</sup> Es un paquete de información con un formato propio que se utiliza para indicar la transmisión exitosa de otro paquete de información de un transmisor a un receptor.



Internet: Es el nivel de red del modelo OSI. Incluye al protocolo IP, que se encarga de enviar los paquetes de información a su destino correspondiente. Es utilizado con esta finalidad por los protocolos del nivel de transporte.

Para transmitir información a través de TCP/IP, ésta debe ser dividida en unidades de menor tamaño. Esto proporciona grandes ventajas en el manejo de los datos que se transfieren y, por otro lado, esto es algo común en cualquier protocolo de comunicaciones. En TCP/IP cada una de estas unidades de información recibe el nombre de datagrama (*datagram*), y son conjuntos de datos que se envían como mensajes independientes.

Red: Es la interfaz de la red real o física. Consta de una capa de interfaz de red responsable de aceptar los datagramas IP y transmitirlos hacia una red específica.

Físico: Análogo al nivel físico del OSI, se trata de la comunicación a nivel de interfaz de hardware. Las especificaciones de este nivel se refieren a las conexiones físicas, incluyendo los conectores, la configuración y los voltajes en los cables.

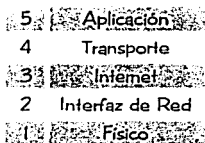


Figura 2.3 Capas del Modelo TCP/IP

### 2.2.1.3 CARACTERÍSTICAS

Ya que dentro de un sistema TCP/IP los datos transmitidos se dividen en pequeños paquetes, éstos resaltan una serie de características.

La tarea de IP es llevar los paquetes de un sitio a otro. Las computadoras que encuentran las vías para llevar los datos de una red a otra (ruteadores<sup>24</sup>) utilizan IP para trasladar los datos. En resumen, IP mueve los paquetes de datos mientras TCP se encarga del flujo y asegura que los datos estén correctos.

Las líneas de comunicación se pueden compartir entre varios usuarios.

Cualquier tipo de paquete puede transmitirse al mismo tiempo y se ordenará y combinará cuando llegue a su destino. Similar a la manera en que se transmite una conversación telefónica. Una

---

<sup>24</sup> Ruteador (Router): Palabra inglesa que se significa "encaminador". Es un dispositivo que tiene dos ó mas interfaces de red. Este examina las direcciones de los protocolos que se utilizan en la red y selecciona una ruta apropiada de viaje para la información y manda un paquete de reconocimiento entre las diferentes redes. Los ruteadores usualmente mandan paquetes correspondientes a una simple familia de protocolos.

vez que se establece una conexión, se reservan algunos circuitos para la persona quien hace la llamada, que no puede emplear en otra llamada, aún si deja esperando a su interlocutor por veinte minutos.

Los datos no tienen que enviarse directamente entre dos computadoras. Cada paquete pasa de computadora en computadora hasta llegar a su destino. Éste, claro está, es el secreto de cómo se pueden enviar datos y mensajes entre dos computadoras aunque no estén conectadas directamente entre sí. Lo que realmente sorprende es que sólo se necesitan algunos segundos para enviar un archivo de miles de bytes de una computadora a otra, aunque estén separadas por miles de kilómetros y pese a que los datos tienen que pasar por múltiples computadoras. Una de las razones de la rapidez es que, cuando algo anda mal, sólo es necesario volver a transmitir un paquete, no todo el mensaje.

Los paquetes no necesitan seguir la misma trayectoria. La red puede llevar cada paquete de un lugar a otro y usar la conexión más idónea que esté disponible en ese instante. No todos los paquetes de los mensajes tienen que viajar por la misma ruta ni tienen que llegar todos al mismo tiempo.

La flexibilidad del sistema lo hace muy confiable. Si un enlace se pierde, el sistema usa otro. Cuando se envía un mensaje, el TCP divide los datos en paquetes, ordena éstos en secuencia, agrega cierta información para control de errores y después los lanza hacia afuera y los distribuye. En el otro extremo, el TCP recibe los paquetes, verifica si hay errores y los vuelve a combinar para convertirlos en los datos originales. De haber error en algún punto, el programa TCP destino envía un mensaje solicitando que se vuelvan a enviar determinados paquetes.

#### 2.2.1.4 PROTOCOLOS

- ✓ FTP (*File Transfer Protocol*): Se utiliza para transferencia de archivos.
- ✓ SMTP (*Simple Mail Transfer Protocol*): Es una aplicación para el correo electrónico.
- ✓ TELNET: Permite la conexión a una aplicación remota desde un proceso o terminal.
- ✓ RPC (*Remote Procedure Call*): Permite llamadas a procedimientos situados remotamente. Se utilizan las llamadas a RPC como si fuesen procedimientos locales.
- ✓ SNMP (*Simple Network Management Protocol*): Se trata de una aplicación para el control de la red.
- ✓ NFS (*Network File System*): Permite la utilización de archivos distribuidos por los programas de la red.
- ✓ X-Window: Es un protocolo para el manejo de ventanas e interfaces de usuario.
- ✓ ICMP (*Internet Control Message Protocol*): Protocolo que resuelve errores y controla mensajes; específicamente los anfitriones y los ruteadores utilizan este protocolo para enviar reportes de problemas relacionados con datagramas que se devuelven a la fuente original que envía el datagrama. El ICMP también incluye una solicitud/réplica de eco utilizada para probar si un destino es accesible y responde.
- ✓ X.25: Protocolo estándar para el servicio de red a nivel transporte.

### 2.2.1.5 CÓMO FUNCIONA TCP/IP

Los protocolos de comunicación, de acuerdo a su tipo de conexión, se clasifican en:

1. No orientados a la conexión o de baja conexión (*connection less*): Son aquellos en los que no existe una coordinación entre las computadoras fuente y destino para la certificación de la correcta transmisión de la información.

Proporcionan un servicio inestable ya que no se garantiza la entrega, sin embargo, sólo fallan por insuficiencia de recursos o en los componentes de la red. En caso de existir error, no realizará acción alguna para recuperar la información. Debido a su simplicidad, éstos protocolos son más rápidos y menos costosos por realizar menos funciones de control durante la transmisión. Se usan generalmente en las redes LAN (Red de Área Local/*Local Area Network*) debido a su baja tasa de errores de transmisión.

2. Orientados a la conexión: Son aquellos en los que al contrario de los no orientados a la conexión, existe una coordinación entre las computadoras fuente y destino para la certificación de la correcta transmisión de la información. Estos protocolos utilizan funciones de control de la transmisión de la información.

### 2.2.1.6 IP

IP es el protocolo primeramente responsable del direccionamiento<sup>25</sup> y ruteo de los paquetes entre los *hosts*.

#### CARACTERÍSTICAS

- ✓ Es un protocolo no orientado a la conexión, es decir, no establece una sesión antes del intercambio de datos.
- ✓ Fragmenta y reensambla los paquetes.
- ✓ Es un protocolo no orientado a la conexión, es decir, no establece una sesión antes del intercambio de datos.
- ✓ No garantiza la entrega de paquetes, siempre hace su mejor esfuerzo pero en el camino puede ser extraviado, salir de secuencia o duplicado.
- ✓ Está basado en la idea de los datagramas interred, los cuales son transportados transparentemente, pero no siempre con seguridad, desde la computadora fuente hasta la computadora destino quizás recorriendo varias redes mientras viaja.

El protocolo IP trabaja de la siguiente manera: La capa de transporte toma los mensajes y los divide en datagramas de hasta 64 octetos cada uno. Cada datagrama se transmite a través de la red interred, posiblemente fragmentándose en unidades más pequeñas, durante su recorrido normal. Al

---

<sup>25</sup> Es la asignación de una dirección (canónica o numérica) de Internet. La forma canónica es más fácil de leer, por ejemplo *minombre@tcpip.com*. La forma numérica es un número binario sin signo de 32 bits, habitualmente expresado en forma de números decimales separados por puntos. Por ejemplo, *9.167.5.8* es una dirección de Internet válida. La forma numérica es usada por el software de IP. La función de mapeo entre los dos la realiza el DNS.

final, cuando todas las piezas llegan a la computadora destinataria, la capa de transporte le reensambla para así reconstruir el mensaje original. Un datagrama IP consta de una parte de cabecera (v. figura 2.4) y una parte de texto. La cabecera tiene una sección fija de 20 octetos y una opcional de longitud variable.

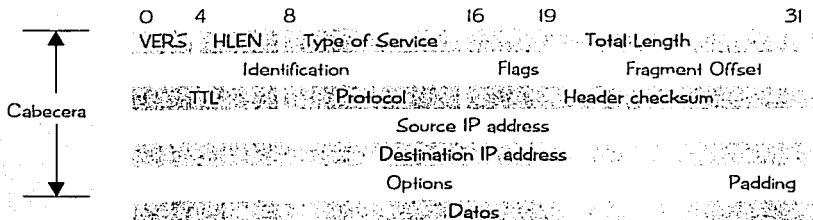


Figura 2.4 Cabecera del datagrama IP

El campo *VERS* (Versión), indica a qué versión del protocolo pertenece cada uno de los datagramas. Mediante la inclusión de la versión en cada datagrama, no se excluye la posibilidad de modificar los protocolos mientras la red se encuentre en operación.

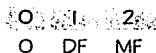
El campo *HLEN* (Longitud de la Cabecera), debido a que la longitud de cabecera no es constante, aquí se indica la longitud que tiene la cabecera en palabras de 32 bits. El valor mínimo es de 5. Tamaño 4 bits.

El campo *Type of Service* (Tipo de Servicio), permite indicarle a la subred el tipo de servicio que desea. Es posible tener varias combinaciones con respecto a la seguridad y la velocidad. Para voz digitalizada, por ejemplo, es más importante la entrega rápida que corregir errores de transmisión. En tanto que, para la transferencia de archivos, resulta más importante tener la transmisión fiable que entrega rápida. También es posible tener algunas otras combinaciones, desde un tráfico rutinario, hasta una anulación instantánea. Tamaño 8 bits.

La *Total Length* (Longitud Total), incluye todo lo que se encuentra en el datagrama (tanto la cabecera como los datos). La máxima longitud es de 65 536 octetos (bytes). Tamaño 16 bits.

El campo *Identification* (Identificación), se necesita para permitir que la computadora destino determine a qué datagrama pertenece el fragmento recién llegado. Todos los fragmentos de un datagrama contienen el mismo valor de identificación. Tamaño 16 bits.

La estructura del campo *Flags* (Banderas), es:



final, cuando todas las piezas llegan a la computadora destinataria, la capa de transporte los reensambla para así reconstruir el mensaje original. Un datagrama IP consta de una parte de cabecera (v. figura 2.4) y una parte de texto. La cabecera tiene una sección fija de 20 octetos y una opcional de longitud variable

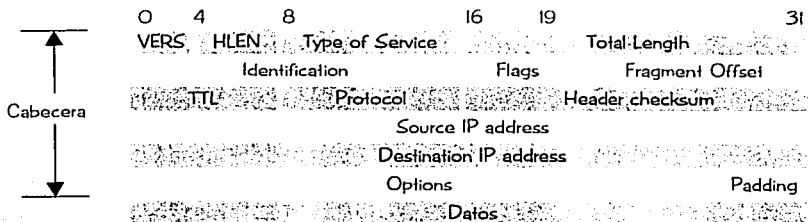


Figura 2.4 Cabecera del datagrama IP

El campo *VERS* (Versión), indica a qué versión del protocolo pertenece cada uno de los datagramas. Mediante la inclusión de la versión en cada datagrama, no se excluye la posibilidad de modificar los protocolos mientras la red se encuentre en operación.

El campo *HLEN* (Longitud de la Cabecera), debido a que la longitud de cabecera no es constante, aquí se indica la longitud que tiene la cabecera en palabras de 32 bits. El valor mínimo es de 5 Tamaño 4 bits

El campo *Type of Service* (Tipo de Servicio), permite indicarle a la subred el tipo de servicio que desea. Es posible tener varias combinaciones con respecto a la seguridad y la velocidad. Para voz digitalizada, por ejemplo, es más importante la entrega rápida que corregir errores de transmisión. En tanto que, para la transferencia de archivos, resulta más importante tener la transmisión fiable que entrega rápida. También es posible tener algunas otras combinaciones, desde un tráfico rutinario, hasta una anulación instantánea. Tamaño 8 bits.

La *Total Length* (Longitud Total), incluye todo lo que se encuentra en el datagrama (tanto la cabecera como los datos). La máxima longitud es de 65 536 octetos (bytes). Tamaño 16 bits.

El campo *Identification* (Identificación), se necesita para permitir que la computadora destino determine a qué datagrama pertenece el fragmento recién llegado. Todos los fragmentos de un datagrama contienen el mismo valor de identificación. Tamaño 16 bits.

La estructura del campo *Flags* (Banderas), es:

0 0 1 1 2 3  
O DF MF

Primero un bit que no se utiliza (debe ser 0), y después dos campos de 1 bit. Las letras DF quieren decir no fragmentar; ésta, es una orden para que los roteadores no fragmenten el datagrama, porque el extremo destinatario es incapaz de poner las partes juntas nuevamente.

Las letras MF significan más fragmentos. Todos los fragmentos, con excepción del último, deberán tener ese bit puesto (es decir, el bit debe tener 1). Se utiliza como una verificación doble contra el campo de Longitud total, con objeto de tener seguridad de que no faltan fragmentos y que el datagrama entero se reensamble por completo.

El campo *Fragment Offset* (Desplazamiento de Fragmento), usado en datagramas fragmentados, indica el lugar del datagrama actual al cual pertenece este fragmento. En un datagrama, todos los fragmentos, con excepción del último, deberán ser un múltiplo de 8 octetos, que es la unidad elemental de fragmentación. Dado que se proporcionan 13 bits, hay un máximo de 8192 fragmentos por datagrama, dando así una longitud máxima de datagrama de 65 536 octetos, que coinciden con el campo Longitud total. Tamaño 16 bits.

El campo *TTL o Time to Live* (Tiempo de Vida), es un contador que se utiliza para limitar el tiempo de vida de los paquetes, especifica el tiempo que se le permite viajar al datagrama. Cuando se llega a cero, el paquete se destruye. La unidad de tiempo es el segundo, permitiéndose un tiempo de vida máximo de 255 segundos. Tamaño 8 bits.

Cuando la capa de red ha terminado de ensamblar un datagrama completo, necesitará saber qué hacer con él.

El campo *Protocol* (Protocolo), indica a qué proceso de transporte pertenece el datagrama. El TCP es efectivamente una posibilidad, pero en realidad hay muchas más. El número utilizado en este campo sirve para indicar a qué protocolo pertenece el datagrama que se encuentra a continuación de la cabecera IP, de manera que pueda ser tratado correctamente cuando llegue a su destino. Tamaño 8 bits.

El *Header Checksum* (Código de Redundancia de la Cabecera), es necesario para verificar si los datos contenidos en la cabecera IP son correctos. Por razones de eficiencia este campo no puede utilizarse para comprobar la información del usuario, ésta se comprobará posteriormente a partir del código de redundancia de la cabecera siguiente y que corresponde al nivel de transporte. Este campo debe calcularse de nuevo cuando cambia alguna opción de la cabecera, como puede ser el tiempo de vida. Tamaño 16 bits.

El campo *Source IP Address* (Dirección de Origen), contiene la dirección del *host* que envía el paquete. Tamaño 32 bits.

El campo *Destination IP Address* (Dirección Destino), es la dirección del *host* que recibirá la información. Los *routers* (roteadores) o *gateways* intermedios deben conocerla para dirigir correctamente el paquete. Tamaño 32 bits.

Primero un bit que no se utiliza (debe ser 0), y después dos campos de 1 bit. Las letras DF quieren decir no fragmentar; ésta, es una orden para que los ruteadores no fragmenten el datagrama, porque el extremo destinatario es incapaz de poner las partes juntas nuevamente.

Las letras MF significan más fragmentos. Todos los fragmentos, con excepción del último, deberán tener ese bit puesto (es decir, el bit debe tener 1). Se utiliza como una verificación doble contra el campo de Longitud total, con objeto de tener seguridad de que no faltan fragmentos y que el datagrama entero se reensamble por completo.

El campo *Fragment Offset* (Desplazamiento de Fragmento), usado en datagramas fragmentados, indica el lugar del datagrama actual al cual pertenece este fragmento. En un datagrama, todos los fragmentos, con excepción del último, deberán ser un múltiplo de 8 octetos, que es la unidad elemental de fragmentación. Dado que se proporcionan 13 bits, hay un máximo de 8192 fragmentos por datagrama, dando así una longitud máxima de datagrama de 65 536 octetos, que coinciden con el campo Longitud total. Tamaño 16 bits.

El campo *TTL o Time to Live* (Tiempo de Vida), es un contador que se utiliza para limitar el tiempo de vida de los paquetes, especifica el tiempo que se le permite viajar al datagrama. Cuando se llega a cero, el paquete se destruye. La unidad de tiempo es el segundo, permitiéndose un tiempo de vida máximo de 255 segundos. Tamaño 8 bits.

Cuando la capa de red ha terminado de ensamblar un datagrama completo, necesitará saber qué hacer con él.

El campo *Protocol* (Protocolo), indica a qué proceso de transporte pertenece el datagrama. El TCP es efectivamente una posibilidad, pero en realidad hay muchas más. El número utilizado en este campo sirve para indicar a qué protocolo pertenece el datagrama que se encuentra a continuación de la cabecera IP, de manera que pueda ser tratado correctamente cuando llegue a su destino. Tamaño 8 bits.

El *Header Checksum* (Código de Redundancia de la Cabecera), es necesario para verificar si los datos contenidos en la cabecera IP son correctos. Por razones de eficiencia este campo no puede utilizarse para comprobar la información del usuario, ésta se comprobará posteriormente a partir del código de redundancia de la cabecera siguiente y que corresponde al nivel de transporte. Este campo debe calcularse de nuevo cuando cambia alguna opción de la cabecera, como puede ser el tiempo de vida. Tamaño 16 bits.

El campo *Source IP Address* (Dirección de Origen), contiene la dirección del *host* que envía el paquete. Tamaño 32 bits.

El campo *Destination IP Address* (Dirección Destino), es la dirección del *host* que recibirá la información. Los *routers* (ruteadores) o *gateways* intermedios deben conocerla para dirigir correctamente el paquete. Tamaño 32 bits.

El campo *Options* (Opciones), se utiliza para fines de seguridad, encaminamiento fuente, informe de errores, depuración, sellado de tiempo, así como otro tipo de información. Esto, básicamente, proporciona un escape para permitir que las siguientes versiones de los protocolos incluyan información que actualmente no está presente en el diseño original. También, para permitir que los investigadores trabajen con nuevas ideas y para evitar, la asignación de *bits* de cabecera a información que muy rara vez se necesita.

El campo *Padding* (Relleno), si se usa una opción el datagrama se rellena con bytes a cero hasta la siguiente palabra *32 bits*.

### LA DIRECCIÓN DE INTERNET

El protocolo IP identifica a cada computadora que se encuentra conectada a la red mediante su correspondiente dirección (v. figura 2.5) la cual, está representada por un número de *32 bits* que debe ser única para cada *host* y, normalmente, suele representarse como cuatro cifras de *8 bits* separadas por puntos.

La dirección de Internet (*IP Address*) se utiliza para identificar tanto a la computadora en concreto como la red a la que pertenece, de manera que sea posible distinguir a las computadoras que se encuentran conectadas a una misma red. Con este propósito, y teniendo en cuenta que en Internet se encuentran conectadas redes de tamaños muy diversos, se establecieron tres clases primarias de direcciones: representadas de acuerdo a tres rangos de valores:

Clase A: Son las que en su primer byte tienen un valor comprendido entre 1 y 126, incluyendo ambos valores. Estas direcciones utilizan únicamente este primer byte para identificar la red, quedando los otros tres bytes disponibles para cada uno de los *hosts* que pertenezcan a esta misma red. Esto significa que podrán existir más de dieciséis millones de computadoras en cada una de las redes de esta clase.

Este tipo de direcciones es usado por redes muy extensas, pero hay que tener en cuenta que sólo puede haber 126 redes de este tamaño. ARPANET es una de ellas, existiendo además algunas grandes redes comerciales, aunque son pocas las organizaciones que obtienen una dirección de esta clase. Lo normal para las grandes organizaciones es que utilicen una o varias redes de clase B.

Clase B: Estas direcciones utilizan en su primer byte un valor comprendido entre 128 y 191, incluyendo ambos. En este caso, el identificador de la red se obtiene de los dos primeros bytes de la dirección, teniendo que ser un valor entre 128.1 y 191.254 (no es posible utilizar los valores 0 y 255 por tener un significado especial). Los dos últimos bytes de la dirección constituyen el identificador del *host* permitiendo, por consiguiente, un número máximo de 64516 computadoras en la misma red.

Este tipo de direcciones tendría que ser suficiente para la gran mayoría de las organizaciones grandes. En caso de que el número de computadoras que se necesitará conectar fuera mayor, sería posible obtener más de una dirección de esta clase, evitando de esta forma, el uso de una de clase A.



Clase C: En este caso el valor del primer byte tendrá que estar comprendido entre 192 y 223, incluyendo ambos valores. Este tercer tipo de direcciones utiliza los tres primeros bytes para el número de la red con un rango desde 192.1 hasta 223.254.254. De esta manera, queda libre un byte para el *host*, lo que permite que se conecten un máximo de 254 computadoras en cada red. Estas direcciones permiten un menor número de *host* que las anteriores, aunque son las más numerosas pudiendo existir un gran número de redes de esta clase (más de dos millones).

Clase	Primer byte	Identificación de red	Identificación de hosts	Número de redes	Número de hosts
A	1...126	1 byte	3 byte	126	16.387.064
B	128...191	2 byte	2 byte	16.256	64.516
C	192...223	3 byte	1 byte	2.064.512	254

Figura 2.5 Tabla de direcciones IP de Internet

En la clasificación de direcciones anterior, se puede notar que ciertos números no se usan, ya que, algunos se encuentran reservados para un posible uso futuro, como es el caso de las direcciones cuyo primer byte sea superior a 223 (clases D y E, que aún no están definidas), mientras que el valor 127 en el primer byte se utiliza en algunos sistemas para propósitos especiales (como pruebas de conectividad). También es importante notar que los valores 0 y 255 en cualquier byte de la dirección no pueden usarse normalmente por tener otros propósitos específicos.

El número 0 está reservado para las computadoras que no conocen su dirección, pudiendo utilizarse tanto en la identificación de red para computadoras que aún no conocen el número de red a la que se encuentran conectadas, como en la identificación de *host* para computadoras que aún no conocen su número de *host* dentro de la red, o en ambos casos.

El número 255 tiene también un significado especial, puesto que se reserva para el *broadcast*<sup>26</sup>. El *broadcast* es necesario cuando se pretende hacer que un mensaje sea visible para todos los sistemas conectados a la misma red. Esto puede ser útil si se necesita enviar el mismo datagrama a un número determinado de sistemas, resultando más eficiente que enviar la misma información solicitada de manera individual a cada uno. Otra situación para el uso de *broadcast* es cuando se quiere convertir el nombre por dominio de una computadora a su correspondiente número IP y no se conoce la dirección del Servidor de Nombres de Dominio (DNS<sup>27</sup>) más cercano.

<sup>26</sup>: *Broadcast*: Tecnología de transmisión en la que un nodo se comunica absolutamente con todos los nodos conectados a él, en una sola transmisión.

<sup>27</sup>: *Sistema de Nombre de Dominios (Domain Name System, DNS)*: Se trata de una base de datos distribuida que gestiona todo lo referente a la conversión de nombres de dominio en formato alfanumérico (lenguaje escrito) al sistema real de direcciones de Internet formado por grupos de 4 números a las cuales se llama direcciones IP. Funciona como un servicio de búsqueda de datos de uso general, distribuido y multiplicado. Su utilidad principal es la búsqueda de direcciones IP de sistemas anfitriones (*hosts*) basándose en los nombres de éstos. El estilo de los nombres de *host* utilizado actualmente en Internet es llamado "nombre de dominio".

Lo usual es que cuando se quiere hacer uso del *broadcast*, se utilice una dirección compuesta por el identificador normal de la red y por el número 255 (todo unos; en binario) en cada byte que identifique al *host*. Sin embargo, por conveniencia también se permite el uso del número 255 255 255 con la misma finalidad, de forma que resulte más simple referirse a todos los sistemas de la red.

El *broadcast* es una característica que se encuentra implementada de formas diferentes dependiendo del medio utilizado, y por lo tanto, no siempre se encuentra disponible. En las líneas punto a punto no es posible enviar *broadcast*, pero si es posible hacerlo en las redes Ethernet<sup>28</sup> donde se supone que todas las máquinas prestarán atención a este tipo de mensajes.

En el caso de algunas organizaciones extensas puede surgir la necesidad de dividir la red en otras redes más pequeñas (subnets). Como ejemplo podemos suponer una red de clase B que, naturalmente, tiene asignado como identificador de red un número de dos bytes. En este caso sería posible utilizar el tercer byte para indicar en qué red Ethernet se encuentra un *host* en concreto.

Esta división no tendrá ningún significado para cualquier otra máquina que esté conectada a una red perteneciente a otra organización, puesto que el tercer byte no será comprobado ni tratado de forma especial. Sin embargo, en el interior de esta red existirá una división y será necesario disponer de un *software* de red especialmente diseñado para ello. De esta forma, queda oculta la organización interior de la red, siendo mucho más cómodo el acceso que si se tratara de varias direcciones de clase C independientes.

### 2.2.1.7 TCP

El Protocolo TCP pertenece al nivel de transporte, se encarga de dividir el mensaje original en segmentos de menor tamaño, y por lo tanto, mucho más manejables. Los segmentos serán dirigidos a través del protocolo IP de forma individual. El protocolo TCP se encarga además de añadir cierta información necesaria a cada uno de los segmentos. Esta información se adiciona al inicio de los datos que componen el segmento en forma de cabecera.

La cabecera de un segmento contiene al menos 160 *bits* que se encuentran repartidos en varios campos con diferente significado. Cuando la información se divide en segmentos para ser enviados, el orden en que éstos lleguen a su destino no tiene que ser el correcto o secuencial. Cada uno de ellos puede llegar en cualquier momento y en cualquier orden e incluso puede que algunos no lleguen a su destino o lleguen con información errónea.

---

<sup>28</sup> Sistema de red de área local de alta velocidad desarrollado por las compañías Xerox, Digital e Intel, que se ha convertido en un estándar en lo que a métodos de acceso LAN respecta.

Para evitar todos estos problemas, el TCP numera los segmentos antes de ser enviados, de manera que sea posible volver a unirlos en el orden adecuado. Esto permite solicitar nuevamente, el envío de los segmentos individuales que no hayan llegado o que contengan errores, sin que sea necesario volver a enviar el mensaje completo.

El segmento de TCP consta de dos partes: Cabecera (v. figura 2.6) y Datos. La cabecera transporta la identificación y la información de control.

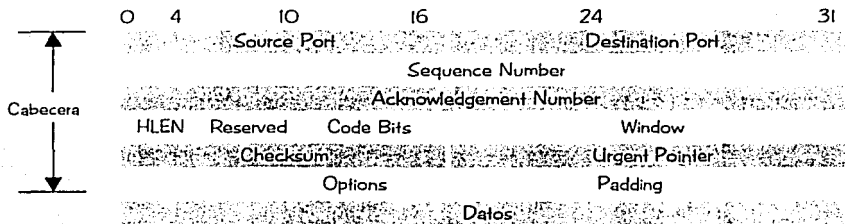


Figura 2.6 Cabecera del segmento de TCP

El campo *Source Port* (Puerto Origen) y el campo *Destination Port* (Puerto Destino), contienen los números del puerto TCP que identifican a los programas de aplicación en los extremos de la conexión.

El campo *Sequence Number* (Número de Secuencia), identifica la posición de los datos del segmento en el flujo de datos del transmisor.

El campo *Acknowledgement Number* (Número de Acuse de Recibo), identifica el número de octetos que la fuente espera recibir después. Este campo tiene un tamaño de 32 bits.

El campo *HLEN* (Longitud de la Cabecera), contiene la longitud total de la cabecera TCP expresada en el número de palabras de 32 bits que ocupa. Esto permite determinar el lugar donde comienzan los datos.

El campo *Reserved* (Reservado), está reservado para usarse en el futuro y es de 6 bits.

El campo *Code Bits* (Bits de Control), TCP utiliza este campo de 6 bits para determinar el propósito y contenido del segmento; los 6 bits indican cómo interpretar otros campos de la cabecera de acuerdo a la siguiente tabla.

Bit	Significado si el bit está en 1
URG	El campo de puntero de urgencia es válido
ACK	El campo de acuse de recibo es válido
PSH	Este segmento solicita una operación push, en donde el receptor no debiera almacenar los datos antes de entregarlos
RST	Iniciación de la conexión
SYN	Sincronizar número de secuencia, se usa para establecer las conexiones
FIN	Indica que el emisor no tiene más datos que mandar

El campo *Window* (Ventana), es de 16 bits y en él se introduce un valor indicando la cantidad de información que el receptor está preparado para procesar. Si el valor llega a cero, será necesario que el emisor se detenga. A medida que la información es procesada, este valor aumenta indicando disponibilidad para continuar la recepción de datos.

El campo *Checksum* (Suma de Verificación), puede ocurrir que la información del segmento llegue con errores a su destino. Para poder detectar cuando sucede esto, se incluye en la cabecera un campo de 16 bits, el cual, contiene un valor calculado a partir de la información del datagrama completo *checksum*. En el otro extremo el receptor vuelve a calcular este valor, comprobando que es el mismo que el suministrado en la cabecera. Si el valor es distinto, significaría que el segmento es incorrecto, ya que, en la cabecera o en la parte de datos del mismo hay algún error.

El campo *Urgent Pointer* (Puntero de Urgencia), apunta al primer byte de datos que sigue a los datos urgentes. Sólo es significativo cuando el bit de control está activo.

Después de la cabecera puede existir información opcional. En cualquier caso, el tamaño de la cabecera debe ser múltiplo de 32 bits, por lo que, puede ser necesario añadir un campo de tamaño variable y que contenga ceros al final para conseguir este objetivo cuando se incluyen algunas opciones (*Options*).

Para establecer una conexión, TCP utiliza el mecanismo *handshake* (saludo), este mecanismo ejecuta dos funciones importantes:

- 1 Garantiza que ambos lados (emisor y receptor) estén listos para transferir datos y que ambos tengan conocimiento de ello.
- 2 Les permite a emisor y receptor acordar un número de secuencia inicial; cada computadora debe seleccionar un número de secuencia inicial de manera aleatoria que se utilizará para identificar octetos en el flujo que se está enviando (el número de secuencia no puede iniciar siempre con el mismo valor).

El mecanismo *handshake* consta de tres etapas (v. figura 2.7).

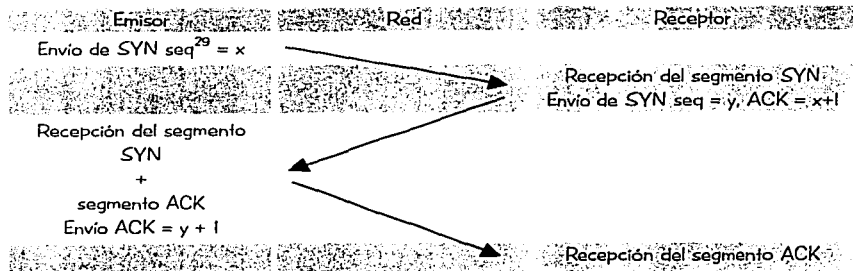


Figura 2.7 Mecanismo *Handshake*

El primer segmento del saludo envía una petición de conexión al receptor y se identifica, porque este segmento tiene activo el bit *SYN* en el campo de código. El emisor transfiere un número de secuencia inicial  $x$ .

El segundo segmento tiene tanto el bit *SYN* como el bit *ACK* activos, indicando el acuse de recibo del primer segmento *SYN* enviado por el emisor, para indicar que se continúa con el intercambio. El receptor recibe el *SYN*, registra el número de secuencia y responde enviando su número de secuencia inicial en el campo de secuencia, así como, un reconocimiento que especifica que espera el octeto  $x + 1$ .

El tercer y último mensaje, es únicamente un acuse de recibo y se utiliza para informar al receptor que ambos extremos están de acuerdo en establecer una conexión. El emisor envía al receptor el acuse de recibo, informando que recibió todos los octetos a través del número de secuencia; los acuses de recibo siguen la convención de utilizar el número del próximo octeto esperado.

### EN QUÉ SE UTILIZA TCP/IP

Muchas grandes redes han sido implementadas con estos protocolos, incluyendo ARPANET. De igual forma, una gran variedad de universidades, agencias gubernamentales y empresas de computadoras están conectadas mediante los protocolos TCP/IP. Cualquier computadora de la red puede comunicarse con otra distinta, esta conectividad permite enlazar redes físicamente independientes, en una red virtual llamada Internet.

#### 2.2.1.8 CAMBIOS EN TCP/IP

Debido a que los protocolos TCP/IP son la base del funcionamiento de la Internet, están vinculados a la evolución de esta última, así que, los procesos que estimulan la evolución de TCP/IP se pueden clasificar en las siguientes cuatro categorías:

---

<sup>29</sup> Abreviatura de número de secuencia

## NUEVAS TECNOLOGÍAS

Se refiere a las nuevas tecnologías de computación y de comunicación, es decir, la gente orientada a la tecnología como los ingenieros e investigadores que cuando sale al mercado una nueva computadora con capacidades superiores a las actuales inmediatamente la utilizan como *hosts* o ruteadores. En cuanto a las nuevas tecnologías de redes, las utilizan para transportar datagramas IP. Por ejemplo, ciertos investigadores han estudiado las redes inalámbricas que se basan en luz infrarroja.

## NUEVAS APLICACIONES

Las nuevas aplicaciones requieren de servicios que los protocolos actuales no pueden proporcionar, así como, de otra infraestructura. Por ejemplo, el gran interés que se tiene en multimedia, requiere de protocolos que puedan transferir imágenes y sonido eficientemente.

## INCREMENTOS EN LA CARGA Y TAMAÑO

En los últimos años ha crecido de manera exponencial el uso de la Internet, ahora no solamente es usada por académicos e investigadores sino que la usa todo tipo de gente con distinto objetivo. El uso de aplicaciones que transfieren multimedia genera más tráfico en la red, de igual manera, las herramientas de búsqueda automatizada generan gran cantidad de tráfico y hacen lento el sondeo de páginas o sitios de Internet para encontrar información.

## NUEVAS POLÍTICAS

Debido a la expansión de la Internet a diferentes países e industrias, adquiere nuevas autoridades y cada una de estas producen cambios en las políticas administrativas definiéndose así, nuevos mecanismos para reforzar dichas políticas. Lo anterior no parará mientras se sigan agregando más redes a la Internet, por lo que muchos investigadores e ingenieros continúan enfocados a encontrar formas de que TCP/IP se pueda adaptar a los nuevos grupos administrativos.

### 2.2.2 IP VERSIÓN 4

En los años 1992-1993 se plantearon los siguientes problemas con la Versión 4 IP, debido al espectacular crecimiento de Internet:

- ✓ Se ve cercano el agotamiento de las direcciones Clase B: Las direcciones Clase C sólo permiten 255 *hosts*, mientras que las de Clase B permiten 65535. En la mayoría de las redes, la Clase C resulta demasiado pequeña, y la B demasiado grande.
- ✓ El crecimiento de las tablas de ruteo empieza a hacerlas intratables para el *software* y *hardware* existente.
- ✓ El espacio de 32 bits para direcciones comienza a resultar escaso.

Los dos primeros problemas son los que se intentan resolver mediante la estrategia de Dominio de Ruteo sin Clase (*Classless Internet Domain Routing*, CIDR) es la primera orientación que permite redistribuir una parte de las direcciones subutilizadas. Sin embargo, esto es transitorio pues, se corre el riesgo de encontrar el mismo problema en menos de diez años.

El tercer problema (el agotamiento del espacio de 32 bits para direcciones), resulta irresoluble en el marco de la versión 4 del protocolo IP.

El IP tiene únicamente la misión de encaminar el datagrama sin comprobar la integridad de la información que contiene. Para ello, se utiliza una nueva cabecera (v. figura 2.8) que se antepone al datagrama. La cabecera IP tiene un tamaño de 160 bits y está formada por varios campos de distinto significado. Estos campos son:

- ✓ *Version* (Versión del protocolo IP utilizado), deberá tener el valor 4. Tamaño 4 bits.
- ✓ *Internet Header Length* (Longitud de la Cabecera, IHL), especifica la longitud de la cabecera expresada en el número de grupos de 32 bit que contiene. Tamaño 4 bit.
- ✓ *Type of Service* (Tipo de Servicio), el tipo o calidad de servicio se utiliza para indicar la prioridad o importancia de los datos que se envían, lo que condicionará la forma en que éstos serán tratados durante la transmisión. Tamaño 8 bits.
- ✓ *Total Length* (Longitud Total), es la longitud en bytes del datagrama completo, incluyendo la cabecera y los datos. Tamaño 16 bits.
- ✓ *Identification* (Identificación), valor de identificación que se utiliza para facilitar el ensamblaje de los fragmentos del datagrama. Tamaño 16 bits.
- ✓ *Flags* (Banderas), indicadores utilizados en la fragmentación. Tamaño 3 bits.
- ✓ *Fragment Offset* (Desplazamiento de Fragmento), usado en datagramas fragmentados, indica el lugar del datagrama actual al cual pertenece este fragmento. Está expresado en número de grupos de 8 bytes (64 bits), comenzando con el valor cero para el primer fragmento. Tamaño 13 bits.
- ✓ *TTL o Time to Live* (Tiempo de Vida), es un contador que se utiliza para limitar el tiempo de vida de los paquetes, especifica el tiempo que se le permite viajar al datagrama; contiene un número que disminuye cada vez que el paquete pasa por un sistema. Si este número llega a cero, el paquete será descartado. Tamaño 8 bits.
- ✓ *Protocol* (Protocolo), el número utilizado en este campo sirve para indicar a qué protocolo pertenece el datagrama que se encuentra a continuación de la cabecera IP, de manera que pueda ser tratado correctamente cuando llegue a su destino. Tamaño 8 bits.
- ✓ *Header Checksum* (Código de Redundancia de la Cabecera o Campo de Comprobación), es necesario para verificar que los datos contenidos en la cabecera IP son correctos. Por razones de eficiencia este campo no puede utilizarse para comprobar los datos incluidos a continuación de la cabecera IP, sino que estos datos de usuario se comprobarán posteriormente a partir del campo de comprobación de la cabecera siguiente, y que corresponde al nivel de transporte. Este campo debe calcularse de nuevo cuando cambia alguna opción de la cabecera, como puede ser el tiempo de existencia. Tamaño 16 bits.
- ✓ *Source IP Address* (Dirección Origen), contiene la dirección del host que envía el paquete. Tamaño 32 bits.
- ✓ *Destination IP Address* (Dirección Destino), es la dirección del host que recibirá la información. Los ruteadores intermedios deben conocerla para dirigir correctamente el paquete. Tamaño 32 bits.

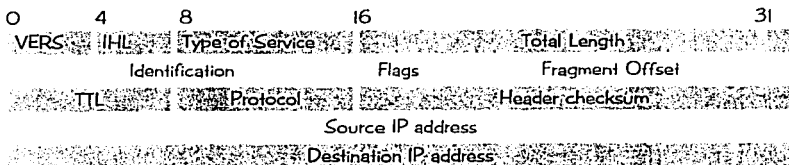


Figura 2.8 Cabecera del segmento de IPv4

En la IPv4, se establecieron tres clases diferentes de direcciones (v. figura 2.9), las cuales se representan mediante tres rangos de valores.

Clase	Primer byte	Identificación de red	Identificación de hosts	Número de redes	Número de hosts
A	1...126	1 byte	3 byte	126	16.387.064
B	128...191	2 byte	2 byte	16.256	64.516
C	192...223	3 byte	1 byte	2.064.512	254

Figura 2.9 Tabla de direcciones IPv4

### 2.2.3 LA NUEVA VERSIÓN IP (IP VERSIÓN 6, IPV6, IPNG)

La nueva versión del protocolo IP recibe el nombre de IPv6, aunque es también conocido comúnmente como IPng (*Internet Protocol Next Generation*). El número de versión de este protocolo es el 6 (que es utilizada en forma mínima) frente a la antigua versión utilizada en forma mayoritaria. Los cambios que se introducen en esta nueva versión son muchos y de gran importancia, aunque la transición desde la versión antigua no debería ser problemática gracias a las características de compatibilidad que se han incluido en el protocolo. IPng se ha diseñado para solucionar todos los problemas que surgen con la versión anterior, y además ofrecer soporte a las nuevas redes de alto rendimiento.

Una de las características más llamativas es el nuevo sistema de direcciones, en el cual se pasa de los 32 a los 128 bits, eliminando todas las restricciones del sistema actual. Otro de los aspectos mejorados es la seguridad, que en la versión anterior constituía uno de los mayores problemas. Además, el nuevo formato de la cabecera se ha organizado de una manera más efectiva, permitiendo que las opciones se sitúen en extensiones separadas de la cabecera principal.

El IPv6 reemplaza la especificación del tipo de servicio del IPv4 con un mecanismo que permite la preasignación de recursos de red. En particular el nuevo mecanismo soporta aplicaciones como video en tiempo real.

Posiblemente el cambio más significativo en el IPv6 es el cambio de un protocolo que especifica completamente todos los detalles a un protocolo que puede permitir características adicionales.



El tamaño de la cabecera (v. figura 2.10) que el protocolo IPv6 añade a los datos es de 320 bits, el doble que en la versión antigua. Sin embargo, esta nueva cabecera se ha simplificado con respecto a la anterior.

Algunos campos se han retirado de la misma, mientras que otros se han convertido en opcionales por medio de las extensiones. De esta manera los ruteadores, no tienen que procesar parte de la información de la cabecera, lo que permite aumentar de rendimiento en la transmisión. El formato completo de la cabecera sin las extensiones es el siguiente:

El campo *Version* (Versión), especifica el número de versión del protocolo IP, que en este caso contendrá el valor 6. Tamaño 4 bits.

El campo *Flow Label* (Etiqueta de Flujo), contiene información que los ruteadores utilizan para asociar un datagrama con una prioridad y un flujo específicos. El campo está subdividido en dos campos:

4 bits      24 bits  
TCLASS    FLOW IDENTIFIER

El campo TCLASS especifica la clase de tráfico para el datagrama. Los valores del 0 al 7 se emplean para especificar la sensibilidad al tiempo del tráfico controlado por flujo; los valores del 8 al 15 se utilizan para especificar una prioridad para tráfico que no es flujo.

El campo de 24 bits restantes contiene el campo FLOW IDENTIFIER (Identificador de Flujo). La fuente selecciona un identificador de flujo cuando establece el flujo.

El campo *Pay Load Length* (Longitud), especifica el número de octetos transportados en un datagrama, excluyendo la cabecera misma. Tamaño 16 bits.

El campo *Next Header* (Siguiete Cabecera), se utiliza para indicar el protocolo al que corresponde la cabecera que se sitúa a continuación de la actual. Tamaño 8 bits.

El campo *Hop Limit* (Limite de Saltos), interpreta el valor como el limite estricto máximo número de saltos que un datagrama puede realizar antes de ser desechado, es un valor que disminuye en una unidad cada vez que el paquete pasa por un nodo. Tamaño 8 bits.

El campo *Source Address* (Dirección Origen), se refiere al número de dirección del host que envía el paquete. Su longitud se ha incrementado en 16 octetos. Tamaño 128 bits.

El campo *Destination Address* (Dirección Destino), número de dirección de destino, aunque puede no coincidir con la dirección del host final en algunos casos. Su longitud se ha incrementado en 16 octetos. Tamaño 128 bits.

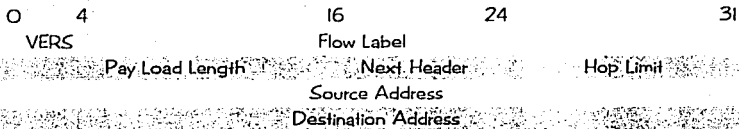


Figura 2.10 Cabecera IP. IPV6

Las extensiones que permite añadir esta versión del protocolo se sitúan inmediatamente después de la cabecera normal, y antes de la cabecera que incluye el protocolo de nivel de transporte. Los datos situados en cabeceras opcionales se procesan sólo cuando el mensaje llega a su destino final, lo que supone una mejora en el rendimiento. Otra ventaja adicional es que el tamaño de la cabecera no está limitado a un valor fijo de bytes como ocurría en la antigua versión.

Por razones de eficiencia, las extensiones de la cabecera siempre tienen un tamaño de múltiplos de 8 bytes. Actualmente se encuentran definidas extensiones para el ruteo de la fuente, fragmentación y ensamblaje, seguridad, confidencialidad de datos, etc.

El sistema de direcciones es uno de los cambios más importantes que afectan a la versión 6 del protocolo IP, donde se han pasado de los 32 a los 128 bits (cuatro veces mayor). Estas nuevas direcciones identifican a una interfaz o conjunto de interfaces y no a un nodo, aunque como cada interfaz pertenece a un nodo, es posible referirse a éstos a través de su interfaz.

El número de direcciones diferentes que pueden utilizarse con 128 bits es enorme. Teóricamente serían 2128 direcciones posibles, siempre que no apliquemos algún formato u organización a estas direcciones. Este número es extremadamente alto, pudiendo llegar a soportar más de 665.000 trillones de direcciones distintas por cada metro cuadrado de la superficie del planeta Tierra. Según diversas fuentes consultadas, estos números una vez organizados de forma práctica y jerárquica quedarían reducidos en el peor de los casos a 1.564 direcciones por cada metro cuadrado, y siendo optimistas se podrían alcanzar entre los tres y cuatro trillones.

Existen tres tipos básicos de direcciones IPng para identificar a una interfaz en concreto o a un grupo de interfaces. Los bits de mayor peso de la dirección IPng son los que permiten distinguir el tipo de dirección, empleándose un número variable de bits para cada caso. Estos tres tipos de direcciones son:

**Direcciones unicast:** Son las direcciones dirigidas a una única interfaz de la red. Las direcciones unicast que se encuentran definidas actualmente están divididas en varios grupos. Dentro de este tipo de direcciones se encuentra también un formato especial que facilita la compatibilidad con las direcciones de la versión 4 del protocolo IP.

**Direcciones anycast:** Identifican a un conjunto de interfaces de la red. El paquete se enviará a una interfaz cualquiera de las que forman parte del conjunto. Estas direcciones son en realidad

direcciones unicast que se encuentran asignadas a varias interfaces. El formato es el mismo que el de las direcciones unicast.

**Direcciones multicast:** Este tipo de direcciones identifica a un conjunto de interfaces de la red, de manera que, el paquete es enviado a cada una de ellas de manera individual.

Las direcciones de broadcast no están implementadas en esta versión del protocolo, debido a que esta misma función puede realizarse ahora mediante el uso de las direcciones multicast.

Podemos decir que, Internet se encuentra en la primera etapa de la fase de transición de una a otra versión. Pero la renovación del principal protocolo no es un hecho caprichoso, y no hay que olvidar que dicha transición durará varios años.

## **2.3 Servicios de Internet**

Podemos decir que Internet es un fenómeno que ha entrado con fuerza en nuestras vidas y su crecimiento en los últimos años está siendo espectacular; una de las razones fundamentales de ello radica en la existencia de la *World Wide Web*, sin embargo, no hay que olvidar que Internet está integrado por otros servicios que, de igual manera, han hecho de esta, la red de redes, tema que tratamos a continuación.

### **2.3.1 SERVICIOS DE RED**

Estos servicios ofrecen la posibilidad de transportar la información entre computadoras y en Internet encontramos los siguientes.

#### **2.3.1.1 SERVICIOS DE ENTREGA DE PAQUETES SIN CONEXIÓN**

Este servicio es el más básico que provee Internet y se dice que es con el menor esfuerzo. Posee las siguientes características:

- ✓ Entrega de paquetes: La información transmitida se divide en unidades más pequeñas llamadas paquetes o datagramas.
- ✓ Sin conexión/no orientado a la conexión/no confiable: No garantiza la entrega de los paquetes a su destino final ni en orden. Pese a lo anterior, los protocolos hacen todo lo posible para que esto se lleve a cabo, por lo que se dice que es con el menor esfuerzo.

El protocolo que ofrece este servicio es el IP (*Internet Protocol*).

#### **2.3.1.2 SERVICIOS DE TRANSPORTE DE FLUJO CONFIABLE**

Este servicio permite que una aplicación que se encuentra en una computadora, establezca una conexión de circuito virtual<sup>30</sup> con una aplicación localizada en otra computadora, con el objeto de enviar datos a través de la conexión de forma tal, que parezca permanente y directa del *hardware*.

---

<sup>30</sup> Es una conexión en la que se determina la ruta lógica (del transmisor al receptor) que va a seguir un mismo flujo de información, es decir, no se requiere elegir una ruta nueva para cada paquete que pertenece a dicho flujo.

Una vez recibido el mensaje, el receptor proporciona un acuse de recibo al transmisor quien procede a enviar el siguiente paquete. El transmisor cuenta con un contador de tiempo<sup>31</sup> que funciona cuando se envía un paquete. Cuando el paquete ha expirado sin recibir acuse de recibo, se procede a retransmitir el paquete.

Aunque el Servicio de Transporte de Flujo Confiable se construye sobre el Servicio de Entrega de Paquetes sin Conexión, a diferencia del segundo, el primero hace posible que la detección y solución de errores de transmisión sean manejadas por un protocolo.

El protocolo que ofrece este servicio es el TCP (*Transport Control Protocol*).

## 2.3.2 SERVICIOS DE APLICACIÓN

Estos servicios se ofrecen a través de la arquitectura cliente/servidor (v. capítulo 3.2.1) y en Internet encontramos los siguientes.

### 2.3.2.1 SESIÓN REMOTA

Una de las piedras angulares de Internet es la sesión remota, a través de la cual, es posible tener un equipo remoto (como su nombre lo indica) en la pantalla local. Evidentemente, no será el equipo en cuestión pero sí una terminal que permitirá operar en el *HostRemoto (Remote-Host)* como si se estuviera físicamente en el mismo, por lo tanto, aunque en modo textual, ofrece todas las posibilidades de que dispone un usuario local con el mismo derecho de acceso. El protocolo TCP/IP que ofrece este servicio es el Telnet, el cual, establece un enlace TCP con la computadora remota.

#### FUNCIONAMIENTO DEL TELNET

Aunque existen diversas implementaciones del Telnet cliente en las computadoras, lo más común es que siga este formato para la dirección<sup>32</sup> del servidor: telnet://milla.dcaa.unam.mx (DNS) ó telnet://132.248.27.10 (dirección IP). Inmediatamente, la computadora a la que hacemos Telnet, nos pedirá un *login* (v. figura 2.11).

De esta manera, cuando se conozca la dirección de una computadora que ofrece un servicio de información Telnet, se debe averiguar cuál es su *login* para poder acceder. Hay otros servidores de Telnet que sólo dan información a los usuarios que tienen una cuenta abierta con ellos.

Para finalizar la conexión remota, se debe teclear la palabra *disconnect* o la suministrada por la propia computadora remota al iniciar la sesión.

---

<sup>31</sup> Proceso que registra hora y fecha en la que se inicio la transmisión.

<sup>32</sup> En Internet, serie de caracteres, numéricos o alfanuméricos, que identifican un determinado recurso de forma única y permiten acceder a él. En la red existen varios tipos de dirección de uso común como la dirección IP.

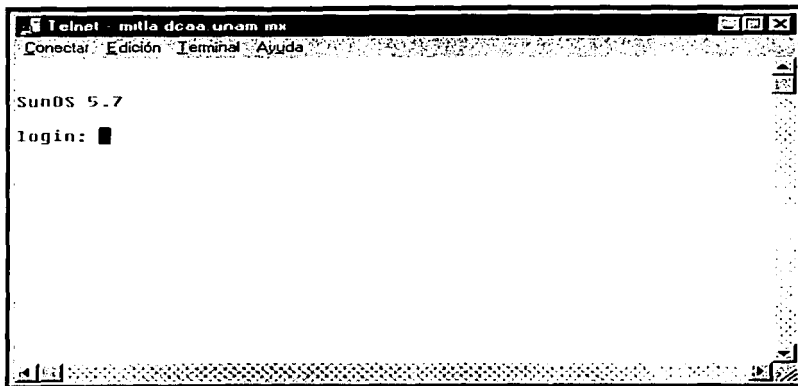


Figura 2.11 Sesión Telnet. Pantalla de bienvenida

### 2.3.2.2 TRANSFERENCIA DE ARCHIVOS (FILE TRANSFER PROTOCOL, FTP)

Desde el principio, Internet se utilizó para el intercambio de mensajes electrónicos pero también se buscó la posibilidad de transmitir por la red amplios documentos y programas. De este modo, en 1971, un grupo de investigación presentó la propuesta en el MIT del primer Protocolo para la transferencia de archivos en Internet.

En los años siguientes, hubo varias propuestas destinadas a corregir determinados aspectos del proyecto original. Pero hasta 1985, el FTP alcanzó la forma vigente actualmente en todo el mundo.

El protocolo está basado en la filosofía cliente/servidor. El servidor, de modo similar al disco duro, permite cargar y guardar archivos.

Un servidor FTP, es aquél que permite intercambiar información entre dos o más computadoras conectadas a la red. Mediante este servicio, los usuarios pueden conectarse al servidor FTP y obtener una copia de la información ahí existe, a su computadora local, o bien, el usuario puede colocar uno o más archivos de su computadora local en el servidor FTP. Ha esto se le conoce como Transferencia de archivos.

El protocolo FTP es uno de los servicios más importantes de Internet dada la actividad tan grande que ocurre con la transferencia de información entre computadoras.

Existen computadoras que proporcionan datos que son públicos y otros son de pago o de uso privado. Aquí encontramos lo que se denomina el servicio de FTP Anónimo; se trata de un servicio público.

## FUNCIONAMIENTO DEL FTP

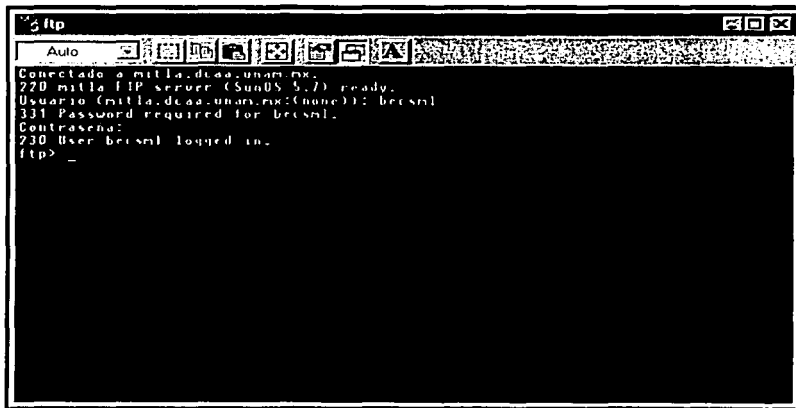
Para acceder a un servidor de archivos, ya sea de uso público o privado, lo primero que tendremos que hacer es conocer su dirección, ya sea en formato IP o DNS. Por ejemplo, si queremos acceder al servidor milta de la Dirección de Sistemas, en la U.N.A.M., debemos teclear:

```
ftp://milla.dcaa.unam.mx ó  
ftp://132.248.27.10
```

En unos momentos aparecerá una pantalla de bienvenida donde tendremos que identificarnos de la siguiente manera:

- ✓ Servidor de archivos públicos: Pide que, como *username/login*<sup>33</sup> tecleemos *anonymous* y, como *password*<sup>34</sup>, es recomendable que dejemos nuestra dirección de correo electrónico. Si es que estamos dentro del servidor FTP Anónimo aparecerá el siguiente mensaje: *Guest login ok, access restrictions apply*
- ✓ Servidor de archivos privados: Deberemos tener un *username/login* y *password* facilitado por el administrador de la red<sup>35</sup>. En este caso, aparecerá un mensaje indicándonos que ya estamos dentro del servidor FTP (v. figura 2.12).

Una vez que hemos accedido al servicio (público o privado) ya podemos iniciar el uso de la información proporcionada



```
Auto
Conectado a milla.dcaa.unam.mx.
220 milla FTP server (SunOS 5.7) ready.
Usuario (milla.dcaa.unam.mx:(none)): brexnl
331 Password required for brexnl.
Contraseña:
230 User brexnl logged in.
ftp> _
```

Figura 2.12 Sesión FTP. Pantalla de bienvenida

<sup>33</sup> Nombre que identifica a un Usuario de un sistema específico.

<sup>34</sup> Clave que identifica a un Usuario de un sistema específico.

<sup>35</sup> Persona cuyo objetivo principal es, mantener la operatividad y buen desempeño de la red para satisfacer las necesidades de los usuarios; para ello se auxilia de distintas herramientas (tanto software como hardware).

### 2.3.2.3 SECURE SHELL (SSH)

Actualmente, las Redes de Telecomunicaciones<sup>36</sup> son los medios digitales más usados en todos los ámbitos de la sociedad para la transferencia de información. Debido a que estos medios se encuentran en redes públicas<sup>37</sup>, están expuestos a intervenciones de una u otra forma.

Cuando se realiza una conexión a un servidor remoto - utilizando alguno de los servicios antes expuestos como Telnet o FTP - el *login* y *password* son transmitidos en la red de forma clara, lo que representa un gran riesgo si llega a existir sobre la red un programa que capture la información como un *sniffer*<sup>38</sup>, ya que, dichos programas pueden obtener tanto el *login* como el *password* y alterar el sistema con dicha información.

Este tipo de problemáticas ha llevado al diseño de herramientas que eviten situaciones como la anterior, tal es el caso del *Secure Shell (ssh)*, desarrollada por Tatu Ylonen en la Universidad Tecnológica de Helsinki en Finlandia.

*Secure Shell*, es un programa que permite realizar conexiones entre computadoras a través de una red para ejecutar comandos en una computadora remota, así como, para mover archivos de una computadora a otra. Proporciona una exhaustiva autenticación y comunicaciones seguras en redes no seguras. Por lo que, nace como un reemplazo a los servicios de Internet (Telnet, FTP) que proporcionan gran flexibilidad en la administración de una red, sin embargo, presenta grandes riesgos en la seguridad de un sistema.

La ventaja más significativa de *ssh* es que no modifica mucho las rutinas. Iniciar una sesión de *ssh*, es similar y sencillo como iniciar una sesión de Telnet (v. figura 2.13).

### TRANSFERENCIA DE ARCHIVOS SCP

*Secure Shell* proporciona una herramienta con sus mismas características de seguridad, que permite realizar transferencia de archivos. Nos referimos a *secure copy (scp)*. El modo como opera *secure copy* es tecleando el comando *scp*. La transferencia de archivos se puede hacer de la computadora local a una remota o viceversa (v. figura 2.14), siempre y cuando se conozcan el *login* y el *password* de la computadora remota.

