

01132
15



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA
CIUDAD UNIVERSITARIA

**DISEÑO DE UNA INTERFAZ WEB PARA LA
BUSQUEDA Y RECUPERACIÓN DE INFORMACIÓN EN
UN DICCIONARIO TERMINOLÓGICO**

T E S I S

QUE PARA OBTENER EL GRADO DE:

INGENIERO EN COMPUTACIÓN

P R E S E N T A N:

**BÁRCENAS FIGUEROA LAURA
MORENO DEL PILAR MARLENE**

DIRECTOR: M. en C. GABRIEL CASTILLO HERNÁNDEZ

MÉXICO, D. F.

2003

A

ÍNDICE

Introducción	1
Capítulo 1 Fundamentos Teóricos	2
1.1 Ingeniería Lingüística	2
1.1.1. La Ingeniería Lingüística en la UNAM	2
1.2. Diccionarios onomasiológicos	3
1.2.1. Descripción de los Diccionarios onomasiológicos	3
1.3. Bases de Datos	3
1.3.1. Definiciones y conceptos de las bases de datos	3
1.3.2. Sistema manejador de bases de datos (DBMS)	5
1.3.2.1. Funciones de un DBMS	7
1.3.2.2. Arquitectura de un DBMS	8
1.3.2.3. Manejador de bases de datos PostgreSQL	9
1.3.3. Modelo de datos	10
1.3.3.1. Modelo relacional	11
1.3.3.1.1. Modelo Entidad-Relación	11
1.3.3.1.2. Normalización	15
1.3.4. Diccionario de datos	18
1.4. Metodología para el desarrollo de sistemas	20
1.4.1. Definición de Metodología	20
1.4.2. Planeación	21
1.4.3. Análisis	22
1.4.4. Diseño	22
1.4.5. Desarrollo	23
1.5. Sistemas de Información vía Internet	23
1.5.1. Internet	23
1.5.2. Arquitectura Cliente Servidor	24
1.5.3. Servidores de Páginas Web	25
1.5.3.1. Servidor Web Apache	26
1.5.4. Protocolos de comunicación	27
1.5.5. World Wide Web	31
1.5.5.1. Concepto de Página Web	31

13

1.5.5.1.1. Páginas Estáticas	33
1.5.5.1.2. Páginas Dinámicas	34
1.5.6. Lenguajes de Programación para web	35
1.5.7. Acerca de los principales visualizadores	38
1.6. Sistema operativo Linux	39
Capítulo 2 Desarrollo del Sistema	42
2.1. Análisis	42
2.1.1. Condición Actual del Sistema	42
2.1.2. Delimitación del Sistema	43
2.2. Diseño	46
2.2.1. Propuesta de Software	46
2.2.2. Diagrama jerárquico funcional del sistema	47
2.2.3. Diseño de la Base de Datos	49
2.2.3.1. Definición de Entidades	50
2.2.3.2. Diseño Conceptual y Físico	52
2.2.4. Diccionario de Datos	55
2.2.5. Interfaz Web para el Diccionario electrónico	58
Capítulo 3 Implantación	64
3.1. Introducción	64
3.1.1. Especificaciones del hardware	64
3.1.2. Seguridad del servidor	64
3.2. Creación de la Base de Datos	65
3.3. Migración de datos	69
3.4. Creación de la interfaz de usuario	70
3.4.1. Visitante	71
3.4.2. Administrador	75
3.4.3. Owner	77
3.4.4. Usuario de Base	79
Capítulo 4 Conclusiones	84
Bibliografía	86
Apéndice 1 Metodología y Escenarios	87
Apéndice 2 Fichas Técnicas	122

INTRODUCCIÓN

La presente tesis expone el diseño y desarrollo de un sistema de información vía Internet, el cual permitirá ingresar a la información de un diccionario onomasiológico así como a los sistemas desarrollados dentro del área de Ingeniería Lingüística del instituto de Ingeniería. Además admitirá la administración de la Base de Datos de el sistema por medio de páginas en Internet.

En el capítulo 1, se hace la diferencia entre los dos tipos de diccionarios existentes (semasiológico y onomasiológico) desde el punto de vista semántico y su utilidad práctica. Se explica además, la necesidad de un sistema que cumpla con los requerimientos de un Lingüista para consultar y manipular la información de este diccionario onomasiológico. También se presentan los fundamentos teóricos que dan pauta al desarrollo de esta tesis.

