UNIVERSIDAD DON BOSCO

FACULTAD DE INGENIERIA

" DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN BUSCADOR DE SITIOS WEB NACIONALES "

TRABAJO DE GRADUACIÓN
PREPARADO PARA LA FACULTAD DE INGENIERIA

PARA OPTAR AL GRADO DE:
INGENIERO EN CIENCIAS DE LA COMPUTACIÓN

PREPARADO POR:
ROLANDO FRANCISCO ALVAREZ CAMPO
JORGE ALEXANDER CRUZ BAUTISTA
GUADALUPE MEJIA SALGUERO

ASESOR:
LIC. JAIME ALBERTO MELÉNDEZ RAMÍREZ

JUNIO - 2000
SOYAPANGO - EL SALVADOR - CENTRO AMERICA

UNIVERSIDAD DON BOSCO

RECTOR

INGENIERO FEDERICO MIGUEL HUGUET RIVERA

SECRETARIO GENERAL

PBRO. PEDRO JOSE GARCIA CASTRO, S.D.B.

DECANO DE LA FACULTAD DE INGENIERIA

INGENIERO CARLOS BRAN

ASESOR DEL TRABAJO DE GRADUACION

LIC. JAIME MELENDEZ

JURADO EVALUADOR

ING. ANA MERCEDES CACERES
ING. MARIA ELENA DE LOBOS

UNIVERSIDAD DON BOSCO

FACULTAD DE INGENIERIA

JURADO EVALUADOR DEL TRABAJO DE GRADUACION

" DISEÑO, DESARROLLO E IMPLEMENTACION DE UN BUSCADOR DE SITIOS WEB NACIONALES "

ING. ANA MERCEDES CACERES
JURADO

LICDA. MARIA ELENA DE LOBOS
JURADO

LIC. JAIME MELENDEZ
ASESOR

Dedico este trabajo de graduación a mis padres y a Dios, quienes con su eterno esfuerzo e incondicional apoyo me han guiado hasta este punto de mi vida; sin ellos nada de esto seria posible.

Agradezco a:

Dios Todopoderoso y a la Virgen María
Por estar siempre con nosotros, por habernos acompañado y guiado, durante todo este proyecto de tesis.

Mi Madrecita Querida, Doña Paquita
Por todos sus cuidados, sus sacrificios, su paciencia y su incondicional amor de madre, aún en los momentos más difíciles y de peor animo. Por haber estado con nosotros cuando nos desanimábamos y por darnos siempre su apoyo.

Mi Querido Papá, Don Paco
Por todos los buenos momentos que me dio y por todo lo que me enseñó, yo sé que en algún lugar por ahí nos acompaño siempre y ahora esta muy alegre por este logró que hemos alcanzado.

Nuestro asesor Jaime Meléndez
Quien se esmeró mucho en su papel, al guiarnos, corregirnos y apoyarnos en todo lo que estaba a su alcance, demostrando interés y siempre estando disponible a nosotros. Muchas Gracias Jaime.

Mis Amigos Guadalupe y Jorge
Por todos los maravillosos momentos que hemos pasado juntos, por su apoyo en los momentos de mayor dificultad, por su comprensión, pero por sobre todo, por su amistad y por ser mis amigos. Muchas Gracias Lupita y Jorge, de todo corazón.

Mis Amigos
Que de una u otra forma nos ayudaron y apoyaron en la culminación de este proyecto y de esta nueva etapa de nuestras vidas. Muchas Gracias a Todos.

Rolando Alvarez.

Agradecimientos

A Dios todopoderoso y a la Virgen María
Por estar siempre a mi lado, por brindarme salud y darme la fuerza de voluntad para lograr esta meta.

A mi Madrecita linda, Laura de Jesús
Quien con su amor y dedicación me alentó siempre a esforzarme y me inclinó siempre por el buen camino. Mil gracias querida mamita.

A mi querido Papá, Jorge Cruz
Quien con su cariño, su apoyo y sus consejos me ayudó a fijar mis objetivos y luchar por ellos. Mil gracias querido papá.

A mis hermanos Mario Alberto y Luis Ernesto
Por su paciencia, su comprensión y su cariño, y sobre todo por el apoyo que me brindaron en los momentos que más lo necesitaba.

A doña Francisca de Alvarez
Por su cariño y su fina atención, por brindarnos mucha comprensión y mantenernos con mucho ánimo. Gracias niña paquita.

A mi amigo Jaime Meléndez
Por ser más que un guía, Por ser una persona humilde que nos regaló mucho de su valioso tiempo y conocimientos.

A mi Amigo del Alma Carlos Roberto
Por su apoyo incondicional y su comprensión hoy y siempre. Gracias hermano.

A mis compañeros y amigos, Guadalupe y Rolando
Por su comprensión y cariño, por la paciencia y el sacrificio que se hizo para finalizar este trabajo, pero sobre todo por su amistad y su solidaridad. Gracias por compartir esos momentos inolvidables.

Jorge Alexander Cruz

Agradecimientos

Agradezco a Dios y a la Virgen
Por estar siempre a mi lado y por permitirme alcanzar otra de mis metas, así como también por la salud y alegrías de este año.

A mis Padres, Blanca Rosa Mejía y Samuel Monge
Por el poyo que me han brindado durante todo este tiempo y por todo lo que han pasado, les doy las gracias de corazón.

A mi Tía, Rosa Linda
Le agradezco por estar siempre pendiente de mis avances en la tesis y por darme ánimos en el momento necesario.

A mis Hermanos, Sonia y Gilberto
Que siempre han estado a mi lado para animarme a seguir adelante.

A las Familias de mis compañeros de tesis
Por estar siempre pendientes y en especial a la Sra. Francisca de Alvarez por ser tan amable y darnos mucho animo y acompañarnos en las noches de desvelo.

A mi Asesor Lic. Jaime Meléndez
Por sus valiosos conocimientos y consejos para el desarrollo del proyecto.

A mis Compañeros de tesis, Rolando y Jorge
Por todos los sacrificios, tristezas y muchas alegrías que pasamos.

Y a todos mis amigos de la universidad y mi trabajo
Que estuvieron pendientes y nos tomaron en cuenta en sus oraciones.

Guadalupe Mejía

Agradecimientos Especiales

Agradecemos de manera especial a todas las personas que nos brindaron su apoyo, amistad y colaboración para el desarrollo de este proyecto.

Ramón E. Díaz
Pablo Valle
Yadira Ríos
Cesar Aguilar
Nelly de Aguilar
Mauricio Mayen
Linda Ibarra
Gloria Hernandez
José A. Posada
Eva Posada
Patricia Merino
Cesar Alfaro
Dennis Mendoza
Josue D. Santin
Ricardo Madrid
Iveth Larios de Madrid
Roberto Ochoa
Coralia Chevez de Chavez
Sara de Flores
Lucy de Jobel
Gotcha, Robin y Carlota

INDICE DE CONTENIDO

INDICE DE ILUSTRACIONES

INTRODUCCION

Actualmente los usuarios de Internet demandan de información precisa y eficiente, muchas veces referente al entorno que los rodea. Esta necesidad convierte a Internet en una fuente de información muy amplia por lo cual es indispensable proveer herramientas que suplan estos requerimientos.

Este documento es una propuesta de proyecto de graduación, titulado ***"Diseño, Desarrollo e Implementación de un Buscador de Sitios Web Nacionales"***, y comprende la investigación y aplicación de los conceptos, procedimientos y métodos que permiten el diseño de un buscador de sitios Web de nuestro país.

En la primera parte se presentan los *Antecedentes y Justificación* del proyecto, enfatizando la necesidad de encontrar información puntual y de carácter regional en el amplio campo del Web. Se plantean los *Objetivos, Alcances y Limitaciones* que definen y aclaran el objetivo primordial perseguido con este proyecto, además de dar a conocer aquellos elementos técnicos que se emplearán para el desarrollo del mismo.

La segunda parte está compuesta por el *Marco Teórico*, que resume los conceptos involucrados en el desarrollo de una herramienta de búsqueda en el Web y de otros elementos relacionados al proyecto.

En la tercera parte del documento se da a conocer el *Desarrollo del Proyecto*, en la cual se plantea la situación actual referente al empleo de herramientas de búsqueda, además se brinda un panorama de los esfuerzos realizados en los procesos de investigación y desarrollo seleccionados, así también se da a conocer la estructura y funcionalidad de Cyber Izalco, mostrando para ello la descripción de los módulos contenidos y tareas realizadas en el motor de búsqueda para cumplir con los objetivos establecidos en el proyecto.

Finalmente se incluyen las conclusiones obtenidas con la investigación y generación de este documento, además se incluyen las recomendaciones y anexos tales como información sobre la empresa en la que se implementará el buscador, el manual de administrativo del motor de búsqueda y el glosario técnico.

CAPITULO I. DESCRIPCION DEL TEMA

1.1 ANTECEDENTES

Internet constituye una de las mayores fuentes de información a escala mundial. En los últimos años su crecimiento ha sido considerablemente rápido, a tal grado que la información disponible aumenta día a día y cada vez es más difícil obtener datos específicos. Esto deja entrever la necesidad de servicios de búsqueda que permitan al usuario típico de Internet acceder de manera eficaz a la información que requiere. Debido a ello surgen buscadores de páginas Web tales como Yahoo, Altavista, Lycos, Infoseek entre otros; los cuales han llegado a convertirse en la principal herramienta de recolección de datos para todo usuario de Internet.

Nuestro país no ha sido la excepción al acelerado incremento de información en el Web, de tal manera que la mayoría de empresas poseen ya un Sitio Web, algunos de estos incluyen acceso a bases de datos (sitios dinámicos), en el cual ponen a disposición su información y ofrecen sus servicios. La creación de SVNET (WWW.SVNET.ORG.SV), organismo nacional coordinador de la red salvadoreña de Internet, ha permitido que las empresas, instituciones y particulares tengan el acceso a todos los servicios de la red y puedan además tener un dominio autorizado para el país.

Como un primer intento para solventar la necesidad de obtener información originada en nuestro país de manera eficiente, en septiembre de 1998 fue desarrollado un buscador nacional por alumnos de la Universidad Centroamericana José Simeón Cañas (UCA). Dicho buscador está construido sobre una plataforma UNIX y utiliza una base de datos MICRO CDS-ISIS. Este tiene dentro de sus desventajas el no contemplar todos los sitios nacionales sino únicamente aquellos del dominio SV, así como la carencia de una interfaz sumamente amigable.

De acuerdo a ello, se plantea este proyecto como el diseño y el desarrollo de un buscador que utilizando una plataforma y base de datos comercial de amplia difusión en nuestro medio; proporcione mediante una interfaz amigable información específica contenida en la diversidad de sitios Web dentro del país, lo que facilitará la proyección de sitios Web nacionales al mundo entero.

1.2 IMPORTANCIA Y JUSTIFICACION

La necesidad de publicar y obtener información a través de Internet se hace cada día más imperativa, lo que contribuye a que la red de comunicación a escala mundial crezca continuamente. El Salvador no se ha quedado ajeno a este desarrollo y es así como día a día aumentan el número de organizaciones, instituciones y personas que publican su información en sitios en Internet.

Sin embargo, encontrar información publicada a nivel nacional dentro de la gran cantidad de sitios que conforman el Web, es una tarea que requiere cada vez más tiempo y esfuerzos; pues aunque existen motores de búsqueda mundiales, es precisamente por su carácter global que se dificulta el acceso a información actualizada publicada localmente, ya que al emplearlos para realizar una búsqueda específica, estos retornan en muchas ocasiones cientos e inclusive miles de enlaces de los cuales una mínima parte contienen los datos realmente requeridos.

Por ello se plantea este proyecto, el cual pretende desarrollar un buscador de sitios Web nacionales sobre una plataforma multitareas, que se caracterice por su facilidad de administración y que sea un sistema operativo ampliamente utilizado en el país. De igual forma la base de datos que se empleará deberá poseer características similares a las del sistema operativo. Con esto se sentarán las bases para que futuros desarrollos en el área sean una realidad alcanzable a corto plazo, debido a la existencia de una infraestructura previa.

De acuerdo a las criterios descritos anteriormente se optará por emplear Windows NT y Microsoft Access, como sistema operativo y base de datos respectivamente. Una ventaja que posee Windows NT es que cuenta con su propio servidor Web, el Internet Information Server, para el cual no se necesita adquirir licencia, por lo que no se incurre en costos adicionales al implantarlo. Este servidor Web provee los mecanismos de seguridad necesarios para proteger la información que se publica, agregando a ello una serie de herramientas gráficas que facilitan la administración del mismo, tales como: Internet Service Manager, Key Manager y otros.

Microsoft Access, es una herramienta de fácil utilización y muy difundida en el medio al igual que Windows NT. Entre sus ventajas se pueden mencionar: una interfaz gráfica de fácil aprendizaje, acceso rápido a datos, soporte de sentencias SQL (Structured Query Language), controladores

ODBC (Open Database Conector) para el acceso a la base desde otras aplicaciones, mecanismos de seguridad robustos y, muy importantemente, un desempeño eficiente en entornos multitarea.

Con el desarrollo del motor de búsqueda planteado se logrará cubrir la demanda de todos aquellos usuarios de Internet que deseen encontrar información publicada en El Salvador, empleando para ello menos tiempo y recursos, de igual forma, al emplear el sistema operativo y la base de datos antes mencionados facilitarán las tareas de administración, operación, actualización y mejora del mecanismo de búsqueda a desarrollar.

1.3 OBJETIVOS DEL PROYECTO

1.3.1 Objetivo General

Diseñar, desarrollar e implementar un buscador de interfaz amigable, que permita a los usuarios de Internet encontrar información de acuerdo a criterios específicos; para lo cual se empleará un Robot Web existente que brinde periódicamente mantenimiento a una base de datos referente al contenido de los sitios Web Nacionales.

1.3.2 Objetivos Específicos

1. Proporcionar a los usuarios del Web, a través del servidor de SALNET S.A. (El Salvador Network), una herramienta de fácil manejo, que permita optimizar recursos en la obtención de información publicada en los sitios nacionales.

2. Implementar un mecanismo de búsqueda de páginas Web sobre una plataforma de amplia difusión a nivel nacional como lo es Windows NT.

3. Almacenar en una base de datos de Microsoft Access información referente a sitios Web nacionales, para la optimización de posteriores búsquedas.

4. Emplear una interfaz compatible con el Internet Information Server, que permita a los usuarios realizar consultas eficientes sobre la información almacenada en la Base de Datos.

5. Documentar todas las fases de investigación y desarrollo del buscador, de tal manera que se genere una guía didáctica, la cual estará a disposición en la Biblioteca de la Universidad Don Bosco, con el propósito de fomentar la generación de proyectos similares a nivel regional.

1.4 ENFOQUE DEL PROYECTO

La utilidad de un buscador es aplicable en todas las áreas, puesto que permite encontrar diversidad de datos según los criterios que se especifiquen. Las principales áreas de nuestro país que se beneficiarán con la existencia del buscador planteado en este proyecto se detallan a continuación:

1. Gubernamental : permitiendo acceso de manera integral a información publicada por ministerios e instituciones estatales.

2. Educación : Las universidades más importantes tienen sus sitios publicados en el web, con el propósito de intercambiar información con instituciones similares a nivel mundial, así como dar a conocer sus planes de estudio. Con el buscador cualquier institución interesada en intercambiar recursos con nuestra comunidad educativa, accederá eficientemente a la rama de su interés.

3. Economía: Transacciones de la bolsa de valores, tipos de cambios vigentes, compra y venta de acciones, estarán a disposición de manera dinámica y puntual para cualquier inversionista tanto nacional como extranjero.

4. Ciencias: Sitios de instituciones como el museo de ciencias Stephen Hawkings y otros, se proyectarán mediante el buscador de acuerdo al interés del usuario.

5. Empresarial: Organizaciones comerciales que pretenden promover sus productos a nivel mundial alcanzarán una mayor proyección pues el buscador reducirá las vías de acceso hacia éstas.

6. Turismo: Fomentar vía electrónica la explotación turística de nuestro país. Pues cualquier turista potencial únicamente especificará en que actividad esta interesado y los sitios virtuales se pondrán a su disposición.

En general, el desarrollo e implementación de este proyecto, está orientado a los usuarios de Internet, que por diferentes motivos requieren información específica publicada en El Salvador; permitiéndoles minimizar esfuerzos de búsqueda para encontrar datos deseados. De igual manera se

beneficiarán todos aquellos usuarios que deseen establecer un contacto más directo con las organizaciones, instituciones, empresas o personas que han publicado dicha información.

Otra de las aplicaciones de la herramienta a desarrollar consiste en el empleo de la documentación generada a través de la investigación, por personas interesadas en el desarrollo de proyectos relacionados. Es de destacar que, por la amplia difusión que existe en nuestro país tanto del sistema operativo como del manejador de base de datos a emplear, los productos generados sentarán las bases sobre un campo lo suficientemente fértil para que se deriven posteriores proyectos, sean estos educativos o empresariales; con lo cual se estaría contribuyendo al desarrollo que nuestra sociedad requiere en estos momentos.

1.5 ALCANCES Y LIMITACIONES

1.5.1 Alcances

1. El proyecto pretende desarrollar un buscador nacional, que brinde al usuario de Internet una herramienta amigable y dinámica que muestre la información requerida.

2. Implementación de todos los componentes necesarios para el mecanismo de búsqueda en el servidor Web de SALNET, incluyendo un robot de búsqueda que alimente automática y periódicamente la base de datos con respecto al contenido de los sitios Web nacionales.

3. El buscador se desarrollará sobre una plataforma comercial de interfaz gráfica, como lo es Windows NT, facilitando así su posterior administración. Así mismo se empleará una base de datos de índole comercial como lo es Microsoft Access.

4. Para el desarrollo del mecanismo de búsqueda se empleará una interfaz compatible con el Internet Information Server, como lo son el Internet Database Conector (IDC) o Active Server Pages (ASP).

5. Se hará uso de las restricciones de acceso que brinda tanto el Internet Information Server como Windows NT, para el manejo de la seguridad de la información almacenada en la base de datos.

6. La interfaz del buscador poseerá ayuda referente a la operación del mismo, facilitando la emisión de criterios de búsqueda por parte de los usuarios.

7. Se elaborará una guía de instalación, configuración y administración de la herramienta resultante, para facilitar su implementación y modificación posterior.

8. La documentación generada con el desarrollo de este proyecto servirá como base para la formulación de proyectos similares en el futuro.

1.5.2 Limitaciones

1. El buscador funcionará solo para aquellos sitios de carácter nacional, como organizaciones, instituciones y empresas nacionales.

2. El buscador funcionará únicamente sobre una plataforma Windows NT empleando una base de datos de Microsoft Access.

3. El buscador se desarrollará tomando en cuenta los requerimientos e infraestructura de la empresa SALNET S.A.

4. El proyecto pretende el desarrollo de los elementos necesarios para el funcionamiento de un motor de búsqueda, sin considerar servicios adicionales como correo gratuito, servicios de comunicación IRC y otros.

5. Para el desarrollo del proyecto se utilizará un Robot Web ya existente con las modificaciones necesarias para su operación.

6. Para las etapas de desarrollo y prueba del buscador se empleará una red local con los elementos necesarios para verificar su funcionalidad, es decir una Intranet para posteriormente extenderse a Internet.

1.6 METODOLOGÍA DE DESARROLLO

El Proyecto denominado "Diseño, Desarrollo e Implementación de un Buscador de Sitios Web Nacionales", se realizó a través de dos etapas: La investigación y el desarrollo del mismo.

1.6.1. Investigación

1. Bibliografía: Se consultaron diversas fuentes escritas que además de ampliar conocimientos en cuanto a conceptos, permiten aclarar muchos procesos y ofrecen criterios para analizar y anticiparse a los requerimientos. Estos textos serán obtenidos de las Bibliotecas más importantes: Biblioteca Nacional, Biblioteca de la Universidad Centroamericana José Simeón Cañas y la Biblioteca de la Universidad Don Bosco.

2. Internet: Se tuvo acceso continuo a Internet, gracias al apoyo prestado por SALNET, para obtener una amplia gama de información sobre el World Wide Web, HTML (Hypertext Markup Languaje) y de temas relacionados. Además se hará uso de Correo Electrónico, mediante el cual se establecerán contactos con personas o entidades de experiencia en el área del proyecto.

3. Consultorías: Con el objetivo de reforzar algunos de los temas y conceptos se obtuvo información mediante entrevistas directas con personas que conocen algunos de los componentes involucrados en el desarrollo de este proyecto..

1.6.2. Desarrollo

Esta etapa se ha desarrollado en dos fases: la primera consistió en la implementación del motor de búsqueda en un laboratorio a nivel de Intranet, creando dominios de prueba en una red local, con el objetivo de evaluar el desempeño del robot en la examinación de los sitios publicados por los mismos; mientras que la segunda fase consistió en la prueba del robot ya implementado en el servidor de SALNET.

Para el desarrollo específico del proyecto se cuenta con el apoyo de la empresa SALNET S.A. (Ver Anexo A), brindando el acceso a cuentas de navegación para la fase de investigación y acceso controlado a su Servidor Web para las etapas avanzadas.

La Metodología empleada es el desarrollo por Prototipos. Este método de desarrollo permite que los usuarios participen en forma más directa en las fases de diseño y análisis del sistema a implementar.

Un prototipo es un sistema que funciona, sin embargo este no tiene todas sus características o lleva a cabo todas las funciones necesarias del sistema final. Más bien incluye los elementos suficientes para permitir a los usuarios utilizar el sistema propuesto, determinando cuales son los aspectos que les agradan y cuales no, identificando también aquellas características que deben cambiarse o añadirse. La evolución de los prototipos se realiza a través de un proceso de refinamiento en el que intervienen tanto usuarios como analistas.

Una razón principal para el desarrollo de prototipos es que permiten aumentar la productividad, ya que a menudo el desarrollo de sistemas de información se convierte en un proceso extenso, para descubrir al final que el sistema desarrollado no suple las necesidades de los usuarios. En cambio, en el desarrollo por prototipos se da la participación de los que emplearán el sistema, permitiendo definir cuales son sus necesidades específicas y qué es necesario cambiar, logrando con ello un nivel óptimo al cubrir los requerimientos.

Etapas del Método de Desarrollo por Prototipos.

1. Identificación de los requerimientos de información que el usuario conoce junto con las características necesarias del sistema.

   En esta fase del desarrollo de sistemas se da la determinación de los requerimientos que el usuario ha identificado y que desea se satisfagan. Para desarrollarlo se deben determinar los fines para los cuales será empleado el buscador de Sitios Web y el alcance de las capacidades del mismo.

   En este caso en particular los principales requerimientos ya han sido identificados en la fase de investigación preliminar del proyecto:

   a. Interfaz de fácil manejo para el usuario.
   b. Páginas con información de carácter dinámico.
   c. Organización de los resultados de las búsquedas de tal forma que sea práctica para el usuario.

2. Desarrollo de un modelo de trabajo.

En esta etapa del método, se desarrolla un modelo del buscador y de su interfaz de usuario que contará solo con algunas de las características que han sido definidas como esenciales, en la fase de determinación de requerimientos, con el objeto que estas puedan ser evaluadas posteriormente por el usuario, para el caso la empresa SALNET S.A.

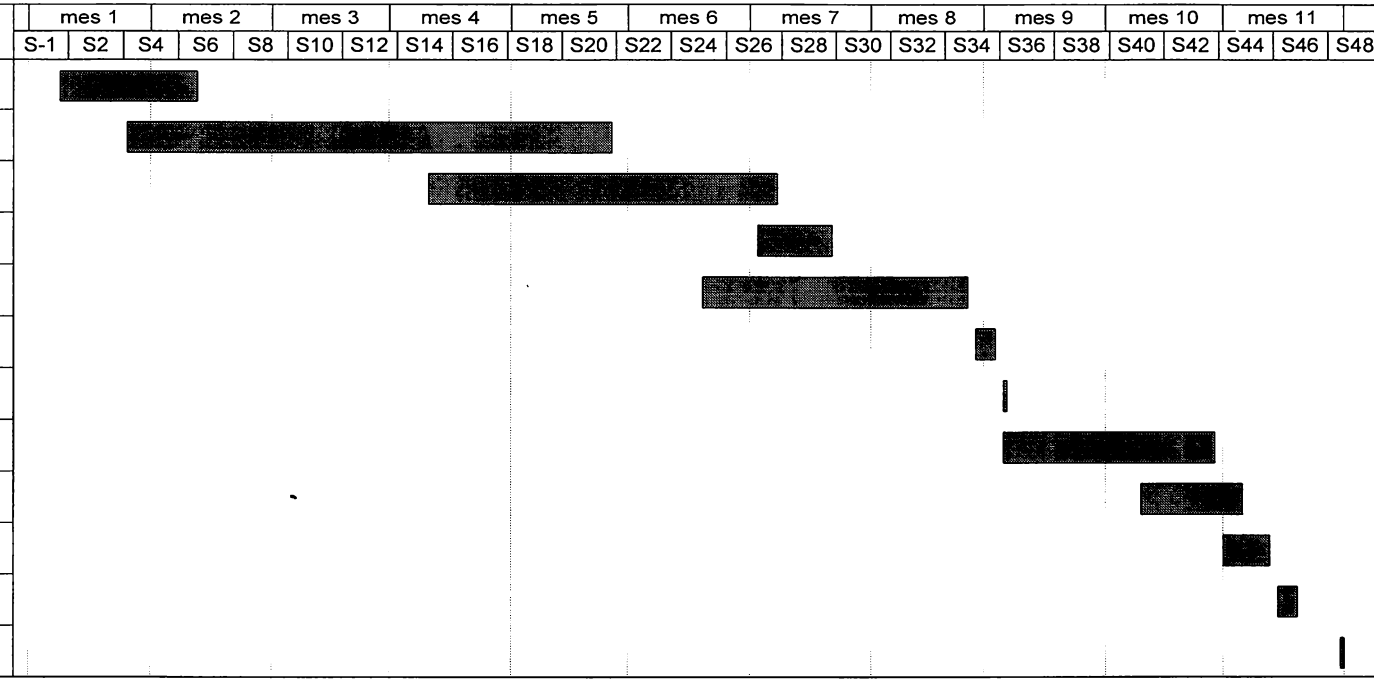
3. Revisión por parte de los usuarios del modelo desarrollado.
   Durante esta etapa de desarrollo los usuarios y el grupo de trabajo, deben de realizar una evaluación del Buscador. La experiencia con el mismo bajo condiciones reales, permite obtener la familiaridad indispensable para determinar los cambios o mejoras que sean necesarios así como la eliminación de características inadecuadas o innecesarias.

4. Revisión del prototipo
   Los cambios que se realizarán al prototipo se evalúan en forma conjunta con los usuarios, antes de llevarlos a cabo. El grupo de tesis será el responsable de realizar los mismos al prototipo.

5. Repetición del proceso cuantas veces sea necesario.
   Se repite el proceso descrito anteriormente, hasta lograr que los requerimientos del usuario sean satisfechos.

1.7 CRONOGRAMA DE ACTIVIDADES

Id	Nombre de tarea
1	Investigación Preliminar
2	Investigación de posibles herramientas
3	Aprendizaje de herramientas a usar
4	Diseño de la solución
5	Elaboración del primer avance
6	Presentación de documento para 1a. Defensa
7	Primera Defensa
8	Desarrollo de Solución
9	Pruebas y correcciones del buscador
10	Elaboración de documento final
11	Presentación de documento final
12	Defensa final



Proyecto: Proying
Fecha: 16/06/2000

Tarea		Resumen		Progreso resumido	
División		Tarea resumida		Tareas externas	
Progreso		División resumida		Resumen del proyecto	
Hito		Hito resumido			

CAPITULO II. MARCO TEORICO Y CONCEPTUAL

2.1 BREVE HISTORIA DE INTERNET

A mediados de la década de los sesenta, el departamento de defensa de los Estados Unidos ideó una red que entrelazaba sus equipos de cómputo con miras a optimizar su potencial. Como consecuencia apareció el ARPANET (Advanced Research Projects Administration Network), sistema militar que interconectaba computadoras entre sí con el objetivo de mantener una constante y segura vía de comunicación de información que les permitiera fortalecer sus fines militares. Posteriormente, se permitió el uso del sistema para actividades no militares. Institutos de investigación y universidades tenían la facilidad de contar con equipos capaces de comunicarse entre sí, dando paso al enriquecimiento de información para la ciencia, industria, comercio, educación, etc.

Pronto fue necesario actualizar los sistemas y protocolos utilizados para la interacción de diversas computadoras y se eligió el sistema TCP/IP (Transfer Control Protocol / Internet Protocol), familia de protocolos que se encargaría de establecer y regular las normas de comunicación en la Internet.

El fenómeno de las redes de cómputo se consolidó definitivamente en los primeros años de la década de los ochenta gracias a la aparición de otras redes, tanto públicas como privadas. En 1985 La Fundación Nacional para la Ciencia de Estados Unidos (NSFNet - National Science Fundation), interconectó cinco supercomputadoras. Para el año de 1988, se tenían alrededor de 28000 computadoras interconectadas en una red, formándose así Internet, permitiendo que en 1992 se generara la columna vertebral de la actual red, evolucionando día a día, creciendo el número de usuarios en el mundo y ofreciendo más y mejores servicios.

Internet es el concepto más general que identifica una red de comunicación, un fenómeno sociocultural cuyos beneficios han despertado el interés en muchas personas, por lo que los usuarios son cada día más. El impacto de Internet, como un nuevo y poderoso medio de comunicación, trae consigo influencias determinantes en ramos de la más diversa índole, ya que la multiplicidad nacional de Internet genera un movimiento promotor de la heterogeneidad de las culturas, en gran medida proveniente del simple hecho de que las personas en el mundo estén comunicadas entre sí.

2.2 CONCEPTOS BASICOS

2.2.1 Concepto de Internet

Internet es una red de redes, un sistema múltiple (capaz de manejar y administrar varias aplicaciones) y en gran medida abierto, que permite a los usuarios de diferentes redes y equipos interactuar entre sí.

En otras palabras, Internet es un gran conjunto de redes de ordenadores (servidores) interconectados. Es un sistema que, por la interactividad que ejercen sus usuarios y la libertad para el intercambio de información que poseen, supera ya toda la gama de temas y recursos que pueden encontrase en cualquiera de los medios de comunicación tradicionales; dando la oportunidad a los usuarios de obtener datos actualizados, lo cual representa una gran ventaja en la calidad de información que esta a disposición.

Internet se ajusta a casi cualquier tipo de servidor, tipo de red, tecnología de conexión y medios físicos empleados.

Internet no tiene una autoridad central, es descentralizada, cada red mantiene su independencia y se une cooperativamente al resto, respetando una serie de normas de interconexión reguladas por la familia de protocolos TCP/IP.

2.2.2 World Wide Web (WWW)

El World Wide Web es un sistema distribuidor de información basado en el concepto de hipertexto. Este fue desarrollado por un grupo de investigadores en el laboratorio europeo de física en partículas, ubicado en Suiza. Diseñado como una herramienta para facilitar la transmisión de documentos compuestos de texto, gráficos y sonidos. El World Wide Web posee elementos tales como: links, frames, bookmarks, tablas y botones, los cuales hacen que el ambiente del hipertexto sea amigable al usuario.

El hipertexto no es otra cosa que una frase, palabra o concepto resaltado en una página Web, sobre el cual se desea adquirir más información. Esta se obtiene dando un clic con el ratón (mouse) sobre la palabra, frase u objeto, apareciendo inmediatamente una segunda página con más datos. En las

hojas Web los enlaces de hipertexto permiten a un usuario seguir ideas y temas de página a página independientemente de si están almacenadas en una sola computadora (o en un servidor) o esparcidas en servidores en todo el mundo. Las hojas Web como se les conoce comúnmente tienen como estándar para el diseño y creación de éstas, el lenguaje HTML.

2.2.3 HTML

Son las siglas de *HyperText Markup Language,* Lenguaje marcador de hipertexto. Está basado en el SGML *(Standard Generalized Markup Language*, que significa, Lenguaje marcador estándar generalizado), mismo que se utiliza para delinear la estructura general de varios tipos de documentos. La atención del HTML se concentra en el contenido del documento, no en su apariencia.

Los archivos que utiliza como fuente son simples archivos de texto ASCII (American Standard Code for Information Interchange), de tal manera que para crearlos se utilizará cualquier editor de texto. Dichos archivos podrán funcionar adecuadamente en todos los sistemas computacionales.

Las herramientas que emplea el HTML son:

1. Para crear archivos fuente de HTML, se usa cualquier editor de texto, con las siguientes indicaciones:
   a. Colocar las extensiones ".html" o ".htm"
   b. Emplear un editor de texto simple: VI, Edit, Notepad, etc.
   c. Si se usa un procesador de palabras (MSWord, Wordperfect, Wordpad, etc.), guardar el archivo en formato "texto".

2. Para poder visualizar el archivo HTML, pueden utilizarse los Navegadores: Netscape, NCSA Mosaic, Lynx, MacWeb, Internet Explorer, etc.

HTML posee las siguientes características:

1. Los documentos en HTML son simples archivos de texto plano.

2. No es necesario incluir información referente al formato ni a las fuentes, ya que esto disminuiría la velocidad y aumentaría, en consecuencia, el tiempo para que el documento fuera cargado y desplegado en pantalla, este trabajo es realizado por el navegador.

3. Los documentos en HTML son independientes de los dispositivos. Esa es una manera elegante de decir que se despliegan en cualquier plataforma; todo lo que se necesita es un navegador para la plataforma en la que trabaje que sea capaz de interpretar el HTML.

2.2.4 Java

Concepto

Es un lenguaje de programación orientado a objetos, moldeado en base a C++. El lenguaje Java se diseñó para ser pequeño, sencillo y portátil a través de plataformas y sistemas operativos, tanto a nivel de código fuente como binario.

Java se escribió como un lenguaje de programación completo, donde es posible realizar las mismas tareas y solucionar el mismo tipo de problemas que con otro lenguaje de programación, como C o C++.

Historia de Java:

El lenguaje Java fue desarrollado por Sun Microsystems en 1991, una empresa reconocida por sus estaciones de trabajo UNIX de alta calidad, como parte de un proyecto de investigación para crear software para los dispositivos electrónicos (equipos de televisión, videocaseteras, tostadores y otros).En este tiempo los objetivos de Java eran programas pequeños, rápidos, eficientes y portátiles para un amplio rango de dispositivos de hardware. Estos mismos objetivos son los que hicieron a Java un lenguaje ideal para distribuir programas ejecutables por el World Wide Web, y también un lenguaje de programación de propósito general para desarrollar programas fáciles de utilizar que pudieran transportarse por diferentes plataformas.

Java independiente de la Plataforma:

La independencia de plataforma es una de las ventajas más representativas que tiene Java sobre otros lenguajes de programación, en particular para los sistemas que necesitan funcionar en varias plataformas.Java mantiene esta independencia de la plataforma tanto a nivel del código fuente como del binario.

A nivel de código fuente, los tipos primitivos de datos de Java tienen tamaños consistentes, en todas la plataformas de desarrollo. Los fundamentos de bibliotecas de Java facilitan la escritura del código, el cual puede desplazarse de plataforma a plataforma sin necesidad de volver a escribirlo para que funcione en cada una de ellas.