Existen actualmente dos protocolos desarrollados sobre *ssh*:

---

<sup>36</sup> Red de Telecomunicaciones: Es el conjunto de elementos (hardware y software) que permiten el transporte electrónico de voz, sonidos, datos, textos, imágenes u otras señales de cualquier naturaleza entre dos o más puntos (nodos).

<sup>37</sup> Red Pública de Telecomunicaciones: Red que se explota para prestar servicios públicos o comerciales de telecomunicaciones, es decir, utilizada para soportar servicios privados o públicos de terceros, y será accesible a todos los que la requieran en igualdad de condiciones, precio, disponibilidad de acceso y trato no discriminatorio.

<sup>38</sup> Sniffer (Husmeador): Término inglés usado para referirse al programa que monitorea los paquetes de datos que circulan por una red en busca de una cadena numérica o de caracteres con objeto de conseguir alguna información. Normalmente se usa con fines ilegales.

- ✓ SSH1: Las versiones de ssh cliente/servidor para Unix que soportan este protocolo, pueden ser utilizadas libremente para propósitos no comerciales; generalmente, se utilizan en ambientes académicos.
- ✓ SSH2: Provee licencias más estrictas que SSH1, ya que es de carácter comercial

The screenshot shows a terminal window titled "mitla.dcaa.unam.mx - default - SSH Secure Shell". The terminal displays the following text:

```

SSH Secure Shell 2.2.0 (build 123)
Copyright (C) 2000 SSH Communications Security Corp - http://www.ssh.com/

This copy of SSH Secure Shell is licensed for educational, charity,
and personal recreational/hobby use.
Any commercial use requires a separate license.

Last login: Mon Jun 11 2001 20:41:20 from orion.dgsc.a.unam
Sun Microsystems Inc. SunOS 5.7 Generic October 1998
You have mail.
Sun Microsystems Inc. SunOS 5.7 Generic October 1998
You have mail.
SONNY-->

```

At the bottom of the terminal, the status bar shows: "Connected to mitla.dcaa.unam.mx", "SSH2 - 3des-cbc - hmac-md5 - none", "77x24", and "INUI".

Figura 2.13 Sesión Secure Shell. Pantalla de bienvenida

The screenshot shows a terminal window titled "mitla.dcaa.unam.mx - default - SSH Secure Shell". The terminal displays the following text:

```

SONNY-->ls
CURSO paginas.txt tarea.doc tesis
SONNY-->scp tesis octavio@51mba.dgsc.a.unam.mx:
octavio@51mba.dgsc.a.unam.mx's password:
tesis | 730B | 0.7 KB/s | TOC: 00:00:01 | 100%
SONNY-->

```

At the bottom of the terminal, the status bar shows: "Connected to mitla.dcaa.unam.mx", "SSH2 - 3des-cbc - hmac-md5 - none", "77x24", and "INUI".

Figura 2.14 Sesión Secure Shell. Utilización del comando scp (secure copy)



#### 2.3.2.4 GRUPOS DE INTERÉS (NEWS)

Es un servicio que permite "lanzar" preguntas al mundo, para que sean contestadas por cualquier persona, dando lugar a mesas redondas virtuales con más de 10 000 temas diversos. Las *news* se organizan en grupos de discusión de acuerdo a los temas.

El servicio funciona siguiendo un modelo cliente/servidor. El usuario debe utilizar un programa para actuar como cliente y así establecer sesiones de *news* (para sistemas multiusuario<sup>39</sup> o para equipos personales). Las operaciones habituales son:

- ✓ Suscribirse al grupo de interés.
- ✓ Seleccionar un grupo para consulta.
- ✓ Hojear los titulares de los artículos del grupo.
- ✓ Leer, guardar o imprimir los artículos seleccionados.
- ✓ Enviar una respuesta o un nuevo mensaje al grupo.
- ✓ Contestar por correo electrónico de forma privada, al autor de una *news*.

El nombre genérico de una conferencia de *news* es una serie de nombres separados por puntos. Por ejemplo: soc.cultura.mexico

La primer palabra, en este caso soc, es la que identifica a que grupo o tipo de conferencia pertenece. Las principales son:

- ✓ soc: Asuntos sociales.
- ✓ sci: Ciencia e investigación.
- ✓ talk: Debates.
- ✓ news: News sobre Internet.
- ✓ rec: Hobbies, arte.
- ✓ bio: Biología.
- ✓ comp: Informática y computadoras.
- ✓ alt: Temas controvertidos, sexo, etc.
- ✓ misc: Todos los demás temas.

#### 2.3.2.5 LISTAS DE DISTRIBUCIÓN

Las listas de distribución es otra de las formas para utilizar Internet como foro de discusión. El concepto es muy similar al de las *news* pero la técnica es distinta. Para acceder a las listas de distribución debemos contar únicamente con el correo electrónico, lo que las hace más flexibles que las *news*.

La lista de distribución es una dirección de correo que alberga a un conjunto de direcciones de correo electrónico, proporcionadas por distintos usuarios con el objeto de intercambiar información de interés común sobre algún tema específico.

---

<sup>39</sup> Característica que permite que varios usuarios ejecuten simultáneamente más de un programa en la misma computadora.

La manera más habitual de operar es la siguiente<sup>40</sup>: Un usuario con dirección de correo electrónico manda un mensaje a la dirección de la lista con un Asunto (*subject*<sup>41</sup>) y Cuerpo (*body*<sup>42</sup>) del mensaje especificando sus opciones (*subscribe* para suscribirse o *unsubscribe* para borrarse de la lista) Desde el momento en que un usuario queda suscrito a una lista, recibirá una copia de cualquier mensaje dirigido a la misma y, por lo tanto, la información enviada por él, será recibida por los demás usuarios<sup>43</sup>.

### 2.3.2.6 TRANSMISIÓN DE CHARLAS EN INTERNET (INTERNET RELAY CHAT, IRC)

Este servicio nos da la posibilidad de charlar en tiempo real con personas que estén conectadas a Internet, independientemente del lugar en el que se encuentren. Para ello, nos conectamos mediante Telnet o mediante un programa cliente IRC a una computadora que nos brinde el servicio también llamada servidor IRC. Muchos servidores IRC no son de acceso libre, sino que exigen que seamos usuarios con cuenta en el sistema. Otros, sin embargo, son de acceso gratuito y no controlado, permitiéndonos conversaciones (normalmente informales) con otros usuarios de Internet.

### 2.3.2.7 CORREO ELECTRÓNICO (E-MAIL)

El correo electrónico, llamado habitualmente *e-mail*, es la aplicación más extendida en Internet. Permite al usuario enviar y recibir mensajes escritos a otros usuarios de la red situados en cualquier lugar del mundo siempre que dispongan de una dirección de correo electrónico (*e-mail address*).

Una característica importante del correo electrónico es que no es necesario que el destinatario de un mensaje esté frente a la pantalla en el momento en que otro usuario se lo envía y, tampoco es necesario que su computadora esté conectada a la red o que esté encendida. Los mensajes que le llegan se almacenan en el servidor en el que el destinatario está dado de alta. Después, cuando éste se conecta con el servidor utilizando su programa de correo, le pide al servidor que le muestre los mensajes que tiene almacenados.

### FUNCIONAMIENTO DEL CORREO ELECTRÓNICO

Dirección electrónica. Para hacer uso del correo electrónico es necesario contar con una dirección electrónica (o dirección *e-mail*). La dirección proporciona toda la información necesaria para que pueda enviarse un mensaje hasta cualquier parte del mundo.

Las direcciones *e-mail* se componen siempre de tres partes:

1. Identificador del usuario.

---

<sup>40</sup> La forma de suscribirse a cualquier lista de distribución puede variar algo de la indicada, pero básicamente el concepto es el expuesto. Un usuario puede estar suscrito a más de una lista.

<sup>41</sup> Dentro de un mensaje de correo electrónico, se le llama *subject* a la breve descripción del asunto por el cual se escribe, el usuario debe colocar dentro de los datos que encabezan dicho mensaje.

<sup>42</sup> Dentro de un mensaje correo electrónico, se le llama *body* al contenido redactado por el usuario, de dicho mensaje.

<sup>43</sup> La dirección de correo de la lista para enviar los mensajes, una vez que ya formamos parte de ella, puede coincidir o no con la que se usa para la suscripción.

2. El signo @ (*at*, en inglés).
3. Dominio (*Domain*) o dirección del servidor del usuario (donde tiene abierta su cuenta).

Dirección del servidor POP3<sup>44</sup> (Protocolo de Oficina de Correos Versión 3/*Post Office Protocol Version 3*). Se trata de la dirección del servidor de cuentas de correo, en este servidor, el usuario de correo tiene ubicado su buzón (contiene todos los mensajes que le envíen a su dirección electrónica). Para poder leer el correo se necesitará el *login* y el *password*.

Dirección del servidor SMTP<sup>45</sup> (Protocolo Simple de Transferencia de Correo/*Simple Mail Transfer Protocol*). Es la dirección del servidor de correo saliente. Cuando un usuario envía un mensaje este llega en primer término a su servidor de correo saliente, esta computadora se preocupa de determinar la mejor ruta y enviar el correo al buzón del destinatario.

Netiqueta (*Netiquette*). Como toda forma de comunicación, también Internet tiene sus reglas éticas o código de buena conducta o de buen uso, que se conoce como *Netiquette*.

El uso del estándar SMTP no fue una herramienta especialmente potente, puesto que sólo soportaba 7 bits y texto ASCII<sup>46</sup>; no era posible enviar caracteres acentuados o especiales de un idioma determinado como la ñ en el caso del español y no era posible enviar archivos binarios. Pero se pudo vencer esta limitación mediante procedimientos como el UNICODE<sup>47</sup>.

## MIME

MIME (Extensión Multipropósito de Correo de la Internet, *Multipurpose Internet Mail Extension*) es un conjunto de especificaciones Internet de libre distribución que permiten, tanto el intercambio de texto escrito en lenguajes con diferentes juegos de caracteres, como el intercambio de archivos de diversos formatos entre computadoras y aplicaciones que sigan los estándares de correo Internet.

Hace algún tiempo el correo electrónico de la Internet sólo servía para mensajes cortos. No se podían enviar archivos anexados, como documentos o gráficos. Hasta la aparición de MIME se pudo enviar, además de mensajes electrónicos, archivos binarios (imágenes, sonido, programas ejecutables, etc.) dentro del mensaje de correo electrónico.

En MIME se definen cinco campos de cabecera:

---

<sup>44</sup> POP. Protocolo diseñado para permitir a sistemas de usuario individual, leer correo electrónico almacenado en un servidor. La Versión 3 es la más reciente y la más utilizada (POP3).

<sup>45</sup> Protocolo que se usa para transferir correo electrónico entre computadoras. Es un protocolo de servidor a servidor, de tal manera que para acceder a los mensajes es preciso utilizar otros protocolos.

<sup>46</sup> ASCII (Estándar Americano de Codificación para el Intercambio de Información/*American Standard Code for Information Interchange*). Conjunto de normas de codificación de caracteres en el que cada letra, dígito y símbolo es representado por un número, es de amplia utilización en informática y telecomunicaciones.

<sup>47</sup> Método de codificación para enviar archivos binarios o textos. Se usa para enviar este tipo de archivos por correo a otros usuarios. Para poder convertirlo de nuevo a un archivo binario hace falta decodificarlo. Algunos programas lectores de correo o de news incorporan un decodificador.

- ✓ *MIME-Version*: Indica la versión del MIME.
- ✓ *Content-Type*: Describe como se ha de interpretar el objeto dentro del cuerpo.
- ✓ *Content-Transfer-Encoding*: Describe cómo está codificado el objeto de modo que se puede incluir en el correo de forma confiable.
- ✓ *Content-Description*: Una descripción en texto plano del objeto del cuerpo que es útil cuando el objeto no es legible (por ejemplo, datos de audio).
- ✓ *Content-ID*: Un valor unívoco especificando el contenido de esta parte del mensaje.

Tipos de *Content Type*.

- ✓ *text*: Archivo de texto.
- ✓ *image*: Imágenes estáticas o imágenes generadas en computadora.
- ✓ *audio*: Grabaciones de sonido.
- ✓ *video*: Grabaciones de video que incluyen movimiento.
- ✓ *application*: Datos para un programa.
- ✓ *multipart*: Mensajes múltiples de los que cada uno tiene una codificación y un tipo de contenido diferentes.
- ✓ *message*: Mensaje *e-mail* completos (por ejemplo un memorándum que se está enviando) o una referencia externa a un mensaje (por ejemplo, un servidor FTP y un servidor de archivo).

### 2.3.2.8 SERVICIOS DE BÚSQUEDA DE INFORMACIÓN

#### ARCHIE

Es una herramienta de búsqueda de información en Internet. Se trata de una base de datos de acceso público que nos permite localizar un archivo determinado dentro de la familia de servidores FTP.

*Archie* mantiene un índice actualizado de los archivos que hay en la red. Actualmente, existe un servidor *Archie* en cada uno de los países importantes que existen en la red.

Para acceder a un servidor *Archie* (si no se dispone de la correspondiente aplicación cliente) se debe ejecutar un Telnet a una máquina que sí disponga de este tipo de aplicaciones. Otra forma para utilizar *Archie* es la siguiente:

- ✓ Por medio de una sesión remota (Telnet): Para conectarse a un servidor *Archie* es necesario establecer una sesión remota utilizando el comando Telnet. Como *login* se utiliza la palabra *archie* o en algunos casos *qarchie*, en los casos en que se solicite un *password* simplemente se presiona *enter*.
- ✓ Por correo electrónico: Algunos servidores de *Archie*, no sólo pueden trabajar interactivamente, sino que pueden mandar al usuario por correo, el resultado de la búsqueda. Utilizar el *Archie* por correo electrónico es más recomendable que utilizarlo a través del Telnet, ya que éste no permite realizar otra actividad mientras está haciendo las búsquedas.

## WAIS

WAIS (Servicio de Información de Red Amplia/*Wide Area Information Service*). WAIS es una herramienta de búsqueda en la Internet que trabaja bajo el modelo cliente/servidor.

Como en otros casos, si no disponemos de un cliente WAIS propio deberemos conectarnos mediante Telnet, a una computadora que disponga de dicho cliente.

Cuando interrogamos a un servidor WAIS lo hacemos mediante el uso de palabras clave que queremos buscar. WAIS interroga a la base de datos y nos devuelve los conceptos que satisfagan la interrogación, seguidos de una puntuación descendente de acuerdo al grado de relación con la búsqueda.

### 2.3.2.9 SERVICIO INTEGRAL DE INFORMACIÓN WORLD WIDE WEB

La traducción a nuestro idioma de la *World Wide Web* podría ser Telaraña Amplia y Mundial o Telaraña de Información Mundial. Ninguna otra vía de comunicación surgida de Internet se ha convertido, con tanta fuerza, en sinónimo de este medio como lo ha hecho la *World Wide Web*.

Podemos definir al WWW como una iniciativa global de recuperación de información hipermedia con acceso universal al inmenso conjunto de documentos en Internet. Lo que la *Web* ha hecho es, proveer a los usuarios de las redes de computadoras el acceso a la información a través de un medio uniforme de manera simplificada, en otras palabras, es un sistema de fácil manejo que puede obtener información de cualquier computadora conectada a la red.

El funcionamiento de este sistema está basado en tres conceptos fundamentales:

- ✓ HTTP<sup>48</sup> (Protocolo de Transferencia de Hipertexto/*Hypertext Transfer Protocol*): Mecanismo de entendimiento entre el programa cliente (quien solicita las páginas *Web*) y el programa servidor (quien responde dicha petición).
- ✓ HTML<sup>49</sup> (Lenguaje de Marcado de Hipertexto/*Hypertext Markup Language*): Lenguaje de descripción y formato para las páginas *Web*. Es una especie de *PostScript*<sup>50</sup> de la *Web*, que sólo puede leerse y que ha sido creado en función de las posibilidades específicas de la representación en línea.
- ✓ URL (Identificador Universal de Recursos/*Uniform Resource Locators*): Mecanismo estandarizado para dar nombre a las páginas y elementos de la *Web*, a través de un título y una ruta de acceso unívocos. Son el requisito imprescindible para formular, dentro de las páginas HTML, referencias a cualquier servidor de la *Web* y sus contenidos.

---

<sup>48</sup> Protocolo encargado de realizar la transferencia de información entre los servidores y los navegadores *Web*.

<sup>49</sup> Lenguaje en el que se escriben las páginas a las que se accede a través de navegadores WWW. Admite componentes hipertexto y multimedia.

<sup>50</sup> *Postscript*: Lenguaje de descripción de páginas, considerado hace años como un estándar. Está diseñado especialmente para imprimir documentos con calidad profesional y alta definición en impresoras láser.

Entre estos tres elementos hay una estricta división del trabajo. HTTP es el responsable de transportar los contenidos de la *Web*, HTML se encarga de la representación y los URL de formular las direcciones.

Otros de los conceptos importantes de la *Web* y estrechamente relacionados con lo anterior, son los siguientes:

- ✓ **Hipertexto:** Un documento en hipertexto, frente a otro tipo de formato, tiene la ventaja de poder relacionar partes de su texto a través de enlaces<sup>51</sup> (*links*), con otros documentos completamente diferentes y, muy posiblemente, de otro autor con un solo *click* del ratón de forma instantánea.
- ✓ **Hypermedia/Multimedia:** La hypermedia es un conjunto de hipertexto. Es la posibilidad de enlazar cualquier elemento a otros. Se abre, por tanto, la posibilidad de no sólo manejar textos, sino imágenes, sonidos, video entre otros muchos elementos.
- ✓ **Hypermedia/Multimedia global:** La interconexión de las computadoras permite que la información esté almacenada en computadoras distribuidas a lo largo de todo el mundo y que, un simple enlace, conecte partes del documento tratado con otro en forma remota.
- ✓ **Navegador (*Browser*):** Aplicación para visualizar todo tipo de información almacenada en páginas HTML que se encuentran disponibles en servidores Web. Su objetivo es, leer los documentos que un servidor proporciona, los interpreta y los muestra al usuario, quedando a la espera de tener que establecer nuevas conexiones - con el mismo sistema u otro - para completar la información suministrada.

A través de la *Web* podemos navegar por toda la red, puesto que la mayoría de los documentos HTML mantienen enlaces activos a otros puntos de la red.

## 2.4 Internet2 (I2)

Desde que el sector académico norteamericano comenzó a utilizar Internet en la década de los 70, esta red se convirtió en un gran suceso porque permitió el intercambio de información y facilitó el desenvolvimiento global de diferentes áreas de investigación.

Hoy en día, Internet continúa su imparable ascenso en número de usuarios, funcionalidad y popularidad y, el sector académico no podía ser la excepción, ya que, ha presentado un crecimiento continuo en los últimos 10 años, debido a la permanente demanda de aplicaciones cada vez más sofisticadas en este sector, aún cuando el principal desarrollo en los servicios de Internet de los últimos años haya sido enfocado a las actividades comerciales como el comercio electrónico.

Al constatar esta realidad, la comunidad académica percibió que era necesario retomar el liderazgo en las investigaciones tecnológicas que permitieran un desarrollo de Internet en una nueva etapa. Por lo anterior, en octubre de 1996, 34 universidades en los Estados Unidos deciden

---

<sup>51</sup> Es un indicador (texto o imagen) que sirve como enlace a otro documento HTML.

establecer una red que permita la utilización de aplicaciones avanzadas que no pueden operar en Internet; su nombre: Internet2 (I2) y, de esta manera, forman el Comité General de Trabajo de I2.

En 1997, el Gobierno Estadounidense anunció el proyecto de la Nueva Generación de Internet (*New Generation Internet*, NGI), que incluía a I2, como un reconocimiento de objetivos comunes. El objeto de esta sinergia es, minimizar las duplicidades y maximizar la compatibilidad e interoperabilidad entre las redes resultantes y sus aplicaciones.

Asimismo, y como resultado de todo el movimiento de la comunidad académica para retomar el liderazgo en el ámbito de la Nueva Generación de Internet, se creó el 1º de octubre del mismo año, la Corporación Universitaria para el Desarrollo Avanzado de la Internet (*University Corporation for Advanced Internet Development*, UCAID) - institución no lucrativa - cuyo objetivo es el de orientar el avance y el desarrollo de I2

En sus inicios, esta corporación la integraban tres universidades americanas líderes en el sector de las investigaciones, cuya misión consistía en orientar los estudios y experimentos relativos a las aplicaciones en todas las áreas del conocimiento, incluidas la ingeniería y herramientas para redes electrónicas de alto rendimiento. Actualmente, la UCAID es una organización formada por todas las entidades que participan en el proyecto I2.

### **NO SUSTITUYE A INTERNET**

I2 no es un sustituto de la Internet actual sino un complemento; todos los participantes de I2 seguirán utilizando Internet tradicional para aplicaciones tales como la *World Wide Web*, el Correo Electrónico y los Foros o Grupos de Noticias (*Newsgroups*); mientras que I2 servirá para nuevas aplicaciones.

Inicialmente, I2 estará haciendo uso de algunas redes existentes en los Estados Unidos, tales como la vBNS (red de alta velocidad de la NSF). Posteriormente, se desarrollarán nuevas tecnologías de mayor velocidad y nuevos servicios mediante grupos de trabajo formados en las diferentes universidades involucradas.

La vBNS comenzó a desarrollarse en 1995 en lo que sería la segunda etapa de desarrollo en la misión de la NSF. En sus inicios, este *backbone* sólo interconectaba cinco centros de supercómputo y algunos puntos de presencia de Internet, sin embargo, su estructura se fue expandiendo gradualmente para ayudar a algunas universidades que necesitaban de una conexión más rápida y eficiente.

Como algunas instituciones integrantes de I2 ya se encontraban conectadas al vBNS de la NSF, este *backbone* de alto desempeño se convirtió en el medio idóneo para dar soporte a I2. Actualmente, la NSF ofrece su *backbone* en cooperación con diversas instituciones que participan en el proyecto de la Nueva Generación de Internet.

## LA TOPOLOGÍA DE LA RED

La arquitectura física de la Red Electrónica que da soporte a I2 incluye la implantación del GigaPOPs, es decir, puntos de presencia con velocidades de tráfico del orden de los Gigabits (Gb). La función principal del GigaPOP es administrar el intercambio de tráfico en I2, de acuerdo con las especificaciones de velocidad y calidad de los servicios previamente establecidos a través de la red y, separar el tráfico entre los miembros de I2 y las instituciones localmente conectadas cuyo tráfico debe ser desviado hacia la Internet comercial. Por consecuencia, cada GigaPOP posee como mínimo dos conexiones: una para I2 y otra para la Internet Comercial.

Cada GigaPOP va a concentrar y administrar el tráfico de los datos originados y destinados a un conjunto de universidades y centros de investigación localizados en una misma región geográfica. Los GigaPOPs poseen políticas locales para la aceptación de conexiones que deberán ser negociadas entre las partes involucradas.

El intercambio de datos entre los GigaPOPs se realiza actualmente por la Red vBNS. La vBNS posee restricciones en lo que se refiere al tipo de tránsito transportado, permitiendo su uso apenas para las instituciones académicas que participan de I2.

## IPv6

IPv6 es el nombre formal dado a la próxima generación del protocolo IP (v. capítulo 2.2.3).

## APLICACIONES

No hay una línea de trabajo única y determinada previamente que oriente las investigaciones de las nuevas posibilidades de aplicaciones que están siendo desarrolladas en I2. Aún hay mucho para ser investigado sobre la necesidad de los usuarios y el potencial de las tecnologías para redes de alto desempeño. De una forma general, no se conoce aún el límite de lo que es técnicamente posible. Se puede decir entonces, que el foco principal de I2 reside en el desarrollo de aplicaciones avanzadas (v. capítulo 4.2.5.8) con uso intensivo de tecnologías multimedia en tiempo real.

En este momento, algunas de las principales líneas de investigación desarrolladas para la aplicación de servicios en red de alto desempeño son:

- ✓ Bibliotecas digitales.
- ✓ Laboratorios virtuales.
- ✓ Teleinmersión.
- ✓ Nano manipulador distribuido.
- ✓ Telemedicina.
- ✓ Sistema avanzado de predicción regional.
- ✓ Proyección de pantallas de computadoras en tres dimensiones (*ImmersaDesk*).
- ✓ Control remoto de microscopios electrónicos para investigaciones médicas.

## GARANTÍA DE CALIDAD EN EL SERVICIO (QoS)

Una de las novedades de I2 es el concepto de Calidad de Servicio (*Quality-of-Service*). En la Internet actual, toda la información tiene la misma prioridad de tránsito al pasar de una computadora a



otra, mientras que I2 permitirá que las aplicaciones prioritarias disfruten de un monto determinado de ancho de banda garantizado, dejando del resto para el demás tráfico de la red.

Existen varias dimensiones de la calidad de servicio que se desea garantizar a las aplicaciones en I2, las de mayor prioridad, son las siguientes:

1. Velocidad de Transmisión (*Transmission Speed*): Un usuario puede necesitar una conexión que nunca sea inferior a 50 Mbps y tampoco no alcance tasas superiores a 100 Mbps. Un ejemplo de aplicación son los programas ya disponibles para videoconferencia.
2. Atraso (*Delay*): Aplicado especialmente para video, audio y servicios de tiempo real (tales como telemedicina), el atraso es el tiempo máximo de interrupción aceptable para una señal en la red, para garantizar el flujo continuo de la transferencia de la información.
3. Rendimiento de Procesamiento (*Throughput*): Se refiere a la cantidad de datos transmitidos en una unidad de tiempo. Un usuario puede especificar, por ejemplo, que 1 Terabyte de información debe ser movido en 10 minutos.
4. Agenda (*Schedule*): Un usuario puede establecer que una cierta conectividad debe estar disponible en horarios futuros durante un periodo de tiempo predeterminado.
5. Tasa de Pérdida (*Loss Rate*): Se refiere a la tasa máxima de pérdida de paquetes que puede ser esperada dentro de un intervalo de tiempo.

### **TRANSMISIÓN MULTIPUNTO (MULTICAST)**

Una de las limitaciones de la Internet actual es la siguiente: cuando se quiere transmitir información a un conjunto de usuarios, por ejemplo en la transmisión de un evento en vivo, se mandan los mismos paquetes de la señal de video a cada uno de los usuarios, con lo que, se multiplica el tráfico en la red. Mientras que en I2, se está experimentando con una tecnología conocida como *multicasting*, en la cual se envía una sola vez cada paquete con la información necesaria para que les llegue a todos los usuarios que deben recibirla.

### **RETARDO REDUCIDO Y UNIFORME (LOW LATENCY/LOW JITTER)**

En aplicaciones sensibles al retraso de la información, es vital reducir éste al mínimo posible; en I2 con la combinación de un gran ancho de banda y el otorgar prioridad a los servicios y técnicas avanzadas de enrutamiento, se logran retardos realmente muy pequeños en el orden de los milisegundos. Esto permite desarrollar sistemas de control a distancia de equipos muy sofisticados, en los cuales, demasiado retardo de la información de control entre el equipo y el manipulador remoto puede resultar fatal.

### **MAYOR SEGURIDAD, PRIVACIDAD Y CONFIABILIDAD**

Otro aspecto importante que se está experimentando en I2, consiste en la mejora de la seguridad y privacidad de la red, utilizando protocolos que permitan autenticar plenamente el origen de los datos y que asegure la integridad y confidencialidad de los mismos.

## **I2 EN MÉXICO**

Siguiendo el desarrollo mundial de redes de datos de mayor capacidad y velocidad que utilizan aplicaciones de alta tecnología; el Gobierno Mexicano, la Comunidad Universitaria y la Sociedad

Mexicana en general, tomaron la iniciativa de desarrollar una red de alta velocidad y unirse a la red I2 con el fin de brindar a la Comunidad Científica y Universitaria de México una red de telecomunicaciones que les permita crear una nueva generación de investigadores, dotándolos de mejores herramientas que les permitan desarrollar aplicaciones científicas y educativas de alta tecnología en el ámbito mundial.

Para tal efecto, se han dado pasos que marcan el inicio de este importante avance en nuestro país:

- ✓ 8 de abril de 1999: Se oficializa en Los Pinos la constitución de la Corporación Universitaria para el Desarrollo de Internet (CUDI), con la presencia del Dr. Ernesto Zedillo Ponce de León (presidente de la República 1994-2000), del Lic. Miguel Limón Rojas (secretario de Educación Pública 1994-2000) y del Lic. Carlos Ruiz Sacristán (secretario de Comunicaciones y Transportes 1994-2000).
- ✓ 20 de mayo de 1999: En la ciudad de San Diego, California, representantes de la CUDI firman dos importantes Memorándums de Entendimiento<sup>52</sup> con destacadas corporaciones universitarias que promueven y coordinan la disponibilidad de redes avanzadas para aplicaciones de investigación y educación en la Unión Americana, las cuales, están comprometidas a colaborar conjuntamente con la CUDI en el desarrollo de tecnologías y aplicaciones de la nueva generación de Internet. Estas corporaciones son:

Corporación Universitaria para el Desarrollo Avanzado de Internet  
*University Corporation for Advanced Internet Development, UCAID*

Corporación de las Iniciativas de la Red de la Educación en California  
*Corporation of Education Network Initiatives in California, CENIC*<sup>53</sup>

Red Canadiense para la Investigación Avanzada de la Industria y la Educación  
*Canadian Network for Advanced Research Industry and Education, CANARIE*<sup>54</sup>

- ✓ 20 de mayo de 1999: Se firma un convenio con la empresa de telefonía Telmex, para una participación como Asociado Institucional.

---

<sup>52</sup> La participación de instituciones extranjeras en Internet2 se establece a través de "Memorándums de Entendimiento", conocidos como MoU (Memorandum of Understanding), es decir, documentos que establecen acuerdos de trabajo mutuo con el objeto de fijar metas comunes para los países o redes participantes del Proyecto. Se halla abierta la participación de esas instituciones, así como la de miembros norteamericanos; es decir, cualquier organización con interés en establecer una relación basada en un MoU con Internet2 y con condiciones técnicas para tal, podrá hacerlo. Por lo general, las instituciones interesadas son organizaciones comprometidas en alcanzar metas similares a las del Proyecto I: en sus respectivos países, además de universidades, centros de investigación e instituciones sin fines lucrativos.

<sup>53</sup> Organismo que representa los intereses de la comunidad de investigadores y académicos de nivel superior en el área de California cuyo objetivo es, facilitar y coordinar el desarrollo, despliegue y operación de un conjunto de servicios de comunicación entre campus capaces de soportar aplicaciones de educación e investigación avanzada.

<sup>54</sup> Organización encargada del desarrollo de la red Internet 2 en Canadá.

- ✓ 6 de Octubre de 1999: En la ciudad de Ottawa Canadá, se firma un Memorándum de Entendimiento entre CUDI y CANARIE, para establecer programas de investigación, educación y colaboración entre ambos países.

## **EL CUDI**

A partir de la iniciativa de 7 de las universidades<sup>55</sup> más grandes de México, interesadas en trabajar en proyectos de investigación conjunta (en el ámbito nacional como internacional), surgió la necesidad de integrar y dar coherencia a los esfuerzos que venían realizando cada una de ellas, a través de un organismo que tuviera personalidad jurídica semejante a la de organismos internacionales dedicadas a coordinar los trabajos de I2 en el ámbito internacional.

De esta forma, el 12 de Mayo de 1999 queda constituida la Corporación Universitaria para el Desarrollo de Internet A. C., para representar jurídicamente los intereses de las Universidades e Instituciones que conforman el proyecto de I2 en México.

## **OBJETIVOS**

- ✓ Promover y coordinar el desarrollo de redes de telecomunicaciones y cómputo, enfocadas al desarrollo científico y educativo en México, a través de la coordinación de las labores encaminadas al desarrollo de la red de cómputo avanzado en el país.
- ✓ Brindar asesoría sobre las aplicaciones que utilizarán esta red y fomentar la colaboración entre sus miembros.

En el proyecto I2 pueden participar todas aquellas universidades, institutos de investigación (del sector público, privado o social), personas morales y sociedades mercantiles que estén comprometidas con el desarrollo, evolución y utilización de aplicaciones educativas y de tecnología avanzada de redes de telecomunicaciones y de cómputo.

## **I2 EN LA UNAM**

Las Universidades están a la cabeza en I2, ya que, son la fuente principal de demanda tanto por las tecnologías de intercomunicación como por el talento necesario para ponerlas en práctica. Las misiones de educación e investigación requieren de sobremanera, la colaboración de personal y de equipo situado en los campus a través de todo el país. Estos son exactamente los tipos de tareas que no son posibles con la Internet actual y las que necesitan de tecnologías que la I2 se propone crear.

Al mismo tiempo, la capacidad en conexiones por medio de computadoras en las Universidades miembro es insuperable. Las Universidades tienen una larga historia de desarrollo de redes avanzadas de investigación y de ponerlas en funcionamiento. Esta combinación de necesidades y recursos

---

<sup>55</sup> Instituto Politécnico Nacional (IPN), Instituto Tecnológico de Estudios Superiores de Monterrey (ITESM), Universidad Autónoma de Nuevo León (UANL), Universidad Autónoma Metropolitana (UAM), Universidad de Guadalajara (UG), Universidad de Las Américas (UDLA), Universidad Nacional Autónoma de México (UNAM).

proporciona el marco perfecto para desarrollar la próxima generación de posibilidades de Internet en México.

El papel de la UNAM en el proyecto de I2 ha sido relevante, ya que, la planeación y desarrollo de sus proyectos han sido beneficiados directamente por las nuevas características que proporciona esta nueva red.

Entre los proyectos que se han desarrollado en la UNAM, se encuentran los siguientes:

### ***CENTRO DE ARCHIVO ASTRONÓMICO EN MÉXICO***

Cuyo objetivo es proveer a la comunidad Astronómica en México, un gran acervo de datos (tanto públicos como de uso restringido) con un acceso de alta velocidad, así como, herramientas que permitan realizar procesos de búsqueda y análisis de datos recuperados.

### ***BIBLIOTECA MÉDICA DIGITAL NACIONAL***

Cuyo objetivo es la creación de una Biblioteca Digital utilizando la infraestructura de cómputo, telecomunicaciones y acervos bibliográficos; permitiendo así, brindar una amplia variedad de servicios y productos información para alumnos académicos y profesionales en el área de la Salud en el ámbito nacional.

### **EL FUTURO DE INTERNET2**

Aunque en estos momentos todo el desarrollo que se está efectuando en I2 es de carácter académico y de investigación, la idea es que las tecnologías que de ellas se van derivando, lleguen al Internet actual y se vuelvan de uso común en los servicios que se ofrece a la comunidad.

### 3. WORLD WIDE WEB

A inicios de 1990 surgió un nuevo fenómeno en el mundo de Internet: la *World Wide Web* (Telaraña Amplia y Mundial o Telaraña de información mundial).

#### 3.1 Historia

A finales de los 80 en Suiza, aparece una organización de vital importancia para Internet aún en nuestros días, nos referimos al Centro Europeo para la Investigación Nuclear<sup>56</sup> (*European Center for Nuclear Research*, CERN). Los responsables del CERN, habían creado una estructura jerárquica para organizar todos los grupos existentes dentro de la misma; compartiendo así, información, *software* y *hardware*. La estructura, desde un punto de vista organizacional, cumplía con las expectativas de los administradores pero dejaba pasar una cuestión de vital importancia: La comunicación entre los diferentes grupos.

La solución a este problema fue propuesta por Tim Berners-Lee en marzo de 1989, quien era un físico egresado de la Universidad de Oxford (Inglaterra). Su propuesta, denominada "*Information Management: A Proposal*", planteaba la utilización de la tecnología llamada *HyperText* (Hipertexto), ya ideada en la década de los 50 por Ted Nelson. Dicha tecnología, introducía dos nuevos conceptos enlace y multimedia o hipermedia.

Tim Berners-Lee, sólo se proponía utilizar el concepto de los enlaces, dejando de lado la multimedia, posiblemente por no estar extendidas las computadoras con interfaz gráfica sino las de interfaz de texto. Para poder poner en práctica estas ideas dentro del CERN, se requerían de las siguientes acciones inmediatas dentro de la organización:

- ✓ Permitir el acceso remoto a las computadoras desde sitios externos a las instalaciones del CERN.
- ✓ Acceso heterogéneo, es decir, posible para cualquier sistema operativo de los utilizados por la comunidad científica
- ✓ El sistema de información no centralizado, para permitir enlaces a cualquier punto de interés.
- ✓ Permitir acceso a bases de datos existentes.
- ✓ Permitir la adición de enlaces privados a y desde cualquier tipo de información pública.
- ✓ La mayor estabilidad posible sobre los datos a los que apuntaran los enlaces.

Las conclusiones de la propuesta fueron que el CERN debería avanzar hacia un sistema

---

<sup>56</sup> En esta organización trabajan un gran número de físicos europeos en la investigación de todo tipo de cuestiones sobre física de altas energías. Las miles de personas relacionadas con el CERN se encuentran dispersas por diferentes países donde desarrollan sus investigaciones formando así grupos internacionales.

universal de información enlazada entre sí, en el cual, características tales como la portabilidad<sup>57</sup> deberían predominar más que el aspecto gráfico o cualquier otra característica. La propuesta no fue tomada en cuenta sino hasta el mes de mayo de 1990.

A Tim Berners-Lee se le unió Robert Cailliau, ingeniero del CERN y, en octubre de dicho año, distribuyen una propuesta inicial con algunas modificaciones en la cual, se habla por primera vez de la *World Wide Web*.

El nuevo proyecto, estaba basado en la utilización de los recursos informáticos del CERN y comprendía dos fases:

- ✓ Primera fase: Hacer uso, tanto del *software* como del *hardware* existente, para la implementación de un navegador sencillo de manejar y para usarlo en estaciones de trabajo<sup>58</sup> (*workstation*), basándose en las necesidades de acceso a la información expuestas por los investigadores relacionados con el CERN. Durante los tres meses previstos para esta fase sólo se podría hacer uso de la información ya existente.
- ✓ Segunda fase: Extender el uso de las herramientas a todas las computadoras personales, y permitir que los usuarios introdujesen, en la estructura de datos de sus proyectos, nuevos enlaces.

Para ambas fases, se incorporaron al proyecto cuatro ingenieros de *software* que tendrían que desarrollar los navegadores para las plataformas utilizadas en aquel momento. Estos navegadores aún no se parecían mucho a los que conocemos en la actualidad, ya que, como se comentaba anteriormente, estaban mucho más enfocados al texto que a los gráficos o a las páginas multimedia.

En noviembre se terminó el primer *software* de navegador y servidor de *Web* existente en el mundo, tenía capacidades *wysiwyg*<sup>59</sup> (*What You See Is What You Get*), soportaba búsquedas dentro de un documento, permitía la edición y creación de *links*, la transferencia de archivos (FTP), el acceso a *news* y al HTTP.

En 1991 se consolidó la *Web* dentro del CERN, extendiéndose su uso entre los principales servidores de la organización y comenzó su presentación oficial a los distintos grupos de investigadores. En mayo de ese año, se instaló el primer servidor fuera del CERN, pero no fue sino hasta agosto cuando Berners-Lee anunció públicamente la disponibilidad de los programas a toda la comunidad internacional.

---

<sup>57</sup> Característica de ciertos programas que les permite ser utilizados en distintos sistemas de cómputo sin que precisen modificaciones de importancia

<sup>58</sup> En redes, se refiere a una computadora de escritorio que ejecuta programas de aplicación y sirve como punto de acceso a la red.

<sup>59</sup> *Wysiwyg* (Lo que se ve es lo que se obtiene/*What you see is what you get*): En informática quiere decir que lo que nosotros vemos en la pantalla es lo que vamos a obtener una vez que se imprima (no siempre sucede porque todo depende del software que se utilice).

La WWW ya estaba en el dominio público y comenzó su ascenso superando rápidamente tecnologías como el *Wais* por muchas razones, principalmente, su capacidad multimedia y la interacción con el usuario.

1992 representó el año de la cimentación del nuevo sistema de información, la *Web* se extendió en laboratorios de física en Hamburgo, Amsterdam y Chicago pero, lo más importante, fue el nacimiento de los navegadores 100% gráficos, es decir, orientados a sistemas operativos de ventanas.

En febrero de 1993, Marc Andressen y Eric Bina pertenecientes al Centro Nacional para Aplicaciones de Supercómputo de Illinois (*National Center for Supercomputing Applications, NCSA*), iniciaron el desarrollo de *Mosaic*, un programa completamente orientado a ventanas que incluía todas las facilidades existentes hasta ese momento dentro de los protocolos *Web*. Todo esto como una iniciativa particular, no enmarcada en ninguno de los proyectos en curso dentro del NCSA. En febrero se distribuyó la versión para terminales gráficas Unix y hubo que esperar hasta septiembre para tener versiones en todas las plataformas.

*Mosaic* era un programa sencillo de instalar por cualquier persona (incluso sin grandes conocimientos en cómputo), era robusto y estable, transmitiendo al usuario, la sensación de trabajar con un *software* profesional y, lo que más llamó la atención, fue que permitía trabajar con imágenes en color.

Para finales de 1993, se tenían varios centenares de servidores *Web* (muchos en centros ajenos a la investigación). Además, el tráfico de protocolos HTTP se había multiplicado de marzo a finales de año, 10 veces.

Tras su trabajo en NCSA, Marc Andressen abandonó la institución y fue contratado por *Enterprise Integration Technologies/Terisa* (EIT/Terisa), donde colaboró en el desarrollo de un protocolo que permitiese realizar transacciones seguras a través de la red, es decir, realizar todo tipo de actividades comerciales salvaguardando la integridad de los datos personales transmitidos por los usuarios.

El protocolo resultante fue un derivado del HTTP, el S-HTTP (*Secure-HTTP*). Estando en Terisa, Andressen recibió una oferta de Jim Clark, antiguo director de *Silicon Graphics*, para formar conjuntamente la compañía *Mosaic Communications Corporation* que, meses después, se convirtió en *Netscape Corporation*, cuya misión fue el desarrollo de dos nuevos productos en el mercado, el navegador *Netscape Navigator* y un protocolo que compitiera con el S-HTTP, para transacciones comerciales, el SSL<sup>60</sup> (Capa de Conexiones Seguras/*Security Socket Layer*).

En mayo de 1994 el CERN, ante la necesidad de profesionales interesados en la *Web* de

---

<sup>60</sup> Protocolo que ofrece funciones de seguridad (encryptando la información proporcionada de manera confidencial, ya sea a un proveedor, una base de datos, etc.) a nivel de la capa de transporte para TCP.

organizar una conferencia donde poder intercambiar las experiencias de los últimos años de trabajo, realizó un congreso en Ginebra (Suiza). Durante la conferencia, se puso de manifiesto la incapacidad de una organización como el CERN, para poder liderar el desarrollo de los protocolos y atender las demandas de nuevos servicios, que reclamaba la comunidad de usuarios de Internet.

Como consecuencia, se iniciaron una serie de conversaciones entre el CERN y el NCSA, el resultado fue la creación en Boston, E.U. del Comité Internacional de Conferencia de la *World Wide Web* (*International World Wide Web Conference Commitee, IW3C2*) en agosto de 1994. En noviembre el CERN abandona las iniciativas iniciadas sobre la *Web* cediendo los derechos.

Podemos decir que la *World Wide Web* ha revolucionado la comunicación, al facilitar el acceso a la información a través de la Internet y, por otra parte, también ha sido un factor muy importante para la popularidad de la misma, a pesar de su reciente creación.

## 3.2 Modelos de desarrollo Web

### 3.2.1 ARQUITECTURA CLIENTE/SERVIDOR

La arquitectura cliente/servidor surge como una alternativa de solución a dos aspectos fundamentales. Por un lado, la construcción de gigantescas aplicaciones monolíticas de los *mainframes*<sup>61</sup>, donde toda la funcionalidad de cada sistema se encontraba contenida en una sola aplicación, y por otro, para la creación de sistemas basados en archivos compartidos.

Por lo que, la arquitectura cliente/servidor, es un modelo para el desarrollo de sistemas de información en el que los procesos se dividen en elementos independientes que cooperan entre sí, para intercambiar información, servicios o recursos.

#### 3.2.1.1 CARACTERÍSTICAS

- ✓ El servidor presenta a todos sus clientes una interfaz única y bien definida.
- ✓ El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- ✓ El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.
- ✓ Los cambios en el servidor implican pocos o ningún cambio en el cliente.

Todos los sistemas desarrollados en arquitectura cliente/servidor poseen las siguientes características distintivas de otras formas de *software* distribuido:

- ✓ Servicio: El servidor es un proveedor de servicios; el cliente es un consumidor de servicios.
- ✓ Recursos compartidos: Un servidor puede atender a muchos clientes al mismo tiempo y regular su acceso a recursos compartidos.

---

<sup>61</sup> *Computadora multiusuario concebida para cubrir requisitos de computación de grandes empresas, se caracteriza por ser capaz de controlar muchas terminales y unidades de almacenamiento.*



- ✓ **Protocolos Asimétricos:** La relación entre cliente y servidor es de muchos a uno; los clientes solicitan servicios, mientras que los servidores, esperan las solicitudes pasivamente.
- ✓ **Transparencia de ubicación:** El *software* cliente/servidor siempre oculta a los clientes la ubicación del servidor.
- ✓ **Mezcla e igualdad:** El *software* es independiente del *hardware* o de las plataformas de *software* del sistema operativo; se puede tener las mismas o diferentes plataformas de cliente y servidor, lo que proporciona una estructura imprescindible para integrar ambientes heterogéneos.
- ✓ **Intercambio basado en mensajes:** Los sistemas interactúan a través de un mecanismo de transmisión de mensajes: la entrega de solicitudes y respuestas del servicio.
- ✓ **Encapsulamiento de servicios:** Los servidores pueden ser sustituidos sin afectar a los clientes, siempre y cuando la interfaz para recibir peticiones y ofrecer servicios no cambie.
- ✓ **Facilidad de escalabilidad:** Los sistemas cliente/servidor pueden escalarse horizontal o verticalmente. Es decir, se pueden adicionar o eliminar clientes (con apenas un ligero impacto en el desempeño del sistema); o bien, se pueda cambiar a un servidor más grande o a servidores múltiples.
- ✓ **Integridad:** El código y los datos del servidor se conservan centralizados; esto implica menor costo de mantenimiento y protección de la integridad de los datos compartidos. Además, los clientes mantienen su individualidad e independencia.

### 3.2.1.2 COMPONENTES

Conceptualmente, los componentes de la arquitectura cliente/servidor son el cliente, el servidor y la infraestructura de comunicaciones (*middleware*) (v. figura 3.1).

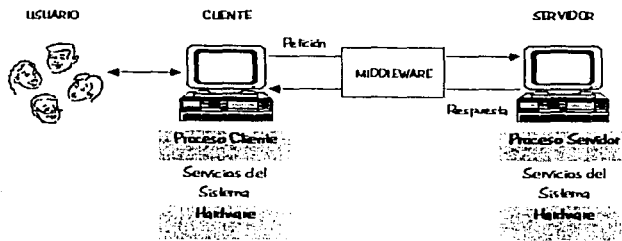


Figura 3.1 Componentes de la arquitectura Cliente/Servidor

#### CLIENTE

Es la entidad por medio de la cual un usuario solicita un servicio, realiza una petición o demanda el uso de recursos. Este elemento se encarga, básicamente, de la presentación de los datos y/o información al usuario en un ambiente gráfico.

Se comunica con procesos auxiliares encargados de establecer conexión con el servidor, enviar

el pedido, recibir la respuesta, manejar las fallas y realizar actividades de sincronización y de seguridad; además, requiere el uso de los recursos de la computadora para cualquier actividad y puede interactuar con uno o varios servidores.

Los clientes se suelen situar en una computadora personal o en estaciones de trabajo, donde permanecen las aplicaciones particulares de cada usuario, y realizan funciones como:

- ✓ Manejo de la interfaz del usuario
- ✓ Captura y validación de los datos de entrada.
- ✓ Generación de consultas e informes sobre las bases de datos.

Como ejemplos de clientes pueden citarse interfaces de usuario para enviar comandos a un servidor, APIs<sup>62</sup> para el desarrollo de aplicaciones distribuidas, herramientas en el cliente para hacer acceso a servidores remotos o aplicaciones que solicitan acceso a servidores para algunos servicios.

## **SERVIDOR**

El servidor es la entidad física que provee un servicio y devuelve resultados; ejecuta el procesamiento de datos, aplicaciones y manejo de la información o recursos.

En algunos casos, existen procesos auxiliares que se encargan de recibir las solicitudes del cliente, verificar la protección, activar un proceso servidor para satisfacer el pedido, recibir su respuesta y enviarla al cliente, además, deben manejar servicios como administración de la red, mensajes, control y administración de la entrada al sistema (*login*), auditoría y contabilidad. Por las razones anteriores, la plataforma computacional asociada con los servidores es más poderosa que la de los clientes. En este caso, se utilizan computadoras personales poderosas, estaciones de trabajo, minicomputadoras o sistemas grandes.

Por su parte los servidores realizan funciones como las siguientes:

- ✓ Gestión de periféricos compartidos.
- ✓ Control de accesos concurrentes a bases de datos compartidos.
- ✓ Enlaces de comunicaciones con otras redes de área local o extensa.
- ✓ Siempre que un cliente requiere un servicio lo solicita al servidor correspondiente y éste, le responde proporcionándolo.

En el servidor permanecen las aplicaciones que deben ser compartidas por varios usuarios. Normalmente, aunque con excepciones, el cliente y el servidor están ubicados en distintos procesadores; incluso, un servidor puede fungir como cliente de otros servidores.

## **SERVIDORES DEDICADOS**

Son aquellos que se clasifican de acuerdo a su funcionalidad y se les llama dedicados, debido a que administran el uso de algún recurso en particular, por ejemplo:

---

<sup>62</sup> API (Interfaz para Programas de Aplicación/Application Program Interface): Conjunto de convenciones de programación que definen cómo se invoca un servicio desde un programa.

**Servidor de Archivos:** El cliente envía solicitudes de registros de archivos al servidor; es un simple servicio de datos compartidos por medio de la red. Son útiles para almacenar archivos (documentos, imágenes, planos, etc.) y aplicaciones de productividad (procesadores de texto, hojas de cálculo).

**Servidor de Bases de Datos:** El cliente envía solicitudes SQL<sup>63</sup> en calidad de mensajes (un mensaje por instrucción); el servidor hace uso de su propia capacidad de procesamiento para encontrar los datos solicitados y devolverlos por medio de la red, sin enviar todos los registros. Este tipo de servidores permiten hacer consultas específicas y obtener reportes flexibles.

**Servidor de Transacciones:** El cliente activa procedimientos remotos que residen en el servidor con un mecanismo de bases de datos de SQL. Es decir, el intercambio por la red consiste en un solo mensaje de solicitud/respuesta que ejecuta un grupo de instrucciones SQL (llamadas transacciones) en el servidor; al crear la aplicación cliente/servidor, se genera código tanto para el cliente como para el servidor. A estas aplicaciones se les llama Procesamiento de Transacciones en Línea (*Online Transaction Processing, OLTP*), las cuales requieren de firmes controles sobre la seguridad e integridad de la base de datos.

**Servidor de Groupware:** Este tipo de servidor es utilizado para el seguimiento de operaciones dentro de la red. El *groupware* dirige la administración de información semiestructurada como texto, imagen, correo, tableros de avisos y flujo de trabajo; estableciendo un contacto directo entre personas. El *software* de *groupware*, generalmente, se trata de aplicaciones creadas en un lenguaje de *scripts*<sup>64</sup> e interfaces basadas en formas.

**Servidor de Objetos:** La aplicación cliente/servidor se genera como un conjunto de objetos de comunicación: Los objetos del cliente se comunican con los objetos del servidor mediante un Corredor de Solicitudes de Objetos (*Object Request Broker, ORB*). El cliente invoca un método de un objeto remoto. El ORB localiza una instancia de esa clase de servidor de objetos, invoca el método solicitado y envía los resultados al objeto cliente. Los objetos del servidor deben ofrecer soporte de concurrencia y participación; el ORB se encarga de reunir todos los elementos.

**Servidor de Web:** Se usan como una forma inteligente para comunicación entre empresas a través de Internet. Este servidor permite transacciones con el acondicionamiento de un navegador específico. Este modelo está integrado por clientes compactos y portátiles en comunicación con servidores amplios. Tal comunicación se da mediante un el protocolo HTTP, el cual define un conjunto simple de órdenes, los parámetros se transmiten en cadenas sin estipulaciones de datos teclados.

---

<sup>63</sup> *SQL (Lenguaje Estructurado de Consulta/Structured Query Language): Es un estándar en el lenguaje de acceso a bases de datos adoptado por ISO. Originalmente, era un lenguaje de acceso al sistema de gestión de bases de datos denominado DB2.*

<sup>64</sup> *Script: Conjunto de caracteres formado por mandatos y secuencias de tecléo que se utiliza muy a menudo en Internet para automatizar tareas muy habituales como la conexión a la red (login).*

## MIDDLEWARE

Para que los clientes y servidores puedan comunicarse se requiere de una infraestructura lógica que proporcione los mecanismos básicos de direccionamiento y transporte. A dicha infraestructura se le denomina *middleware*. El *middleware* abarca a todo el *software* distribuido necesario para el soporte de interacciones entre clientes y servidores.

El *middleware* es un módulo intermedio que no pertenece a los dominios del servidor, ni a la interfaz de usuario, ni a la lógica de la aplicación en los dominios del cliente; tampoco debe confundirse con la red física en sí (cableado, señales de radio o infrarrojas), sino que se trata de una interfaz lógica estándar de los servicios de red.

### FUNCIONES

- ✓ Independizar las dos entidades: El cliente y el servidor no necesitan saber intercomunicarse, sino cómo comunicarse con el módulo de *middleware*.
- ✓ Traducir la información de una aplicación y pasarla a la otra: Acepta consultas y datos recuperándolos de la aplicación cliente, los transmite y envía la respuesta de regreso. También genera los códigos de error.
- ✓ Controlar las comunicaciones: Da a la red características adecuadas de desempeño, confiabilidad, transparencia y administración.

El *middleware* es una herramienta adecuada que no sólo es flexible y segura, sino que también protege la inversión en tecnología y permite manejar diferentes ambientes de computación.

### MIDDLEWARE GENERAL

Es un extracto de la mayoría de las interacciones de cliente/servidor, por otro lado, incluye también las extensiones del sistema operativo de redes.

### MIDDLEWARE DE SERVICIOS ESPECÍFICOS

Es necesario para cumplir un tipo particular de servicio de cliente/servidor, de esta forma, existe un *middleware* específico para los servidores dedicados, un *middleware* para bases de datos, *middleware* para *groupware*, etc.

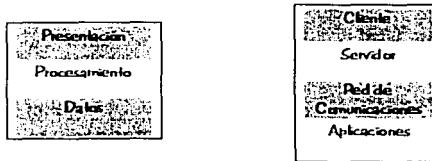
### 3.2.1.3 LA ESTRUCTURA DE UNA APLICACIÓN Y SU RELACIÓN CON LA ARQUITECTURA CLIENTE/SERVIDOR

Toda aplicación de *software* tiene tres funciones fundamentales: administración de los datos, lógica de la aplicación (procesos) y lógica de la presentación (interfaz de usuario). Así, los procesos se efectúan mediante el uso de los dispositivos que forman parte del *hardware*; a su vez, los datos y programas que constituyen parte del *software*, interactúan entre sí, realizando las funciones lógicas necesarias para correr una aplicación, misma que genera un despliegue de información (presentación) para el usuario.

La relación entre las funciones de una aplicación y la arquitectura cliente/servidor es tal, que los procesos, datos y presentación se ejecutan compartiendo recursos del sistema en red. Esta relación

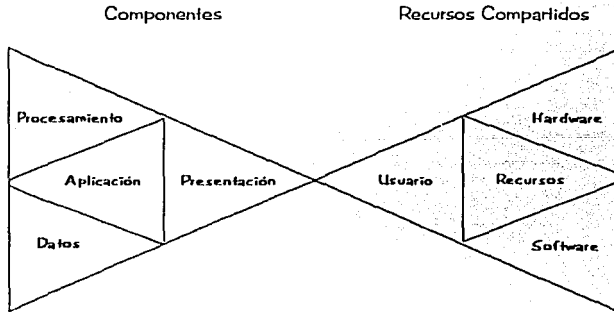
está presente tanto en las PC's como en los sistemas grandes y pretende lograr la integración de ambas plataformas, combinando lo mejor de cada una dentro de un mismo sistema.

Por esta razón, existe un conjunto de variantes de la arquitectura cliente/servidor, dependiendo de dónde se ejecuten las diversas funciones de una aplicación (qué asume el cliente y qué el servidor), como se aprecia en la siguiente figura:



### CLIENTE/SERVIDOR OVERVIEW

Conjuntando ambas estructuras y adicionando los recursos que serán compartidos, la relación entre componentes para la Arquitectura cliente/servidor se visualiza como lo vemos en la siguiente figura:



Donde todos los componentes están estrechamente vinculados de tal forma que, los procesos, datos y presentación de cualquier tipo de aplicación se ejecutan compartiendo los recursos del sistema en red.

### PRESENTACIÓN DISTRIBUIDA

En este modelo, se distribuye la presentación entre el cliente y el servidor pero, éste último, ejecuta todos los procesos y almacena la totalidad de los datos. Es similar a la arquitectura tradicional de un *host* y terminales; el cliente se utiliza sólo para mejorar la presentación desde un punto de vista cosmético. Tiene un bajo costo de desarrollo, pero dificulta el mantenimiento del sistema y no

se aprovecha la red.

### **PRESENTACIÓN REMOTA**

Aquí, la interfaz del usuario está completamente en el cliente, la presentación soporta la captura de datos, incluyendo una validación parcial de los mismos y una presentación de las consultas. La lógica de la aplicación y los datos está en el servidor. Con este modelo, el cliente aprovecha bien la presentación y la red; sin embargo, los procesos de la aplicación pueden resultar complejos en cuanto a su desarrollo y si existe un alto tráfico en la red, puede ser difícil la operación de aplicaciones muy pesadas.

### **PROCESO DISTRIBUIDO**

Se da cuando la presentación está en el cliente, la base de datos está en el servidor y la lógica de la aplicación está distribuida entre el cliente y el servidor. Este modelo permite distribuir los programas del sistema al componente más apropiado, aunque es difícil de diseñar, ya que el diseñador debe definir los servicios y las interfaces de manera que los papeles de cliente y servidor sean intercambiables (excepto el control de los datos, que es responsabilidad exclusiva del servidor).