En el capítulo 2 se analiza la condición actual de un sistema similar pero más limitado y se especifican todas aquellas características y requerimientos del nuevo sistema. Se explica también la etapa de diseño, donde se da a conocer la propuesta de software, un diagrama jerárquico funcional y la creación de la Base de Datos, se muestra además la utilidad de los escenarios y fichas técnicas ejemplificando el diseño de la interfaz de usuario.

En el capítulo 3 se explica la etapa de implantación, donde se lleva a cabo la elección de la plataforma operativa, la creación de la base de datos y la programación de la interfaz. También, se expone el código principal para el desarrollo de las pantallas más representativas del sistema, se explica la migración de los datos a la base de datos de SQL Server a PostgreSQL y se ejemplifica el recorrido habitual de cada uno de los diferentes usuarios del sistema.

Finalmente en el capítulo 4, se exponen las conclusiones acerca del sistema y los logros obtenidos. La tesis contiene además dos apéndices que son: Metodología Para El Desarrollo De Sitios Web y Fichas Técnicas Del Sistema.

1 FUNDAMENTOS TEÓRICOS

1.1 INGENIERÍA LINGÜÍSTICA

El Grupo de Ingeniería Lingüística (GIL) representa un grupo de investigación en la que dos áreas, al parecer alejadas, retoman el sentido de unidad e interdependencia para formar un solo núcleo. Estas áreas son la lingüística y la ingeniería.

La ingeniería lingüística es el área interdisciplinaria de investigación aplicada al desarrollo de sistemas computacionales para reconocer, interpretar y generar lenguaje humano. Existe una correspondencia biunívoca, de forma que la lingüística permite la creación de modelos en lenguaje natural que puedan ser utilizados por los sistemas computacionales, mientras que la ingeniería permite el desarrollo de sistemas que puedan resolver las necesidades específicas planteadas por los problemas lingüísticos.

1.1.1 LA INGENIERÍA LINGÜÍSTICA EN LA UNAM

El Grupo de Ingeniería Lingüística, en el seno del Instituto de Ingeniería de la UNAM, se conformó con el fin, primero, de crear una base de conocimiento relativa y concerniente a esta área de trabajo, y segundo, de formar personal especializado y comprometido con el estudio y desarrollo de las diversas áreas que ésta ofrece.

Gracias al apoyo del Instituto de Ingeniería y con el patrocinio del Consejo Nacional de Ciencia y Tecnología y de la propia UNAM, el Grupo de Ingeniería Lingüística ha venido realizando proyectos vinculados con el procesamiento de lenguaje natural. Como proyecto central del GIL, alrededor del cual giran las diferentes líneas de investigación, se tiene el diseño y creación del diccionario integral que comprende de cinco fases bien definidas: adquisición de datos, creación de bases de datos y captura de información, determinación de paradigmas semánticos, diseño del motor de búsqueda y diseño de la interfaz del usuario. Este proyecto, motor del Grupo de Ingeniería Lingüística, persigue crear un sistema de búsqueda onomasiológica, esto es, un diccionario que permita la búsqueda de términos a partir de la descripción del concepto mediante el uso de lenguaje natural.

1.2 DICCIONARIOS ONOMASIOLÓGICOS

1.2.1 DESCRIPCIÓN DE LOS DICCIONARIOS ONOMASIOLÓGICOS

Desde un punto de vista semántico, se pueden diferenciar dos tipos de diccionarios (Baldinger 1980): el semasiológico, en el que se parte del término para conocer el significado del mismo, y el onomasiológico, en el que se busca el término que corresponde a un concepto determinado.

Cada uno de estos tipos de diccionario tiene su utilidad práctica. El semasiológico ayuda a los lectores en la interpretación de textos, en tanto el onomasiológico ayuda a los escritores a producir textos.

La mayoría de los lexicógrafos reconoce la necesidad de los diccionarios onomasiológicos para ayudar a los usuarios en la búsqueda de una palabra que ha escapado de la memoria pero que se recuerda el concepto.

Para ello se han creado los diccionarios ideológicos, los ordenados por materias, los pictóricos y los de sinónimos. Sin embargo, en la praxis de la búsqueda onomasiológica, cuando un autor creativo desea expresar una idea pero desconoce el término adecuado para ello, no tiene a la fecha ningún diccionario.