La independencia de la plataforma, sin embargo, no se detiene a nivel del código fuente. Los archivos binarios Java también son independientes de la plataforma, y pueden ejecutarse en múltiples plataformas sin necesidad de volver a compilar la fuente; la razón es que los archivos binarios Java se encuentran en una forma llamada bytecode.

El ambiente de desarrollo Java tiene dos partes: un compilador y un intérprete Java. El compilador Java toma su programa Java y en lugar de generar códigos de máquina para sus archivos fuente, genera un bytecode.

Para ejecutar un programa Java, debe ejecutar un programa llamado intérprete de bytecode, el cual a su vez ejecuta su programa Java (véase la figura).

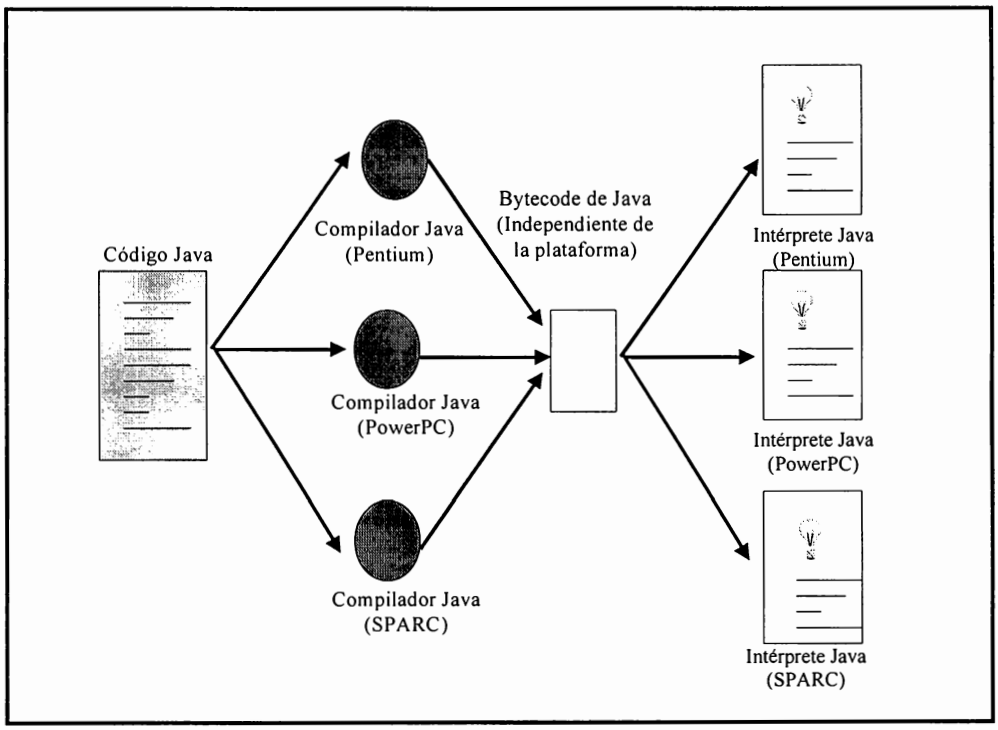


Ilustración 1. Compilación de un programa en Java

2.2.5 Perl

Breve historia de Perl y sus principales características:

Perl (Practical Extraction and Report Language, por sus siglas en inglés), es un lenguaje intérprete, desarrollado por Larry Wall.

Tom Christiansen y Nathan Tarkington (1997) establecen que este lenguaje es de alto nivel, derivado del lenguaje C de programación, así como de Sed, Awk y el Shell de Unix, además de otras herramientas y lenguajes.

Las facilidades para la manipulación de procesos, archivos, y texto hace que este lenguaje se encuentre particularmente bien situado en las tareas donde se involucra el rápido desarrollo de programas, desarrollo de utilerías para el sistema operativo, herramientas de software, tareas relacionadas con la administración de sistemas, manejo de base de datos, programación de gráficas, redes, y del Word Wide Web. Estas fortalezas hacen que Perl sea un lenguaje de programación muy popular para los administradores de sistemas UNIX y los creadores de *"CGI scripts"*. Aunque lo anterior, no es una limitación para que cualquier persona se involucre y use este lenguaje de programación.

Los programas de Perl son almacenados como archivos ASCII. Hay dos tipos de programas que Perl distribuye: Módulos y *Scripts*. Los Módulos de Perl (extensión .pm) contienen la definición de clases y rutinas en librerías. Los *Sripts* de Perl (extensión .pl ó cgi) contienen programas que hacen uso de las librerías y clases contenidas en los módulos de Perl.

2.2.6 Sitio Web

También se le llama Home Page, y es un archivo que ha sido escrito utilizando generalmente el lenguaje de programación HTML, y que está alojado en un servidor (computador) que es reconocido dentro de Internet. Dentro de este archivo es posible colocar instrucciones que hagan conexión con otros archivos o sitios, los cuales pueden ser creados por la misma persona, o por otras que ya los hayan colocado en el Web. El contenido de los archivos HTML en un Sitio Web pueden incluir texto, gráficos, fotografías, sonido, animación, vídeo, formas para llenar en línea, y muchas otras características.

En el Web se han difundido en gran medida las empresas interesadas en hacer negocios, pero también muchas con el objetivo primordial de manejar (acceder, distribuir, compartir) información y datos, que es en definitiva una parte crucial para lograr el éxito de cualquier tarea.

2.2.7 Navegadores (Browser)

Es un programa (como Netscape, Mosaic, Microsoft Internet Explorer) que permite ver la información en el WWW en un formato gráfico. El intérprete estándar usado para el navegador del Web es el HTML. Algunos navegadores también interpretan VRML (Virtual Reality Markup Language) y/o Java, el cual permite tener algunas características para la animación de páginas Web. Las características de un navegador son:

1. Pueden interactuar de diversas formas, tales que les permiten comunicarse con todos los servidores como Gopher, FTP-File Transfer Protocol y Web, actuando como clientes y utilizando los protocolos adecuados.

2. Emplean una interfaz gráfica con el usuario.

3. Permiten hacer referencias hacia información en hipertexto o hipermedia. De esta forma, cualquier palabra, frase o imagen puede funcionar como enlaces hacia cualquier otra información.

2.2.8 Sistema de Base de Datos

Es básicamente un sistema computarizado de almacenamiento en registros, utilizado con el propósito de darle mantenimiento a la información y hacerla disponible ante cualquier demanda.

Los componentes de un sistema de base de datos son:

1. Datos. Son alojados totalmente en una base de datos para brindar una mayor seguridad, además de ser compartidos e integrados con una mayor facilidad; se entiende por 'Compartir' el hecho de que distintos usuarios puedan hacer uso de manera simultánea de los datos, para distintos propósitos, y, por 'Integración' se entenderá la unificación de distintos archivos de datos que no posean redundancia.

2. Hardware. Son los volúmenes de almacenamiento en los cuales reside la base de datos, de una manera física, y también los dispositivos de entrada/salida.

3. Software. Se encuentra entre la base de datos física y el usuario; en definitiva, el software es el administrador de bases de datos (DB manager), más usualmente conocido como Sistema administrador de bases de datos (DBMS).

4. Usuarios. Pueden ser programadores de aplicaciones, usuarios finales o administradores de bases de datos, que de una u otra forma se mantienen en contacto con la base de datos, realizando cualquier operación relacionada con la información que ésta contenga.

2.2.9 DBMS (Database Management System)

Es el software que dirige cualquier acceso a la base de datos. Conceptualmente, lo que sucede se describe en los pasos siguientes:

1. Un usuario realiza una requisición de acceso, utilizando algún lenguaje particular de datos (SQL - Structured Query Language).

2. El DBMS intercepta la requisición y la analiza.

3. El DBMS inspecciona conceptos tanto a un nivel interno como externo, de la base de datos y de la requisición del usuario, como: mapeados, esquemas externos e internos y la definición de la estructura de almacenamiento.

4. El DBMS ejecuta las operaciones necesarias en la base de datos.

En general, el DBMS provee al usuario de una interfaz con el sistema de base de datos. Gracias a la utilización del HTML para construir páginas Web, se tiene la posibilidad de acceder a una gran cantidad de sitios, de diversos tipos y contenidos, ya existentes en Internet. Dichos sitios pueden desplegar páginas Web con contenido de texto o con formatos específicos que permitan interactuar al usuario con datos alojados en listas o bases de datos. Esta interacción no puede ser directa entre el navegador del cliente y una base de datos, sino que necesita de la utilización de una interfaz para tener acceso a la información disponible, y realizar cualquier tipo de búsqueda.

2.2.10 Integración de Bases de Datos en el Web

La facilidad de comunicarse con personas o instituciones en cualquier parte del mundo permite tener acceso a la más variada gama de información, situación que aumenta las capacidades de desarrollo de las empresas, organizaciones y usuarios de Internet.

La mayor parte de información que esta a disposición en la red, se encuentra almacenada estáticamente en hojas Web; sin embargo, también existe información en bases de datos con contenidos y formatos muy diversos, la cual es accedida mediante interfaces, protocolos y controladores para ser desplegada finalmente en navegadores mediante hojas Web, superando así la opción de encontrar información estática en los documentos HTML.

En este sentido el sistema operativo que se utiliza resulta transparente para el Web, siendo esta una notable ventaja. Esto permite establecer conexión entre plataformas distintas para el cliente (navegador) y el servidor (servidor Web), sin necesidad de cambiar el formato o estructura de la información dentro de las bases de datos.

Empleando bases de datos en el Web, este se convierte en un medio capaz de localizar, enviar y recibir información de diversos tipos, optimizando así el acceso a la misma y cumpliendo el propósito principal de Internet, el cual consiste en el compartir información.

En el pasado, las bases de datos quedaban limitadas a una utilización exclusivamente al interior de las empresas, por medio de redes locales; ahora, gracias al Web es posible acceder a bases de datos de cualquier parte del mundo, ofreciendo a través de la red, un manejo dinámico, facilidad de actualización y una gran flexibilidad de los datos, como ventajas que no podrían obtenerse a través de otro medio informativo.

Con estos propósitos, los usuarios de Internet pueden obtener un medio que puede adecuarse a sus necesidades de información, con un costo, inversión de tiempo y recursos mínimos. De igual forma, las bases de datos pueden ser empleadas para permitir el acceso y manejo de la variada información que se encuentra a lo largo de la red.

2.2.11 SQL (Structured Query Language)

Es un sub-lenguaje estándar para acceder y manipular la información que se encuentra dentro de los Sistemas de bases de datos relacionales.La mayoría de dichos sistemas que se usan en la actualidad, tal como Oracle o Sybase, soportan sentencias SQL.

Con SQL se pueden realizar ciertas acciones sobre la información, tales como:

1. Creación de tablas
2. Acceder a la información dentro de la base de datos, para realizar consultas especificas.
3. Actualizar la información dentro de la base de datos.
4. Insertar nueva información a la base de datos.
5. Borrar información específica de la base de datos.
6. Modificar la definición de los datos en la base.

En resumen, el código SQL que manipula y accede a la base de datos es portable, rápido y fácilmente de trasladar a cualquier plataforma en que se desee trabajar.

2.2.12 ODBC (Open Database Conector)

ODBC es un conjunto estándar de rutinas que permiten que una aplicación acceda sistemas de gestión de bases de datos relacionales y no relacionales. Este controlador está basado en el lenguaje de consulta SQL y una característica es que su implementación es transparente al usuario.

La función principal de ODBC es independizar el gestor de bases de datos, generando informes basados en consultas y aplicaciones en general para los distintos tipos de datos (Bases de datos) existentes en el mercado como: Oracle, Sybase, SQL Server, Microsoft Access, FoxPro, etc. Es decir, ODBC separa el formato estricto de una base de datos del programa gestor de bases de datos, de manera que todos los programas compatibles con ODBC puedan acceder a los formatos soportados por ODBC.

2.2.13 JDBC

JDBC es un API de Java para la ejecución de sentencias SQL. Este consiste en un conjunto de clases e interfaces escritas en lenguaje de programación Java, lo que posibilita escribir aplicaciones con acceso a bases de datos, empleando para ello únicamente código Java.

JDBC es una interfaz de bajo nivel, lo cual se refiere a que puede ser utilizado para invocar comandos de SQL directamente; además, combina muy bien su capacidad y su sencillez de empleo, para ser una base con la cual construir herramientas e interfaces de alto nivel o amigables al usuario.

Usando JDBC es sencillo enviar sentencias SQL virtualmente a cualquier base de datos relacional. En otras palabras, con JDBC no es necesario escribir un programa para acceder a una base de Sybase, otro para una base de Oracle y otro específico para cada tipo diferente de base de datos. Es posible escribir un solo programa utilizando JDBC y el programa será capaz de enviar la sentencia SQL a la base de datos apropiada, siempre y cuando se respete la estructura de los datos. Combinando las virtudes antes mencionadas de una aplicación escrita en Java, no será necesario preocuparse de escribir diferentes aplicaciones para correr en diferentes plataformas. Con lo cual la aplicación permanecerá funcional sin importar que exista cambio en la plataforma de base de datos.

En forma general los pasos necesarios para acceder a una base de datos empleando JDBC son:

1. Establecer una conexión con el controlador apropiado de la base de datos, definiendo una variable para hacer referencia a esta conexión.
2. Definir una variable de sentencia que almacenará la instrucción SQL a ejecutar.
3. Definir una variable de tipo ResultSet o "Conjunto de Resultado", que almacenará los datos que devuelva la sentencia SQL en caso que así sea
4. Ejecutar la sentencia SQL.
5. Manipulación de los resultados.

2.2.14 Interfaz

Define un conjunto de reglas o normas entre los procesos cliente y servidor, de tal forma que puedan comunicarse entre sí en un nivel más alto que el envío y recepción de simples cadenas de bytes, en un ambiente heterogéneo de interconexión.

2.2.15 IDC (Internet Database Connector).

La interfaz IDC es una biblioteca que permite publicar información almacenada en una base de datos en el Web usando el controlador ODBC respectivo.

Conceptualmente, el acceso a las bases de datos se realiza a través del Servidor Web: Internet Information Server, como se muestra en la siguiente figura :

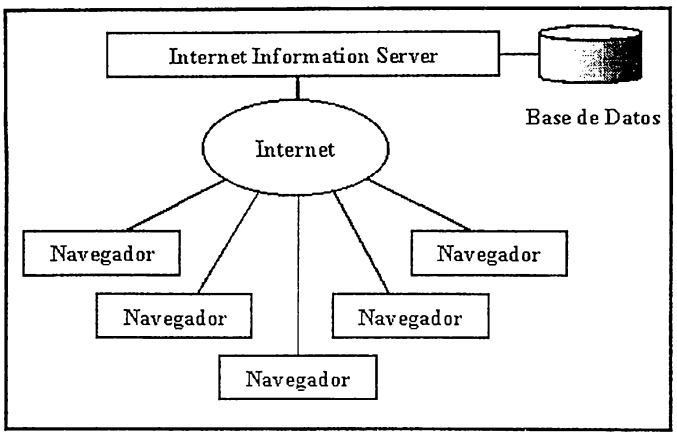


Ilustración 2. Funcionamiento Conceptual del Internet Database Conector

Funcionamiento del IDC

Para realizar una requisición de acceso desde el Web hasta una base de datos no solo se necesita un navegador Web y un servidor Web, sino también de un software de procesamiento, que en este caso puede ser IDC/HTX, técnica utilizada para establecer el acceso a la base de datos mediante controladores e interfaces, de esta manera:

1. El usuario referencia la dirección de un archivo IDC en el explorador. La extensión del archivo IDC se relaciona con la biblioteca de conectores de base de datos especial: HTTPODBC.DLL que se carga a continuación.

2. La biblioteca lee el archivo IDC y carga el controlador de Conectividad Abierta de Bases de Datos correcto para el origen de los datos.

3. El controlador conecta con el origen de datos, después de lo cual la biblioteca pasa la instrucción de la consulta de origen de datos, en este caso a una base de datos de Access.

4. Access abre la base de datos, ejecuta la consulta en el contenido del archivo IDC y envía los resultados a la biblioteca mediante el controlador ODBC.

5. La biblioteca recurre de nuevo al archivo IDC en busca de información sobre el nombre del archivo HTX en el que se colocarán los resultados, de acuerdo al formato definido en su interior.

6. La biblioteca añade los resultados de la consulta al archivo HTX en donde están los marcadores de posición, y envía el archivo HTML, ahora completo, al servidor Web mediante la biblioteca del conector.

7. El servidor transfiere el archivo al explorador, con lo cual se termina la consulta iniciada por el archivo IDC original.

Los navegadores Web (tales como el Internet Explorer y el Netscape) registran sus solicitudes al servidor Web utilizando el HTTP (Hypertext Transfer Protocol). El servidor Web responde con un documento diseñado en HTML.

El IDC se utiliza en conjunto con un archivo de extensión HTX, que no es abreviatura de nada; tan solo es un indicador de que se trata de un archivo de plantilla especial en la familia de archivos HTML.

Los archivos IDC y HTX son de texto sencillo. El archivo IDC contiene, en lo fundamental, tres datos:

1. El origen de los datos (nombre del conector hacia la base de datos).

2. El nombre de la plantilla o archivo HTX en donde aparecerán los datos en el explorador.

3. Una consulta en el formato de lenguaje explorador de consulta estructurado (SQL) correcto, que pueda interpretar el origen de los datos.

4. Opcionalmente puede contener una contraseña y nombre de usuario, si es necesaria la autorización para conectarse a la base de datos.

El Archivo HTX es un archivo HTML con etiquetas y marcadores de posición, con que se define la estructura del archivo HTML.

La siguiente ilustración muestra los componentes para la conexión a la base de datos desde el Internet Information Server :

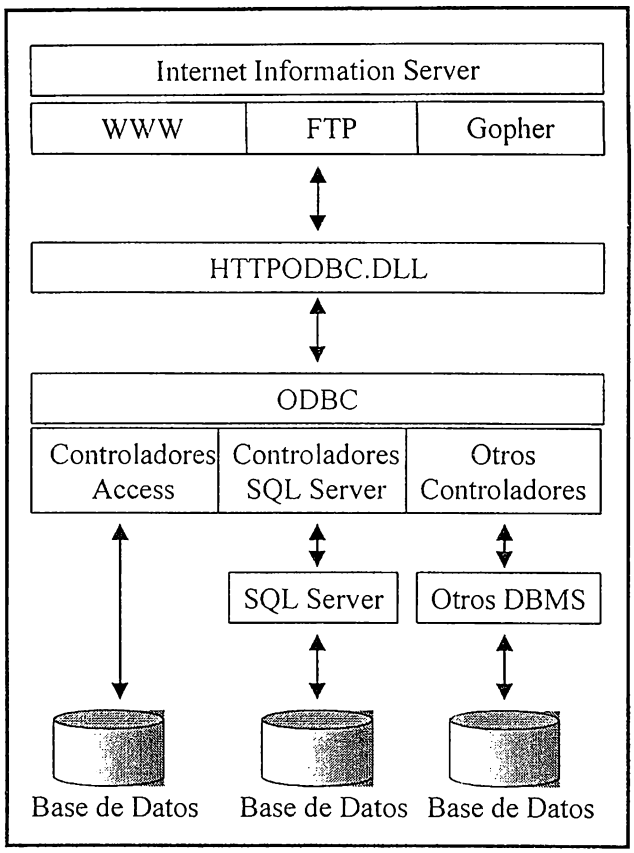


Ilustración 3. Conexión a una Base de Datos empleando IDC.

2.2.16 ASP (Active Server Page)

Es una interfaz que permite no sólo el desarrollo de páginas Web dinámicas sino también el acceder a bases de datos. Esta incluye uno o más *Scripts* (Pequeñas rutinas de programa). Los *Scripts* pueden hacer referencia a componentes que se encuentren en el servidor Web local ó cualquier otro servidor para acceder bases de datos, aplicaciones, o procesar información. Un *script* comienza a ejecutarse en el momento en que un navegador solicita un archivo ".asp" del servidor. El servidor llama entonces al ASP, el cual lee todo el archivo solicitado de inicio a fin, ejecutando cualquier comando y enviando un archivo HTML al navegador. Los *Scripts* incrustados en el archivo ASP pueden procesarse tanto en el cliente como en el servidor.

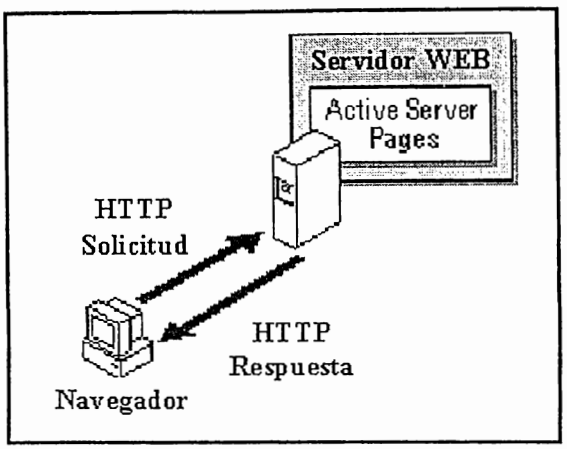


Ilustración 4. Diagrama Esquemático del Active Server Pages

Cuando los *scripts* corren en el servidor en lugar de hacerlo en las máquinas clientes, es éste el que realiza todo el trabajo que involucra el generar las páginas en formato HTML y enviarlas a los navegadores, no es necesario preocuparse de que el navegador pueda procesar las páginas, el servidor realiza todo el proceso.

Composición de un archivo .asp

Los archivos de Active Server Pages poseen extensiones ".asp", y son archivos de texto que pueden contener alguna combinación de los elementos siguientes: texto, etiquetas de HTML o comandos *script*.

Para hacer que un archivo de *script* este disponible para los usuarios del Web, es necesario guardar el archivo en un directorio de publicación Web, asegurándose que el directorio virtual asociado a este tenga habilitado el permiso de ejecución.

Al ejecutar un *script*, la serie de comandos que lo componen se envían al mecanismo de decodificación, el cual las interpreta para el servidor. Los *scripts* son escritos en lenguajes que tienen reglas específicas, es así como si se desea emplear un lenguaje de programación, el servidor deberá de poder correr el mecanismo que interprete dicha codificación. ASP provee de manera integral los mecanismos de decodificación de *Visual Basic Script (VBScript)* y *Java Script (JScript)*; el lenguaje de codificación primario, que por defecto ASP asume que se está empleando, es *VBScript*.

Funcionamiento de ASP

Para realizar el acceso a una base de datos desde el Web, empleando Active Server Pages, es necesario que el servidor Web sea capaz de interpretar los segmentos de código dentro del archivo con extensión ".asp", para lo cual debe de tener instalado el interprete apropiado, así como los controladores apropiados para el tipo de base de datos a emplear. Los pasos necesarios para realizar la conexión a la base de datos desde el Web, son los siguientes:

1. El cliente de Internet realiza la solicitud de un archivo con extensión ".asp", al servidor Web.

2. El Internet Information Server reconoce que es un archivo ASP el que se le solicita y transfiere el control a la interfaz del Active Server Pages.

3. El Active Server Pages, realiza una lectura secuencial del archivo solicitado, ejecutando los segmentos de código que va encontrando con el interprete adecuado según se especifica.

4. Mientras ejecuta la página, al mismo tiempo, la interfaz construye en memoria una página Web, para devolverla al cliente de Internet que solicitó el archivo.

5. Al requerirse el acceso a una base de datos, se realiza la carga del controlador apropiado de conexión, por ejemplo el controlador de Microsoft Access.

6. El controlador abre la base de datos y ejecuta la sentencia SQL especificada, devolviendo los resultados de la consulta a la interfaz, para que estos sean manipulados de acuerdo a las instrucciones del archivo ".asp", si es que se retornan registros.

7. Una vez concluida la ejecución del archivo se envía la página HTML resultante al cliente, como resultado de su solicitud.

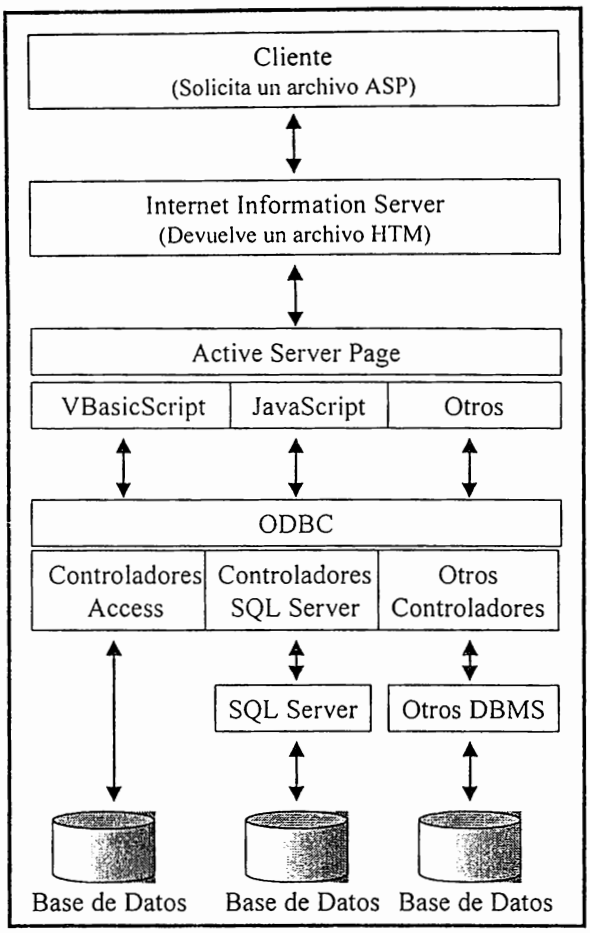


Ilustración 5. Conexión a una Base de Datos empleando ASP

2.2.17 Protocolo

Definición: Es un conjunto estricto de reglas o procedimientos que se requieren para iniciar y mantener las comunicaciones.

Los protocolos de comunicación de datos son los que hacen posible el intercambio de información, después de establecer una llamada a través de un canal informativo.

El sistema de protocolos que fue desarrollado como producto de las primeras investigaciones realizadas por el Departamento de Defensa de los Estados Unidos, llegó a conocerse como TCP/IP, después de que los dos protocolos iniciales fueron desarrollados: el protocolo de Control de Transmisión (TCP) y el Protocolo Internet (IP).

2.2.18 TCP/IP

Es un conjunto de protocolos desarrollado para permitir a las computadoras compartir recursos a través de la red, de tal manera que establecen una comunicación constante entre ellas. Por ello, éste es un conjunto básico para la comunicación y transmisión de datos en Internet.

TCP (Transmission Control Protocol)

Es el responsable de verificar el correcto manejo y movilización de la información, desde el cliente hasta el servidor o viceversa, ya que los datos pueden ser perdidos en el camino. Además, detecta errores y se encarga de una retransmisión hasta que la información sea recibida completa y correctamente.

IP (Internet Protocol)

Es el responsable de mover paquetes de datos desde un nodo a otro, guiándose cada paquete mediante el uso de una dirección de 4 bytes (dirección IP), que funciona en las máquinas conectadas a la red. Los servicios más importantes prestados por TCP/IP son:

1. Transferencia de archivos (FTP - File transfer protocol), permitiendo al usuario que desde cualquier computadora pueda obtener archivos que están en otra computadora o, por el contrario, enviar archivos desde la suya hasta a otra máquina.

2. Login remoto (TELNET - Network terminal protocol), que permite al usuario entrar en sesión remota con cualquier otra computadora en la red.

3. Correo Electrónico, que permite enviar mensajes a usuarios en otras computadoras.

4. HTTP, que permite acceder al World Wide Web.

2.2.19 HTTP (Hypertext Transfer Protocol)

Ha estado en uso en el World Wide Web desde 1990, presentándose como un protocolo genérico orientado a objetos, que puede ser usado para distintas tareas tales como servidores de aplicaciones y sistemas de control de distribución de información, a través de sus listas de extensión. Una

característica de HTTP es que permite al sistema cargarse independientemente de que los datos se estén transfiriendo.

El propósito del HTTP es que los sistemas de información sean más funcionales que simplemente dar una respuesta a un requerimiento hecho por el usuario, sino que también deberán incluir búsquedas, anotaciones y actualizaciones continuas.

En el Internet, la comunicación se lleva a cabo gracias a la conexión que realiza el TCP/IP, pero esto no le debe permitir dirigir a cualquier otro protocolo en el Internet o en otras redes, de tal manera que la estructura de búsqueda establecida por HTTP para analizar y responder una solicitud, y luego transportar las unidades de datos, no puede ser dominada por el TCP/IP.

El HTTP es básicamente estable, y la transmisión que realiza se divide en los pasos siguientes:

1. Conexión: que es establecida desde el cliente hacia el servidor.

2. Solicitud: que es enviada por el cliente y consiste en un mensaje de solicitud al servidor.

3. Respuesta: enviada por el servidor hacia el cliente, y es una respuesta a la solicitud de éste.

4. Cierre: es el cierre o finalización de la conexión, tanto por parte del cliente como del servidor.

El formato de las partes de requisición y respuesta es definido por el HTTP, mientras que la información de cabecera definida en esta especificación es enviada en caracteres latinos ISO(International Standards Organization), y la transmisión de objetos es realizada, si es posible, en forma binaria.

2.2.20 Servidores Web

También son llamados Servidores HTTP, debido a que el protocolo que usa para comunicarse con el navegador es el Protocolo de transferencia de hipertexto. Estos servidores interactúan con los tipos de datos que las personas más usan: de hipertexto y de multimedia.

De manera similar en que se establecen conexiones entre computadoras clientes y servidores a través de la red mundial de Internet, se pueden conectar otras con funciones muy parecidas, pero a nivel interno en corporaciones específicas, para garantizar la obtención de beneficios particulares para la empresa.

2.2.21 Cliente / Servidor

Es una arquitectura computacional que involucra procesos de clientes que se encuentran requiriendo servicios de procesos de servidor.

Cliente / Servidor es el concepto computacional que viene a ser la extensión lógica de la programación modular, la cual asume fundamentalmente la separación de grandes piezas de software, en partes más pequeñas llamadas "módulos", creando la posibilidad de obtener un desarrollo más fácil y darle un mejor mantenimiento. El proceso Cliente / Servidor reconoce que estos módulos no necesitan ser ejecutados dentro del mismo espacio de memoria, de tal manera que al utilizar esta arquitectura, el módulo que realiza la llamada se convierte en el "cliente" (que es quien hace la requisición de un servicio), y el módulo que es llamado se convierte en el "servidor" (que es el que provee el servicio).

Para aplicar dicho concepto, el siguiente paso será tener a clientes y servidores corriendo en el hardware, y bajo el software de la plataforma apropiados para realizar sus funciones. Por ejemplo, servidores de manejo de sistemas de bases de datos, ejecutándose en plataformas especialmente diseñadas y configuradas para manejar requisiciones en forma de pregunta, o archivos de servidores corriendo en plataformas con elementos especiales para manejo de archivos.

Proceso Cliente

El cliente es un proceso (programa) que envía un mensaje a un proceso (programa) servidor, requiriéndole a éste la realización de una tarea (servicio). El programa cliente usualmente maneja la parte de la aplicación que hace interfaz con el usuario, validando los datos introducidos por éste, enviando las requisiciones al programa servidor, y a veces ejecutando lógicamente las tareas.

Proceso Servidor

Un proceso (programa) servidor satisface las requisiciones del cliente realizando la tarea solicitada. El programa servidor general recibe las solicitudes desde el programa cliente, ejecuta las extracciones de información de las bases de datos, las actualiza, manejando la integridad de los datos, y envía respuestas a las interrogantes del cliente.

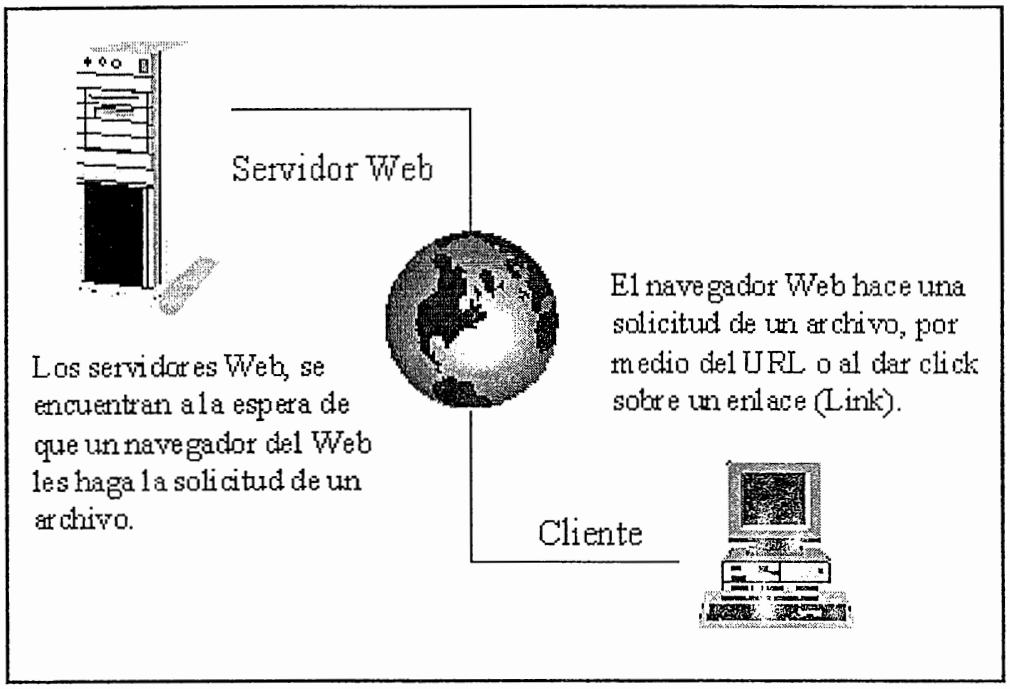


Ilustración 6. Diagrama Cliente / Servidor

2.2.22 Intranet

La aparición y consiguiente explosión del World Wide Web es gracias a la aceptación mundial de un modo de transporte común (TCP/IP), a la estandarización de servidores (HTTP), y a un lenguaje de marcado (HTML).

Muchas compañías han descubierto que estas mismas tecnologías pueden usarse para aplicaciones a nivel interno de Cliente/Servidor, con la misma facilidad con que son usadas en el Internet. A partir de ello, nació el concepto de "Intranet", el cual es el uso de tecnologías de Internet para la implantación de aplicaciones a nivel interno del concepto Cliente/Servidor.

Una de las ventajas de Intranet, basado en servidores Web, es la gran reducción del problema de manejo de código en el cliente. Si se asume que se tendrá un navegador estándar en el escritorio de trabajo, todos los cambios en la interfaz con el usuario pueden ser hechos cambiando el código en el servidor HTTP. Esto es mucho más fácil que si se actualizara el código en cada una de las estaciones de trabajo de los usuarios.

Una segunda ventaja es que si la compañía ya está haciendo uso de Internet, no se necesita de la instalación de código adicional en las estaciones de usuarios, de tal manera que para ellos, la información de servidores internos y externos, aparece integrada.

Una desventaja, que está desapareciendo rápidamente, es la poca habilidad de proveer el envío de código al cliente. Hace algún tiempo, el Web tenía pocas vías de interacción con el cliente, ya que era esencialmente sólo de lectura, pero con la aparición de herramientas para la creación de código, tales como *Java, JavaScript*[1], *Programación CGI (Common Gateway Interface)*[2], etc., esta limitación ya no está dando tanto problema.