### **GESTIÓN DE DATOS REMOTA**

Para este modelo, tanto la presentación como los procesos de la aplicación residen en el cliente, mientras que las bases de datos permanecen centralizadas en el servidor. Es fácil de desarrollar, ya que, la lógica de la aplicación no está distribuida y, además, libera la carga del servidor; pero, como desventaja, la totalidad de los datos viajan a través de la red ya que no hay procesamiento en el mismo.

### **BASES DE DATOS DISTRIBUIDAS**

En este último modelo, la presentación, los procesos de la aplicación y parte de los datos de la base de datos están en el cliente, el resto de los datos están en el servidor. Esto permite que la ubicación de los datos sea transparente para la aplicación y que se pueda acceder a datos almacenados en ambientes heterogéneos, aunque éste acceso es dependiente del proveedor del *software* administrador de la base de datos, el cual, divide sus componentes entre el cliente y el servidor.

La anterior clasificación de la arquitectura cliente/servidor, nos lleva a distribuir las partes de una aplicación en uno de los dos extremos, definiendo dos modelos básicos:

### **SERVIDORES AMPLIOS**

Cuando la parte preponderante de la aplicación está en el servidor (presentación distribuida); son más fáciles de administrar y en ellos se pretende reducir al mínimo los intercambios en red mediante la creación de niveles de servicio más abstractos. En este modelo, el cliente aporta la interfaz gráfica de usuario, e interactúa con el servidor a través de llamadas a procedimientos remotos.

## CLIENTES GRANDES

Cuando la mayor parte de la aplicación corre en el cliente (gestión de datos remota y bases de datos distribuidas); aquí, se aprovecha la potencia de procesamiento del cliente y se minimiza el costo del procesamiento en el servidor. Este modelo se emplea en *software* de apoyo de decisiones y personal; ofrece flexibilidad y oportunidades para crear herramientas frontales que les permitan a los usuarios finales generar sus propias aplicaciones.

Cada modelo de cliente/servidor tiene sus propios usos; en muchos casos, los modelos se complementan entre sí. Los servidores de *groupware*, de transacciones y servidores *Web* son ejemplos de servidores amplios; los servidores de bases de datos y archivos son ejemplos de clientes grandes. Los objetos distribuidos pueden ser de ambas clases.

### 3.2.1.4 MODELOS DE LA ARQUITECTURA CLIENTE/SERVIDOR

Dado que la percepción sobre la Arquitectura cliente/servidor es algo abstracto algunos autores la definen en base a modelos de dos y tres niveles, de acuerdo al número de estratos respectivos para representar a sus componentes:

#### SISTEMA CLIENTE/SERVIDOR DE DOS NIVELES (TWO THERIED)

Se considera como modelo cliente/servidor de dos niveles o capas a la estructura más simple, cuyos componentes son:

- ✓ Cliente: Por medio de la interfaz con el usuario, vía una petición, se solicita un servicio, el uso de un recurso, o bien el acceso a un conjunto de datos.
- ✓ Servidores: Satisfacen la solicitud del usuario recibiendo la petición, direccionándola y enviando la respuesta al cliente, ya sea la consulta respectiva de datos, ejecutando el proceso requerido o permitiendo el acceso y/o uso del recurso. En este modelo se acostumbra instalar las bases de datos dentro del servidor, por las ventajas de almacenamiento y velocidad que ofrece en comparación con las del cliente.

#### SISTEMA CLIENTE/SERVIDOR DE TRES NIVELES (THREE THERIED)

El objetivo de este modelo es dividir las funciones de una aplicación en tres componentes:

- ✓ Presentación: Este componente se encarga de la interacción hombre-máquina a través del monitor, teclado, ratón, o bien, mediante algún otro medio como reconocedores de voz.
- ✓ Servidores: Compuesto por varios servidores o componentes de *software* localizados en una o más plataformas, que se encargará de conectar los sistemas existentes.
- ✓ Información: En este componente se incluye la información en sí, los sistemas y aplicaciones existentes

### 3.2.1.5 TECNOLOGÍAS DE DESARROLLO

La tecnología cliente/servidor no es sólo una tecnología sino muchas. Las tecnologías que compiten por la creación de aplicaciones cliente/servidor, han reinventado la computación, no solo en cuanto a bases de datos, sino también, con los objetos distribuidos y la *Web*. El paradigma cliente/servidor presenta cinco tecnologías competitivas, las cuales se describen brevemente a continuación.

## **SERVIDORES DE BASES DE DATOS**

Los servidores de bases de datos de SQL son el modelo dominante para la creación de aplicaciones cliente/servidor. Como se sabe, SQL es un poderoso lenguaje orientado a conjuntos para bases de datos que siguen el modelo relacional y, se utiliza, para la manipulación, definición y control de datos.

En una arquitectura cliente/servidor centralizada en base de datos, por lo general, una aplicación cliente solicita datos y servicios relacionados con datos a un servidor de bases de datos. El servidor de bases de datos, responde a las solicitudes del cliente y ofrece acceso protegido a datos compartidos. Con una sola instrucción SQL, una aplicación puede recuperar y modificar un conjunto de registros de la base de datos del servidor.

### ***CARACTERÍSTICAS DE UN SERVIDOR DE SQL***

- ✓ Administra el control y ejecución de comandos de SQL.
- ✓ Proporciona las vistas lógica y física de los datos.
- ✓ Genera planes de acceso optimizado para la ejecución de comandos SQL.
- ✓ La mayoría ofrece características y utilerías de administración del servidor que facilitan el manejo de datos.
- ✓ Mantiene tablas dinámicas de catálogos que contienen información sobre los objetos SQL alojados en él.
- ✓ Permite que múltiples aplicaciones accedan a la misma base de datos al mismo tiempo.
- ✓ Administra los aspectos de recuperación, concurrencia, seguridad y consistencia de una base de datos: anula los efectos de una transacción en caso de falla; obtiene y entrega candados durante la ejecución de una transacción y protege los objetos de la base de datos contra acceso no autorizado.

Existen tres tipos de arquitectura empleada para tratar bases de datos remotas y sus servidores.

### ***ARQUITECTURA DE PROCESO POR CLIENTE***

A cada cliente de la base de datos, se le otorga su propio espacio de direccionamiento del proceso; la base de datos ejecuta uno o más procesos independientes en segundo plano. Esto protege a los usuarios entre sí y, al mismo tiempo, protege al administrador de la base de datos, de los usuarios.

### ***ARQUITECTURA MULTIHILOS***

Ejecutan todas las conexiones del usuario, las aplicaciones y la base de datos en el mismo espacio de direccionamiento, brindando el mejor desempeño. Esta arquitectura cuenta con su propio creador de itinerarios interno y no depende de esquemas de tareas y protección de direcciones del sistema operativo local. La ventaja es que conserva memoria y ciclos de CPU al no requerir de conmutaciones de contexto frecuentes. Además, las implementaciones del servidor, tienden a ser más exportables entre plataformas porque no requieren de tantos servicios del sistema operativo local.



## **ARQUITECTURAS HÍBRIDAS**

Poseen tres componentes:

1. Un programa de red multihilos que participa en la conexión inicial asignando al cliente a un proceso planificador.
2. Una base compartida de planificadores que colocan los mensajes en una cola de solicitud/respuesta interna, posteriormente, se saca la respuesta de ésta y se envía al cliente.
3. Una base compartida de procesos del servidor que extraen el trabajo de la cola, lo ejecutan y regresan las respuestas a la misma.

Su ventaja es que proporciona un entorno protegido para la ejecución de las tareas del usuario, sin asignar a cada uno de éstos un proceso permanente.

### **EXTENSIONES DE PROCEDIMIENTOS INTEGRADAS**

Se trata de procedimientos de todo tipo que, en lugar de mantener la independencia de datos y código, han insertado el código a la base de datos; esto, debido a que las bases de datos relacionales disponen ahora de extensiones muy útiles pero ajenas al estándar de SQL. Las extensiones mencionadas son tres y estos son sus servicios:

#### **PROCEDIMIENTOS ALMACENADOS (TP LIGERO)**

Son conjuntos nombrados de instrucciones y lógica de procedimientos de SQL compilado, verificado y almacenado (como cualquier otro objeto) en la base de datos del servidor.

El cliente invoca un procedimiento remoto y le transmite los parámetros requeridos a un procedimiento almacenado para hacer un trabajo. Es decir, un solo mensaje remoto desencadena la ejecución de un conjunto de SQL almacenadas; esto, provoca una reducción de tráfico en la red y mejor desempeño.

Este concepto, ofrece un tipo de servicio de base de datos orientado a objetos apto para aplicaciones OLTP y brinda, además, una mayor autonomía local (mediante la ejecución local de programas se modifican las tablas remotamente y no es necesario recompilar todas las aplicaciones remotas).

#### **DESENCADENANTES Y REGLAS**

Los desencadenantes, son acciones especiales definidas por el usuario que son automáticamente invocadas por el servidor, con base en eventos relacionados con los datos. Pueden realizar acciones complejas y emplear toda la capacidad de un lenguaje de procedimientos.

Una regla es un tipo especial de desencadenante que sirve para desempeñar verificaciones simples de datos.

Tanto desencadenantes como reglas, están unidos a operaciones específicas de tablas específicas. Cuando un evento ocurre, es porque algo cambió en la base de datos; los

desencadenantes y reglas manipulan esos eventos para adoptar la acción adecuada en respuesta a cada uno de ellos.

A diferencia de los procedimientos almacenados que son llamados explícitamente por aplicaciones cliente, los desencadenantes, son llamados implícitamente por eventos generados por la base de datos. Aunque, también son ajenos al estándar y específicos del proveedor.

### **MIDDLEWARE ESPECÍFICO PARA BASES DE DATOS**

El hecho de que existan múltiples proveedores de servidores de bases de datos, genera la necesidad de tener un *middleware* específico, para que un cliente de bases de datos pueda acceder a un servidor. No se puede emplear SQL directamente debido a que las extensiones de SQL, los protocolos de manejo de mensajes en red y las API's, son específicas del proveedor.

El *middleware* necesario para lograr que clientes y servidores de SQL, trabajen en redes de bases de datos heterogéneas de proveedores múltiples, presenta dos opciones: Proveedor único o proveedores múltiples.

Una solución común de *middleware* de un solo proveedor puede satisfacer todas las necesidades de datos compartidos (para proyectos pequeños); sin embargo, cuando éste no es el caso, se llega a una serie de inconsistencias obvias por las diferentes herramientas de cada proveedor. Las soluciones de *middleware* de proveedores múltiples - para empresas con grandes sistemas de apoyos de decisiones - crean la idea de una base de datos única y se concentran en el acceso de SQL simple, una conexión a la vez (instrucción por instrucción).

Los elementos importantes del *middleware* de SQL son:

- ✓ La API de SQL del cliente: Es la parte del cliente por medio de la cual, se invoca un servicio; comprende la transmisión de la solicitud por la red y la respuesta resultante, pero no incluye el *software* que presta el servicio como tal.  
Las diferentes API's de SQL no son aplicaciones comunes porque cada una tiene sus propias extensiones de SQL, lo cual, genera complicaciones cuando se utilizan clientes de diferentes proveedores.
- ✓ El controlador de SQL: Es un pequeño elemento de tiempo de ejecución del cliente que acepta las llamadas de la API, da formato a un mensaje de SQL y maneja los intercambios con el servidor. El formato del mensaje y el manejo de enlace se conocen como formatos y protocolos (*Formats and Protocols*, FAP), los cuales, generalmente son definidos por el proveedor.
- ✓ Soporte de FAP para pilas de protocolos: La mayoría de los proveedores soportan múltiples pilas de protocolos; algunas están integradas a los controladores, otras soportan una interfaz de transporte común (por ejemplo los *sockets*<sup>65</sup>), donde es necesario aportar las pilas. En cuanto al servidor, el proveedor proporciona "escuchadores" para sus propias

---

<sup>65</sup> Es un punto final de un enlace de comunicación de dos vías entre dos programas que se ejecutan, a través, de la red.

pilas. Algunos ofrecen sus propios *gateways* de protocolos internos. Los protocolos de BD de diferentes proveedores comparten la LAN, pero no pueden comunicarse entre sí.

- ✓ *Gateways* a BD de otros proveedores: Algunos proveedores ofrecen *gateways* que hacen que las bases de datos de otros proveedores, se vuelvan semejantes a las suyas; es decir, solo brindan una intersección de las características soportadas por las dos BD (generalmente SQL dinámico). Son adecuados para extractos y consultas de datos simples pero no para procesamiento de transacciones.
- ✓ Herramientas de administración de bases de datos: La mayoría de los proveedores hacen posible que se maneje y administre la base de datos desde una estación de trabajo remota a través de una GUI<sup>65</sup>. De este modo, se cuenta con un solo punto de administración del *middleware*. Los clientes y los servidores (siempre y cuando sean del mismo proveedor). Si se utilizan múltiples herramientas de administración, los administradores de las BD deben familiarizarse con una amplia variedad de estaciones de trabajo de administración, cada una con su propia semántica e interfaces de usuario.
- ✓ Herramientas frontales de desarrollo de aplicaciones gráficas y consultas: Estas permiten crear interfaces visuales con el servidor de bases de datos. Cada proveedor ofrece herramientas de GUI para sus propios servidores y para sus plataformas de servidor favoritas.

## OBJETOS DISTRIBUIDOS

Objetos: Son una impresionante combinación de datos y funciones, con propiedades como polimorfismo<sup>67</sup>, herencia<sup>68</sup> y encapsulamiento<sup>69</sup>.

Distribuidos: Son objetos que participan en relaciones cliente/servidor con otros objetos.

Objetos Distribuidos. La tecnología de los objetos modifica radicalmente el estilo de desarrollo de sistemas de *software*. Cualquiera de los objetos, puede ser modificado o reemplazado sin afectar al resto de los componentes en el sistema ni su modo de interactuar. Son aptos para la creación de sistemas flexibles de cliente/servidor, porque los datos y lógica de negocios se encapsulan dentro de los objetos, permitiéndoles así, ubicarse en cualquier punto dentro de un sistema distribuido. Cada

---

<sup>65</sup> GUI (Interfaz Gráfica de Usuario/Graphical User Interface): Sistema de interacción entre el ordenador y el usuario, caracterizado por la utilización de iconos y elementos gráficos en su concepción. Es un paso más allá de las interfaces basadas en caracteres, que solo incluían líneas de texto para introducir comandos y conocer las respuestas del sistema.

<sup>67</sup> Polimorfismo Término usado en programación para referirse a la habilidad de un objeto para adoptar diferentes formas. Esto se refiere al hecho de que múltiples objetos puedan poseer interfaces idénticas.

<sup>68</sup> Herencia Término utilizado en programación que consiste en la habilidad para poder escribir una clase, la cual, hereda variables y métodos de otras clases. Gracias a esta capacidad, se pueden crear clases que incorporen nuevas características respecto a las clases ya existentes.

<sup>69</sup> Encapsulamiento: Nombre que se da en programación al proceso de agrupar información abstraída (verabstraer) de objetos con las operaciones (métodos) que un programa debe realizar con dicha información. En redes, es el proceso de tomar paquetes de un sistema de red y colocarlos dentro de otro sistema de red.

objeto receptor reaccionará de distinta manera al mensaje, con base en su tipo de objeto.

Los objetos distribuidos no bastan por sí solos, se les debe empaquetar en series de componentes capaces de operar juntos. En estas series se combina lo mejor de la tecnología cliente/servidor y de la tecnología de objetos distribuidos.

### **MONITORES DE TP**

Es un sistema operativo para el procesamiento de transacciones que conecta, en tiempo real, a miles de personas con un fondo de procesos compartidos del servidor. Se consideran los más potentes y flexibles administradores de transacciones. Bajo el control de un monitor de TP, una transacción puede manejarse desde su punto de origen (cliente) y a lo largo de uno o más servidores, para volver después al cliente originario.

Los monitores de TP de cada ubicación deben, administrar la parte que les corresponde de las transacciones. Dentro de una ubicación, el monitor de TP coordina las transacciones con los subsistemas y administradores de recursos locales, todo esto, se lleva a cabo con el empleo de un protocolo de grabación en dos fases que coordina la grabación o el aborto de la transacción, a través, de sitios múltiples.

### **PROTOCOLO DE TRANSACCIÓN EN DOS FASES**

Sirve para sincronizar actualizaciones en diferentes computadoras, a fin de que todas resulten un fracaso o un éxito.

El monitor de TP supervisa todos los aspectos de una transacción distribuida y administra recursos de un solo servidor o servidores múltiples y la cooperación en otros monitores de TP.

El monitor de TP es excelente para dos cosas:

- ✓ La administración de procesos: Incluye poner en marcha los procesos del servidor, canalizar trabajo en dirección a ellos, vigilar su ejecución y equilibrar sus cargas de trabajo.
- ✓ La administración de transacciones: Garantiza las propiedades ACID para todos los programas que operen bajo su protección.

### **WORLD WIDE WEB**

La simplicidad de la *Web* permite que el modelo cliente/servidor haga uso de protocolos exportables sobre TCP/IP.

### **GROUPWARE**

El *groupware* de cliente/servidor es un conjunto de tecnologías que nos permiten representar procesos complejos centrados en actividades humanas de colaboración. Se apoya en cinco tecnologías fundamentales:

1. Administración de documentos con multimedia.
2. Flujo de trabajo.
3. Correo electrónico.

4. Conferencias.
5. Planificación.

### **SEGURIDAD EN EL WEB**

La seguridad es un factor decisivo en el establecimiento y aceptación de aplicaciones comerciales de la *Web*; es una cuestión bilateral que involucra tanto al cliente (navegador) como al servidor.

La tecnología de seguridad en el *Web* se concentra en la resolución de cuatro problemas:

- ✓ Codificación.
- ✓ Autenticación.
- ✓ Muros de Protección (firewalls).
- ✓ Sin repudio.

El *Web* soporta dos protocolos de seguridad: la capa de sockets de seguridad (*Secure Sockets Layer, SSL*) de Netscape y el HTTP protegido (*S-HTTP: secure HTTP*). Ambos soportan codificación de llave pública para la codificación de datos, autentifican a usuarios y ofrecen procedimientos de no repudio de firmas electrónicas.

## 4. SISTEMAS DE INFORMACIÓN ADMINISTRATIVOS

En este capítulo se estudia la naturaleza y los aspectos relevantes de los sistemas en una clase particular: Los Sistemas de Información Administrativos. De esta manera, se llega a un tema que es crucial tratar, dado el objetivo de este trabajo: Los sistemas de Información Orientados a la *Web*.

### 4.1 CONCEPTOS BÁSICOS

Para comprender que es un sistema de información administrativo, definiremos de forma elemental el concepto de sistema.

**Sistema:** Conjunto de elementos interrelacionados que llevan a cabo una o más funciones con el propósito de lograr un objetivo común.

Es decir:

1. Un conjunto de elementos.

Un sistema tiene más de un elemento. Por ejemplo, un timón no es un sistema, pero es una parte vital de un sistema que se llama barco.

2. Elementos interrelacionados.

Debe existir una relación lógica entre los elementos del sistema. Por ejemplo, los sistemas mecánicos y electrónicos (como lavadoras y juegos de video) tienen muchos componentes que trabajan juntos.

3. El propósito de lograr algún objetivo.

El sistema se diseña para alcanzar uno o más objetivos. Todos los elementos del sistema deben estar ligados y controlados, de manera que se logre el objetivo del sistema.

Algo que podemos notar de acuerdo a lo anterior, es que un sistema (o parte de un sistema) es casi todo aquello con lo que estamos en contacto durante nuestra vida como son la sociedad, los aparatos electrodomésticos, la empresa en la que laboramos, un sistema numérico, un río y hasta nuestro propio cuerpo. Por lo tanto, son posibles muchas formas de clasificar los sistemas, y empezaremos por dividirlos en dos categorías:

#### 4.1.1 SISTEMAS NATURALES

Se refiere a los sistemas que existen en la naturaleza y sirven a sus propios fines. Se subdividen en:

##### 4.1.1.1 SISTEMAS FÍSICOS

Incluyen:

- ✓ Sistemas estelares: Las galaxias y sistemas solares.

- ✓ Sistemas geológicos: Ríos y cordilleras.
- ✓ Sistemas moleculares: Organizaciones complejas de átomos vivientes.

#### 4.1.1.2 SISTEMAS VIVIENTES

Comprenden toda gama de animales y plantas que nos rodean, al igual que la raza humana.

#### 4.1.2 SISTEMAS HECHOS POR EL HOMBRE

Se refiere a los sistemas que son construidos, organizados y mantenidos por humanos, entre los que se encuentran:

- ✓ Sistemas sociales: Doctrinas y costumbres.
- ✓ Sistemas de transporte: Redes de carreteras y aerolíneas.
- ✓ Sistemas de comunicación: Teléfono y clave morse.
- ✓ Sistemas de manufactura: Fábricas y líneas de ensamblado.
- ✓ Sistemas financieros: Contabilidad, inventarios y bolsa de valores.

La mayoría de los sistemas hechos por el hombre incluyen las computadoras, de hecho, muchos no podrían existir sin ellas. Sin embargo, es importante señalar que dichos sistemas existían antes de que hubiera computadoras; aunque algunos sistemas continúan por completo sin computarizar y podrían permanecer así durante muchas décadas más. Otros contienen a la computadora como componente, pero incluyen uno o más componentes no computarizados (o manuales).

Es en este punto donde surge el siguiente concepto.

#### 4.2 SISTEMAS AUTOMATIZADOS O SISTEMAS DE INFORMACIÓN

Conocidos también como *Sistemas de Información* o *Sistemas Informáticos*, son aquellos sistemas creados por el hombre que interactúan y/o son controlados por una o más computadoras. El objetivo de este tipo de sistemas, es llevar a cabo un conjunto de funciones específicas.

#### 4.2.1 ELEMENTOS DE UN SISTEMA DE INFORMACIÓN

- ✓ **Recurso Humano:** Lo integran aquellas personas que interactúa con el sistema. Por una parte se encuentran los individuos que desarrollan, administran y operan el sistema; por otra parte, aquellos que utilizan los resultados que de él se generan.
- ✓ **Datos fuente:** Son introducidos al sistema, es decir, son todas las entradas necesarias para generar la información que se desea.
- ✓ **Hardware:** Es la parte física de las computadoras, necesaria para que el sistema de información pueda operar. Se integra por elementos electrónicos, cables y equipo periférico.
- ✓ **Software:** Lo constituyen los programas que hacen que los datos fuente sean procesados correctamente y, así, se generen los resultados esperados.
- ✓ **Documentación:** Descripción escrita del sistema y su operación, que permite la integración de funciones entre el hombre y la computadora. Esta información se encuentra en los manuales de políticas, métodos y procedimientos de la organización.

En la actualidad, se han integrado las Redes de Telecomunicaciones como parte fundamental de los sistemas de información. Una Red de Telecomunicaciones es el conjunto de elementos (*hardware* y *software*) que permiten el transporte electrónico de voz, sonidos, datos, textos, imágenes u otras señales de cualquier naturaleza entre dos o más puntos (nodos) de la red.

Por otra parte, el concepto de Red de Telecomunicaciones se pueden incluir en un término más general: Tecnologías de Información (*Information Tecchnology*, IT). IT, se refiere a todas aquellas tecnologías que permiten y dan soporte a la construcción y operación de los sistemas de información (redes, satélites, fibra óptica, *software*, servicio de transferencia electrónica, unidades de almacenamiento de datos, etc.)

#### **4.2.2 ACTIVIDADES BÁSICAS DE UN SISTEMA DE INFORMACIÓN**

- ✓ Entrada de Información: Datos con los que se inicia el funcionamiento del sistema. Las entradas pueden ser manuales (se proporcionan en forma directa por el usuario) o automáticas (también llamadas interfaces automáticas, provienen o son tomados de otros sistemas o módulos)
- ✓ Almacenamiento de información: Es una de las actividades más importantes, ya que, a través de ésta, el sistema puede recordar la información guardada en el proceso anterior.
- ✓ Procesamiento de Información: Es la capacidad del sistema para efectuar una secuencia de operaciones preestablecida para un objetivo específico. Dichas operaciones, pueden efectuarse con datos introducidos recientemente en el sistema o bien, con datos almacenados.
- ✓ Salida de Información: Es la capacidad de un sistema para obtener información específica, una vez procesados los datos. Es importante aclarar que la salida de un Sistema de Información, puede constituir la entrada a otro Sistema de Información o módulo.

#### **4.2.3 SISTEMAS DE INFORMACIÓN ADMINISTRATIVOS**

Dentro de los Sistemas de Información, encontramos los Sistemas de Información Administrativos, cuyo objetivo es apoyar las actividades de una organización.

Un sistema de información administrativo se compone de los elementos antes mencionados, así como, de las actividades antes descritas.

##### **4.2.3.1 OBJETIVOS**

- ✓ Automatización de procesos operativos.
- ✓ Proporcionar información que sirva de apoyo al proceso de toma de decisiones.
- ✓ Lograr ventajas competitivas a través de su implantación y uso.



### 4.2.3.2 CLASIFICACIÓN

#### SISTEMAS TRANSACCIONALES

Son aquellos que logran la automatización de procesos operativos dentro de una organización. Su función primordial consiste en procesar transacciones<sup>70</sup>. Algunos ejemplos de este tipo de sistemas son la facturación, nóminas, contabilidad general, etc.

#### CARACTERÍSTICAS

- ✓ Automatizan tareas operativas de la organización, por lo que, a través de ellos, suelen lograrse ahorros significativos.
- ✓ Por lo general, son el primer tipo de sistema que se implanta en las organizaciones; dado que, se inicia apoyando las tareas a nivel operativo, se continúa con los mandos intermedios y se finaliza con la alta administración conforme evolucionan.
- ✓ Son intensivos en entrada y salida de información; sus cálculos y procesos suelen ser simples y poco sofisticados. Este tipo de sistemas requieren de mucho manejo de datos para poder realizar sus operaciones y como resultado generan también grandes volúmenes de información.
- ✓ Son recolectores de información, es decir, a través de ellos se cargan las grandes bases de información para su explotación posterior. Estos sistemas son los encargados de integrar gran cantidad de la información que se maneja en la organización, la cual, se utilizada posteriormente para apoyar a los mandos intermedios y altos.
- ✓ Son fáciles de justificar ante la dirección general, ya que, sus beneficios son visibles y palpables. El proceso de justificación puede realizarse enfrentando ingresos y costos. Esto se debe a que, en el corto plazo, se pueden evaluar los resultados y ventajas del mismo.
- ✓ Proporcionan reportes y listados.
- ✓ El procesamiento de transacciones incluye, entre otras, las siguientes actividades: cálculos, clasificación, ordenamiento, almacenamiento y recuperación y generación de resúmenes.

#### SISTEMAS DE APOYO A LAS DECISIONES

Los sistemas que proporcionan la información que será empleada en el proceso de toma de decisiones son los *Sistemas de Soporte a la Toma de Decisiones (DSS)*, *Sistemas para la Toma de Decisiones de Grupo (GDSS)*, *Sistemas Expertos de Soporte a la Toma de Decisiones (EDSS)* y *Sistemas de Información para Ejecutivos (EIS)*.

#### CARACTERÍSTICAS

- ✓ Suelen introducirse después de haber implantado los Sistemas Transaccionales más relevantes de la empresa, ya que estos últimos, constituyen su plataforma de información.
- ✓ La información que generan, sirve de apoyo a los mandos intermedios y a la alta administración en el proceso de toma de decisiones.

---

<sup>70</sup> Transacción: Es cualquier suceso o actividad que afecta a toda la organización. Por ejemplo: pagos, cobros, pólizas, etc.

- ✓ Suelen ser intensivos en cálculos y escasos en entradas y salidas de información. Así, por ejemplo, un modelo de planeación financiera requiere poca información de entrada, genera poca información como resultado, pero puede realizar muchos cálculos durante su proceso.
- ✓ No suelen ahorrar mano de obra, por lo que, la justificación económica para el desarrollo de estos sistemas es difícil.
- ✓ Suelen ser sistemas de información interactivos y amigables, con altos estándares de diseño gráfico y visual, ya que, están dirigidos al usuario final.
- ✓ Este tipo de sistemas puede incluir la programación de la producción, compra de materiales, flujo de fondos, proyecciones financieras, modelos de simulación de negocios, modelos de inventarios, etc.

### **SUBCLASIFICACIÓN**

- ✓ Sistemas de Apoyo a la Toma de Decisiones (*Decision Support System, DSS*).
- ✓ Sistemas de apoyo a la Toma de Decisiones de Grupo (*Group Decision Support System, GDSS*).
- ✓ Sistemas de Información para Ejecutivos (*Executive Information System, EIS*).
- ✓ Sistemas Expertos de Apoyo para la Toma de Decisiones (*Expert Decision Support System, EDSS*).

### **SISTEMAS ESTRATÉGICOS**

Los Sistemas Estratégicos tienen un uso y objetivo claramente específico; se desarrollan en las organizaciones con el fin de lograr ventajas competitivas, a través, del uso de la tecnología de información.

### **CARACTERÍSTICAS**

- ✓ Su función es lograr ventajas que los competidores no posean, como en costos y servicios diferenciados con clientes y proveedores. Por ejemplo, el uso de cajeros automáticos en los bancos es un Sistema Estratégico, ya que, brinda ventaja sobre un banco que no posee tal servicio. Si un banco nuevo decide abrir sus puertas al público, tendrá que dar este servicio para tener un nivel similar al de sus competidores.
- ✓ Apoyan el proceso de innovación de productos y procesos dentro de la empresa, debido a que buscan ventajas respecto a los competidores y una forma de hacerlo es innovando o creando productos y procesos.

Un ejemplo de estos Sistemas de Información dentro de la empresa puede ser un sistema de Planeación del Recurso de Manufactura (*Manufacturing Resource Planning, MRP*), enfocado a reducir sustancialmente el desperdicio en el proceso productivo, o bien, un Centro de Información que proporcione todo tipo de información; como situación de créditos, embarques, tiempos de entrega, etc. En este contexto, los ejemplos anteriores constituyen un Sistema de información Estratégico si, y sólo si, apoyan o dan forma a la estructura competitiva de la empresa.

## 4.2.4 LOS SISTEMAS DE INFORMACIÓN EN LA ESTRATEGIA DE LA EMPRESA

Años atrás, la visión estratégica que había en las empresas respecto a los Sistemas de Información era, que debían cubrir la automatización de sus transacciones y proporcionar a los altos mandos, la información necesaria para la toma de decisiones.

En la actualidad, los Sistemas de Información son vistos como áreas de oportunidad para lograr ventajas en el terreno de los negocios, ya que, éstos pueden representar un diferencial o valor agregado con respecto a los competidores. La perspectiva estratégica considera a los Sistemas de Información como una herramienta para mejorar la estructura competitiva del negocio, por lo que, tienen su área de influencia en el medio ambiente de la empresa, por ejemplo: a través de nuevos servicios a clientes, oportunidades de inversión y nuevos productos y mercados.

Por otra parte, los Sistemas de Información pueden influenciar la manera en que la empresa desarrolla su trabajo interno, ya sea para aumentar la productividad y/o disminuir los costos. Por lo tanto, la visión estratégica es la necesidad de entender de qué forma la tecnología de la información puede colocar a la organización un paso delante de la competencia.

A continuación, se presentan algunos mecanismos y estrategias a través de los cuales, la organización logra ventaja competitiva.

- ✓ Reducción de costos: Se puede generar ventaja competitiva cuando una empresa es capaz de vender más productos a menor precio, siempre y cuando mantenga la calidad; al mismo tiempo que logra incrementar su margen de utilidad. Lo que permitirá reducir los precios al consumidor es la completa automatización de los procesos operativos y productivos de un negocio, lo que lo lleva a una mayor productividad.
- ✓ Tecnología de entrada al mercado: Existe mayor facilidad de hacer negocio si en un mercado hay pocos competidores; pero la tecnología puede ser un obstáculo para la nueva empresa, ya que, si está no cuenta por lo menos con la tecnología o los elementos básicos que poseen las empresas, simplemente no podrá competir en el mercado.
- ✓ Cambio de proveedor o servicio: Entendemos que cuando una empresa trata de cambiar de proveedor o servicio, esto conlleva a un aumento de gastos (tiempo y dinero); pero una vez que la empresa logra dicha introducción, aumentará su ventaja competitiva.
- ✓ Generación de nuevos productos o servicios: De esta manera, se genera una ventaja competitiva que perdurará hasta el momento en que otro competidor logre introducir al mercado un producto similar. Por ejemplo, las empresas que venden información, procuran innovar sus servicios en forma continua creando nuevas y diferentes bases de datos de interés para el público.
- ✓ Campañas publicitarias: Cuyo objetivo es persuadir a los consumidores de que los productos o servicios ofrecidos, son mejores que los que ofrece la competencia.
- ✓ Mejoramiento de productos o servicios: Si el producto o servicio es mejor que cualquiera de los que ofrecen los competidores, generará ventaja competitiva. Por ejemplo la empresa norteamericana Hertz, dedicada principalmente a la renta de automóviles, incluyó un sistema de navegación en el automóvil llamado *NeverLost*; dicho dispositivo, guía al usuario hasta

su destino, indica continuamente la ubicación exacta del automóvil y proporciona al usuario información actualizada sobre carreteras, hoteles, restaurantes, campos de golf, complejos deportivos, servicios de emergencia, centros de interés comercial, aeropuertos e incluso direcciones particulares. *NeverLost* combina un sistema de ubicación global mediante satélite con elementos tecnológicos de cálculo de estima giroscópica<sup>71</sup> y comparación de mapas para guiar a los clientes hasta el destino elegido.

- ✓ Creación de alianzas: Entendiendo por alianzas, la combinación de dos o más grupos de individuos que se unen para lograr un objetivo común. Las alianzas se pueden lograr a través de la adquisición de nuevas empresas, de la creación de una nueva empresa con aportaciones de capital o de dos o más empresas o bien, a través de la cooperación entre varias organizaciones o individuos.

#### **4.2.5 SISTEMAS DE INFORMACIÓN ORIENTADOS A LA WEB**

Desde que nació la *World Wide Web*, se ha ido integrando a la vida de un gran número de personas que la utilizan como instrumento de ocio, herramienta de trabajo o fuente enorme de información. En años recientes, las organizaciones reconocen que la *Web* no es solamente un enorme depósito de información, sino que es también, una fuente de conocimiento que les permite identificar a sus clientes, qué ofrecen sus competidores, saber sobre posibles competidores potenciales a futuro, conocer mejor a sus usuarios y sus necesidades, así como, las fuentes complementarias de la información de interés potencial.

Los Sistemas de Información *Web* son el medio por el cual, la organización logra lo antes mencionado.

##### **4.2.5.1 CONCEPTO**

Los Sistemas de Información *Web* (*Web Information Systems, WIS*), son aquellos sistemas construidos bajo la tecnología *Web*.

Actualmente, los *WIS* proporcionan una plataforma que puede ayudar a establecer y manejar relaciones de cliente con mecanismos que no eran factibles con modelos tradicionales de negocio. Además, facilitan la entrega de contenido personalizado para terminar a consumidores, reflejando sus necesidades únicas y preferencias individuales.

##### **4.2.5.2 REQUERIMIENTOS**

- ✓ Navegador (*browser*).
- ✓ Conexión a Internet, que puede ser a través de un Proveedor de Servicios de Internet (*Internet Service Provider, ISP*) o una red LAN.
- ✓ Y para su desarrollo, tecnologías *Web* (descritas en este mismo capítulo).

---

<sup>71</sup> *Giroscópico: Movimiento cónico, pendular que deriva del centro de gravedad con el movimiento de rotación.*

### 4.2.5.3 ROL DE USUARIO

En la *Web* existe el concepto multi-rol, se refiere a que los usuarios pueden ser productores, consumidores, o productores/consumidores de información.

### 4.2.5.4 CARACTERÍSTICAS

- ✓ Son útiles para promocionar a una organización, sin tener límites de fronteras (no existen restricciones al acceso de información).
- ✓ Para buscar información acerca de un tópico específico.
- ✓ Para comunicarse con otras personas en diferentes partes del mundo.
- ✓ Para compartir y difundir el conocimiento.
- ✓ Para difundir la cultura.
- ✓ Para realizar transacciones, sin la necesidad de estar dentro de la oficina; puede ser desde el hogar, el avión, e inclusive desde la playa.
- ✓ Para vender productos a nivel nacional e internacional, sin la necesidad de conocer personalmente al comprador (no hay contacto humano entre vendedor y cliente).
- ✓ Los documentos son implementados a través de páginas HTML, habilitando: la portabilidad y una interfaz homogénea.
- ✓ Las aplicaciones están basadas en documentos que son creados, intercambiados, actualizados y vistos por varios tipos de usuarios (el acceso no está limitado a un tipo de usuario).
- ✓ Los documentos están compuestos de manera estática (texto, gráficos, etc.) y dinámica (applets Java, formularios HTML, etc., conceptos que se describirán en este mismo capítulo).
- ✓ Los datos pueden ser almacenados dentro de los documentos o en Bases de Datos.

### 4.2.5.5 EVOLUCIÓN

Los Sistemas de Información *Web* han seguido un desarrollo (v. figura 4.1) de acuerdo a las necesidades de la organización, de este modo, se ha llegado a un complejo modelo que presentamos a continuación.

#### ETAPA 1: WIS ESTÁTICOS

Los WIS se componen de documentos hypermedia que son desplegados en un navegador. Su principal característica es que el flujo de información es unidireccional (del servidor al cliente o usuario), no existe retroalimentación por parte del usuario.

#### ETAPA 2: WIS DINÁMICOS

Es introducida la retroalimentación por parte del usuario, se utilizan formularios como interfaz para que el usuario ingrese su información y esta es enviada al servidor a través de programas CGI (concepto que se describirá en este mismo capítulo). También se usan *cookies*<sup>72</sup> para conocer las preferencias del usuario de acuerdo a las páginas visitadas.

<sup>72</sup> *Cookie: Dentro de la Web, procedimiento ejecutado por el servidor, que consiste en guardar información acerca del cliente para su posterior recuperación. En la práctica, la información es proporcionada desde el*

### ETAPA 3: WIS ADAPTABLES

La retroalimentación del usuario comienza a tener un importante impacto en el diseño de los sitios, ya que causa la instantánea regeneración de contenido, presentación y estructura de navegación. Se crean aplicaciones comerciales, es decir WIS dedicados a realizar transacciones de venta en línea; estas aplicaciones abarcan el manejo de inventarios en el almacén, mercadotecnia, traspaso de dinero.

### ETAPA 4: COMPLEJOS AGENTES DE NEGOCIACIÓN (MÁS ALLÁ DE LA ARQUITECTURA CLIENTE/SERVIDOR)

Los WIS mejorarán su comunicación con respuestas dinámicas y con el surgimiento del siguiente concepto. Agente (*Agent*), que es un sistema que incrementa la flexibilidad y cambia radicalmente, las características inherentes de los WIS

Se reemplaza la arquitectura cliente/servidor, por el modelo *Principal/Agent* (Director-Agente), el cual, a través de *software-agent* inteligente, actúa como cliente y servidor al mismo tiempo, dependiendo de las preferencias del *Principal*, los requerimientos de sus tareas y predefiniendo la coordinación de los mecanismos.

El modelo *request-responses* (petición-respuesta) *Web* y HTTP, se sustituye por la interacción directa de socios en un *agent-driven*, es decir un agente que dirige la comunicación en la red. En esta etapa se hace énfasis en el proceso de negociación.

Los agentes de *software* inteligentes pueden ser categorizados de acuerdo a su funcionalidad y a los atributos de su arquitectura dentro de la información, cooperación y agentes de transacción:

- ✓ Agentes de información: Usados para individualizar la comunicación en Internet y presentar a cada uno de los usuarios una interfaz intuitiva y personalizada, que oculta la más compleja arquitectura del sistema. Generalmente, almacenan el perfil del usuario del lado del cliente (por lo general en un navegador) o del lado del servidor (a través de un WIS). En lo que respecta a la adaptabilidad de los WIS, estos son más útiles como una puerta para el ofrecimiento de información.
- ✓ Sistemas de Agentes Cooperativos: Están basados en las evaluaciones subjetivas de personas (información social) acerca del contenido en línea, liberado. A través de Sistemas de filtrado, se identifican patrones de conducta de un usuario en particular. Ante esto, los agentes de cooperación son empleados para sincronizar, compartir y comunicar las preferencias subjetivas de grupos de usuarios usando analogías, correlaciones y otros métodos estadísticos
- ✓ Agentes de Negociación o Transacción: Son programas altamente especializados que son usados para llevar a cabo la negociación y las fases de establecimiento de comercio

---

*navegador al servidor Web, a través de un archivo de texto, que queda almacenado en el disco duro de la computadora y puede ser recuperado posteriormente, cuando se acceda al servidor.*

electrónico. Los procesos de negociación entre entidades de negocio, son caracterizados con altos grados de imprevisibilidad, complejidad e importancia para las organizaciones. Los agentes de negociación, por lo tanto, ampliarán y derribarán marcos existentes para los WIS y los mercados electrónicos.

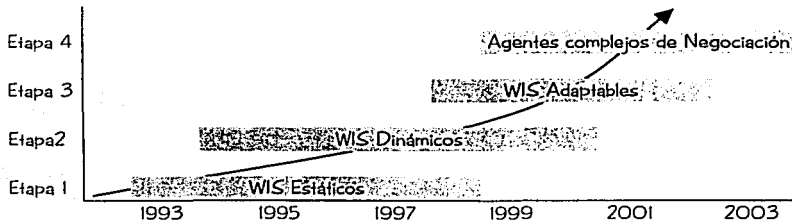


Figura 4.1 Evolución de los Sistemas de Información Web (WIS)

El desarrollo descrito, comienza a ser evidente en la automatización de mecanismos complejos de negociación y modelos para aplicaciones comerciales.

#### 4.2.5.6 TECNOLOGÍAS WEB

El surgimiento de las Tecnologías de la Información juega un papel cada vez más importante. No sólo automatizan tareas dentro de organizaciones, también proporcionar la infraestructura para facilitar la comunicación, a través, de fronteras de las mismas, pone en práctica estrategias de control de comercialización de uno a uno, o manejar relaciones de negocio.

Las tecnologías *Web* son el conjunto de conocimientos, técnicas y procesos para el diseño y construcción de objetos que funcionan en la *Web*.

#### APLICACIÓN WEB

Retomando, la *Web* se diseñó originalmente como un medio para suministrar páginas estáticas a los usuarios de Internet, utilizando el Protocolo de Transferencia de Hipertexto (HTTP) como su principal protocolo de transporte. Cuando un navegador envía una consulta HTTP a un servidor *Web*<sup>73</sup>, este último extrae un archivo de consulta de su sistema de archivos y lo devuelve al navegador a través de la conexión HTTP. Sin embargo, lo que devuelve el servidor *Web* no tiene por qué ser siempre un archivo estático almacenado en el servidor, puede tratarse de la salida de un programa. En otras palabras, el servidor *Web* puede actuar, junto con un programa (por ejemplo de recuperación de base de datos) como un programa de aplicación que responde a consultas HTTP. De este modo, ya podemos hablar propiamente del concepto de aplicación *Web*.

<sup>73</sup> *Web Server (Servidor Web):* Dentro de la World Wide Web, programa que acepta solicitudes de información que cumplen con el protocolo HTTP. El servidor procesa las solicitudes y envía lo que se le ha solicitado.

Para lograr el objetivo de las aplicaciones, han surgido y han ido evolucionando una variedad de herramientas, las principales se presentan a continuación.

## HERRAMIENTAS PARA EL DESARROLLO DE APLICACIONES WEB

### HTML

Como hemos visto, el HTML (Lenguaje de Marcado de Hipertexto/*Hypertext Markup Language*), es un lenguaje de descripción y formato para las páginas *Web*. Es una especie de *PostScript* de la *Web*, que sólo puede leerse y que ha sido creado en función de las posibilidades específicas de la representación en línea.

Dentro de los elementos del código HTML, encontramos los formularios, que añaden una dimensión nueva a las páginas HTML, ya que, éstas pueden ser más o menos sofisticadas pero sin los formularios son de sólo lectura.

Un formulario es una Interfaz Gráfica de Usuario (*Graphical User Interface, GUI*), que permite enviar información al servidor para ser procesada por un programa de tratamiento denominado CGI (*Common Gateway Interface/Interfaz Común de Pasarela*). En otras palabras, los formularios son plantillas que permiten incorporar a un documentos HTML, peticiones de datos al servidor y viceversa. La principal utilidad de los formularios es la posibilidad de crear cuestionarios, encuestas, páginas de comentarios o cualquier documento en la que se desee una interacción por parte del usuario.

### PROGRAMAS CGI

CGI consiste en un conjunto de normas que permiten a una página *Web*, acceder programas externos a la misma. Estos programas deben, a su vez, seguir también una serie de normas. A los programas que cumplen estas reglas, impuestas por la interfaz CGI, se les llama programas CGI, *scripts* CGI o simplemente CGI's, y pueden estar escritos en cualquier lenguaje, tanto interpretados<sup>74</sup> como compilados<sup>75</sup> siendo los siguientes, los más usados:

- ✓ Perl (Lenguaje Práctico de Extracción e Informes/*Practical Extraction and Report Language*). Es un lenguaje creado por Larry Wall, con el objetivo principal de simplificar las tareas de administración de un sistema UNIX; en realidad hoy en día se ha convertido en un lenguaje de propósito general, y una de las principales herramientas en Internet. En cuanto a sus características técnicas, Perl es un lenguaje interpretado, aunque en realidad, el intérprete de Perl como todos los intérpretes modernos, compila los programas antes de ejecutarlos.

---

<sup>74</sup> *Intérprete: Nombre con el que se conoce a aquellos programas que son capaces de ejecutar una aplicación escrita en un lenguaje de programación de alto nivel (comandos escritos) convirtiendo su código fuente en el momento de iniciarla a código máquina, que es el único que entiende realmente la computadora.*

<sup>75</sup> *Compilador: Programa que lee las líneas escritas en un lenguaje de programación y las traduce a otro que pueda ejecutar la computadora. Los programas compilados se ejecutan más rápido que los interpretados, debido a que han sido completamente traducidos a lenguaje de máquina y no necesitan compartir memoria con el intérprete.*



- ✓ Cualquier *shell* de Unix: Son usados por su increíble sencillez y facilidad de aprendizaje. Sin embargo, tienen fallas de seguridad y no es aconsejable usarlos excepto para CGI's muy sencillos.
- ✓ C o C++: Son también muy usados. Ofrecen la ventaja de que son lenguajes mucho más potentes, pero tienen otros inconvenientes: que más difíciles de aprender, es necesario compilarlos y sobre todo porque lleva más tiempo hacer un programa usándolos.

### SCRIPTS EN JAVASCRIPT

*JavaScript* es un poderoso lenguaje escrito (*script*) diseñado exclusivamente para la WWW. El código *JavaScript* es escrito directamente en una página HTML y es interpretado y ejecutado por los navegadores del lado del cliente. El primer uso *JavaScript* fue para crear más interacción en las interfaces de usuario, en específico las interfaces de usuario creadas con formularios HTML. Pero *JavaScript* puede estar en funciones de formularios que son llamadas a través de formularios y eventos del ratón, y proveen a una página mejor presentación y funcionalidad. Con *JavaScript* también se puede acceder a componentes de una página y del navegador como son formularios, ligas, imágenes, ventanas y la *history* de navegación.

### APPLETS JAVA

Java es un completo lenguaje de programación orientado a objetos desarrollado por Sun Microsystems. Los programas escritos en Java (archivos .java) son precompilados dentro de uno o más archivos .class, llamados bytecodes. Estos bytecodes son referenciados en páginas HTML, por medio de la etiqueta "APPLET". Cuando estas páginas son cargadas en el cliente los bytecodes son transferidos, interpretados y ejecutados por el navegador en cualquier plataforma y sistema operativo.

La mejor característica de Java es que puede ser usada para crear programas (applets) que se ejecutan desde las páginas HTML del lado del cliente, pero Java provee casi toda la funcionalidad presentada en otros lenguajes como C++. Además periódicamente se está actualizando el lenguaje ya que se han ido agregando nuevas características como mejorar el control de seguridad, conexión desde Internet a la Base de Datos (a través del paquete JDBC), seguridad en las transacciones comerciales (a través de Java *Commerce API*), programación distribuida (a través de RMI, *Remote Method Invocation*) y programación de componente (Java *Beans*). Todo esto hace a Java un lenguaje especial para el desarrollo en Internet.

### XML

XML (*eXtensible Markup Language*) es un lenguaje de metamarcado que ofrece un formato para la descripción de datos estructurados. Esto facilita unas declaraciones de contenido más precisas y unos resultados de búsquedas más significativos en varias plataformas. Además, XML habilitará una nueva generación de aplicaciones para ver y manipular datos basadas en la *Web*.

XML, que proporciona un estándar de datos que puede codificar el contenido, la semántica y los esquemas de una gran variedad de casos, desde los más simples a los más complejos, sirve para marcar lo siguiente:

- ✓ Un documento normal.

- ✓ Un registro estructurado, como un registro de citas o un pedido de compra.
- ✓ Un objeto con datos y métodos, como el formulario permanente de un objeto Java o de un control ActiveX.
- ✓ Un registro de datos, como el conjunto de resultados de una consulta.
- ✓ Metacontenido sobre un sitio *Web*, como el formato de definición de canal (CDF).
- ✓ Representaciones gráficas, como la interfaz de usuario de una aplicación.
- ✓ Entidades y tipos de esquema estándar.
- ✓ Todos los vínculos entre datos y personas que hay en la *Web*.

Cuando los datos llegan al escritorio del cliente, se pueden manipular, editar y presentar en varias vistas, sin tener que regresar al servidor. Ahora los servidores pueden ser más escalables, gracias a la reducción de las cargas de ancho de banda y computación. Además, dado que los datos se intercambian en el formato XML, se pueden combinar fácilmente desde distintas fuentes.

XML es muy valioso para Internet, así como para los entornos de intranets<sup>76</sup> corporativas, ya que proporciona interoperabilidad mediante un formato basado en estándares flexible y abierto, con formas nuevas de acceso a las bases de datos existentes y de entregar datos a clientes de *Web*. Las aplicaciones se pueden generar más rápidamente, su mantenimiento es más sencillo y pueden ofrecer fácilmente varias vistas de los datos estructurados.

En un principio, no rivalizarán HTML y XML, estos se complementarán el uno al otro, anudándose ambas gramáticas. Este Lenguaje de marcas extensible (XML) es una versión abreviada del SGML (Standard Generalized Markup Language).

#### **4.2.5.7 CLASIFICACIÓN**

A continuación, presentamos los Sistemas que han hecho, en su momento y aún en la actualidad, de la *Web*, el más importante servicio que brinda Internet.

#### **BUSCADORES**

En la actualidad, la cantidad de información disponible en Internet es inmensa y sigue incrementándose día con día, lo que implica, cierta dificultad para realizar una búsqueda y obtener los resultados deseados, además, ocurre muchas veces que el usuario llega a perderse entre tantas páginas *Web*.

Por lo anterior, surgieron en escena servidores de páginas *Web*, dedicados exclusivamente y de forma estructurada, a recopilar, clasificar y entregar información específica a los usuarios. Dichos servidores se denominan Buscadores.

---

<sup>76</sup> *Red informática de comunicación privada de una empresa u organización de cualquier tipo que funciona con las mismas tecnologías usadas en la red Internet y que se puede conectar a ella de forma sencilla dada su misma naturaleza tecnológica.*

## CUALIFICACIÓN

### TEMAS RELACIONADOS

- ✓ La estructura de un sitio de datos<sup>11</sup> es imprescindible para indicar las categorías de los datos. Pero que a su vez se subdividen en subcategorías, es sucesivamente hasta conseguir niveles e las páginas registradas.
- ✓ Es imprescindible que el usuario registre, posteriormente, de de ante las páginas Web para figurar en a lista de datos de servicios.
- ✓ La información proporcionada por el usuario se codifica manualmente por el mismo o por intermediarios de los datos Web. En estos últimos en muchos casos analizar, procesar y organizar la lista de páginas, cuando la dirección asistida en los datos proporcionados por el usuario con el fin de determinar a este último con los requisitos necesarios para ser capaz de ara, acceder, verificar que los datos introducidos sean correctos.
- ✓ Una vez el proceso de ara para que el usuario registre una buena posición nominalmente no es suficiente, es necesario que el usuario sea<sup>12</sup> es suficiente que dentro del código HTML el título "titulo", de una manera que a ara más cercana a la " en el apartado o por alguno de los términos relacionados de la clasificación ISO. Esto se debe a que los resultados devueltos en este apartado en la mayoría de los casos y ordenados por categorías específicas.
- ✓ Conviene para hacer páginas específicas con temas comunes, buscar información general o usar una materia más común.

### Los motores de búsqueda

- ✓ Los contenidos son indexados<sup>13</sup> por medio de un robot<sup>14</sup>.

<sup>11</sup> Base de datos: Conjunto estructurado de, como su nombre indica, datos. Estos datos son representaciones de sucesos y objetos en diferente nivel, existentes en el mundo real en su conjunto, representan algún tipo de entidad existente. En el mundo real se tiene percepción sobre las entidades u objetos y sobre los atributos de esos objetos; en el mundo de los datos, hay registros de eventos y datos de eventos. Además, en ambos mundos se puede, incluso, distinguir una tercera faceta: aquella que comprende las definiciones de las entidades, externas, o bien las definiciones de los registros y de los datos.

<sup>12</sup> Son aquellas palabras que ingresan el usuario en el proceso de registro del sitio Web y que van a identificar el contenido de la página. Estas palabras son utilizadas por los buscadores al momento de desplegar resultados de sitios solicitados.

<sup>13</sup> Etiqueta HTML que permite especificar el título de un documento. Este título no aparece en el documento HTML en sí, sino se muestra en la parte superior izquierda de la ventana del navegador. Esta etiqueta es una manera en como el sitio es identificado por algunos buscadores.

<sup>14</sup> Indexar: Acción de ordenar los registros contenidos en una base de datos de una forma especial en función de parámetros predefinidos.

<sup>15</sup> Programa que atraviesa la Web con distintos fines: Análisis sobre el crecimiento de la red, número de servidores activos; Mantenimiento de la estructura hipertexto de la Web; Duplicación de directorios FTP (Altores), para incrementar su disponibilidad a un mayor número de usuarios; Investigación, localizando recursos Web para el usuario.

- ✓ Es aconsejable que el usuario dé de alta el sitio *Web* para figurar en él pero no imprescindible, pues la mayoría de los motores buscan por la *Web* por ellos mismos, indexando y recopilando toda la información que hallan a su paso.
- ✓ Las bases de datos de los motores son mucho más amplias que las de los índices, como norma general, y se actualizan frecuentemente. Esto es debido a la forma en como es colocada la información en ambos casos.
- ✓ Para lograr una buena posición es necesario el correcto uso de palabras clave y de etiquetas (*metatags*<sup>B2</sup>), estas últimas, dentro del código HTML.
- ✓ Es difícil encontrar la información que deseamos en un motor, si no se conoce el lenguaje de interrogación<sup>B3</sup> a su base de datos, ya que, podría suceder que nuestra solicitud fuese devuelta en blanco o llena de información inútil. Por el contrario, si éste se conoce, resultan ser los buscadores más potentes.
- ✓ Los motores son ideales para hallar temas poco comunes.
- ✓ Los robots se clasifican de la siguiente manera:
  - Orugas (*Web crawlers*): Robots con capacidad de búsqueda mucho más amplia y con capacidad de actualización automática.

Vagabundos (*Wanderers*): Robots encargados de la medición del crecimiento de la *WWW*, a través, de un análisis para determinar el número de servidores activos por ejemplo.

*Knowbots*: Robots programados para localizar referencias hipertexto dirigidas hacia un documento, servidor, etc., en particular. Este tipo de Robots, permiten evaluar el impacto de las distintas aportaciones que engrosan las distintas áreas de conocimiento presentes en la red.

Arañas (*Spiders*): Robots encargados de generar una base de datos tomando como referencia, los distintos nexos que configuran la estructura hipertexto de la *Web*. Esta base de datos posibilitará la ejecución de ecuaciones de búsqueda sobre una materia concreta y, la localización inmediata de recursos informativos eludiendo la necesidad de navegar a través de la red siguiendo la estructura hipertexto.

Mecanismos de Búsqueda Unificados (*Unified Search Engines*): Se refiere a la concentración de distintos motores de búsqueda a los que se puede acceder de forma

---

<sup>B2</sup> *Metatags (Metacaracteres):* Sentencias de código HTML que, si bien no son visibles desde el explorador, ayudan a que los motores de búsqueda localicen un sitio cuando se realiza una búsqueda. Las etiquetas dentro del código HTML son: título (*title*), descripción (*description*) y palabras clave (*keywords*) mismas que, deben contener la información más apropiada y contundente relacionada con nuestro sitio Web.

<sup>B3</sup> *Lenguaje de interrogación:* Conjunto de órdenes, operadores y estructuras que, organizados según ciertas normas lógicas, permiten la consulta de fuentes y recursos de información electrónica. El resultado de la combinación de estos elementos, siguiendo las normas establecidas, es una expresión que se identifica con el nombre "ecuación" capaz de interrogar el contenido de la fuente de información.

consecutiva en una misma página, no se trata de una interfaz que posibilite la interrogación simultánea de distintos buscadores.

Gusanos (*Worms*): Robot encargado de tareas de duplicación (creación de *Mirrors*<sup>84</sup>).

Hormigas (*WebAnts*): Conjunto de Robots físicamente alejados que cooperan para la consecución de distintos objetivos.

#### Multibuscadores

- ✓ No tienen una base de datos propia.
- ✓ Buscan la página en unos cuantos motores de búsqueda e índices temáticos, y combina los resultados de la búsqueda en esos buscadores.
- ✓ Para registrar una dirección de manera que aparezca en un multibuscador, debemos mandársela a algún buscador donde el multibuscador va a recoger los resultados.
- ✓ En el caso de los Motores de búsqueda y de los Índices temáticos, el contenido de sus bases de datos se puede clasificar en:

Geográficos: La base de datos se limita a la pertenencia o no de una zona concreta. Los buscadores multinacionales abarcan varios países; los nacionales están restringidos a un único país; los regionales se limitan a comunidades autónomas, regiones o incluso provincias y, los locales, a ciudades o comarcas.

Lingüísticos: El idioma es el criterio unificador de la base de datos, trascendiendo el ámbito geográfico.

Étnicos: Normalmente, se refiere a grupos de personas o minorías raciales.

Temáticos: Las direcciones se organizan en torno a un tema concreto.

Una vez tratado lo anterior, podemos decir que, no sólo es importante crear una página *Web* original y útil sino que existen otras consideraciones para que esta página sea visitada.

El conocimiento acerca del funcionamiento de cada uno de los Buscadores en Internet, es relevante para obtener una buena posición, y con ello visitas.