1.3 BASES DE DATOS

1.3.1 DEFINICIONES Y CONCEPTOS DE LAS BASES DE DATOS

Una base de datos es una colección integrada de datos, que requiere tener una organización lógica y física en una forma especial. La organización física consiste en seleccionar los mecanismos de almacenamiento (estructuras, métodos de acceso, dispositivos, etc.) de los datos; por otro lado, la elección del modelo de datos (jerárquico, red o relacional) determina su organización lógica.

Los cuatro componentes principales de una base de datos son: los datos a manejar, el hardware, el software, y el personal encargado del manejo del sistema. A continuación se describen de manera breve cada uno de ellos.

Datos.- Los datos de una base de datos, cumplen con dos características:

- ❖ Deben cumplir con la regla de integridad referencial (*este tema se explica con más detalle en componentes de Integridad*), es decir, deben recoger toda la información necesaria, sin que exista repetición de la misma dentro de la base.
- ❖ Deben ser compartidos a nivel de aplicación, es decir varios usuarios pueden acceder al mismo tiempo a la base y hacer uso de estos datos.

Hardware.- Es el soporte físico que permite almacenar la información de la base de datos. Normalmente está integrado por la computadora central (host computer en inglés), la unidad central de procesamiento (CPU, por sus siglas en inglés) y sus aditamentos asociados, tales como la memoria principal, los dispositivos de Entrada-Salida y la unidad de almacenamiento secundario.

Software.- El software permite interactuar con la base de datos de manera eficiente. El Sistema Manejador de bases de datos (DBMS por sus siglas en inglés) es el encargado de administrar la base de datos, y debe ofrecer facilidades para la inserción, borrado y modificación de la información. Por lo tanto, todas las operaciones que se realicen sobre las bases han de ser atendidas por el DBMS.

Personal.- Las bases de datos son manipuladas por personal con diferentes perfiles, entre ellos podemos mencionar:

➤ **Administradores del DBMS (DBA por sus siglas en inglés)**

El DBA es el encargado de la administración de las bases de datos, es decir el control y manejo de la misma. Sus funciones principales consisten en la administración de –permisos, derechos y creaciones sobre nuevas bases de datos (Espacio en disco y prioridades de procesamiento etc.)-

➤ **Administradores de la base (DBO)**

El Administrador de la base (DBO por sus siglas en inglés) tiene como principales tareas: diseñar la base de datos, definir las estructuras de almacenamiento y métodos de acceso (tablas), modificar el diseño y organización física de la Base y una de las más importantes es el otorgar permisos de acceso a usuarios dentro de la base.

➤ **Programadores de aplicaciones o Desarrolladores.**

Los programadores de aplicaciones se encargan de diseñar y programar las aplicaciones necesarias para la utilización de las bases de datos, realizando las peticiones pertinentes al DBMS e interactuando con el sistema por medio de llamadas en DML (Lenguaje de Manipulación de Datos), estas peticiones están incorporadas en un programa escrito en un lenguaje de alto nivel (Por ejemplo, COBOL, PL/I, Pascal, C, etc.)

➤ **Usuarios**

Los usuarios son aquellos que utilizan el sistema de base de datos por medio de una interfaz gráfica e interactúan con los mismos dependiendo de los permisos que le hayan sido otorgados, puede manipular la información al insertar, borrar, modificar y buscar, sin necesidad de saber nada del diseño interno del sistema de base de datos, es decir, no tienen control directo sobre la forma en la que se guarda la información en la base de datos.

Para que los usuarios en cada uno de estos niveles interactúen con los datos de la base, es necesario contar con ciertas herramientas de manipulación (*interfaz*), las cuales son proporcionadas por los sistemas manejadores de bases de datos de las que hablaremos a continuación.

1.3.2 SISTEMA MANEJADOR DE BASES DE DATOS (DBMS)

Desde un punto de vista técnico, un sistema manejador de bases de datos es un conjunto coordinado de programas, procedimientos, lenguajes, etc. que suministran, a los distintos usuarios de la base, los medios necesarios para describir y manipular los datos almacenados en la base.

El DBMS se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de control de datos, los cuales se explican a continuación.

Lenguaje de definición de datos (DDL por sus siglas en inglés):

Permite establecer un modelo de base de datos por medio de una serie de definiciones que se expresan en un lenguaje especial, el resultado se almacena en un archivo llamado diccionario de datos.