Para la realización de todas estas interconexiones, se necesita, además, hacer uso de protocolos específicos que ayuden a realizar distintas tareas, relacionadas con la comunicación e intercambio de información.

2.2.23 URL (Uniform Resource Locator)

Es una dirección empleada para especificar un objeto en Internet. Puede ser un fichero, una página, un grupo de noticias, una imagen, etc.
Algunos ejemplos:

1. file://www.uco.es/www-docs/HTMLprimer.txt
2. Http://www.cica.es/
3. telnet://lucano.uco.es
4. news://alt.cad.autocad

---

[1] *JavaScript:* Leguaje simple de programación utilizado para pequeñas tareas dentro de las páginas web. Estos scripts normalmente corren en la computadora cliente, en el contexto del navegador.
[2] *CGI:* Es una interfaz estándar que permite que programas externos puedan correr bajo un servidor de información, actualmente los servidores soportados son HTTP.

Una dirección de un sitio en Internet comienza normalmente con un nombre de protocolo, el cual especifica el método de acceso, seguido de la organización que mantiene el sitio; el sufijo identifica el tipo de organización. Por ejemplo, "HTTP: //WWW.YALE.EDU" identifica al servidor Web de la Universidad de Yale. "HTTP: //WWW" indica que es un servidor Web que utiliza el protocolo HTTP y ".EDU" identifica a Yale como una institución educativa. Generalmente, las direcciones de los sitios comerciales terminan con ".COM" y las direcciones de los sitios del gobierno terminan con ".GOV".

Si la dirección apunta a una página específica, se incluye información adicional como el nombre del puerto, el directorio donde se encuentra la página y el nombre del archivo de la página. Las páginas Web que se han creado utilizando HTML (Lenguaje de marcas de hipertexto) terminan a menudo con una extensión ".htm" o ".html".

2.3 MOTORES DE BUSQUEDA

2.3.1 ¿Por qué surgen los Motores de Búsqueda?

Día a día el número de proveedores de información y servidores Web en Internet, aumentan a gran velocidad, lo que hace muy difícil el buscar información relacionada a un tema específico, para solucionar este problema, han surgido diversas herramientas, genéricamente llamadas Motores de Búsqueda. Un motor de búsqueda es un servicio que permite a los usuarios de Internet buscar información sobre un tema en particular en una base de datos, obteniendo como resultado URLs relacionados a los criterios específicos de la búsqueda. Cabe destacar que esta base de datos ha sido, por lo general, previamente alimentada por un Robot Web.

2.3.2 Componentes de los Motores de Búsqueda

Los componentes que forman a un motor de búsqueda son:

1. Un servidor Web, capaz de atender una gran cantidad de peticiones simultáneas, de llamar a las aplicaciones de búsqueda que sean necesarias y generar resultados en forma eficiente y poseer un sistema de seguridad debido a la gran cantidad de usuarios que lo visitan.

2. Un medio de almacenamiento, que permita guardar en forma local la información contenida en las páginas Web indexadas, manteniendo la integridad y seguridad de los

datos; y posibilitando a la vez la consulta posterior de esta información de acuerdo a los requerimientos de los usuarios.

3. Una interfaz de comunicación entre el medio de almacenamiento y el Servidor Web, la cual debe ser rápida y producir resultados confiables, capaz de obtener información sin importar la plataforma o sistema operativo del cliente, y además que no requiera alterar el formato de almacenamiento de la información de la base de datos.

4. Aplicación de alimentación automática o robot que sea capaz de ejecutarse periódicamente en el Web para extraer URLs y agregarlos a la base de datos, y que sea capaz de verificar los ya existentes.

El siguiente diagrama ejemplifica la integración de los componentes que conforman un motor de búsqueda:

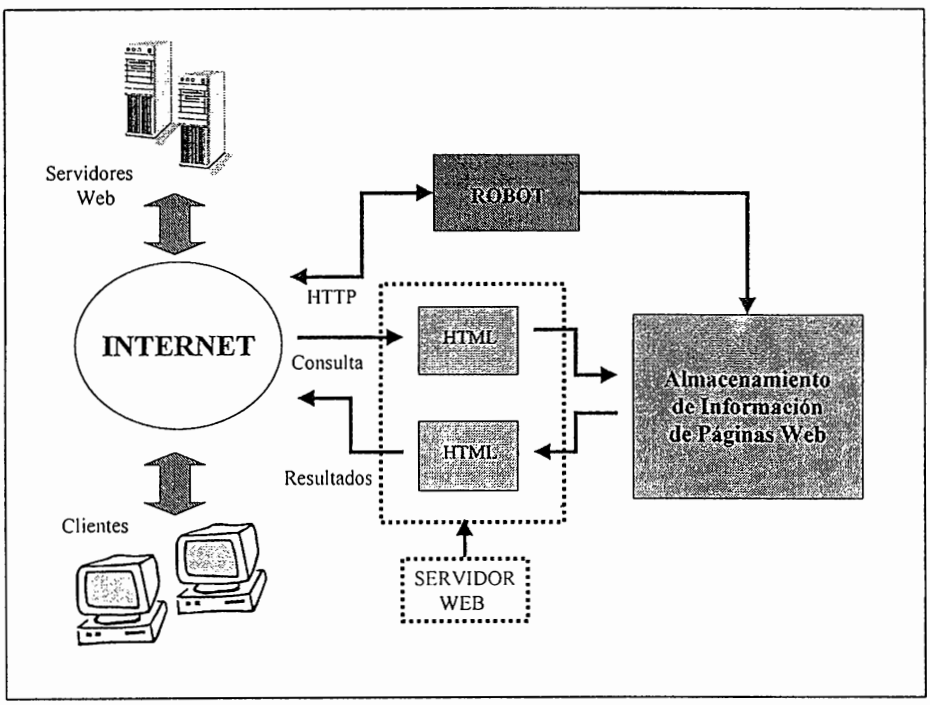


Ilustración 7. Componentes de un Motor de Búsqueda Genérico.

2.3.3 Clasificación de los Motores de Búsqueda

Los motores de búsqueda se pueden clasificar en dos tipos:

1. **Web Robots:** Están basados en el concepto de una base de datos única y centralizada, donde toda la información es almacenada. Esta información se recoge mediante robots o mediante el envío directo de URLs por parte de los administradores Web. Una vez los URLs han sido recogidos, el sistema construye un índice para permitir posteriores preguntas a partir de palabras claves.

2. **Directorios específicos:** Estos sistemas ofrecen un conjunto de páginas con referencia a servidores Web relacionados con un cierto tema. Este servicio agrupa las referencias o URLs en habitaciones temáticas de acuerdo con el tipo de información que cada URL ofrece y agrupados según los criterios del autor del índice.

2.3.4 Robots Web

Los Robots Web aparecieron en el World Wide Web alrededor del año de 1994 aproximadamente, estos surgieron cuando el tamaño de Internet creció más allá de unos cuantos sitios, así como el número de páginas por sitio, planteándose la necesidad de desarrollar nuevos métodos para encontrar información en el menor tiempo posible y de manera eficaz, pues la navegación por una porción significativa del Web se convirtió en una tarea que se dificultaba en gran medida. Desde su surgimiento estos han recibido varios términos para nombrarlos, como lo son: Spiders, Worms, Crawlers o Ants.

Los robots son algoritmos automáticos, que utilizando el HTTP como protocolo de comunicación, recorren periódicamente todo o parte del Web. Estos parten de una lista inicial de direcciones conocidas, y a partir de estas descubren recursivamente otras páginas Web, sin interacción del usuario final. Dicha información puede ser utilizada para alimentar motores de búsqueda, para efectos estadísticos, para realizar copias de respaldo, etc.

Los principales tipos de robots son los siguientes:

1. **Robot de Análisis Estadístico:** Este tipo de robot extrae información y la utiliza para realizar cálculo de tipo estadístico, tales como controlar el número de servidores Web

existentes en un dominio específico, el número promedio de documentos existentes en los servidores de un dominio o región y el tamaño de los documentos existentes en los servidores de un dominio. El primer robot que fué desarrollado pertenecía a esta categoría y su propósito era el de contar el número de Servidores Web existentes hasta ese momento.

2. **Robot de Mantenimiento:** Este tipo de robot se ocupa de verificar la validez de un URL, en caso que no lo sea, informa a la base de datos el estado del recurso y especifica si es posible accederlo en una ubicación diferente.

3. **Robot de Copia a Espejo:** Son utilizados para mantener estructuras de hipertexto redundantes con el objetivo de proveer de respuesta rápida y segura a los fallos en los servidores Web. También se emplean para distribuir la carga de servidores de tal manera que usuarios en zonas remotas puedan acceder a servidores remotos que contienen la misma información recientemente actualizada. Este tipo de distribución de recursos es normal en sitios FTP, debido a que se tienen limitaciones en el número de conexiones permitidas por un sitio que maneja este protocolo.

4. **Robot de Búsqueda de Recursos:** Se caracterizan por explorar y resumir grandes porciones del Web. Además los URLs extraídos son colocados en bases de datos indexados para luego ponerlos a disposición de usuarios en el Web a través de motores de búsqueda. Una de las tareas de importancia que realizan estos robots es la actualización periódica de la base de datos del buscador.

5. **Robot Combinado:** Combinan ciertas características de los robots arriba detallados para fines específicos, por ejemplo: comúnmente se implantan los robots estadísticos combinados con robots de mantenimiento, y los robots estadísticos junto a los de copia a espejo.

Básicamente los robots trabajan bajo el mismo principio de un navegador, haciendo solicitudes de páginas Web a servidores específicos, sin embargo este se caracteriza por no necesitar de un usuario que salte de una página a otra, esta es una tarea que el robot desarrolla en forma automática, para ello el robot baja una de las páginas ubicada en su lista inicial, la recorre en busca de enlaces hacia otras, para luego saltar hacia uno de los URL's encontrados y comenzar el proceso nuevamente; una

vez el robot llega a una página sin saltos, este regresa uno o dos niveles y salta hacia uno de los URL's que no había considerado anteriormente.

Una vez activado un robot web este puede llegar a recorrer una gran cantidad de páginas en poco tiempo, gracias a que para desarrollar su labor no necesita bajar los gráficos que conforman la hoja que está analizando, solamente toma en consideración el texto de la misma.

Cuando el robot ya ha obtenido la página solicitada, este analiza la información de la misma para trabajarla según su propósito particular; es así como en el caso del robot de ordenamiento de recursos además de identificar los enlaces, debe decidir que información introducirá a la base de datos, lo cual depende en gran medida del robot mismo, algunos almacenan en la base solamente un número definido de palabras, párrafos o títulos. Para introducir esta información a la base de datos emplean un controlador ODBC, que le permite por medio de instrucciones SQL agregar o modificar la información existente en la base de datos.

Es necesario aclarar que un robot bien diseñado y operado es capaz de brindar un valioso servicio a los usuarios del Web, lo que lo convierte en un elemento indispensable para aquellos usuarios que requieran encontrar información específica, sin embargo un robot requiere de mucho cuidado al momento de su desarrollo e implementación, pues de no ser así puede llegar a causar sobrecarga ya sea en el servidor que visita como en donde habita.

La Exclusión de los Robots

La mayoría de los servidores de páginas Web no implementan sistemas para restringir el acceso de robots. El gran número de peticiones que los primeros robots recuperadores de páginas Web lanzaban a los servidores de páginas, no sólo agotaban el ancho de banda de conexión del servidor, sino también agotaban la paciencia de sus administradores, llegando a ser frecuente el caso de administradores que tenían que restringir manualmente el acceso desde un determinado host realizando determinados *scripts* o incluso prohibir las solicitudes y el acceso al servidor desde las máquinas donde se detectaban dichas solicitudes.

Por la razón antes mencionada se crearon una serie de reglas de cortesía que los robots deberían de cumplir, facilitando así su convivencia con los servidores de páginas Web. De no ser así se arriesgan a que los administradores de los servidores Web les desconecten cuando detecten que sus solicitudes no son atendidas.

Existen dos protocolos para la exclusión de robots:

a) Protocolo de administración del servidor.

Se basa en tres apartados:

1. **Identificación** del usuario y el robot en las cabeceras HTTP cuando se realicen las solicitudes. Toda solicitud de un recurso HTTP puede llevar a su vez una serie de informaciones (en el apartado que se denomina cabecera o "head" HTTP) sobre qué usuario está solicitando el recurso y demás información. En concreto las cabeceras HTTP "From" y "User-Agent", son fundamentales para el caso.

2. Que el robot espere un cierto **intervalo de tiempo** entre solicitudes a un mismo servidor (se recomienda de 15 a 30 segundos entre solicitud y solicitud). Así se alivia en cierta medida la sobrecarga que se va a producir en el servidor, permitiéndole atender en los intervalos las peticiones realizadas desde otras máquinas.

3. La lectura del **fichero "robots.txt"** de cada servidor, que mediante una sintaxis especial especifica a qué robots se les limita o autoriza a solicitar determinadas páginas/directorios. Este fichero ha de ser generado por los administradores del servidor de páginas Web y almacenado en esa ruta (ruta relativa al directorio de inicio del servidor Web), con lo que los administradores sólo deberán actualizarlo, limitando así el acceso a páginas con escritos cgi y demás. Un ejemplo de fichero sería el siguiente:

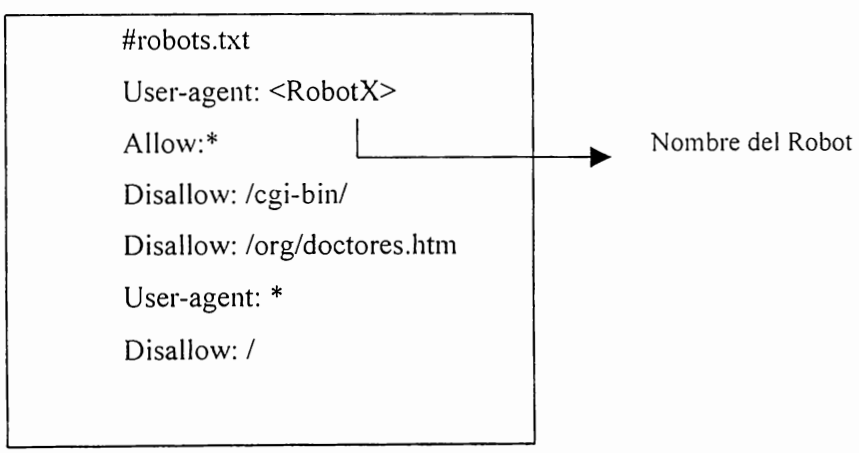


Ilustración 8. Ejemplo de fichero /robots.txt

Con esta configuración no se permitiría a cualquier robot acceder al Servidor Web, sino únicamente al llamado RobotX. A éste, sin embargo se le permite solicitar cualquier recurso que no sea ni la página Web "/org/doctores.htm", ni que dependa del directorio "/cgi-bin". Así por ejemplo podría solicitar el recurso "/index.htm", pero no el recurso "/cgi-bin/mas/form."

b) Protocolo para los diseñadores de páginas Web.

Un segundo protocolo se basa en las etiquetas **"META"** de las páginas Web. Las etiquetas META son parte del código HTML del que se compone una página Web, en las que la persona que las ha creado puede especificar si esta página es "recuperable" o no y si sus enlaces se deben seguir por los robots o no, mediante una etiqueta META.

Sintaxis: <META NAME="ROBOTS" CONTENT="NOINDEX">.

Existen seis valores válidos para la etiqueta **"CONTENT":**

1. INDEX: Indica que los robots pueden presentar esta página en los índices de las búsquedas que realicen.

2. NOINDEX: Indica que los robots no pueden presentar esta página en los índices de las búsquedas que realicen.

3. FOLLOW: Indica que los robots pueden seguir los enlaces de esta página para continuar su búsqueda.

4. NOFOLLOW: Indica que los robots no pueden seguir los enlaces de esta página para continuar su búsqueda.

5. ALL: Equivale a las etiquetas INDEX y FOLLOW a la vez.

6. NONE: Equivale a las etiquetas NOINDEX y NOFOLLOW.

Con este otro protocolo, la responsabilidad de restricción de acceso queda para los diseñadores de páginas en lugar de a los administradores de los servidores Web.

CAPITULO III. DESARROLLO DEL PROYECTO

3.1 SITUACIÓN ACTUAL

Actualmente se ha incrementado el empleo de herramientas de búsqueda limitándolas a áreas específicas; el objetivo de éstas es optimizar recursos de tal manera que se invierta menos tiempo y se obtenga información que realmente supla las necesidades del usuario.

El Mirador (www.svnet.org.sv/mirador/) es la única herramienta desarrollada a nivel nacional que busca solventar la necesidad de encontrar información publicada en el país. Sin embargo, este buscador contempla únicamente parte de la información publicada a nivel nacional; además de presentar una interfaz bastante elemental para el usuario final.

Existe además otra herramienta de búsqueda en el sitio Web www.cipotes.com.sv, ésta permite extraer información relacionada al país y ha sido desarrollada en Long Island, New York.

3.2 FASE INVESTIGATIVA

La fase de desarrollo del proyecto se centró inicialmente en la búsqueda de material bibliográfico a nivel nacional, para lo cual se visitaron bibliotecas, centros de estudios y posteriormente librerías; sin embargo, fue mínima la información encontrada con relación directa al tema, de tal manera que sólo se obtuvieron conceptos básicos relacionados a los motores de búsqueda. Por ejemplo: tipos de motores y componentes de un motor de búsqueda. A partir de este punto la investigación se orientó hacia Internet en donde fue posible encontrar mayor cantidad de información conceptual relacionada al desarrollo del proyecto.

Gracias a la información recopilada se identificó que los elementos principales sobre los cuales sería necesario trabajar, son: el Robot Web, que alimenta la base de datos con la información de los sitios nacionales y la Interfaz Web - Base de Datos, que permite la consulta de esa información a través del Web; ya que sobre el Sistema Operativo, la Base de Datos y el servidor Web, ya existía algún grado de conocimiento por parte de los miembros del grupo.

Al investigar sobre los robots Web en Internet se encontraron sitos Web que poseen referencias a información sobre una serie de robots desarrollados o en vías de implementación; diseñados para trabajar en diferentes plataformas, lenguajes, con diversos propósitos y características específicas. Para lo cual se recopilaron aquéllos que cumplen con las definiciones planteadas en el presente proyecto.

En forma paralela a la investigación de los robots Web, se obtuvo información relacionada al empleo de las interfaces compatibles con el servidor Web a utilizar en el desarrollo del proyecto. Dentro de las interfaces investigadas se contemplan el Internet Database Conector (IDC) y Active Server Pages (ASP).

Adicionalmente fue necesario investigar sobre los controladores para establecer la conexión a la base de datos a partir del robot Web y de la interfaz, para ello se estudió la configuración y funcionamiento del Open DataBase Conector (ODBC) y JDBC. De igual manera fue necesario iniciar el estudio de los lenguajes de programación Perl y Java, debido a que la mayor parte de robots existentes se encuentran en dichos lenguajes.

3.3 FILTRO DE INVESTIGACIÓN

En esta etapa se procedió a analizar toda la información recopilada en la fase anterior, de tal manera que se restringiera el volumen de la misma, en especial lo referente a robots. Dicha depuración se llevó a cabo considerando características específicas para los robots Web e interfaces, tales como: distribución gratuita de la herramienta, plataforma sobre la que trabaja; para los robots se tomó en cuenta además disponibilidad de código fuente para su modificación, el tipo del robot, lenguaje de programación en que se desarrolló, modularidad en el diseño del robot, cumplimiento del estándar de exclusión de robots y eficiencia en su desempeño.

Inicialmente el proceso de filtrado se enfocó en aquellos mecanismos que trabajan sobre una plataforma de Windows NT, sin embargo la mayoría de las herramientas de este tipo con las características requeridas, estaban a la venta con costos muy elevados; de acuerdo a ello se incluyeron dentro de las opciones de robots a evaluar, aquellos que se desarrollaron para trabajar en otras plataformas empleando lenguajes de programación portables a otros sistemas operativos, tales como Windows NT.

Así también dentro de esta fase, se mantuvo contacto por medio de correo electrónico con los desarrolladores de algunos de los robots que cumplieron con el grupo de características antes mencionadas, con el objetivo de obtener el apoyo y asesoría necesarias sobre la instalación y funcionamiento de los mismos.

3.4 EVALUACION DE ALTERNATIVAS

3.4.1 Montaje del Laboratorio de Prueba

Para la instalación de la Intranet o red interna, fue necesario considerar los elementos que brindaran un ambiente adecuado para llevar a cabo la ejecución de los robots seleccionados. Entre los principales componentes utilizados se tienen tres computadoras, cuyas características se presentan a continuación:

1. PC 1. Computadora Pentium a 200 Mhz con un disco duro de 4 Gb, operando bajo un sistema operativo Windows NT (Versión 4.0) y Windows'95. En esta unidad se instaló el Service Pack 3.0, y la Interfaz Active Server Page. Además cuenta con los intérpretes de Java y Perl.

2. PC 2. Computadora Pentium a 133 Mhz, con una capacidad en disco duro de 2 Gb, tiene instalado como sistema operativo Windows'95 y fue necesario instalar Personal Web Server y Microsoft Access, además de Java y Perl.

3. PC 3. Esta unidad es un Clon que funciona a una velocidad de 450 Mhz, posee un disco duro de 6.4 Gb. Funciona con Windows NT (Versión 4.0) y Windows'95 y se le instaló Microsoft Access y los intérpretes de Java y Perl.

Las máquinas con sistemas operativos de Windows NT, emplearon como servidor Web el Internet Information Server. Estas unidades de equipo se interconectaron por medio de una red de área local, utilizando para su comunicación cableado coaxial.

3.4.2 Evaluación de Robots Web

Se seleccionaron doscientos noventa robots para un nivel de investigación primario, de los cuales a partir del proceso de filtrado, se obtuvieron siete robots Web que fueron candidatos para efectuar

pruebas de funcionalidad a nivel de una Intranet. Las pruebas de la implementación de los robots, se resumen a continuación:

SpiderBot 1.0

Nombre:	SpiderBot 1.0
Desarrollado por:	Ignacio Cruzado Nuño
Última Actualización:	Septiembre 1998
Lenguaje de Programación:	C++ y Tcl
Plataforma:	Unix y Microsoft Windows
Número de Archivos:	317
Número de Directorios:	19
Base de Datos:	----------
Alcance:	Sitio Web.
Descripción:	Es un recuperador de páginas web inteligente, que por medio de un criterio y de URL's iniciales, realiza una búsqueda en Internet; salvando las páginas que explora, ordenándolas según las preferencias del usuario y modificándolas para que este pueda navegar por ellas en forma local, sin necesidad de estar conectado a Internet mientras lee.
Comentarios:	Con la información que se obtuvo de esta herramienta, se procedió a la fase de pruebas y mientras se analizaba el funcionamiento de los módulos del robot, se descubrió que en varios de ellos se hacia referencia a segmentos de código que no estaban disponibles dentro de los archivos obtenidos. Por lo tanto se procedió a establecer comunicación, vía correo electrónico con el desarrollador de este sistema, quien manifestó no poder brindar el código completo del robot en forma gratuita, por lo tanto se continuó la investigación de otras herramientas. Al finalizar la prueba, se determinó que aunque SpiderBot no es el idóneo para los fines perseguidos en este proyecto, algunas de las bibliotecas empleadas en este robot podrían ser de utilidad posteriormente para modificar el robot seleccionado.

ht://Dig

Nombre:	ht://Dig
Desarrollado por:	Andrew Scherpbier, Agosto 1999
Última Actualización:	Abril 1999
Lenguaje de Programación:	C y C++
Plataforma:	Unix
Número de Archivos:	1431
Número de Directorios:	608
Base de Datos:	Berkeley v.2.6.4, Dic 1998
Alcance:	Intranet.
Descripción:	El ht://Dig es un sistema completo de indexamiento y búsqueda en el World Wide Web, desarrollado para cubrir necesidades de búsqueda para compañías, campus universitarios o inclusive subsecciones de un sitio web; fue desarrollado en la universidad del estado de San Diego, con el propósito de buscar dentro de los servidores web del campus universitario. Actualmente esta siendo utilizado por una infinidad de organismos, instituciones y empresas, entre las que se cuentan: Alfred University (www.alfred.edu), Arizona State University East (www.east.asu.edu), Astronomy Net (www.astronomy.net), The Australian Parlament (www.parlament.gv.at) y otros. La distribución de los archivos de este sistema se encuentran divididos modularmente de acuerdo a los propósitos de sus programas, cada directorio cuenta con las librerías necesarias y los códigos fuentes de los módulos.
Comentarios:	Para las pruebas de este motor de búsqueda se procedió a configurar e instalar los archivos necesarios para que trabajara sobre una plataforma de sistema operativo Win32 (Windows NT). Sin embargo, no fue posible la migración total de todos los archivos, especialmente aquellos relacionados a la configuración del sistema, debido a que están estrechamente ligados y son dependientes de configuraciones propias de un sistema operativo Unix.

BDDBot

Nombre:	BDDBot
Desarrollado por:	Tim Macinta, Febrero 1998
Última Actualización:	Febrero 1998
Lenguaje de Programación:	Java
Plataforma:	Toda aquella que soporte Java.
Número de Archivos:	35
Número de Directorios:	15
Base de Datos:	Archivo de Texto
Alcance:	Intranet.
Descripción:	Es un motor de búsqueda compuesto por: un robot Web, un grupo de scripts de búsqueda y un servidor Web, permite indexar y realizar consultas de información almacenada en una Intranet. Este ha sido desarrollado totalmente en Java, por lo que en teoría deberá de correr en cualquier plataforma sobre la cual se pueda instalar este lenguaje de programación. La distribución física de los archivos fuente y objeto que componen el robot se encuentra organizada, a través de directorios, dependiendo del propósito y dependencia entre sí, ya sea que formen parte del robot o del servidor Web.
Resultados de Evaluación:	Una vez logrando instalar y configurar debidamente tanto Java como el motor de búsqueda, se procedió a identificar los *scripts* que forman el robot Web. Se realizaron modificaciones de tal manera que el robot no sólo buscara en el servidor Web, sino pudiera expandirse a buscar en otros servidores Web configurados en una Intranet, interconectados por medio de una red local. Esta tarea fue ejecutada de manera satisfactoria.

Resultados de Evaluación:	El robot Web desarrollado para este motor, maneja archivos de texto totalmente independientes del código fuente del mismo, para definir la lista inicial de URL's, un listado de sitios a excluir y un listado de sitios permitidos, lo cual brinda dinamismo y mayor versatilidad en su ampliación. Para almacenar la información de las páginas Web, actualmente emplea un archivo de texto en el cual introduce tanto los URL´s que ha podido indexar, como cada una de las palabras que ha encontrado, sin repetirlas. Para el desarrollo del proyecto se modificó el código para realizar pruebas con JDBC, insertando por medio de sentencias SQL a una base de datos de Microsoft Access las palabras de las páginas Web. Lo cual se llevó a cabo de manera satisfactoria.

WebMonkey

Nombre:	Webmonkey
Desarrollado por:	Brian Slesinsky
Última Actualización:	23 de Abril de 1997
Lenguaje de Programación:	Perl 4
Plataforma:	UNIX
Número de Archivos:	2 scripts cgi y 1 archivo html
Número de Directorios:	1
Base de Datos:	Berkeley DB
Alcance:	Sitio Web.
Descripción:	Webmonkey es un robot que explora un Sitio Web y construye a partir de las páginas Html visitadas dos archivos: Uno de ellos que contiene los nombres de la páginas y sus correspondientes Url's, y el otro que forma una tabla de palabras conocida como Archivo Invertido, en donde cada palabra tiene una referencia hacia la página que la contiene. Toda esta información es almacenada en una base de datos Berkeley (search_index.db), que facilita la búsqueda de palabras específicas. WebMokey maneja dos *scripts cgi*, el primero llamado crawler (find.cgi) que lee todos los archivos html en el Sitio Web y construye el archivo invertido almacenando la información en una base de datos Berkeley. El segundo llamado Search (search.cgi) que realiza la búsqueda de palabras que el usuario solicita a través de un formulario, para ello se apoya en el uso de comandos de búsqueda.

Resultados de Evaluación:	Inicialmente el código del Webmonkey está diseñado para trabajar sobre Unix, debido a ello se realizaron las modificaciones necesarias para su funcionamiento sobre Windows NT. Las pruebas realizadas a nivel de sitio Web fueron satisfactorias. Webmonkey genera un archivo invertido sobre una base de datos Berkeley, se procedió a la instalación de los controladores ODBC para Perl, se modificó el script para que, utilizando dichos controladores ODBC y sentencias SQL, insertara en una base de datos de Microsoft Access. WebMonkey necesita para su adecuado funcionamiento en el Web la instalación de la librería LibWWW-perl. La importancia de la librería en la ejecución del robot, radica en que permite que éste se extienda hacia el Web, y para esto se deben manejar los comandos propios de dicha librería. La instalación de la librería antes mencionada se concluyó sin mayores problemas; sin embargo al combinar el funcionamiento del robot y ésta , los resultados no fueron del todo satisfactorios, puesto que el robot presentó una serie de errores relacionados al empleo de comandos de dicha librería.

MOMSpider

Nombre:	MOMSpider
Desarrollado por:	Roy T. Fielding
Última Actualización:	6 de Mayo de 1995
Lenguaje de Programación:	Perl 4
Plataforma:	UNIX
Número de Archivos:	2 scripts cgi y 1 archivo html
Número de Directorios:	1
Base de Datos:	----------
Alcance:	Intranet.
Descripción:	El MOMSpider es un robot que recorre el Web validando los enlaces y generando estadísticas. Usualmente puede ejecutarse sobre la familia de sistemas operativos Unix, pero se deben considerar ciertas reglas para su configuración. Esta herramienta surgió como un proyecto de investigación de la Universidad de California en 1993, con el objetivo de poner a disposición la información manejada dentro de la misma.
Resultados de Evaluación:	Momspider emplea para su funcionamiento los procesos de Unix llamados Demonios; para proceder a ejecutar el software obtenido, se inició con la búsqueda de los procesos análogos en Windows NT. Se tuvo especial cuidado en la parte de la configuración del robot y en la creación del ambiente en el cual correría. Así por ejemplo, se instaló Perl 5 y la librería Libwww-perl. Debido a que esta librería es muy importante para la ejecución de este robot, se dedicó tiempo para investigar los lineamientos de la instalación, así como también los módulos que la constituyen y que proveen una simple y consistente interfaz de programación para el World Wide Web. En la fase de configuración del robot surgieron muchos problemas especialmente cuando el robot emplea comandos de Unix, además de ciertas librerías que no estaban disponibles en el sitio donde se obtuvo el código del MOMSpider; dichas librerías fueron construidas por el desarrollador y no pudieron obtenerse.

Xavatoria

Nombre:	Xavatoria Indexed Search Engine
Desarrollado por:	Jeff Carnahan
Última Actualización:	Julio 1998
Lenguaje de Programación:	Perl 5
Plataforma:	UNIX, Windows NT
Número de Archivos:	2 archivos de perl. 2 archivos html y 1 archivo texto(data file).
Número de Directorios:	2
Base de Datos:	Archivo texto.
Alcance:	Intranet.
Descripción:	Xavatoria es un robot de búsqueda escrito en Perl, diseñado para recorrer Sitios Web con grandes cantidades de texto. Xavatoria está formado principalmente por dos archivos PL: Build.pl y Search.pl. Build.pl se encarga de revisar en una Intranet las páginas html y crear a partir de la información que contienen, el archivo de datos Index.txt. Este *script* requiere de parámetros como la ubicación del archivo índice, la ubicación de los archivos sobre el sistema en el cual se iniciará la búsqueda. Es decir, se definen el directorio base, los URL's base, las extensiones de páginas que se revisarán, directorios y URL's que no se tomarán en cuenta. Search.pl es el encargado de realizar la búsqueda de palabras en el archivo de datos Index.txt, aquí se especifican la dirección de la página html donde se presentarán los resultados y la dirección de un archivo html donde se lleva un registro de las búsquedas realizadas y los URL's resultantes. Los *scripts* de este robot funcionan en teoría sobre Unix y Windows NT, siempre y cuando se tenga instalado Perl 5 y se hagan ciertas modificaciones de configuración.

Resultados de Evaluación:	Para realizar las pruebas de Xavatoria se estudió el código de los *scripts* manejados por dicho robot de búsqueda. Se procedió a modificar los parámetros de configuración dentro del *script* Build.pl, luego se editaron el código referente a la construcción del archivo invertido y el de búsqueda de criterios. Las pruebas efectuadas a nivel de una Intranet fueron satisfactorias. Se incluyó la utilización de una base de datos de Microsoft Access empleando para ello los controladores ODBC para Perl. Xavatoria está diseñado de manera que emplea una lista de directorios y URL's iniciales definidos dentro del *script* build.pl, existen además otras asignaciones que se definen estáticamente dentro de dicho *script.*

Perlfect Search 3.03

Nombre:	Perlfect Search 3.03
Desarrollado por:	Perlfect Solutions
Ultima Actualización:	1 de abril de 1998
Lenguaje de Programación:	Perl 5.004
Plataforma:	UNIX, Windows NT
Número de Archivos:	4 archivos PL, 4 archivos db, 2 archivos texto y 1 archivo Html
Número de Directorios:	3
Base de Datos:	Berkeley DB
Alcance:	Sitio Web
Descripción:	Perlfect Search es un robot que indexa y realiza búsquedas sobre un sitio Web. Está compuesto básicamente por dos elementos: El Indexer (indexer.pl) que revisa y construye un archivo invertido llamado inv_index_db, en el cual almacena palabras que tienen como fuente el sitio Web donde reside; además crea otro archivo llamado docs_db, almacenando en este los URL's de las páginas visitadas. El otro componente de esta herramienta lo constituye el *script* de búsqueda (search.pl), un *script cgi* que ejecuta consultas sobre el índice y despliega los resultados en una página html, en un formato estándar incluyendo título, descripción y la información relevante para cada documento que cumple los criterios establecidos. Perlfect Search mantiene un control total sobre el contenido del índice empleando para ello reglas de exclusión de cada archivo o directorio. Características avanzadas incluyen Stop Words, un potente mecanismo de exclusión a través del archivo stopwords.txt.

Comentarios:	Perlfect Search cuenta con un asistente para su instalación (setup.pl), este genera la información que necesita el archivo conf.pl para la correcta configuración de dicho robot. Se realizó la configuración de Perlfect Search directamente desde el archivo conf.pl, debido a que el asistente tenía parámetros específicos para Unix. Se procedió a efectuar las pruebas sobre dicho robot siendo estas funcionales a nivel de sitio Web. Originalmente, Perlfect Search emplea una base de datos Berkeley. A través del uso de los controladores ODBC para Perl y de sentencias SQL se implementó el uso de una base de datos de Access. Posteriormente se hicieron las modificaciones para su ampliación a una Intranet, obteniendo buenos resultados en su funcionamiento aunque el rendimiento en el proceso de indexamiento no fue satisfactorio.