### COMERCIO ELECTRÓNICO ORIGEN DEL TÉRMINO COMERCIO ELECTRÓNICO

El término Comercio Electrónico (*Electronic Comerse*, EC) se propuso hace algunos años, refiriéndose a una extensión del término anteriormente acuñado como intercambio electrónico de

---

<sup>84</sup> *Mirror* (espejo, duplicación, réplica): Servidor Web cuyo contenido es una copia exacta de otro servidor o de parte de él. Normalmente este tipo de servidores, cuentan con la aprobación del servidor original y sirven para reducir el tiempo de acceso del usuario a servidores situados en lugares muy distantes.

datos (*Electronic Data Interchange, EDI*), esto, debido a que en las compañías requerían algo más dentro de su estrategia de tecnología de información (*Information Technology, IT*).

El Intercambio electrónico de datos (EDI), es aquella parte de un sistema de información capaz de cooperar con otros sistemas de información mediante el intercambio de mensajes EDI.

Los efectos del comercio electrónico los hemos visto y forman parte de nuestra vida cotidiana, ya que, retirar dinero de un cajero automático o consultar el saldo de una cuenta bancaria por teléfono, son algunos de sus efectos perceptibles; otros menos evidentes son las formas en las que las grandes organizaciones lo han adoptado como una estrategia empresarial para asegurar su permanencia en el mercado.

El crecimiento de Internet y en particular de la *Web*, ha producido una masa crítica de consumidores y empresas que participan en un mercado interactivo global, por lo que, la adopción de Internet como medio comercial, ha provocado que las compañías experimenten formas innovadoras de llegar a sus clientes utilizando la combinación de las tecnologías de computación y de telecomunicaciones, ampliando su función de medio de comunicación al de un nuevo medio de mercado.

### **CONCEPTO**

Podemos decir que, el comercio electrónico es una metodología moderna para hacer negocios que detecta la necesidad de las empresas, comerciantes y consumidores de reducir costos, así como, mejorar la calidad de los bienes y servicios.

Por lo tanto, no debe de seguir contemplándose al comercio electrónico como una tecnología, sino que, es el uso de la tecnología para mejorar la forma de llevar a cabo las actividades empresariales, de tal forma, que resulten transparentes a las personas que lo utilizan dentro de las empresas.

Ahora bien, el comercio electrónico se puede entender como la posibilidad de realizar transacciones comerciales empleando medios electrónicos. La venta en el comercio electrónico se realiza de la misma forma en que se ha desarrollado la venta a través de los tiempos: hay un cliente que necesita un producto ó servicio y un proveedor que lo proporciona; este último, informa sobre todas las condiciones de su oferta y el cliente decide si la misma cubre sus necesidades. Si se llega a un acuerdo, la venta se realiza.

El comercio electrónico tiene múltiples variantes, desde la simple presencia de un catálogo de productos hasta la entrega de la mercancía al consumidor final; puede o no tener interacción con inventarios y sistemas contable administrativos o bien, contar con la posibilidad de que el propio comprador personalice la información que recibe o el producto mismo. De esta manera, aparece un número creciente de formas de realizar las transacciones comerciales, se caracterizan por su evolución a una mayor complejidad de los modelos y a una mayor integración de tareas.

Los modelos más conocidos son las tiendas virtuales (el comerciante lleva el control total de las operaciones y catálogos) y la plaza comercial (varias tiendas aparecen en un dominio común compartiendo infraestructuras y gastos). Sin embargo, existen también modelos de mayor complejidad como los dedicados a licitaciones, subastas y plataformas de colaboración; o bien, los que tienen contemplado un programa de atención al cliente y actualización automática de inventarios.

### ***PROCESOS DEL COMERCIO ELECTRÓNICO***

El comercio electrónico implica la conjunción del mundo físico con el virtual, se puede hablar de varios tipos de comercio electrónico. El primero de ellos, hace referencia a que todo el proceso comercial se realiza en línea; lo anterior, es posible con mercancías que, por su naturaleza, pueden ser transmitidas por Internet (como información, un curso o datos estadísticos). El segundo tipo se establece con productos que necesariamente deben hacerse llegar físicamente al consumidor (como la ropa, un aparato eléctrico o alimentos).

### ***SEGURIDAD***

La seguridad es un punto medular para el comercio electrónico. Por un lado, las partes involucradas deben estar seguras de que el otro es quien dice ser, pues de lo contrario, en el momento de un desacuerdo no sabrán con quién solucionarlo.

El cliente requiere que se le garantice que su información personal no será difundida ni empleada por terceros. Además, necesita garantía de que el producto que se le ofrece es igual al que se le entrega y que, de lo contrario, contará con los instrumentos e instancias jurídicas para hacer valer sus derechos.

El vendedor necesita tener la certeza de que, independientemente de la forma de pago, se le entregará la cantidad acordada por la mercancía y, de no ser así, también tendrá los instrumentos y mecanismos para hacer valer sus derechos.

### ***DIFERENCIA ENTRE COMERCIO ELECTRÓNICO Y NEGOCIO ELECTRÓNICO***

El término negocio electrónico hace referencia al uso de Internet en las diferentes fases del ciclo de vida de una empresa; es decir, incluye tanto la compra-venta de productos, facturación, gestión de la información, atención al cliente, publicidad, intercambio de información entre clientes y proveedores, etc.; en tanto que, el comercio electrónico, únicamente hace referencia a la compra-venta de productos empleando Internet.

### ***LEY DE COMERCIO ELECTRÓNICO***

No existe una ley de Comercio Electrónico en México como tal, es decir, no se trata de un documento único en el que estén incluidos todos los quehaceres jurídicos y normativos en la materia; sin embargo, los aspectos de transacciones por Internet se rigen por disposiciones incluidas en El Código Civil, El Código de Procedimientos Civiles, El Código de Comercio (Federal) y Ley Federal para la Protección del Consumidor.

Desde el 29 de mayo de 2000, cuando se publicó en el Diario Oficial de la Federación un decreto para reformar: El Código Civil, El Código de Procedimientos Civiles, El Código de Comercio (Federal) y Ley Federal para la Protección del Consumidor, se contemplan en todos ellos, aspectos propios de las nuevas tecnologías de la información.

### **CATEGORÍAS**

El comercio electrónico ha dado un importante paso evolutivo con su aplicación en Internet y ha desarrollado en la *Web*, características muy diferentes al tipo de comercio electrónico que se venía desarrollando entre organizaciones. Desde el punto de vista de la generalidad de ambos tipos, el comercio electrónico que se desarrolla en la *Web* es un comercio electrónico abierto y, el que se desarrolla entre organizaciones con el fin de intercambiar datos y documentos, es un comercio electrónico cerrado.

#### **COMERCIO ELECTRÓNICO CERRADO**

- ✓ Entre empresas (*Business to Business, B2B*):  
Se refiere a una empresa que hace uso de una red para hacer ordenes de compra a sus proveedores, recibir facturas y realizar los pagos correspondientes. Esta categoría ha sido utilizada por muchos años, particularmente, haciendo uso del Intercambio Electrónico de Datos (*Electronic Data Interchange, EDI*) sobre redes privadas o redes de valor agregado (*Value added Networks, VAN*).
- ✓ Entre empresa y administración (*Business to Administration, B2A*):  
Se refiere a todas las transacciones llevadas a cabo entre las empresas y las diferentes organizaciones de gobierno. Por ejemplo, para realizar transacciones como el regreso del IVA y el pago de impuestos corporativos.

#### **COMERCIO ELECTRÓNICO ABIERTO**

- ✓ Entre empresa y consumidor (*Business to Consumer, B2C*):  
Este tipo de relación es la más intuitiva, se sustituye por parte de los consumidores la visita a las tiendas físicas. La relación directa entre empresa y consumidor permite la disminución de costo en los productos. Por otra parte, permite utilizar la información que se recaba de los clientes mediante cuestionarios, formularios o el seguimiento remoto de cada recorrido por el sitio, para enfocar a cada cliente hacia intereses particulares y, por lo tanto, aumentar el nivel de satisfacción en cada visita posterior.
- ✓ Entre el ciudadano y la administración (*Consumer to Administration, C2A*):  
Es una nueva modalidad basada en el B2A pero, en este caso, se trata de las transacciones entre ciudadanos y las disposiciones gubernamentales, pudiendo realizar las mismas transacciones de manera electrónica.
- ✓ Entre ciudadanos (*Consumer to Consumer, C2C*):  
Son los mercados electrónicos en los que se venden productos entre ciudadanos particulares; los ejemplos más significativos son las tiendas de subastas. Las posibilidades potenciales de este tipo de comercio electrónico son enormes pues, lo único importante es habilitar buenos mecanismos que, en la medida de lo posible, garanticen la fiabilidad de la oferta y el interés real del comprador. Habitualmente, se pone en contacto a los



consumidores entre si de forma totalmente gratuita en una primera fase y, con el tiempo, se cobra el servicio.

### **MODELOS DE COMERCIO ELECTRÓNICO**

Dentro de los programas integrados de comercio se combinan elementos de varios medios para maximizar la efectividad de un programa de comunicaciones pero, a pesar del gran interés de coordinar esos esfuerzos, no hay una adopción difundida ni una implementación del concepto que permita mostrar una solución única.

El concepto de comercio integrado, mantiene la atracción y la promesa de esfuerzos de negocios dentro de la *Web*, debido a que ofrece un enorme potencial para el desarrollo de las relaciones con el cliente y para personalizar la oferta a los clientes de forma individual.

A continuación, se definen seis categorías funcionales de páginas comerciales dentro de la *Web*, cada una puede considerarse como un elemento en un programa integrado de mercado dentro del contexto del comercio electrónico.

1. Aparador.
2. Presencia.
3. Contenido.
4. Centro comercial.
5. Incentivo.
6. Buscador.

**Aparador:** Estos sitios ofrecen ventas directas mediante un canal que utiliza un catálogo electrónico u otro formato más innovador. Los consumidores piden artículos mediante formularios, teléfonos, registros o correo ordinario. Los aparadores cubren una amplia variedad de ofertas, como por ejemplo libros, flores, discos, entre otros.

Las oportunidades abundan dentro de este modelo, ya que, combina elementos de mercado directo con compras en almacén, y tiene el potencial de ser ampliamente más eficiente que los otros. Dentro de este enfoque, hay grandes oportunidades para la personalización y para el mercadeo de relación.

Dentro de este entorno, es posible una definición más amplia de producto y categorías de servicios. El desarrollo último lo constituyen aquellos productos que pueden existir o ser consumidos únicamente dentro de la *Web*.

Más aún, este modelo plantea ciertos retos como por ejemplo, las velocidades normales de acceso pueden hacer que las compras en línea sean frustrantes y tediosas. Además, el modelo terrestre de compras, necesariamente sienta los estándares de las ofertas interactivas. En muchos casos, las versiones informáticas palidecen en comparación con las experiencias del mundo real al hojear catálogos elegantes o al comprar en grandes almacenes.

**Presencia:** Estos sitios proveen una presencia virtual para una empresa y sus ofertas. Pueden servir también, como señal a los clientes normales y futuros, y a los competidores para que sepan que la empresa está en vanguardia. Existen tres tipos de presencias: anuncio plano, imagen e información.

Los anuncios planos, son anuncios simples de una página sin conexiones con otras. Habitualmente los proveedores de acceso a Internet ofrecen gratuitamente o a un costo simbólico la posibilidad de que las empresas coloquen sus anuncios, en los que realizan una presentación de sus productos y servicios más representativos, así como, una dirección postal y una persona, teléfono y fax como medio para establecer un contacto comercial. Ésta es la mínima presencia empresarial en Internet

Las imágenes son llamadas emocionales al consumidor más que llamadas racionales. La información acerca del producto, si es que la hay, se suministra en el contexto en el que el producto se consume, o tiene algún significado para el consumidor. Estos sitios parecen apropiados para productos que tienen poca información densa en su contenido.

Los sitios de información, proporcionan datos racionales y detallados acerca de la empresa y de sus ofertas. Los sitios de información en Internet pueden tener formas sofisticadas y son más apropiadas para ofrecer contenidos con alto grado de información densa. Pueden utilizarse ayudas en la toma de decisiones que faciliten la navegación. Un objetivo de tales sitios es crear una relación con el cliente, incluso antes de que la necesidad de compra del producto o servicio se presente.

**Contenido:** En los sitios de contenido basados en honorarios, el proveedor paga por el contenido, y el cliente paga por acceso. Se espera que estos sitios aumenten a medida que los mecanismos de seguridad para realizar los pagos se van implementando. Sin embargo, hasta ahora este modelo ha tenido un éxito limitado, quizás porque los usuarios no están dispuestos a pagar por contenidos entregados de esta forma.

Una tendencia reciente es la comisión y precio por el uso de la información, donde los visitantes pueden acceder a resúmenes de noticias sin cargo, pero incurrir en un pequeño pago por el texto completo de una historia.

Los sitios de contenido patrocinado venden espacios de publicidad para reducir o eliminar el cargo de comisiones a los visitantes. Así pues, la publicidad aparece de una gran variedad de fuentes y suscribe el contenido de la editorial. Una tendencia reciente se dirige hacia contenidos patrocinados de entretenimiento y agentes de búsqueda también patrocinados.

Los retos de los sitios de contenido provienen precisamente de la facilidad de implementación. Los paralelos a los medios tradicionales representan obstáculos de ejecución significativa a los sitios de contenido.

**Centro comercial:** El sitio centro comercial constituye típicamente una colección de escaparates interactivos, cada uno de los cuales puede contener diferentes categorías de artículos para su venta. El proveedor carga alquileres a cambio del inmueble virtual y puede ofrecer una serie de servicios al aparador. Algunos centros comerciales aceptan también publicidad.

**Incentivo.** El sitio incentivo presenta una única forma de publicidad que atrae a consumidores potenciales al sitio. El objetivo es atraer al usuario al sitio comercial, ayudando a las personas dedicadas al mercadeo a generar tráficos a sus sitios en la *Web*. El contenido puede ser de naturaleza transitoria y puede parecer que sirve como si fuera un anuncio de servicio público u ofrecer incentivos.

**Buscadores.** El propósito de los buscadores es el de identificar otros sitios de la *Web* mediante claves de búsqueda en una base de datos que se extiende a todo la *Web*. Una tendencia reciente en estos sitios, es la emergencia de agentes de búsquedas basados en honorarios o patrocinados por anunciantes.

Estos seis tipos funcionales proporcionan los bloques para construir los sitios prósperos. Una estrategia integrada debe incluir a todos, aunque con diferente uso.

Dentro de los principales retos para los especialistas de mercado, está el atraer visitantes y generar repetidas visitas significativas. El conocimiento conduce a intentar una visita inicial al sitio, de tal forma que, el problema de tentativa depende de la saturación en el tráfico de la *Web*. Sin embargo, los sitios sólo tendrán éxito a largo plazo si generan tráfico repetido, lo que es mucho más difícil conseguir que el primer intento. El problema de repetir la visita es parcialmente una función del diseño del sitio de la *Web* y depende en gran medida de la necesidad del cliente.

### ***BENEFICIOS QUE OFRECEN A LAS EMPRESAS***

Las actividades comerciales son, información (personas adecuadas con la información apropiada en el momento correcto), privacidad de la información privilegiada, o compartición de la información relevante para varios usuarios.

El continuo avance en el mejoramiento de la infraestructura de las telecomunicaciones ha proporcionado los medios para el intercambio casi instantáneo de los datos. El comercio electrónico hace uso de estos flujos de datos de una forma eficaz.

El intercambio eficiente de la información determina el éxito de una empresa; cuando se logra asegurar que el personal de una compañía consume más tiempo en la producción o venta de productos y servicios, en lugar de reprocesar innecesariamente los datos, podemos decir que ha iniciado la optimización de su potencial.

Internet ha demostrado que estamos viviendo en una comunidad global, esto quiere decir que no importa las diferencias de horario, ni la ubicación geográfica, lo que implica que la actividad empresarial puede realizarse con cualquier persona y en cualquier parte del mundo de manera

instantánea. Esto significa que las empresas deben replantear sus estrategias empresariales, sean cuales fueran sus objetivos: ampliación del mercado, obtención de bienes por parte de proveedores nuevos, externalización de servicios (*outsourcing*<sup>85</sup>), o búsquedas de nuevas alianzas.

El comercio electrónico se introduce en las empresas en tres fases:

1. Sustitución de las operaciones manuales basadas en papel por alternativas electrónicas.
2. Replanteamiento y simplificación de los flujos de información.
3. Uso novedoso y dinámico de los flujos de información.

La sustitución de los sistemas basados en papel, tanto dentro de una empresa, como entre ellas, produce pocos beneficios reales. Es cierto que reduce los costos administrativos y mejora el nivel de precisión en el intercambio de datos pero, no aborda las cuestiones relacionadas con que las actividades comerciales se hagan eficientemente a la primera. Las aplicaciones del comercio electrónico ayudan a reformar la manera de realizar las actividades comerciales y, con frecuencia, actúan como un catalizador para que las empresas utilicen la reingeniería<sup>86</sup> de procesos empresariales (*Business Process Reengineering, BPR*).

## PORTALES

Internet por su propia naturaleza, no tiene una estructura y desde sus orígenes hasta nuestros días, se ha hecho necesario realizar esfuerzos encaminados a facilitar la navegación y la localización de los crecientes recursos existentes en la red.

Algunas personas tuvieron la iniciativa de crear lugares en la red donde se le proporcionara al usuario herramientas para facilitarle la navegación, en otras palabras, para tratar de ponerle orden.

Con lo anterior, surgieron lugares especializados en indexar los contenidos de las páginas existentes en la *Web*. Poco a poco se fueron sofisticando los sistemas de indexación, y algunos de estos lugares dedicados a la búsqueda de información fueron incorporando herramientas como robots.

Posteriormente, se vislumbra la posibilidad de implantar contenidos atractivos y determinados servicios en las páginas para potenciar más el tráfico recibido, creando un vínculo más arraigado con el usuario consiguiendo así, que desde ciertos lugares, muchos usuarios encontraran la mayoría de la información y servicios que habitualmente necesitaban de Internet.

Es en esta situación, cuando debemos empezar a hablar del término Portal.

---

<sup>85</sup> *Outsourcing*: Situación que se da cuando una organización o empresa usa las instalaciones físicas de otra, que le provee, además, mantenimiento y las aplicaciones necesarias para que pueda desarrollar sus gestiones.

<sup>86</sup> *Reingeniería*: Conjunto de actividades tendientes a reformular de manera integral, los procesos organizacionales, administrativos, financieros y contables de una empresa, lo que implica una reconversión, transformación y adaptación a los cambios tecnológicos, y a nuevos modelos estructurales empresariales, con la finalidad de incrementar su productividad, eficiencia y eficacia, procurando su mejora continua y modernización.

## CONCEPTO

Podemos definir portal como un punto de entrada a Internet donde se organizan contenidos, y se concentran servicios y productos, de forma que le permitan al usuario hacer cuanto necesite o, al menos, que pueda encontrar allí todo cuanto utiliza con más frecuencia sin tener que salir de dicho sitio.

El objetivo es, conseguir que los usuarios no usen el portal de forma eventual sino que se habitúen a usarlo a diario, estableciendo algún tipo de vínculo casi personal entre el usuario y el portal.

Así, entre otras cosas, se puede conseguir no sólo que los visitantes coloquen dicha página (el portal) en su *bookmark*<sup>B7</sup>, sino que ésta sea su página de inicio del navegador, lo que garantizará para el sitio un tráfico alto y constante, y le ayudará a asegurar su supervivencia en la red en gran parte, gracias a la publicidad en forma de *banners*<sup>B8</sup> y a los servicios adicionales que se ofrezcan.

## CLASIFICACIÓN

La proliferación del número de portales y de usuarios, así como el vertiginoso crecimiento de Internet, hicieron que pronto surgiera la posibilidad de clasificar los portales en función del público hacia el que van dirigidos, es decir, de acuerdo a la línea de contenidos que pretenden cubrir.

Así, distinguimos entre los portales orientados a toda la población (Portales Generales), los dedicados a usuarios interesados en un tema concreto especializados temáticamente (Portales Especializados) y los destinados a las personas relacionadas con una empresa o institución (Portales Corporativos).

### **PORTALES GENERALES U HORIZONTALES (MEGAPORTALES)**

Un portal de carácter general, orientado a todo tipo de público, ofrece contenidos de carácter muy amplio y tiene como objetivo, cubrir las temáticas más demandadas. Habitualmente, ofrece servicios de valor añadido como comunidades virtuales, espacio *Web* gratuito, información de diversos tópicos, personalización de la información, *chat*, *e-mail* gratuito, mensajes a teléfonos móviles, *software* gratuito, grupos de discusión, comercio electrónico, buscador, etc.

Sin embargo, este modelo empieza a quedarse obsoleto pues ante el nivel tan amplio de cobertura, sólo pueden sobrevivir unos pocos y empieza a haber demasiados portales generales. Los actuales portales de este tipo, están orientados a usuarios principiantes y se basan en servicios poco especializados e inadecuados para los usuarios más expertos, lo que hace necesario redimensionar este mercado a través de la especialización geográfica, temática o corporativa.

---

<sup>B7</sup> *Bookmark (Libro de marcas, marca de página, marca, marcapáginas): Nombre del directorio en el que se guardan las direcciones en Internet de los sitios (URL) que más frecuentemente visita el usuario y que son de mayor interés para él.*

<sup>B8</sup> *Banner (Anuncio, faldón, pancarta, viñeta): Palabra usada en Internet para indicar la publicidad de forma gráfica o sólo texto, que aparecen en las páginas Web.*

### **PORTALES ESPECIALIZADOS**

Cada vez hay más usuarios a los que, ya sea por su grado de experiencia o por sus necesidades profesionales, los portales generales no satisfacen plenamente sus necesidades por tener un contenido demasiado global y, por lo tanto, demasiado superficiales e insuficientes para sus demandas. Ello plantea una situación en la que existe una coyuntura favorable para aplicar el modelo de portal que ya conocemos, a aspectos más específicos que cubran, por ejemplo, un área geográfica determinada, un tema concreto o bien, destinado a cubrir las necesidades de las personas relacionadas con una corporación concreta (empleados, directivos, clientes, proveedores, etc.).

Esta cobertura la ofrecen los denominados portales verticales y los portales corporativos.

### **PORTALES CORPORATIVOS**

Un portal corporativo es una intranet<sup>89</sup> que provee de información de la empresa a los empleados, así como, de acceso a una selección de sitios *Web* públicos y de mercado vertical (proveedores, vendedores, etc.). Incluye un buscador para documentos internos así como la posibilidad de personalizar el portal para diferentes grupos de usuarios y particulares. Sería el equivalente interno a los portales de carácter general de Internet.

Los portales corporativos tienden a ser una prolongación natural de las intranet corporativas (en las que se ha cuidado la organización de la información y la navegación) donde se permite, y sobre todo se potencia, el acceso a información de la propia institución, la edición de material de trabajo propio, el contacto con clientes y proveedores, etc.

En ellos se distingue la parte interna y la externa, dependiendo del tipo de usuario o miembro de la organización.

### **PORTALES VERTICALES (VORTALES)**

Un portal vertical también conocido como Vortal (*Vertical Portal*), es un sitio *Web* que provee de información y servicios a una industria en particular. Es el equivalente industrial específico, de los portales generales de la *Web* pero, en este caso, además de ofrecer los típicos servicios de valor añadido característicos de los portales generales, la cobertura, en cuanto a contenidos, se centra en un tema o sector concreto.

Los portales verticales contienen más datos sobre su mercado que cualquier otro sitio, lo que en sí mismo, ya es un valor añadido considerable siempre manteniendo los datos de la(s) organización(es) que lo integran.

---

<sup>89</sup> *Intranet: Red informática de comunicación privada de una empresa u organización de cualquier tipo que funciona con las mismas tecnologías usadas en la red Internet y que se puede conectar a ella de forma sencilla dada su misma naturaleza tecnológica.*

Mediante este tipo de portales, se puede cubrir las carencias que muestran muchos de los portales generales que pretenden llegar a todo tipo de usuarios pero que, en realidad, no son capaces de cubrir de una forma profunda ninguna de estas áreas temáticas.

### **RETO**

El gran reto del Portal para situarse entre los más importantes y conseguir captar un gran número de usuarios se sintetiza en lo siguiente:

- ✓ **Innovación en los servicios:** Principalmente personalización, no sólo porque quien mejor segmenta a su público consigue mayor eficacia en sus impactos publicitarios, sino porque atraer a un mayor número de usuarios comienza a satisfacer las necesidades personales de cada uno. Esto lleva a la creación de grandes subcanales temáticos moldeables por el usuario, en cuanto a qué y cómo recibir la información del canal. La tecnología lleva a la generalización del uso de Internet y a llevarlo a las personas sin conocimientos informáticos. Este fenómeno conlleva la necesidad de crear niveles de acceso e interactividad, de forma que, dicha generalización no supone una simplificación de la misma que sufran los expertos, ni se convierta en un complejo ejercicio para los menos hábiles tecnológicamente.
- ✓ **Publicidad:** Es el gran objetivo de la concentración, cuanto mayor sea el número de usuarios mayores serán los ingresos por publicidad. Debe adecuarse más el espíritu de servicio, pues hoy se siguen patrones publicitarios tradicionales de simple aparición indiscriminada que pueden producir el mismo rechazo que la publicidad convencional. La posibilidad de segmentar públicos en Internet ofrecida, a través, de la personalización de los portales y la interactividad del medio, sugieren usos mucho más ágiles y eficaces en publicidad.
- ✓ **Seguridad en el comercio electrónico:** Para que la publicidad en Internet tenga aún mayor relevancia como medio de financiamiento de los portales, no basta con que reciban un gran número de visitas, sino que, es necesario que consigan materializar el fin último del esfuerzo comercial: la venta.

### **BENEFICIOS QUE OFRECEN A LAS EMPRESAS**

Una de las cosas más importantes sobre los Portales en la Organización, específicamente hablando, es que son verdaderamente redes de redes. A partir de una interfaz única y muy simple, los usuarios pueden acceder a documentos, programas y datos procedentes de las aplicaciones de la organización. El objetivo es que el dato fluya, no sólo de manera interna a la empresa, sino también por la Extranet de los Proveedores e incluso por la *Web* pública. Los usuarios no tienen por qué saber de donde viene la información.

El sistema tiene que proveer una infraestructura de información y una interfaz que convierta al usuario más inexperto en el más avanzado, los Portales lo pueden conseguir.

La realidad es que, en el futuro, los Portales en la Empresa integrarán elementos impensables hasta hace algún tiempo.

#### **4.2.5.8 FUTURO DE LOS SISTEMAS WEB**

##### **DESARROLLO DE APLICACIONES EN I2**

No hay una línea de trabajo única y predeterminada, que oriente las investigaciones de las nuevas posibilidades de aplicaciones desarrolladas en I2. Aún hay mucho para ser investigado sobre la necesidad de los usuarios y el potencial de las tecnologías para redes de alto desempeño. De una forma general, no se conoce aún el límite de lo que es técnicamente posible. Se puede decir entonces, que el foco principal de I2 reside en el desarrollo de aplicaciones avanzadas con uso intensivo de tecnologías multimedia en tiempo real.

En este momento, algunas de las principales líneas de investigación desarrolladas para la aplicación de servicios en red de alto desempeño son:

##### ***BIBLIOTECAS DIGITALES***

Una Biblioteca digital es una colección de información que se guarda y accede electrónicamente. Toda la información contenida en la Biblioteca debe referirse a un tema común. Es decir, una Biblioteca digital puede diseñarse para gráficas por computadora, sistemas operativos, redes o cualquier otro tema, todas estas librerías separadas pueden combinarse bajo una misma interfaz, pero es esencial que la información contenida dentro de cada una se mantenga separada.

##### ***LABORATORIOS VIRTUALES***

Un laboratorio virtual es un ambiente heterogéneo distribuido que permite a un grupo de investigadores, situados alrededor del mundo, trabajar juntos en un conjunto común de proyectos. Como en cualquier laboratorio, las herramientas y las técnicas son específicas al dominio de la investigación, pero los requisitos básicos de infraestructura se comparten a través de las disciplinas.

##### ***TELEINMERSIÓN***

Es una de las aplicaciones que ha motivado la creación de I2 por sus altos requerimientos de ancho de banda. Un sistema de teleinmersión, permite a individuos de diferentes localidades compartir un mismo ambiente virtual. Por ejemplo, los participantes podrán interactuar con otras personas en una sala de conferencias virtual, de manera similar a lo que sucedería en un cuarto físico. Los individuos podrán compartir y manipular datos, simulaciones y modelos de una estructura molecular, una construcción física o económica y, conjuntamente, participar en la simulación, diseño o evaluación de un proceso.

##### ***NANO MANIPULADOR DISTRIBUIDO***

Es una interfaz desarrollada con técnicas de realidad virtual que permite a químicos, biólogos y físicos observar a distancia, la superficie de un material a escala atómica y sentir su textura, a través, de una palanca que cuenta con un mecanismo de retroalimentación que va dando la sensación de la textura del material a la persona que está manipulándola mientras se observa a través del monitor el desplazamiento que se va realizando. Utilizando esta misma palanca y oprimiendo un botón, puede también modificarse la superficie. Este sistema permite tener laboratorios en los cuales, pueden colaborar equipos de científicos geográficamente distantes.



## **TELEMEDICINA**

Es el uso de las telecomunicaciones para el diagnóstico médico y la atención del paciente. Implica el uso de la tecnología de las comunicaciones como un medio para llevar servicios médicos a lugares remotos.

El concepto es extenso, ya que, comprende desde el uso del servicio telefónico estándar, hasta la transmisión a alta velocidad y con amplio ancho de banda de señales digitalizadas; utilizando computadoras, fibra óptica, satélites y otros dispositivos, además de *software* sofisticado.

Se puede dividir la Telemedicina en tres áreas:

1. Ayuda para el diagnóstico: Como ayuda en el proceso de toma de decisiones, la telemedicina incluye áreas tales como los sistemas expertos a distancia, que contribuyen al diagnóstico del paciente o el uso de bases de datos en línea. Este es el uso más antiguo de la telemedicina.
2. Control a distancia: Consiste en la transmisión de información del paciente (radiografías, datos clínicos, bioquímicos, etc.). Este es el uso más actual.
3. Colaboración en tiempo real para el manejo del paciente a distancia: Permite a un médico, observar y discutir los síntomas de un paciente que está siendo asistido por otro médico a distancia. Este es el uso del futuro, cuando se normaticen ciertos aspectos tales como, responsabilidad, acreditación, formas de pago, etc.

### **PROYECCIÓN DE PANTALLAS DE COMPUTADORAS EN TRES DIMENSIONES**

Esto se logra, a través de la utilización de *ImmersaDesk* (especie de gran pantalla de TV que proyecta las imágenes en 3D).

El *Immersadesk* es una computadora de realidad virtual desarrollada en el *Electronic Visualization Laboratory* de la Universidad de Illinois de Chicago; cuenta con un par de pantallas a través de lentes especiales la gente se interactúa en un ambiente virtual, puede ser utilizado hasta por cinco usuarios, portando anteojos obturados para ver en alta resolución. Este sigue los movimientos de la cabeza del usuario para generarle una perspectiva exacta del ambiente virtual en el que está inmerso.

### **4.2.5.9 POR QUÉ TODO TIENDE A INTERNET**

Hoy en día, Internet se ha convertido en una herramienta muy importante en el mundo de los negocios y está siendo usado por personas, empresas y negociantes en todas partes del mundo, en cientos de formas diferentes que van, desde la simple publicación de un curriculum vitae, hasta las mega-corporaciones y las empresas basadas exclusivamente en la *Web*.

Los servicios en línea son cada vez más crecientes y existe un verdadero impacto que mucho más allá de ser una simple moda, se está convirtiendo en una parte integrante de la composición de cualquier empresa.

Las siguientes, son algunas de las razones por las que una empresa está en Internet:

- ✓ Proveer acceso instantáneo a catálogos y otras informaciones de la empresa, ahorrando costos de impresión. La *Web* es extremadamente eficiente para distribuir información. Con un solo archivo que sea creado, pueden acceder a él, millones de personas en el mundo entero.
- ✓ Mantener a los clientes al día acerca de nuevos productos o servicios.
- ✓ Hacer más y mejores contactos con proveedores y clientes de cualquier parte del mundo. Cualquier fabricante o distribuidor puede obtener información de la empresa al instante, y comunicarse con el personal correspondiente de la empresa de manera inmediata. Actualmente, no existe una forma más económica y eficiente de comunicación.

Las empresas que han desarrollado una estrategia en Internet descubren que ésta les ayuda a reducir sus costos significativamente. Por ejemplo:

- ✓ Costos de comunicación global: La *Web* permite dar información detallada a cualquier persona interesada que se encuentre en cualquier parte del mundo durante las 24 horas y los siete días de la semana.
- ✓ Costos de gestión de la información: La burocracia se reduce y facilita la organización de la información. También disminuyen los errores, se agilizan los procesos y se facilita la gestión de los documentos.
- ✓ Costos de personal: Se pueden automatizar y simplificar aquellas tareas más repetitivas y de menor valor añadido (toma de pedidos, facturación, información, etc.), liberando recursos para otras actividades.
- ✓ Costos de producción: El inventario de productos acabados y materias primas disminuye, porque aumenta la producción bajo pedido. Las empresas tienen, también, mayor facilidad para concentrarse en sus competencias esenciales y subcontratar todas aquellas actividades en las que no están especializadas.

Ante el veloz cambio de la información, las empresas que están presentes on Internet, requieren de una toma de decisiones realizada a mayor velocidad.

#### **4.2.6 METODOLOGÍAS DE DESARROLLO DE SISTEMAS**

Las metodologías de desarrollo de sistemas son un conjunto de procedimientos, técnicas y ayudas a la documentación, para el desarrollo de productos *software*. Es una especie de libro de recetas de cocina, en el que se van indicando paso a paso todas las actividades a realizar para lograr el producto (*software*) deseado, se debe indicar qué personas deben participar en el desarrollo de las actividades y su papel. Por otra parte, detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzar dicha actividad.

Las técnicas indican cómo debe ser realizada una actividad determinada identificada en la metodología. Combinan el empleo de modelos o representaciones gráficas junto con el empleo de procedimientos detallados. Se debe tener en consideración, que una técnica determinada puede ser utilizada en una o más actividades de la metodología de desarrollo de *software*.

Las dos metodologías tradicionales son:

#### 4.2.6.1 ANÁLISIS MODERNO ESTRUCTURADO, DE EDWARD YOURDON

Es una de las metodologías que enfatiza la importancia de la integración coherente de modelos gráficos en el diseño de sistemas. Los modelos usados son:

Modelo de Ambiente: Define las fronteras del sistema y sus actores externos.

Componentes:

- ✓ Diagrama de Contexto.
- ✓ Lista de eventos.

Modelo de Comportamiento: Especifica el modelo estático, dinámico y funcional del sistema

Componentes:

- ✓ Diagrama Entidad Relación.
- ✓ Diagramas de Contexto de cada transacción.
- ✓ Diagrama Transición Estado.

Modelo de Implementación: Descripción detallada de cada transacción. Pueden ser usadas otras notaciones para especificación detallada, tales como Diagramas de navegación, Diagramas de Transición Estado.

Componentes:

- ✓ Diagramas de Flujo de Datos.
- ✓ Diagramas de Estructura.
- ✓ Diagramas de Navegación.
- ✓ Diagramas de Acción.
- ✓ Diagramas de Transición Estado.

#### 4.2.6.2 METODOLOGÍA DE DESARROLLO ORIENTADO A OBJETOS

La orientación a objetos es una manera de pensar, otra manera de resolver un problema; lo más reciente en metodologías de desarrollo de *software*. Es un proceso mental humano aterrizado en una computadora. Anteriormente, se adecuaba al usuario al entendimiento de la computadora, ahora, se le enseña a la computadora a entender el problema.

La Orientación a Objetos es un paradigma, es decir, es un modelo para aclarar algo o para explicarlo. La Orientación a Objetos es el paradigma que mejora el diseño, desarrollo y mantenimiento del *software* ofreciendo una solución a largo plazo a los problemas y preocupaciones que han existido desde el comienzo del desarrollo del *software*. La falta de portabilidad del código, su reusabilidad, la modificación (que antes era difícil de lograr), ciclos de desarrollo largo, técnicas de programación no intuitivas.

La Orientación a Objetos está basada en los tres métodos de organización que utilizamos desde la infancia; entre un objeto y sus atributos (automóvil > marca, color, número de llantas, etc.). Entre un objeto y sus componentes donde incluso otros objetos pueden formar parte de otros objetos (agregación) (camión > motor, parabrisas, llantas); entre un objeto y su relación con otros objetos (camión > vehículos automotores; una bicicleta no entraría en esta relación).

De manera resumida consiste en:

- ✓ Saber el espacio del problema
- ✓ Realizar una abstracción
- ✓ Crear los objetos (espacio de la solución)
- ✓ Instanciarlos (esto es, traerlos a la vida)
- ✓ Dejarlos vivir (ellos ya saben lo que tienen que hacer)

Existen diferentes modelos de diseño orientados a objetos, y la mayoría utilizan los siguientes modelos

#### Modelo de Objetos

- ✓ Diagrama de objetos: Muestra las clases del sistema (con atributos y métodos) y las relaciones estáticas entre ellos (agregación y herencia).

#### Modelo Dinámico

- ✓ Diagramas de Escenarios: Muestran las secuencias de interacción típicas o más comunes.
- ✓ Diagramas de Flujo de Eventos: Resumen los eventos y los intercambios entre las clases.
- ✓ Diagramas de Estado: Especifican el comportamiento dinámico para una clase.

#### Modelo Funcional

- ✓ Diagrama de Entradas y Salidas: Muestra las entradas y salidas del sistema.
- ✓ Diagramas de Colaboración: Muestran el intercambio de mensajes entre objetos.

## 5. CASO ESPECÍFICO: "AVISO DE OCASIÓN DEL PORTAL UNAM"

### 5.1 Introducción

La Dirección General de Servicios de Cómputo Académico (DGSCA), es el organismo de la UNAM encargado de impulsar la cultura informática ofreciendo variados servicios en cómputo y telecomunicaciones, así como, fomentar la integración de estas nuevas tecnologías en todos los sectores de la sociedad, especialmente en la educación y la empresa, a través, de asesorías, capacitación y desarrollo de sistemas, entre otros.

La DGSCA se encuentra integrada por las siguientes dependencias:

- ✓ Dirección de Cómputo para la Investigación.
- ✓ Dirección de Cómputo para la Docencia.
- ✓ Dirección de Sistemas.
- ✓ Dirección de Telecomunicaciones Digitales.
- ✓ Subdirección de la DGSCA.
- ✓ Jefatura de la Unidad Administrativa.

Específicamente hablando, dentro de la Subdirección de la DGSCA se encuentra la Coordinación de Servicios de Red (CSR), encargada de administrar y desarrollar los Servicios Institucionales de la UNAM; evaluar las herramientas de *software* y *hardware* que contribuyan al óptimo aprovechamiento de los servicios existentes, y considera la factibilidad de incorporación de nuevas herramientas y servicios, basándose en los recursos, tanto humanos como materiales, disponibles para ello.

Por este motivo, la CSR no sólo se limita a la supervisión y mejoramiento de los servicios a su cargo, sino que también, proporciona asesoría, soporte y documentación especializada hacia todos aquellos usuarios (particulares e institucionales) que deseen incorporar y/o extender dichos servicios a sus respectivas áreas de competencia.

Dentro de los proyectos que la CSR tiene a su cargo se encuentra, el mantener la presencia en Internet del sitio *Web* de la UNAM a través de <http://www.unam.mx> (Portal UNAM).

Los Portales Educativos, un tipo de Portal Vertical, como en el caso del Portal UNAM, son espacios *Web* que ofrecen múltiples servicios a los miembros de la comunidad educativa.

La mayoría de los portales son gratuitos y constituyen una cortesía de las instituciones educativas para sus usuarios. De esta manera, mediante la oferta de servicios de interés para los miembros de la comunidad educativa, pretenden difundir una buena imagen institucional y/o contactar con clientes potenciales.

Algunas de las ventajas que proporcionan los portales educativos a sus usuarios son:

- ✓ Ofrecen instrumentos para realizar distintos tipos de búsqueda.
- ✓ Permiten el acceso a múltiples recursos didácticos de manera gratuita.

- ✓ Proporcionan recursos para la formación continua del profesorado.
- ✓ Dan diversos tipos de asesoramiento.
- ✓ Abren nuevos canales de comunicación.
- ✓ Brindan instrumentos para la comunicación.
- ✓ Permiten acceder a recursos culturales.
- ✓ Permiten acceder a recursos para el entretenimiento.

Cabe destacar que dichas ventajas, van de acuerdo a los servicios que cada portal ofrece, y a la accesibilidad desde Internet en cualquier momento y lugar.

En abril del año 2000 se liberó el Portal UNAM, incorporando desde entonces y hasta la fecha, nuevos servicios, como son:

- ✓ Tipos de búsqueda: Sistema de Búsqueda General de la UNAM<sup>90</sup>, Sabueso<sup>91</sup>, Mapa de CU<sup>92</sup>.
- ✓ Recursos didácticos gratuitos: Publica tu obra<sup>93</sup>.
- ✓ Recursos para la formación continua del profesorado: SER-UNAM<sup>94</sup>.
- ✓ Diversos tipos de asesoramiento: Becas<sup>95</sup>, Centro de Orientación Educativa<sup>96</sup>.
- ✓ Nuevos canales de comunicación: (ejemplo en el Portal de la UNAM: Foros de Discusión en Línea<sup>97</sup>, Chat Paidoteca<sup>98</sup>)
- ✓ Instrumentos para la comunicación: Correo UNAM<sup>99</sup>.
- ✓ Recursos culturales: Cartelera Cultural<sup>100</sup>, Coordinación de Difusión Cultural<sup>101</sup>.

<sup>90</sup> Se encuentra localizado en la página principal del Portal de la UNAM y es un sistema que realiza búsquedas en todas las páginas localizadas en el dominio de la UNAM.

<sup>91</sup> Este servicio es un directorio en línea que contiene las direcciones de correo electrónico de personas en México y/o el mundo que se registren, a fin de ser localizadas desde cualquier lugar en Internet <http://www.unam.mx/sabueso>

<sup>92</sup> Servicio de localización de edificios en el campus de CU <http://www.mapa.unam.mx/>

<sup>93</sup> Es un servicio que ofrece la publicación y consulta de textos, de temas en general de manera gratuita <http://www.tuobra.unam.mx/cgi-bin/vercero.cgi>

<sup>94</sup> SER-UNAM es un programa destinado a fomentar y apoyar el trabajo participativo, la investigación y la vinculación entre entidades académicas mediante la incorporación de las nuevas tecnologías en los procesos educativos <http://www.ser.unam.mx>

<sup>95</sup> Este servicio ofrece orientación de los distintos programas de becas de la UNAM <http://www.unam.mx/becas/becas.htm>

<sup>96</sup> Se ofrece asesoría a los universitarios sobre diversos servicios de su interés <http://www.unam.mx/dov/INDEXO3.HTML>

<sup>97</sup> Espacio que tiene como objetivo propiciar la libre y responsable manifestación de ideas entre la comunidad universitaria <http://www.unam.mx/foros/foros.htm>

<sup>98</sup> Espacio en el que se puede conversar e intercambiar opiniones en línea con especialistas, profesionales y conocedores de diferentes temáticas de interés <http://paidoteca.dgsca.unam.mx/chat/>

<sup>99</sup> Este servicio es proporcionado por la Dirección de Telecomunicaciones de la Dirección General de Servicios de Cómputo Académico de la UNAM y, está dirigido, a toda la comunidad universitaria <http://www.correo.unam.mx/>

<sup>100</sup> Aquí se ofrece la calendarización de eventos culturales <http://www.agendacultural.unam.mx/>

- ✓ Recursos para el entretenimiento: Dirección General de Actividades Deportivas y Recreativas<sup>102</sup>, Ludotecas<sup>103</sup>.

El principal objetivo del Portal UNAM es servir a la comunidad universitaria, sin embargo, gracias a la enorme labor que la Universidad realiza en beneficio de la sociedad, es también de utilidad para todo el público en general.

Por otra parte, el Portal UNAM es punto de partida para el acceso a la información de las diferentes escuelas, facultades, centros e institutos que conforman esta Universidad, así como, de otras instituciones de educación superior. Siendo sus principales productos:

**Oferta educativa:** Se proporciona toda la información acerca de las carreras, planes de estudio y programas de posgrado que se ofrecen en la Universidad, incluyendo los cursos que se brindan en la modalidad de educación a distancia. Los estudiantes pueden realizar cursos diplomados, licenciaturas (Relaciones Internacionales, Contaduría, Administración, Derecho, Economía y Enfermería), así como, ciertas asignaturas y otros cursos. Hoy en día, es posible llevar a cabo trámites como la inscripción a licenciatura en algunas facultades, asesorías individualizadas por correo electrónico. Por otra parte, la UNAM coordina la Red Interamericana de Formación en Educación y Telemática de la Organización Universitaria Interamericana, cuyo proyecto principal, es el establecimiento de una Maestría en Educación y Telemática con alcance Interamericano, la cual, se encuentra en fase piloto con el curso en línea Estrategia y toma de decisiones para la educación a distancia.

**Biblioteca digital:** Las bibliotecas, sitios obligados para cualquier estudiante, tienen la ventaja, al estar en línea, de facilitar la búsqueda de información y ahorrar al usuario la tarea de desplazarse físicamente al lugar. Cuenta con una biblioteca digital diseñada especialmente para tal fin que contiene libros y revistas como la Revista Digital Universitaria, acceso a museos en México y periódicos en el mundo entero.

**Información general de la propia Universidad:** Acerca de su historia, insignia, rectores, estadísticas, trámites que se pueden realizar vía fax o por teléfono, el estudio de idiomas, los requisitos para otorgar becas, las actividades deportivas y recreativas, consulta a la bolsa universitaria de trabajo y servicio social, entre otros.

**Videoconferencias interactivas:** Permiten a los usuarios asistir a conferencias impartidas en lugares remotos, e intercambiar opiniones entre el conferencista y los asistentes desde cualquier lugar que se conecte a la sede. La capacidad instalada es de 2,500 horas por mes y 22,000 horas/aula de actividades por este medio.

---

<sup>101</sup> Se conjuntan los sitios Web de organismos de la UNAM encaminados a la cultura, así como, los principales eventos culturales próximos a efectuarse <http://difusion.cultural.unam.mx/>

<sup>102</sup> En esta página se encuentran diversas opciones que ofrece la UNAM para la práctica deportiva y la recreación, servicios médicos deportivos y próximos eventos de igual índole <http://www.unam.mx/dgadyr/>

<sup>103</sup> Servicio recreativo que, como opción para el uso del tiempo libre, ofrece la UNAM.

Global English: A través de este servicio se imparten clases de inglés, mediante 10 cursos que se ofrecen de manera gratuita. Aquí se cuenta, entre otras cosas, con un centro de conversación con miembros de todas partes del mundo; con diccionarios, libros de gramática y material de lectura; y con profesores en línea las 24 horas del día, los 365 días del año.

SEPAcómputo: Es un programa de educación continua a distancia para la difusión y enseñanza del cómputo y las telecomunicaciones, que ofrece materiales de apoyo en línea y es transmitido por el canal 22 y el canal 13 de Edusat.

Comercio electrónico: En la tienda virtual se pueden adquirir, de manera segura, artículos universitarios como libros, manuales, discos compactos, etc.

Noticias: Información sobre México y el mundo.

Acceso a otros sitios: Se clasifican otros sitios por especialidad, por temas y de forma alfabética para que el usuario los localice rápidamente sin tener que desplazarse a otros sitios en su búsqueda.

También ofrece servicios de base como son: buscadores eficientes, correo electrónico gratuito, el pronóstico del tiempo, concursos mediante trivias, encuestas, el aviso de ocasión, entre otros.

El Aviso de Ocasión es el servicio que desarrollamos como sistema para el caso específico de la presente tesis.

A continuación, se presenta la documentación correspondiente al desarrollo del sistema Aviso de Ocasión del Portal UNAM.

## **5.2 ANÁLISIS**

### **NOMBRE DEL PROYECTO**

Aviso de Ocasión del Portal UNAM

### **DESCRIPCIÓN DEL PROBLEMA**

En la UNAM, existen miles de sitios que la comunidad universitaria utiliza como medio de difusión, para ofrecer productos y servicios utilizando carteles; sin embargo, muchos de estos no tienen éxito. Lo anterior, se debe a que sólo un número reducido de gente ve o se entera de dichos carteles, por ejemplo, si un estudiante de la Facultad de Psicología quiere vender una computadora, únicamente tienen conocimiento de ello la población de la propia facultad, dejando a un lado, posibles compradores. En resumen, esta situación está sujeta a limitaciones geográficas y, como consecuencia, de difusión.



## DESCRIPCIÓN GENERAL DEL SISTEMA

El Aviso de Ocasión del Portal UNAM, es un sistema que llevará a cabo la automatización de los carteles de productos y servicios que ofrece la comunidad universitaria.

## OBJETIVO GENERAL

Concentrar la información de productos y/o servicios que ofrece la comunidad universitaria, así como, el control de los usuarios y de sus anuncios.

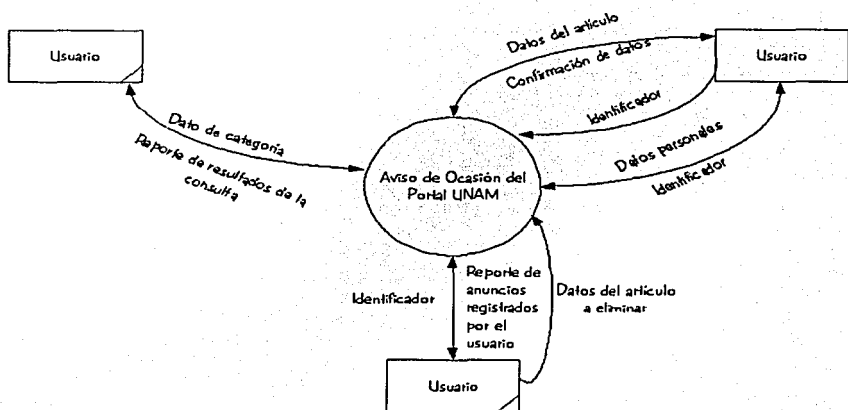
## ALCANCES

- ✓ Registrar usuarios.
- ✓ Registrar anuncios.
- ✓ Almacenar la información correspondiente a cada uno de los usuarios.
- ✓ Almacenar la información correspondiente a cada uno de los anuncios de usuarios.
- ✓ Gestionar la información de los anuncios almacenados.
- ✓ Gestionar la información de los usuarios.
- ✓ Generar de los siguientes reportes:
  - Concentrado de anuncios por día.
  - Concentrado de anuncios por mes.
  - Concentrado de anuncios por categoría.
  - Reporte de anuncios registrados por el usuario.
  - Reporte de resultados de la consulta.
- ✓ Brindar información actualizada de los anuncios a la comunidad universitaria.
- ✓ Brindar un espacio de difusión a todas las instituciones, centros, facultades y escuelas que pertenecen a la UNAM.
- ✓ Brindar un canal de comunicación a la comunidad universitaria.

## RESTRICCIONES

- ✓ El sistema no es una forma de hacer negocio comercial.
- ✓ La estancia de un anuncio en la *Web* no es por tiempo ilimitado.
- ✓ El sistema no garantiza la veracidad de los artículos o servicios publicados.
- ✓ El sistema no comunica de forma directa al vendedor con el comprador.

## DIAGRAMA DE CONTEXTO



## ESPECIFICACIONES DE FLUJOS DE DATOS

Datos personales. Contiene los siguientes datos:

- + Nombre o nombres del usuario.
- + Apellidos
- + Teléfono
- + R.F.C.
- + Dirección
- + E-mail
- + Sexo [Masculino | Femenino]
- + Ocupación [Estudiante UNAM | Trabajador UNAM | Académico UNAM | Otro]
- + Nivel de estudios [Bachillerato | Licenciatura | Posgrado | Otro]

Identificador. Es una palabra única para cada usuario, la cual, está formada por el RFC del usuario más un número aleatorio.

Datos del artículo. Contiene los siguientes datos:

- + Artículo (nombre, raza o marca del artículo)
- + Descripción breve del artículo
- + Precio del artículo
- + Fotografía del artículo

Confirmación de datos. Contiene los siguientes datos:

- + Artículo (nombre, raza o marca del artículo)
- + Descripción breve del artículo
- + Precio del artículo
- + Fotografía del artículo
- + Fecha de registro del artículo

Dato de categoría. Contiene los siguientes datos:

- + Categoría [Vehículos | Artículos Escolares | Música | Equipo de Cómputo | Enseñanza | Mascotas | Alojamiento | Diversos]

Reporte de resultados de la consulta. Contiene los siguientes datos referentes, a cada uno de los anuncios registrados, de acuerdo a la categoría seleccionada:

- + Artículo (nombre, raza o marca del artículo)
- + Descripción breve del artículo
- + Precio del artículo
- + Fotografía del artículo
- + Nombre completo del usuario que registro el anuncio
- + Teléfono
- + E-mail

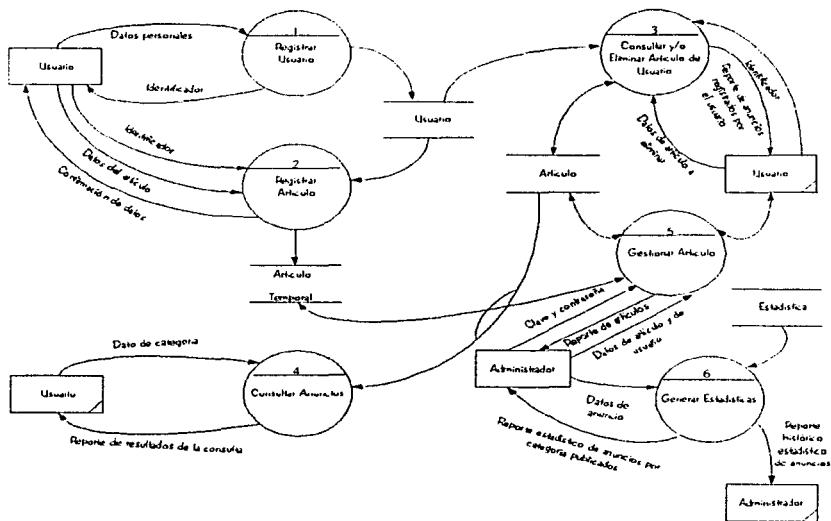
Reporte de anuncios registrados por el usuario. Contiene los siguientes datos referentes, a cada uno de los anuncios registrados por el usuario:

- + Artículo (nombre, raza o marca del artículo)
- + Descripción breve del artículo
- + Precio del artículo
- + Fotografía del artículo

Datos de artículo a eliminar. Contiene los siguientes datos:

- + Artículo (nombre, raza o marca del artículo)
- + Descripción breve del artículo
- + Precio del artículo
- + Fotografía del artículo

## DIAGRAMA DE FLUJO DE DATOS DE PRIMER NIVEL



### MINIESPECIFICACIONES

Nombre del proceso 1: Registrar Usuario.

Participantes: Usuario.

Descripción: Este proceso tiene como fin, introducir los datos del usuario al sistema mediante la captura de estos, misma que será hecha por el propio usuario.

Entrada: Datos personales.

Salida: Identificador.

Nombre del proceso 2: Registrar Artículo.

Participantes: Usuario.

Descripción: Este proceso tiene como fin, introducir los datos del artículo al sistema mediante la captura de estos, misma que será hecha por el propio usuario.

Entrada: Identificador y Datos de artículo.

Salida: Confirmación de datos.

Nombre del proceso 3: Consultar y/o Eliminar artículos de Usuario.

Participantes: Usuario.

Descripción: Este proceso permite al usuario, consultar sus anuncios registrados y poder eliminarlos del sistema, si así lo desea.

Entrada: Identificador, Datos de artículo a eliminar.

Salida: Reporte de anuncios registrados por el usuario.

Nombre del proceso 4: Consultar Anuncios.

Participantes: Usuario

Descripción: Este proceso permite al usuario consultar todos los anuncios registrados en el sistema.

Entrada: Dato de categoría.

Salida: Reporte de resultados de la consulta.

Nombre del proceso 5: Gestionar Artículo.

Participantes: Administrador.

Descripción: Este proceso permite al administrador seleccionar los anuncios a publicar, así como, revisar los reportes estadísticos.

Entrada: Clave y contraseña, Datos de artículo y de usuario.

Salida: Reporte de artículos.

Nombre del proceso 6: Generar Estadísticas.

Participantes: Administrador.

Descripción: Este proceso permite al administrador, revisar los reportes estadísticos por categoría de anuncios publicados y por el histórico de anuncios.

Entrada: Clave y contraseña, Datos de artículo.

Salida: Reporte Estadístico de anuncios publicados por categoría y Reporte Histórico Estadístico de anuncios.

El Reporte Estadístico de anuncios publicados por categoría, contiene los datos:

- + Nombre de la categoría
- + Total de avisos publicados
- + Avisos publicados hasta la fecha
- + Identificador

El Reporte Histórico Estadístico de anuncios, contiene los datos:

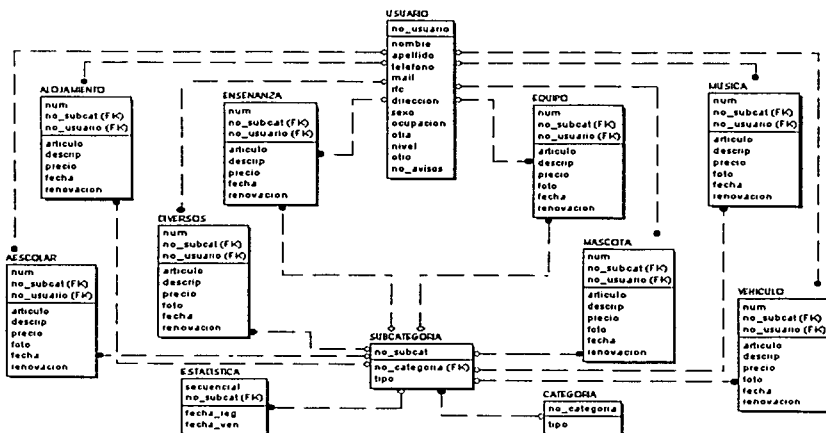
- + Nombre de la categoría
- + Nombre de subcategoría
- + Porcentaje de anuncios correspondientes a la subcategoría

## **ROLES**

**Usuario:** Puede ser cualquier persona que esté interesada en publicar y/o consultar anuncios.

**Administrador:** Es la persona encargada de revisar cada uno de los anuncios y decidir, cuáles se publican y cuáles se eliminan. Además, atiende los comentarios de los usuarios.