Lenguaje Manipulador De Datos (DML por sus siglas en inglés):

Se refiere a una serie de expresiones que permiten manipular los datos. Entre las operaciones más comunes podemos mencionar: insertar, recuperar, eliminar o modificar los datos. Dicho lenguaje permite el acceso de los usuarios a los datos.

Existen básicamente 2 tipos de lenguajes de manipulación de datos:

- ❖ *Lenguajes con procedimientos o procedurales*, manipulan la base de datos registro a registro, en ellos se especifica qué operaciones se deben realizar para obtener los datos resultado.
- ❖ *Lenguajes sin procedimientos o no procedurales* son aquellos que operan sobre conjuntos de registros, en los cuales se especifica qué datos deben obtenerse sin decir como hacerlo. El lenguaje no procedural más utilizado es el SQL (Structured Query Language) que de hecho, es un estándar y es el lenguaje de los DBMS relacionales.

Lenguaje de control de datos (DCL por sus siglas en inglés):

Contiene elementos útiles para trabajar en un entorno multiusuario, en el que es importante la protección de los datos, la seguridad de las tablas y el establecimiento de restricciones en el acceso, así como elementos para coordinar el proceso de compartir los datos por parte de usuarios concurrentes, asegurando que no interfieren unos con otros.

En resumen podemos decir que este lenguaje maneja las transacciones, los usuarios y los privilegios.

1.3.2.1 FUNCIONES DE UN DBMS

La función del DBMS no se limita solo a permitir mediante la definición y manipulación de datos el diálogo entre los usuarios y la base de datos sino también brinda mecanismos para controlar la concurrencia de usuarios, la seguridad e integridad de la base datos (incluso después de que haya ocurrido una falla en el sistema, ya sea esta provocada por software o por hardware).

Las funciones principales de un DBMS se enlistan a continuación:

- Crear y organizar la base de datos
- Establecer y mantener las trayectorias de uso de la base de datos de tal forma que los datos puedan ser accedidos rápidamente
- Manejar los datos de acuerdo a las peticiones de los usuarios
- Registrar el uso de las bases de datos.

En la figura 1 se muestra un resumen de lo explicado hasta ahora.

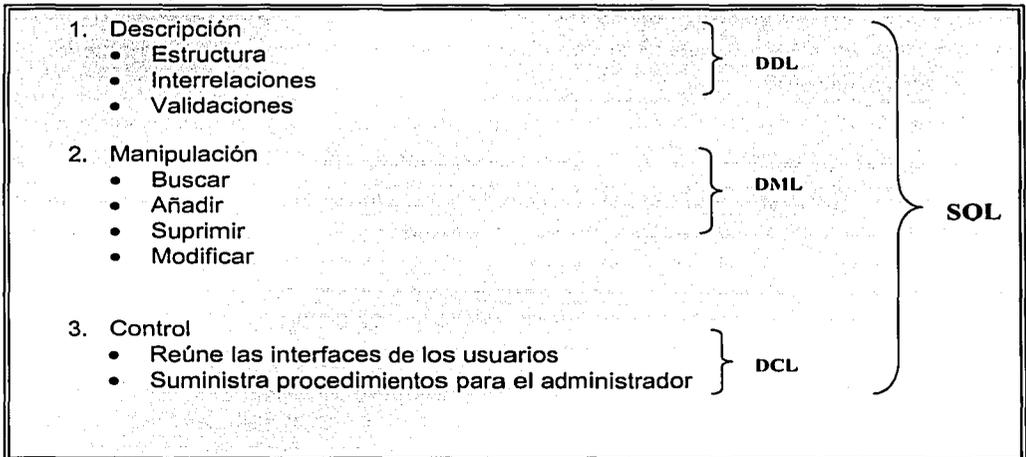


Figura 1. Funciones de un DBMS

1.3.2.2 ARQUITECTURA DE UN DBMS

Existen diferentes niveles de abstracción para simplificar la interacción de los usuarios con el sistema, estos son: el nivel interno, el conceptual y el externo, los cuales se muestran en la figura 2.