3.4.3 Evaluación de Interfaces

La evaluación de las interfaces se llevo a cabo en dos etapas principales, consistiendo la primera en el estudio del funcionamiento y características propias de cada interfaz, como sus requerimientos e instalación; mientras que la segunda etapa se enfocó en el estudio y análisis de los ejemplos de empleo de las mismas, arrojando así los siguientes datos:

Internet Database Conector (IDC)

Nombre:	Internet DataBase Conector
Desarrollada por:	Microsoft Corporation
Bases de Datos:	Bases Microsoft y toda base de datos que soporte ODBC
Sistema Operativo:	Windows NT v.4.0
Disponibilidad:	Distribuida con el Internet Information Server al adquirir Windows NT Server 4.0
Ejecución:	IDC se ejecuta como aplicación DLL a través del programa HTTPODBC.DLL y empleando ODBC para conectarse a las bases de datos.
Características:	- Funciona en combinación del servicio WWW y los dispositivos ODBC provistos con el IDC. - Requiere de un archivo ".idc" y otro ".htx" para proveer el acceso a la base de datos. - IDC se ejecuta como una aplicación DLL - Soporta consultas o actualizaciones de las bases de datos a través de sentencias SQL.

Active Server Pages (ASP)

Nombre:	Active Server Pages
Desarrollada por:	Microsoft Corporation
Bases de Datos:	Bases Microsoft y toda base de datos que soporte ODBC
Sistema Operativo:	Windows NT v.4.0
Disponibilidad:	Distribuida a partir del Service Pack 3, para Windows NT 4.0.
Ejecución:	ASP se ejecuta como aplicación DLL, así como los interpretes de los lenguajes que soporta. Para conectarse a las bases de datos emplea ODBC
Características:	- Requiere de un solo archivo con extensión ".asp" - Combina dentro de un archivo los *scripts* de código a ejecutar, con etiquetas HTML y texto. - Permite el empleo de más de un lenguaje de programación dentro del mismo archivo. - Permite ejecutar segmentos de código del lado del cliente para liberar de carga al servidor Web. - Soporta sentencias SQL estándar para acceder a bases de datos. - Utiliza ODBC para conectarse a Bases de Datos.

3.5 SELECCIÓN DE ALTERNATIVAS

3.5.1 Selección del Robot Web

El Robot Web con el cual se ha decidió trabajar en el desarrollo del este proyecto es el BDDBot, desarrollado en Java. Esta selección se llevó a cabo en base a los siguientes criterios:

1. Al ser Java el lenguaje de programación empleado en un 100 % para el desarrollo del motor de búsqueda, provee las ventajas que acompañan a todas las aplicaciones desarrolladas en este lenguaje, entre las que podemos mencionar:

    a. Portabilidad del lenguaje, permitiendo gracias a la forma de compilar los programas, el poder ejecutarse sobre cualquier plataforma en la cual se pueda instalar este lenguaje, con cambios mínimos.

    b. Programación orientada a objetos, permitiendo utilizar en su totalidad las ventajas de la metodología orientada a objetos, tales como el empleo de clases y la herencia, además de sus capacidades para crear programas flexibles y modulares, así como reutilización de código.

    c. Simplicidad de codificación, gracias a que uno de los propósitos de diseño inicial de Java era que debería ser pequeño y simple, y en consecuencia fácil de escribir, compilar y depurar.

    d. Existencia de un API prediseñado para la conexión a bases de datos, permitiendo la ejecución de sentencias SQL y haciendo posible el escribir aplicaciones que permitan interactuar con la Base de Datos, utilizando para ello únicamente Java.

2. Considerando que el Robot Web encontrado fue sujeto a muchos cambios y modificaciones, la característica de modularidad y organización de los archivos que lo conforman fue una de las más importantes, puesto que un mejor control y ubicación de los programas, permitió facilitar el proceso de análisis y comprensión del funcionamiento e interrelación de los mismos.

La forma en que se llevó a cabo la programación y distribución de las clases y módulos para este software, brinda la ventaja de estar organizado dentro de directorios identificados con las funciones que desempeñan sus archivos. Así por ejemplo, el directorio SPIDER almacena todas las clases que conforman al Robot Web; mientras que en el directorio QUERY se encuentran las clases que permiten la realización de consultas a la Base de Datos.

Un factor de gran importancia para la comprensión y depuración de todo programa, lo constituye la comentarización adecuada y documentación del código que conforma un software, característica que cubre el desarrollo del BDDBot, ya que cada una de las clases que lo componen se encuentran debidamente comentarizadas con breves explicaciones del propósito general de la clase, así como de cada uno de los métodos que la forman y la función de sus variables globales.

3. Facilidad de instalación y configuración, debido a que cuenta con documentación en formato HTML, presentando información adicional en relación a la jerarquía de las clases y funcionalidad de las mismas, así como información general de la configuración y ejecución del mismo.

3.5.2 Selección de la Interfaz

La interfaz empleada en el desarrollo del proyecto permitió interpretar una solicitud de información, lograr la conexión a la base de datos y obtener la información requerida. Por medio de esta se pudo acceder a la base de datos para almacenar información recopilada a través de un formulario disponible en el sitio Web del buscador. Estas funciones se llevaron a cabo en conjunto con los controladores de conexión ODBC.

De acuerdo a ello, la interfaz a emplear fue Active Server Page (ASP). Las razones por las que se optó por esta interfaz se enumeran a continuación:

1. Active Server Page es una interfaz que además de permitir el acceso a una base de datos permite el desarrollo de páginas Web dinámicas, logrando así acceder a una base de datos para almacenar, recuperar y presentar información de la misma.

2. En un archivo .ASP puede emplearse código de varios lenguajes de programación, brindando así la facilidad de utilizar un lenguaje de programación adecuado al proceso que se desee implementar.

3. Active Server Page permite la descentralización de código del lado del servidor, logrando con ello la optimización de recursos de éste, al realizar ciertas tareas por parte del cliente.

4. ASP viene incorporado en el Internet Information Server (IIS), por lo cual no se necesita adquirir licencia adicional para su utilización.

3.6 DISEÑO Y FUNCIONALIDAD DEL MOTOR DE BUSQUEDA

3.6.1 Cambios y Modificaciones Realizadas sobre el BDDBot para Originar el Cyber Izalco

1. Originalmente el BDDBot fue desarrollado para trabajar dentro de una Intranet, descubriendo las páginas web de los servidores interconectados, sin embargo para emplearlo en el desarrollo de este proyecto ha sido necesario ampliar su funcionalidad para que sea capaz de trabajar en Internet, empleando el protocolo HTTP y permitiendo la creación de un índice que contenga la información de los sitios pertenecientes a las páginas Web nacionales, empleando para ello una Base de Datos en Microsoft Access.

2. Debido a que en el diseño original del BDDBot, este contempla exclusivamente características para funcionar como un robot de indexamiento, ha sido necesario el desarrollar e integrar un modulo adicional al robot, cuyo propósito es el de implementar la funcionalidad de mantenimiento sobre los sitios, logrando con ello verificar la validez de los URL's previamente almacenados en la base de datos.

   El módulo de mantenimiento, consulta cada uno de los URL's almacenados en la base de datos, dentro de la tabla T_URL (sitios indexados por el robot), conectándose a cada uno de ellos para verificar que su estado sea activo; si por alguna razón el URL no está activo, se marca dicho URL como pendiente, si en la siguiente ocasión en que el URL sea verificado, este no responde, entonces se elimina dicho registro, junto con cada uno de los registros relacionados con este URL.

3. Originalmente el BDDBot es un motor de búsqueda que está conformado por un Robot Web, un servidor Web y una interfaz de búsqueda, sin embargo para el desarrollo de este proyecto en particular solamente se emplea, su robot Web, por lo cual se han realizado las modificaciones necesarias para lograr la separación de los módulos, buscando el independizar al robot de los otros componentes de la herramienta.

   Para lograr la separación de los módulos, se realizaron cambios sobre las clases de EnginePrefs, Monitor, Crawler y otras, logrando la independencia de estas.

4. El BDDBot crea su índice dentro de un archivo de texto (Main.db), en el cual almacena las palabras encontradas dentro de las páginas y sus URL's respectivos, organizándolas en orden alfabético, sin embargo, esta metodología no es adecuada si se considera el volumen de información que se espera manejar de las páginas pertenecientes al dominio SV; por lo que es necesario que toda esta información se registre dentro de una base de datos, con el propósito de agilizar el acceso a la misma.

   Con este fin se modificaron las clases EnginePrefs, Indexer y URLStatus, para que estas se conecten a la base de datos, empleando un controlador ODBC llamado "buscador", a través del cual se consulta la información previamente almacenada o se insertan nuevos registros dentro de la base de datos, permitiendo crear el índice de la información contenida en las páginas Web nacionales.

   Con el empleo de controladores ODBC, se deja abierta la posibilidad de emplear diversos manejadores de bases de datos, siendo necesario únicamente que el controlador tenga el nombre de "buscador" y que el modelo Entidad - Relación de la base de datos, cumpla con la estructura y denominación de objetos definidos, así por ejemplo la base podría estar en Microsoft Access, como en Sybase o Oracle.

5. Todo robot Web para iniciar su recorrido por Internet, parte de una lista inicial de URL's. En el caso del BDDBot esta lista se encuentra dentro de un archivo de texto llamado "Urls.txt". Con el objetivo de incrementar su funcionalidad, así como facilitar su administración, se ha modificado el código del robot, con el propósito que pueda leer estos URL's de la base de datos, permitiendo además que el robot busque dentro de los

URL's que hayan sido agregados desde el Web, por personas interesadas en que sus sitios sean indexados por el buscador.

La lista inicial de URL's se obtiene a partir de las tablas T_INICIAL y T_DEPURADOS almacenadas dentro de la base de datos, la primera de ellas corresponde a sitios pertenecientes al dominio SV, mientras que en la segunda se almacenan aquellos URL's ingresados a través del formulario en el sitio Web, los cuales no necesariamente deben de pertenecer al dominio de El Salvador (SV) y que han sido seleccionados previamente por el administrador de la aplicación, como sitios realmente válidos, es decir, desarrollados por salvadoreños o con información relativa al país, ejemplo de estos sitios podrían ser, sitios ubicados en Web Hostings gratuitos como Geocities, Xoom o LatinWeb.

6. Originalmente el BDDBot es un motor de búsqueda desarrollado para trabajar en una Intranet, por lo cual no maneja ningún tipo de restricción referente a donde desarrollara la búsqueda de paginas, ni del dominio sobre el cual trabaja; es por ello que al ampliar la funcionalidad del robot para que busque en el Web, ha sido necesario desarrollar un mecanismo para restringir su búsqueda a solo los sitios nacionales, es decir, pertenecientes al dominio SV, incluyendo además aquellos publicados por personas particulares empleando el servicio de Web Hostings gratuitos y que hayan previamente solicitado que el robot indexe sus sitios, introduciendo los URL's e información adicional a través de un formulario con este propósito.

   La restricción del dominio sobre el cual trabajara el robot, se ha desarrollado realizando una validación sobre cada uno de los URL's a indexar, dependiendo del tipo de URL que sea y del cual provenga.

   Para las páginas pertenecientes al dominio SV se seguirán todos aquellos enlaces que pertenezcan también a este dominio; mientras que en aquellas páginas agregadas y que no pertenecen al dominio, solo se consideraran los enlaces al interior del sitio en el cual se alojan.

Así por ejemplo:
Suponiendo que la página http://members.xoom.com/ralvarez/index.html tiene enlaces a http://www.ibm.com y a http://members.xoom.com/ralvarez/info.htm solo se considerara el segundo URL, porque este se encuentra dentro del mismo sitio, el primero no se seguirá, ya que no pertenece al dominio, ni al sitio que se indexaba originalmente.

Para cumplir con este objetivo se han modificado las clases EnginePrefs, Indexer y URLStatus

7. Idealmente todos los robots web, por razones éticas deben de cumplir con ciertas reglas o normas establecidas en Internet, que están relacionadas a indexar o no los sitios Web que encuentran, a partir de permisos que otorga el WebMaster de cada sitio, por medio de la creación del archivo robots.txt . A estas normas se les conoce como protocolo de exclusión de robots.

   Con la finalidad que el robot web del Cyber Izalco cumpla con este protocolo, se ha modificado el código fuente del robot para que lea e interprete las reglas de exclusión dentro del archivo robots.txt del raíz de cada servidor web, si es que lo poseen, así como por el chequeo del HEADER de las páginas, verificando en ambos casos los permisos de indexamiento sobre las mismas.

8. La ampliación del robot para que trabaje sobre internet, trae consigo la responsabilidad de evitar por medio de políticas que el robot interfiera con los sitios Web que visite, al cargarlos demasiado con múltiples requisiciones de páginas simultaneas. Por ello en el robot se han desarrollado una serie de políticas orientadas a no causar saturación sobre los servidores que se visitan, para el caso se ha definido un periodo de espera de 5 segundos entre cada página descargada, este cambio se efectuó en la clase EnginePrefs.

9. Ya que el BDDBot fue desarrollado para manejar volúmenes de información limitados, no considera restricciones en cuanto a la información que se almacena en el archivo de índice; sin embargo, considerando la cantidad y relevancia de la información que el robot deberá indexar, perteneciente al dominio SV, se modificaron los programas necesarios de tal forma que se introduzca a la base de datos, solamente aquellas

palabras significativas, excluyendo elementos gramaticales tales como artículos, preposiciones, conjunciones y otros caracteres especiales.

Con este objetivo se ha modificado el robot, de tal forma que durante el proceso de inicialización se lea la tabla T_ELEGRAMATICAL, en la cual se encuentran aquellas palabras que no se desea almacenar dentro de la base de datos, debido a que no aportan mayor relevancia en cuanto al contenido de las paginas y por su alta frecuencia de repetición en el contenido de las mismas.

Entre los elementos gramaticales a excluir están palabras en inglés y español, las cuales pueden ser administradas fácilmente gracias al sistema de administración desarrollado en Microsoft Access. De igual forma dentro de la base no se incluirán las cifras numéricas encontradas en las páginas Web.

3.6.2 Descripción General de Cyber Izalco.

Cyber Izalco es un motor de búsqueda que ha sido diseñado para almacenar en una base de datos, el contenido de las páginas pertenecientes al dominio SV de El Salvador y todas aquellas páginas agregadas manualmente y que no necesariamente se encuentren dentro de este dominio, previa depuración por parte del administrador del motor de búsqueda.

El siguiente diagrama muestra la integración de los componentes del motor de búsqueda en un servidor web, a través de este gráfico se puede apreciar como el robot es una aplicación totalmente independiente del servidor web, ya que para acceder a Internet no necesita de este.

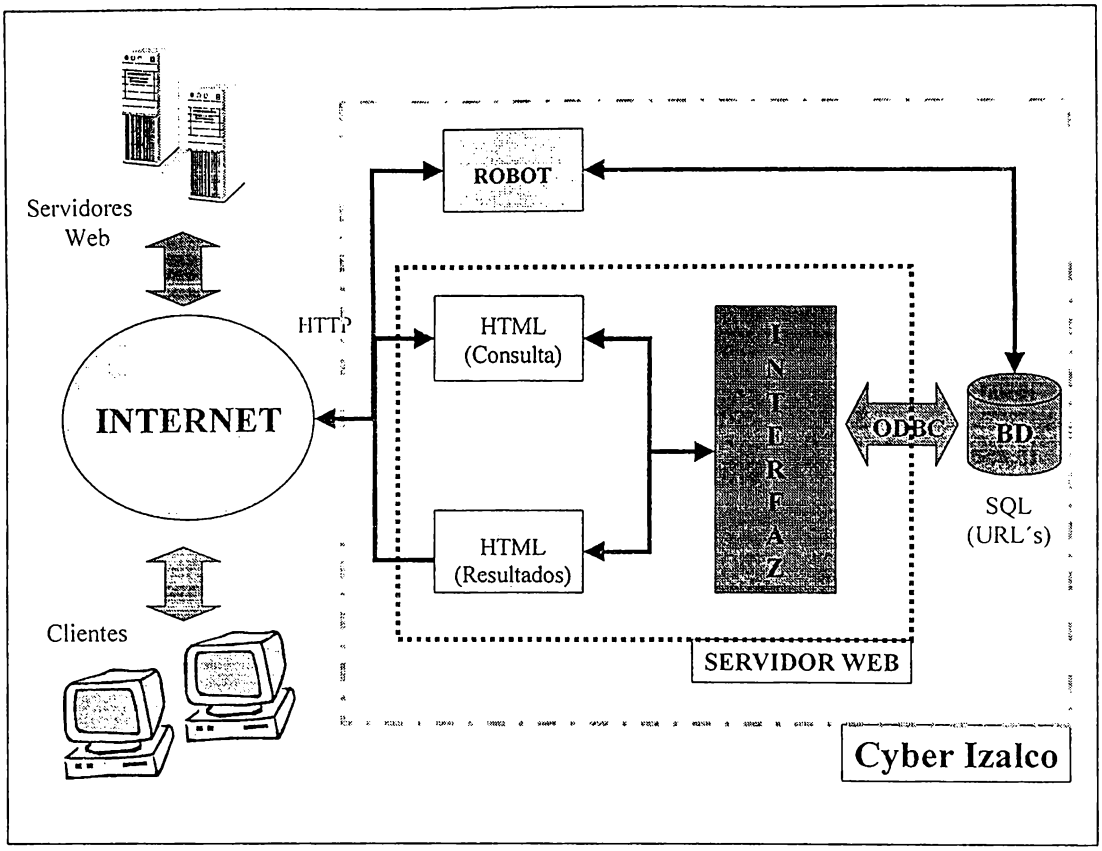


Ilustración 9. Esquema Funcional del Motor de Búsqueda Cyber Izalco.

Como puede notarse en este diagrama los principales componentes del Cyber Izalco son:

1. Robot Web, el cual está compuesto de dos módulos desarrollados en java, los cuales le brindan las características de robot de indexamiento y de mantenimiento, así el primero de ellos se encarga de alimentar la base de datos con la información que encuentra en las páginas web, pertenecientes al dominio de búsqueda; mientras que el segundo permite dar mantenimiento a los URLs almacenados previamente dentro de la base de datos, esta tarea consiste en verificar la validez de cada uno de los URLs de la tabla T_URL, marcando o eliminando aquellos URLs que por uno u otra razón no respondan.

2. Interfaz Web - Base de Datos: Empleando Active Server Pages (ASP) y utilizando segmentos de código de Visual Basic Script y Java Script, los cuales permiten acceder a la base de datos a través de controladores ODBC, para la presentación de resultados en el Web y adición de información.

3. Base de Datos: la cual se ha implementado utilizando Microsoft Access, sin embargo, como ya se había mencionado anteriormente es posible emplear cualquier otra base que soporte acceso por medio de controladores ODBC, con la única condicionante de mantener el diseño de las tablas en Access en el DBMS a emplear.

3.6.3 Secuencia de Ejecución de los Módulos del Motor de Búsqueda

Módulo de Indexamiento

La secuencia de ejecución del módulo de captura de información del Cyber Izalco se da según los siguientes puntos:

1. Se inicializa el proceso del Crawler ejecutándolo desde el monitor del robot o desde la consola de comandos.

2. A continuación se leen las reglas de exclusión del robot, las cuales indican qué sitios no deben de ser indexados, aún cuando sean candidatos para ello, dichas reglas se encuentran definidas en el archivo llamado "Rules.txt".

3. Posteriormente se cargan los URL's almacenados dentro de la tablas "T_INICIALES" y "T_DEPURADOS" a la lista de URL's a ser descargados por el Crawler, marcándolos con una bandera para identificar los dos tipos de URLs y verificando previamente que no se encuentren dentro de la lista de URL's a excluir.

4. Se inicializa el proceso del Indexer, el cual se ejecuta en paralelo con el Crawler, almacenando el contenido de las páginas que se descarguen.

5. El Crawler intenta obtener la página del primer URL de su lista.

6. Al obtener la página se almacena en un archivo temporal en el directorio "CIzalco\Searchtmp", para posteriormente transferirlo a la lista de URL's pendientes del Indexer y se detiene por algunos segundos antes de bajar la siguiente página, con el propósito de no sobrecargar el servidor que atiende las requisiciones.

7. El Indexer toma el archivo temporal y lo recorre obteniendo todas las palabras que no impliquen código HTML, para introducirlas a un arreglo, en donde si la palabra ya existe se incrementa un contador asociado a cada palabra, el cual indica el número de veces que se encontró esa palabra en la página, por el contrario si es la primera vez que encuentra la palabra, simplemente la agrega e inicializa su contador con uno.

8. Una vez obtenidas todas las palabras de la página descargada, se recorre el arreglo donde están almacenadas, para introducir a la base de datos toda aquella palabra que no pertenezca a la lista de elementos gramaticales a excluir, previamente definida. Al ingresar cada una de las palabras a la base de datos, estas se asocian con el URL de la página en donde se encontró, posibilitando la generación de consultas por palabra.

9. Posteriormente se recorre de nuevo el archivo temporal, obteniendo de él todos los enlaces que contiene.

10. Se chequean cada uno de los URL's encontrados dentro de la página y se validan en relación al tipo de URL del cual provienen, es decir, si el URL del que se origina es del dominio SV, entonces el nuevo URL debe de ser del mismo dominio para que sea considerado; si el URL del que proviene no es del dominio SV, entonces solo agrega aquellos URL's que se encuentren dentro de ese sitio, sin considerar las referencias a sitios externos.

11. Se agregan los URL's que se filtraron en el paso anterior a la lista de URL's pendientes, para que el Crawler obtenga las páginas web de estos.

12. Una vez finalizado el almacenamiento del contenido de la página, se elimina el archivo temporal correspondiente al URL indexado.

13. El indexer toma el siguiente archivo temporal obtenido por el Crawler, mientras se analizaba la página anterior y se iteran las tareas desde el paso seis en adelante, hasta que se obtenga la información completa del dominio SV y de aquellos sitios agregados.

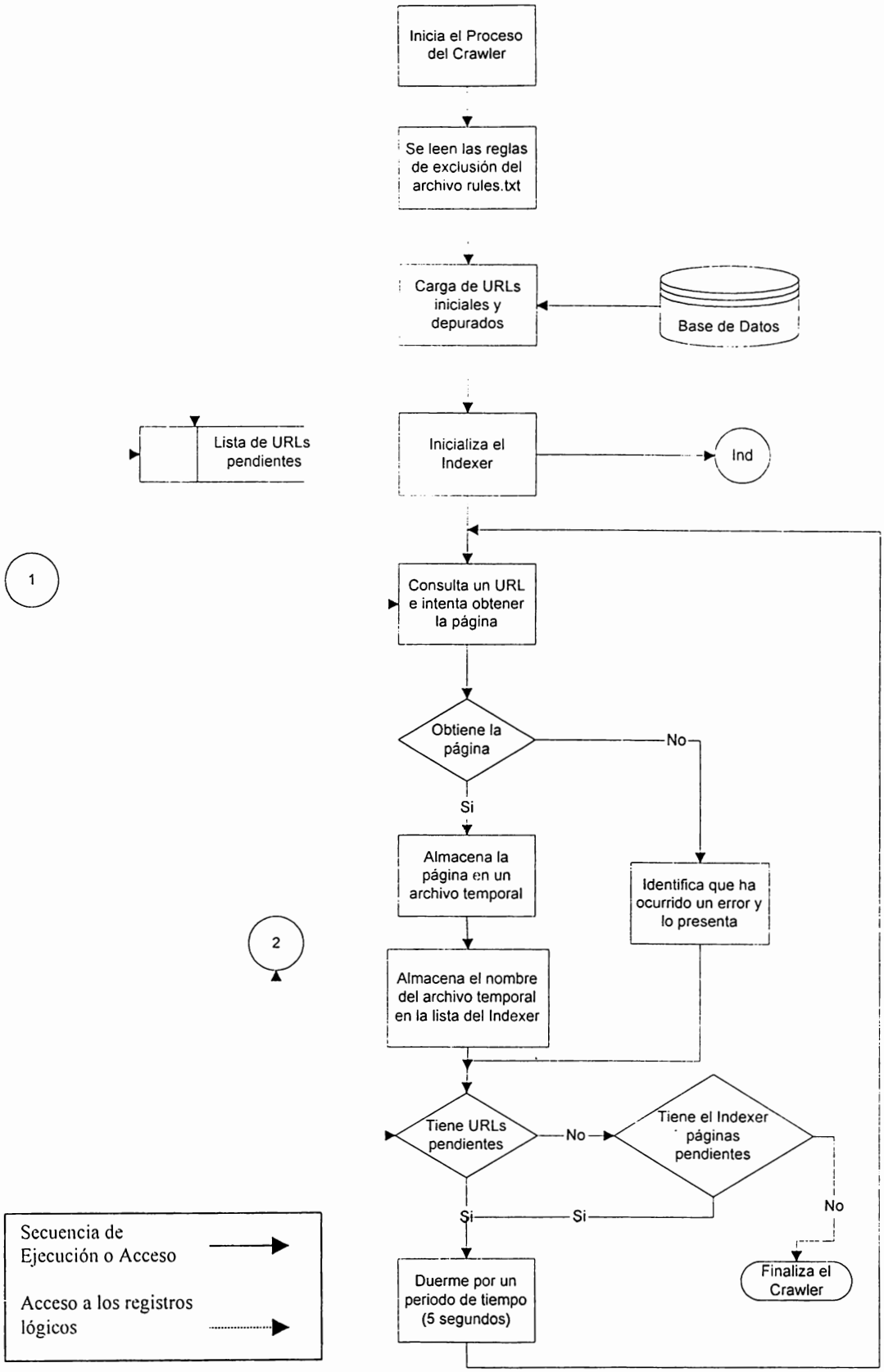


Ilustración 10. Secuencia de Ejecución del Módulo de Indexamiento, Primera Parte.

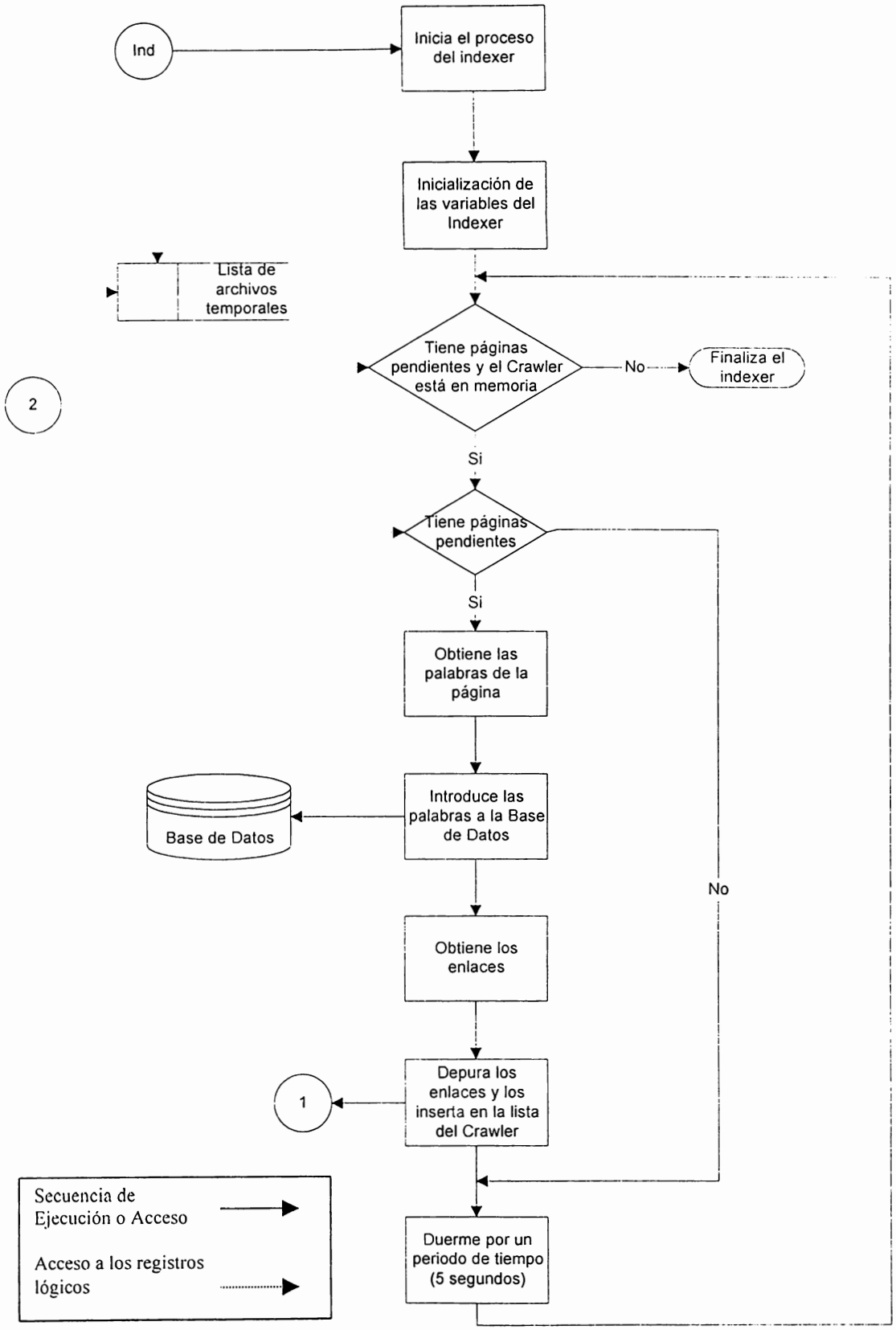


Ilustración 11. Secuencia de Ejecución del Módulo de Indexamiento, Segunda Parte

Módulo de Mantenimiento

La secuencia de ejecución del módulo de Mantenimiento se da en la forma siguiente:

1. Se ejecuta el módulo de mantenimiento desde la interfaz gráfica o de DOS.

2. Una vez iniciada la ejecución del módulo de mantenimiento se crea una conexión a la base de datos y se leen todos los URL's de la tabla T_URL, agregándolos a una lista, así como los estados de cada uno de ellos, (A)ctivo o (P)endiente.

3. Se lee uno de los URL's de la lista y se intenta obtener una respuesta de este URL.

4. Si se obtiene una respuesta del URL y su estado es "A", entonces solamente se actualiza la fecha del URL, indicando la última fecha en que se verificó que ese URL estaba activo.

5. Si se obtiene una respuesta del URL y su estado es "P", entonces se actualiza su estado a "A" y se actualiza la fecha del URL.

6. Si no se obtiene respuesta del URL y su estado es "A", entonces se actualiza el estado del URL a "P", indicando que el URL no respondió la última vez que se les brindó mantenimiento a los sitios y se actualiza la fecha del URL.

7. Si no se obtiene respuesta del URL y su estado es "P", entonces se elimina el registro de la tabla T_URL y todos los registros asociados a este en la tabla T_RELACION.

8. Una vez hechas las actualizaciones al URL, se pasa a leer el siguiente en la lista y se repite el proceso desde el paso tres.

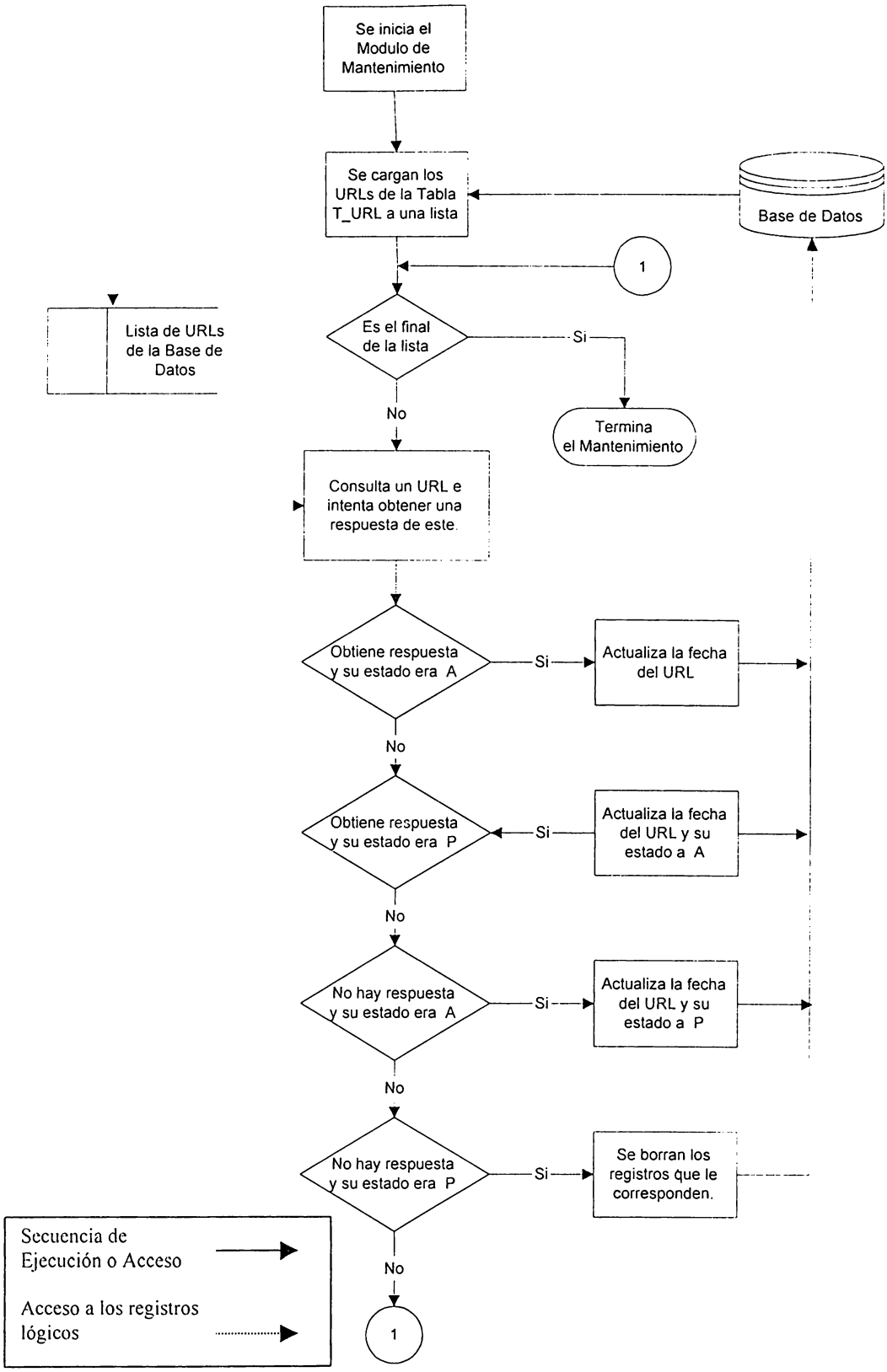


Ilustración 12. Secuencia de Ejecución del Módulo de Mantenimiento.

3.6.4 Ubicación de los Archivos y Función Principal de cada uno de ellos.

La estructura de archivos para el Robot es la siguiente:

```
CIzalco\
        Search\
                Spider\
                        Crawler.*
                        HTMLWordExtractor.*
                        HTMLLinkExtractor.*
                        Indexer.*
                        LinkExtractor.*
                        NullLinkExtractor.*
                        TextWordExtractor.*
                        URLStatus.*
                        WordExtractor.*
                EnginePrefs.*
                Monitor.*
                Mtto.*

        Searchdb\
                Rules.txt
                Cyber_Izalco.mdb

        Searchtmp\

        Util\
                FIFOQueue.*
```

Directorio\ : Nombre de directorio en la estructura de archivos.

*: Indica la existencia de un archivo fuente ".java" y un objeto ".class" con el mismo nombre.