## DIAGRAMA ENTIDAD RELACIÓN



## DESCRIPCIÓN DE TABLAS

### USUARIO

Esta tabla contiene atributos que pertenecen al usuario del sistema, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

- Número de usuario
- Nombre(s)
- Apellidos
- Teléfono
- R.F.C.
- Dirección
- E-mail
- Sexo
- Ocupación
- Nivel de estudios

### ASESCOLAR

Esta tabla contiene atributos que pertenecen a los artículos existentes en la categoría Artículos Escolares, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

- Número

Número de subcategoría  
Número de usuario  
Artículo  
Descripción  
Foto  
Fecha  
Renovación

### **ALOJAMIENTO**

Esta tabla contiene atributos que pertenecen a los artículos existentes en la categoría Alojamiento, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número  
Número de subcategoría  
Número de usuario  
Artículo  
Descripción  
Fecha  
Renovación

### **DIVERSOS**

Esta tabla contiene atributos que pertenecen los artículos existentes en la categoría Diversos, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número  
Número de subcategoría  
Número de usuario  
Artículo  
Descripción  
Foto  
Fecha  
Renovación

### **ENSEÑANZA**

Esta tabla contiene atributos que pertenecen a los servicios existentes en la categoría Enseñanza, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número  
Número de subcategoría  
Número de usuario  
Artículo  
Descripción  
Fecha  
Renovación

### **EQUIPO**

Esta tabla contiene atributos que pertenecen a los artículos existentes en la categoría Equipo de Cómputo, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número

Número de subcategoría

Número de usuario

Artículo

Descripción

Foto

Fecha

Renovación

### **MASCOTA**

Esta tabla contiene atributos que pertenecen a los artículos existentes en la categoría Mascotas, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número

Número de subcategoría

Número de usuario

Artículo

Descripción

Foto

Fecha

Renovación

### **MUSICA**

Esta tabla contiene atributos que pertenecen a los artículos existentes en la categoría Música, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número

Número de subcategoría

Número de usuario

Artículo

Descripción

Foto

Fecha

Renovación

### **VEHICULO**

Esta tabla contiene atributos que pertenecen a los artículos existentes en la categoría Vehículos, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número

Número de subcategoría

Número de usuario



Artículo  
Descripción  
Foto  
Fecha  
Renovación

### **SUBCATEGORIA**

Esta tabla contiene atributos que pertenecen a las subcategorías existentes, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número de subcategoría  
Número de categoría  
Tipo

### **CATEGORIA**

Esta tabla contiene atributos que pertenecen a las categorías existentes, los datos han sido seleccionados tomando en cuenta la información importante para el sistema y son:

Número de categoría  
Tipo

### **ESTADISTICA**

Esta tabla contiene atributos necesarios para generar estadísticas y son:

Número de subcategoría  
Secuencial  
Fecha de registro  
Fecha de vigencia

## DICCIONARIO DE DATOS LÓGICO

Tabla	USUARIO
Campos	@no_usuario + nombre + apellido + telefono + mail + rfc + direccion + sexo + ocupacion + otra + nivel + otro + no_avisos
Descripción de Campos	<p>no_usuario = "Número identificador de usuario" {carácter legal   entero}13</p> <p>nombre = "Nombre del usuario" {carácter legal}30</p> <p>apellido = "Apellido del usuario" {carácter legal}40</p> <p>telefono = "Teléfono del usuario" {carácter legal   entero}20</p> <p>mail = "Dirección de correo electrónico del usuario" {carácter legal   entero}29</p> <p>rfc = "R.F.C. del usuario" {carácter legal   entero}10</p> <p>direccion = "Dirección del usuario" {carácter legal   entero}150</p> <p>sexo = "Género del usuario" {carácter legal}1</p> <p>ocupacion = "Ocupación del usuario" {carácter legal}20</p> <p>otra = "Ocupación especificada por el usuario" {carácter legal}30</p> <p>nivel = "Nivel académico del usuario" {carácter legal}15</p> <p>otro = "Nivel académico especificado por del usuario" {carácter legal}30</p> <p>no_avisos = "Número de avisos publicados del usuario" {entero}2</p>

Tabla	CATEGORIA
Campos	@no_categoria + tipo
Descripción de Campos	<p>no_categoria = "Número de categoría" {entero}4</p> <p>tipo = "Tipo de categoría" {carácter legal}20</p>

Tabla SUBCATEGORIA

Campos @no\_subcat + no\_categoria + tipo

Descripción de Campos  
 no\_subcat = 'Número de subcategoría'  
 (entero)4  
 no\_categoria = 'Número de categoría'  
 (entero)4  
 tipo = 'Tipo de subcategoría'  
 (carácter legal)25

Tabla ESTADISTICA

Campos @no\_subcat + secuencial + fecha\_reg + fecha\_ven

Descripción de Campos  
 no\_subcat = 'Número de subcategoría'  
 (entero)4  
 secuencial = 'Número consecutivo de anuncios'  
 (entero)4  
 fecha\_reg = 'Fecha de registro del anuncio'  
 (carácter legal | entero)9  
 fecha\_ven = 'Fecha de vencimiento del anuncio'  
 (carácter legal | entero)9

Tabla AESCOLAR

Campos @num + no\_subcat + no\_usuario + articulo + descrip + precio + foto + fecha + renovacion

Descripción de Campos  
 num = 'Número consecutivo del artículo'  
 (entero)4  
 no\_subcat = 'Número de subcategoría del artículo'  
 (entero)4  
 no\_usuario = 'Número identificador del usuario'  
 (carácter legal | entero)13  
 articulo = 'Nombre del artículo'  
 (carácter legal | entero)20  
 descrip = 'Descripción del artículo'  
 (carácter legal | entero)150  
 precio = 'Precio del artículo'  
 (carácter legal | entero)15  
 foto = 'Nombre de la foto del artículo'  
 (carácter legal | entero)80  
 fecha = 'Fecha de registro del artículo'  
 (carácter legal | entero)9  
 renovacion = 'Estatus de las renovaciones del artículo'

{entero}4

Tabla ALOJAMIENTO

Campos @num + no\_categoria + no\_usuario + articulo + descrip + precio + fecha + renovacion

Descripción de Campos

num = 'Número consecutivo del artículo'  
{entero}4  
no\_categoria = 'Número de categoría del artículo'  
{entero}4  
no\_usuario = 'Número identificador del usuario'  
{caracter legal | entero}13  
articulo = 'Nombre del artículo'  
{caracter legal | entero}20  
descrip = 'Descripción del artículo'  
{caracter legal | entero}150  
precio = 'Precio del artículo'  
{caracter legal | entero}15  
fecha = 'Fecha de registro del artículo'  
{caracter legal | entero}9  
renovacion = 'Estatus de las renovaciones del artículo'  
{entero}4

Tabla DIVERSOS

Campos @num + no\_categoria + no\_usuario + articulo + descrip + precio + foto + fecha + renovacion

Descripción de Campos

num = 'Número consecutivo del artículo'  
{entero}4  
no\_categoria = 'Número de categoría del artículo'  
{entero}4  
no\_usuario = 'Número identificador del usuario'  
{caracter legal | entero}13  
articulo = 'Nombre del artículo'  
{caracter legal | entero}20  
descrip = 'Descripción del artículo'  
{caracter legal | entero}150  
precio = 'Precio del artículo'  
{caracter legal | entero}15  
foto = 'Nombre de la foto del artículo'  
{caracter legal | entero}80  
fecha = 'Fecha de registro del artículo'  
{caracter legal | entero}9

renovacion = 'Estatus de las renovaciones del artículo'  
(entero)4

Tabla **ENSEÑANZA**

Campos @num + no\_categoria + no\_usuario + articulo + descrip + precio + fecha + renovacion

Descripción de Campos num = 'Número consecutivo del artículo'  
(entero)4

no\_categoria = 'Número de categoría del artículo'  
(entero)4

no\_usuario = 'Número identificador del usuario'  
(caracter legal | entero)13

articulo = 'Nombre del artículo'  
(caracter legal | entero)20

descrip = 'Descripción del artículo'  
(caracter legal | entero)150

precio = 'Precio del artículo'  
(caracter legal | entero)15

fecha = 'Fecha de registro del artículo'  
(caracter legal | entero)9

renovacion = 'Estatus de las renovaciones del artículo'  
(entero)4

Tabla **EQUIPO**

Campos @num + no\_subcat + no\_usuario + articulo + descrip + precio + foto + fecha + renovacion

Descripción de Campos num = 'Número consecutivo del artículo'  
(entero)4

no\_subcat = 'Número de subcategoría del artículo'  
(entero)4

no\_usuario = 'Número identificador del usuario'  
(caracter legal | entero)13

articulo = 'Nombre del artículo'  
(caracter legal | entero)20

descrip = 'Descripción del artículo'  
(caracter legal | entero)150

precio = 'Precio del artículo'  
(caracter legal | entero)15

foto = 'Nombre de la foto del artículo'  
(caracter legal | entero)80

fecha = 'Fecha de registro del artículo'

renovacion = 'Estatus de las renovaciones del artículo'  
{carácter legal | entero}\_9  
{entero}\_4

Tabla: MASCOTA

Campos @num + no\_subcat + no\_usuario + articulo + descrip + precio + foto + fecha + renovacion

Descripción de Campos

- num = 'Número consecutivo del artículo'  
{entero}\_4
- no\_subcat = 'Número de subcategoría del artículo'  
{entero}\_4
- no\_usuario = 'Número identificador del usuario'  
{carácter legal | entero}\_13
- articulo = 'Nombre del artículo'  
{carácter legal | entero}\_20
- descrip = 'Descripción del artículo'  
{carácter legal | entero}\_150
- precio = 'Precio del artículo'  
{carácter legal | entero}\_15
- foto = 'Nombre de la foto del artículo'  
{carácter legal | entero}\_80
- fecha = 'Fecha de registro del artículo'  
{carácter legal | entero}\_9
- renovacion = 'Estatus de las renovaciones del artículo'  
{entero}\_4

Tabla: MUSICA

Campos @num + no\_subcat + no\_usuario + articulo + descrip + precio + foto + fecha + renovacion

Descripción de Campos

- num = 'Número consecutivo del artículo'  
{entero}\_4
- no\_subcat = 'Número de subcategoría del artículo'  
{entero}\_4
- no\_usuario = 'Número identificador del usuario'  
{carácter legal | entero}\_13
- articulo = 'Nombre del artículo'  
{carácter legal | entero}\_20
- descrip = 'Descripción del artículo'  
{carácter legal | entero}\_150
- precio = 'Precio del artículo'  
{carácter legal | entero}\_15

foto = 'Nombre de la foto del artículo'  
 {carácter legal | entero}80  
 fecha = 'Fecha de registro del artículo'  
 {carácter legal | entero}9  
 renovación = 'Estatus de las renovaciones del artículo'  
 {entero}4

Tabla VEHÍCULO

Campos @num + no\_subcat + no\_usuario + articulo + descrip + precio + foto + fecha + renovación

Descripción de Campos

num = 'Número consecutivo del artículo'  
 {entero}4  
 no\_subcat = 'Número de subcategoría del artículo'  
 {entero}4  
 no\_usuario = 'Número identificador del usuario'  
 {carácter legal | entero}13  
 articulo = 'Nombre del artículo'  
 {carácter legal | entero}20  
 descrip = 'Descripción del artículo'  
 {carácter legal | entero}150  
 precio = 'Precio del artículo'  
 {carácter legal | entero}15  
 foto = 'Nombre de la foto del artículo'  
 {carácter legal | entero}80  
 fecha = 'Fecha de registro del artículo'  
 {carácter legal | entero}9  
 renovación = 'Estatus de las renovaciones del artículo'  
 {entero}4

carácter legal {[a-z] | [A-Z]}

entero {[0-9]}

Términos especializados

Aviso. Anuncio que contiene la información correspondiente a un artículo o servicio que ofrece un miembro de la comunidad universitaria.

Cartel. Papel, impreso o manuscrito, que se fija en un paraje público para hacer saber alguna cosa.

## RECURSOS PARA EL DESARROLLO DEL SISTEMA

### Equipo de Trabajo

ROL	NÚMERO DE PERSONAS
Lider de Proyecto	1
Analista de Sistemas	2
Programador	2
Diseñador Gráfico	2

### Hardware

Dos computadoras personales con las siguientes características:

- ✓ Procesador Pentium III.
- ✓ Disco duro de 10 GB.
- ✓ 128 MB en RAM.
- ✓ Sistema Operativo Windows 98.

### Software

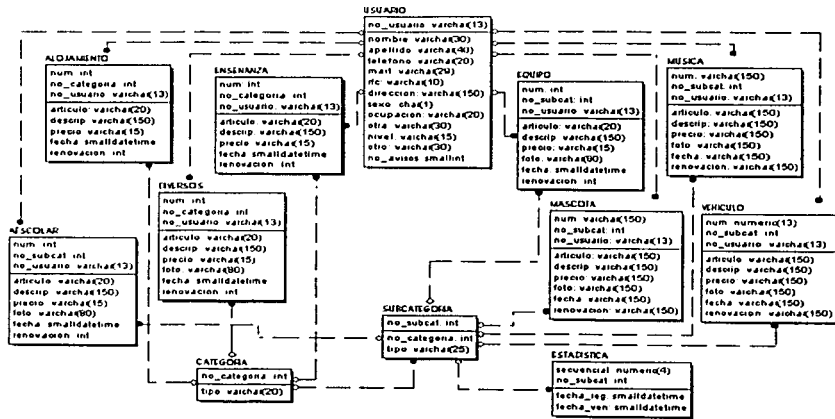
El sistema Aviso de Ocasión se desarrollará bajo las siguientes tecnologías:

- ✓ Lenguaje de programación Perl versión 5.0 para la elaboración de los scripts.
- ✓ Manejador de bases de datos Sybase versión II para la manipulación de datos.
- ✓ DreamWeaver versión 3.0 para el diseño de las páginas Web.
- ✓ Photoshop versión 6.0 para la creación y edición de imágenes.
- ✓ Servidor Web Apache versión 1.3.9 que proporciona la implementación del protocolo HTTP.



### 5.3 DISEÑO

#### DIAGRAMA ENTIDAD-RELACIÓN FÍSICO



#### DICCIONARIO DE DATOS FÍSICO

USUARIO							
Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
nombre	Nombre del usuario		varchar	30	SI	NO	Angeles
apellido	Apellidos del usuario		varchar	40	SI	NO	Sánchez
telefono	No. telefónico del usuario		varchar	20	SI	NO	56060675
mail	Correo electrónico del usuario		varchar	29	NO	NO	Angeles@yahoo.com
rfc	Registro Federal de Causante del usuario		varchar	10	SI	SI	SAZA771004
direccion	Domicilio del		varchar	150	NO	NO	Zapote #22 Col. Rey

usuario		Neza Cd. Neza					
sexo	Género del usuario	varchar	1	NO	NO	F M	
ocupacion	Ocupacion del usuario	varchar	20	NO	NO	Estudiante UNAM Trabajador UNAM Académico UNAM Otro	
ora	Ocupación especificada por el usuario	varchar	30	NO	NO	Estudiante UNAM	
nivel	Nivel de estudios del usuario	varchar	15	NO	NO	Bachillerato Licenciatura Posgrado Otro	
otro	Nivel de estudios especificado por el usuario	varchar	30	NO	NO	Maestría	
no_avisos	Número de anuncios registrado por el usuario	smallint		SI	NO	2	

### CATEGORIA

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
no_categoria	Número de categoría	PK	int		SI	SI	1
Tipo	Nombre de la categoría		varchar	20	SI	SI	MASCOTA

### SUBCATEGORIA

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
no_subcat	Número de subcategoría	PK	int		SI	SI	1
no_categoria	Número de categoría	FK	int		SI	SI	1
tipo	Nombre de la subcategoría		varchar	25	SI	SI	PERRO

### ESTADISTICA

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
secuencial	Número consecutivo que indica el artículo publicado	PK	int		SI	SI	4

no_subcat	Número de subcategoría	FK	int	SI	SI	1
fecha_reg	Fecha en la que se registro el anuncio		smaldate time	SI	NO	Jun012001
fecha_ven	Fecha de vencimiento del anuncio publicado		smaldate time	SI	NO	Jul152001

### AESCOLAR

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
num	Número consecutivo que identifica el artículo	PK	int		SI	SI	1
no_subcat	Número de subcategoría	PK	int		SI	SI	1
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
artículo	Nombre del artículo		varchar	20	SI	NO	Calculadora científica
descrip	Breve descripción del artículo		varchar	150	SI	NO	Seminueva marca CASIO
precio	Precio de venta del artículo		varchar	15	NO	NO	\$100
foto	Foto del artículo		varchar	80	NO	NO	calcu.jpg
fecha	Fecha de registro del artículo		smaldate time		SI	NO	Sep012001
renovacion	Indica el número de veces que se ha renovado la publicación del anuncio		int		SI	NO	0

### ALOJAMIENTO

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
num	Número consecutivo que identifica el artículo	PK	int		SI	SI	1
no_categoria	Número de categoría	PK	int		SI	SI	2
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
artículo	Nombre del artículo		varchar	20	SI	NO	Cuarto para señoritas
descrip	Breve descripción del artículo		varchar	150	SI	NO	Con todos los servicios no se aceptan animales
precio	Precio de venta del artículo		varchar	15	NO	NO	\$1000 al mes

fecha	Fecha de registro del artículo	smalldate time	SI	NO	Sep012001
renovacion	Indica el número de veces que se ha renovado la publicación del anuncio	int	SI	NO	1

### DIVERSOS

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
num	Número consecutivo que identifica el artículo	PK	int		SI	SI	1
no_categoria	Número de categoría	PK	int		SI	SI	4
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
articulo	Nombre del artículo		varchar	20	SI	NO	Colecciones de postales
descrip	Breve descripción del artículo		varchar	150	SI	NO	Paisajes de todo el mundo
precio	Precio de venta del artículo		varchar	15	NO	NO	\$500
foto	Foto del artículo		varchar	80	NO	NO	
fecha	Fecha de registro del artículo		smalldate time		SI	NO	Sep012001
renovacion	Indica el número de veces que se ha renovado la publicación del anuncio		int		SI	NO	0

### ENSEÑANZA

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
num	Número consecutivo que identifica el artículo	PK	int		SI	SI	1
no_categoria	Número de categoría	PK	int		SI	SI	6
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
articulo	Nombre del artículo		varchar	20	SI	NO	Clases de inglés
descrip	Breve descripción del artículo		varchar	150	SI	NO	Clases a domicilio
precio	Precio de venta del artículo		varchar	15	NO	NO	Aparta

fecha	Fecha de registro del artículo	smalldate time	SI	NO	Sep012001
renovacion	Indica el número de veces que se ha renovado la publicación del anuncio	int	SI	NO	1

### EQUIPO

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
num	Número consecutivo que identifica el artículo	PK	int		SI	SI	1
no_subcat	Número de subcategoría	PK	int		SI	SI	13
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
articulo	Nombre del artículo		varchar	20	SI	NO	Computadora VAIO
descrip	Breve descripción del artículo		varchar	150	SI	NO	Seminueva en buen estado
precio	Precio de venta del artículo		varchar	15	NO	NO	
foto	Foto del artículo		varchar	80	NO	NO	
fecha	Fecha de registro del artículo		smalldate time		SI	NO	Sep012001
renovacion	Indica el número de veces que se ha renovado la publicación del anuncio		int		SI	NO	2

### MASCOTA

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Único	Ejemplo
num	Número consecutivo que identifica el artículo	PK	int		SI	SI	1
no_subcat	Número de subcategoría	PK	int		SI	SI	1
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
articulo	Nombre del artículo		varchar	20	SI	NO	Cachorro maltés
descrip	Breve descripción del artículo		varchar	150	SI	NO	Color blanco de 3 meses
precio	Precio de venta del artículo		varchar	15	NO	NO	3000

foto	Foto del artículo	varchar	80	NO	NO	Perro.jpg
fecha	Fecha de registro del artículo	smalldate time		SI	NO	Sep012001
renovacion	Indica el número de veces que se ha renovado la publicación del anuncio	int		SI	NO	2

### MUSICA

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Unico	Ejemplo
num	Número consecutivo que identifica el artículo	PK	int		SI	SI	1
no_subcat	Número de subcategoría	PK	int		SI	SI	28
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
artículo	Nombre del artículo		varchar	20	SI	NO	CD del gran silencio
descrip	Breve descripción del artículo		varchar	150	SI	NO	Libres y locos
precio	Precio de venta del artículo		varchar	15	NO	NO	90
foto	Foto del artículo		varchar	80	NO	NO	egs.jpg
fecha	Fecha de registro del artículo		smalldate time		SI	NO	Sep012001
renovacion	Indica el número de veces que se ha renovado la publicación del anuncio		int		SI	NO	2

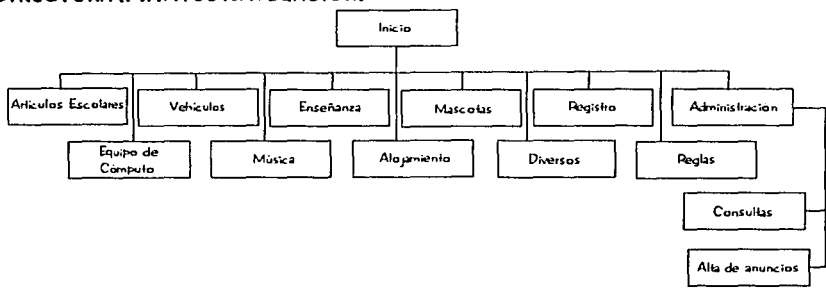
### VEHICULO

Atributo	Descripción	Tipo de Llave	Tipo de Dato	Tamaño	Requerido	Unico	Ejemplo
num	Número consecutivo que identifica el artículo	PK	int		SI	SI	1
no_subcat	Número de subcategoría	PK	int		SI	SI	23
no_usuario	Número identificador de usuario	PK	varchar	13	SI	SI	SAZA771004456
artículo	Nombre del artículo		varchar	20	SI	NO	VW Sedan
descrip	Breve descripción del artículo		varchar	150	SI	NO	Modelo 97 color rojo cereza
precio	Precio de venta del artículo		varchar	15	NO	NO	
Foto	Foto del artículo		varchar	80	NO	NO	Vw.jpg

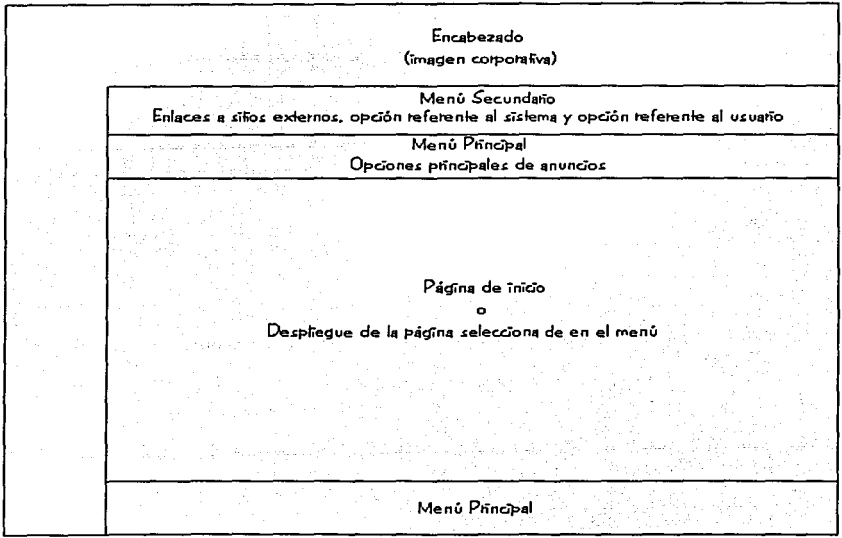
Renovacion      Indica el número de veces que se ha renovado la publicación del anuncio

Int      SI      NO      2

**ESTRUCTURA (MAPA DE NAVEGACIÓN)**



**INTERFAZ**



## **SEGURIDAD**

Para iniciar una sesión, el sistema realizará una autenticación del usuario, a través, de un nombre de usuario y una contraseña o *password* únicos proporcionados por el sistema al momento de llevarse a cabo el registro.

Para la administración del sistema, éste, realizará la autenticación del administrador, a través, de un nombre de usuario y una contraseña o *password* únicos.

## **5.3 DESARROLLO**

**PROGRAMAS FUENTE (V. APÉNDICE A)**  
**PANTALLAS DEL SISTEMA (V. APÉNDICE B)**



## CONCLUSIONES

En la UNAM, el uso de las tecnologías de la información desempeña un papel fundamental en el propio desarrollo de la institución, gracias a ello, se han generado las condiciones idóneas para mejorar la calidad de la enseñanza, la difusión del conocimiento y de la cultura.

México ha sido precursor a nivel Latinoamérica en el uso de las nuevas tecnologías de información enfocado a apoyar el proceso de enseñanza-aprendizaje, dichas tecnologías, son un instrumento valioso para brindar acceso a la información, llevar a efecto actividades educativas y propiciar la comunicación entre individuos.

La capacidad de la sociedad para procesar la información y alcanzar el conocimiento con altos niveles de calidad y cantidad, es determinante en el desarrollo de la misma. Por otra parte, como el conocimiento tiene un valor estratégico, las instituciones educativas son grandes protagonistas sociales, se les plantean nuevos retos para formar técnicos, profesionales e investigadores y ofrecerles a todos ellos, educación actualizada y permanente y, generar, transferir y difundir el conocimiento

El modelo educativo actual, debe aprovechar las nuevas tecnologías de la información para mejorar la calidad y la cobertura de la enseñanza superior, con el fin de responder, lo mejor posible, a las exigencias de la sociedad a la que pertenecemos.

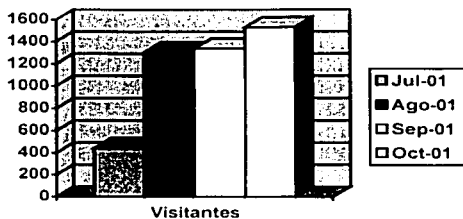
El uso del servicio de Internet *World Wide Web*, ha ayudado a nuestra Universidad en la creación de comunidades de aprendizaje y facilitando la comunicación y el acceso a la información, a través, de distintos servicios diseñados, exclusivamente, a las necesidades de la comunidad universitaria, los cuales, encontramos en el Portal UNAM (<http://www.unam.mx>). Como consecuencia, nuestra máxima casa de estudios cumple su misión: "Investigación, Docencia y Difusión de la Cultura".

El Sistema de Información Orientado a la *Web* (WIS) "Aviso de Ocasión del Portal UNAM" se liberó el pasado 04 de Abril de 2001, lleva funcionando hasta el mes de término de esta tesis (noviembre del mismo año) 7 meses, en el URL: <http://www.deocasion.unam.mx>.

Este sistema, funciona como un servicio en el que la comunidad universitaria y el público en general pueden ofrecer a la venta productos y/o servicios.

Los medios de publicidad por los que se ha dado a conocer este sistema han sido el Portal UNAM y distintas publicaciones que emite la Universidad como el periódico mural: "El Mural".

Conforme ha pasado el tiempo ha ido aumentando el número de visitantes de manera gradual, se puede observar en la siguiente gráfica que muestra la audiencia que ha tenido el Aviso de Ocasión en los últimos 4 meses



La información plasmada en la gráfica anterior, fue recabada a través del sitio *Web* <http://www.certifica.com>, que es un servicio de medición y certificación de audiencia para Internet que permite acceder a despliegues de visitantes únicos, calculados para periodos diarios, semanales y mensuales.

El número de visitantes promedio por día del Aviso de Ocasión es 208, lo cual ubica al sistema en el segundo lugar más visitado del Portal UNAM.

Durante los siete meses que el sistema ha estado en línea ha obtenido los siguientes resultados:

- ✓ Avisos Publicados: 1231
- ✓ Usuarios registrados: 1214



## APÉNDICES

### A. PROGRAMAS FUENTE

#### cgi-bin

bajaspercron.pl

```
#!/usr/bin/perl
$|=1. #para agilizar la respuesta

require "home/users/D0/ocasion/cgi-bin/var.pl".
use Sybase DBlib.

$fecha= date 'DATE %m-%d-%Y | cut -d"|" -f2-4'.

$dbbh = new Sybase DBlib $log, $pas, $ser, 'ocasion'.
#si la diferencia entre la fecha(de vencimiento) y la de hoy es 1
$dbbh->dbcmd("select
    dia1 = (datediff(day,fecha,getdate()))
    from AESCOLAR
    where (datediff(day,fecha,getdate()))>0
    union
    select dia2 = (datediff(day,fecha,getdate()))
    from ALOJAMIENTO
    where (datediff(day,fecha,getdate()))>0
    union
    select dia3 = (datediff(day,fecha,getdate()))
    from DIVERSOS
    where (datediff(day,fecha,getdate()))>0
    union
    select dia4 = (datediff(day,fecha,getdate()))
    from ENSEMANZA
    where (datediff(day,fecha,getdate()))>0
    union
    select dia5 = (datediff(day,fecha,getdate()))
    from EQUIPO
    where (datediff(day,fecha,getdate()))>0
    union
    select dia6 = (datediff(day,fecha,getdate()))
    from MASCOTA
    where (datediff(day,fecha,getdate()))>0
    union
    select dia7 = (datediff(day,fecha,getdate()))
    from MUSICA
    where (datediff(day,fecha,getdate()))>0
    union
    select dia8 = (datediff(day,fecha,getdate()))
    from VEHICULO
    where (datediff(day,fecha,getdate()))>0
    ")
$dbbh->dbsql<exec
$dbbh->dbresults,
$dbbh->DBROWS

$status=$dbh->DBROWS.

if ($status==0)
{
    print "LA BASE DE DATOS NO HA SUFRIDO MODIFICACIONES";
}
else
{
    $CATEGORIA="AESCOLAR";
    $NUMERO=1.
    &actualizacion ($CATEGORIA, $NUMERO);
    $CATEGORIA="ALOJAMIENTO";
    $NUMERO=2.
    &actualizacion ($CATEGORIA, $NUMERO);
    $CATEGORIA="DIVERSOS";
    $NUMERO=3
    &actualizacion ($CATEGORIA, $NUMERO);
    $CATEGORIA="ENSEMANZA";
    $NUMERO=4.
    &actualizacion ($CATEGORIA, $NUMERO);
    $CATEGORIA="EQUIPO";
    $NUMERO=5.
    &actualizacion ($CATEGORIA, $NUMERO);
    $CATEGORIA="MASCOTA";
    $NUMERO=6.
}
```

```

&actualizacion ($CATEGORIA, $NUMERO),
$CATEGORIA="MUSICA";
$NUMERO=7;
&actualizacion ($CATEGORIA, $NUMERO),
$CATEGORIA="VEHICULO";
$NUMERO=8;
&actualizacion ($CATEGORIA, $NUMERO),

```

```

print "LA BASE DE DATOS HA SIDO MODIFICADA"
}

```

```

sub actualizacion
{

```

```

    local ($var),
    $CATEGORIA=$_I[0],
    $NUMERO=$_I[1],
    $conex="&sdbr"&"$NUMERO";
    $conex = new Sybase DBlib $log, $pas, $ser, 'ocasion';
    $conex->dbcmd("select $CATEGORIA no_usuario, USUARIO no_avisos as avisos, count(*) as menos
    from $CATEGORIA, USUARIO
    where (dateid((day.$CATEGORIA fecha.getdate()))>0
    and $CATEGORIA no_usuario like USUARIO no_usuario
    group by $CATEGORIA no_usuario
    having $CATEGORIA no_usuario like USUARIO no_usuario");

```

```

    $conex->dbsqlxec;

```

```

    $conex->dbresults;

```

```

    $conex->DBROWS;

```

```

    $status=$conex->DBROWS;

```

```

    $var=0;

```

```

    while(%cat = $conex->dbnextrow(1))
    {

```

```

        $avisosn=$cat(avisos);

```

```

        $numero=$cat(menos);

```

```

        $navisos=$avisosn-$numero;

```

```

        $ncta=$cat(no_usuario);

```

```

        $conex = new Sybase DBlib $log, $pas, $ser, 'ocasion';

```

```

        for($var=0, $var<=$status, $var++)
        {

```

```

            $conex->dbcmd("update USUARIO
            set no_avisos=$navisos
            where no_usuario like '$ncta'");

```

```

            $conex->dbsqlxec;

```

```

            $conex->dbresults;

```

```

        } #fin del for
    } #fin del while

```

```

    $conexbf = new Sybase DBlib $log, $pas, $ser, 'ocasion';

```

```

    if ( ($CATEGORIA ne "ALOJAMIENTO") || ($CATEGORIA ne "ENSENANZA") )
    {

```

```

        $conexbf->dbcmd("select foto
        from $CATEGORIA
        where (dateid((day.fecha.getdate()))>0");

```

```

        $conexbf->dbsqlxec;

```

```

        $conexbf->dbresults;

```

```

        while(%fotos = $conexbf->dbnextrow(1))
        {

```

```

            $dfil=$fotos(foto);

```

```

            $dfifoto="rm /home/users00/ocasion/htdocs/FOTOS/$dfil";

```

```

            system($dfifoto);
        }
    }

```

```

    $conexb = new Sybase DBlib $log, $pas, $ser, 'ocasion';

```

```

    $conexb->dbcmd("delete
    from $CATEGORIA
    where (dateid((day.fecha.getdate()))>0");

```

```

    $conexb->dbsqlxec;

```

```

    $conexb->dbresults;
}

```

checavlgcron.pl

```

#!/usr/bin/perl

```

```

$|=1; #para agilizar la respuesta

```

```

require "home/users00/ocasion/cgi-bin/var.pl";
use Sybase DBlib;

```

```

#guarda la fecha del dia

```

```

$fecha = date "DATE %m-%d-%Y" | cut -d" " -f2-4;

```

```

    $dbr = new Sybase DBlib $log, $pas, $ser, 'ocasion';

```

```

    #la vigencia del anuncio es de tres semanas (21 dias) pero checamos la vigencia tres dias antes para mandar el correo

```

```

Sdbh->dbcmd("select
dia1 = (datediff(day,getdate(),fecha))
from AESCOLAR
where (datediff(day,getdate(),fecha))=3
union
select dia2 = (datediff(day,getdate(),fecha))
from ALOJAMIENTO
where (datediff(day,getdate(),fecha))=3
union
select dia3 = (datediff(day,getdate(),fecha))
from DIVERSOS
where (datediff(day,getdate(),fecha))=3
union
select dia4 = (datediff(day,getdate(),fecha))
from ENSEANZA
where (datediff(day,fecha,getdate()))=3
union
select dia5 = (datediff(day,getdate(),fecha))
from EQUIPO
where (datediff(day,getdate(),fecha))=3
union
select dia6 = (datediff(day,getdate(),fecha))
from MASCOTA
where (datediff(day,getdate(),fecha))=3
union
select dia7 = (datediff(day,getdate(),fecha))
from MUSICA
where (datediff(day,getdate(),fecha))=3
union
select dia8 = (datediff(day,getdate(),fecha))
from VEHICULO
where (datediff(day,getdate(),fecha))=3
")
Sdbh->dbstate+ec
Sdbh->dbresults
Sdbh->DBROWS
Sstatus=$dbh->DBROWS,

```

```

if ($status=0) #si no hay anuncios que expren
{
print "NO SE HAN MANDADO CORREOS",
}
else
{
$CATEGORIA="AESCOLAR"
$NUMERO=1
&actualizacion ($CATEGORIA $NUMERO)
$CATEGORIA="ALOJAMIENTO"
$NUMERO=2
&actualizacion ($CATEGORIA $NUMERO)
$CATEGORIA="DIVERSOS"
$NUMERO=3
&actualizacion ($CATEGORIA $NUMERO)
$CATEGORIA="ENSEANZA"
$NUMERO=4
&actualizacion ($CATEGORIA $NUMERO)
$CATEGORIA="EQUIPO"
$NUMERO=5
&actualizacion ($CATEGORIA $NUMERO)
$CATEGORIA="MASCOTA"
$NUMERO=6
&actualizacion ($CATEGORIA $NUMERO)
$CATEGORIA="MUSICA"
$NUMERO=7
&actualizacion ($CATEGORIA $NUMERO)
$CATEGORIA="VEHICULO"
$NUMERO=8
&actualizacion ($CATEGORIA $NUMERO)
print "SE HAN MANDADO CORREOS",
}

```

sub actualizacion

```

{
local ($var)
CATEGORIA=$_[0]
NUMERO=$_[1]
$comex="$dbh"."$NUMERO"
$comex = new Sybase DBLib Slog $pas, $ser, 'ocasion',
$comex->dbcmd("select $CATEGORIA no_usuario, $CATEGORIA num,
$CATEGORIA no_cat, $CATEGORIA no_subc,
$CATEGORIA descrip, $CATEGORIA articulo,
USUARIO mail,

```

```

USUARIO no_avisos as avisos, count(*) as menos
from $CATEGORIA.$USUARIO
where (datediff(day,getdate()),$CATEGORIA.fecha)=3
and $CATEGORIA.no_usuario like USUARIO no_usuario
group by $CATEGORIA no_usuario
having $CATEGORIA no_usuario like USUARIO no_usuario
and (datediff(day,getdate()),$CATEGORIA.fecha)=3").

```

```

$conex->dbselectec,
$conex->dbresults,
$conex->DBROWS,
$status=$conex->DBROWS
$var=0,

```

```

while(%cat = $conex->dbnextrow(1))
{
    $avisosn=$cat(avisos),
    $numero=$cat(menos),
    $navisos=$avisosn-$numero,
    $ncta=$cat(no_usuario),
    $articulo=$cat(articulo),
    $mail=$cat(mail),
    $descripcion=$cat(descrip),
    $clave1=$cat(no_cat),
    $clave2=$cat(no_subc),
    $clave3=$cat(num),
    $prefijo=0
    if ($clave1 < 10){ $clave1=$prefijo $clave1, }
    if ($clave2 < 10){ $clave2=$prefijo $clave2, }
    $claveanuncio=$clave1 $clave2 $clave3,
    $conexa = new Sybase DBLib $pas, $ser, $ocasion",
    for($var=0 $var-$status $var++)
    {

```

```

        if($mail ne "" || $mail ne " "){
            @parte=(split(/@/ $mail)),
            $uno=$parte[0]
            $dos=$parte[1]

```

```

            system("sed 's/articulo/$articulo/' /home/users00/ocasion/cgi-bin/vigilancia.txt > /home/users00/ocasion/cgi-bin/vigaux.txt"),
            system("sed 's/nd/$claveanuncio/' /home/users00/ocasion/cgi-bin/vigaux.txt > /home/users00/ocasion/cgi-bin/vigaux2.txt"),
            system("sed 's/descripcion/$descripcion/' /home/users00/ocasion/cgi-bin/vigaux2.txt > /home/users00/ocasion/cgi-
bin/vigaux3.txt")

```

```

            system("mail $uno@$dos < /home/users00/ocasion/cgi-bin/vigaux3.txt"),
            system("rm /home/users00/ocasion/cgi-bin/vigaux.txt"),
            system("rm /home/users00/ocasion/cgi-bin/vigaux2.txt"),
            system("rm /home/users00/ocasion/cgi-bin/vigaux3.txt"),
        } #fin if
    } #fin del for
} #fin del while

```

datosb2.pl

```

#!/usr/bin/perl
$|=1, #para agilizar la respuesta

```

```

require "var.pl"
use Sybase DBLib,

```

```

use CGI,
$query=new CGI

```

```

$nombre=$query->param('nombre'),
$ap=$query->param('apellidos'),
$cuenta=$query->param('cuenta'),
$proc=$query->param('proceso'),
$foto=$query->param('foto'),

```

```

print "Content-type text/html\n\n",
require "sup.txt",

```

#####Validacion del tipo de usuario#####

```

if($cuenta > 999999)
{
    $tabla = "Estudiante",
}
else
{
    $tabla = "Trabajador",
}

```

```

print "<font face='Arial, Verdana, Helvetica, sans-serif' size='4'>
<center><br> El registro de,</center>
<br><center> <font face='Arial, Verdana, Helvetica, sans-serif' size='2'>
$nombre $ap con clave $cuenta ha sido borrado <br>".

```

```

Scadena=$Cuenta "e" $proc,
$del="grep -v \"$Scadena\" usuario.txt > usutmp.txt",
system($del),
system("touch /touch.txt"),
$cambo="mv usutmp.txt usuario.txt",
system($cambo),
$dfotol="rm /hdocs/FOTOS/$foto",
system($dfotol),
print "<br><br><br><a href=\"$datos2.pl\">REGRESA</a></font></center>",
require "inf.txt",

```

datos2.pl

```

#usr/bin/perl

print "Content-type text/html\n\n",

require "sup.txt",

open(AVISO,"-usuario.txt"),
$num=1,
while($linea = <AVISO>)
{
  chop $linea
  ($nom $ap $tel $ia $om $cat $ncat, $subc, $ari, $desc, $proc, $foto, $cta, $proc, $fecha)=split(/=/, $linea),
  print "
<form name=\"inserta$num\" method=\"POST\">
<br>
<p align=\"center\"><font face=\"Arial, Verdana, Helvetica, sans-serif\" size=\"3\">
<center><b>AVISO NUMERO $num</b></center>
<br>
<br>
Nombre <br>
<input name=\"nombrel\" type=\"text\" value=\"$nom\">
<br>
Apellidos <br>
<input name=\"apellidos\" type=\"text\" value=\"$apl\">
<br>
Teléfono <br>
<input name=\"telefonol\" type=\"text\" size=\"20\" value=\"$tel\">
<br>
Lada <br>
<input name=\"$ladal\" type=\"text\" size=\"15\" value=\"$lal\">
<br>
E-mail <br>
<input name=\"email\" type=\"text\" size=\"30\" value=\"$eml\">
<br>
Artículo <br>
<input name=\"articulol\" type=\"text\" size=\"20\" value=\"$artil\">
<br>
Categoría <br>
<input name=\"$catl\" type=\"text\" size=\"20\" value=\"$catal\">
<br>
No de Categoría <br>
<input name=\"$ncatl\" type=\"text\" size=\"20\" value=\"$ncatal\">
<br>
Subcategoría <br>
<input name=\"$subcatal\" type=\"text\" size=\"20\" value=\"$subcatal\">
<br>
Descripción <br>
<input name=\"$descripcionl\" type=\"text\" size=\"50\" value=\"$descri\">
<br>
Precio <br>
<input name=\"$precol\" type=\"text\" size=\"10\" value=\"$precl\">
<br>
Foto <br>
<input name=\"$fotol\" type=\"text\" size=\"20\" value=\"$fotol\">
<br>
Clave <br>
<input name=\"$cual\" type=\"text\" size=\"40\" value=\"$cual\">
<br>
No de proceso <br>
<input type=\"text\" name=\"$proceso\" value=\"$procl\">
<br>
Fecha <br>
<input type=\"text\" name=\"$fecl\" value=\"$fechal\">
<br>

if ($foto ne "")
{
  print "<img src=\"/FOTOS/$foto\" width=\"100\" height=\"100\"></img>";
}

```

```

print " <br> <br>
<center><input type='button' name='insertar' value='Insertar' Onclick='JavaScript inserta$num action='datosr33-
libre pf inserta$num submit()' /></center>
<left> <input type='button' name='Borrar' value='Borrar'
Onclick='JavaScript inserta$num action='datosb2 pf.inserta$num submit()' /></left>
</font>
<tr SIZE='51' color='#000099' NOSHADA WIDTH='100%'>
</tr>
</form> "
$num +=
$numnum=$num-1
) #fin del while
print " <br> <br>
<font face='Arial, Verdana, Helvetica, sans-serif' size='13!'>
<center><b>N&uacute;mero total de avisos $numnum</b></center>
</font> "
close(AVISO)
require "inf.txt"

```

datosr33-libre.pl

```

#Usar/bin/perl
$! = 1 #para agilizar la respuesta

require "var.pl"
use Sysbase DBlib

use CGI;
$query=new CGI;

$subcatlogo = $query->param('subcat')
$ncat = $query->param('ncat')
$nombre = $query->param('nombre')
$sap = $query->param('apellidos')
$tel = $query->param('telefono')
$lada = $query->param('lada')
$email = $query->param('email')
$art = $query->param('articulo')
$cat = $query->param('cat')
$desc = $query->param('descripcion')
$precio = $query->param('precio')
$foto = $query->param('foto')
$cuenta = $query->param('cuenta')
$proc = $query->param('proceso')
$fecha = $query->param('fec')

@bus = (split(/ /, $foto))
if($bus[1] eq "FOTOVACIA"){$foto=""}

#####Validacion del tipo de usuario#####
$tabla="USUARIO"
$dbh = new Sysbase DBlib $log, $pas, $ser, 'ocasion';
$dbh->dbcmd("select no_avisos from USUARIO
where no_usuario=\"$cuenta\"");
$dbh->dbsqlrec
$dbh->dbresults

while(%cat = $dbh->dbnextrow(1))
{
    $avisos="$cat[no_avisos]";
    $navisos=$avisos+1;
} #fin del while

print "Content-type: text/html\n\n";
require "sup.txt";

#####Validacion del numero de avisos#####
# el numero de avisos queda sin efecto por el momento
$pass="si";
if ($pass eq "no")
{
    print "<center><br><br><br>
<font face='Arial, Verdana, Helvetica, sans-serif' size='14!'>
El usuario <b>$nombre $ap</b>
<br> EXCEDE EL LÍMITE DE AVISOS</center></font>";
} #fin del if
else
{
#####VERIFICAR SI EL REGISTRO NO SE REPITE
$dbh1 = new Sysbase DBlib $log, $pas, $ser, 'ocasion';
$dbh1->dbcmd("select " from $cat
where no_usuario=\"$cuenta\"
and descrip=\"$desc\"");

```



```

$dbh1->dbsql<exec
$dbh1->dbresults
$dbh1->DBROWS
$status=$dbh1->DBROWS.
if ($status == 0)
{
    print "<center><br><br><br>
    <font face='Arial, Verdana, Helvetica, sans-serif' size='14'>
    El fondo del usuario <b>$nombre $ap </b>
    <br> HA SIDO INSERTADO </center></font>".
}
#####BORRA REGISTRO DEL ARCHIVO usuario.txt
$cadena=$uenta "m" $proc.
$del="grep -v \"$cadena\" usuario.txt > /dev/null"
system($del)
$scambio="mv /dev/null usuario.txt".
system($scambio).
#####INSERTAR DATOS EN LA TABLA DEL ARTICULO
$dbh2 = new Sybase DBlib $log $pas $ser $ocasion.
if ( ($cat eq "ALOJAMIENTO") || ($cat eq "ENSEÑANZA") )
{
    $dbh2->dbcmd("insert into
    $cat(num_no_usuario no_cat no_subc articulo descrip precio fecha renovacion)
    values($proc,$$ocuenta,$$ncat,$$subcatego,\"$art\", \"$desc\", \"$precio\", dateadd(day,21,\"$fecha\"),0)").
}
else
{
    $dbh2->dbcmd("insert into
    $cat(num_no_usuario no_cat no_subc articulo descrip precio fecha foto renovacion)
    values($proc,$$ocuenta,$$ncat,$$subcatego,\"$art\", \"$desc\", \"$precio\", dateadd(day,21,\"$fecha\"),\"$foto\",0)").
}
$dbh2->dbsql<exec
#####INSERTAR DATOS EN LA TABLA DEL USUARIO
$dbh3 = new Sybase DBlib $log $pas $ser $ocasion.
$dbh3->dbcmd("update $tabla set no_avisos=$navisos where no_usuario=\"$ocuenta\"").
$dbh3->dbsql<exec
$dbh3->dbresults
$dbh3->dbclose.
#####INSERTAR DATOS EN LA TABLA DE ESTADISTICA
$dbh4 = new Sybase DBlib $log $pas $ser $ocasion.
$dbh4->dbcmd("select max(secuencial) as maximo from ESTADISTICA").
$dbh4->dbsql<exec
$dbh4->dbresults
$dbh4->dbclose.
while(%cat4 = $dbh4->dbnextrow(1))
{
    $siguiente="$cat4(maximo)".
} #fin del while
$siguiente=$siguiente+1.
$dbh4->dbclose.
$dbh5 = new Sybase DBlib $log $pas $ser $ocasion.
$dbh5->dbcmd("insert into ESTADISTICA values($siguiente, $ncat, $subcatego, \"$fecha\", dateadd(day,21,\"$fecha\"))").
$dbh5->dbsql<exec.
$dbh5->dbresults.
$dbh5->dbclose.
# lineas agregadas para avisar al usuario que ya se publico su anuncio
if($email ne "" || $email ne " ")
{
    @parte=(split(/@/ $email)).
    $uno=$parte[0].
    $dos=$parte[1].
    system("sed \"s/articulo/$art/\" /home/users00/ocasion/cgi-bin/vanuncioya.txt > /home/users00/ocasion/cgi-bin/yaaux1.txt").
    system("sed \"s/cat/$cat/\" /home/users00/ocasion/cgi-bin/yaaux1.txt > /home/users00/ocasion/cgi-bin/yaaux2.txt").
    system("sed \"s/descripcion/$desc/\" /home/users00/ocasion/cgi-bin/yaaux2.txt > /home/users00/ocasion/cgi-bin/yaaux3.txt").
    system("mail $uno@ $dos < yaaux3.txt").
    system("rm /home/users00/ocasion/cgi-bin/yaaux1.txt").
    system("rm /home/users00/ocasion/cgi-bin/yaaux2.txt").
} # if mail
} #fin del if de registros repetidos
else
{
    print "<center><br><br><br>
    <font face='Arial, Verdana, Helvetica, sans-serif' size='14'>
    Este registro del usuario $nombre $ap
    <br> YA EXISTE $status2</center></font>".
} #fin del else de registros repetidos
} #fin del else de no de avisos

```

```
print "<center><br><br><br><font face='Arial, Verdana, Helvetica, sans-serif' size='21'><a
href='<datos2.pl?>REGRESA</a></font></center>";
require "nl.txt"
```

#### individual.pl

```
#!/usr/bin/perl
$|=1 #para agilizar la respuesta

require "var.pl"
use Sybase DBlib

use CGI
$query=new CGI

$clave=$query->param('numero');
$clave=(uc($clave));
#guarda la fecha del dia
$fechaoy="date " . DATE "%m-%d-%Y" | cut -d" " -f2-4";

print "Content-type: text/html\n\n";
require "sup.txt";

$dbhu=new Sybase DBlib $log $pas, $$ser, 'ocasion';
$dbhu->dbcmd("select *
from USUARIO
where no_usuario like \"\$clave\" ");
$dbhu->dbsqlexec
$dbhu->dbresults
while(%catu = $dbhu->dbnextraw(1))
{ # while catu
  $elusuario=$catu{nombre};
  $elusuario=$elusuario . " " $catu{apellido};
  $susavisos=$catu{no_avisos};
} # while catu

$dbh = new Sybase DBlib $log $pas, $$ser, 'ocasion';
$dbh->dbcmd("select *
from AESCOLAR
where no_usuario like \"\$clave\"
UNION
select *
from MASCOTA
where no_usuario like \"\$clave\"
UNION
select *
from VEHICULO
where no_usuario like \"\$clave\"
UNION
select *
from DIVERSOS
where no_usuario like \"\$clave\"
UNION
select *
from MUSICA
where no_usuario like \"\$clave\"
UNION
select *
from EQUIPO
where no_usuario like \"\$clave\" ");

$dbh->dbsqlexec;
$dbh->dbresults;
$dbh->DBROWS;
$$status=$dbh->DBROWS;
$$lstatus=$dbh->dbnumcols;

$dbh2=new Sybase DBlib $log $pas, $$ser, 'ocasion';
$dbh2->dbcmd("select *
from ALOJAMIENTO
where ALOJAMIENTO no_usuario like \"\$clave\"
UNION
select *
from ENSEANZA
where ENSEANZA no_usuario like \"\$clave\" ");

$dbh2->dbsqlexec;
$dbh2->dbresults;
$dbh2->DBROWS;
$$status2=$dbh2->DBROWS;
$$mra = $dbh2->dbnumcols;
$$cuantosa=($$status+$$mra);
```

```

if ($status==0 && $status2==0) #si no hay anuncios registrados
{ #id 1
    print "<br>"
    <font face=V'Arial, Verdana, Helvetica, sans-serif" size=V'4">
    <center><br><b>NO TIENES ANUNCIOS REGISTRADOS </b><br>
    </font>
    </center>"
} #id 1
else
{ #else 1
    print "<hr>";
    $num=1;
    while(%cat = $dbh->dbnextrow(1))
    { # while 1
        if ($cat{no_cat}=1) { $tblaux="AESCOLAR"; }
        if ($cat{no_cat}=4) { $tblaux="DIVERSOS"; }
        if ($cat{no_cat}=7) { $tblaux="EQUIPO"; }
        if ($cat{no_cat}=8) { $tblaux="MASCOTA"; }
        if ($cat{no_cat}=11) { $tblaux="VEHICULO"; }
        if ($cat{no_cat}=12) { $tblaux="MUSICA"; }
        print "<form name=V'inserta$num' method=V'POST'>";
        print "<table width=100%>";
        print "<tr><td>";
        $numero=$cat{num};
        $numcatego=$cat{no_cat};

        $dbh4 = new Sybase DBlib $log, $pas, $ser, 'ocasion';
        $dbh4->dbcmd("select tipo
        from CATEGORIA
        where no_cat=$numcatego ")

        $dbh4->dbsqlc-ec
        $dbh4->dbresults
        $dbh4->DBROWS
        while(%cat3 = $dbh4->dbnextrow(1))
        { #while catego
            $tanuncio=$cat3{tipo};
            print "CATEGORIA $cat3{tipo} <BR>";
        } #while catego
        print "CLAVE DEL ANUNCIO $numero<br>";
        print "<input type=V'hidden' name=V'tanuncio' value=V'$numero'>";
        print "<input type=V'hidden' name=V'usuario' value=V'$clave'>";
        print "<input type=V'hidden' name=V'tanuncio' value=V'$tanuncio'>";
        $usuario=$cat{no_usuario};
        print "CLAVE DEL USUARIO $usuario<br>";
        $articulo=$cat{articulo};
        print "ARTICULO $articulo<br>";
        $descripcion=$cat{descrip};
        print "DESCRIPCION $descripcion<br>";
        $precio=$cat{precio};
        print "PRECIO $precio<br>";
        $foto=$cat{foto};
        print "FOTO $foto<br>";
        $fecha=$cat{fecha};
        print "FECHA DE VENCIMIENTO $fecha<br>";
#-----
        para ver la diferencia de fechas
        $dbhfec = new Sybase DBlib $log, $pas, $ser, 'ocasion';
        $dbhfec->dbcmd("select diavenc=(datediff(day,getdate()),fecha), renovacion
        from $tblaux
        where num=$numero and no_usuario=V'$usuario' ");
        $dbhfec->dbsqlc-ec
        $dbhfec->dbresults
        $dbhfec->DBROWS
        while(%catfec = $dbhfec->dbnextrow(1))
        { #while catego
            $dfec=$catfec{diavenc};
            $renova=$catfec{renovacion};
            # renova es el numero de renovaciones que lleva
            # print "diferencia de fechas $dfec";
        } #while catego
#-----
        if($foto eq "" || $foto eq " "){
            print "<td><td> &nbsp;";
        } else {
            print "</td><td><img src=V' /FOTOS/$foto' width=V'130' height=V'130'>";
            print "<input type=V'hidden' name=V'foto' value=V'$foto'>";
        }
        print "</td></tr>";
        print "<tr><td colspan=2><center>";
        if($dfec<4 && $renova<2) {
            print "<input type=V'button' name=V'insertar' value=V'Renovar' OnClick=V'JavaScript:inserta$num.action=usuario-
            renueva pf:inserta$num submit(1).V'> &nbsp;";
        }
    }
}

```

```

) #! diffe<4
print "<input type='button' name='Borrar' value='Borrar' Onclick='JavaScript.inserta$num action='usuario-
borra pf.inserta$num submit()'>"
print "</center></td></tr>"
print "</table>"
print "<hr></form>"
$num++
} #while 1
$num=1
while(%cat = $dbh2->dbnextrow(1))
{ # while 2
if ($cat{no_cat}==2) { $tablaux="ALOJAMIENTO"; }
if ($cat{no_cat}==6) { $tablaux="ENSEÑANZA"; }
print "<form name='inserta$num' method='POST'>";
print "<table width=100%>";
print "<tr><td>";
$numcatego=$cat{no_cat}.

$dbh4 = new Sybase DBLib $log, $pas, $ser, 'ocasion';
$dbh4->dbcmd("select tipo
from CATEGORIA
where no_cat=$numcatego");
$dbh4->dbsqlxexec.
$dbh4->dbresults
$dbh4->DBROWS
while(%cat3 = $dbh4->dbnextrow(1))
{ #while catego
print "CATEGORIA $cat3{tipo} <BR>";
$numuncio=$cat3{tipo}.
} #while catego
$numero=$cat{num}.
print "CLAVE DEL ANUNCIO $numero<br>";
print "<input type='hidden' name='nanuncio' value='$numero'>";
print "<input type='hidden' name='lanuncio' value='$lanuncio'>";
$numario=$cat{no_usuario}
print "<input type='hidden' name='nusuario' value='$usuario'>";
print "<input type='hidden' name='foto' value='x1'>";
print "CLAVE DEL USUARIO $usuario<br>";
$articulo=$cat{articulo}.
print "ARTICULO $articulo<br>";
$descripcion=$cat{descrip}.
print "DESCRIPCION $descripcion<br>";
$precio=$cat{precio}.
print "PRECIO $precio<br>";
$fecha=$cat{fecha}.
print "FECHA DE VENCIMIENTO $fecha<br>";
#----- para ver la diferencia de fechas
$dbh5 = new Sybase DBLib $log, $pas, $ser, 'ocasion'.
$dbh5->dbcmd("select diavenc=(datediff(day,getdate(),fecha)), renovacion
from $tablaux
where num=$numero and no_usuario='$usuario'");
$dbh5->dbsqlxexec.
$dbh5->dbresults
$dbh5->DBROWS
while(%cat = $dbh5->dbnextrow(1))
{ #while catego
$diff=$cat{diavenc}
$renova=$cat{renovacion}
} #while catego
#-----
print "</td></td> &nbsp;";
print "</td></tr>";
print "<tr><td colspan=2><center>";
if($diff < 4 && $renova < 2) {
print "<input type='button' name='Insertar' value='Renovar' Onclick='JavaScript.inserta$num action='usuario-
renueva pf.inserta$num submit()'>";
} #! diffe<4
print "<input type='button' name='Borrar' value='Borrar' Onclick='JavaScript.inserta$num action='usuario-
borra pf.inserta$num submit()'>";
print "</center></td></tr>";
print "</table>";
print "<hr></form>";
$num++
} #while 2
print "<br><br></form>";
} #else 1
print "<br><font face='Arial, Verdana, Helvetica, sans-serif' size='2'><center><a
href='http://www.deocasion.unam.mx/index.html'>Volver a la p&agrave;gina principal </a></center></font>";
require "inf.txt";