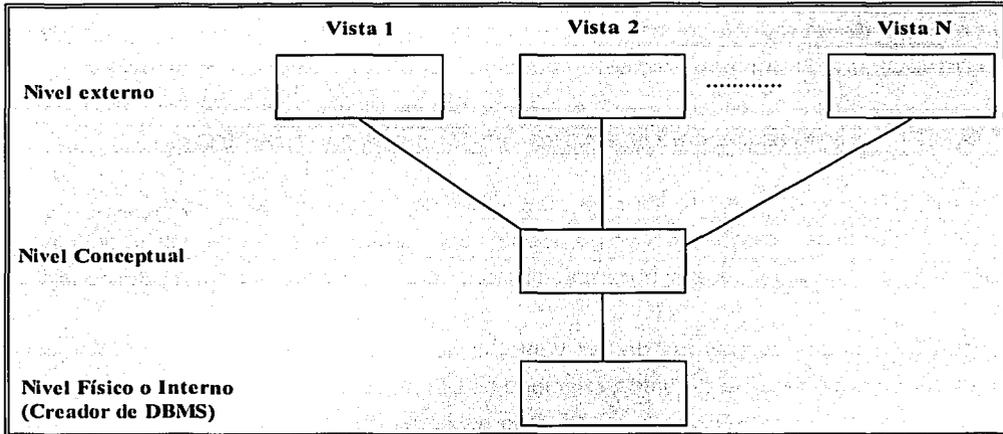


Figura 2. Niveles de Abstracción de información

En la figura anterior podemos observar la relación existente entre los niveles de abstracción en un DBMS y a continuación se explica la función que tiene cada uno.

Nivel físico

El nivel físico es la representación del nivel más bajo de abstracción, en éste se describe en detalle la forma de como se almacenan los datos en los dispositivos de almacenamiento.

Nivel conceptual

El nivel conceptual es el siguiente nivel más alto de abstracción, describe qué datos son almacenados realmente en la base de datos y las relaciones que existen entre los mismos, describe la base de datos completa en términos de su estructura de diseño. El nivel conceptual de abstracción lo usan los administradores de bases de datos, quienes deben decidir qué información se va a guardar en la base de datos.

Nivel de visión

El nivel de visión es el nivel más alto de abstracción, es lo que el usuario final puede visualizar del sistema terminado, muestra sólo una parte de la base de datos al usuario acreditado para verla.

Un ejemplo de un manejador de bases de datos de gran importancia en la actualidad es PostgreSQL, por lo que hablaremos un poco de sus características a continuación:

1.3.2.3 MANEJADOR DE BASES DE DATOS POSTGRESQL

Al igual que todo el software libre, el manejador de bases de datos PostgreSQL cuenta con dos ventajas claras: un código fuente optimizado que puede ser modificado y adaptado, y una baja inversión por implantación, ya que no existen costos por licencia.

PostgreSQL ofrece la mayoría de las ventajas que otros programas comerciales tienen. Es un manejador de base de datos relacionales orientado a objetos e incorpora casi todas las funcionalidades de SQL, incluyendo tipos de datos definidos por el usuario y una gran variedad de transacciones.

La interfaz más usada suele ser PHP, aunque algunas aplicaciones más complejas pueden llegar a requerir el uso de programas CGI de PERL. También se pueden acceder sus funciones desde lenguajes como C, C++, Java, ODBC, Python y otros.

Para el desarrollo de bases de datos se cuenta con la herramienta Pgaccess, que brindan una poderosa interfaz gráfica que acorta los tiempos de desarrollo. Además, esta herramienta facilita la generación de reportes, una tarea fundamental cuando se administran bases de datos.

A continuación se presentará a detalle las definiciones del modelo de datos, los fundamentos del modelo relacional en términos del modelo anterior y posteriormente se analizan a detalle los conceptos y definiciones del álgebra relacional.

1.3.3 MODELO DE DATOS

Para definir lo que es un modelo de datos comenzaremos explicando el concepto de modelo.

Modelo es una representación de la realidad que contiene las características generales de algo que se va a realizar Ullman1999¹:

Ahora bien, el concepto de modelo de datos se define como sigue: Una colección de herramientas conceptuales para describir los datos, las relaciones que existen entre ellos, la semántica asociada a los datos y restricciones de consistencia.

Según Codd, en Silberschatz²: El modelo de datos es una combinación de tres componentes:

- ❖ Una colección de estructuras de datos (los bloques constructores de cualquier base de datos que conforman el modelo).
- ❖ Una colección de operadores o reglas de inferencia, los cuales pueden ser aplicados a cualquier instancia de los tipos de datos listados en una estructura de datos, para consultar o derivar datos de cualquier parte de estas estructuras en cualquier combinación deseada.
- ❖ Y por último una colección de reglas generales de integridad, las cuales explícita o implícitamente definen un conjunto de estados consistentes --estas reglas algunas veces son expresadas como reglas de insertar-actualizar-borrar --.