Descripción de la funcionalidad de cada uno de los archivos del Robot Web:

1. **Crawler.*** Archivo principal para el proceso de búsqueda en Internet, esta clase define el proceso del Crawler y carga los URLs iniciales y depurados almacenados en la base de datos de Microsoft Access, para iniciar la ejecución del robot.

2. **Cyber_Izalco.mdb** Base de datos en Microsoft Access, que contiene la información recabada por el robot.

3. **EnginePrefs.*** Archivo de configuración que contiene información general que se necesita compartir entre las clases como por ejemplo: la declaración de cuales serán los directorios de trabajo, el nombre y ubicación del archivo de "RULES.TXT", la definición de la conexión a la base de datos e información de carácter general.

   Además en esta clase se encuentra el método de validación, que define si el URL puede ser indexado o no, así como la definición del método que implementa el protocolo de exclusión de páginas, a través de la lectura e interpretación del archivo ROBOT.TXT del raíz del servidor Web.

4. **FIFOQueue.*** Clase que permite definir un tipo de objeto con un comportamiento de cola dinámica, la cual puede crecer de acuerdo a las necesidades que se presentan, a su vez optimizando el espacio no utilizado de la cola. Esta clase se emplea par definir la cola de URL's a descargar por el Crawler y la cola de archivos temporales listos para ser analizados por el Indexer.

5. **HTMLWordExtractor.*** Clase que contiene la mayoría de los métodos necesarios para extraer de un archivo de formato HTML las palabras a indexar, este archivo trabaja en estrecha relación con el WordExtractor.

6. **HTMLLinkExtractor.*** Archivo que define los métodos relacionados a la extracción y validación de los URLs encontrados dentro de una página de formato HTML, para posteriormente validarlos e introducirlos a la lista de URL's del Crawler.

7. **Indexer.*** Archivo principal del proceso de Indexer, construye la clase del mismo nombre e inicializa las variables a nivel de clase para la ejecución de este proceso. Recibe los URL's de los cuales el Crawler a logrado obtener las páginas Web, almacenándolas en archivos temporales, sobre los que se clasificarán las palabras de contenido y los enlaces que posean.

8. **LinkExtractor.*** Clase que permite definir un tipo de objeto genérico al momento de extraer los enlaces de las páginas Web, es decir, permite el empleo de la característica de polimorfismo para obtener los enlaces de los archivos con formato HTML y devolver valor nulo de los archivos que no son de este tipo y por lo tanto no pueden

tener enlaces, con esto se discrimina archivos tales como aplicaciones CGI o ejecutables en general.

9. **Monitor.*** Clase que construye la interfaz gráfica o monitor del robot Web, en la cual se presentan progresivamente los resultados de la ejecución del mismo.

10. **Mtto.*** Clase que define el proceso de mantenimiento de la información de la base de datos, con lo cual se logra convertir al robot en uno de tipo combinado. Con esta clase se obtiene un módulo capaz de mantener información actualizada y valida para el dominio que se ha diseñado

11. **NullLinkExtractor.*** Esta clase devuelve valor nulo para aquellos tipos de documentos que no contengan enlaces como por ejemplo los de extensión ".TXT"

12. **Rules.txt.*** Archivo de configuración que contiene un listado de los URL's que no deben de ser tomados en cuenta durante el proceso de escrutinio del Web.

13. **TextWordExtractor.*** Clase que define los métodos necesarios para extraer las palabras de contenido de los archivos de tipo ".TXT", con el propósito de introducirlas a la base de datos.

14. **URLStatus.*** En esta clase se encuentran definidos los métodos necesarios para trabajar con el protocolo HTTP y copiar la página Web localmente, así como reconocer si la página ha sido trasladada de sitio, adicionalmente se definen los métodos necesarios para introducir tanto los URL's que no se encuentren en la base previamente, como las palabras que contiene estos enlaces.

15. **WordExtractor.*** Este archivo hereda sus métodos al HTMLWordExtractor y al TextWordExtractor, siendo el encargado directamente de la extracción y ordenamiento de cada una de las palabras de las páginas Web en el archivo temporal, así como calcular el número de veces que la palabra se encuentra dentro de la página.

3.6.5. Funcionalidad del Motor de Búsqueda "Cyber Izalco"

Cyber Izalco constituye un excelente punto de partida para encontrar información publicada en el dominio de El Salvador. El URL temporal de este buscador es http://www.cyber-izalco.com.sv. La página inicial se muestra en la Ilustración 13.

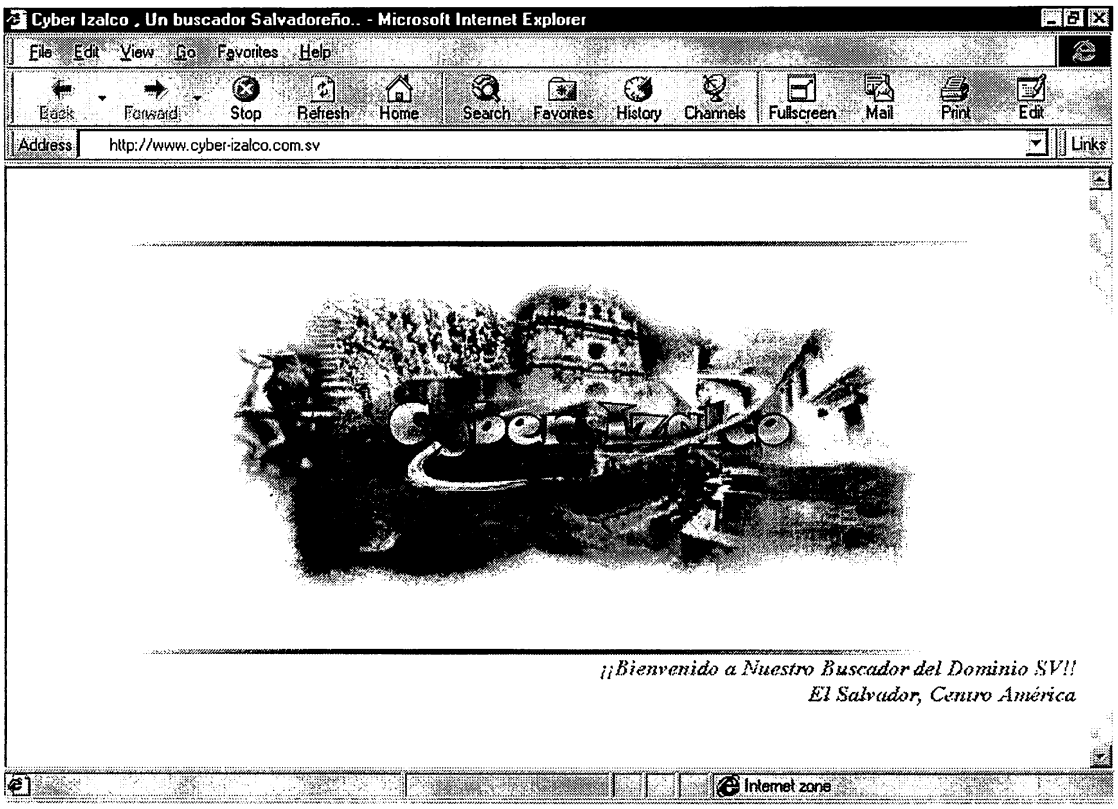


Ilustración 13. Página de Inicio

La interfaz presentada a los usuarios de Internet proporciona un menú por categorías dentro de las cuales se pueden encontrar clasificados todos aquellos enlaces que se han ingresado explícitamente a través del formulario de Cyber Izalco (Ilustración 19). Cada una de estas categorías contiene una serie de enlaces agrupados en subcategorías.

El menú de categorías se presenta en la siguiente ilustración:

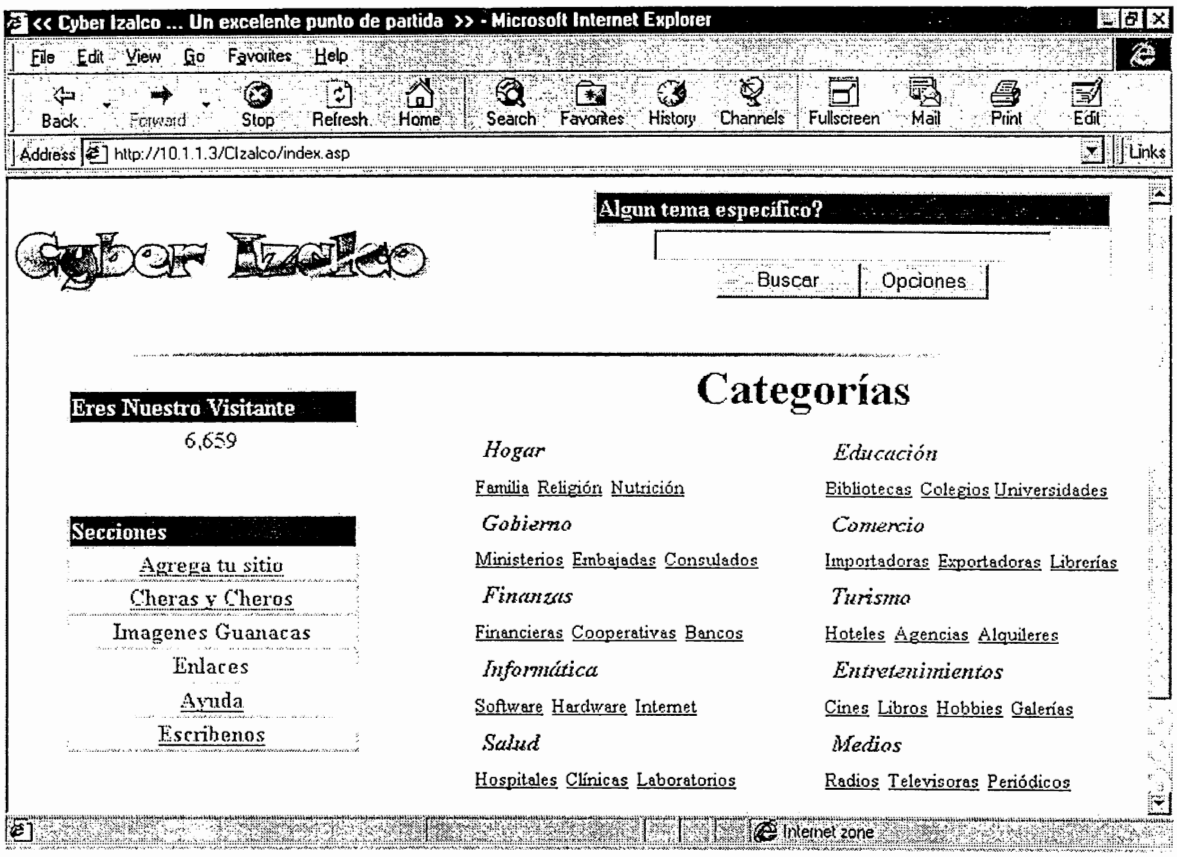


Ilustración 14. Menú de Categorías de Cyber Izalco

Las categorías existentes son:

1. Hogar: Familia, Religión, Nutrición, Fundaciones, Organizaciones.
2. Educación: Colegios, Universidades, Institutos, Bibliotecas.
3. Comercio: Importadoras, Exportadoras, Almacenes, Floristerías, Librerías, Restaurantes, Servicios, Fábricas, Maquilas.
4. Gobierno: Ministerios, Embajadas, Consulados, Superintendencias, Autónomas.
5. Finanzas: Financieras, Cooperativas, Bancos.
6. Salud: Hospitales, Clínicas, Laboratorios, Farmacias.
7. Informática: Software, Hardware, Internet, Tiendas, Publicaciones.
8. Medios: Radios, Televisoras, Periódicos, Revistas, Instituciones, Periodismo.
9. Turismo: Hoteles, Agencias, Cultura, Alquileres, Artesanías.
10. Entretenimiento: Cines, Libros, Revistas, Grupos, Deportes, Humor, Fotografías, Galerías.

La siguiente figura muestra los enlaces dentro de la categoría Educación:

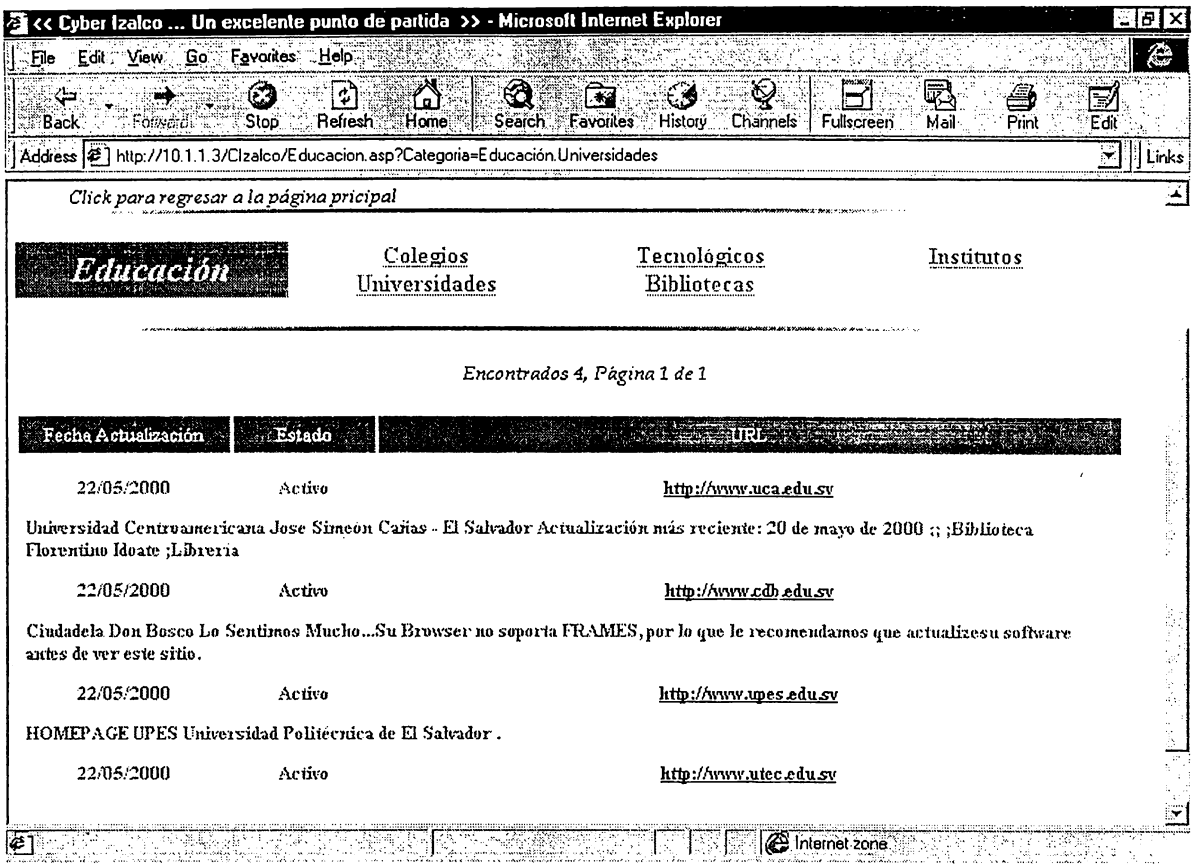


Ilustración 15. Ejemplo de Categoría de Educación

Además, la interfaz de búsqueda de Cyber Izalco incluye un formulario en el cuál se le especifican los criterios de búsqueda. Dicho formulario contiene también el botón **opciones**, el cual permite obtener ayuda sobre el funcionamiento y la búsqueda avanzada de Cyber Izalco.

La forma básica de solicitud de información consiste en utilizar como parámetro de consulta la palabra exacta indicada en la casilla de criterios, siendo esta en español o en inglés. La siguiente figura muestra un ejemplo de un criterio en inglés y sus resultados.

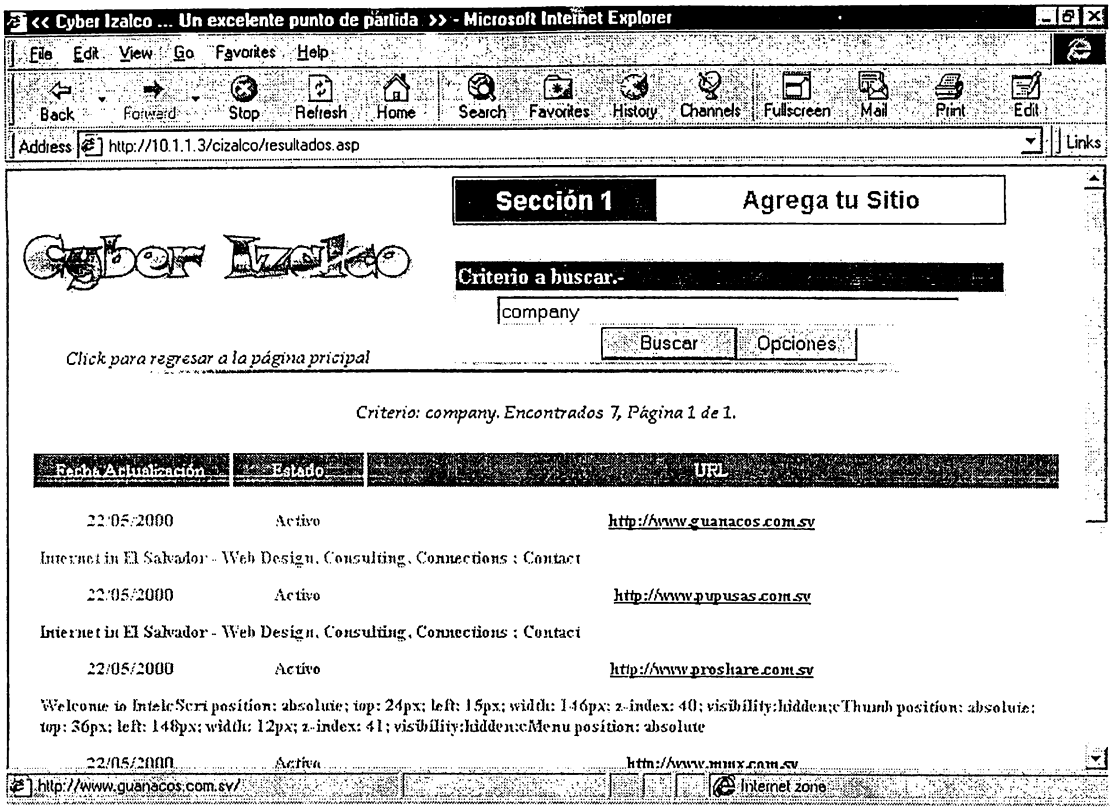


Ilustración 16. Criterios de Búsqueda en Inglés

La búsqueda de una frase es interpretada por el motor de *búsqueda* como consulta de aquellos enlaces que contienen alguna de las palabras dentro de la frase especificada (OR lógico). Suponiendo que se ingresa la frase *motores de búsqueda*, Cyber Izalco recuperará todos aquellos enlaces que contengan la palabra *motores* ó que contengan la palabra *búsqueda*. De esta forma quedan excluidas las palabras tales como adjetivos, preposiciones y otras, definidas previamente por el administrador de la base de datos de Cyber Izalco.

Resultados de búsqueda

La página de resultados muestra inicialmente el criterio a partir del cual ha realizado la búsqueda, luego se presentan los resultados ordenados en cuanto a relevancia.

Cyber Izalco muestra 10 enlaces por página y 10 páginas por grupo presentado, de manera que los enlaces desplegados al inicio de la página son generalmente aquellos con mayor cantidad de

información en cuanto a los criterios especificados. Cada enlace posee también la fecha de actualización dentro de la base de datos, es decir la fecha en la que se verificó la validez del URL; otro dato presentado es el estado de dicho URL (Activo ó Inactivo) y finalmente una descripción corta (250 caracteres) sobre la información que contiene dicho URL.

Búsqueda Avanzada

Cyber Izalco permite la búsqueda avanzada empleando operadores lógicos tales como OR y AND; además se agrega el uso del símbolo de porcentaje (%).

OR: La búsqueda de criterios que incluyan varias palabras separadas por espacios retornará los enlaces que contienen alguna ó todas esas palabras.

AND: Para consultar sobre aquellos documentos que deben contener información relacionada a varias palabras se emplea el símbolo **más (+)**. Por ejemplo, el criterio de búsqueda *empresa* + *informática* retornará todos aquellos enlaces cuyas páginas incluyan ambas palabras. La Ilustración 17 muestra los resultados de una búsqueda en donde se utilizan los operadores lógicos AND y OR.

%: Este símbolo es utilizado para la solicitud de información relacionada a palabras incompletas, es decir, se emplea como un comodín de búsqueda. Por ejemplo, si se ingresa como criterio la palabra *conver%*, los resultados presentados se obtienen basándose en todos aquellos sitios que contengan información con una palabra que inicie con *conver*, tal como *conversación, conversar, convergente, etc.* Dicho símbolo puede incluirse al principio, al final o en medio de una palabra. La Ilustración 18 muestra los resultados generados realizando una consulta con el empleo del carácter comodín.

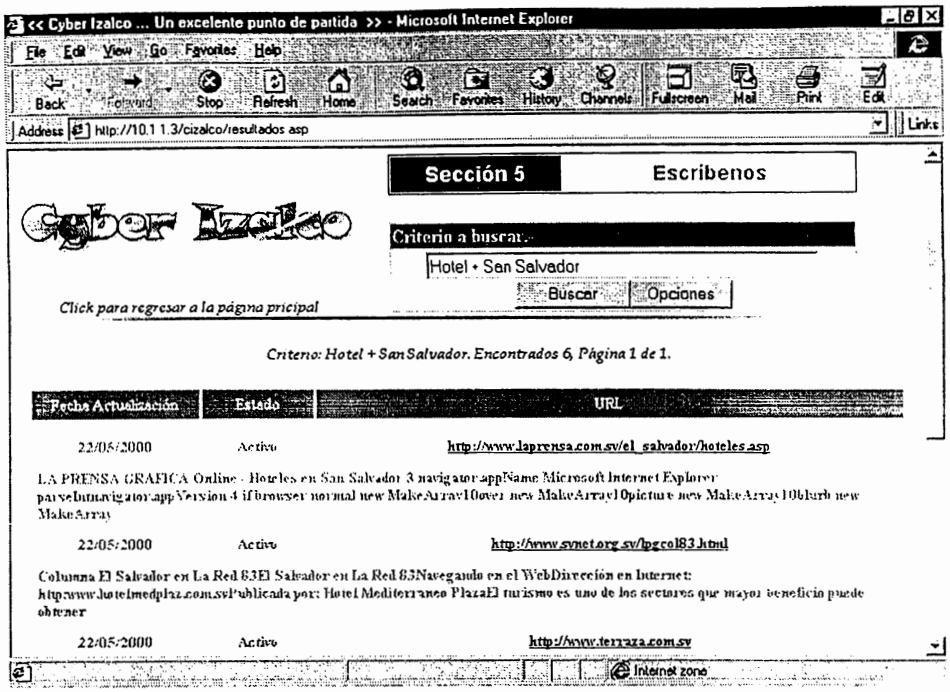


Ilustración 17. Empleo de operadores lógicos AND (+) y OR (Espacio)

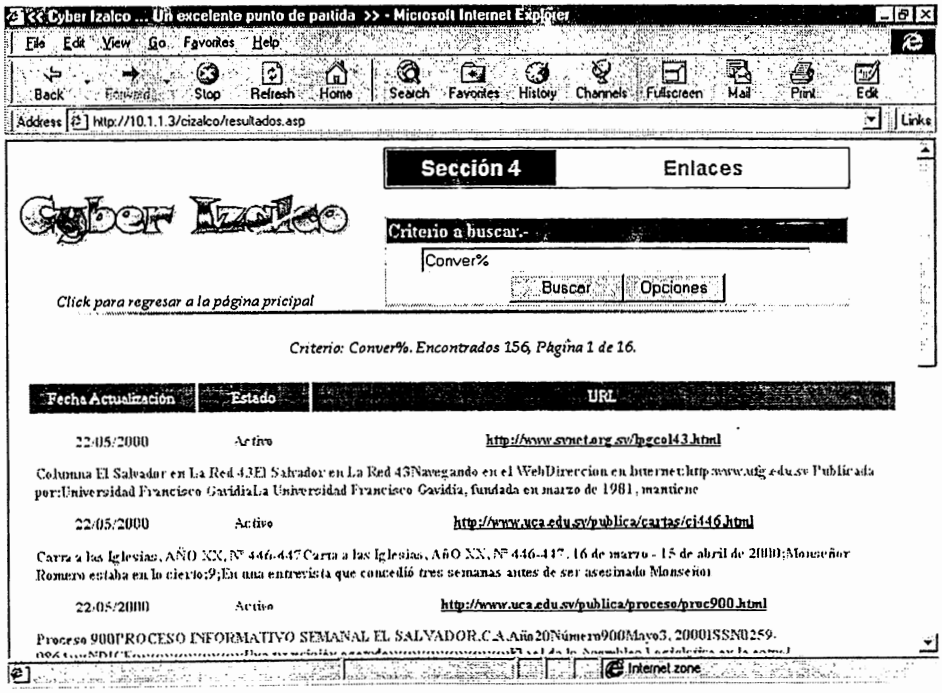


Ilustración 18. Empleo del símbolo comodín (%)

Consejos para la realización de Búsquedas

1. Formular consultas basándose en palabras claves sobre un tema específico.
2. Búsqueda a través de ideas y conceptos fundamentados en palabras claves empleando más de una palabra en la búsqueda.
3. Emplear la búsqueda de criterios Avanzada. Por ejemplo una búsqueda por *Izalco* retornará resultados mucho más específicos que una búsqueda de *Volcanes Salvadoreños*.

Secciones de Cyber Izalco

Agrega tu Sitio

Agrega tu Sitio es una de las secciones más interesantes de Cyber Izalco, ya que le permite a los usuarios de Internet adicionar un sitio Web que no necesariamente corresponden al dominio de El Salvador (SV), por medio de un formulario de captura de información, estos sitios pasan por una revisión previa por el administrador de la Base de Datos para hacer un chequeo de la información que contienen y calificarlos como válidos si el contenido del sitio es salvadoreño o de lo contrario serán eliminados de la Base de Datos. El formulario mediante el cual se agregan los datos es el siguiente:

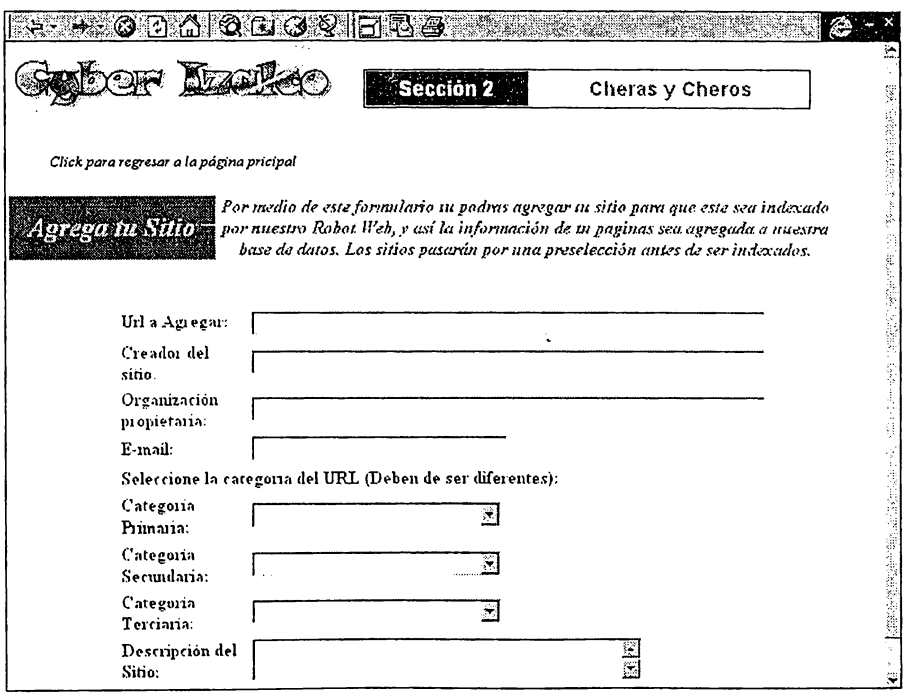


Ilustración 19. Formulario de Ingreso, Sección Agrega tu Sitio

Cada uno de los campos que se solicitan a través del formulario son validados, por ejemplo los campos que deben contener información de manera obligatoria para almacenar en la Base de Datos son: URL, nombre o creador del sitio, organización, email y las categorías; a las cuales hace referencia el contenido del sitio, los campos restantes son opcionales o dependen de que el usuario quiera ingresarlos.

En el caso de dejar en blanco el campo de creador del sitio debe llenarse el campo de organización o viceversa para que la información del formulario sea válida.

A continuación se presenta el formulario de ingreso con todos los datos que debe contener normalmente para que sean introducidos a la Base de Datos.

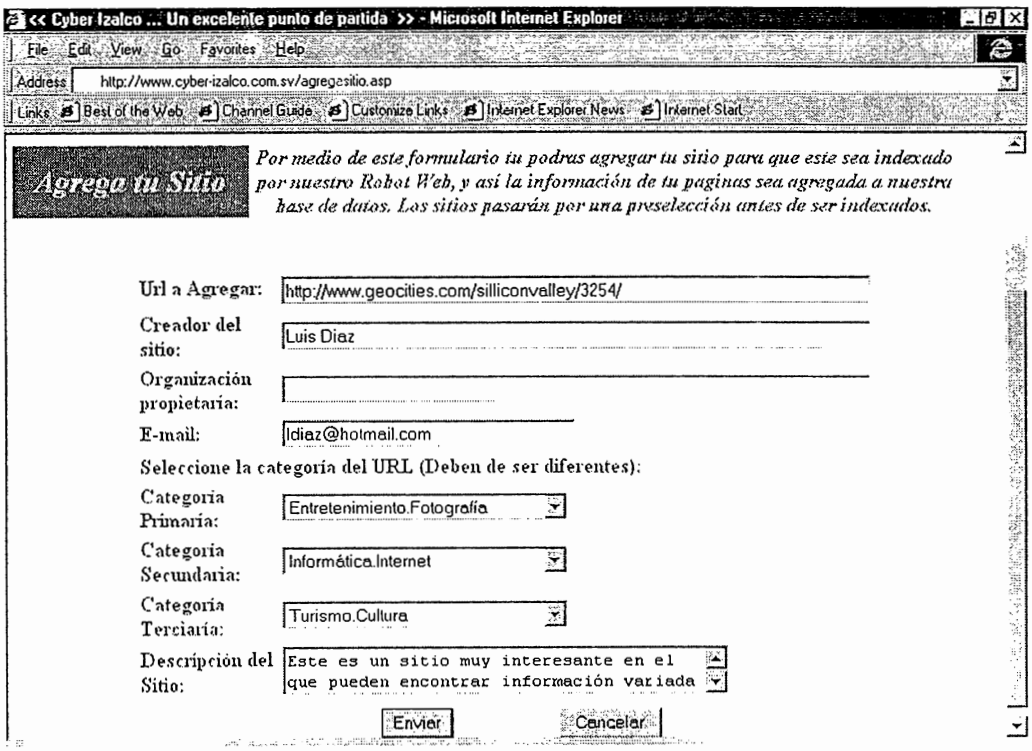


Ilustración 20. Ejemplo de Formulario de Ingreso, Sección Agrega tu Sitio

Cuando el usuario termine de llenar el formulario puede realizar dos acciones: si está seguro de ingresar el URL a la Base de Datos para que sea indexado posteriormente por el robot, debe presionar el botón de "Enviar" de lo contrario "Cancelar" para bloquear la operación.

Cuando el usuario presiona el botón de "Enviar" se obtiene la siguiente confirmación de los datos que se han almacenado en la base de datos de Cyber Izalco.

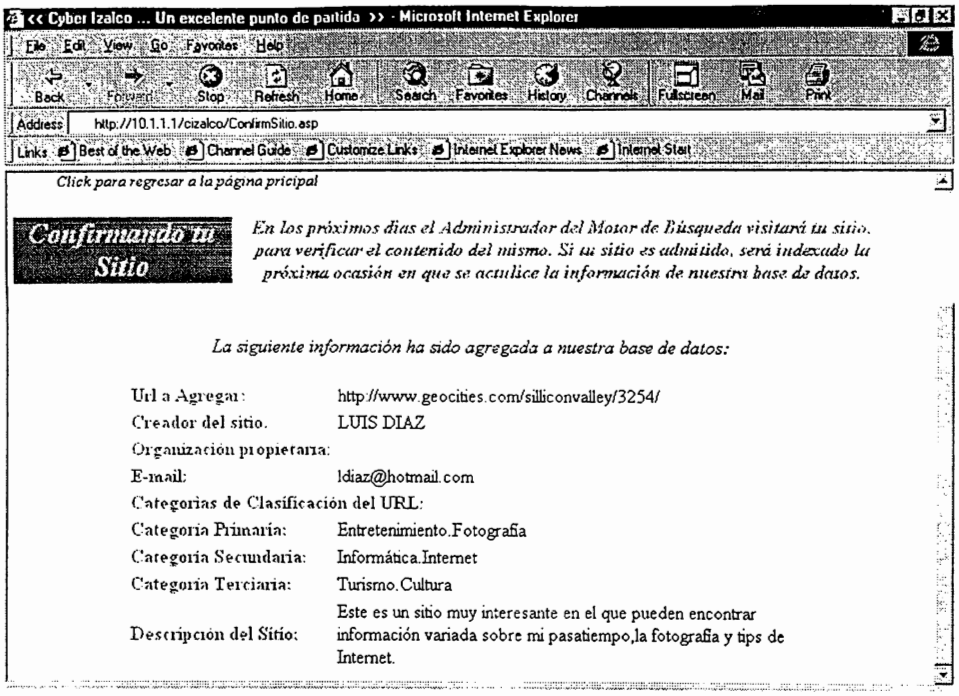


Ilustración 21. Confirmación de Ingreso de Información, Sección Agrega tu Sitio

La siguiente figura presenta la validación de los campos "Creador del Sitio", "Organización" y "Email", los cuales son obligatorios en el formulario, cabe mencionar que para los campos restantes del formulario que son indispensables se obtendrá el mismo resultado. Se muestra la confirmación de los errores encontrados en el formulario al validar los campos.