```

```

#!/usr/bin/perl
$|=1. #para agilizar la respuesta

use CGI;
require "var.pl";
use Sybase DBlib;

$query=new CGI;

#----cachamos los datos del usuario-----
$NOMBRE=$query->param('nombre');
$PELLIDOS=$query->param('apellidos');
$TELEFONO=$query->param('telefono');
$LADA=$query->param('lada');
$CORREO=$query->param('correo');
#-----
$proceso=$query->param('oculto');
$fecha=$query->param('fec');
$mmandafoto=$query->param('mmandafoto');
$NUMERO=$query->param('numero');
#----cachamos los datos del articulo-----
$TABLA=$query->param('tabla');
$NO_CAT=$query->param('no_cat');
$SUBCATEGORIA=$query->param('no_subcat');
$ARTICULO=$query->param('articulo');
$DESCRIPCION=$query->param('descripcion');
$PRECIO=$query->param('precio');
$FOTO=$query->param('foto');
$mmandafoto=$query->param('mmandafoto');
#juntamos categoria, subcategoria y proceso para identificar cada aviso
$idaviso="$NO_CAT""$SUBCATEGORIA""$proceso";
#-----
$DESCRIPCION =~ s/\n /g;
$DESCRIPCION =~ s/\$341/a/g;
$DESCRIPCION =~ s/\$341/a/g;
$DESCRIPCION =~ s/\$351/e/g;
$DESCRIPCION =~ s/\$355/i/g;
$DESCRIPCION =~ s/\$363/o/g;
$DESCRIPCION =~ s/\$372/u/g;
$DESCRIPCION =~ s/\$301/A/g;
$DESCRIPCION =~ s/\$311/E/g;
$DESCRIPCION =~ s/\$315/I/g;
$DESCRIPCION =~ s/\$323/O/g;
$DESCRIPCION =~ s/\$332/U/g;
$DESCRIPCION =~ s/\$361A321/g;
$ARTICULO =~ s/\n /g;
$ARTICULO =~ s/\$341/a/g;
$ARTICULO =~ s/\$341/a/g;
$ARTICULO =~ s/\$351/e/g;
$ARTICULO =~ s/\$355/i/g;
$ARTICULO =~ s/\$363/o/g;
$ARTICULO =~ s/\$372/u/g;
$ARTICULO =~ s/\$301/A/g;
$ARTICULO =~ s/\$311/E/g;
$ARTICULO =~ s/\$315/I/g;
$ARTICULO =~ s/\$323/O/g;
$ARTICULO =~ s/\$332/U/g;
$ARTICULO =~ s/\$361A321/g;
$repetido=0;
$prepetido=0;
$linearep=0;
$linea2="$NOMBRE=$PELLIDOS=$TELEFONO=$LADA=$CORREO=$TABLA=$NO_CAT=$SUBCATEGORIA=$ARTICULO=$DESCRIPCION=$PRECIO=$FOTO=$NUMERO=$proceso=$fecha";
$linea2="$NOMBRE=$PELLIDOS=$TELEFONO=$LADA=$CORREO=$TABLA=$NO_CAT=$SUBCATEGORIA=$ARTICULO=$DESCRIPCION=$PRECIO=$FOTO=$NUMERO=$proceso=$fecha";
print "Content-type: text/html\n\n";
require "sup.txt";
print "$datos";
if($mandafoto eq "no"){
print "<font color=\W000099\><b>ESTOS SON LOS DATOS DE TU ANUNCIO</b></font><hr color=\W000099\>";
}
#busca que no haya una linea exactamente igual en el archivo
$repetido = `grep "$linea" usuario.txt | wc -l`;
#busca que no haya dos numeros de procesos iguales
$prepetido = `cut -d= -f14 usuario.txt | grep "$proceso" | wc -l`;
#busca dos lineas iguales pero con distinto numero de proceso
$linearep = `cut -d= -f13,15- usuario.txt | grep "$linea2" | wc -l`;
if($mandafoto eq "no" && $repetido==0 && $prepetido==0 && $linearep==0){ #1
&pondatos. #subrutina que pone los datos del usuario en el navegador
$ya=1;
} #1mandafoto eq no

```

```

if($repetido=0 && $prepetido=0 && $linearep=0){ #2
if ($mandafoto eq "si")
{ #3
$ya=1
print "<cp> La foto del art&acuteculo en cuesti&ocute;n se mostrara junto con tu anuncio <br>".
#-aqui mostramos la opcion para que envie la foto
print << "fotohtml".
<SCRIPT LANGUAGE="JavaScript">
function validamagen()
{
objeto=document forma proceso.
medir=objeto value length.
cuatro=4.
size=medir-cuatro.
if (objeto value == "")
{
alert("DEBE ESPECIFICAR AL MENOS UN ARCHIVO");
objeto focus().
return false
}
else
{
if (objeto value substring(size,medir) toLowerCase()!=".gif" &&
objeto value substring(size,medir) toLowerCase()!=".jpg")
{
alert("EL ARCHIVO NO ES VALIDO, VERIFIQUE").
objeto focus()
return false
}
}
}
function cambiaNombre()
{
var value=document forma proceso value.
(document forma upfile value=document forma upfile value+$proceso).
}
}
</SCRIPT>
<form name="forma" onSubmit="return validamagen()"
method="POST" enctype="multipart/form-data"
action=" /cgi-bin/uploadindice.pl" >
<br>
<center><h3>INTRODUCE LA INFORMACI&ocute;n QUE SE TE PIDE</h3></center>
<br>
<input type="text" value="Fotograf&acute;a a enviar" <br>
<input type="file" name="proceso" size="65">
<input type="hidden" name="oculto" value="$proceso">
<input type="hidden" name="numero" value="$NUMERO">
<input type="hidden" name="lineadatos" value="$linea">
<input type="submit" value="ENVIAR">
<input type="reset" value="LIMPIAR">
<br>
</form>
fotohtml
} #3cierra if si manda foto

if($ya = 0){ &pondatos. } #subrutina que pone los datos del usuario en el navegador
} #2f repetido
#pero si se va a mandar foto los datos se guardan hasta que se envie la foto
$NOMBRE=$s2. /lg
#para ello ponemos todos los datos en una sola linea
$linea="$NOMBRE=$APELIDOS=$TELEFONO=$LADA=$CORREO=$TABLA=$NO_CAT=$SUBCATEGORIA=$ARTICULO=$DESCRIPCION=$PRECIO=$FOTO=$NUMERO=$proceso=$fecha".
$linea2="$NOMBRE=$APELIDOS=$TELEFONO=$LADA=$CORREO=$TABLA=$NO_CAT=$SUBCATEGORIA=$ARTICULO=$DESCRIPCION=$PRECIO=$FOTO=$NUMERO=$proceso=$fecha".
$archivo="usuario.txt"
#pero primero validamos que no haya una linea gual
$repetido = grep "$linea" usuario.txt | wc -l
$prepetido = cut -d- -f14 usuario.txt | grep "$proceso" | wc -l
#busca dos lineas iguales pero con distinto numero de proceso
$linearep= cut -d- -f11-13-15- usuario.txt | grep "$linea2" | wc -l.
#despues abrimos el archivo y a&ntilde;adimos la linea al final
if ($repetido=0 && $prepetido=0 && $linearep=0 && $mandafoto eq "no")
{
open (USUARIO ">>$archivo").
print USUARIO
"$NOMBRE=$APELIDOS=$TELEFONO=$LADA=$CORREO=$TABLA=$NO_CAT=$SUBCATEGORIA=$ARTICULO=$DESCRIPCION=$PRECIO=$FOTO=$NUMERO=$proceso=$fecha".
close(USUARIO).
print "<br>".
print " <a href='http://www.deocasion.unam.mx/index.html'>Volver a la p&acut;gina principal.</a><cp>".
} else
}

```

```

if($mandafoto eq "si" && $repetido==0 && $prepetido==0 && $linearep==0){
    print "<hr>";
}
elsif($mandafoto eq "no" && $repetido==0 && $prepetido==0 && $linearep==0){
print "<font color='W0000991'><b>ESTOS SON LOS DATOS DE TU ANUNCIO</b></font><hr color='W0000991'> ";
    open (USUARIO ">>$archivo")
    print USUARIO
    $NOMBRE=$APELIDOS=$TELEFONO=$LADA=$CORREO=$TABLA=$NO_CAT=$SUBCATEGORIA=$ARTICULO=$DESCRIPCION
    =$PRECIO=$FOTO=$NUMERO=$proceso=$fecha;
    close(USUARIO)
    &pondatos;
    print "<br>";
    print "<a href='http://www.deocasion.unam.mx/index.html'>Volver a la pagina principal</a> <br> ";
}
}
else{
    # print "<font color='W0000991'><b>ESTOS SON LOS DATOS DE TU ANUNCIO</b></font><hr color='W0000991'> ";
    &pondatos;
    print "<ps>#3>Anuncio ya registrado<p></#3>";
    print "<a href='http://www.deocasion.unam.mx/index.html'>Volver a la página principal</a> <br>";
}
}
# y ejecutamos la funcion que pasa los datos a mayusculas que es mayusculas
&mayusculas;
require "int.txt";
##### subrutinas #####
sub &pondatos{ # rutina para poner los datos del usuario
    $desaux=(uc($DESCRIPCION));
    print "NOMBRE $NOMBRE<br>
    APELLIDOS $APELIDOS<br>
    TELEFONO $TELEFONO<br>
    E-MAIL $CORREO<p>
    ARTICULO $artaux<br>
    DESCRIPCION $desaux<br>
    PRECIO $PRECIO<br>
    FECHA $fecha<br>";
    print "<br><font color='red'>Recuerda que tu anuncio será publicado un día, a la hora, bil después de su registro </font>";
}
#-----
sub &mayusculas
{
    $archivo2 = "aux.txt";
    #aquí va el archivo usuario.txt
    $archivo = "usuario.txt";
    open (USUARIO, "$archivo");
    while ($linea=<USUARIO>){
        $hola="hola";
        $linea2= uc $linea;
        @datos = split(//, $linea);
        $correo = lc $datos[4];
        $foto = uc $datos[11];
        @datos2 = split(//, $linea2);
        $datos2[4]= $correo;
        $datos2[11]=$foto;
        $linea3= join(" ", @datos2);
        open (AUX1, ">>$archivo2");
        print AUX1 "$linea3";
        close(AUX1)
    }
    close(USUARIO);
    # aquí va el nombre de usuario.txt
    system("cp aux.txt usuario.txt");
    system("rm aux.txt");
}

#/usr/bin/perl
$|=1 #para agilizar la respuesta

require "var.pl";
use Sybase DBlib;

use CGI;
$query=new CGI;

$letra = $query->param('letra'); #tiene la letra por la que va a buscar
$subcat = $query->param('subcat');
$letram = (uc $letra); # tiene la letra por que va a buscar en mayuscula
$catego = $query->param('catego');

```

```

$tabla = $query->param('tabla');
$inicial = $query->param('inicial');
$contareg = $query->param('contareg');
$final = $query->param('final');

```

```

if ($subcat eq "AUTOMOVILES")
{
    $numsub = 23;
}
if ($subcat eq "MOTOCICLETAS")
{
    $numsub = 24;
}
if ($subcat eq "BICICLETAS")
{
    $numsub = 25;
}

```

```

$dbh = new Sybase DBlib $log, $pas, $ser, 'aviso';
$dbh->dbcmd(" select * from $tabla where no_subc = $numsub and articulo like '[$]tram,$[tra]%' ");
$dbh->dbselectec;
$dbh->dbresults;
$dbh->DBROWS;
$status=$dbh->DBROWS;
print "Content-type text/html\n";
require "sup.txt";
if($status=0){
    # si no hay anuncios se pone la plantilla primer_parte pone la parte de arriba de la plantilla
    # enmedio se ponen los resultados y segunda_parte pone la parte de abajo
    require "sup.txt";
    print "<h2 align='center'><b><font face='Arial,Helvetica,sans-serif'>";
    print "No hay anuncios registrados para esta categoria ";
    print "</font></b></h2>";
    require "inf.txt";
}
if($status==0)
else{
    #if($status=0)
    require "sup.txt";
    $cuantosreg=0;
    while(%cat = $dbh->dbnextrow(1))
    {
        # while 1
        $cuantosreg++
        $no_usuario=$cat{no_usuario};
        if($no_usuario > 999999){
            $tabla2 = "ESTUDIANTE";
        }
        else{
            $tabla2 = "TRABAJADOR";
        }
        if (($inicial<=$contareg) && ($contareg<$final+1))
        {
            ##if contador
            print "<table width=98%><tr><td ALIGN='LEFT'>";
            <font face='Arial, Verdana, Helvetica, sans-serif' size='4'>
            <b>No $cuantosreg</b></td></tr>";
            <font face='Arial, Verdana, Helvetica, sans-serif' color='ff000099' size='3'>
            <b>Art&iaculo culo </b></font>
            <font face='Arial, Verdana, Helvetica, sans-serif' size='4'>
            $cat{articulo} <br>
            <b>Descripci&ocauton </b>
            $cat{descrip} <br>
            <b>Precio </b>
            $cat{precio}</font><br>";
            #Metemos a la persona a contactar
            print "<br>";
            <font face='Arial, Verdana, Helvetica, sans-serif' size='4' color='ffCC33'>
            <b>Contactar con </b></font>";
            $dbh1=new Sybase DBlib $log, $pas, $ser, 'aviso';
            $dbh1->dbcmd("select * from USUARIO where no_usuario='$no_usuario'");
            $dbh1->dbselectec;
            $dbh1->dbresults;
            while(%usu = $dbh1->dbnextrow(1))
            {
                $nombre=$usu{nombre};
                $apellido=$usu{apellido};
                $telefono=$usu{telefono};
                $mail=$usu{mail};
                print " <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>";
                $nombre $apellido <br>
                <b>Telefono </b> $telefono <br>
                <b>E-mail </b><a href='mailto: $mail'> $mail</a> <br></font>";
            }
            #cierra while
            print "</td>";
        }
    }
}

```



```

#si hay foto se muestra
$vacio="";
if ($cat(foto) ne "" && $cat(foto) ne "" )
{
  $foto= $cat(foto);
  print "
    <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
    <td ALIG=1>1right'>
    <img src='1' /FO.TOS/$foto' width='130' height='130'>
    </img></font></td></tr></table> <hr>";
}
else{
  print "<td></td></tr></table><hr>";
}
}### cierra el contador Si trae mas de cinco registros entra al siguiente elseif
elseif ($contareg==$final+1)
{ ##dos
  print "<table width='90%' border='1' align='center'>";
  $inicial = $inicial+5;
  $final = $final+5
  print "<tr><td><div align='center'> <a href='1' /cgi-
  bin/listabc.pl?otra=$otra&subcal=$subcal&catogo=$catogo&tabla=$tabla&inicial=$inicial&final=$final&contareg=1'>
  Siguiete</font></a>
  </div>
  </td>
  </tr>
  <div align='center'>";
  $final = $final-5;
  #----- para poner anterior-----
  if($final>5){
    $inicial=$inicial-10; #aquí era 5
    $final=$final-5;
    print "<a href='1' /cgi-
    bin/listabc.pl?otra=$otra&subcal=$subcal&catogo=$catogo&tabla=$tabla&inicial=$inicial&final=$final&contareg=1'>
    <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
    <center>Anterior</center></font>
    </font>
    </a>";
  }
  else
  { print " "; }
  #----- para poner anterior-----
  print "</div>";
  print "</td>";
  print "</tr>";
  print "</table>";
} ##dos
$contareg++;
} #cierra el primer while

if (($cuantosreg>5 && $cuantosreg<$final) || ($cuantosreg>5 && $cuantosreg==$final))
{
  $inicial=$inicial-5;
  $final=$final-5;
  print"<a href='1' /cgi-
  bin/listabc.pl?otra=$otra&subcal=$subcal&catogo=$catogo&tabla=$tabla&inicial=$inicial&final=$final&contareg=1'>
  <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
  <center> Anterior</center>
  </font></a>";
}
print "<table width='90%' border='1' align='center'>
  <tr><td>
  <div align='left'>
  <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
  N&uacute;mero de anuncios encontrados de este tipo <b>$cuantosreg</b>
  </div>
  </td><td nowrap>
  <div align='right'>
  <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
  <a href='http://www.deocasion.unam.mx/index.html'>Regresa a la pagina principal</a>
  </font>
  </div></td></tr><tr>
  <td align='center'>
  # y aquí debe de cerrar el renglon central y poner lo que falta de la pagina
} #cierra else de if($status==D)
require "inf.txt".

```

listas2-libre.pl

```

#1usr/bin/perl
$|=1. #para agilizar la respuesta

```

```

require "var.pl"
use Sybase DBlib

use CGI
$query=new CGI
$categoria = $query->param("categoria")
$titulogl = $query->param("titulogl")
$inicial = $query->param("inicial")
$contareg = $query->param("contareg")
$final = $query->param("final")
$dos=2
$cuatro=4

#-----
# el siguiente if es para hacer la consulta directa si la categoria E
# escogida es alojamiento o diversos que no tienen subcategorias
if ( ($categoria eq $dos) || ($categoria eq $cuatro) )
  { #si no tiene subcategorias
    #hace la consulta directamente
    if ($categoria eq $dos)
      {
        $tabla="ALOJAMIENTO"
        $titulogl="alojamiento_1_gl"
      }
    else
      {
        $tabla="DIVERSOS"
        $titulogl="diversos_1_gl"
      }
    $dbh = new Sybase DBlib $log, $pas, $ser, 'aviso'
    $dbh->dbcmd("select * from $tabla")
    $dbh->dbselect
    $dbh->dbresults
    $dbh->DBROWS
    $dbh->DBCOUNT
    $status=$dbh->DBROWS

    $dbhc = new Sybase DBlib $log, $pas, $ser, 'aviso'
    $dbhc->dbcmd("select COUNT(*) from $tabla")
    $dbhc->dbselect
    $dbhc->dbresults
    $renglones=$dbhc->dbresults

    print "Content-type: text/html\n\n"
    print "HTML "
    print "<HEAD><TITLE>AVISO DE OCASION =Subcategorias=</TITLE></HEAD>\n"
    if($status = 0){
      # si no hay anuncios se pone la plantilla
      require "sup.txt"
      #-----celda central-----
      print " <h2 align='center'><b>
        <font face='Arial,Helvetica, sans-serif'>
          No hay anuncios registrados para esta categoria
        </font>
        </b></h2>"
      #-----celda central-----
      require "inf.txt"
    }
    else{ #si hay anuncios
      #el while que esta en la siguiente subrutina se ejecuta
      #de nuevo para mostrar los resultados por bloques
      require "sup.txt"
      $Resultados=Bloque
      require "inf.txt"
    } #si hay anuncios
  } #cierra if($categoria eq $dos || $categoria eq $cuatro)
#-----
else
  { ##### else que encierra la consulta a categorias con subcategorias
    $dbh = new Sybase DBlib $log, $pas, $ser, 'aviso'
    $dbh->dbcmd("select tipo from SUBCATEGORIA where no_cat=$categoria")
    $dbh->dbselect
    $dbh->dbresults
    print "Content-type: text/html\n\n"
    require "sup.txt"
    #aqui empieza el renglon y se meten los datos
    print "
      <font face='Arial, Verdana, Helvetica, sans-serif' size='4'>
      <center><b>S U B C A T E G O R I A S</b></center>
      <br>
      <form name='favis02' method='POST' action='.../cgi-bin/listas3-libre.pl'>"
  }

```

```

$numauxiliar=0,
while(%cat = $dbh->dbnextrow(1))
{
if($numauxiliar==0){
print "
<font face='Arial, Verdana, Helvetica, sans-serif' size='13'>
<input name='subcat1'type='radio' value='\$cat{1po}' checked> \$cat{1po}
<br><br>";
}
else{
print "
<font face='Arial, Verdana, Helvetica, sans-serif' size='13'>
<input name='subcat1'type='radio' value='\$cat{1po}'> \$cat{1po}
<br><br>";
}
$numauxiliar ++,
}
if ($categoria == 11)
{
print "<input name='alfabetico' type='checkbox' value='1'> Buscar por Indice Alfabetico <br><br>";
print "<input type='hidden' name='v_inicial' value='11'>";
print "<input type='hidden' name='v_final' value='15'>";
print "<input type='hidden' name='v_contareg' value='11'>";
print "<input type='submit' value='Siguiente'> <input type='reset' value='Limpiar'>";
print "<input type='hidden' name='v_categoria' value='\$categoria'>";
$dbbh->dbcmd("select tipo from CATEGORIA where no_cat=$categoria");
$dbbh->dbqtoex;
$dbbh->dbresults;
while(%tipocat = $dbh->dbnextrow(1))
{
if ($tipocat{1po} eq "Articulos Escolares")
{
$stabi="AESCOLAR";
$titulop3="artesco_1 gif";
}
elseif ($tipocat{1po} eq "Alojamiento") {
$stabi="ALOJAMIENTO";
$titulop3="artesco gif";
}
elseif ($tipocat{1po} eq "Bienes Raicos") {
$stabi="BIENESR";
$titulop3="bienesraicos_1 gif";
}
elseif ($tipocat{1po} eq "Diversos") {
$stabi="DIVERSOS";
$titulop3="diversos_1 gif";
}
elseif ($tipocat{1po} eq "Electrodomesticos") {
$stabi="ELECTROD";
$titulop3="electrodomesticos_1 gif";
}
elseif ($tipocat{1po} eq "Enseñanza") {
$stabi="ENSEÑANZA";
$titulop3="enseñanza_1 gif";
}
elseif ($tipocat{1po} eq "Equipo de Computo") {
$stabi="EQUIPO";
$titulop3="equicomp_1 gif";
}
elseif ($tipocat{1po} eq "Mascotas") {
$stabi="MASCOTA";
$titulop3="mascotas_1 gif";
}
elseif ($tipocat{1po} eq "Muebles") {
$stabi="MUEBLE";
$titulop3="muebles_1 gif";
}
elseif ($tipocat{1po} eq "Oficios") {
$stabi="OFICIO";
$titulop3="artesco gif";
}
elseif ($tipocat{1po} eq "Vehiculos") {
$stabi="VEHICULO";
$titulop3="vehiculos_1 gif";
}
elseif ($tipocat{1po} eq "Musica") {
$stabi="MUSICA";
$titulop3="musica_1 gif";
}
}
else { print "Opcion no valida!"; }
} #cierra while 1

```

```

print "<input type='hidden' name='Tabla1' value='\$Tabla1'>"
print "<input type='hidden' name='Titulo1' value='\$Titulo1'>"
print "</form>"
#aquí va el resto del código de la plantilla
&inferior.
} ##else que encierra las sub
#####
#####SUBROUTINAS#####
#####
sub ResultadosBloque
{
    $cuantos=0
    while(%cat = $dbh->dbnextrow(1)) #while 1
    {
        $cuantos+ .
        $row_num = $dbh->DBCOUNT.
        $no_usuario = $cat(no_usuario).
        if($no_usuario > 999999){
            $tabla2 = "USUARIO".
        }
        else{
            $tabla2 = "USUARIO".
        }
    }
    if (($Sincial = $contareg) && ($contareg < $final+1))
    {
        ##uno
        print "<table width=98%><tr><td ALIGN='LEFT'>"
        <font face='Arial, Verdana, Helvetica, sans-serif' size='4'>
        <b>No $cuantos </b></font>.
        if($tabla eq "DIVERSOS"){
            print "
            <font face='Arial, Verdana, Helvetica, sans-serif' size='4' color='W000099'><b>Art&iacute;culo</b>
            </b></font>
            <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
            $cat(articulo)</font> <br>".
        }
        if($tabla eq "ALOJAMIENTO"){
            print <font face='Arial, Verdana, Helvetica, sans-serif"
            size='4' color='W000099'><b>Alojamiento </b></font>
            <font face='Arial, Verdana, Helvetica, sans-serif"
            size='3'>
            $cat(articulo)</font> <br>".
        }
        print "
            <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
            <b>Caracter&iacute;sticas </b></font>
            <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
            $cat(descrip) <br>
            <b>Precio </b>
            $cat(precio) <br>
            </font>
            <br>
            <font face='Arial, Verdana, Helvetica, sans-serif' size='4' color='WFCC33'>
            <b>Contactar con </b></font> ".
        $dbh2 = new Sybase DBlib $log, $pas, $ser, 'aviso'.
        $dbh2->dbcmd("select * from USUARIO
            where no_usuario='\$no_usuario'").
        $dbh2->dbsql>ec.
        $dbh2->dbresults
        while(%cat2 = $dbh2->dbnextrow(1)) #while 2
        {
            $nombre=$cat2(nombre).
            $apellido=$cat2(apellido).
            $telefono=$cat2(telefono).
            $mail=$cat2(mail)
            print "
            <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
            $nombre $nombre2 $apellido $apellido2<br>
            <b>Tel&iacute;fono </b> </b> $telefono $telefono2<br>
            <b>E-mail </b><a href='mailto:$mail $mail2'>$mail $mail2</a></font><br>
            <tr>".
        } #cierra while
        print "</td>".
        #si hay foto se muestra
        $foto=""
        if ($cat(foto) ne "" && $cat(foto) ne "")
        {
            $foto = $cat(foto)
            print "
            <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
            <td ALIGN='right'>
            <img src=' /OTOS/$foto' width='130' height='130'>

```

```

</img></font></td></tr></table>".
}
else{
print "<td></td></tr></table>".
}
}##uno
elseif ($contareg== $final+1)
{ ##dos
$inicial = $inicial+5,
$final = $final+5,
#asigno la liga anterior a una variable
$sigue="<a href='\"/cgi-bin/listas2-libre.pl?categoria=$categoria&inicial=$inicial&final=$final&contareg=1\"> <font face='\"Arial, Verdana, Helvetica, sans-serif\" size='\"3\"><center>Siguiente</center></font></a>".
$final=$final-5,
#---- para poner anterior-----
if ($final>5)
{
$inicial=$inicial-10, #aque era 5
$final=$final-5,
$atras="<a href='\"/cgi-bin/listas2-libre.pl?categoria=$categoria&inicial=$inicial&final=$final&contareg=1\"><font face='\"Arial, Verdana, Helvetica, sans-serif\" size='\"3\"><center>Anterior</center></font></a>";
}
else
{
}
#---- para poner anterior si es el ultimo-----
print "<table width='\"90%\" border='\"0\" align='\"center\">".
print "<tr><td><p align='\"center\">".
print "$atras </p></td>".
print "<td><p align='\"center\">".
print "$sigue </p></td></tr></table>".
}##dos
$contareg++;
} #cierra while 2(%cat = $dbh->dbnextrow(1))
if ( ($cuantos>5 && $cuantos<$final) || ($cuantos>5 && $cuantos==$final) )
{
$inicial=$inicial-5,
$final=$final-5,
print"<a href='\"/cgi-bin/listas2-libre.pl?categoria=$categoria&inicial=$inicial&final=$final&contareg=1\">
<font face='\"Arial, Verdana, Helvetica, sans-serif\" size='\"3\">
<center>Anterior</center>
</font>
</a>".
}
print "<table width='\"90%\" border='\"0\" align='\"center\">
<tr><td>
<p align='\"left\">
<font face='\"Arial, Verdana, Helvetica, sans-serif\" size='\"3\">
N&uacute;mero de anuncios encontrados de esta tipo: <b>$cuantos</b>
</p>
</td><td nowrap>
<p align='\"right\">
<font face='\"Arial, Verdana, Helvetica, sans-serif\" size='\"3\">
<br>
</font>
</p>
</td></tr>
</table>".
}

```

#### listas3-libre.pl

```

#!/usr/bin/perl
$|=1. #para aglizar la respuesta

require "var.pl";
use Sybase DBlib;

use CGI;
$query=new CGI;
$subcatego = $query->param('subcat');
$alfabetico = $query->param('alfabetico');
$catego = $query->param('categoria');
$tabla = $query->param('tabla');
$titulos = $query->param('titulos');
$inicial = $query->param('inicial');
$contareg = $query->param('contareg');
$final = $query->param('final');

print "Content-type text/html\n\n";
if ($alfabetico no "si")
{ ##a
if ($subcatego eq "CALCULADORAS" {

```

```

$numsub=1.
}
elseif ($subcatego eq "PAPELERIA") {
$numsub=2.
}
elseif ($subcatego eq "LIBROS") {
$numsub=3.
}
elseif ($subcatego eq "VENTA/RENTA DE CASAS") {
$numsub=4.
}
elseif ($subcatego eq "VENTA/RENTA DE DEPTOS") {
$numsub=5.
}
elseif ($subcatego eq "VENTA/RENTA DE TERRENOS") {
$numsub=6.
}
elseif ($subcatego eq "LAVADORAS") {
$numsub=7.
}
elseif ($subcatego eq "COCINAS") {
$numsub=8.
}
elseif ($subcatego eq "REFRIGERADORES") {
$numsub=9.
}
elseif ($subcatego eq "APARATOS DE SONIDO") {
$numsub=10.
}
elseif ($subcatego eq "CLASES_PARTICULARES") {
$numsub=11.
}
elseif ($subcatego eq "CURSOS") {
$numsub=12.
}
elseif ($subcatego eq "COMPUTADORAS") {
$numsub=13.
}
elseif ($subcatego eq "PERIFERICOS") {
$numsub=14.
}
}
elseif ($subcatego eq "MUEBLES PARA COMPUTADORAS") { #cambiar por muebles
$numsub=15.
}
elseif ($subcatego eq "PERROS") {
$numsub=16.
}
}
$numsub=17.
}
elseif ($subcatego eq "GATOS") {
}
elseif ($subcatego eq "OTROS" && $tabla eq "MASCOTA") {
}
elseif ($subcatego eq "HOGAR") {
}
elseif ($subcatego eq "OFICINA") {
}
elseif ($subcatego eq "OFERTA DE EMPLEO") {
}
elseif ($subcatego eq "OFERTA DE SERVICIOS") {
}
}
elseif ($subcatego eq "AUTOMOVILES") {
}
}
elseif ($subcatego eq "MOTOCICLETAS") {
}
}
elseif ($subcatego eq "BICICLETAS") {
}
}
elseif ($subcatego eq "ACCESORIOS") {
}
}
elseif ($subcatego eq "INSTRUMENTOS") {
}
}
elseif ($subcatego eq "DISCOS") {
}
}
$numsub=24.
}
$numsub=25.
}
$numsub=26.
}
$numsub=27.
}
$numsub=28.
}

```

```

    }
    elseif ($subcatego eq "REPRODUCTORES") {
        $numsub=29.
    }
    elseif ($subcatego eq "OTROS" && $staba eq "AESCOLAR") {
        $numsub=30.
    }
    elseif ($subcatego eq "OTROS" && $staba eq "ENSENANZA") {
        $numsub=32.
    }
    elseif ($subcatego eq "OTROS" && $staba eq "EQUIPO") {
        $numsub=33.
    }
    elseif ($subcatego eq "OTROS" && $staba eq "MUSICA") {
        $numsub=36.
    }
    elseif ($subcatego eq "OTROS" && $staba eq "VEHICULO") {
        $numsub=35.
    }
    else { print "Opcion no valida x $subcatego x". }
    $dbh = new Sybase DBlib $log, $pas, $ser, $ocasion".
    $dbh->dbcmd("select * from $staba where no_subc = $numsub").
    $dbh->dbsqlrec.
    $dbh->dbresults
    $dbh->DBROWS
    $status=$dbh->DBROWS.
    if($status==0){
        require "sup.txt".
        print "<h2 align='center'><b><font face='Arial,Helvetica,sans-serif'>".
            print " No hay anuncios registrados para esta categoria ".
            print "</font></b></h2>".
        require "int.txt".
        } #if($status==0)
        else{ #if($status!=0)
            require "sup.txt".
            #-----Resultados por bloques-----
            $cuntosreg=0
            while(%cat = $dbh->dbnextrow(1))
            { # while 1
                $cuanlosreg++
                # print "<br><br>".
                $no_usuario=$cat[no_usuario].
                if($no_usuario > 999999){
                    $staba2 = "ESTUDIANTE".
                }
                else{
                    $staba2 = "USUARIO".
                }
                if (($inicial=$contareg) && ($contareg < $final+1))
                { #### contador
                    print "<table width=98%><tr><td ALIGN='LEFT'>".
                        <font face='Arial, Verdana, Helvetica, sans-serif' size='4'>
                        <b>No $cuanlosreg</b></font><br>
                        <font face='Arial, Verdana, Helvetica, sans-serif' color='1'><b>00000991 size='3'>
                        <b> Ar&iacute,culo </b></font>
                        <font face='Arial, Verdana, Helvetica, sans-serif' size='3'>
                        $cat[articulo] <br>
                        <b> Descripción&ocute,n </b>
                        $cat[descrip] <br>
                        <b> Precio </b>
                        $cat[precio]</font><br>".
                    #Metemos a la persona a contactar
                    print "
                    <br>
                    <font face='Arial, Verdana, Helvetica, sans-serif' size='4' color='1'><b>FFCC33'>
                    <b> Contactar con </b></font>".
                    $dbh1=new Sybase DBlib $log, $pas, $ser, $ocasion".
                    $dbh1->dbcmd("select * from USUARIO
                    where no_usuario='\$no_usuario'").
                    $dbh1->dbsqlrec.
                    $dbh1->dbresults
                    while(%usu = $dbh1->dbnextrow(1))
                    {
                        $nombre=$usu[nombre].
                        $nombre2=$usu[nombre_1].
                        $apellido=$usu[apellido].
                        $apellido2=$usu[apellido_1].
                        $telefono=$usu[telefono].
                        $telefono2=$usu[telefono_1].
                        $mail=$usu[mail].
                        $mail2=$usu[mail_1].
                        print"

```

```

<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
Nombre $nombre2 $apellido $apellido2<br>
<b>Telefono </b></b> Telefono $telefono2 <br>
<b>E-mail </b><a href="mailto: $mail $mail2">$mail $mail2</a><br></font>;
#Termina persona a contactar
} #Cierra while
print "</td>";
#si hay foto se muestra
$vacio="";
if ($cat{foto} ne "" && $cat{foto} ne "" )
{
$foto= $cat{foto},
print "
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<td ALIGN="right">

</img></font></td></tr></table> <tr>;"
}
} else{
print "<td></td></tr></table><tr>";
}
}### Cierra if contador. Si trae mas de cinco registros entra al siguiente elsif
elsif ($contareg=$final+1)
{
##dos
$inicial= $inicial+5,
$final= $final+5,
$$sigue="<a href=" /cgi-bin/listas3-
libre.pl?subcat=$subcat&categoria=$cat&tabla=$tabla&inicial=$inicial&final=$final&contareg=1&titulos=$titulos">Siguiete</a>;"
$final=$final-5
#----- para poner anterior-----
if ($final>5)
{
$inicial=$inicial-10, #aqui era 5
$final=$final-5
$atras="<a href=" /cgi-bin/listas3-
libre.pl?subcat=$subcat&categoria=$cat&tabla=$tabla&inicial=$inicial&final=$final&contareg=1">Anterior</a>;"
}
} else
{
print "" ;
#-- para poner anterior cuando ya es el ultimo-----
print "<table width="90%" border="0" align="center">;"
print "<tr><td><p align="center">;"
print "$atras </p></td>;"
print "<td><p align="center">;"
print "$sigue </p></td></tr></table>;"
} ##dos
$contareg++;
} #Cierra el primer while
if ( ($Cuantosreg>5 && $Cuantosreg<$final) || ($Cuantosreg==5 && $Cuantosreg==$final) )
{
$inicial=$inicial-5
$final=$final-5,
print"<a href=" /cgi-bin/listas3-
libre.pl?subcat=$subcat&categoria=$cat&tabla=$tabla&inicial=$inicial&final=$final&contareg=1&titulos=$titulos"><center>Anterior
</center></a>;"
}
print "<table width="90%" border="0" align="center">
<tr><td>
<div align="left">
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
N&uacute;mero de anuncios encontrados de este tipo: <b>$Cuantosreg </b>
</font>
</div>
</td><td nowrap>
<div align="right">
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
</font>
</div>
</td></tr>
<br>
</table>;"
# y aqui debe de cerrar el rengion central y poner lo que falta de la pagina
#----- fin resultados por bloque-----
require "inf.txt";
} #Cierra else de if($status==0)
} ##a
else
{
##b
require "inf.txt",
$tabla_indice,
require "inf.txt",
} ##b

```



##### SUBRUTINAS#####

```
sub tabla_indice
{
print << "letras".
<p>Escanee la letra para buscar el modelo o marca </p>
<p><br>
</p>





```

```

</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=o&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">O</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=p&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">P</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=q&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">Q</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=r&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">R</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=s&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">S</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=t&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">T</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=u&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">U</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=v&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">V</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=w&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">W</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=x&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">X</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=y&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">Y</a></div>
</td>
<div align="center"><a href=" /cgi-bin/listabc-
libre pi?letra=z&subcat=$subcatego&catago=$catego&tabla=$tabla&inicial=1&final=5&contareg=1">Z</a></div>
</td>
</tr>
</table>
</td>
</tr>
</table>

```

```

letras
}

```

pruebas-libre.pl

```

#!/usr/bin/perl
$|=1, #para agilizar la respuesta

require "home/users00/ocasion/cgi-bin/vr.pl";
use Sysbase DBlib;

use CGI;
$query=new CGI;

$fecha = date '+DATE %m-%d-%Y' | cut -d' ' -f2-4';

$dbh = new Sysbase DBlib $log, $pas, $ser, 'ocasion';
print "Content-type: text/html\n\n";
require "sup.pl";
#si la diferencia entre la fecha(de vencimiento) y la de hoy es 1
$dbh->dbcmd("select
dia1 = (datediff(day,fecha,getdate()))
from AESSCLAR
where (datediff(day,fecha,getdate()))=0
union
select dia2 = (datediff(day,fecha,getdate()))
from ALJAMIENTO
where (datediff(day,fecha,getdate()))=0
union
select dia3 = (datediff(day,fecha,getdate()))

```

```

from DIVERSOS
where (datediff(day.fecha,getdate()))>0
union
  select dia4 = (datediff(day.fecha,getdate()))
from ENSEANZA
where (datediff(day.fecha,getdate()))>0
union
  select dia5 = (datediff(day.fecha,getdate()))
from EQUIPO
where (datediff(day.fecha,getdate()))>0
union
  select dia6 = (datediff(day.fecha,getdate()))
from MASCOTA
where (datediff(day.fecha,getdate()))>0
union
  select dia7 = (datediff(day.fecha,getdate()))
from MUSICA
where (datediff(day.fecha,getdate()))>0
union
  select dia8 = (datediff(day.fecha,getdate()))
from VEHICULO
where (datediff(day.fecha,getdate()))>0
)
)
$dbh->dbselect uc.
$dbh->dbresults;
$dbh->DBROWS
$status=$dbh->DBROWS.
if ($status=0)
{
  print "
  <br><br><br>          <br><font face='Arial, Verdana, Helvetica, sans-serif' size='14'>
  <center><br><b>LA BASE DE DATOS NO HA SUFRIDO MODIFICACIONES</b><br>
  </font>
  <font face='Arial, Verdana, Helvetica, sans-serif' size='13'>
  <br>ningun aviso excedia su fecha limite de publicacion.<br>
  </font></center>".
}
else
{
  $CATEGORIA="AESCOLAR".
  $NUMERO=1.
  &actualizacion ($CATEGORIA, $NUMERO).
  $CATEGORIA="ALojAMIENTO".
  $NUMERO=2.
  &actualizacion ($CATEGORIA, $NUMERO).
  $CATEGORIA="DIVERSOS".
  $NUMERO=3.
  &actualizacion ($CATEGORIA, $NUMERO).
  $CATEGORIA="ENSEANZA".
  $NUMERO=4.
  &actualizacion ($CATEGORIA, $NUMERO).
  $CATEGORIA="EQUIPO".
  $NUMERO=5.
  &actualizacion ($CATEGORIA, $NUMERO).
  $CATEGORIA="MASCOTA".
  $NUMERO=6.
  &actualizacion ($CATEGORIA, $NUMERO).
  $CATEGORIA="MUSICA".
  $NUMERO=7.
  &actualizacion ($CATEGORIA, $NUMERO).
  $CATEGORIA="VEHICULO".
  $NUMERO=8.
  &actualizacion ($CATEGORIA, $NUMERO).
  print "<br><br><br>
  <font face='Arial, Verdana, Helvetica, sans-serif' size='14'>
  <center><br><b>LA BASE DE DATOS HA SIDO MODIFICADA</b><br>
  </font>
  <font face='Arial, Verdana, Helvetica, sans-serif' size='13'>
  <br>Los avisos que excedian su fecha limite de publicacion, han sido borrados<br>
  </font></center>".
}
print "<br><br><br><font face='Arial, Verdana, Helvetica, sans-serif' size='12'><center><a
href='http://www.deccasion.unam.mx/index.html'>Volver a la pagina principal </a></center></font>".
require "inf.txt".
sub actualizacion
{
  local ($var).
  $CATEGORIA=$_[0].
  $NUMERO=$_[1].
  $conex="&$dbh"&$NUMERO".
  $conex = new Sybase DBlib $log, $pas, $$ser, 'ocasion'.
  $conex->dbcmd("select $CATEGORIA no_usuario, USUARIO no_avisos as avisos, count(*) as menos

```

```

from $CATEGORIA, USUARIO
where (datediff(day,$CATEGORIA fecha,getdate()))>0
and $CATEGORIA no_usuario like USUARIO no_usuario
group by $CATEGORIA no_usuario
having $CATEGORIA no_usuario like USUARIO no_usuario";
$conex->dbsqlxec.
$conex->dbresults.
$conex->DBROWS.
$status=$conex->DBROWS.
$var=0.
while(%cat = $conex->dbnextrow(1))
{
  $avisosn=$cat(avisos).
  $numero=$cat(menos).
  $navisos=$avisosn-$numero.
  $ncta=$cat(no_usuario).
  $conexa = new Sybase DBlib $log, $pas, $ser, 'ocasion',
  for($var=0 $var<=$status, $var++)
  {
    $conexa->dbcmd("update USUARIO
    set no_avisos=$navisos
    where no_usuario like \"'$ncta'").
    $conexa->dbsqlxec.
    $conexa->dbresults.
  } #fin del for
} #fin del while
$conexbf = new Sybase DBlib $log, $pas, $ser, 'ocasion';
if ( ($CATEGORIA ne "ALOJAMIENTO") || ($CATEGORIA ne "ENSENANZA") )
{
  $conexbf->dbcmd("select foto
  from $CATEGORIA
  where (datediff(day,fecha,getdate()))>0").
  $conexbf->dbsqlxec.
  $conexbf->dbresults.
  while(%fotos = $conexbf->dbnextrow(1))
  {
    $delf=$fotos(foto).
    $delfoto = "m /home/users00/ocasion/htdocs/FOTOS/$delf".
    system($delfoto).
  }
  $conexb = new Sybase DBlib $log, $pas, $ser, 'ocasion';
  $conexb->dbcmd("delete
  from $CATEGORIA
  where (datediff(day,fecha,getdate()))>0");
  $conexb->dbsqlxec.
  $conexb->dbresults.
}

```

datos12.pl

```

#!/usr/bin/perl
$|=1.

```

```

require "var.pl";
use Sybase DBlib.
use CGI.
$query=new CGI.
$dbh = new Sybase DBlib $log, $pas, $ser, 'avisos';
print "Content-type: text/html\n";
$proceso=$$.
#obtenemos la fecha actual
$fecha = 'date '+DATE '%m-%d-%Y' | cut -d" " -f2-4';
print << "formato_página-inicio".
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html, charset=iso-8859-1">
<SCRIPT LANGUAGE="JavaScript">
function validaForm()
{
  objeto=document.felatosarticulo.
  nieltas=objeto.descripcion.value.length.
  medir=objeto.proceso.value.length.
  cuatro=4.
  size=medir-cuatro.
  if (objeto.articulo.value == "")
  {
    alert("Debe llenar el campo del articulo").
    objeto.articulo.focus().
    return false.
  }
  if (objeto.descripcion.value == "")

```

```

    alert("Debe llenar el campo de descripcion");
    objeto.descripcion.focus();
    return false;
}
if (objeto.descripcion.value.length > 150)
{
    alert("Has excedido el numero de letras, tenes " + nletras);
    objeto.descripcion.focus();
    return false;
}
}
if(objeto.proceso.value != ""){
    if (objeto.proceso.value.substring(size.medir).toLowerCase()!=".gif" &&
objeto.proceso.value.substring(size.medir).toLowerCase()!=".jpg")
    {
        alert("EL ARCHIVO NO ES VALIDO, VERIFIQUE",
objeto.proceso.focus());
        return false;
    }
}
}

```

```

</SCRIPT>
</HEAD>
formato_pagina-micro
require "sup.txt".
#----- cachamos los datos que anexamos a cada liga -----
if ($ENV{'REQUEST_METHOD'} eq "POST")
{
    read(STDIN, $entrada, $ENV{'CONTENT_LENGTH'});
}
elsif ($ENV{'REQUEST_METHOD'} eq "GET")
{
    $entrada=$ENV{'QUERY_STRING'};
}
else
{
    $entrada= $ARGV[1];
}
#separa primero por & y luego mete las parejas a una lista
foreach (split(/&/, $entrada))
{
    ($campo, $valor) = split(/=/, $_)
    #lista asociativa
    $parametros{$campo} = $valor;
}
#----- para ver cada pareja -----
@claves = keys %parametros;
foreach (@claves)
{
}
#-----
#ahora solo agarramos los campos que nos interesan
$TABLA = $parametros{categoria};
$REC = $parametros{recuerda};
#para las categorias que no se pone opcion de foto
if($TABLA eq "ALOJAMIENTO" || $TABLA eq "ENSENANZA")
{
    $sejecutaprograma="insercion-aviso.pl";
}
else{
    $sejecutaprograma="uploadindice.pl";
}
$NO_CAT = $parametros{nc};
$NO_SUBCAT = $parametros{subcategoria};
$numero_real=$parametros{numero};
$tabla=$parametros{tabla};
$mandafoto=$parametros{mandafoto};
$tabla="USUARIO";
if ($tabla eq "USUARIO")
{
    $dbh->dbcmd("select * from USUARIO where no_usuario=\"$numero_real\"");
    $dbh->dbselectexec;
    $dbh->dbresults;
    $dbh->DBROWS;
    while(%cat = $dbh->dbnextrow(1))
    {
        $nombre=$cat{nombre};
        $apellidos=$cat{apellido};
        $tel=$cat{telefono};
        $correo=$cat{mail};
    }#while do{ select
    $nombreaux=$nombre " " $apellidos;
} #if tabla eq USUARIO

```









```

print " <input type='hidden' name='nombre' value='\$nombre'>
<input type='hidden' name='apellidos' value='\$apellidos'>
<input type='hidden' name='telefono' value='\$tel'>
<input type='hidden' name='ladal' value='\$ladal'>
<input type='hidden' name='correo' value='\$correo'>
<input type='hidden' name='tabla' value='\$TABLA'>
<input type='hidden' name='no_cat' value='\$NO_CAT'>
<input type='hidden' name='no_subcat' value='\$NO_SUBCAT'>
<input type='hidden' name='numero' value='\$numero_real'>
<div align='center'>
<input type='submit' value='Enviar'>
<input type='reset' value='Limpiar'>
</div>
</form>";

```

#### registro-nuevo-libre.pl

```

#!/usr/bin/perl
$|=1;

```

```

$proceso=$$.
print "Content-type text/html\n\n";

```

```

# Ponemos la validacion para todas las formas para que se llene el campo de descripcion y ponemos la cabecera
print "<=validacion";
require "sup.txt";

```

```

<SCRIPT LANGUAGE="JavaScript">
function validaForma()
{
    objeto=document fdatosarticulo;
    var or_nombre = /^[A-Z]{4,4}[0-9]{6,6}$/;
    if (objeto.nombre value == "")
    {
        alert("Debes llenar la forma y proporcionar tu nombre");
        objeto.nombre focus();
        return false;
    }
    if (objeto.apellidos value == "")
    {
        alert("Debes llenar el campo de apellidos");
        objeto.apellidos focus();
        return false;
    }
    if (objeto.telefono1 value == "")
    {
        alert("Debes proporcionar un telefono");
        objeto.telefono1 focus();
        return false;
    }
    if (objeto.rfc value == "")
    {
        alert("Debes llenar el campo del rfc");
        objeto.rfc focus();
        return false;
    }
    if ( ( (or_nombre test(objeto.rfc value)) ) ) {
        alert("Contenido de RFC no valido");
        objeto.rfc focus();
        return false;
    }
}

```

```

</SCRIPT>
<form name="fdatosarticulo" onSubmit="return validaForma()" method="POST" enctype="multipart/form-data" action="usuario-nuevo-libre.pl">

```

```

<table width=99% border="0">

```

```

<tr>
<td width=50% colspan="2"><center>
<font face="Arial, Verdana, Helvetica, sans-serif" size="4">
<b>DATOS PERSONALES</b><br>
(Los datos marcados con asterisco son necesarios para registrarte)
</font></center><br></td>

```

```

</tr>
validacion
print "<=uno";

```

```

<td valign="top">
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>Nombre(s) </b><br>
<input name="nombre" type="text" value="$nombre" maxlength="30">
</font><br>
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>Apellidos </b><br>
<input name="apellidos" type="text" value="$apellidos" maxlength="40">

```

```
</font><br>
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>Teléfono fono </b><br>
<input name="telefono1" type="text" size="20" value="321" maxlength="15">
</font>
<br>
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>R.F.C. (Sin Homoclave)</b><br>
<font face="Arial, Verdana, Helvetica, sans-serif" size="1">
(La inicial de tu primer apellido <br>
la primera vocal de tu primer apellido.<br>
la primera letra del
segundo apellido y <br> la primera de tu nombre <br> Junto tu fecha de nacimiento indicando año/mes/día)<br>
Ejemplo Ramirez Moncada Carlos 17/nov/1972<br>RAMC721117 (Sin guiones ni espacios y en mayúsculas)</font><br>
<input name="rfc" type="text" size="20" value="321" maxlength="10">
<br>
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>Dirección</b><br>
<input name="direccion" type="text" size="20" value="321" maxlength="150">
<br>
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>E-mail</b><br>
<input type="text" name="correo" size="20" value="3correo" maxlength="29">
<br>
</font>
</td>
<td rowspan="5" style="text-align:center">
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>Sexo</b><br>
Masculino
<input type="radio" name="sexo" value="M" checked>
Femenino
<input type="radio" name="sexo" value="F">
</font><br>
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>Ocupación</b><br>
<select name="ocupacion" size="1">
<option value="Otro">Elija una opción</option>
<option value="Estudiante UNAM">Estudiante UNAM</option>
<option value="Trabajador UNAM">Trabajador UNAM</option>
<option value="Academico UNAM">Academico UNAM</option>
<option value="Otro">Otro</option>
</select>
<br>
Si eligió &quot;Otro&quot; especifique por favor
<br><input type="text" name="otrooc" size="30">
<br>
<font face="Arial, Verdana, Helvetica, sans-serif" size="3">
<b>Nivel de estudios</b><br>
<select name="estudios" size="1">
<option value="Otro">Elija una opción</option>
<option value="Bachillerato">Bachillerato</option>
<option value="Licenciatura">Licenciatura</option>
<option value="Posgrado">Posgrado</option>
<option value="Otro">Otro</option>
</select>
<br>
Si eligió &quot;Otro&quot; especifique por favor
<br><input type="text" name="otroon" size="30">
</center></td>
</tr>
<tr>
<td colspan="2">
uno
# -----
print " <br><br>
<center>
<input type="hidden" name="proceso1" value="3proceso1">
<input type="submit" value="Enviar">
<input type="reset" value="Limpiar">
</center>
</form>".
require "inf.txt"

# /usr/bin/perl

require "001",
"iso strict,

MAIN
{
my (%cgi_data, # El form data
%cgi_cfn, # Los nombres de los archivos cargados desde el cliente
```

uploadindice.pl

```

%cgi_ct, # El content type de los archivos cargados desde el cliente
%cgi_sfn, # Los nombres de los archivos cargados en el servidor (esta maquina)
$ret, # Valor Return de la llamada ReadParse
$buf, # Buffer para los datos leidos desde disco
$proceso, #cambiar por upfile
@archivo,
$numero,
@numero,
$nombre,
$archivotxt,
$archivobdf,
$ruta,
$var,
$var2,
$extension,
$cambia,
$tag,
$inicio,
$frase,
$lin,
$bandera,
$arc,
$lineadatos, #trae los datos octavio=rodriguez=9744=...
@datos, #aqui guardamos los datos separados
@campos, #solo tiene los nombres de los campos que da el usuario
$contador,
$repetido, #va guardar 0 si no esta ya la linea en usuario txt
$repetido, #va guardar 0 si no hay un proceso igual en usuario txt
$lineapro, #va guardar 0 si no esta la linea con numero de proceso diferente
$archivotxt, #tiene el nombre del archivo
$lineafoto, #tiene los datos e incluye el nombre de foto
@lineafoto, #para meter el nombre de la foto
$archivo2, #variables que uso en sub mayusculas
$archivo,
$linea,
$linea2,
@datos,
$correo,
$foto,
@datos2,
$fotoaux, #guarda el nombre de la foto en minusculas
$nombrecos,
$linea3,
$sonia,
$son1,
$varbasura,
$articulo,
$descripcion,
$precio,
$aleatorio,
$idaviso
).

```

```

$cgi_lib writefiles = "/home/users00/ocasion/cgi-bin",
$cgi_lib maxdata = 600000, #determina el tamaño maximo que puede tener
if($cgi_cfn[proceso]!=""){ $cgi_cfn[proceso]="fotovacia", }
$ret = &ReadParse(%cgi_data,%cgi_cfn,%cgi_ct,%cgi_sfn),
if (defined $ret) {
    &CgiDie("Error in reading and parsing of CGI input"),
}
elseif (!$ret) {
    &CgiDie("Faltan parametros"),
        "Por favor complete la forma <a href='http://www.deocasion.unam.mx/csr.htmf'>Lenar forma</a>\n",
}
elseif (defined $cgi_data[proceso]) {
    &CgiDie("Faltan Datos"),
        "Por favor complete la forma <a href='http://www.deocasion.unam.mx/csr.htmf'>Lenar forma</a>\n",
}
}
(@archivo)=split($cgi_cfn[proceso]),#upfile
$numero=$#archivo,
$nombrecos=$numeroof(@archivo),
$aleatorio= int(rand 100),
$nombrecos=$nombrecos $aleatorio,
print &Print header,
require "sup.tst",
print "$lineafoto",
@datosu=(split=/$cgi_data[lineadatos]), #lineadatos trae octavio=rodriguez=...
$articulo=$cgi_data[articulo],
$articulo = -s/~/g,
$articulo = -s/341/a/g,
$articulo = -s/341/a/g,
$articulo = -s/351/e/g,

```

```

$articulo = -sA355/u/g.
$articulo = -sA363/o/g
$articulo = -sA372/u/g
$articulo = -sA301/A/g
$articulo = -sA311/E/g.
$articulo = -sA315/l/g.
$articulo = -sA323/O/g.
$articulo = -sA332/U/g.
$articulo = -sA361A321/g.
$datosu[8]=$articulo.
$articulo=$cgi_data["descripcion"].
$articulo = -sA/l/g.
$articulo = -sA341/a/g
$articulo = -sA341/a/g
$articulo = -sA351/e/g
$articulo = -sA355/u/g.
$articulo = -sA363/o/g.
$articulo = -sA372/u/g.
$articulo = -sA301/A/g
$articulo = -sA311/E/g
$articulo = -sA315/l/g
$articulo = -sA323/O/g
$articulo = -sA332/U/g
$articulo = -sA361A321/g
$datosu[9]= $articulo
$articulo=$cgi_data["precio"].
$datosu[10]=$articulo.
$diviso=$datosu[6]"$datosu[7]"$datosu[13]".
print "<font color='W000099f'><b>ESTOS SON LOS DATOS REGISTRADOS DE TU ANUNCIO</b></font><p>".
#-----agregado para mostrar datos del usuario 31-may-00-----
$contador=0
@campos=("NOMBRE","APELLIDOS","TELEFONO","LADA","E-
MAIL","CATEGORIA","NO_CAT","SUBCATEGORIA","DESCRIPCION","PRECIO","FOTO.$archivo[numero]","NUMERO",
"NO DEPROCESO","FECHA").
{
    $lineayfoto{$contador}=$. #metemos los datos
    if(($_ ne "") || ($_ ne ""))
        if($campos{$contador} eq "NO_CAT" || $campos{$contador} eq "SUBCATEGORIA" || $campos{$contador} eq "NUMERO" ||
$campos{$contador} eq "NO DEPROCESO" || $campos{$contador} eq "LADA")
        {
            print".
        }
        else{
            $varbasura=(uc($_.))
            if($campos{$contador} eq "FOTO fotovacia")
            {
                print "FOTO Sin foto <br>".
            }
            else{
                print "$campos{$contador} $varbasura <br>".
            }
        }
    }
    else{
        print "$campos{$contador} x<br>".
    }
}
$contador++
} #cerra for each
print "<br><font color='red'>Recuerda que tu anuncio ser&aacute; publicado un d&iacute;a, a h&aacute;s de spu&eacute;s de su
registro </font>".
$nombreres= lc.$archivo[numero].
$archivo[numero]=$nombreres.
$datosu[11] = "$cgi_data[numero] $archivo[numero] $naleatorio";
$datosu[11]=(uc($datosu[11])).
$lineafoto=join(" ",@datosu). #y al arreglo lo volvemos una cadena
#} else
#-----
print <<EOT.
</>
EOT
#-----antes de guardar checamos que no este repetido el registro-----
$archivotxt="usuario.txt".
$repetido=0.
$prepetido=0.
$linearep=0.
#-----
$repetido = `grep "$lineafoto" usuario.txt | wc -l`;
$prepetido = `cut -d= -f14 usuario.txt | grep "$datosu[13]" | wc -l`;
$linearep=`cut -d= -f1-13,15- usuario.txt | grep "$datosu[13]" | wc -l`;
if ($repetido==0 && $linearep==0)
{
    open (USUARIO, ">>$archivotxt").