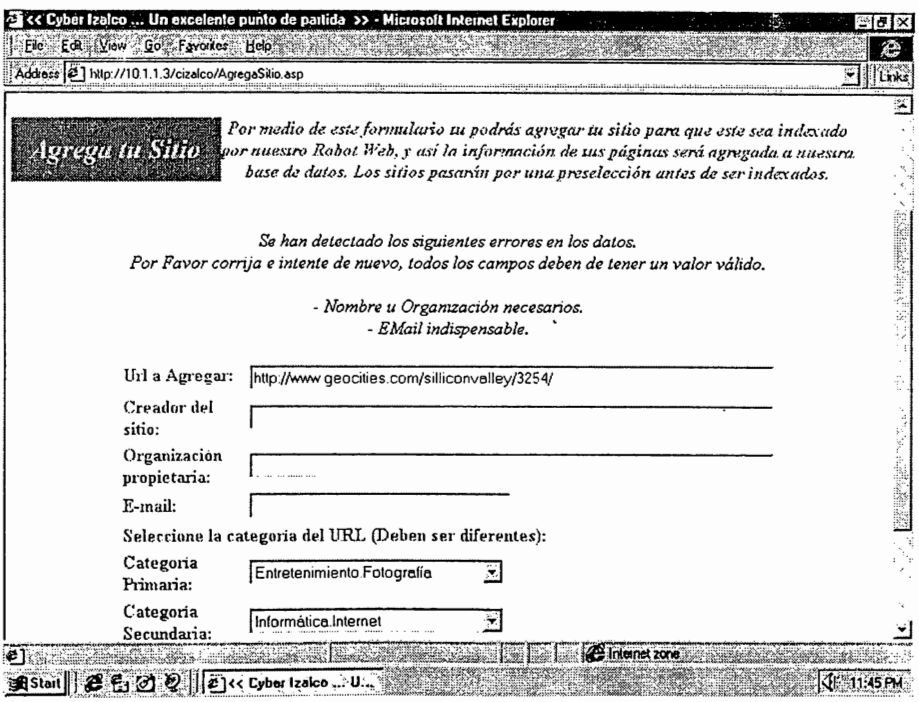


Ilustración 22. Ejemplo de errores de validación, Sección Agrega tu Sitio

Cheras y Cheros

En la sección de "Cheras y Cheros" se podrá consultar la información personal que otras personas han introducido a la base de datos. La base de datos se puede consultar de acuerdo a la letra inicial del apellido de la persona o por rangos de edades, como se muestra en la ilustración siguiente:

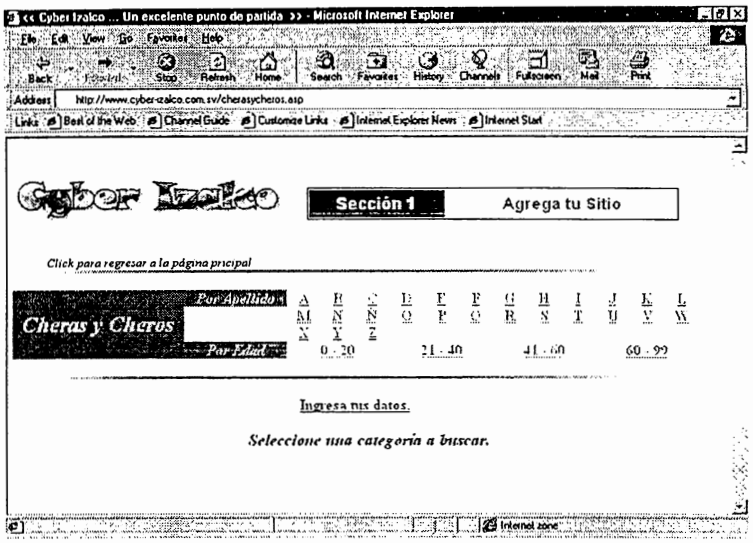


Ilustración 23. Sección Cheras y Cheros

Si al dar click sobre cualquiera de las letras o de los rangos de fechas, la consulta no encuentra ningún dato que cumpla este criterio, abajo aparecerá un mensaje indicándolo, en caso contrario los resultados de la consulta se presentan en forma de una tabla, permitiendo dar click sobre el nombre de la persona si se desea mandarle un correo; el formato de la hoja de resultados se muestra en la ilustración 24.

Para agregar información a la base de datos, se debe de acceder al formulario de ingreso, dando click sobre la leyenda "Ingresa tus Datos", de la hoja de consulta. Hecho esto aparecerá una página como la que se muestra en la ilustración 25.

En este formulario se deben de llenar los siguientes datos: nombres, apellidos, edad, país de origen, correo electrónico, ocupación y un comentario personal, finalmente para ingresar la información se debe de presiona el botón de "Ingresar" o "Cancelar" si ya no se desea continuar con el proceso.

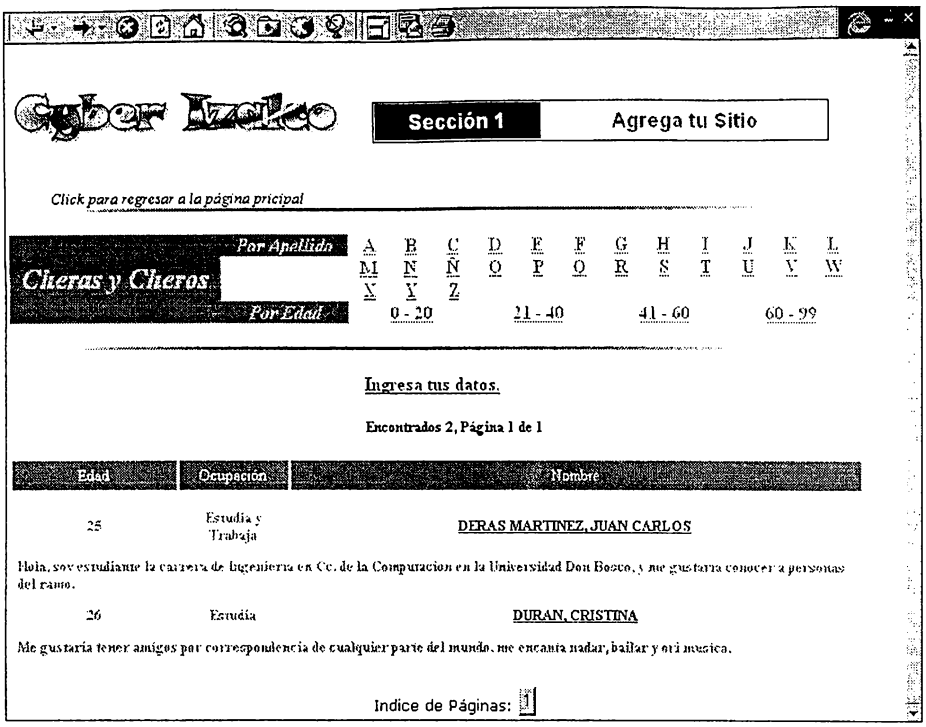


Ilustración 24. Consulta de Información, Sección Cheras y Cheros

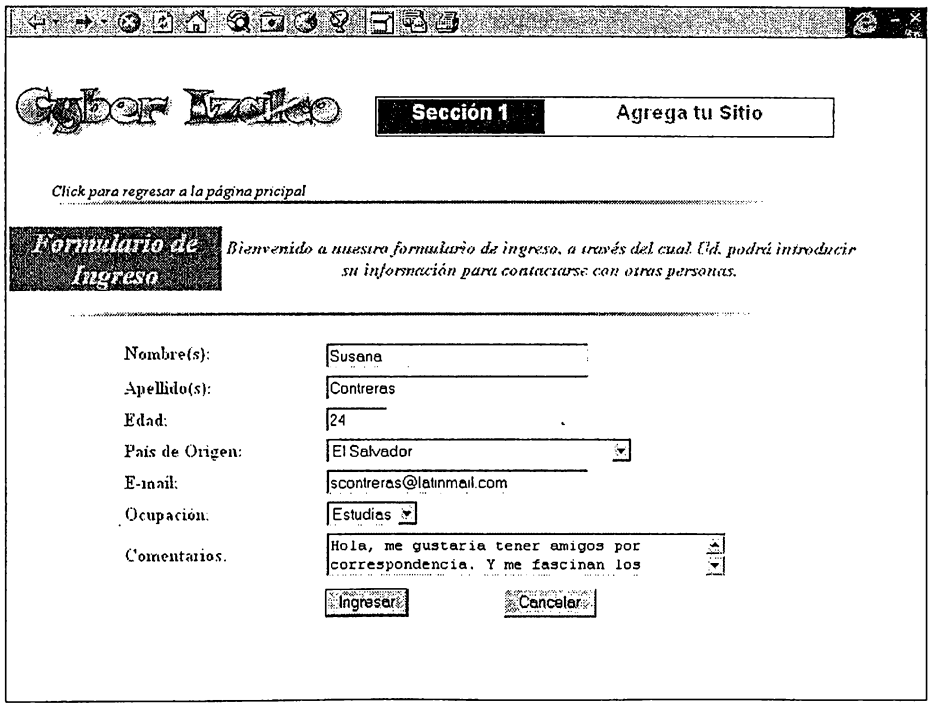


Ilustración 25. Ejemplo de Formulario de Ingreso, Sección Cheras y Cheros

Una vez ingresada la información en la base de datos aparecerá una hoja confirmando la información que ha sido introducida, tal como aparece en la ilustración siguiente:

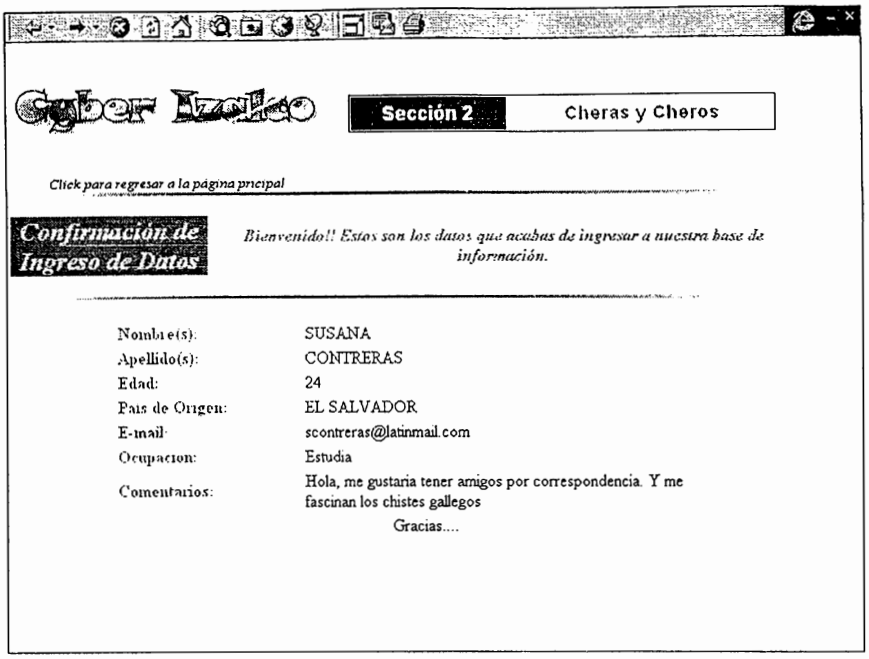


Ilustración 26. Ejemplo de Confirmación de Ingreso, Sección Cheras y Cheros

Galería de Imágenes

Esta interesante y divertida sección muestra una lista de imágenes de nuestro país, las cuales se encuentran clasificadas tal como se muestra en la ilustración:

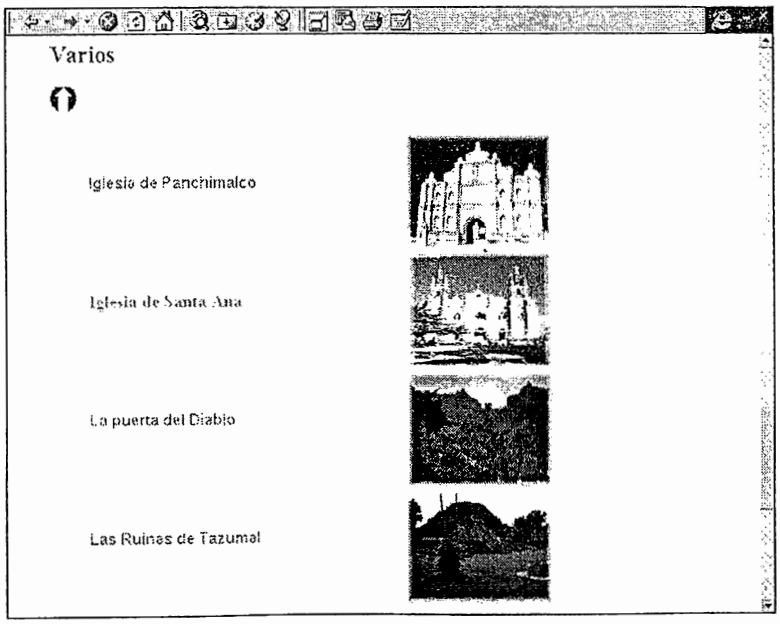


Ilustración 27. Sección de Imágenes Guanacas

Ayuda

Esta sección constituye una pequeña guía sobre el uso de la interfaz de búsqueda de Cyber Izalco. Aquí se incluyen la forma en la que se realizan consultas avanzadas, la manera en que deben interpretarse los resultados y se agregan consejos para la realización de búsquedas de manera óptima.

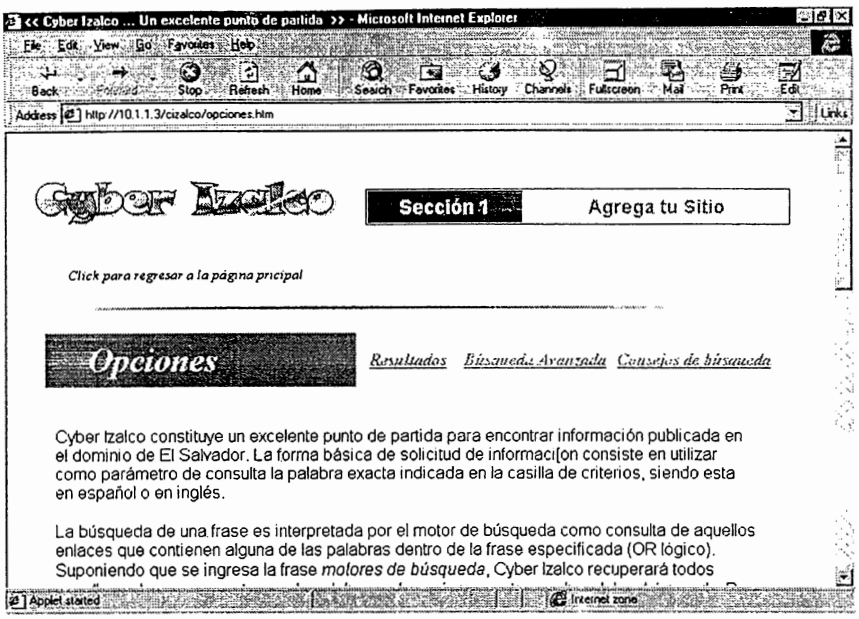


Ilustración 28. Sección de Ayuda.

CONCLUSIONES

1. La integración de bases de datos con el Web permite que usuarios desde múltiples plataformas accedan la información almacenada a través de interfaces.

2. Los robots empleados en los buscadores son segmentos de código que recorren todo o parte del Web alimentando una base de datos con la información que encuentran, facilitando la consulta posterior de datos.

3. El desarrollo de motores de búsqueda con alcance definido constituyen herramientas prácticas que permiten ahorrar recursos y facilitar la búsqueda de información en una región determinada.

4. Al delimitar el robot Web de Cyber Izalco para que indexe únicamente los sitios Web nacionales, se facilita la actualización con mayor frecuencia de la información existente en la base de datos, debido a que sólo se recorrerá una porción del Web.

5. Para el desarrollo de Cyber Izalco se emplea de software tales como: Windows NT, Microsoft Access, ASP y Java, los cuales representan un precedente en el desarrollo de buscadores Web, puesto que se emplean elementos que forman parte de las tendencias contemporáneas de la informática.

6. Cyber Izalco representa un medio efectivo de proyección de información nacional al mundo entero, lo cual contribuye al intercambio en áreas tales como cultural, científica, académica y económica con otras naciones.

7. Cyber Izalco es una herramienta que posee los mecanismos necesarios para facilitar las necesidades de búsqueda de Sitios Web nacionales, contando con una interfaz amigable y atractiva al usuario de Internet.

8. Cyber Izalco sienta las bases fundamentales que posibilitan el desarrollo de portales a nivel regional, ya que el legado de este proyecto es fácilmente reutilizable para estos fines.

9. Este proyecto demuestra que es posible desarrollar aplicaciones Web a corto y mediano plazo a partir de la infraestructura existente en nuestro medio.

10. El empleo de Java como lenguaje de programación Web garantiza la portabilidad de la aplicación desarrollada entre diferentes plataformas.

11. Cyber Izalco es un motor de búsqueda fácilmente adaptable a recorrer otros dominios tanto a nivel de Intranet como Internet; además de ser flexible a otros idiomas diferentes al español.

RECOMENDACIONES

1. Mantener un seguimiento constante sobre los sitios agregados desde el Web, de tal modo que la información almacenada a la Base de Datos permanezca íntegra.

2. A efectos de que diversos sitios Web a nivel nacional mantengan la privacidad requerida en sus páginas, cuando sean visitadas por algún robot Web, es necesario fomentar el uso de protocolos de exclusión de robot.

3. Efectuar procesos de compactación periódicos sobre la Base de Datos con el objeto de eliminar los almacenamiento temporales existentes.

4. Si se desea emplear Cyber Izalco con otros servidores Web distintos al Internet Information Server, únicamente es necesario modificar lo referente a las interfaces Web-Base de Datos.

5. El robot Web empleado en Cyber Izalco está debidamente probado en un dominio específico y en una Intranet, por lo cual no debe intentar ejecutarse sin límite alguno pues sus resultados son impredecibles. Para esto es necesario un estudio detallado del robot a efectos de extenderlo sin caer en búsquedas infinitas.

6. Si el dominio a indexar es bastante grande se recomienda el empleo de una base relacional de mayor envergadura como lo son Oracle, Sybase o SQL Server, ya que el manejo de millones de registros es óptimo en estas plataformas; lo cual no representaría modificación sustancial en el motor de búsqueda desarrollado.

7. Se recomienda ejecutar el robot en períodos de bajo tráfico en el Web como podrían ser horarios nocturnos o fines de semana, además se propone como período de ejecución un intervalo de tres semanas.

8. Una alternativa para el procesos de indexamiento lo constituye el módulo de Mantenimiento del robot, para verificar la validez de los URL's de la Base de Datos permitiendo ejecutarlo con mayor frecuencia que el robot.

BIBLIOGRAFIA

- Lemay, Laura, ***Aprendiendo HTML para Web en una semana,*** Prentice Hall, 1994.
- Bonilla Castañeda, Diego, ***Mercadotecnia e Imagen en Internet***, Grupo Editorial Iberoamérica, 1996.
- Microsoft Corporation, ***Microsoft Internet Information Server, Installation and Administration Guide***, CD de instalación de NT v. 4.0.
- Evans, Tim, ***Construya su propia Intranet***, Prentice Hall, 1997.
- Gralla, Preston, ***Como Funciona Internet,*** Prentice Hall, 1996.
- Larson, Michael A., ***Aprendiendo a Publicar en Web, con Microsoft Office 97,*** Prentice Hall, 1997.
- Ferreyra, Gonzalo, ***Internet Paso a Paso, Hacia la autopista de la Información,*** McGraw Hill, 1996
- Córtez, Carlos; Tobar, Mercedes; Meléndez Jaime, ***Interfaz CGI para Servidores Web y Sistemas de Administración de Bases de Datos***, Tesis UCA, 1997.
- Vides Mendoza, David, **Diseño e *Implementación de un Buscador de Sitios Web para el dominio de El Salvador***, Tésis UCA, 1998.
- Jepson, Brian, ***World Wide Web Database Programing for Windows NT***, Wiley Computer Publishing, 1996.
- Lemay, Laura; Perkins, Charles L, ***Aprendiendo Java en 21 Días***, Prentice Hall, 1996.
- Cruzado Nuño, Ignacio, ***Robot Recuperador Web Spider Bot 1.0***, Tésis Universidad de Burgos, 1999.
- Eckel, Bruce, ***Thinking in Java***, Prentice Hall, 1998.
- ***WWW Robots, Wanderers and Spiders***, http://info.webcrawler.com/mak/proyects/robots/robots.html

- *Herramientas de recuperación de páginas Web,*
  http://ai.bpa.arizona.edu/demo/spider.html
- *Organización que creó el World Wide Web. RFC y HTML.*
  http://www.w3.org
- *Sitio Web de información referente a Perl,*
  http://www.perl.com
- *Perl Search Tools Product Listings,*
  http://www.searchtools.com/tools/tools-perl.html
- *Java Search Tools Product Listings,*
  http://www.searchtools.com/tools/tools-java.html
- *Windows Search Tools Product Listings,*
  http://www.searchtools.com/tools/tools-win.html
- *Análisis del rendimiento del HTTP,*
  http://www.isi.edu/lsam/ib/http.perf
- *ScriptSearch a CGI Library,*
  http://scriptsearch.com/

ANEXO A.

SALNET S.A.

De entre las compañías a las cuales se consultó sobre el interés para el desarrollo de este proyecto, SALNET S.A. fue elegida puesto que además de ser uno de los proveedores de servicios de internet mejor colocados en el medio nacional, por la calidad de los servicios que proveen, brindaba las mejores condiciones para la implementación del motor de búsqueda.

SALNET, es una compañía 100 % Salvadoreña, que inició sus operaciones en 1996, ofreciendo una variedad de servicios de comunicaciones, con equipo de fibra óptica, terminales digitales y personal técnico altamente calificado.

Sus planes de servicios ofrecen buenas alternativas de conexión a Internet, ya sea para uso personal o empresarial bajo sistema conmutado o enlace dedicado.

Así también ofrecen servicio de telefonía internacional a través del carrier 147, para llamadas por cobrar o directas, empleando tarjetas de prepago, centros de llamada internacionales ubicadas en el interior del país.

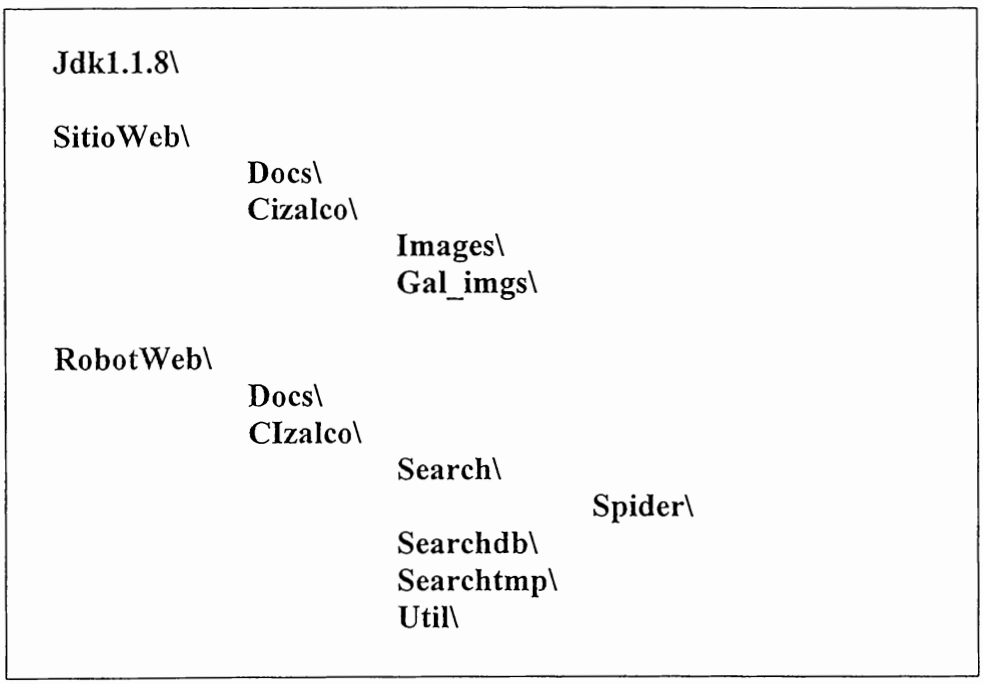
ANEXO B.

Manual Administrativo del Motor de Búsqueda

Sección I. Robot Web

CD de Instalación

Adjunto a este documento se presenta un CD conteniendo todo el software y los archivos necesarios para llevar a cabo la instalación, configuración y ejecución del motor de búsqueda. El árbol de directorios de este CD se presenta a continuación:



Contenido de los directorios:

Jdk1.1.8 .- Contiene el instalador del Java Developer Kit v. 1.1.8, versión con la cual ha sido desarrollado y modificado el Robot Web.

SitioWeb .- Este directorio consta de los sub-directorios Docs y CIzalco.

Docs.- Contiene los archivos de ayuda referentes al sitio Web.

CIzalco.- Contiene las páginas web, las clases de los applets y un archivo de inicialización global del robot. Este directorio contiene un sub-directorio Images en el cual se encuentran todas las imágenes empleadas por el robot.

RobotWeb .- Consta de los sub-directorios Docs y CIzalco.

Docs.- Contiene los archivos de ayuda correspondientes a la forma de operar el robot.

CIzalco.- Este directorio consta de los archivos de ejecución del robot. Contiene los sub-directorios: Search, Spider, Searchdb, Searchtmp y Util. Estos sub-directorios poseen los archivos esenciales para el funcionamiento del robot, cada uno de ellos se explica detalladamente en la descripción general de Cyber Izalco.

Searchdb.- Contiene el archivo Rules.txt con los sitios que se excluirán de la búsqueda.

Searchtmp.- Es el directorio donde se crean archivos temporales de las páginas que el robot baja para indexarlas.

Requerimientos

Los siguientes puntos se deben de cumplir para llevar a cabo la instalación y ejecución del Robot del Cyber Izalco:

1. La máquina sobre la cual se ejecutará el Robot Web debe de cumplir con las siguientes características:
   Procesador: Pentium o superior
   Velocidad del Procesador : 200 MHz o superior
   RAM: 64 Mb.
   Espacio Libre en Disco: 250Mb o más.
   Sistema Operativo: Cualquier plataforma compatible con JAVA.

2. La máquina sobre la cual se instale el robot debe de tener acceso libre a internet por medio de una dirección pública, válida en internet, de preferencia a través de un nodo dedicado, con el fin de optimizar la velocidad de obtención de las páginas, por parte del Robot.

3. Para poder compilar y ejecutar el Robot Web es necesario haber instalado previamente el Java Developer Kit, al menos en su versión jdk1.1.8, este kit se puede obtener del sitio http://java.sun.com/products/jdk/ totalmente gratis o del CD que acompaña a este documento en el directorio: "Jdk1.1.8\".

4. La máquina debe poseer un manejador de base de datos como por ejemplo Microsoft Access o una Base de Datos que soporte los controladores ODBC para facilitar la administración de la Base de Datos.

Instalación y Configuración

Para la instalación de los archivos del Robot, es necesario copiar la carpeta del CD "RobotWeb\CIzalco" con todo su contenido, a una carpeta llamada "CIzalco" en una unidad disponible del disco duro. Es necesario respetar el nombre de esta carpeta para la correcta ubicación de los archivos, por parte del robot.

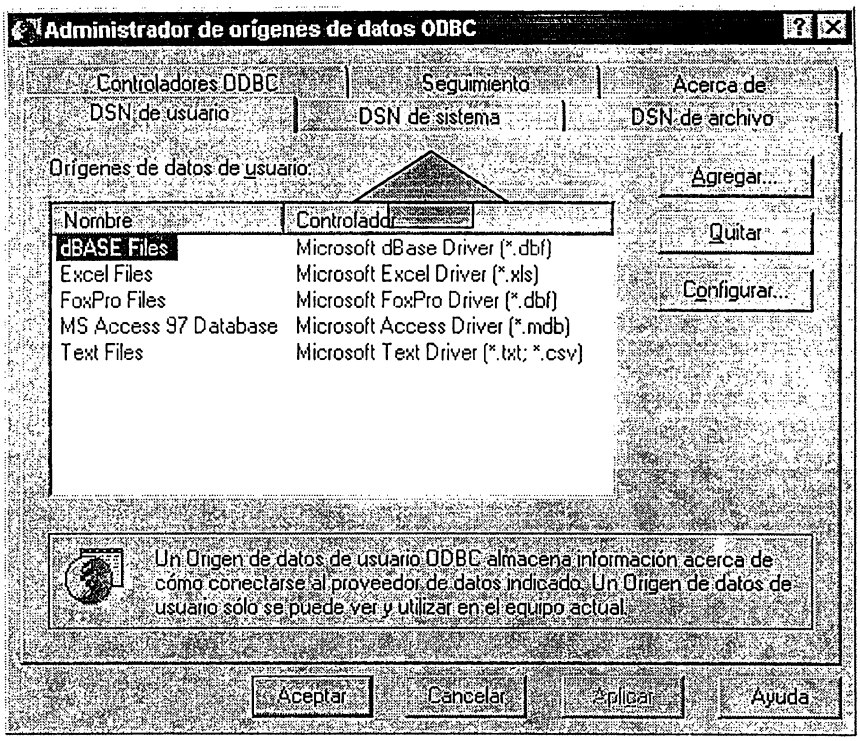
Configuración del ODBC

Con el fin que el robot pueda acceder a la Base de Datos es necesario crear un controlador ODBC, llamado "buscador", para lo cual se deben seguir los siguientes pasos; en el caso de instalar el robot sobre una máquina con Windows 95:

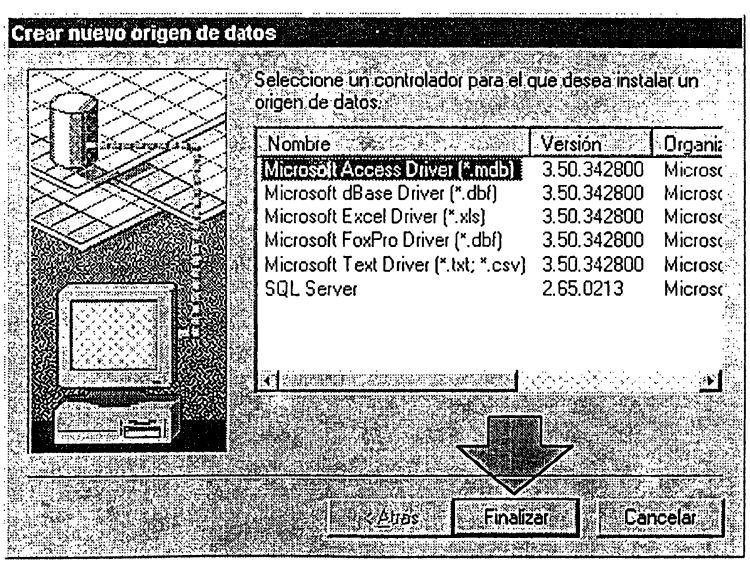
1. Del Panel de Control, ejecutar el programa de configuración de los controladores ODBC:

ODBC de 32 bits

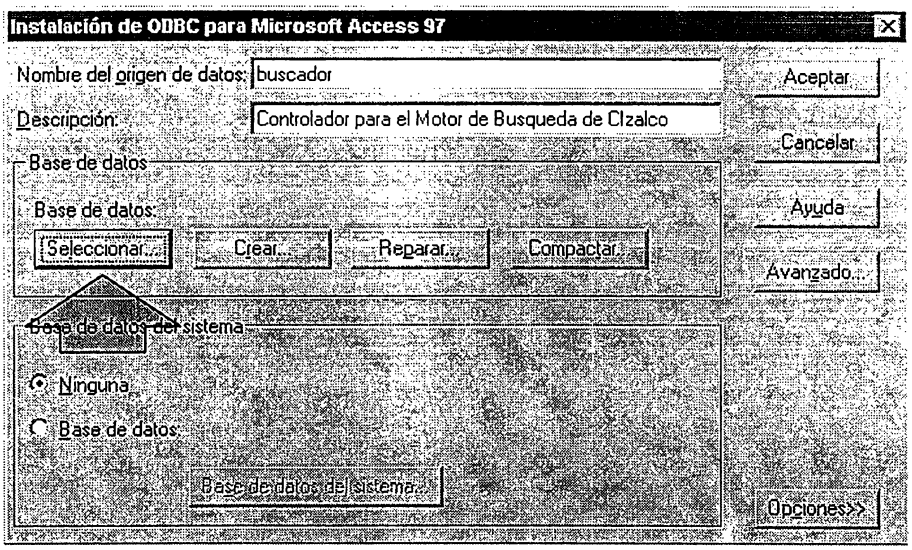
2. Seguido lo cual aparecerá la ventana de configuración, de esta seleccionar el tab de DSN de Sistema o System DNS.



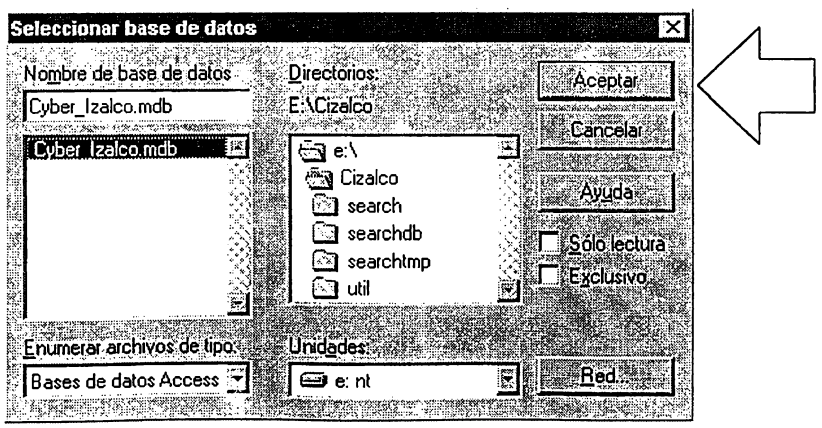
3. Una vez en el tab de DSN de Sistema, presione el botón de Agregar para que aparezca la ventana de selección de los controladores a emplear, seleccione los controladores de Microsoft Access y presione el botón de finalizar.



4. Seleccionado el controlador aparecerá la ventana de configuración del mismo, especifique en el campo de "Nombre de origen de datos" el nombre del controlador el cual debe ser "buscador"; tal y como aparece abajo, de igual forma en el campo de "Descripción" puede detallar un mensaje similar al que aparece. Hecho esto debe presionar el botón de Seleccionar.



5. Al presionar el botón Seleccionar le aparecerá una ventana similar a la siguiente, para especificar la base de datos a la cual este controlador hará referencia, la cual podrá encontrar dentro del directorio de "CIzalco" del Robot Web. Presione el Botón de Aceptar, para pasar a la ventana de configuración anterior y finalmente presione Aceptar de nuevo para terminar la creación del controlador.



Configuración del Robot

Antes de correr el Robot del Cyber Izalco por primera vez, se deben establecer ciertos parámetros, con el propósito de personalizarlo y especificar los datos necesarios para el correcto funcionamiento del mismo.

El archivo que se debe de actualizar es el EnginePrefs.Java, que se encuentra en la ruta "CIZALCO\SEARCH\", personalizando los siguientes datos:

1. El correo del administrador y responsable del Robot, este correo electrónico se deja en el registro de cada servidor Web visitado por el robot, con el propósito de que el WebMaster de ese equipo sea capaz de establecer contacto con el administrador del robot si lo requiere.

   ```
   String email_address = "nobody@nowhere.edu";
   ```

2. Período de tiempo en segundos que separará el acceso a diversos URL's, tanto para el Crawler como el Indexer, este período de tiempo es muy importante ya que evita la saturación de los servidores web visitados por el robot. Se aconseja que el periodo de tiempo especificado no disminuya de 5 segundos.

   ```
   public int pause_time = 5;
   ```

Las siguientes variables de la clase EnginePrefs son opcionales de actualización, si es que se necesita mover los directorios del robot hacia sitios diferentes, sin embargo se recomienda emplear los parámetros predefinidos.

1. Directorio donde se almacena el archivo de reglas de exclusión de URLs:

   ```
   File main_dir = new File("cizalco/searchdb");
   ```

2. Nombre del archivo que contiene las reglas de exclusión:

   ```
   File rules = new File(main_dir, "rules.txt");
   ```

3. Directorio de trabajo en donde se crean los archivos temporales correspondientes a cada una de las páginas web que el robot logra obtener:

```
File working_dir = new File("CIzalco/searchtmp");
```

Las instrucciones que limitan al Robot Web para que recorra solamente las páginas del dominio SV, es decir el dominio de El Salvador, se encuentran en el archivo EnginePrefs.java, ubicado en la ruta "CIzalco\Search\"; de tal modo que si se desea recorrer e indexar páginas de otro dominio, únicamente se debe sustituir el parámetro ".sv" por el dominio de interés. El dominio a indexar debe de escribirse en minúsculas en las instrucciones siguientes:

Línea 103.

"...endsWith(".sv"))"

Línea 113.

"....endsWith(".sv")) return true;"

Una vez realizados todos los cambios pertinentes a la clase, es necesario compilarla de nuevo para poder ejecutar el robot con los nuevos datos.

Una vez verificadas estas condiciones se procede a ejecutar la siguiente instrucción, asumiendo "X" como la unidad donde se almacena el robot, es necesario aclarar que JAVA hace distinción entre mayúsculas y minúsculas, por lo que se necesita que el archivo *.java se escriba tal y como esta en la siguiente instrucción:

X:\javac cizalco\search\EnginePrefs.java

javac es el programa compilador de java.

cizalco\search\EnginePrefs.java corresponden a la ruta y al archivo a compilar.

Ejecución del Robot

Ya que de la frecuencia de ejecución del robot depende la actualización de la información almacenada en la base de datos, esta debe de realizarse de acuerdo a los requerimientos establecidos por el Administrador del Robot, basándose en los siguientes criterios:

1. La cantidad de páginas a indexar, aproximadamente.
2. La carga que el indexamiento significa para los servidores web que se visitan.
3. La necesidad de información actualizada en lo referente al contenido de las páginas.

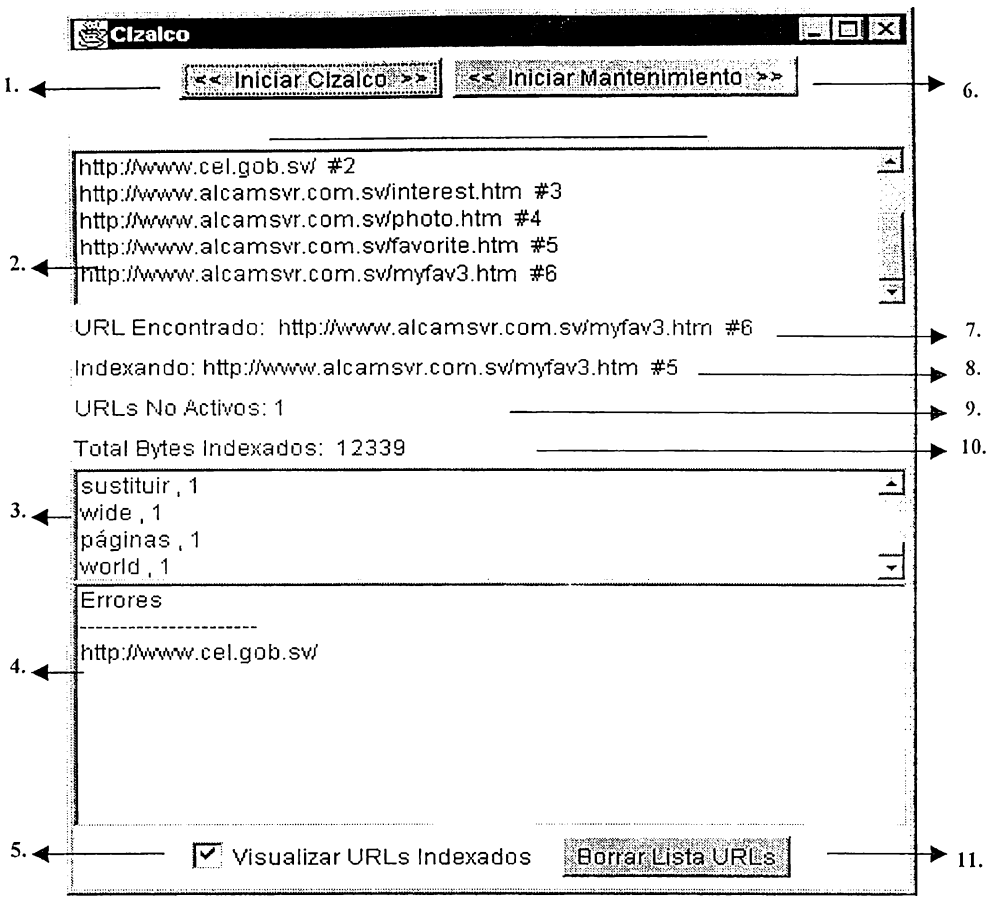
De igual forma es necesario considerar la posibilidad de ejecutar el proceso de indexamiento de páginas, en periodos de no muy alta demanda para los servidores Web, como por ejemplo por la noche o en fines de semana.

Una alternativa al indexamiento de páginas, lo constituye el proceso de mantenimiento, el cual exige menor demanda a los servidores Web, ya que solamente consulta si la página se encuentra activa o no y actualiza con esta información el contenido de la base de datos.

La instrucción para ejecutar el monitor del Robot es:

```
X:\java cizalco.search.Monitor
```

Al ejecutar esta instrucción se cargará la ventana del Monitor del Robot de Indexamiento, en la que podemos identificar los siguientes elementos:



1. Botón de ejecución del Crawler, este botón permite iniciar el indexamiento d los sitios web, necesario e indispensable para cargar información por primer vez en la base de datos.

2. Listado de URL's encontrados dentro del dominio SV y de los sitios agregado desde el Web por medio del formulario de ingreso de datos.

3. Lista de las palabras ingresadas a la base de datos.

4. URL's que por una u otra razón han dado error al intentar obtener sus páginas.

5. Caja de Chequeo, que permite especificar si se desea que aparezcan los URL y demás datos en pantalla

6. Botón de ejecución del módulo de mantenimiento de los URL's ingresados a la base de datos.

7. URL junto con su número secuencial correspondiente, encontrado en la página más recientemente indexada y que cumple con los criterios para ser indexado también.

8. URL o dirección que actualmente el robot está indexando.

9. Número de URLs inactivos, porque ya no existen o simplemente son inaccesibles en ese momento, lo cual genera un error al intentar bajar las páginas.

10. Sumatoria del total de bytes indexados por el robot, presentados en unidades de bytes.

11. Botón que limpia los datos presentados en el monitor del Robot, tanto de las listas como de los datos puntuales, únicamente se refiere a la visualización de información, en ningún momento representa una alteración a los datos previamente procesados.

Sección II. Base de Datos

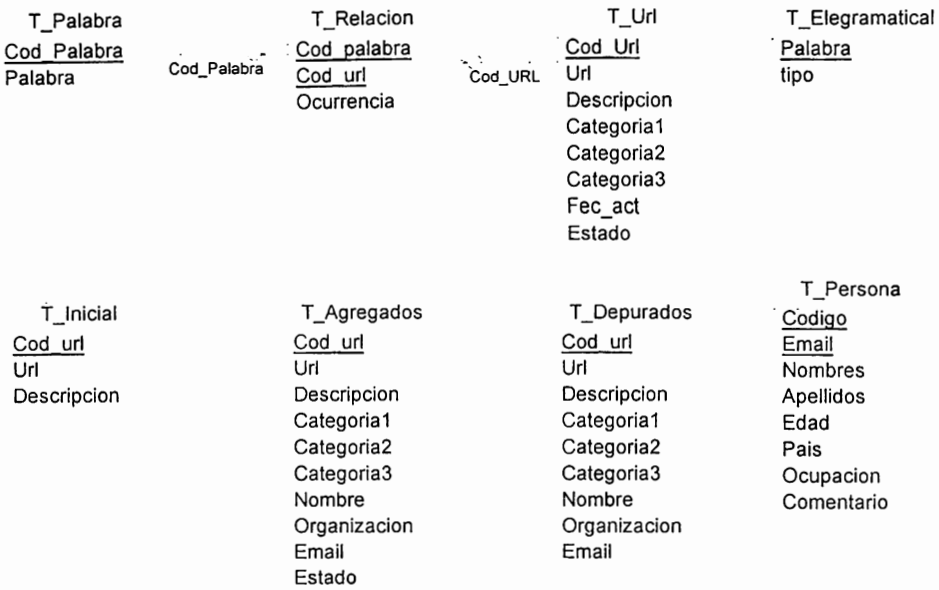
Como se ha mencionado la Base de Datos que emplea Cyber Izalco tiene como plataforma Microsoft Access versión 7.0.

En este manual se describe en detalle cada uno de los elementos que constituyen el sistema de Base de Datos, como lo son: tablas, consultas, formularios y macros. Lo cual facilitará la tarea de administración al personal encargado de brindarle mantenimiento a la Base de Datos.

Tablas

La Base de Datos que almacena información de aquellas páginas Web que han sido visitadas por el robot o agregadas explícitamente, constan de las siguientes tablas:

1. T_Agregados
2. T_Depurados
3. T_Elegramatical
4. T_Inicial
5. T_Palabra
6. T_Persona
7. T_Relacion
8. T_Url

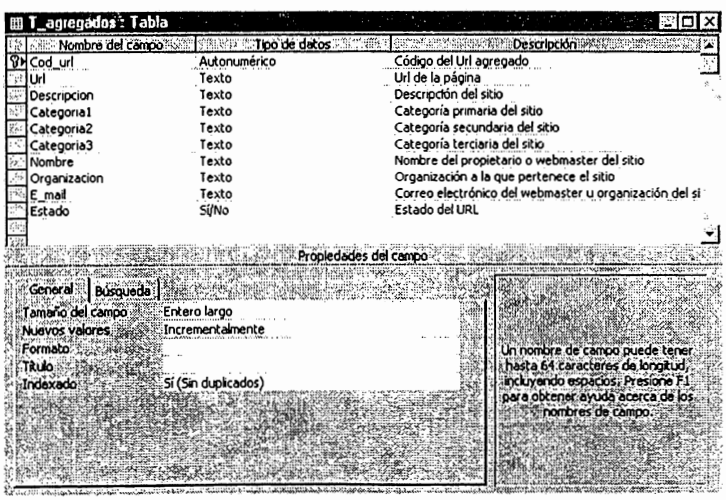


Modelo Entidad Relación Base de Datos

Descripción de Tablas

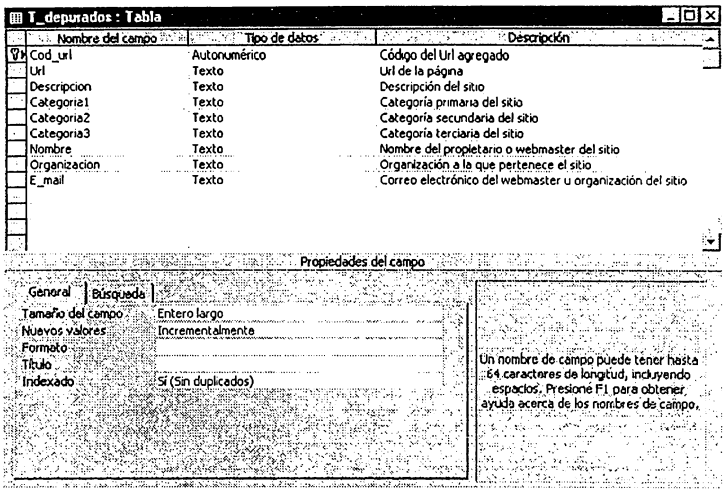
T_Agregados

Esta tabla contiene los URL's introducidos mediante el formulario de captura de datos, presentado a los usuarios de Internet. El contenido de estos URL's debe ser de carácter nacional para que el administrador de la Base de Datos pueda considerarlos válidos.



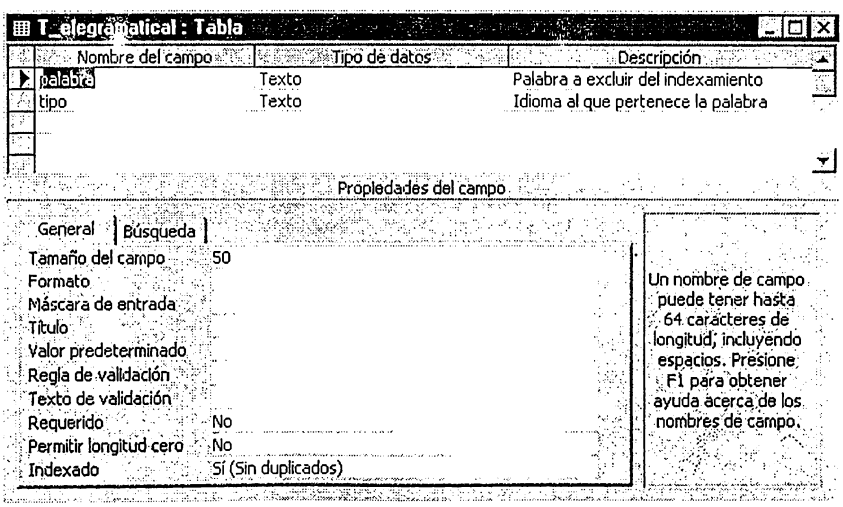
T_Depurados

Almacena los URL's filtrados de la tabla T_Agregados que el administrador determina que pueden ser indexados por el robot.



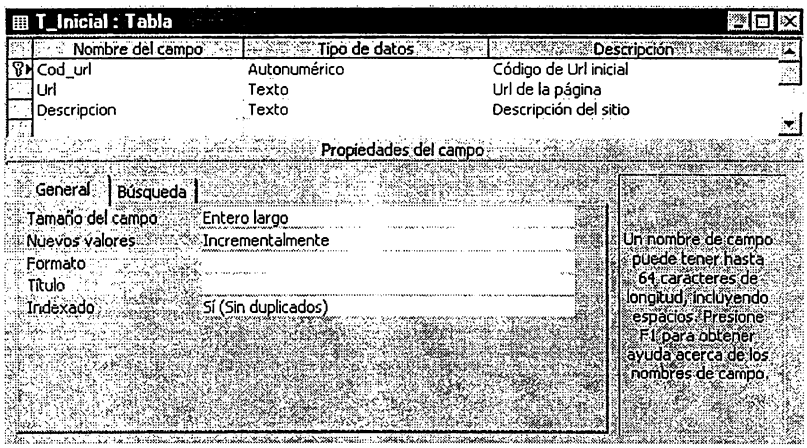
T_Elegramatical

Almacena todos los elementos gramaticales que el robot obviará cuando obtenga las palabras de una página web determinada. Cabe mencionar que para que el robot sea portable a otro idioma, basta con reemplazar los elementos gramaticales de esta tabla.



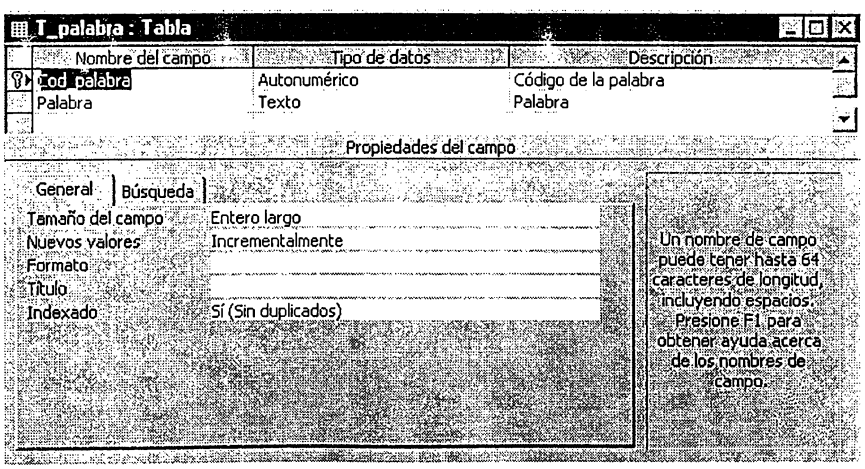
T_Inicial

Contiene los URL's con los cuales el robot iniciará el recorrido del dominio SV. Estos URL's pueden ser modificados por el administrador de la Base de Datos para considerar aquellos que según su criterio, sean adecuados para que el robot empiece el recorrido en el Web.



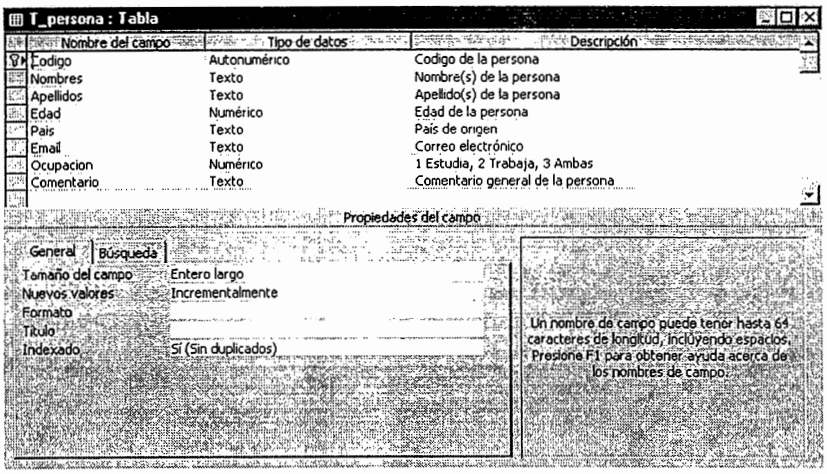
T_Palabra

Almacena la información relacionada a las palabras encontradas en las páginas Web visitadas por el robot. Para almacenar las palabras en esta tabla, se considera la tabla de los elementos gramaticales, que como se ha explicado, contiene palabras que el robot obvia en su búsqueda.



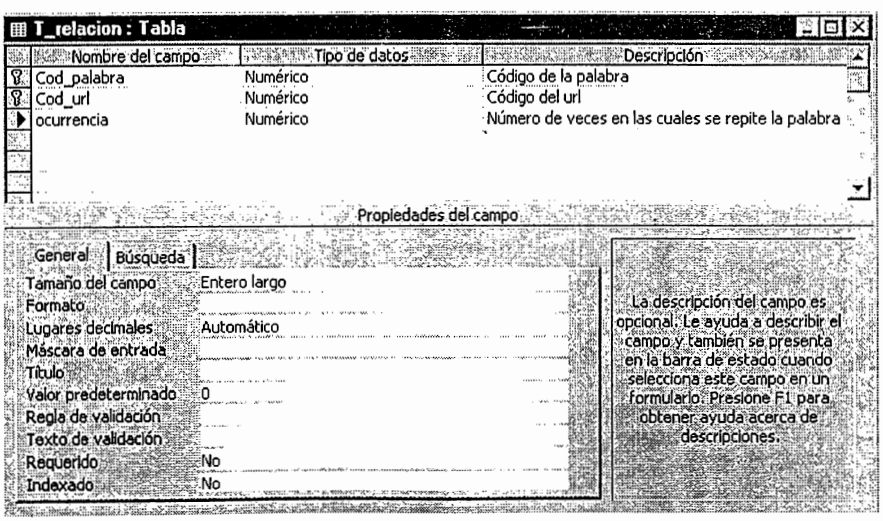
T_Persona

Almacena los datos de las personas que deseen dar a conocer su cuenta de correo electrónicos e información personal.



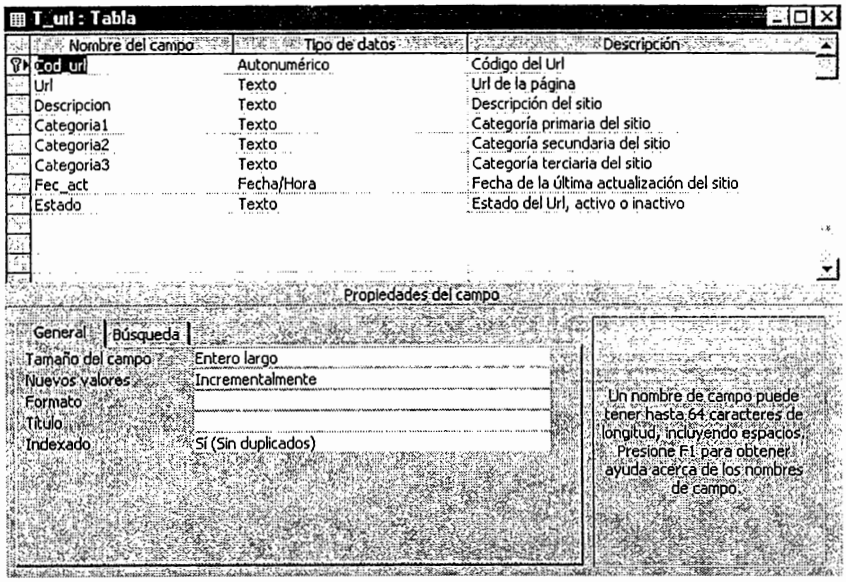
T_Relacion

Utilizada para almacenar el número de ocurrencias por palabra. La ocurrencia por palabra es considerada para evitar que una palabra que se encuentre repetida sea ingresada más de una vez a esta tabla.



T_Url

Almacena los datos relativos a las direcciones de los sitios Web nacionales recorridos, clasificando los URL's introducidos por medio del formulario en línea de acuerdo a la categoría del contenido.



Consultas

Agrega_Urls

La consulta "Agrega_Urls" está relacionada con la tabla T_Agregados y el objetivo de esta consulta es adicionar los URL's ya revisados por el administrador de la Base de Datos, a la tabla T_Depurados para que puedan ser considerados por el robot. Esta consulta implica la ejecución de la sentencia SQL siguiente:

```
INSERT INTO T_Depurados ( Cod_url, Url, Descripcion, Categoria1, Categoria2,
Categoria3, Nombre, Organizacion, Email )
SELECT T_Agregados.Cod_url, T_Agregados.Url, T_Agregados.Descripcion,
T_Agregados.Categoria1, T_Agregados.Categoria2, T_Agregados.Categoria3,
T_Agregados.Nombre, T_Agregados.Organizacion, T_Agregados.Email AS Expr1
FROM T_Agregados
WHERE (((T_Agregados.Estado)=Yes))
ORDER BY T_Agregados.Cod_url;
```

Formularios

El Mantenimiento de la Base de Datos lo constituyen los siguientes formularios:

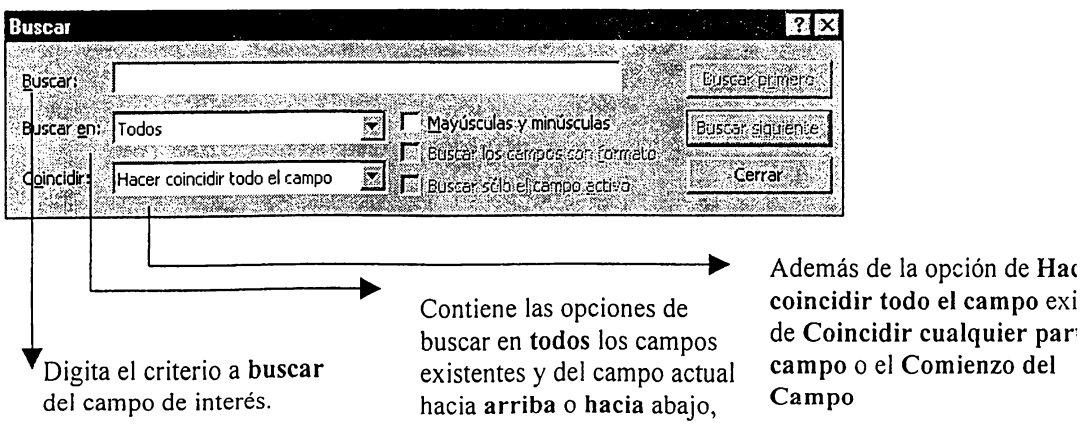
1. Cons_Url_Inicial
2. Consulta_Elementos_Gramaticales
3. Consulta_Urls_Agregados
4. Elem_Gram
5. Menu
6. Menu_Consultas
7. Url_Agregados
8. Url_Inicial

En la mayoría de los formularios se emplean los botones descritos a continuación:

Se utiliza para adicionar un nuevo registro en la tabla.

Guarda la información que se ha modificado de un registro determinado.

Busca un registro dentro de la tabla específica. Al activar esta opción se obtiene la siguiente ventana de diálogo:



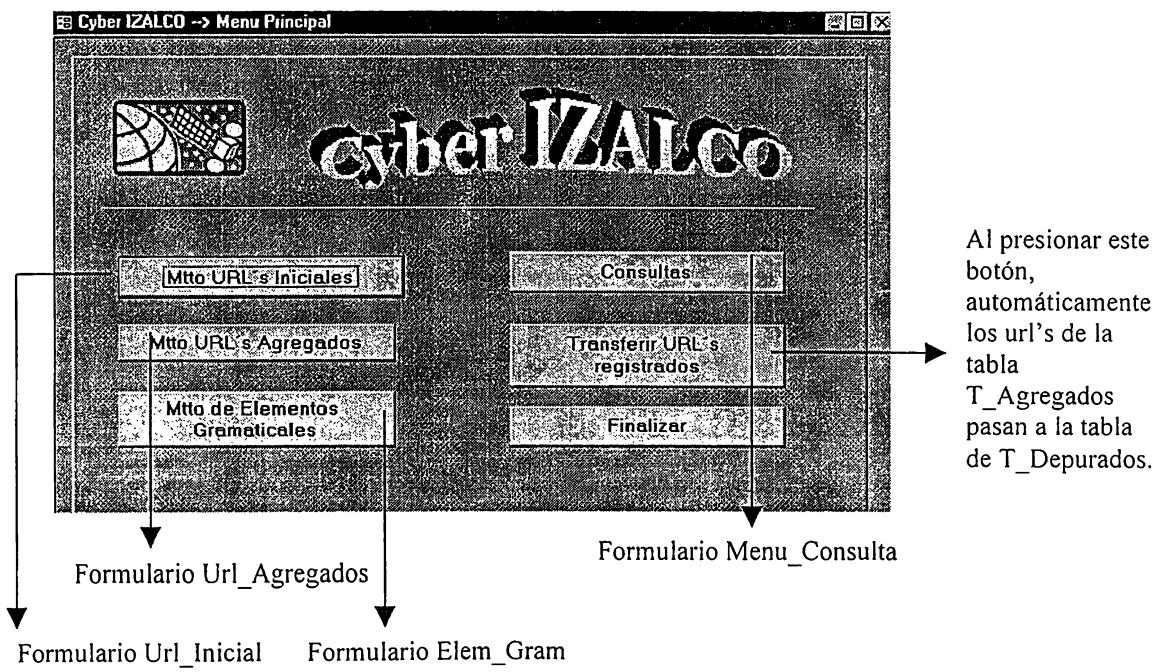
Elimina el registro actual en la tabla de trabajo.

Salir de la opción actual dentro del mantenimiento.

Descripción de los Formularios que constituyen el Mantenimiento.

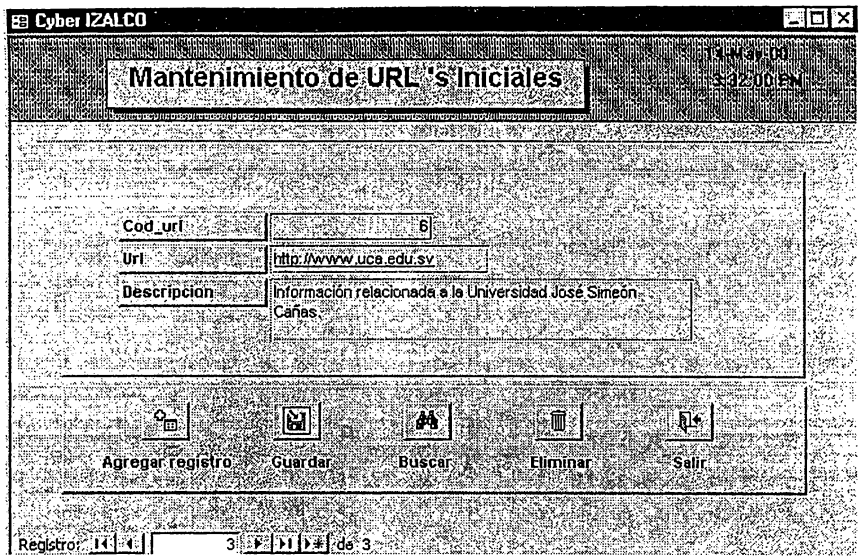
Menu

Formulario Inicial del Mantenimiento de la Base de Datos.



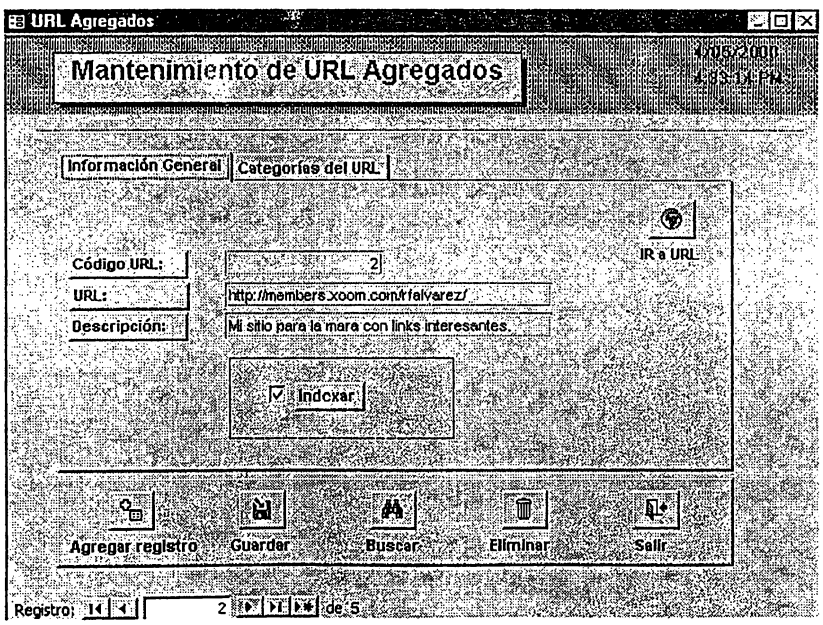
Url_Inicial

Brinda mantenimiento a la tabla T_Inicial, en la cual se encuentra los URL's con los que el robot inicia la búsqueda.



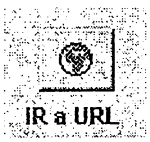
Url_Agregados

El mantenimiento que brinda este formulario se refiere a aquellas direcciones que se han agregado a través del Web de manera explícita por algún usuario interesado en que su información sea tomada en cuenta por el robot del Cyber Izalco.



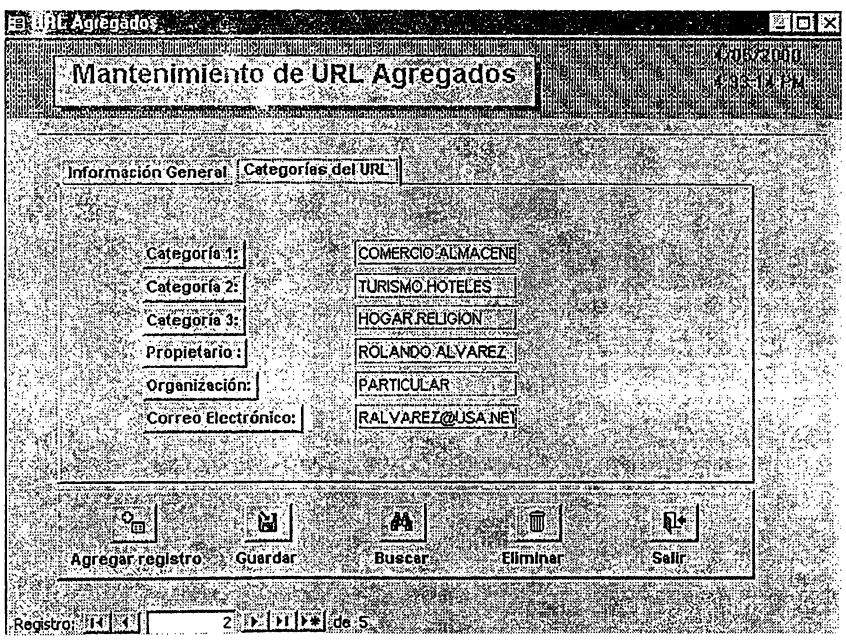
Esta información no está alojada bajo el dominio SV por lo cual debe depurarse en base a los siguientes criterios:

1. Que el URL sea válido.
2. Que el URL contenga información referente a El Salvador.
3. Que el URL no represente riesgo por parte de "hackers" o similares.



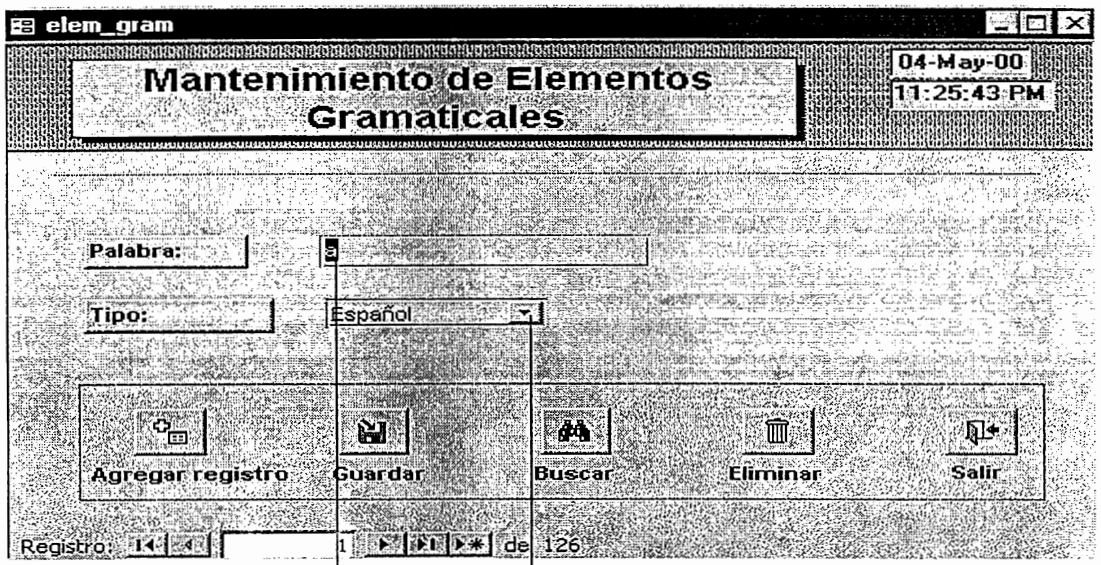
Por medio de este botón, el administrador de la Base de Datos puede analizar el URL y verificar que este sea válido. Cuando el administrador de la Base de Datos considera que el URL es aceptado, coloca un cheque en la casilla "Indexar".

Este mantenimiento está formado por dos tabuladores, en el primero se muestra la información general del URL agregado y en el segundo, la información del usuario de Internet que ingresó el URL, como por ejemplo, las categorías que relaciona el URL, el nombre del propietario, organización y el correo electrónico.