```

```

print USUARIO
"$lineafoto"; #se quito el salto de linea \n lo pone solo
close(USUARIO);
}
else
{
print "<h3>Anuncio ya registrado</h3><p>";
}
$archivo2 = "aux.txt";
$archivo = "usuario.txt";
open (USUARIO, "$archivo");
while ($linea=<USUARIO>) {
$linea2 = uc $linea;
@datos = split(/=/, $linea);
$correo = lc $datos[4];
@datos2 = split(/=/, $linea2);
$datos2[4] = $correo;
$linea3 = join("=", @datos2);
open (AUX1, ">$archivo2");
print AUX1 "$linea3";
close(AUX1);
}
close(USUARIO);
system("cp aux.txt usuario.txt");
system("rm aux.txt");
#-----
open (UPFILE, "$cgi_sfn(proceso) or #upfile
&CgiError("ERROR NO SE PUEDE ABRIR EL ARCHIVO
$cgi_sfn(proceso) $'\n'"); #upfile
$buf = "";
while (read (UPFILE $buf, 8192)) {
$buf = $buf.</> $buf;
}
close (UPFILE);
#-----8 mayo cambio oculto por numero 3 -----
COMO </b><br><dd>$cgi_data[numero] $archivo$numero)\n</center>";
system("mv /home/users00/ocasion/cgi-bin/cgi-lib $$ 1 /home/users00/ocasion/htdocs/FOTOS/$cgi_data[numero] $sonia.$snaletorio");
system("chmod 755 /home/users00/ocasion/cgi-bin/cgi_data[numero] $sonia");
system("rm /home/users00/ocasion/htdocs/FOTOS/FOTOVACIA");
require "mf.txt";
print &HtmlBot;
} #cierra main

usuario-borra.pl

#usr/bin/perl
$|=1; #para agilizar la respuesta

require "var.pl"
use Sybase DBlib

use CGI;
$query = new CGI;
$numuncio = $query->param('nanuncio');
$numusuario = $query->param('nusuario');
$tabla = $query->param('tabla');
$foto = $query->param('foto');
$tipotabla = $query->param('tipotabla');
$fecha = date '+DATE %m-%d-%Y' | cut -d' ' -f2-4';
$dbh = new Sybase DBlib $log, $pas, $$er, 'ocasion';
print "Content-type: text/html\n\n";
require "sup.txt";
if($tipotabla eq "Articulos Escolares"){
$CATEGORIA="AESCOLAR";
$NUMERO=1;
&actualizacion ($CATEGORIA, $NUMERO);
}
if($tipotabla eq "Alojamiento"){
$CATEGORIA="ALOJAMIENTO";
$NUMERO=2;
&actualizacion ($CATEGORIA, $NUMERO);
}
if($tipotabla eq "Diversos"){
$CATEGORIA="DIVERSOS";
$NUMERO=3;
&actualizacion ($CATEGORIA, $NUMERO);
}
if($tipotabla eq "Enseñanza"){
$CATEGORIA="ENSEÑANZA";
$NUMERO=4;
&actualizacion ($CATEGORIA, $NUMERO);
}

```

```

if($!tipotabla eq "Equipo de Computo"){
$CATEGORIA="EQUIPO".
$NUMERO=$5.
&actualizacion ($CATEGORIA, $NUMERO).
}
if($!tipotabla eq "Mascotas"){
$CATEGORIA="MASCOTA".
$NUMERO=$6.
&actualizacion ($CATEGORIA $NUMERO).
}
if($!tipotabla eq "Musica"){
$CATEGORIA="MUSICA".
$NUMERO=$7.
&actualizacion ($CATEGORIA $NUMERO).
}
if($!tipotabla eq "Vehiculos"){
$CATEGORIA="VEHICULO".
$NUMERO=$8.
&actualizacion ($CATEGORIA $NUMERO).
}
print "<br>
<font face='Arial, Verdana, Helvetica, sans-serif' size='14">
<center><br><b>TU ANUNCIO HA SIDO BORRADO $!status/<br></center>";
print "<br><br><font face='Arial, Verdana, Helvetica, sans-serif' size='12
'><center><b><font face='Arial, Verdana, Helvetica, sans-serif' size='14">
Volver a mis anuncios</a></center></font>".
print "<br><br><br><font face='Arial, Verdana, Helvetica, sans-serif' size='12"><center><b
href='http //www.deocasion unam mx/index.html">Volver a la p&acuteregina principal </a></center></font>";
require "inf.txt".

sub actualizacion
{
local ($var).
$CATEGORIA=$ _[0].
$NUMERO=$ _[1].
$conex="&dbn"+"$NUMERO".
$conex = new Sybase DBlib $log, $pas, $$ser, 'ocasion';
$conex->dbcmd("select * from USUARIO
where no_usuario='&snusuario'");
$conex->dbsqlxexec.
$conex->dbresults.
$conex->DBROWS.
$status=$conex->DBROWS.
$var=0.
while(%cat = $conex->dbnextrow(1))
{
$avisosn=$cat[no_avisos]
$numero=1.
$navisos=($avisosn-1).
$ncat=$cat[no_usuario]
} #fin del while
$conexbf = new Sybase DBlib $log, $pas, $$ser, 'ocasion'.
if ( ( $CATEGORIA ne "ALOJAMIENTO" ) || ( $CATEGORIA ne "ENSENANZA" ) )
{
$conexbf->dbcmd("select foto
from $CATEGORIA
where num=&nanuncio and no_usuario like '&snusuario' ");
$conexbf->dbsqlxexec.
$conexbf->dbresults.
while(%fotos = $conexbf->dbnextrow(1))
{
$def=$fotos(foto)
$defoto="rm /tmpdocs/FOTOS/$def".
system($defoto).
$conexa = new Sybase DBlib $log, $pas, $$ser, 'ocasion';
for($var=0, $var<=, $var++)
{
$conexa->dbcmd("update USUARIO
set no_avisos=$navisos
where no_usuario like '&ncat'");
$conexa->dbsqlxexec
$conexa->dbresults
} #fin del for
} #while fotos
}
$conexb = new Sybase DBlib $log, $pas, $$ser, 'ocasion';
$conexb->dbcmd("delete
from $CATEGORIA
where num=&nanuncio and no_usuario like '&snusuario' ");
$conexb->dbsqlxexec.
$conexb->dbresults.
$conexb->DBROWS.

```

```

$statalso=$conex->DBROWS,
$conex1 = new Sybase DBlib $log, $pas, $ser, 'ocasion',
for($var=0, $var<1, $var++)
{
    $conex1->dbcmd('update USUARIO
    set no_avisos=$navisos
    where no_usuario like \''$ctal*')
    $conex1->dbclose,ec,
    $conex1->dbresults,
} #fin del for
}

```

usuario-nuevo-libre. pl

```

#!/usr/bin/perl
$|=1, #para agilizar la respuesta

```

```

require "var.pl",
use Sybase DBlib,

```

```

use CGI,
$query=new CGI,

```

```

$tipo=$query->param('tipo'),
$nombre=$query->param('nombre'),
$ap=$query->param('apellidos'),
$tel=$query->param('telefono1'),
$correo=$query->param('correo'),
$rfc=$query->param('rfc'),
$direccion=$query->param('direccion'),
$sexo=$query->param('sexo'),
$ocupacion=$query->param('ocupacion'),
$otraoc=$query->param('otraoc'),
$estudios=$query->param('estudios'),
$otromiv=$query->param('otromiv'),
$proceso=$query->param('proceso'),
$nombre = -s/A341/w/g,
$nombre = -s/A341/z/g,
$nombre = -s/A351/w/g,
$nombre = -s/A355/w/g,
$nombre = -s/A363/o/g,
$nombre = -s/A372/w/g,
$nombre = -s/A301/w/g,
$nombre = -s/A311/E/g,
$nombre = -s/A315/l/g,
$nombre = -s/A323/O/g,
$nombre = -s/A332/U/g,
$nombre = -s/A361/N/g,
$ap = -s/A341/a/g,
$ap = -s/A351/o/g,
$ap = -s/A355/w/g,
$ap = -s/A363/o/g,
$ap = -s/A372/w/g,
$ap = -s/A301/w/g,
$ap = -s/A311/E/g,
$ap = -s/A315/l/g,
$ap = -s/A323/O/g,
$ap = -s/A332/U/g,
$ap = -s/A361/N/g,
$ap=(uc($ap)),
$nombre=(uc($nombre)),
$tel=(uc($tel)),
$rfc=(uc($rfc)),
$direccion=(uc($direccion)),
$otraoc=(uc($otraoc)),
$otromiv=(uc($otromiv)),
$correo=(uc($correo)),
$largoRFC=(length($rfc)),
$checha=(substr($rfc,0,4)),
$nacimento=(substr($rfc,4,6)),
if( $checha == /A-Z/ )
{
    $octavio="si",
}
else
{
    $octavio="no",
}

```

```

print "Content-type text/html\n\n";

```

```

$tabla="USUARIO",
$status2=0,

```

```

$statuso=$conexb->DBROWS,
$conexb = new Sybase DBlib $log, $pas, $ser, $ocasion',
for($var=0, $var<1, $var++)
{
    $conexb->dbcmd("update USUARIO
                    set no_aviso=$naviso
                    where no_usuario like '$$ctal'");
    $conexb->dbsqlexec,
    $conexb->dbresults,
} #fin del for
}

```

### usuario-nuevo-libre. pl

```

#!/usr/bin/perl
$|=1, #para agilizar la respuesta

require "var.pl",
use Sybase DBlib,

uso CGI,
$query=new CGI,

$tipo=$query->param('tipo'),
$nombre=$query->param('nombre'),
$ap=$query->param('apellidos'),
$tel=$query->param('telefono1'),
$correo=$query->param('correo'),
$rfc=$query->param('rfc'),
$direccion=$query->param('direccion'),
$sexo=$query->param('sexo'),
$ocupacion=$query->param('ocupacion'),
$otraoc=$query->param('otraoc'),
$estudios=$query->param('estudios'),
$otroniv=$query->param('otroniv'),
$proceso=$query->param('proceso'),
$nombre = ~s/A341/a/g,
$nombre = ~s/A341/w/g,
$nombre = ~s/A351/e/g,
$nombre = ~s/A355/u/g,
$nombre = ~s/A363/o/g,
$nombre = ~s/A372/u/g,
$nombre = ~s/A301/A/g,
$nombre = ~s/A311/E/g,
$nombre = ~s/A315/l/g,
$nombre = ~s/A323/O/g,
$nombre = ~s/A332/U/g,
$nombre = ~s/A361/N/g,
$ap = ~s/A341/a/g,
$ap = ~s/A351/e/g,
$ap = ~s/A355/u/g,
$ap = ~s/A363/o/g,
$ap = ~s/A372/u/g,
$ap = ~s/A301/A/g,
$ap = ~s/A311/E/g,
$ap = ~s/A315/l/g,
$ap = ~s/A323/O/g,
$ap = ~s/A332/U/g,
$ap = ~s/A361/N/g,
$ap=(uc($ap)),
$nombre=(uc($nombre)),
$tel=(uc($tel)),
$rfc=(uc($rfc)),
$direccion=(uc($direccion)),
$otraoc=(uc($otraoc)),
$otroniv=(uc($otroniv)),
$correo=(lc($correo)),
$largorfc=(length($rfc)),
$checa=(substr($rfc,0,4)),
$nacimento=(substr($rfc,4,6)),
if( $checa =~ /(A-Z)/ )
{
    $octavio="si",
}
else
{
    $octavio="no",
}

print "Content-type text/html\n\n",

$tabla="USUARIO",
$$status2=0,

```



```

$largo=(length($cuenta))
$anio=(substr($cuenta,0,2))
$anio2=(substr($cuenta,1,2))
$anio3=(substr($cuenta,0,3))
$fecha=(localtime($time))
$fecha2=(substr($fecha,11,8))
$fecha=$fecha2
$fecha=- $/ //g
$auxiliar=($proceso*123)
$longpro=(length($auxiliar))
$ultimo=$longpro-3
$verificador=(substr($auxiliar,$ultimo,3))
$claveunica=$rfc $verificador.

```

```

$dbh = new Sybase DBlib $log, $pas, $ser, 'ocasion';
$dbh->dbcmd("select no_usuario from USUARIO
where no_usuario='\$claveunica"
");

```

```

$dbh->dbsqlxexec.
$dbh->dbresults
$dbh->DBROWS
$status=$dbh->DBROWS.

```

```

$dbh = new Sybase DBlib $log, $pas, $ser, 'ocasion';
$dbh->dbcmd("select * from USUARIO
where nombre='\$nombref'
and apellido='\$apf' ");

```

```

$dbh->dbsqlxexec.
$dbh->dbresults
$dbh->DBROWS
$status=$dbh->DBROWS

```

```
require "sup.txt".
```

```
if($status==0 && $status==0) #si no existe
{ #if status 0
```

```

$dbh3 = new Sybase DBlib $log, $pas, $ser, 'ocasion';
$dbh3->dbcmd("insert into
USUARIO(no_usuario,nombre,apellido,telefono,mail,rfc,
direccion,sexo,ocupacion,otra,nivel,otro,no_avisos)
values('\$claveunica','\$nombref','\$apf','\$tel',
'\$correo','\$rfc','\$direccion','\$sexo','\$ocupacion',
'\$otraoc','\$estudios','\$otroniv','0');
$dbh3->dbsqlxexec.
$dbh3->dbresults.

```

```

print "Esta es tu clave de usuario, anotala y con ella podra acceder al
sistema para publicar tus anuncios</h3>
<center><b><font size=5 color=$red>$claveunica</font></center></b></center>";
print "<br><center><b>Estos son los datos con que han sido registrado;</b><p></center>";
print "<b>Nombre <b> $nombre $ap<br>";
print "<b>RFC <b> $rfc<br>";

```

```

if($direccion ne ""){
print "<b>Direccion <b> $direccion<br>";
}
print "<b>Telefono <b> $tel<br>";
if($correo ne ""){
print "<b>Correo Electrónico <b> $correo<br>";
}
print "<b>Sexo <b> $sexo<br>";
if($ocupacion eq "Otro"){
if($otraoc ne ""){
print "<b>Ocupación <b> $otraoc<br>";
}
}

```

```

}else{
print "<b>Ocupación <b> $ocupacion<br>";
}
if($estudios eq "Otro"){
if($otroniv ne ""){
print "<b>Estudios </b> $otroniv<br><br>";
}
}
}else{
print "<b>Estudios </b> $estudios<br>";
}
}
} #if status 0
else{ #if status 0
print "<center><h3><b>
Ya est&aacute;s registrado
</b></h3><br>";
} #if status 0
require "inf.txt".

```

usuario-renueva.pl

```
#!/usr/bin/perl
$|=1. #para agilizar la respuesta

require "var.pl";
use Sybase DBlib.

use CGI;
$query=new CGI.

$nanuncio=$query->param('nanuncio').
$usuario=$query->param('usuario').
$tabla=$query->param('lanuncio').
$foto=$query->param('foto').

if ($tabla eq "Articulos Escolares") { $tablaux="AESCOLAR"; }
if ($tabla eq "Diversos") { $tablaux="DIVERSOS"; }
if ($tabla eq "Equipo de Computo") { $tablaux="EQUIPO"; }
if ($tabla eq "Mascotas") { $tablaux="MASCOTA"; }
if ($tabla eq "Vehiculos") { $tablaux="VEHICULO"; }
if ($tabla eq "Musica") { $tablaux="MUSICA"; }
if ($tabla eq "Alojamiento") { $tablaux="ALOJAMIENTO"; }
if ($tabla eq "Enseñanza") { $tablaux="ENSEÑANZA"; }

$fecha= date '+DATE %m-%d-%Y' | cut -d '-' -f 1-2-4;

print "Content-type text/html\n\n";
require "sup.txt".

$dbhr = new Sybase DBlib $log, $pas, $ser, 'ocasion';
$dbhr->dbcmd("select renovacion from $tablaux
where num=$nanuncio
and no_usuario='\$usuario' ");
while(%catu = $dbhr->dbnextrow(1))
{ # while catu
$renova=$catu{renovacion}
$renova=$renova+1.
} # while catu
$dbhr->dbclosec.
$dbhr->dbresults.

$dbh = new Sybase DBlib $log, $pas, $ser, 'ocasion';
$dbh->dbcmd("update $tablaux
set fecha=(dateadd(day,7,$fecha))
where num=$nanuncio
and no_usuario='\$usuario' ");
$dbh->dbclosec.
$dbh->dbresults.
$dbh->DBROWS.
$status=$dbh->DBROWS.

$dbh3 = new Sybase DBlib $log, $pas, $ser, 'ocasion';
$dbh3->dbcmd("update $tablaux
set renovacion=renovacion+1
where num=$nanuncio
and no_usuario='\$usuario' ");
$dbh3->dbclosec.
$dbh3->dbresults.
print "<br>
<font face='Arial, Verdana, Helvetica, sans-serif' size='4'>
<center><br><br><b>TU ANUNCIO HA SIDO RENOVADO POR 7 DIAS MAS A PARTIR DE HOY<br><br>
</center>";

print "<br><br><font face='Arial, Verdana, Helvetica, sans-serif' size='2'
'><center><a href='http://www.deocasion.unam.mx/cgi-bin/individual.pl?numero=$usuario'>Volver
a mis anuncios</a></center></font>".
print "<br><br><br><font face='Arial, Verdana, Helvetica, sans-serif' size='2'><center><a
href='http://www.deocasion.unam.mx/index.html'>Volver a la p&aacute;gina principal </a></center></font>";
require "inf.txt".

#/usr/bin/perl
$|=1. #para agilizar la respuesta

require "var.pl";
use Sybase DBlib.

use CGI;
$query=new CGI.

$tabla = "USUARIO".
```

valu3-j-libre.pl

```

$numero= $query->param('numero');
$numero=(uc($numero));
$clave = $query->param('clave');
$metiop = $query->param('metiop');
$dbh = new Sybase DBlib $log, $pas, $ser, 'aviso';
$argo=(length($numero));
$anio=(substr($numero,0,2));
$anio3=(substr($numero,0,3));

print "Content-type: text/html\n\n";

require "sup.txt";

if($valido eq "mal"){
    &base_vacia;
} else{
    if ($tabla eq "USUARIO")
    {
        $dbh->dbcmd("select * from $tabla where no_usuario=\"$numero\"");
        $dbh->dbsqlxec;
        $dbh->dbresults;
        $dbh->DBROWS;
        $status=$dbh->DBROWS;

        if ($status==0)
        {
            &base_vacia;
        }
        else
        {
            while(%cat = $dbh->dbnextrow(1))
            {
                $numav=$cat{no_avisos};

            }#while del select
            #para que no importe cuantos anuncios tiene
            $pasa="si";
            if ($pasa eq "no")
            {
                if($metiop==0){
                    &pasar_limite;
                }
                else{
                    print "<font color='white'>";
                    $buscando="grep -v \"$clave\" claves.txt";
                    $como=(system($buscando));
                    print "</font>";

                    if($corrio==0){
                        #borramos el password del archivo
                        $borrapw="grep -v \"$clave\" claves.txt > clavea.txt";
                        system($borrapw);
                        $cambio="mv clavea.txt claves.txt";
                        system($cambio);

                        #le restamos 1 a no_avisos
                        $nuevonum = ($numav-1);
                        $dbh = new Sybase DBlib $log, $pas, $ser, 'aviso';
                        $dbh->dbcmd("update USUARIO set no_avisos=$nuevonum where no_usuario=\"$numero\"");
                        $dbh->dbsqlxec;
                        $dbh->dbresults;
                        $dbh->DBROWS;
                        &pon_categorias;
                    }
                    else{
                        print "<center><b>Clave equivocada</b></center></font>";
                        &pasar_limite;
                    }
                    #f corrio
                }#f metiop
            } #if numav>3
            else
            {
                print "<p align='center'>";
                &pon_categorias; #funcion que escribe las categorias
            }
        }#else de si esta en la base
        #if si la tabla es estudiante
        else{ #de lo contrario selecciona de la tabla trabajador
            print "hola";
        } #else de lo contrario selecciona trabajador
    } #else valido
}

```

```

#####
##### subrutinas #####
#####

```

#####

sub pon\_categorias

{  
\$mandafoto = "no".

print << "et2"

<br>

<font face="Arial, Verdana, Helvetica, sans-serif" size="4">  
<center><b>SELECCIONA LA CATEGORIA PARA TU ANUNCIO</b></center></font>

<p align="center">&nbsp;.</p>

<div align="center"><center>

<table COLS="2" WIDTH="100%" border="0">

<tr>

<td ALIGN="LEFT" VALIGN="TOP"><div>

<!-- \*\*\*\*\*ligas de vehiculos\*\*\*\*\* -->

<b>

<font color="#000099" face="Arial, Verdana, Helvetica, sans-serif" size="3">VEH&iacute;lacule,CULOS</font>

<font size="3" color="#cc6600" face="Arial, Verdana, Helvetica, sans-serif" size="3">

<!-- ID="AUTO">

<a href="" /cgi-bin/registro-de-aviso-

libre.pl?categoria=VEHICULO&nc=11&subcategoria=23&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEF

ONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Automovil">Autom&ocute,viles</a></li>

<!-- ID="AUTO">

<a href="" /cgi-bin/registro-de-aviso-

libre.pl?categoria=VEHICULO&nc=11&subcategoria=24&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEF

ONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Motocicleta">Motocicetas</a></li>

<!-- ID="AUTO">

<a href="" /cgi-bin/registro-de-aviso-

libre.pl?categoria=VEHICULO&nc=11&subcategoria=25&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEF

ONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Bicicletas">Bicicletas</a></li>

<!-- ID="AUTO">

<a href="" /cgi-bin/registro-de-aviso-libre.pl?categoria=VEHICULO&nc=11&subcate

goria=3&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$ TELEFONO&lad

da=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Otros">Otros</font></a></li></b>

<br>

</div>

</td>

<!-- \*\*\*\*\*ligas de articulos escolares\*\*\*\*\* -->

<td ALIGN="LEFT" VALIGN="TOP"><div>

<font color="#000099" face="Arial, Verdana, Helvetica, sans-serif" size="3">

<b>ART&iacute;lacule CULOS ESCOLARES</font></b>

<b>

<font color="#cc6600" face="Arial, Verdana, Helvetica, sans-serif" size="3">

<!-- ID="CALCULADORA">

<a href="" /cgi-bin/registro-de-aviso-

libre.pl?categoria=AESCOLAR&nc=01&subcategoria=01&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEF

ONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Calculadora">Calculadoras</a></li>

<!-- ID="CALCULADORA">

<a href="" /cgi-bin/registro-de-aviso-

libre.pl?categoria=AESCOLAR&nc=01&subcategoria=02&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEF

ONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Papeleria">Papeler&iacute;a,</a></li>

<!-- ID="PAPELERIA">

<a href="" /cgi-bin/registro-de-aviso-

libre.pl?categoria=AESCOLAR&nc=01&subcategoria=03&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEF

ONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Libro">Libros</a></li>

<!-- ID="PAPELERIA">

<a href="" /cgi-bin/registro-de-aviso-libre.pl?categoria=AESCOLAR&nc=01&subcat

egoria=30&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$ TELEFONO&la

da=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Otros">Otros</b></a></font></li>

<br>

<br>

</div>

</td>

<!-- \*\*\*\*\*ligas de musica\*\*\*\*\* -->

<tr>

<td ALIGN="LEFT"><div>

<b>

<font face="Arial, Verdana, Helvetica, sans-serif" size="3" color="#000099">

M&Uacute;lacule SICA</font></b>

<b>

<font face="Arial, Verdana, Helvetica, sans-serif" size="3" color="#CC6600">

<!-- ID="DISCO">

<a href="" /cgi-bin/registro-de-aviso-

libre.pl?categoria=MUSICA&nc=12&subcategoria=2&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFON

ON&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Disco">Discos</a></li>

<!-- ID="INSTRUMENTO">

<a href="" /cgi-bin/registro-de-aviso-

libre.pl?categoria=MUSICA&nc=12&subcategoria=27&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFON

ON&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Instrumento">Instrumentos</a></li>

<!-- ID="REPRODUCTOR">

<!-- ID="REPRODUCTOR">

<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=MUSICA&nc=12&subcategoria=29&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Reproductor">Reproductores</a></li>  
<li ID="REPRODUCTOR">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=MUSICA&nc=12&subcategoria=36&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Otros">Otros</a></li></br>  
<br>  
</dir>  
</td>  
<td ALIGN="LEFT" VALIGN="TOP"><dir>  
<li>\*\*\*\*\*ligas de equipo de computo \*\*\*\*\*</li>  
<br>  
<b>  
<font face="Arial, Verdana, Helvetica, sans-serif" size="3" color="#000099">EQUIPO DE C&Oacute;MPUTO</font></b>  
<br>  
<font face="Arial, Verdana, Helvetica, sans-serif" size="3" color="#cc6600">  
<li ID="COMPUTADORA">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=EQUIPO&nc=07&subcategoria=13&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Computadora">Computadoras</a></li>  
<li ID="PERIFERICO">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=EQUIPO&nc=07&subcategoria=14&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Perifericos">Perifericos</a></li>  
<li ID="CONSUMIBLE">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=EQUIPO&nc=07&subcategoria=15&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Muebles">Muebles para equipo de c&Oacute;mputo</a></li>  
<li ID="CONSUMIBLE">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=EQUIPO&nc=07&subcategoria=33&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Otros">Otros</a></li></br>  
<br>  
<br>  
</dir>  
</td>  
</tr>  
<tr>  
<td ALIGN="LEFT" VALIGN="TOP"><dir>  
<br>  
<b><font color="#000099" size="3" face="Arial, Verdana, Helvetica, sans-serif">ENSEÑANZA</font></b>  
<br>  
<font size="3" color="#CC6600" face="Arial, Verdana, Helvetica, sans-serif">  
<li ID="CLASE">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=ENSEÑANZA&nc=06&subcategoria=11&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&nmandafoto=\$nmandafoto&recuerda=Clases particulares">Clases Particulares</a></li>  
<li ID="CLASE">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=ENSEÑANZA&nc=06&subcategoria=12&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&nmandafoto=\$nmandafoto&recuerda=Curso">Cursos</a></li>  
<li ID="CLASE">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=ENSEÑANZA&nc=06&subcategoria=32&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&nmandafoto=\$nmandafoto&recuerda=Otros">Otros</a></li></br>  
<br>  
<br>  
</dir>  
</td>  
<td ALIGN="LEFT"><dir>  
<li>\*\*\*\*\*ligas de mascotas\*\*\*\*\*</li>  
<br>  
<font face="Arial, Verdana, Helvetica, sans-serif" color="#000099" size="3">  
<b>MASCOTAS</font></b>  
<br>  
<font face="Arial, Verdana, Helvetica, sans-serif" size="3" color="#CC6600">  
<li ID="PERRO">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=MASCOTA&nc=08&subcategoria=16&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Perro">Perros</a></li>  
<li ID="GATO">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=MASCOTA&nc=08&subcategoria=17&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Gato">Gatos</a></li>  
<li ID="GATO">  
<a href="" /cgi-bin/registro-de-aviso-libre pi?categoria=MASCOTA&nc=08&subcategoria=18&nombre=\$NOMBRE&apellidos=\$APELIDOS&tabla=\$tabla&telefono=\$TELEFONO&lada=\$LADA&email=\$CORREO&numero=\$numero&recuerda=Otros">Otros</a></li>





```
#####
#Ruta de los directorios
$cg-bin="/cgi-bin",
$htdocs="http://e5000 dcaa unam mx 2015",
$admin="/admin",
#####
```

## htdocs

admin.html

```
<html>
```

```
<head>
```

```
<title>Untitled Document</title>
```

```
<meta http-equiv="Content-Type" content="text/html, charset=iso-8859-1">
```

```
</head>
```

```
<body bgcolor="#003399" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" onLoad="">
```

```
<table width="100%" border="0" cellspacing="0" cellpadding="0" bordercolordark="#E0E0E0" height="751">
```

```
<tr>
```

```
<td width="76"><a href="http://www.unam.mx">
```

```
name="escudo"></a></td>
```

```
<td align="center" valign="middle"></td>
```

```
</tr>
```

```
<tr>
```

```
<td width="76">&nbsp;</td>
```

```
<td>
```

```
<table width="100%" border="0" cellspacing="0" cellpadding="0">
```

```
<tr>
```

```
<td width="5" bgcolor="#FF9900">&nbsp;</td>
```

```
<td width="100%">
```

```
<table width="100%" border="0" cellspacing="0" cellpadding="0">
```

```
<tr>
```

```
<td colspan="3">
```

```
<table width="100%" border="0" cellspacing="0" cellpadding="0">
```

```
<tr>
```

```
<td bgcolor="#FF9900">&nbsp;</td>
```

```
<td bgcolor="#FF9900">&nbsp;</td>
```

```
<td bgcolor="#FFCC66">&nbsp;</td>
```

```
</tr>
```

```
</table>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td colspan="3">
```

```
<table width="100%" border="0" cellspacing="0" cellpadding="0">
```

```
<tr>
```

```
<td width="5" bgcolor="#FFFFFF"></td>
```

```
<td bgcolor="#FFFFFF" height="20" align="center"><font face="Arial, Verdana, Helvetica, sans-serif" size="2"><a
```

```
href="http://www.unam.mx">UNAM</a>
```

```
<font color="#FF9900"> | <font color="#000000"><a href="http://www.mapa.unam.mx/">Mapa de CU</a> <font> | <font
```

```
color="#000000"><a href="http://www.unam.mx/indices/unam.htm">WWW en la UNAM</a> <font> | <font color="#000000"><a
```

```
href="http://www.dgsc.unam.mx">DGSCA</a> <font> </td>
```

```
<td width="5" bgcolor="#FFFFFF" align="right"></td>
```

```
</tr>
```

```
</table>
```

```
</td>
```

```
</tr>
```

```
<tr align="center" valign="top">
```

```
<td colspan="3" bgcolor="#FFFFFF" height="722">
```

```
<p>&nbsp;</p>
```

```
<table width="90%" border="0" cellspacing="0" cellpadding="0">
```

```
<tr>
```

```
<td>
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
</br><br>
```

```
<br><br>
```

```
<br><br>
```

```
<p align="center"><font face="Arial, Verdana, Helvetica, sans-serif" size="4">
```

```
<center><b>
```

```
ADMINISTRACI&Oacute;N
```

```
<p align="center"><font face="Arial, Verdana, Helvetica, sans-serif" size="3">
```

```
<a href=" /index.htm">CONSULTAS</a>
```

```
<br>
```

```
<a href=" /registro-libre.htm">REGISTRO</a>
```

```
<br>
```

```
<br>
```

```
<a href=" / /cgi-bin/datos2.pl">ALTA DE ANUNCIOS</a>
```

```
<br>
```

```
<a href=" / /cgi-bin/pruebab-libre.pl">BAJA DE ANUNCIOS</a>
```

```
</b>
```









```

</tr>
</table>
</td>
</tr>
<tr align="center" valign="top">
<td colspan="2" bgcolor="FFFFFF" height="325">
<table border=0 cellpadding=0 cellspacing=0 width="703" height="325">
<tbody>
<tr align="center" valign="top">
<td width="28%">
<p align="center">
</p>
</td>
<td width="650" border="0" cellspacing="0" cellpadding="0" height="290">
<tr>
<td height="30">
<table width="640" border="0" cellspacing="0" cellpadding="0" height="20" align="center">
<tr>
<td width="220" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=1&titulogif=artesco_t.gif"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('articulos','','gifs/art_esc_on.gif','mientras','','jgs/artesco.jpg,1')">img src="gifs/art_esc.gif" width="220"
height="20" name="articulos" border="0"></a></td>
<td width="197" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=7&titulogif=equicompl_t.gif"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('equipos','','gifs/equip_comp_on.gif','mientras','','jgs/equip.jpg,1')">img src="gifs/equip_comp.gif"
width="197" height="20" name="equipos" border="0"></a></td>
<td width="130" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=11&titulogif=vehiculos_t.gif"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('enseñanla,anza','','gifs/vehu_on.gif','mientras','','jgs/vehiculos.jpg,1')">img src="gifs/vehu.gif"
width="130" height="20" name="enseñanla,anza" border="0"></a></td>
<td width="121" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=1&titulogif=musica_t.gif"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('ajornamento','','gifs/musica_on.gif','mientras','','jgs/musica.jpg,1')">img src="gifs/musica.gif"
width="121" height="20" name="ajornamento" border="0"></a></td>
</tr>
</table>
</td>
</tr>
</tr>
<tr align="center" valign="top">
<td height="240">img src="jgs/mientras.jpg" width="400" height="240" name="mientras" usemap="#mientrasMap"
border="0"><map name="mientrasMap"><area shape="rect" coords="269,169,305,186"
href="http://www.dedocasion.unam.mx/reglamento.htm" alt="Reglamento" title="Reglamento"></map></td>
</tr>
<tr align="center" valign="top">
<td height="20">
<table width="650" border="0" cellspacing="0" cellpadding="0" height="33" align="center">
<tr>
<td width="161" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=6&titulogif=enseñanza_t.gif"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('mientras','','jgs/ensenanza.jpg','vehiculos','','gifs/ense_on.gif,1')">img src="gifs/ense.gif" width="161"
height="20" name="vehiculos" border="0"></a></td>
<td width="198" height="20"><a href=" /cgi-bin/listas2-
libre.pl?categoria=2&incial=1&contareg=1&final=5&titulogif=ajornamento_t.gif" onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('montas','','jgs/ajornamento.jpg','muebles','','gifs/aloja_on.gif,1')">img src="gifs/aloja.gif" width="198"
height="20" name="muebles" border="0"></a></td>
<td width="168" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=8&titulogif=mascoas_t.gif"
onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('mientras','','jgs/mascoas.jpg','electrodomesticos','','gifs/masco_on.gif,1')">img src="gifs/masco.gif"
width="168" height="20" name="electrodomesticos" border="0"></a></td>
<td width="141" height="20"><a href=" /cgi-bin/listas2-
libre.pl?categoria=4&incial=1&contareg=1&final=5&titulogif=diversos_t.gif" onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('mientras','','jgs/diversos.jpg','bienes','','gifs/diver_on.gif,1')">img src="gifs/diver.gif" width="141"
height="20" name="bienes" border="0"></a></td>
</tr>
</table>
</td>
</tr>
</tr>
</table>
<table width="703" border="0" cellspacing="0" cellpadding="0" align="center">
<tr>
<td width="18" height="26" align="left" valign="top">
<div align="left">img src="gifs/cor_l_t.gif" width="18" height="26"></div>
</td>
<td width="668" align="center" valign="middle" height="26">
<div align="center"><a href="registro-libre.htm" onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('imagen1','','jgs/bot_reg_on.jpg,1')">img src="jgs/bot_reg.jpg" width="113" height="26" align="middle"
border="0" name="imagen1"></a></div>
</td>
<td width="17" height="26" align="right" valign="top">
<div align="right">img src="gifs/cor_r_t.gif" width="17" height="26"></div>
</td>
</tr>
</table>

```

```

        </table>
        </td>
        </tr>
        </tbody>
        </table>
        </td>
        </tr>
        <td colspan="3">
        <table width="100%" border="0" cellspacing="0" cellpadding="0" bordercolordark="#003898">
        <tr>
        <td bgcolor="#FFF900" height="10">&nbsp;  </td>
        <td bgcolor="#FFF900" height="10">&nbsp;  </td>
        <td bgcolor="#FFCC66" height="10">&nbsp;  </td>
        </tr>
        </table>
        </td>
        </tr>
        </table>
        </td>
        <td width="5" bgcolor="#FFCC66"></td>
        </tr>
        </table>
        </td>
        </tr>
        </table>
        <p>&nbsp;  </p>
        <p>&nbsp;  </p>
        <p>&nbsp;  </p>
        <SCRIPT language="javascript">
        <!--
        /* Certifica.com Version 3 00 (01/05/2001)*/
        Certifica_resolucion=otro";
        Certifica_colores="otro";
        Certifica_java_activado="otro";
        Certifica_browser="otro";
        Certifica_version="otro";
        Certifica_desde=""+"&escape(document.referrer);
        Certifica_cpu_XXX="otro";
        Certifica_idioma_sistema="otro";
        if (Certifica_desde == "")
        {
        Certifica_desde="otro";
        }
        Certifica_resolucion=screen.width;
        Certifica_browser=navigator.appName;
        Certifica_version_navegador=navigator.appVersion.charAt(0) + navigator.appVersion.charAt(1);
        Certifica_version_navegador=Certifica_version_navegador + navigator.appVersion.charAt(2);
        Certifica_cpu_XXX=navigator.cpuClass;
        Certifica_idioma_sistema=navigator.userLanguage;
        if (Certifica_browser != "Netscape")
        {
        Certifica_colores=screen.colorDepth;
        }
        else {
        Certifica_colores=screen.pixelDepth;
        }
        }
        Certifica_java_activado=navigator.javaEnabled();
        Certifica_NS2Ch_a=0
        if (navigator.appName == "Netscape" && navigator.appVersion.charAt(0) == "2")
        {
        Certifica_NS2Ch_a=1;
        }
        }
        if (Certifica_NS2Ch_a == 0) {
        Certifica_url_source="http://this.0/c/cert/hit.dll?sitio_id=5408&area_id=25963&pagina_id=277663&vc=3 00";
        Certifica_url_source+="&size="+Certifica_resolucion+"&color="+Certifica_colores;
        Certifica_url_source+="&rafers="+Certifica_desde+"&java="+Certifica_java_activado;
        Certifica_url_source+="&idio="+Certifica_idioma_sistema+"&cpu="+Certifica_cpu_XXX+"&vers="+Certifica_version_navegador+"";
        document.writeln("<A HREF='"+http://www.certifica.com/link/index.asp?sitio_id=5408&vc=3 00'><IMG BORDER=0
        SRC='"+Certifica_url_source+"'></A>");
        }
        <!--
        <SCRIPT>
        <NOSCRIPT>
        <A HREF="http://www.certifica.com/link/index.asp?sitio_id=5408&vc=3 00"><IMG BORDER=0
        SRC="http://this.0/c/cert/hit.dll?sitio_id=5408&area_id=25963&pagina_id=277663&vc=3 00"></A>
        </NOSCRIPT>
        </body>
        </html>
    </html>
    </head>
    <title>Mis Anuncios</title>

```

individual.html

```

<meta http-equiv="Content-Type" content="text/html, charset=iso-8859-1">
</head>

<body bgcolor="003399" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" onLoad="">
<table width="100%" border="0" cellspacing="0" cellpadding="0" bordercolordark="#E0E0E0" height="751">
<tr>
<td width="76"><a href="http //www unam mx"><img src="" /gifs/escudo.gif" width="76" height="75" align="middle" border="0"
name="escudo"> </a></td>
<td align="center" valign="middle"><img src="" /jpgs/tituloaz .jpg" width="400" height="69"></td>
</tr>
<td width="76">&nbsp;</td>
<td>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="5" bgcolor="#FF9900">&nbsp;</td>
<td width="100%">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td colspan="3">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td colspan="3" bgcolor="#FFF900">&nbsp;</td>
<td colspan="3" bgcolor="#FFF900">&nbsp;</td>
<td colspan="3" bgcolor="#FFCC66">&nbsp;</td>
</tr>
</table>
</td>
</tr>
<tr>
<td colspan="3">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="5" bgcolor="#FFFC00"><img src="" /gifs/corner_l.gif" width="20" height="20"></td>
<td colspan="2" bgcolor="#FFFC00" height="20" align="center">
</tr>
</table>
</td>
</tr>
<tr>
<td colspan="3">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td colspan="3" align="center">
</tr>
</table>
</td>
</tr>
</table>
</td>
</tr>
</table>

<font face="Verdana, Arial, Helvetica, sans-serif" size="2"><a href="http //www unam mx">
UNAM</a>
<font color="#FF9900">| <font color="#000000"><a href="h
ttp //www mapa unam mx">Mapa de CU</a></font>
| <font color="#000000"><a href="http //www unam mx/ind
ces/wwwunam.htm">WWW en la
UNAM</a></font> | &nbsp;<a href="http //www dgsga unam mx"><font color="#000000"
>DGSCA</font></a>
| &nbsp;<font color="#000000"><a href="index.htm">Inicio
</a></font>
| &nbsp;<font color="#000000"><a href="individual.htm"><b>Mis
&nbsp;</b></font>
</a></font> | &nbsp;<font color="#000000">
<a href="creditos.htm">Cr&eacute;ditos</a></font> </font> </font>
</td>
<td align="5" bgcolor="#FFFFFF" align="right"><img src="" /gifs/corner_r.gif" width="20" height="20"></td>
</tr>
</table>
</td>
</tr>
</table>
</td>
</tr>
</table>
<tr align="center" valign="top">
<td colspan="3" bgcolor="#FFFFFF" height="722">
<p>&nbsp;</p>
<table width="90%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td>
<form name="datos" onSubmit="" method="POST" enctype="multipart/form-data" action="..cgi-bin/individual.pl">
<p align="center"><font face="Arial, Verdana, Helvetica, sans-serif" size="3"><b>
Ingresa tu clave de usuario para ver tus anuncios
</b></font></p>
<p > <br>
<font size="3" face="Arial, Verdana, Helvetica, sans-serif">
<b>CLAVE DE IDENTIFICACI&Oacute;N</b> <br>
(Sin guiones ni espacios ) <br>
<br>
<input type="password" name="numero" maxlength="15" size="15">
</td>
</tr>
</table>
<table align="left">
<tr>
<td align="center">
<input type="submit" name="enviar" value="Ingresar">
&nbsp;<br>
</td>
</tr>
</table>
</td>
</tr>
</table>
</p>
</td>
</tr>
</table>
</p>&nbsp;</p>

```



```

var digitos = objeto.numero.value.length;
if (objeto.numero.value == "")
{
    alert("Debes proporcionar tu número Gracias").
    objeto.numero.focus();
    return false;
}

//comprueba que numero solo tenga digitos del 0 al 9
if (er_estudiante.test(objeto.numero.value) )
{
    alert("El número no es válido, favor de verificar").
    return false;
}
} //cierra funcion validaForma
</script>
</head>

<body leftmargin="0" topmargin="0" marginwidth="0" marginheight="0"
onload="MM_preloadImages('gifs/equip_comp_on.gif','jggs/equip.jpg','gifs/vehul_on.gif','jggs/vehiculos.jpg','gifs/musica_on.gif','jggs/music
a.jpg','jggs/senseofianza.jpg','gifs/sense_on.gif','jggs/alasamiento.jpg','gifs/alopa_on.gif','jggs/mascotas.jpg','gifs/masca_on.gif','jggs/diverso
s.jpg','gifs/diver_on.gif','jggs/bot_reg_on.jpg'); link="#000000"; bgcolor="#003399">
<table width="100%" border="0" cellspacing="0" cellpadding="0" bordercolordark="#E0E0E0" align="left" height="400">
<tr>
<td width="67" height="58"><a href="http://www.unam.mx"></a></td>
<td align="center" valign="middle" height="58"></td>
</tr>
<tr>
<td width="67" align="center" height="360"></td>
<td valign="top" align="left" height="360">
<table width="713" border="0" cellspacing="0" cellpadding="0" align="left">
<tr>
<td width="5" bgcolor="#FF9900"></td>
<td width="100%" valign="top" align="center">
<table width="703" border="0" cellspacing="0" cellpadding="0">
<tr>
<td colspan="3">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td bgcolor="#FF9900">&nbsp;</td>
<td bgcolor="#FF9900">&nbsp;</td>
<td bgcolor="#FF9900">&nbsp;</td>
</tr>
</table>
</td>
</tr>
<tr>
<td colspan="3">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="18" bgcolor="#FFFFFF" height="18"></td>
<td bgcolor="#FFFFFF" height="18" align="center">&nbsp;</td>
<td align="center" height="18" align="center">&nbsp;</td>
</tr>
</table>
</td>
</tr>
<tr>
<td align="center" height="18" align="center">&nbsp;</td>
<td align="center" height="18" align="center">&nbsp;</td>
<td align="center" height="18" align="center">&nbsp;</td>
</tr>
</table>
</td>
</tr>
<tr>
<td colspan="3">
<table border="0" cellpadding="0" cellspacing="0" width="703" height="325">
<tbody>
<tr align="center" valign="top">
<td width="28%">
<p><font face="Tahoma, Arial"> </font>
<center>
<table width="650" border="0" cellspacing="0" cellpadding="0" height="290">
<tr align="center" valign="top">

```



```

        <td height="260">
            <form name="tdatos" onSubmit="" method="POST" enctype="multipart/form-data"
            action="" /cgi-bin/valusu3-libre.pl">
                <p align="center"><font face="Arial, Verdana, Helvetica, sans-serif" size="3"><b>
                    Ingrese tu clave de usuario para registrar tu anuncio,
                    recuerda que este ser&aacute; publicado un d&aacute;a h&aacute;bil despu&eacute;s de que lo registres
                    </b></font></p>
                <font face="Arial, Verdana, Helvetica, sans-serif" size="3">
                <b>Si no est&aacute;s registrado haz click</b> <a href="" /cgi-bin/registro-nuevo-libra.pl">
                <b>aqu&iacute;a.</b></a>. <br> si ya has publicado anuncios anteriormente
                s&eacute; lo ingresa tu n&aacute;mero de usuario <br>
                <p><br>
                <font size="3" face="Arial, Verdana, Helvetica, sans-serif">
                <b>CLAVE DE IDENTIFICACI&Oacute;N</b> <br>
                (Sin guiones, ni espacios ) <br>
                <br>
                <input type="password" name="numero" maxlength="15" size="15">
                <input type="hidden" name="metop" value="0">
                </font></p>
                <p align="left">
                <input type="submit" name="enviar" value="Ingresar">
                </p>
                </font>
            </form>
        </td>
        </tr>
        </table>
        </td>
        </tr>
        </table>
        </td>
        </tr>
        </table>
        </td>
        </tr>
        </table>
        </td>
        </tr>
        </table>
        <td width="5" bgcolor="#FFCC66"></td>
        </tr>
        </table>
        </td>
        </tr>
        </table>
        <p>&nbsp;</p>
        <p>&nbsp;</p>
        <p>&nbsp;</p>
        <SCRIPT language="javascript">
        !* Certifica.com Version 3 00 (01/05/2001)*!
        Certifica_resolucion="otro".
        Certifica_colores="otro".
        Certifica_java_activated="otro".
        Certifica_browser="otro".
        Certifica_version="otro".
        Certifica_desde="" +escape(document.referrer).
        Certifica_cpu_XXX="otro".
        Certifica_idioma_sistema="otro".
        if (Certifica_dosdo=="")
        {
            Certifica_desde="otro".
        }
        Certifica_resolucion=screen.width.
        Certifica_browser=navigator.appName.
        Certifica_version_navegador=navigator.appVersion charAt(0) + navigator.appVersion charAt(1).
        Certifica_version_navegador=Certifica_version_navegador + navigator.appVersion charAt(2).
        Certifica_cpu_XXX=navigator.cpuClass.
        Certifica_idioma_sistema=navigator.userLanguage.
        if (Certifica_browser != "Netscape")
        {
            Certifica_colores=screen.colorDepth.
        } else {
            Certifica_colores=screen.pixelDepth.
        }
        Certifica_java_activated=navigator.javaEnabled().
    
```

```

Certifica_NS2Ch_a=0;
if (navigator appName == "Netscape" && navigator appVersion charAt(0) == "2")
{
  Certifica_NS2Ch_a=1;
}
if (Certifica_NS2Ch_a == 0) {
  Certifica_url_source="http://htts e c/certifnt dll?sitio_id=5408&area_id=25963&pagina_id=431749&vc=3.00";
  Certifica_url_source+="&size="+Certifica_resolucion+"&colors="+Certifica_colores;
  Certifica_url_source+="&reforers="+Certifica_desde+"&java="+Certifica_java_activado;
  Certifica_url_source+="&idio="+Certifica_idioma_sistema+"&cpu="+Certifica_cpu_XXX+"&vers="+Certifica_version_navegador+"";
  document.writeln("<A HREF='http://www.certifica.com/link/index.asp?sitio_id=5408&vc=3.00'> <IMG BORDER=0
SRC='\""+Certifica_url_source+"\"></A>\"");
} // ...
</SCRIPT>
<NOSCRIPT>
<A HREF='http://www.certifica.com/link/index.asp?sitio_id=5408&vc=3.00'><IMG BORDER=0
SRC='http://htts e c/certifnt dll?sitio_id=5408&area_id=25963&pagina_id=431749&vc=3.00'></A>
</NOSCRIPT>
</body>
</html>

```

#### reglamento.htm

```

<html>
<head>
<title>Aviso de Ocas&oacute;n UNAM</title>
<meta http-equiv="Content-Type" content="text/html, charset=iso-8859-1">
<script language="JavaScript">
<!--
function MM_swapimgRestore() { //v3.0
var i,x,a=document MM_sr; for(i=0,a&&i<a.length&&(x=a[i])&&x.oSrc;i++) x.src=x.oSrc;
}
function MM_preloadImages() { //v3.0
var d=document, ifd=images[]; if(!d.MM_p) d.MM_p=new Array();
var i,j,d MM_p.length,a=MM_preloadImages arguments, for(i=0, i<a.length, i++)
if (a[i].indexOf("#")!=""&& i<a.length) d.MM_p[i]=new Image, d.MM_p[i].src=a[i];
}
function MM_findObj(n, d) { //v3.0
var p,x, if(!d) d=document, if(!p=n.indexOf("(")&&parent.frames.length) {
d=parent.frames[n.substring(p+1)] document, n=n.substring(p,0);
}
if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0, x&&i<d.forms.length, i++) x=d.forms[i][n];
for(i=0, x&&d.layers&&i<d.layers.length, i++) x=MM_findObj(n, d.layers[i].document), return x,
}
function MM_swapImage() { //v3.0
var i,j=0, x,a=MM_swapImage arguments, document MM_sr=new Array, for(i=0, i<(a.length-2), i+=3)
if ((x=MM_findObj(a[i]))!=null){document MM_sr[j++]=x, if(!x.oSrc) x.oSrc=x.src, x.src=a[j+2]}
} //...
</script>
</head>
<body leftmargin="0" topmargin="0" marginwidth="0" marginheight="0"
onLoad="MM_preloadImages(gifs/art_esc_on.gif,pggs/artescopa.jpg, gifs/equip_comp_on.gif,pggs/equip.jpg, gifs/veh_h_on.gif,pggs/vehicul
os.jpg, gifs/musica_on.gif,pggs/musica.jpg,pggs/ense%Fanza.jpg, gifs/ense_on.gif,pggs/alomiento.jpg, gifs/aloja_on.gif,pggs/mascot
as.jpg, gifs/masco_on.gif,pggs/diversos.jpg, gifs/diver_on.gif,pggs/boi_reg_on.jpg" links="000000" bgcolor="003399">
<table width="100%" border="0" cellspacing="0" cellpadding="0" bordercolordark="00E0E0" align="left">
<tr>
<td width="67" height="66"><a href="http://www.unam.mx"></a></td>
<td align="center" valign="middle"></td>
</tr>
<tr>
<td width="67" align="center"></td>
<td valign="top" align="left">
<table width="713" border="0" cellspacing="0" cellpadding="0" align="left">
<tr>
<td width="5" bgcolor="#FF9900"></td>
<td width="100%" valign="top" align="center">
<table width="703" border="0" cellspacing="0" cellpadding="0">
<tr>
<td colspan="3">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td bgcolor="#FF9900">&nbsp;&nbsp;&nbsp;</td>
<td bgcolor="#FF9900">&nbsp;&nbsp;&nbsp;</td>
<td bgcolor="#FFCC66">&nbsp;&nbsp;&nbsp;</td>
</tr>
</table>
</td>
</tr>
<tr>
<td colspan="3">
<table width="100%" border="0" cellspacing="0" cellpadding="0">

```

```

</tr>
<td width="18" bgcolor="#FFFFFF" height="18"></td>
<td bgcolor="#FFFFFF" height="18" align="center">
<font face="Verdana, Arial, Helvetica, sans-serif" size="2">
<a href="http://www.unam.mx">UNAM</a>
<font color="#FF9900">
|<font color="#000000"><a href="http://www.mapa.unam.mx">Mapa de CU</a></font>
|<font color="#000000"><a href="http://www.unam.mx/indices/wwwunam.htm">WWW en
la UNAM</a></font>
|<font color="#000000"><a href="http://www.dgsca.unam.mx">DGSCA</a></font>
|&nbsp;.<font color="#000000"><a href="index.html">Inicio </a></font>
|&nbsp;.<font color="#000000"><a href="individual.html"><b>Mis avisos</b>
</a></font>
|&nbsp;.<font color="#000000"><a href="creditos.htm">Cr&eacute;ditos</a></font>
</td>
</tr>
</td>
<td width="17" bgcolor="#FFFFFF" align="right" height="18"></td>
</tr>
</table>
</td>
</tr>
<tr align="center" valign="top">
<td colspan="3" bgcolor="#FFFFFF" height="325">
<table border=0 cellpadding=0 cellspacing=0 width="703" height="325">
<tbody>
<tr align="center" valign="top">
<td width="28%">
<p><font face="Tahoma, Arial"> </font>
<center>
<table width="650" border="0" cellspacing="0" cellpadding="0" height="290">
<tr>
<td height="30">
<table width="640" border="0" cellspacing="0" cellpadding="0" height="20" align="center">
<tr>
<td width="220" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=1&titulogl=artesco_t.gif"
onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimgaga('articulos','','gifs/art_esc_on.gif','mientras','','jpps/artesco.jpg,1)'"></a></td>
<td width="197" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=7&titulogl=equicompl_t.gif"
onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimgaga('equipo','','gifs/equip_comp_on.gif','mientras','','jpps/equip.jpg,1)'"></a></td>
<td width="130" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=11&titulogl=vehiculos_t.gif"
onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimgaga('ense&ntilde;anza','','gifs/vehl_on.gif','mientras','','jpps/vehiculos.jpg,1)'"></a></td>
<td width="121" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=12&titulogl=musica_t.gif"
onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimgaga('ajamiento','','gifs/musica_on.gif','mientras','','jpps/musica.jpg,1)'"></a></td>
</tr>
</table>
</td>
</tr>
<tr align="center" valign="top">
<td height="240">

```

para la confirmaci&ocirc;le,n del aviso </font></p>
 <p><font face="Verdana, Arial, Helvetica, sans-serif">6 - La duraci&ocirc;le,n del aviso en el sistema ser&acute,s de tres semanas, si en ese lapso no lo vendes, tendr&acute,s la opci&ocirc;le,n de renovarlo por una semana m&acute,s, se te avisar&acirc;le,n por mail con tres d&iacute;as de anticipaci&ocirc;le,n para que mandes un e-mail a <a href="mailto:ocasion@dragon.dgsca.unam.mx">ocasion@dragon.dgsca.unam.mx</a> para renovar, esta renovaci&ocirc;le,n s&ocirc;le,n pod&acirc;le,n hacerlo dos veces </font></p>
 <p><font face="Verdana, Arial, Helvetica, sans-serif">7 - S&ocirc;le,n lo se publicar&acirc;le,n bienes cuyo comercio no se encuentre expresamente prohibido por la ley </font></p>
 <p><font face="Verdana, Arial, Helvetica, sans-serif">8 - No se publicar&acirc;le,n anuncios cuyo contenido viole las leyes sobre derechos de autor, marcas, modelos, etc </font></p>
 <p><font face="Verdana, Arial, Helvetica, sans-serif">9 - El usuario ser&acirc;le,n responsable en cualquier caso de la veracidad del contenido del anuncio. La UNAM solamente proporciona el servicio de poner en contacto a las personas, por lo que no

asume ninguna responsabilidad sobre la veracidad de los anuncios publicados, ni sobre los resultados de las transacciones derivadas de ellos. </font></p>
 <p><font face="Verdana, Arial, Helvetica, sans-serif">10 - La UNAM se reserva el derecho de publicar anuncios en este servicio </font></p>
 <p><font face="Verdana, Arial, Helvetica,

sans-serif">11 - Para publicar una foto junto con su anuncio, esta deber&acirc;le,n corresponder al art&iacute;culo en cuesti&ocirc;le,n, de lo contrario su anuncio no ser&acirc;le,n tomado en cuenta </font></p>
 <p><font face="Verdana, Arial, Helvetica,

sans-serif">12 - El registro de anuncios en este servicio implica la aceptaci&ocirc;le,n plena del presente reglamento </font></p>
 </div>

```

</td>
</tr>
<tr align="center" valign="top">
<td height="20">
<tr>
<td width="165" border="0" cellspacing="0" cellpadding="0" height="33" align="center">
<tr>
<td width="161" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=6&titulogf=ensenanza_t.gif"
onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimg('mientras','','pgs/ensenanza.jpg','vehiculos','','gifs/ense_on.gif,1)'"></a></td>
<td width="198" height="20"><a href=" /cgi-bin/listas2-
libre.pl?categoria=2&inicial=1&contareg=1&final=5&titulogf=alojamiento_t.gif" onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimg('mientras','','pgs/alojamiento.jpg','muebles','','gifs/aloja_on.gif,1)'"></a></td>
<td width="168" height="20"><a href=" /cgi-bin/listas2-libre.pl?categoria=8&titulogf=mascofas_t.gif"
onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimg('mientras','','pgs/mascotas.jpg','electrodomesticos','','gifs/masco_on.gif,1)'"></a></td>
<td width="141" height="20"><a href=" /cgi-bin/listas2-
libre.pl?categoria=4&inicial=1&contareg=1&final=5&titulogf=diversos_t.gif" onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimg('mientras','','pgs/diversos.jpg','bienes','','gifs/diver_on.gif,1)'"></a></td>
</tr>
</td>
</tr>
</table>
</tr>
<table width="703" border="0" cellspacing="0" cellpadding="0" align="center">
<tr>
<td width="18" height="26" align="left" valign="top">
<div align="left"></div>
</td>
<td width="668" align="center" valign="middle" height="26">
<div align="center"><a href="registro-libre.htm" onMouseOut="MM_swapimgRestore()"
onMouseOver="MM_swapimg('imagen1','','pgs/bot_reg_on.jpg,1)'"></a></div>
</td>
<td width="17" height="26" align="right" valign="top">
<div align="right"></div>
</td></tr>
</table>
</td>
</tr>
</tbody>
</table>
</div>
</tr>

```

```
<tr>
  <td colspan="3">
    <table width="100%" border="0" cellspacing="0" cellpadding="0" bordercolordark="#003898">
      <tr>
        <td bgcolor="#FFF990" height="10">&nbsp;</td><td bgcolor="#FFF990" height="10">&nbsp;</td><td bgcolor="#FFCC66"
height="10">&nbsp;</td></tr>
      </tr>
    </table>
  </td>
</tr>
</table>
</tr>
</table>
</td>
<td width="5" bgcolor="#FFCC66"></td>
</tr>
</table>
</tr>
</table>
</tbody>
</html>
```

## B. Pantallas del Sistema

F: Aviso de Ocasión UNAM Netscape

File Edit View Go Comunicador Help

Back Forward Reload Home Search Netscape Print Security Shop Close

Location: http://www.deocasion.unam.mx/ What's Related

Instant Message WebMail Radio People Yellow Pages Download Calendar Channels

# Aviso de Ocasión UNAM

UNAM - Mizada CU - WWW en la UNAM - DGECA - Mis avisos - Créditos

ARTICULOS ESCOLARES EQUIPO DE COMPUTO VEHICULOS MUSICA

Bienvenido al Aviso de Ocasión UNAM  
Un espacio abierto a la Comunidad Universitaria.

En este sitio se publican anuncios de intercambio que abarcan todas las áreas de la comunidad universitaria.

Por favor, los usuarios registrados, clic en el botón de gestión de la cuenta.

Para que sus anuncios cumplan las reglas.

Comenzamos de nuevo, visiten nuestra página  
[www.deocasion.unam.mx](http://www.deocasion.unam.mx)

ENSEÑANZA ALOJAMIENTO PERRAS Y MASCOTAS DIVERSOS

REGISTRO

<http://www.deocasion.unam.mx>  
Página de inicio del Aviso de Ocasión

Aviso de Ocasión UNAM Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Shop History

Bookmarks Location http://www.deocasion.unam.mx/creditos.htm What's Related

Instant Message WebMail Radio People Yellow Pages Download Calendar Channels

## Créditos

UNAM MEXICO WWW.ENLUNAM DSCA MEXAVISIA Inicio

ARTICULOS ESCOLARES EQUIPO DE COMPUTO SERVICIOS MUSICA

Sitio WEB desarrollado e implementado por la  
Coordinación de Servicios de Red

Universidad Nacional Autónoma de México

Dirección General de Servicios de Cómputo Académico  
Director General  
Dr. Victor Manuel Guerra Ortiz

Subdirección de Servicios de Cómputo Académico  
Ing. Francisco Becerra Caballero

Coordinación de Servicios de Red  
Coordinadora  
L. I. Luz María Ramírez Romero

Incorporación y Extensión de Servicios  
Jefe de Departamento  
Ing. César Francisco German Rosas

Programación  
Sonia Carolina Madrigal Loyola  
Octavio César Rodríguez Arellano

Administración de Servicios Institucionales

Document Done

<http://www.deocasion.unam.mx/creditos.htm>  
Página de créditos del Aviso de Ocasión



<http://www.deocasion.unam.mx/registro-libre.htm>

Página de inicio para utilizar el sistema, una vez que ya se ha registrado el usuario y cuenta con una clave de acceso, así como, para aquellos que aún no se registran.



Registro de Usuario Nuevo - Netscape

File Edit View Go Comunicador Help

Back Forward Reload Home Search in Netscape Print Security Stop

Bookmarks Location http://www.deocasion.unam.mx/cgi-bin/registro-nuevo-libre.pl

Instant Message WebMail Radio People Yellow Pages Download Calendar Chatrooms

## Registro

UNAM: [Mapa de CCU](#) [WWW de la UNAM](#) [DGECA](#) [Instit. de Estudios](#) [Creditos](#)

**DATOS PERSONALES**  
(Los datos marcados con asterisco son necesarios para registrarte)

|   |  |
|---|--|
| <p>Nombre(s) _____</p> <p>*Apellido(s) _____</p> <p>Telefono _____</p> <p>INEC (Sin Homoclave)<br/><small>El INEC es un número de identificación personal que se genera al momento de la inscripción en el sistema de la UNAM. Este número es necesario para poder acceder al sistema de la UNAM. Si ya tienes un INEC, puedes ingresarlo en este campo. Si no lo tienes, puedes solicitarlo en el área de atención al usuario de la UNAM.</small></p> <p>Dirección _____</p> <p>E-mail _____</p> | <p>Sexo<br/>Masculino <input checked="" type="checkbox"/> Femenino <input type="checkbox"/></p> <p>Ocupación<br/>[Elegir una opción] _____<br/>Si eligió "otro" especificar por favor _____</p> <p>Nivel de estudios:<br/>[Elegir una opción] _____<br/>Si eligió "otro" especificar por favor _____</p> |
|---|--|

[Cancelar] [Registrar]

Document Done

<http://www.deocasion.unam.mx/cgi-bin/registro-nuevo-libre.pl>  
Página que contiene la forma a llenar por aquellos usuarios que aún no están registrados en el sistema.

Alfa de Usuario Netscape  
File Edit View Go J Comunicador Help  
Back Forward Reload Home Search Netscape Print Security Shop Stop  
Bookmarks Location http://www.deocasion.unam.mx/cgi-bin/usuario-nuevo-libre.pl  
Internet Message WebMail RSS Feeds People Yellow Pages Downloads Calendar Channels  
What's Related

## Aviso de Ocasión UNAM

UNAM Mapa de CV WWW en la UNAM ROSCA Inicio Mis cursos Créditos

Esta es tu clave de usuario, gústala y con ella podrás acceder al sistema para publicar tus trabajos

**MAL8980926047**

Este es tu ID de usuario con que has sido registrado

Nombre: **SOMIA CAROLINA MADRIDAL LOYOLA**  
RFC: **MAL2980926**  
Residencia: **TICHUCHITILAH ME9 LT 6-A COL REY HEZA CD IZAPANALCOVOYILE EDO DE MEX**  
Teléfono: **54110309**  
Correo Electrónico: **becami@mal8 de la unam.mx**  
Sexo: **F**  
Organización: **Estudiante UNAM**  
Escribes: **Licenciatura**

Document Done

<http://www.deocasion.unam.mx/cgi-bin/usuario-nuevo-libre.pl>  
Página que confirma el registro de un nuevo usuario al sistema.

Untitled Document - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Shop Stop

Location: http://www.deocasion.unam.mx/cgi-bin/valu3-libre.pl

Instant Message WebMail Radio People Yellow Pages Download Calendar Channels

# Registro

LITZRA MORA DE CUJ WACON EN LA UNAM DOSCA Inicio Mis anncios Cerrar

SELECCIONA LA CATEGORIA PARA TU ANUNCIO

|   |  |  |  |
|---|--|--|--|
| <p>VEHICULOS</p> <ul style="list-style-type: none"> <li>• Autos usados</li> <li>• Moto ciclitas</li> <li>• Bicicletas</li> <li>• Otros</li> </ul> | <p>MUSICA</p> <ul style="list-style-type: none"> <li>• Clases</li> <li>• Instrumentos</li> <li>• Grabaciones</li> <li>• Otros</li> </ul> | <p>ARTICULOS ESCOLARES</p> <ul style="list-style-type: none"> <li>• Camisetas</li> <li>• Escaritas</li> <li>• Libros</li> <li>• Otros</li> </ul> | <p>EQUIPO DE COMPUTO</p> <ul style="list-style-type: none"> <li>• Servicios</li> <li>• Perifericos</li> <li>• Programas para equipo de cómputo</li> <li>• Otros</li> </ul> |
| <p>ENSEÑANZA</p> <ul style="list-style-type: none"> <li>• Clases Particulares</li> <li>• Cursos</li> <li>• Otros</li> </ul>                       | <p>MASCOTAS</p> <ul style="list-style-type: none"> <li>• Perros</li> <li>• Gatos</li> <li>• Otros</li> <li>• Accesorios</li> </ul>       |  |  |

Document Done

http://www.deocasion.unam.mx/cgi-bin/valu3-libre.pl  
 Página con la que el usuario inicia el registro de un anuncio.

Unfiled Document Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print No Scripts Stop Stop

Bookmarks Location: todos=kt&kt=USUARIO&telefono=kt&id=Semal=numero=MAL598092047&recuerda=Perifericos What's Related

Instant Message WebMail Radio People Yellow Pages Download Calendar Channels

# Registro

1997 Miraz de CUI WWW.en la UTM de ROSCA Inicio Mis Avisos Estado

|  |   |
|--|---|
| <p><b>DATOS PERSONALLS</b></p> <p>Nombre SONIA CAROLINA MADRIGAL LOTOLA</p> <p>Telefono 54410300</p> <p>e-mail becami@mitta.deca.unam.mx</p> | <p><b>DATOS SOBRE EL ARTÍCULO</b><br/>(EQUIPO Perifericos)</p> <p>Los datos con asterisco son obligatorios</p> <p>*Artículo<br/>(Nombre, raza o marca de su artículo)</p> <p>*Descripción:<br/>(Breve descripción del aviso, no más de 150 letras)</p> <p>Precio:<br/>(Puede incluir e no el precio. Ejemplo \$300.9233.55, etc.)</p> <p>Foto</p> |
|--|---|

Document Done

<http://www.deocasion.unam.mx/cgi-bin/registro-de-aviso-libre.pl>  
 Página en la que el usuario describe las características del artículo o servicio en venta.

The screenshot shows a Netscape browser window with the following details:

- Address Bar:** <http://www.deocasion.unam.mx/cgi-bin/uploadindice.pl>
- Page Title:** Registro
- Navigation Bar:** Back, Forward, Reload, Home, Search, Netscape, Print, Security, Shop, Stop.
- Toolbar:** Bookmarks, Location, Instant Message, WebMail, Find, People, Yellow Pages, Download, Calendar, Channels.
- Page Content:**

UNAM Mapa de CU WWW en la UNAM COSCA Inicio Mis avisos Créditos

**ESTOS SON LOS DATOS REGISTRADOS DE TU ANUNCIO**

HOMBRE: SONIA CAROLINA  
APELLIDOS: MADRIGAL LOYOLA  
TELÉFONO: 5410109  
EMAIL: BEC316@MITLA.DCAA.UNAM.MX  
CATEGORÍA: EQUIPO  
ARTÍCULO: TECLADO  
DESCRIPCIÓN: MARCA NOMBRE DESCOMPUESTO, ACEPTO CUALQUIER COSA A CAMBIO  
PRECIO:  
FOTO: Sin foto  
FECHA: 09-06-2001

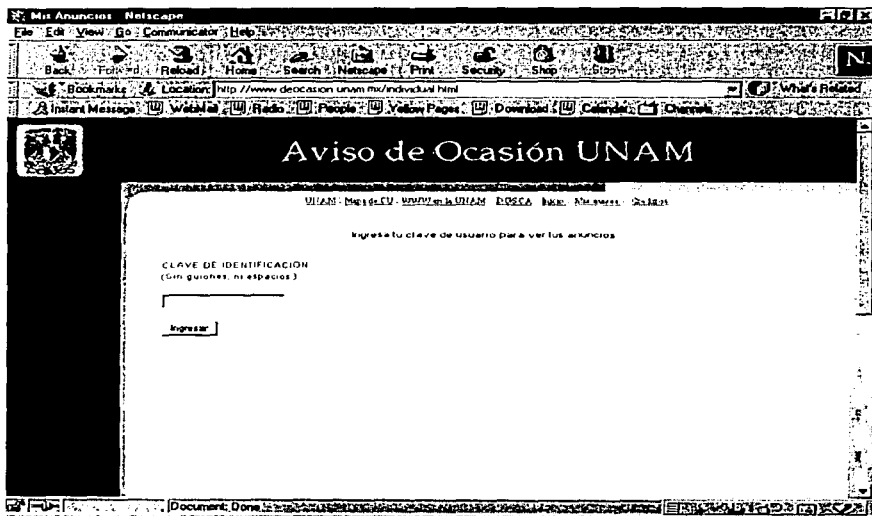
Para confirmar tu anuncio, simplemente haz clic en el botón de aceptar de tu anuncio.
- Status Bar:** Document Done

<http://www.deocasion.unam.mx/cgi-bin/uploadindice.pl>  
Página en la que se confirma al usuario los datos del anuncio registrado.



<http://www.deocasion.unam.mx/cgi-bin/valu3-libre.pl>

Página en la que se le notifica al usuario que la clave que introdujo no existe en la base de datos y, por lo tanto, no es válida.



<http://www.deocasion.unam.mx/individual.html>

Página en la que el usuario puede consultar qué avisos tiene publicados.

Mis anuncios Netscape

File Edit View Go Communications Help

Back Reload Home Search Netscape Print Security Stop

Bookmarks Location: <http://www.deocasion.unam.mx/cgi-bin/individual.pl>

Install Messages WebMail Radio Pages Web Pages Download Calendar Overview

## Aviso de Ocasión UNAM

UNAM Map de CU WWW en la UNAM CENSA Mapa Mascotas Cursos

CATEGORIA: Mascotas  
CLAVE DEL ANUNCIO: 4660  
CLAVE DEL USUARIO: GALA350726487  
ARTICULO: COOKIES  
DESCRIPCION: BUSCO NOVIA PARA RUFFO, PEPITO COLOR MEL CLARO, RAZA PURA DE DOS AÑITOS. COMUNICATE  
PRECIO:  
FOTO:  
FECHA DE VENCIMIENTO: Sep 17 2001 12:00AM

[Detallar](#)

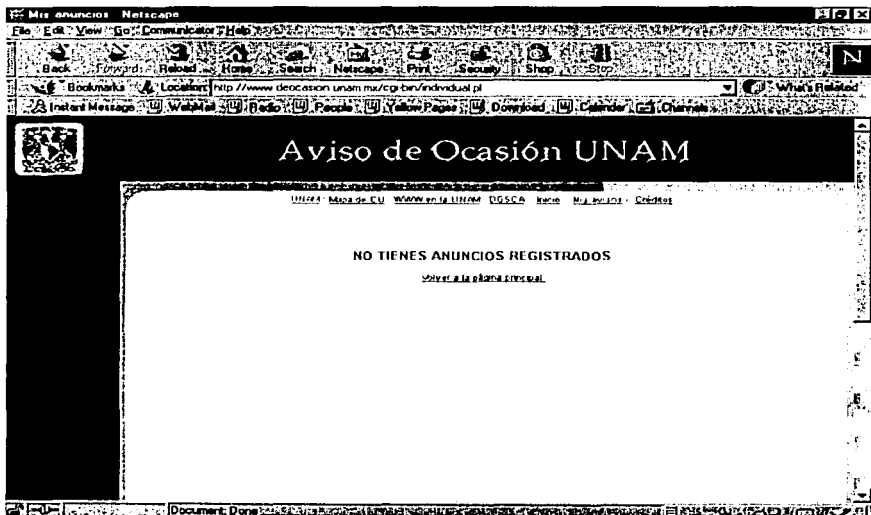
[Volver a la página anterior.](#)

Document Done

<http://www.deocasion.unam.mx/cgi-bin/individual.pl>

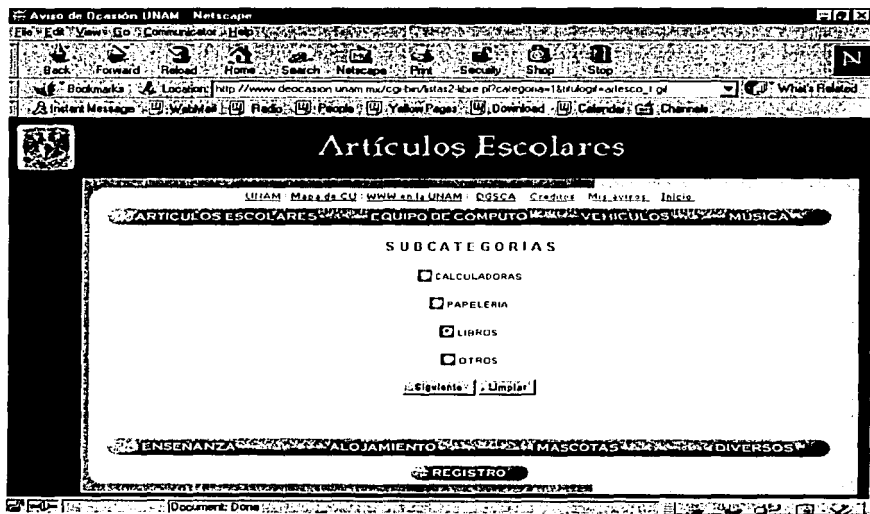
Página en la que se le muestra al usuario la información de los anuncios que tiene publicados.





<http://www.deocasion.unam.mx/cgi-bin/individual.pl>

Página en la que se le notifica al usuario que no tiene anuncios publicados, ya sea porque aún no se dan de alta o su clave no existe en la base de datos.



<http://www.deocasion.unam.mx/cgi-bin/listas2-libre.pl>

Página que muestra las subcategorías de la categoría seleccionada (ya sea Artículos Escolares, Equipo de Cómputo, Vehículos, Música o Mascotas) en el caso de las categorías Enseñanza y Alojamiento, muestra directamente la información de los anuncios publicados.

Aviso de Decisión UNAM Netscape

File Edit View Go Communications Help

Back Forward Reload Home Search Netscape Print Security Shop Stop

Bookmarks Location: <http://www.deocasion.unam.mx/cgi-bin/listas3-libre.pl> What's Related

Instant Message WebMail RSS Feeds People Yellow Pages Downloads Delete Channels

## Artículos Escolares

UNAM MEXICO CV WWW.UNAM.mx DEOCA Decisión México Inicio

ARTICULOS ESCOLARES EQUIPO DE COMPUTO VEHICULOS MUSICA

**No. 1**  
 Artículo: ESTADISTICA NO PARAM  
 Descripción: ESTADISTICA NO PARAMETRICA, APLICADA A LAS CIENCIAS DE LA CONDUCTA. EDIT TRILLAS 2DA EDICION.  
 POR SIEGEL L CASTELLAN. MUY BUEN ESTADO.  
 Precio: 181 PESOS.

ESPERANZA LUISA GARRIDO O  
 Telefono: 55509200  
 E-mail: [ggarrido@deoca.com](mailto:ggarrido@deoca.com)

---

**No. 2**  
 Artículo: ANATOMIA MOORE  
 Descripción: ANATOMIA CON ORIENTACION CLINICA. TERCERA EDICION. KEITH L. MOORE. MUY BUEN ESTADO, PARECE  
 NUEVO.  
 Precio: \$430.00

BEATRIZ GARAY  
 Telefono: 56-44-31-38  
 E-mail: [beatrizout@correo.unam.mx](mailto:beatrizout@correo.unam.mx)

---

**No. 3**

Document Done

<http://www.deocasion.unam.mx/cgi-bin/listas3-libre.pl>

Página en la que se despliegan los resultados de la búsqueda por subcategoría, para artículos publicados en las categorías Artículos Escolares, Equipo de Cómputo, Vehículos, Música, Enseñanza y Mascotas.

## C. GLOSARIO

### TÉRMINOS

@ (arroba): Este signo es uno de los componentes de las direcciones de correo electrónico y separa el nombre del usuario de los nombres de dominio del servidor de correo (ejemplo: srmsz@milla.dcaa.unam.mx); el origen de su uso en Internet está en su frecuente empleo en inglés como abreviatura de la preposición *at* (en).

Acuse de recibo: Es un paquete de información con un formato propio que se utiliza para indicar la transmisión exitosa de otro paquete de información de un transmisor a un receptor.

Administrador de la red (*Network administrator*): Persona cuyo objetivo principal es, mantener la operatividad y buen desempeño de la red para satisfacer las necesidades de los usuarios; para ello se auxilia de distintas herramientas (tanto *software* como *hardware*).

API (Interfaz para Programas de Aplicación/*Application Program Interface*): Conjunto de convenciones de programación que definen cómo se invoca un servicio desde un programa.

Aplicación: Cada uno de los programas que una vez ejecutados, permiten trabajar con la computadora para realizar alguna tarea concreta, como son procesadores de texto, hojas de cálculo, etc.

ASCII (Código Estándar Americano para el Intercambio de Información, *American Standar Code for Information Exchange*): Conjunto de normas de codificación de caracteres en el que cada letra, dígito y símbolo es representado por un número, es de amplia utilización en informática y telecomunicaciones.

*Backbone*: Centro primario de conexión en un sistema de red distribuido. Todos los sistemas que se conectan al *backbone* pueden, a la vez, conectarse entre sí.

*Banner* (Anuncio, faldón, pancarta, viñeta): Palabra usada en Internet para indicar la publicidad de forma gráfica o sólo texto, que aparecen en las páginas *Web*. Puede decirse que son los patrocinadores de dichas páginas.

Base de datos: Conjunto estructurado, como su nombre dice, de datos. Estos datos son representaciones de sucesos y objetos, a diferente nivel, existentes en el mundo real: en su conjunto, representan algún tipo de entidad existente. En el mundo real se tiene percepción sobre las entidades u objetos y sobre los atributos de esos objetos; en el mundo de los datos, hay registros de eventos y datos de eventos.. Además, en ambos escenarios se puede incluso distinguir una tercera faceta: aquella que comprende las definiciones de las entidades externas, o bien, las definiciones de los registros y de los datos.

**Bit:** Es la unidad de información más pequeña digital que no puede ser dividida en elementos menores. Un bit sólo puede tener dos estados: 0 o 1, encendido (presencia de corriente eléctrica) o apagado (ausencia de corriente eléctrica).

**Bookmark** (Libro de marcas, marca de página, marca, marcapáginas): Nombre del directorio en el que se guardan las direcciones en Internet de los sitios (URL) que más frecuentemente visita el usuario y que son de mayor interés para él. A ese directorio también se le conoce como "favoritos".

**Broadcast:** Tecnología de transmisión en la que un nodo se comunica absolutamente con todos los nodos conectados a él, en una sola transmisión.

**Browser** (Navegador): Aplicación para visualizar todo tipo de información almacenada en páginas HTML que se encuentran disponibles en servidores Web.

**CGI** (*Common Gateway Interface*/Interfaz Común de Pasarela): Interfaz de intercambio de datos estándar en WWW, a través de la cual, se organiza el envío y recepción de datos entre navegador y programas residentes en servidores WWW.

**Circuito Virtual:** Es una conexión en la que se determina la ruta lógica (del transmisor al receptor) que va a seguir un mismo flujo de información, es decir, que no se requiere elegir una ruta nueva para cada paquete que pertenece a dicho flujo.

**Consola:** Nombre asignado a las terminales conectadas a una computadora central gracias a las cuales, es posible monitorear el funcionamiento, así como acceder al sistema de las mismas.

**Compilador:** Programa que lee las líneas escritas en un lenguaje de programación y las traduce a otro que pueda ejecutar la computadora. Los programas compilados se ejecutan más rápido que los interpretados, debido a que han sido completamente traducidos a lenguaje de máquina y no necesitan compartir memoria con el intérprete.

**Cookie:** Dentro de la *Web*, procedimiento ejecutado por el servidor, que consiste en guardar información acerca del cliente para su posterior recuperación. En la práctica, la información es proporcionada desde el navegador al servidor *Web*, a través de un archivo de texto, que queda almacenado en el disco duro de la computadora y puede ser recuperado posteriormente, cuando se acceda al servidor.

**Correo Electrónico:** Expresión usada para referirse a todo lo relacionado con la gestión, envío y recepción de mensajes por medios electrónicos.

**Criptografía:** Término formado a partir del griego *kruptos*, oculto (Arte de escribir con clave secreta o de un modo enigmático). Es criptográfico cualquier procedimiento que permita a un emisor ocultar el contenido de un mensaje de modo que sólo personas en posesión de determinada clave puedan leerlo, tras haberlo descifrado.

**Demonio (*Daemon*):** Aplicación UNIX que está alerta permanentemente en un servidor para realizar determinadas tareas como, enviar un mensaje de correo electrónico.

**Dirección:** En Internet, serie de caracteres, numéricos o alfanuméricos, que identifican un determinado recurso de forma única y permiten acceder a él.

**Dirección IP (*IP Address*):** La dirección IP consta de 32 bits y, se utiliza para identificar tanto a la computadora en concreto como la red a la que pertenece, de manera que sea posible distinguir a las computadoras que se encuentran conectadas a una misma red. Se representa usualmente mediante notación decimal separada por puntos. Un ejemplo de dirección IP es 193.127.88.345

**Direccionamiento:** Es la asignación de una dirección (canónica o numérica) de Internet. La forma canónica es más fácil de leer, por ejemplo: minombre@ticip.com. La forma numérica es un número binario sin signo de 32 bits, habitualmente expresado en forma de números decimales separados por puntos. Por ejemplo, 9.167.5.8 es una dirección de Internet válida. La forma numérica es usada por el *software* de IP. La función de mapeo entre los dos la realiza el DNS.

**Dominio (*Domain*):** Nombre alfanumérico ASCII con el que se conocen las direcciones de Internet usadas para localizar los servidores existentes que se encuentran conectados que están formadas por una dirección separada por puntos con al menos dos partes: la primera es más concreta y designa la compañía, institución, organización, etc., mientras que la segunda, conocida como sufijo de código de país o el tipo de organización al que pertenece la primera parte. Los dominios se usan como alternativa a tener que usar los números de las direcciones IP reales para acceder a los sistemas de red, los cuales resultan mucho más difíciles de recordar que los nombres de dominio.

**DNS (*Sistema de Nombre de Dominios/Domain Name System*):** Se trata de una base de datos distribuida que gestiona todo lo referente a la conversión de nombres de dominio en formato alfanumérico (lenguaje escrito) al sistema real de direcciones de Internet formado por grupos de 4 números a las cuales se llama direcciones IP. Y, funciona como un servicio de búsqueda de datos de uso general, distribuido y multiplicado. Su utilidad principal es la búsqueda de direcciones IP de sistemas anfitriones (*hosts*) basándose en los nombres de éstos. El estilo de los nombres de *host* utilizado actualmente en Internet es llamado "nombre de dominio". Algunos de los dominios más importantes, que sin embargo son muy escasamente utilizados fuera de los Estados Unidos de América, son: .com (comercial-empresas), .edu (educación, centros docentes), .org (organización sin ánimo de lucro), .net (operación de la red), .gov (Gobierno USA) y .mil (ejército USA). La mayoría de los países tienen un dominio propio, por ejemplo, .mx (México).

**Encapsulamiento:** Nombre que se da en programación al proceso de agrupar información abstraída (verabstraer) de objetos con las operaciones (métodos) que un programa debe realizar con dicha información. En redes, es el proceso de tomar paquetes de un sistema de red y colocarlos dentro de otro sistema de red.

**Encriptación, Cifrado (*Encryption*):** El cifrado es el tratamiento de un conjunto de datos, contenidos o no en un paquete, a fin de impedir que nadie excepto el destinatario de los mismos pueda leerlos. Hay muchos tipos de cifrado de datos, que constituyen la base de la seguridad de la red.

**Ethernet:** Sistema de red de área local de alta velocidad desarrollado por las compañías Xerox, Digital e Intel, que se ha convertido en un estándar en lo que a métodos de acceso LAN respecta.

**Gateway (pasarela):** Hoy se utiliza el término *router* (direccionador, encaminador, enrutador) en lugar de la definición original de *gateway*. Una pasarela es un programa o dispositivo de comunicaciones que transfiere datos entre redes que tienen funciones similares pero implantaciones diferentes.

**GUI (Interfaz Gráfica de Usuario/*Graphical User Interface*):** Sistema de interacción entre el ordenador y el usuario, caracterizado por la utilización de iconos y elementos gráficos en su concepción. Es un paso más allá de las interfaces basadas en caracteres, que sólo incluían líneas de texto para introducir comandos y conocer las respuestas del sistema.

**Hardware:** Conjunto de componentes físicos de un sistema informático. Cada una de las partes físicas que integran una computadora, incluyendo los periféricos.

**Herencia:** Término utilizado en programación que consiste en la habilidad para poder escribir una clase, la cual, hereda variables y métodos de otras clases. Gracias a esta capacidad, se pueden crear clases que incorporen nuevas características respecto a las clases ya existentes.

**Host (Anfitrión):** Computadora que, mediante la utilización de los protocolos TCP/IP, permite a los usuarios comunicarse con otros sistemas anfitriones de una red. Los usuarios se comunican utilizando programas de aplicación (correo electrónico, Telnet, WWW, FTP).

**HTML (Lenguaje de Marcado de Hipertexto/*Hypertext Markup Language*):** Lenguaje en el que se escriben las páginas a las que se accede a través de navegadores WWW. Admite componentes hipertexto y multimedia.

**HTTP (Protocolo de Transferencia de Hipertexto/*Hypertext Transfer Protocol*):** Protocolo encargado de realizar las transferencias entre los servidores y los navegadores *Web*.

**IMP (*Interface Message Processor*):** Es el antecedente de lo que hoy denominamos ruteador.

**Indexar:** Acción de ordenar los registros contenidos en una base de datos de una forma especial en función de parámetros predefinidos.

**Intercambio Electrónico de Datos (*Electronic Data Interchange, EDI*):** Es aquella parte de un sistema de información capaz de cooperar con otros sistemas de información mediante el intercambio de mensajes EDI.

**Intérprete:** Programa traductor de lenguajes de programación de alto nivel, que interpreta y ejecuta el programa simultáneamente. Se encarga de traducir una sentencia de programa a lenguaje de máquina, y de ejecutarla repitiendo las demás sentencias en forma sucesiva.

**Intranet:** Red informática de comunicación privada de una empresa u organización de cualquier tipo que funciona con las mismas tecnologías usadas en la red Internet y que se puede conectar a ella de forma sencilla dada su misma naturaleza tecnológica.

**IP (Protocolo de Internet/*Internet Protocol*):** Es el protocolo estándar utilizado por los dispositivos que se comunican por Internet. Actualmente existen dos versiones de este protocolo, el Ipv4 y el Ipv6.

**ISO (Organización Internacional para la Estandarización/*International Standards Organization*):** Es una organización que ha definido un conjunto de protocolos diferentes, llamados protocolos ISO/OSI. Esta organización de carácter voluntario fue fundada en 1946 y es responsable de la creación de estándares internacionales en muchas áreas, incluyendo la informática, las ecológicas y las comunicaciones.

**IT (Tecnologías de la Información/*Information Technologies*):** Forma de denominar al conjunto de herramientas (habitualmente de naturaleza electrónica) utilizadas para el recibimiento, almacenamiento, tratamiento, difusión y transmisión de la información.

**Kilobit:** Unidad de medida de memoria que indica la cantidad de 1 000 bits.

**Kilobyte:** Unidad de medida de memoria que indica la cantidad de 1 024 bits.

**Lenguaje de interrogación:** Es el conjunto de órdenes, operadores y estructuras que, organizados según unas normas lógicas, permiten la consulta de fuentes y recursos de información electrónica. El resultado de la combinación de estos elementos, siguiendo las normas establecidas, es una expresión, a la que se identifica con el nombre "ecuación", capaz de interrogar el contenido de la fuente de información.

**Lenguaje de programación:** Conjunto de normas sintácticas que especifican cómo y cuándo pueden usarse un conjunto determinado de instrucciones, al que se le da un nombre específico. A partir de los lenguajes de programación, es posible diseñar programas, los cuales, serán convertidos a código máquina por el compilador de dicho lenguaje para su ejecución.

**Línea redundante:** El concepto de línea redundante consiste en unir dos puntos no sólo por una única línea, sino por varias que realicen la conexión por caminos diferentes para, así esta manera, aumentar las posibilidades de supervivencia en caso de fallo.

**Link (Liga o enlace):** Es un indicador (texto o imagen) link sirve como enlace a otro documento HTML.



**Logitr:** Nombre que identifica a un Usuario de un sistema específico.

**Macrocomputadoras (Mainframe):** Computadora multiusuario concebida para cubrir requisitos de computación de grandes empresas, se caracteriza por ser capaz de controlar muchas terminales y unidades de almacenamiento.

**Memorándums de Entendimiento (Memorandum of Understanding):** La participación de instituciones extranjeras en Internet2 se establece a través de "Memorándums de Entendimiento", conocidos como MoU, es decir, documentos que establecen acuerdos de trabajo mutuo con el objeto de fijar metas comunes para los países o redes participantes del Proyecto. Se halla abierta la participación de esas instituciones, así como la de miembros norteamericanos; es decir, cualquier organización con interés en establecer una relación basada en un MoU con Internet2, y con condiciones técnicas para tal, podrá hacerlo. Por lo general, las instituciones interesadas son organizaciones comprometidas en alcanzar metas similares a las del Proyecto I2 en sus respectivos países, además de universidades, centros de investigación e instituciones sin fines lucrativos.

**Metatags (Metacaracteres):** Sentencias de código HTML que, si bien no son visibles desde el explorador, ayudan a que los motores de búsqueda localicen un sitio cuando se realiza una búsqueda. Las etiquetas dentro del código HTML son: título (*title*), descripción (*description*) y palabras clave (*keywords*) mismas que, deben contener la información más apropiada y contundente relacionada al sitio *Web*.

**Minicomputadora:** Computadora multiusuario concebida para cubrir necesidades de una empresa o de un departamento de la misma. Una minicomputadora es más poderosa que una computadora personal pero menos que un *mainframe*.

**MIME (Extensión Multipropósito de Correo de la Internet/Multipurpose Internet Mail Extension):** Conjunto de especificaciones Internet de libre distribución que permiten, tanto el intercambio de texto escrito en lenguajes con diferentes juegos de caracteres, como el intercambio de archivos de diversos formatos entre computadoras y aplicaciones que sigan los estándares de correo Internet.

**Mirar (espejo, duplicación, réplica):** Servidor Internet cuyo contenido es una copia exacta de otro servidor o de parte de él. Normalmente, este tipo de servidores cuenta con la aprobación del servidor original y sirven para reducir el tiempo de acceso del usuario a servidores situados en lugares muy distantes.

**Módem:** Palabra inglesa resultante de unir los términos MODulador-DEModulador. Es un dispositivo que permite conectar la computadora a una red informática por medio de la red telefónica. En el proceso de emisión, el dispositivo modula la información y en el proceso de recepción demodula la señal recibida y reconstruye la original.

**Multiprocesamiento:** Se refiere a tener dos o más procesadores para resolver una tarea.

**Multiprogramación:** Capacidad o característica de un procesador, para atender varias tareas con un tiempo mínimo de diferencia.

**Multipuerto:** Los dispositivos que poseen esta característica, tienen la capacidad de ser conectados a más de una computadora.

**Multiusuario:** Característica que permite que varios usuarios ejecuten simultáneamente más de un programa en un mismo sistema.

**Netiqueta:** Reglas éticas o código de buena conducta o de buen uso en Internet.

**Nodos:** Nombre genérico que se le da a cualquier dispositivo *hardware* con su correspondiente *software* que se encuentra conectado a una red y que es accesible desde la misma.

**Outsourcing:** Situación que se da cuando una organización o empresa usa las instalaciones físicas de otra, que le provee además mantenimiento y le proporciona las aplicaciones necesarias para que pueda desarrollar sus gestiones.

**Password:** Clave que identifica a un Usuario de un sistema específico.

**Plug in (Conector):** Pequeño programa que añade alguna función a otro programa, habitualmente de mayor tamaño. Un programa puede tener uno o más conectores. Son muy utilizados en los programas navegadores para ampliar sus funcionalidades.

**Portabilidad:** Característica de ciertos programas que les permite ser utilizados en distintos sistemas de cómputo sin que precisen modificaciones de importancia.

**Postscript:** Lenguaje de descripción de páginas considerado hace años como un estándar. Está diseñado especialmente para imprimir documentos con calidad profesional y alta definición en impresoras láser.

**Proceso distribuido:** Método de procesamiento informático que se caracteriza porque los datos y los programas están distribuidos entre las distintas computadoras que conforman la red.

**Programa:** Bloque secuencial de instrucciones escritas en lenguaje de programación determinado diseñado para resolver un problema concreto.

**Protocolo:** Conjunto de normas de diseño electrónico y de programación que permiten abrir un canal de comunicación entre dos computadoras.

**Polimorfismo:** Término usado en programación para referirse a la habilidad de un objeto para adoptar diferentes formas. Esto se refiere al hecho de que múltiples objetos puedan poseer interfaces idénticas.

**POP:** Protocolo diseñado para permitir a sistemas de usuario individual, leer correo electrónico almacenado en un servidor. La Versión 3 es la más reciente y la más utilizada (POP3).

**Puerto:** Es un elemento *hardware* que se usa como conector mediante el cual la computadora puede recibir y enviar datos de y hacia los periféricos que se le conecten. También se le llama puerto al canal por el que puede circular información en Internet.

**Red:** Sistema de comunicaciones e intercambio de información que se crea mediante la conexión física y lógica de dos o más computadoras.

**Redes de Telecomunicaciones:** Como parte fundamental de los sistemas de información. Una Red de Telecomunicaciones es el conjunto de elementos (*hardware* y *software*) que permiten el transporte electrónico de voz, sonidos, datos, textos, imágenes u otras señales de cualquier naturaleza entre dos o más puntos (nodos).

**Red Privada de Telecomunicaciones:** Red destinada exclusivamente a servicios privados de telecomunicaciones, es decir que no soporta ningún servicio público o comercial de telecomunicaciones, ni servicios privados de terceros y que no se hallan interconectadas a una red pública en ninguna forma que permita la prestación de servicios públicos o comerciales de telecomunicaciones.

**Red Pública de Telecomunicaciones:** Red que se explota para prestar servicios públicos o comerciales de telecomunicaciones, o sea, utilizada para soportar servicios privados o públicos de terceros y, será accesible a todos los que la requieran en igualdad de condiciones, precio, disponibilidad de acceso y trato no discriminatorio.

**Remoto:** Palabra muy genérica utilizada en informática para referirse a computadoras o periféricos que se encuentran físicamente separados de una unidad central de proceso. El término también se refiere al mantenimiento de sistemas a distancia, al acceso de aplicaciones residentes en unidades físicamente distantes, etc.

**Robot:** Programa que atraviesa la *Web* con distintos fines: Análisis sobre el crecimiento de la red, número de servidores activos; Mantenimiento de la estructura hipertexto de la *Web*; Duplicación de directorios FTP (*Mirrors*), para incrementar su disponibilidad a un mayor número de usuarios; Investigación, localizando recursos *Web* para el usuario.

**Ruteador (*Router*):** Palabra inglesa que se significa "encaminador". Es un dispositivo que tiene dos o más interfaces de red. Este examina las direcciones de los protocolos que se utilizan en la red y selecciona una ruta apropiada de viaje para la información y manda un paquete de reconocimiento entre las diferentes redes. Los ruteadores usualmente mandan paquetes correspondientes a una simple familia de protocolos.

**Ruteo:** Es el proceso de seleccionar una ruta sobre la cual se enviarán los paquetes de datos. Este proceso se inicia en un host TCP/IP cuando envía paquetes P a un receptor y se complementa con un receptor P.

**Script:** Conjunto de caracteres formado por mandatos o secuencias de texto, que se utiliza muy a menudo en internet para automatizar tareas muy habituales como, por ejemplo, la conexión a la red local.

**Sistema Operativo:** Conjunto de programas que administran los recursos del sistema para su mejor aprovechamiento. Hay un administrador para cada uno de los recursos: Administrador de Memoria, Administrador de Dispositivos de Entrada/Salida, Administrador de Almacenamiento, Administrador de Información, Administrador del Procesador y Administrador de Protección y Tratamiento de Errores. A cada administrador le es un programa. Por lo tanto, el sistema operativo es un conjunto de programas.

**SMTP:** Protocolo que se usa para transferir correo electrónico entre computadoras. Es un protocolo de servidor a servidor de tal manera que para acceder a los mensajes es preciso utilizar otros protocolos.

**Snet/Finameador:** Término inglés usado para referirse al programa que monitorea los paquetes de datos que circular por una red en busca de una cadena numérica o de caracteres con objeto de conseguir alguna información. Normalmente se usa con fines legales.

**Socket:** Es un punto final de un enlace de comunicación de dos vías entre dos programas que se ejecutan a través de la red.

**Software:** Conjunto de programas de distinto tipo que hacen posible operar con la computadora.

**SQL (Lenguaje Estructurado de Consulta/Structured Query Language):** Es un estándar en el lenguaje de acceso a bases de datos adoptado por ISO. Originalmente, era un lenguaje de acceso al sistema de gestión de bases de datos denominado DB2.

**SSL (Capa de Conexiones Seguras/Secure Sockets Layer):** Protocolo que ofrece funciones de seguridad (encriptando la información proporcionada de manera confidencial, ya sea a un proveedor, una base de datos, etc.) a nivel de la capa de transporte para TCP.

**Supercomputadora:** Refinada y costosa computadora diseñada para ejecutar cálculos complejos a la máxima velocidad permitida por los avances tecnológicos actuales. Las supercomputadoras se usan en la investigación científica.

**Tabla de ruteo:** Contiene información del nodo que va a enviar a los nodos por donde pueda transmitir la información y por donde se va a transferir, tomando como base la dirección de red.

**Ruteo:** Es el proceso de seleccionar una ruta sobre la cual se enviarán los paquetes de datos. Este proceso se inicia en un host TCP/IP, cuando envía paquetes IP a un ruteador y, se complementa con un ruteador IP.

**Script:** Conjunto de caracteres formado por mandatos y secuencias de teclado, que se utiliza muy a menudo en Internet para automatizar tareas muy habituales como, por ejemplo, la conexión a la red (*login*).

**Sistema Operativo:** Conjunto de programas que administran los recursos del sistema para su mejor aprovechamiento. Hay un administrador para cada uno de los recursos (Administrador de Memoria, Administrador de Dispositivos de Entrada/Salida, Administrador de Almacenamiento, Administrador de Información, Administrador del Procesador y, Administrador de Protección y Tratamiento de Errores) y, cada administrador es un programa. Por lo tanto, el sistema operativo es un conjunto de programas.

**SMTP:** Protocolo que se usa para transferir correo electrónico entre computadoras. Es un protocolo de servidor a servidor, de tal manera que para acceder a los mensajes es preciso utilizar otros protocolos.

**Sniffer (Husmeador):** Término inglés usado para referirse al programa que monitorea los paquetes de datos que circulan por una red en busca de una cadena numérica o de caracteres con objeto de conseguir alguna información. Normalmente se usa con fines ilegales.

**Socket:** Es un punto final de un enlace de comunicación de dos vías entre dos programas que se ejecutan a través de la red.

**Software:** Conjunto de programas de distinto tipo que hacen posible operar con la computadora.

**SQL (Lenguaje Estructurado de Consulta/*Structured Query Language*):** Es un estándar en el lenguaje de acceso a bases de datos adoptado por ISO. Originalmente, era un lenguaje de acceso al sistema de gestión de bases de datos denominado DB2.

**SSL (Capa de Conexiones Seguras/*Secure Sockets Layer*):** Protocolo que ofrece funciones de seguridad (encriptando la información proporcionada de manera confidencial, ya sea a un proveedor, una base de datos, etc.) a nivel de la capa de transporte para TCP.

**Supercomputadora:** Refinada y costosa computadora diseñada para ejecutar cálculos complejos a la máxima velocidad permitida por los avances tecnológicos actuales. Las supercomputadoras se usan en la investigación científica.

**Tabla de ruteo:** Contiene información del nodo que va a enviar a los nodos por donde puede transmitir la información y por donde se va a transferir, tomando como base la dirección de red.

**Teletipo (Teletype TTY):** Método de representación de caracteres en un monitor en el cual, los caracteres se generan y envían, uno a uno, a la pantalla de un monitor; luego, a medida que se reciben, la pantalla se va llenando línea a línea. Una vez llena, la pantalla se recorre hacia arriba para acomodar las nuevas líneas de caracteres que aparecen en la parte inferior.

**Terminal:** Pequeño equipo informático formado normalmente por un monitor y un teclado, que se sitúa bastante lejos de la computadora central (*host*) al que está conectado y desde donde se pueden pedir servicios a la computadora central.

**Tramas:** Se refiere a la unidad de datos que viaja de una computadora a otra.

**URL (Localizador Universal de Recursos, *Universe Resource Locator*):** Formato estandarizado a nivel mundial para dar nombre de dirección a todos los servidores conectados a la red Internet. Un URL contiene 3 partes: el protocolo para extraer la información (HTTP, FTP, etc.), el nombre de la computadora que le sirve y, por último, la ruta o camino hacia el documento.

**UNIX:** Sistema operativo multitarea y multiusuario, lo cual significa que puede ejecutar varios programas y que puede gestionar a varios usuarios de manera simultánea. Se desarrolló en 1969 en los laboratorios Bell (por Brian W. Kernighan y Ken Thompson), y aunque al principio se diseñó para el PDP-11, una máquina de Digital, ahora se ejecuta en gran cantidad de plataformas con muchos tipos de microprocesadores diferentes, haciéndolo un sistema multiplataforma y, provocando por tanto que un programa en código máquina ejecutable en una plataforma en UNIX no tenga por qué ser ejecutable en otra.

**UUCODE:** Método de codificación para enviar archivos binarios como textos. Se usa para enviar este tipo de archivos por correo, bien a otros usuarios. Para poder convertirlo de nuevo a un archivo binario hace falta decodificarlo. Algunos programas lectores de correo o de *news* incorporan un decodificador.

**Web Server (Servidor Web):** Dentro de la *World Wide Web*, programa que acepta solicitudes de información que cumplen con el protocolo HTTP. El servidor procesa las solicitudes y envía lo que se le ha solicitado.

**Webring (Anillo de Webs):** Un *Webring* se forma a partir de un grupo de interés. La idea parte básicamente de agrupar a páginas *Web* con un tema en común.

Frecuentemente, cuando se visitan páginas que tratan el mismo tema, se puede encontrar al pie de alguna, un icono que dice: "xxxx Webring - Next Page". Eso significa que la página actual, pertenece a un *Webring* y que si se da *click* en ese icono se puede pasar a ver otra página que trata el mismo tema, hecha por otro autor.

Esto les permite a los creadores independientes de páginas, poder unirse y difundir su trabajo de manera totalmente gratuita. El concepto de un *Webring* es justamente colaborar desinteresadamente unos con otros, poniendo un link a la siguiente página que trate de lo mismo.

La persona interesada, puede unirse a un *Webring* existente o, crear el suyo propio.

*Workstation* (Estación de Trabajo): En redes, se refiere a una computadora de escritorio que ejecuta programas de aplicación y sirve como punto de acceso a la red.

*Wysiwyg* (Lo que se ve es lo que se obtiene/*What you see is what you get*): En informática quiere decir que lo que se ve en la pantalla es lo que se va a obtener una vez que se imprima (en algunos casos no siempre sucede). Todo depende del *software* que se utilice.

## SIGLAS

ANS Red Avanzada y Servicios, Inc. (*Advanced Network and Services, Inc.*)

API Interfaz para Programas de Aplicación (*Application Program Interface*)

ARPA Agencia para Proyectos de Investigación Avanzada (*Advanced Research Projects Agency*)

ASCII Estándar Americano de Codificación para el Intercambio de Información (*American Standard Code for Information Interchange*)

BBN *Bolt Beranek and Newman Ltd.*

CAN Red de Área de Campus (*Campus Area Network*)

CANARIE Red Canadiense para la Investigación Avanzada de la Industria y la Educación (*Canadian Network for Advancing Research Industry and Education*)

CCR Investigación sobre Sistemas de Mando y Control (*Command and Control Research*)

CENIC Corporación de las Iniciativas de la Red de la Educación en California (*Corporation of Education Network Initiatives in California*)

CERN Centro Europeo para la Investigación Nuclear (*European Center for Nuclear Research*)

CIDR Dominio de Ruteo Sin Clase (*Classless Internet Domain Routing*)

CIX Asociación de Intercambios Comerciales en Internet (*Comercial Internet Exchange*)

CUDI Corporación Universitaria para el Desarrollo de Internet

DARPA Agencia de la Defensa para Proyectos de Investigación Avanzada (*Defense Advanced Research Projects Agency*)

DoD Departamento de Defensa de los Estados Unidos (*Departmant of Defense*)

EDI Intercambio Electrónico de Datos (*Electronic Data Interchange*)

GAN Red de Área Global (*Global Area Network*)

GUI Interfaz Gráfica de Usuario (*Graphical User Interface*)

FTP Protocolo de Transferencia de Archivos (*File Transfer Protocol*)

HTML Lenguaje de Marcado de Hipertexto (*Hypertext Markup Language*)

HTTP Protocolo de Transferencia de Hipertexto (*Hypertext Transfer Protocol*)

ICCC Conferencia Internacional sobre Comunicaciones por Computadora (*International Conference on Computer Communications*)

INWG Grupo de Trabajo de Redes Internacional (*International Network Working Group*)

IPTO Oficina para las Tecnologías de Procesado de la Información (*Information Processing Techniques Office*)

ISO Organización Internacional para la Estandarización (*International Standards Organization*)

ISOC Sociedad Internet (*Internet Society*)

ISP Proveedor de Servicios de Internet (*Internet Service Provider*)

IT Tecnologías de la Información (*Information Technologies*)

Iz Internet 2

IW3C2 Comité Internacional de Conferencia del *World Wide Web* (*International World Wide Web Conference Commite*)

LAN Red de Área Local (*Local Area Network*)

MAN Red de Área Metropolitana (*Metropolitan Area Network*)

MIME Extensión Multipropósito de Correo de la Internet (*Multipurpose Internet Mail Extension*)

MIT Instituto Tecnológico de Massachusetts (*Massachusetts Institute of Technology*)



NAP Puntos de Acceso a la Red (*Network Access Points*)

NCC Centro de Control de la Red (*Network Control Center*)

NCAR Centro Nacional para la Investigación Atmosférica (*National Center for Atmospheric Research*)

NCSA Centro Nacional para Aplicaciones de Supercómputo de Illinois (*National Center for Supercomputing Applications*)

NFS Fundación Nacional de Ciencias de los E.U. (*National Science Foundation*)

NOC Centro de Operaciones de la Red (*Network Operation Center*)

NWG Grupo para el Funcionamiento de la Red (*Network Working Group*)

OASC Oficina de Cómputo Científico Avanzado (*Office of Advanced Scientific Computing*)

OLTP Procesamiento de Transacciones en Línea (*Online Transaction Processing*)

ORB Corredor de Solicitudes de Objetos (*Object Request Broker*)

OSI Modelo de Referencia de Interconexión de Sistemas Abiertos (*Reference Model of Open Systems Interconnection*)

PERL Lenguaje Práctico de Extracción e Informes (*Practical Extraction and Report Language*)

POP Protocolo de Oficina de Correos (*Post Office Protocol*)

QoS Garantía de calidad en el servicio (*Quality-of-Service*)

RFC Petición para Comentarios (*Request For Comments*)

SMTP Protocolo Simple de Transferencia de Correo (*Simple Mail Transfer Protocol*)

SQL Lenguaje Estructurado de Consulta (*Structured Query Language*)

SRI Instituto de Investigaciones de Stanford (*Stanford Research Institute*)

SSH *Secure Shell*

SSL Capa de Conexiones Seguras (*Security Socket Layer*)

TIMP Terminal IMP

UCAID Corporación Universitaria para el Desarrollo Avanzado de la Internet (*University Corporation for Advanced Internet Development*)

UCSD Universidad de California en San Diego

URL Identificador Universal de Recursos (*Uniform Resource Locators*)

WAIS Servicio de Información de Red Amplia (*Wide Area Information Service*)

WAN Red de Área Extensa (*Wide Area Network*)

WWW Telaraña Amplia y Mundial/Telaraña de información mundial (*World Wide Web*)

## **D. REFERENCIAS**

### **BIBLIOGRÁFICA**

#### **Redes de ordenadores**

Tanenbaum, Andrew S.

Tr. Víctor Manuel Carbajal Castañeda

Prentice Hall Hispanoamericana

759 p.p.

TK5105.5 T3518

México, D.F. 1991

#### **Análisis estructurado moderno**

Eward Yourdon

Tr. Alexandra Taylor Armitage

Prentice Hall Hispanoamericana

735 p.p.

QA76.9.S84 Y6818

México, D.F. 1993

#### **Redes Globales de Información con Internet y TCP/IP. Principios básicos, protocolos y arquitectura**

Douglas E. Comer

Tr. Hugo Alberto Acuña Soto

Prentice Hall

3ra. edición.

621 p.p.

**México 1996**

#### **Apuntes de clase: Programación de Interfaces Gráficas**

Ismael Israel Perea Camarillo

Facultad de Contaduría y Administración

Licenciatura en Informática

Semestre 1998-I

#### **Sistemas de Información para los negocios. Un enfoque de toma de decisiones**

Daniel Cohen Kanen, Enrique Asin Lares

McGraw-Hill

413 p.p.

HD30.23 C645 2000  
México 2000, 3ra edición

**Breve Historia de Internet**

Ranz Abad, Jesús  
Anaya Multimedia  
210 p.p.  
TK5105.875157 R37  
Madrid, España 1997

**Diccionario de Informática e Internet**

Enrique de Alarcon Álvarez  
Anaya Multimedia  
351 p.p.  
QA76.15 A53  
Madrid, España 2000

**Diccionario de términos de computación**

Bryan Pfaffenberger  
Tr. César Germán Romero Solís, Rebeca Alicia Sánchez López  
Prentice Hall  
560 p.p.  
QA76.15 P43818  
Nauicalpan de Juárez Edo. de México 1999

**ELECTRÓNICA**

**PORTALES**

<http://www.um.es/gliweb/portales/>  
<http://www.ciberestrella.com/secciones/saber/saber0.htm>  
<http://personales.mundivia.es/astruc/manina07.htm>  
<http://www.topmadrid.com/consultorios/eprofecias/art/portals.asp>  
[http://www.gestiondelconocimiento.com/documentos2/juan\\_carrion\\_maroto/articulo\\_prof\\_portales.htm](http://www.gestiondelconocimiento.com/documentos2/juan_carrion_maroto/articulo_prof_portales.htm)  
<http://www.terra.es/personal/mms00001/curso~portales00.htm>  
<http://dewey.uab.es/pmarques/evaport2.htm>  
<http://www.galiciaonline.es/busca2.htm>  
<http://www.um.es/gliweb/portales/>

## SISTEMAS

<http://ciberconta.unizar.es/>  
<http://www.sortes.com/>  
<http://www.monografias.com/trabajos/1gralsis/1gralsis.shtml>  
<http://www.monografias.com/trabajos/anaydisis/anaydisis.shtml>  
<http://ciberconta.unizar.es/LECCION/GLOBAL/OOOF2.HTM>  
<http://www.illp.edu.mx/publica/tutoriales/analisis/index.htm>  
<http://www.geocities.com/elplanetamx/informacionsistemas.html>  
<http://www.ucm.es/info/multidoc/multidoc/revista/cuadern5/blesa.htm>  
<http://www.nmp.com.mx/>  
<http://www.scientificamerican.com/2001/0501issue/0501berners-lee.html>  
<http://www.w3.org/2001/sw/>  
<http://www.w3.org/TR/2001/WD-rdf-nt-20010925/>  
<http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/>  
<http://www.w3.org/People/Berners-Lee/>  
<http://www.acm.org/crossroads/espanol/xrds5-4/multiagent.html>  
[http://www.ibit.org/websemantica/glossari\\_es.cfm](http://www.ibit.org/websemantica/glossari_es.cfm)  
<http://www.nature.com/nature/webmatters/agents/agents.html>  
<http://siddall.denison.edu/talks/wis/index.htm>  
<http://www.december.com/web/develop.html>  
<http://www.wiwi.hu-berlin.de/~myra/WIS-Mining/index.html>  
<http://ausweb.scu.edu.au/aw99/papers/bauer/>  
<http://ausweb.scu.edu.au/aw99/papers/scharl/>  
<http://www.enredando.com/cas/cgi-bin/entrevista/plantilla.pl?ident=70>  
[http://www.compucentro.com.co/espanol/notas/Julio\\_2000/Julio\\_2000.htm](http://www.compucentro.com.co/espanol/notas/Julio_2000/Julio_2000.htm)  
<http://www.lacompu.com/desarrollo/notas/activexexam/index.php3>  
<http://www.20.brinkster.com/fmartinez/tecnological.htm>  
[http://mailweb.udlap.mx/~tesis/ostrovskaya\\_ya/capitulo3.pdf](http://mailweb.udlap.mx/~tesis/ostrovskaya_ya/capitulo3.pdf)  
<http://www.lacasadegara.org/nueva/timreyweb.html>  
<http://www.acm.org/crossroads/doc/indices/features.html>  
<http://www.bib.uc3m.es/~mendez/RDF/schema/CR-rdf-schema.htm>  
<http://www.sidar.org/traduc/rdfesp.htm>

## BUSCADORES

<http://www.abcdatos.com/buscadores/>  
<http://personal.redestib.es/necora/buscador/l.htm>  
<http://www.abcdatos.com/tutoriales/internet.html>  
<http://www.desarrolloweb.com/articulos/205.php>  
<http://www.vdigitalm.com/csi005.htm>  
<http://www.nexo.net/utilidades/consejosbusqueda.asp>

## INTERNET2

<http://www.cudi.edu.mx/>  
<http://www.launion.edu.pe/info99-22.htm>  
<http://mailweb.udlap.mx/~oneve/intern2.htm>  
<http://www.uam.mx/internet2/>  
[http://www.mnp.br/mnp2\\_es/mnp2-intemet2.html](http://www.mnp.br/mnp2_es/mnp2-intemet2.html)

## INTERNET

<http://www.isoc.org/>  
<http://www.ibarrolaza.com.ar/zakon/hit.html>  
<http://www.idg.es/iworld/especial/origen/origen.html>  
<http://www.nerja.net/aen/manual/general/home.html>  
<http://www.pangea.org/~pacoc/ppf/interno0.htm>  
<http://www.simonsays.com/tilles/O684812010/wizards.html>  
<http://www.baquia.com/com/legacy/8782.html>  
<http://tejo.usal.es/~nines/d.alumnos/ipv6/index.html>

## SERVICIOS DE INTERNET

<http://www.uv.mx/ayudas/unix/>  
<http://www.rcanaria.es/ccaf/intehtml/contenid.html>  
[http://www3.uniovi.es/Vicest/Otros/FAQs/www.spanish.html#hy\\_global](http://www3.uniovi.es/Vicest/Otros/FAQs/www.spanish.html#hy_global)  
<http://www.geocities.com/SiliconValley/Horizon/2598/cap04.html>  
<http://www.learnthenet.com/spanish/html/20how.htm>

## COMERCIO ELECTRONICO

<http://www.finmall.com.mx/comelec.htm>  
<http://cei.mrecic.gov.ar/public/int0999/coelec.htm>

## FUTURO DE INTERNET Y WWW

<http://www.baquia.com/com/20010523/art000008.html>  
<http://www.baquia.com/com/20000716/art000004.html>

## REDES

<http://tiny.uasnet.mx/prof/cdn/ccu/mario/REDES/>  
<http://www.cs.virginia.edu/~knabe/iic3512/apuntes.html>

## SITIOS DE LA UNAM

<http://www.unam.mx>  
<http://www.dgsca.unam.mx>  
<http://www.asc.unam.mx/>  
<http://www.unam.mx/sabueso>  
<http://www.mapa.unam.mx/>  
<http://www.tuobra.unam.mx/cgi-bin/vercero.cgi>

<http://www.ser.unam.mx>  
<http://www.unam.mx/becas/becas.htm>  
<http://www.unam.mx/dov/INDEXO3.HTML>  
<http://www.unam.mx/foros/foros.htm>  
<http://paidoteca.dgsca.unam.mx/chat/>  
<http://www.correo.unam.mx/>  
<http://www.agendacultural.unam.mx/>  
<http://difusion.cultural.unam.mx/>  
<http://www.unam.mx/dgadyr/>

## ESTADÍSTICAS

<http://www.cerifica.com>