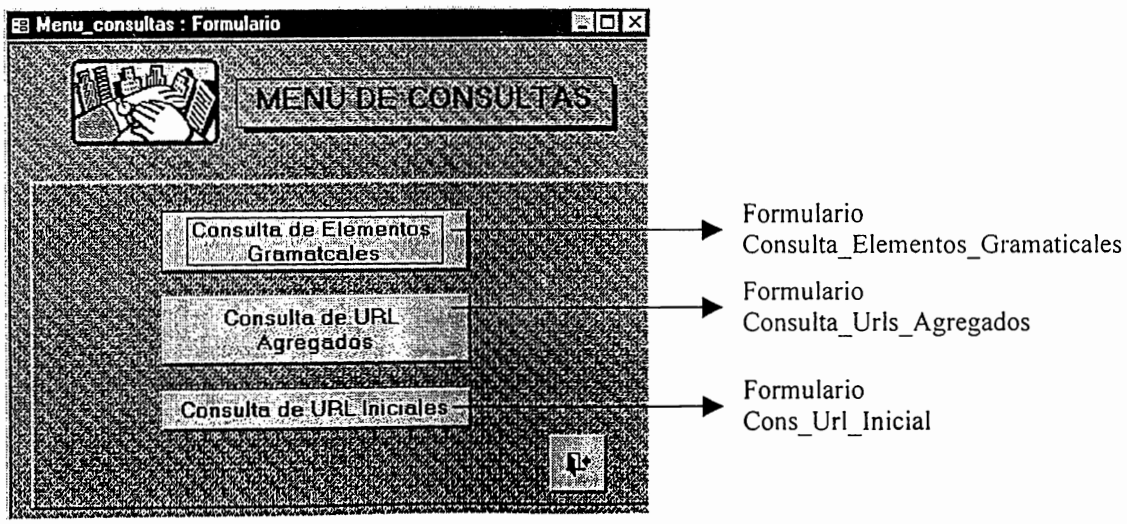
Elem_Gram

Mantenimiento de la tabla T_Elegramatical, se pueden modificar o agregar nuevos elementos gramaticales, para que el robot los ignore al buscar las palabras en la páginas Web.

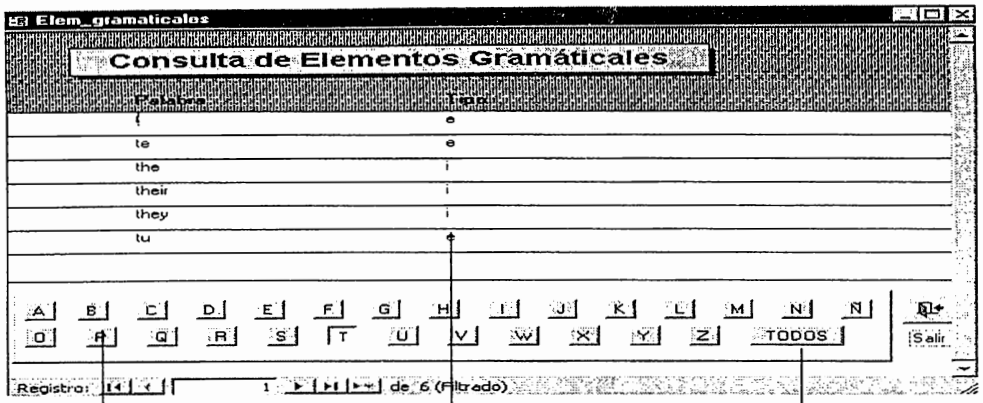


Menu_Consultas

Este mantenimiento posee las consultas generales del mantenimiento de la Base.



a. Consulta_Elementos_Gramaticales: Esta consulta se puede realizar en base a diversos criterios sobre la información almacenada en al tabla T_Elegramatical.

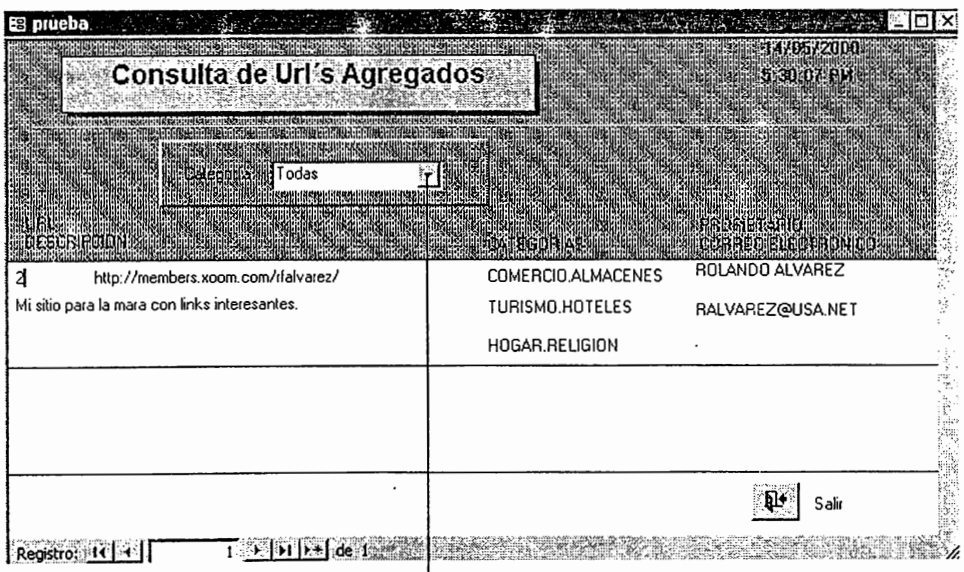


Se filtran de manera ordenada los elementos gramaticales que empiezan con la letra elegida.

Palabra en inglés (i) , español (e) o html (h)

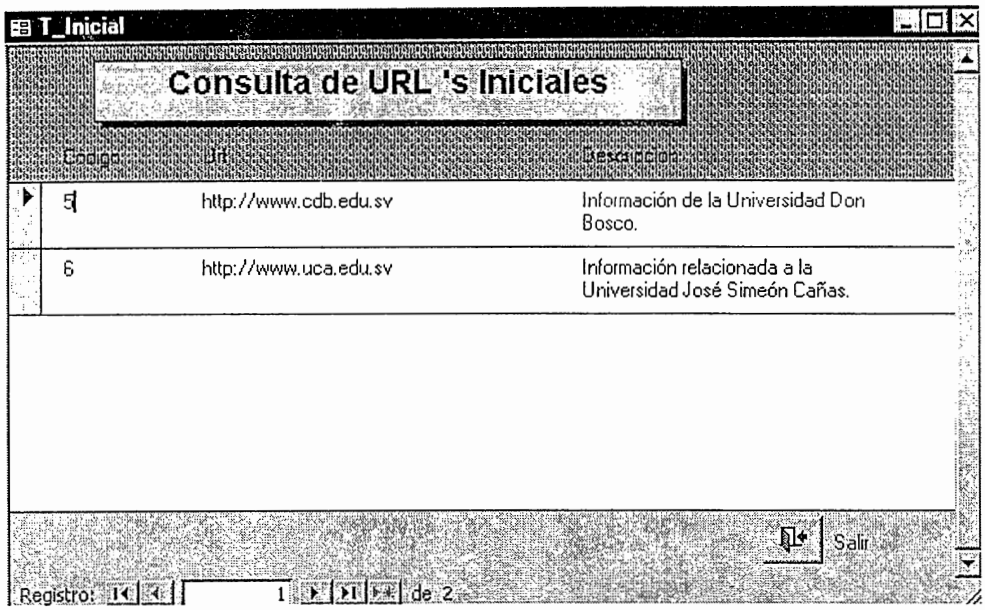
Presenta todos los elementos gramaticales

b. Consulta_Urls_Agregados: Esta consulta presenta la información referente a los URL's que han sido adicionados por medio del formulario en línea y han sido almacenados en la tabla de T_ Agregados.



Permite que se muestren los URL's por las categorías que se encuentren existentes o muestra todos los urls sin especificar la categoría.

c. Cons_Url_Inicial: Muestra el listado de los URL's que le sirven de punto de partida al robot Web para su recorrido.



MACROS

El Mantenimiento de la Base de Datos consta de las siguientes macros:

1. Agrega_Urls's
2. Busqueda_por_Categoria
3. Cons_Elem_Gramaticales
4. Cons_Url_Agregados
5. Cons_Ulr_Iniciales
6. Finalizar
7. Inicio_del_Programa
8. Mantto_Elem_Gramaticales
9. Mantto_Url_Agregados
10. Mantto_Url_Iniciales
11. Menu_Consultas
12. Opciones_Alfabeticamente

Descripción de Macros

1. Agrega_URL's: Por medio de esta macro se hace posible que los URL's de la tabla T_Agregados y que han sido previamente analizados por el administrador de la Base de Datos, puedan ser transferidos automáticamente a la tabla T_Depurados.

2. Busqueda_por_Categoria: Se utiliza para presentar en el formulario de URL's Agregados, los URL's ordenados por las categorías existentes en la tabla, además da opción de mostrar los URL's sin importar en qué categoría se encuentren.

3. Cons_Elem_Gramaticales: Por medio de esta macro se hace referencia al formulario "Consulta_Elmentos_Gramaticales", en el cual se muestran los elementos gramaticales, que el robot no toma en cuenta al momento de leer una determinada página.

4. Cons_Url_Agregados: Esta macro hace referencia al formulario "Cons_Url_Agregados", que es donde se presentan los datos almacenados en la tabla de T_Agregados, presentando los URL's por categorías con su respectiva información.

5. Cons_Url_Iniciales: Ejecuta el formulario "Cons_Url_Inicial", el cual permite mostrar los URL's iniciales, o los URL's con los que el robot inicia su recorrido en el Web.

6. Finalizar: Ejecuta la terminación del programa.

7. Inicio_del_Programa: Ejecuta el formulario "Menu", que es el que presenta la forma que contiene todas las opciones del sistema.

8. Mantto_Elem_Gramaticales: Por medio de esta macro se ejecuta el formulario del mantenimiento llamado "Elem_Gram", en el cual se presentan los elementos gramaticales.

9. Mantto_Url_Agregados: Esta macro permite ejecutar el mantenimiento de los URL's Agregados.

10. Mantto_Url_Iniciales: Ejecuta el mantenimiento de los URL's iniciales, mostrando los URL's iniciales contenidos en la tabla T_Inicial.

11. Menu_Consultas: Ejecuta el formulario que contiene las consultas generales del Mantenimiento de la Base de Datos.

12. Opciones_Alfabeticamente: Se utiliza para crear un filtro de los elementos gramaticales por medio de la letra que se desea mostrar o en su defecto presentar la información sin filtro alguno.

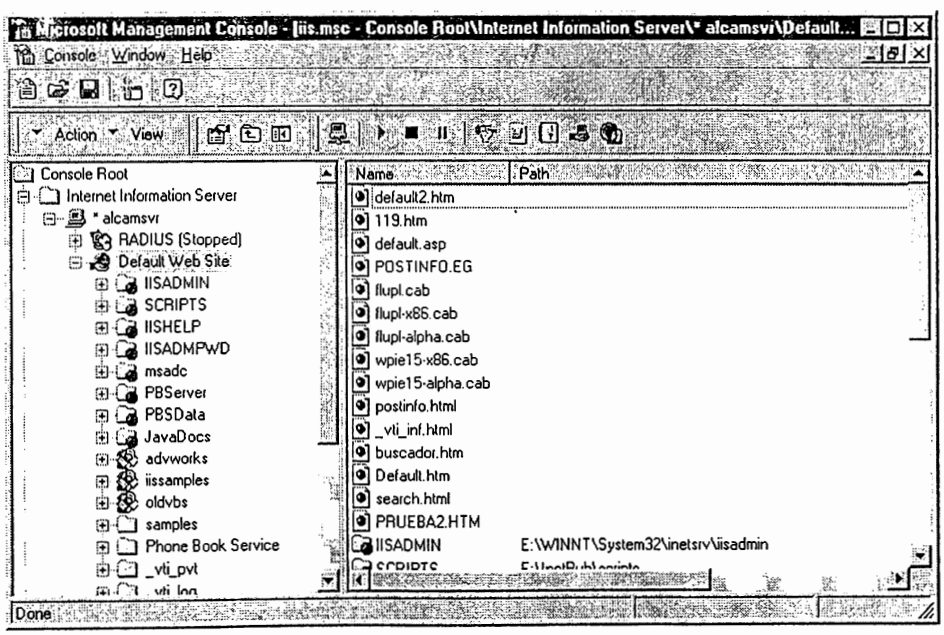
Sección II. Instalación de Páginas Web

Para realizar la instalación de las páginas web que conforman la interfaz gráfica para el usuario de internet, se debe de crear un directorio llamado "CIzalco" en la siguiente ruta "D:\Inetpub\", suponiendo que la unidad D es la unidad en la que se han instalado los directorios del Internet Information Server de Windows NT.

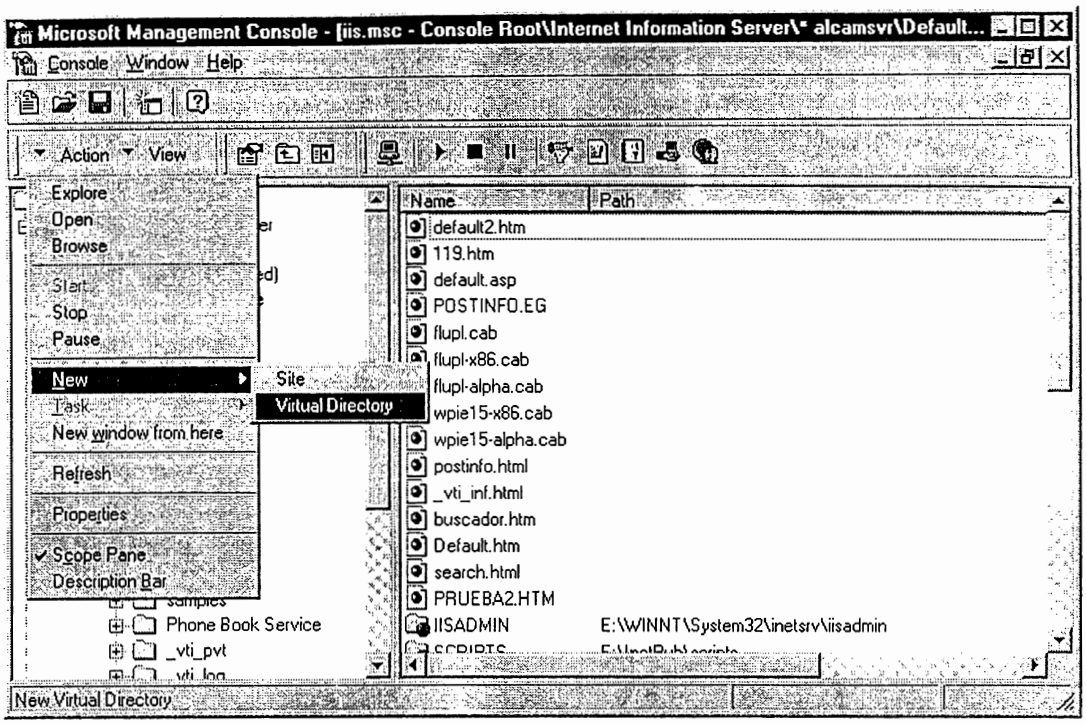
Una vez creado este directorio se deben de copiar los archivos de las páginas Web, imágenes y Applets, los cuales se encuentran en la siguiente ruta del CD de instalación: "\SitioWeb\CIzalco\".

Finalmente se debe de crear un directorio virtual desde la consola del Administrador del Internet Information Server, llamado CIzalco, haciendo referencia al directorio creado anteriormente y asignándole derechos de ejecución, con el propósito que los scripts de las páginas puedan ser ejecutados por el servidor. Para ello, se muestran a continuación los pasos a seguir para crear este directorio virtual.

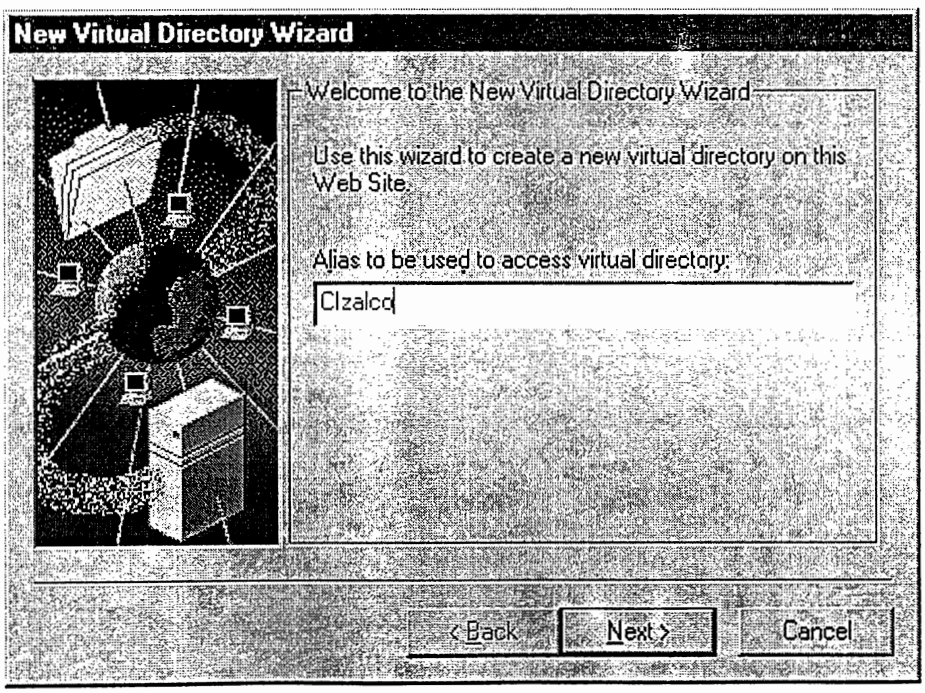
1. Si usted cuenta con la versión 4.0 del Microsoft Management Console, le aparecerá una ventana similar a la siguiente.



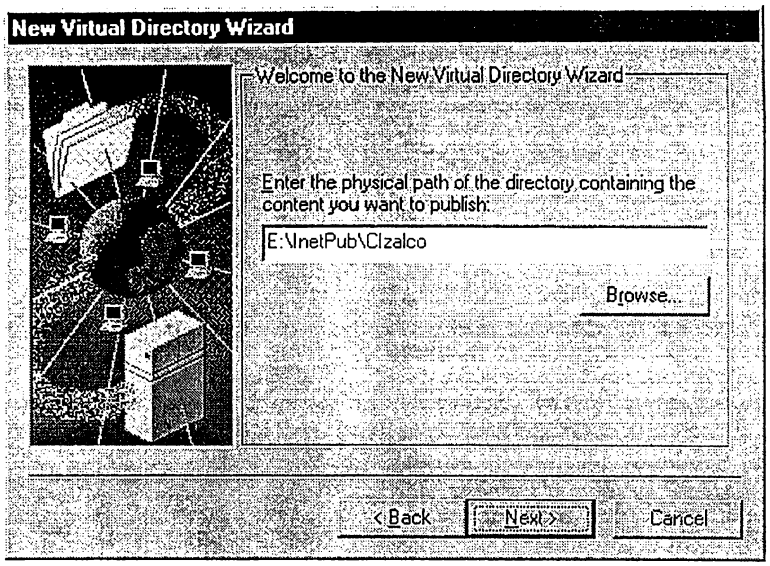
2. Del menú de Action debe de seleccionar New - Virtual Directory



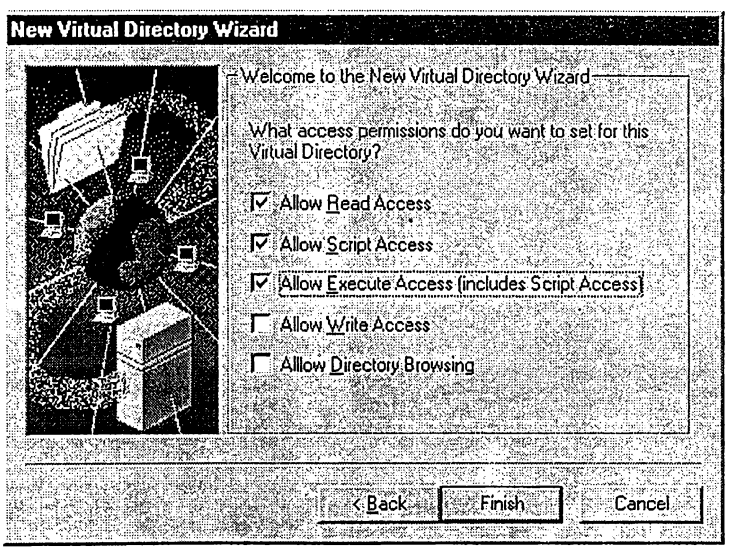
3. Hecho lo anterior, le aparecerá una ventana para definir el nombre del directorio virtual, es decir "CIzalco", luego presione "Next".



4. Le aparecerá una ventana solicitándole que especifique cual es el directorio físico al cual el directorio virtual hará referencia, puede digitar la ruta o buscarla empleando el botón de "Browse". Presione "Next"



5. En la siguiente ventana se le pide que especifique cuales serán los permisos de este directorio, es importante que seleccione la casilla de "Allow Execute Access", para permitir que las páginas de ASP sean ejecutadas. Por último presione "Finish" para finalizar la creación del directorio virtual.



Sección IV. Problemas Frecuentes

A continuación se presentan algunos de los errores o problemas más frecuentes al trabajar con Java.

Es muy importante recalcar que Java es un lenguaje "Case Sensitive", esto es, hace diferencia entre letras mayúsculas y minúsculas, por lo que se debe de tener mucho cuidado al momento de compilar o ejecutar las clases.

Caso 1.

Error. Si al momento de intentar compilar algún archivo, le aparece el mensaje siguiente.

```
E:\>javac cizalco\search\EnginePrefs.java
Comando o nombre de archivo incorrecto
```

Acción. Debe de verificar que el PATH tenga entre sus directorios, los directorios de ejecución de Java, los cuales son: "X:\jdk1.1.8" y "X:\jdk1.1.8\bin", donde "X" es la unidad en que se ha instalado Java, de no ser así puede agregarlos, mediante la instrucción: "PATH = %PATH%;X:\jdk1.1.8\bin"

Caso 2.

Error. Si al intentar compilar alguna de las clases de Robot, le aparece el siguiente mensaje.

```
E:\>javac cizalco\search\Engineprefs.java
cizalco\search\Engineprefs.java:22: Public class cizalco.search.EnginePrefs must be
defined in a file called "EnginePrefs.java".
public class EnginePrefs {
             ^
1 error
```

Acción. Este error se refiere a que el nombre del archivo al momento de llamarlo debe de ser exactamente igual al de la clase, considerando las mayúsculas y minúsculas dentro del nombre .

Caso 3.

Error. Si al intentar compilar alguna clase del Robot, le aparece el siguiente mensaje.

```
E:\>javac cizalco.search.Monitor
javac: invalid argument: cizalco.search.Monitor
use: javac [-g][-O][-debug][-depend][-nowarn][-verbose][-classpath path][-nowrite]
[-deprecation][-d dir][-J<runtime flag>] file.java...
```

Acción: Este error se produce debido a que se ha escrito mal la sintaxis para la compilación de las clases, debe de chequear la ruta y el nombre del archivo y recordando que los directorios deben de separarse con una " \ " y el nombre de la clase a compilar debe de ser escrito incluyendo su extensión, por ejemplo: "Clase.java"

Caso 4.

Error. Si al ejecutar el robot le aparece el siguiente mensaje:

```
E:\>java cizalco.search.Monitor
java.sql.SQLException: [Microsoft][Controlador ODBC Microsoft Access 97] No se
pudo encontrar el archivo '(desconocido)'.
        at sun.jdbc.odbc.JdbcOdbc.createSQLException(Compiled Code)
        at sun.jdbc.odbc.JdbcOdbc.standardError(JdbcOdbc.java:3814)
        at sun.jdbc.odbc.JdbcOdbc.SQLDriverConnect(JdbcOdbc.java:1029)
        at sun.jdbc.odbc.JdbcOdbcConnection.initialize(JdbcOdbcConnection.java:145)
        at sun.jdbc.odbc.JdbcOdbcDriver.connect(JdbcOdbcDriver.java:165)
        at java.sql.DriverManager.getConnection(Compiled Code)
        at java.sql.DriverManager.getConnection(DriverManager.java:126)
        at cizalco.search.EnginePrefs.<init>(EnginePrefs.java:65)
        at cizalco.search.Monitor.<init>(Compiled Code)
        at cizalco.search.Monitor.main(Monitor.java:204)
```

Acción: Este mensaje ha sido ocasionado debido a que el controlador ODBC no logró encontrar la base de datos en el directorio indicado, para corregirlo debe de revisar el nombre del archivo de la base de datos y su ubicación, para que esta coincida con los datos del controlador ODBC creado.

Caso 5.

Error. Si al ejecutar el robot le aparece.

```
E:\>java cizalco.search.Monitor
java.sql.SQLException: [Microsoft][Administrador de controladores ODBC] El nombre
del origen de datos no se encontr¾ y no se especific¾ ning·n controlador
predeterminado
        at sun.jdbc.odbc.JdbcOdbc.createSQLException(Compiled Code)
        at sun.jdbc.odbc.JdbcOdbc.standardError(JdbcOdbc.java:3814)
        at sun.jdbc.odbc.JdbcOdbc.SQLDriverConnect(JdbcOdbc.java:1029)
        at sun.jdbc.odbc.JdbcOdbcConnection.initialize(JdbcOdbcConnection.java:145)
        at sun.jdbc.odbc.JdbcOdbcDriver.connect(JdbcOdbcDriver.java:165)
        at java.sql.DriverManager.getConnection(Compiled Code)
        at java.sql.DriverManager.getConnection(DriverManager.java:126)
        at cizalco.search.EnginePrefs.<init>(EnginePrefs.java:65)
        at cizalco.search.Monitor.<init>(Compiled Code)
        at cizalco.search.Monitor.main(Monitor.java:204)
```

Acción: Este mensaje se ha producido debido a que java no encuentra el controlador indicado, para corregirlo debe de revisar que el nombre del controlador que ha creado sea "buscador".

ANEXO C.
GLOSARIO TECNICO

ANCHO DE BANDA

Es la capacidad de transporte de información de un canal de comunicación. Técnicamente el ancho de banda es el rango de frecuencia , éste rango denota el número de canales de comunicación para esta línea.

API (Aplication Program Interface)

Consiste en una serie de estándares de interfaz definidos para una aplicación. Un API típicamente define como una aplicación deberá presentarse a un usuario, como las entradas deberán ser solicitadas y obtenidas y como las salidas deberán ser realizadas.

ARPANET

Es una abreviación de Advanced Research Projects Administration Network, el sistema de red informática del cual nació el Internet. ARPANET comenzó en 1969 como un experimento del Ministerio de Defensa de los EE.UU. que probaba las redes de comunicación por medio de paquetes de información.

BERKELEY, BASE DE DATOS

La base de datos Berkeley, es un sistema manejador de base de datos capaz de realizar acceso a los datos por medio de llaves. El software es distribuido en forma de código fuente, con el objeto que los desarrolladores interesados puedan modificarlo y recompilarlo, para posteriormente emplearlo en aplicaciones específicas.

Esta base de datos soporta tanto lecturas como escrituras simultaneas y garantiza que todos los cambios son capaces de ser recobrados, aún en caso de fallas catastróficas de hardware durante la actualización de una base de datos.

La librería de la Base de Datos Berkeley, no provee de interfaz de usuario, interfaz gráfica de entrada, soporte de SQL o cualquier de los otros estándares de interfaz de base de datos; lo que provee son los bloques programáticos que le permitirán a los

desarrolladores fácilmente proveer de la funcionalidad de bases de datos a interfaces y aplicaciones.

BYTECODE

Son un conjunto de instrucciones muy parecidas al código de máquina, pero que no son específicas para algún procesador.

CGI

Es una interfaz estándar que permite que programas externos puedan correr bajo un servidor de información, actualmente los servidores soportados son HTTP. Es una de las formas más comunes de proveer acceso a un manejador de Bases de Datos desde el Web.

CRITERIO DE BUSQUEDA

Una búsqueda de documentos conteniendo una o más palabras especificadas por el usuario.

FTP (File Transfer Protocol)

FTP permite transmitir ficheros sobre internet entre una máquina local y otra remota. Los comandos básicos de FTP son:

- Open 'nombre de nodo o dirección'

  Abre una sesión FTP en el ordenador indicado.

- Dir

  Lista los ficheros del directorio del ordenador al que nos hemos conectado.

- Pwd

  Visualiza el directorio remoto en el que estamos situados.

- Cd 'nombre del directorio'

  Cambio al directorio especificado.

- Lcd 'nombre del directorio'

  Comando de movimiento para directorios locales.

- Binary

  Establece modo binario de transferencia.

- Ascii

  Establece modo ascii de transferencia. Solo para ficheros de texto.

- Get 'nombre archivo'

  Obtiene un determinado fichero desde el ordenador remoto al local.

- Put 'nombre archivo'

  Transmite un determinado fichero desde nuestro directoriolocal al remoto.

- Bye

  Cierra una sesión FTP.

GOPHER

Gopher es un sistema de entrega de información distribuido. Utilizando gopher es posible acceder a información local o bien acceder a servidores de información gopher de todo el mundo.

Gopher combina las características de BBS (Bulletin Board Service) y bases de datos, permitiendo establecer una jerarquía de documentos, y permitiendo búsquedas en ellos por palabras o frases clave. Concebido y desarrollado en la Universidad de Minnesota en el año 91 es de libre distribución para fines no comerciales.

Gopher soporta directorios, ficheros de texto, item de búsqueda, sesiones telnet y tn3270, multimedia y texto formateado (postcript y otros). Algunos ejemplos de la información que gopher puede ofrecer: pronósticos y mapas del tiempo, recetas, problemas y respuestas de temas de computación, animaciones sobre reacciones químicas, libros clásicos, catálogos de bibliotecas de todo el mundo, canciones, catálogos de cursos universitarios, etc.

Gopher trabaja en arquitectura Cliente/Servidor, existiendo clientes para: Macintosh, DOS, Microsoft Windows, Unix (Terminales ASCII, EMACS y X-Windows) entre otros; y servidores para: UNIX, VMS, Macintosh, VM/CMS, DOS, OS/2, MVS, etc.

HTML (Hypertext Markup Language)

Lenguaje usado para escribir documentos para servidores World Wide Web. Es una aplicación de la ISO Standard 8879:1986 (SGML, Standard Generalized Markup Language).

Los Archivos que utiliza como fuente son archivos de texto ASCII, incluyendo códigos de formato y enlaces con otros documentos, de tal manera que para crearlos puede utilizarse cualquier editor de texto.

HTTP (Hypertext Transfer Protocol)

HTTP es un protocolo que se encarga de distribuir y manejar sistemas de información hipermedia. Es un protocolo genérico orientado al objeto, que puede ser usado para muchas tareas como servidor de nombres y sistemas distribuidos orientados a objetos por extensión de los comandos, o métodos usados. Una característica de HTTP es la independencia en la visualización y representación de los datos, permitiendo a los sistemas ser construidos independientes del desarrollo de nuevos avances en la representación de los datos.

INTERNET INFORMATION SERVER (IIS)

Es un Servidor Web para Windows NT que permite publicar información en una Intranet o en el Internet. El ISS transmite información utilizando el Protocolo de Transferencia de Hipertexto, así también puede ser configurado para proveer la transferencia de archivo por medio del protocolo FTP y servicios Gopher.

INDEPENDENCIA DE LA PLATAFORMA

Es la capacidad del programa de trasladarse con facilidad de un sistema computacional a otro.

INDICE

El catálogo de documentos creados por el software de un motor de búsqueda. También llamado catálogo índice es a menudo utilizado como sinónimo de motor de búsqueda.

INTERFAZ

Define un protocolo entre los procesos cliente y servidor, de tal forma que puedan comunicarse entre sí en un nivel más alto que el envío y recepción de simples cadenas de bytes, en un ambiente heterogéneo de interconexión.

INTERNET

Es una red de redes de computadoras. Nacida como resultado de un experimento del ministerio de defensa Americano, conoce su difusión más amplia en el ámbito científico-universitario.

Desde el punto de vista técnico, Internet es un gran conjunto de redes de computadores interconectados entre sí. Es una basta fuente de información de todo tipo. En cuanto a funcionamiento interno, Internet no se ajusta a ningún tipo de ordenador, tipo de red, tecnología de conexión y medios físicos empleados. Internet no tiene una autoridad central, es descentralizada. Cada red mantiene su independencia y se une cooperativamente a las otras respetando una serie de normas de interconexión.

INTRANET

Red interna de una compañía, en la cual se toma la tecnología de internet y se aplica dentro de una unidad de negocios de cualquier tamaño para mejorar la productividad y la transferencia de información.

MICRO CDS/ISIS

Es un sistema avanzado de almacenamiento y consulta de información no numérica que fue desarrollado por UNESCO, desde 1985 para satisfacer la necesidad expresada por muchas instituciones, especialmente en países en vías de desarrollo, para poder sacar adelante sus actividades de procesamiento de información con el uso de módems.

MOTOR DE BUSQUEDA

Es el software que busca dentro de un índice y retorna los resultados de dicha búsqueda de acuerdo a un formato específico. Motor de búsqueda es a menudo utilizado como

sinónimo de índice o spider, aun cuando estos son componentes separados que trabajan dentro del motor.

ODBC (Open Database Connectivity)

La conectividad abierta de base de datos es un controlador que abre la conexión con la fuente de datos en el proceso de cascada IDC / HTX.

ROBOTS

Es un término aplicado a programas de comunicación que utilizando HTTP como protocolo de comunicación, exploran grandes porciones del Web y de manera recursiva extraen información de ellas. Dicha información puede después ser utilizada para alimentar motores de búsqueda, para efectos estadísticos, para realizar copias de respaldo, etc.

SCRIPTS

Un SCRIPT es un conjunto de comandos de un lenguaje específico, los cuales pueden por ejemplo: asignar un valor a una variable, indicarle al servidor Web que procese y envíe algún tipo de información.

SPIDER

Sinónimo de Robot, es el software que recorre los documentos siguiendo los enlaces de éstos y adicionándolos a un índice.

STOP WORDS

Son palabras tales como: conjunciones, preposiciones, artículos e incluyendo palabras como I, PARA y UN, que aparecen a menudo en los documentos e inclusive solas pueden tener algún significado.

TCP/IP (Transmission Control Protocol / Intener Protocol)

Familia de protocolos que hacen posible la interconexión y tráfico de red en Internet. A ella pertenecen por ejemplo FTP, SMTP, NNTP, etc. Los dos protocolos más importantes son los que dan nombre a la familia: IP y TCP.

URL (Uniform Resource Locator)

Utilizado para especificar un objeto en internet. Puede ser un fichero, grupo de news, ghopher, etc.

Algunos ejemplos :

- file://www.uco.es/www-docs/HTMLprimer.txt
- Http://www.cica.es/
- telnet://lucano.uco.es

WWW (World Wide Web)

Servidor de información, desarrollado en el Cern (Laboratorio Europeo de Física de Partículas), buscando construir un sistema distribuido de hipermedia e hipertexto. Existen gran cantidad de Servidores WWW para diferentes plataformas.