Por otro lado, existen fundamentalmente tres alternativas disponibles para diseñar las bases de datos: *el modelo jerárquico, el modelo de red y el modelo relacional*, que explicaremos a continuación.

¹ Ullman1999

Ullman, Jeffrey y Widom, Jennifer, *Introducción a los Sistemas de Bases de Datos*. Editorial Prentice Hall, México 1999, ISBN: 970-17-0256-5.

² Silberschatz

Silberschatz, Abraham, Korth, Henry F. y Sudarshan, S., *Fundamentos de Bases de Datos*, tercera edición, Editorial McGraw Hill, España 1998, ISBN: 0-07-044756-X.

Modelo Jerárquico: Este modelo puede representar dos tipos de relaciones entre los datos: relaciones de uno a uno y relaciones de uno a muchos.

Modelo De Red: Este modelo permite la representación de muchos a muchos, de tal forma que cualquier registro dentro de la base de datos puede tener varias ocurrencias superiores a él.

Modelo Relacional: Este modelo es el más empleado en la práctica, debido a las ventajas que ofrece sobre los dos modelos anteriores. En esta tesis éste es el modelo empleado y por lo tanto el que se explica a continuación.

1.3.3.1 MODELO RELACIONAL

Un sistema de información de bases de datos relacional se organiza en forma de tablas, las tablas se organizan en renglones y columnas, cada renglón se denomina registro y éste registro contiene información referente a una instancia³, cada columna se denomina campo y es información de un solo tipo para todas las instancias. De esta forma las tablas representan la información de una forma más compacta y es posible acceder a la información contenida en dos o más tablas.

Como tema importante del modelo relacional hablaremos del Modelo Entidad Relación donde explicaremos los conceptos de relación, entidad, atributo y dominio.

1.3.3.1.1 MODELO ENTIDAD-RELACIÓN

El modelado de datos entidad-relación (E-R) se basa en una percepción del mundo real, la cual está formada por objetos básicos llamados entidades y las relaciones entre estos objetos, además de las características de estos objetos llamados atributos.

Una *entidad* es un objeto que existe y se distingue de otros objetos de acuerdo a sus características llamadas atributos.

³ Instancia. Estado que presenta una base de datos en un tiempo dado.

Una entidad se caracteriza y distingue de otra por los *atributos*, en ocasiones llamadas propiedades. Los atributos de una entidad pueden tomar un conjunto de valores permitidos al que se le conoce como dominio del atributo. Así cada entidad se describe por medio de un conjunto de parejas formadas por el atributo y el valor de dato. Habrá una pareja para cada atributo del conjunto de entidades. Por otra parte una *relación* es la asociación que existe entre dos a más entidades y la cantidad de entidades en una relación determina el *grado* de la relación.

Es posible además analizar el modelo relacional en términos de tres componentes importantes: *estructuras de datos*, *integridad* y *operadores*; estos tres elementos se explican a continuación.

Estructuras de Datos

El primer paso para crear una base de datos, es planificar el tipo de información que se quiere almacenar, teniendo en cuenta dos aspectos: la información disponible y la información que necesitamos.

La planificación de la estructura de la base de datos, en particular de las tablas, es vital para la gestión efectiva de la misma. El diseño de la estructura de una tabla consiste en una descripción de cada uno de los campos que componen el registro y los valores o datos que contendrán cada uno de esos campos. La definición de un campo requiere: el nombre del campo, el tipo de campo, el ancho del campo, etc.

Generalmente los diferentes tipos de campos que se pueden almacenar son los siguientes: Texto (caracteres), Numérico (números), Fecha / Hora, Lógico (informaciones lógicas si/no, verdadero/falso, etc.) e imágenes.

Por lo tanto, el principal aspecto a tener en cuenta durante el diseño de una tabla es determinar claramente los campos necesarios, definirlos en forma adecuada con un nombre especificando su tipo y su longitud.

Integridad

El concepto de integridad se refiere a la seguridad de que una información perteneciente a la base de datos no ha sido alterada, borrada, reordenada, copiada, etc.

Los elementos que determinan la integridad para el modelo relacional son: la llave primaria, la llave foránea, los valores nulos y un par de reglas de integridad que mencionaremos enseguida:

- Llave Primaria: Columna o grupo de columnas que identifica de manera única a cada renglón de la tabla. Símbolo: PK
- La llave primaria cumple con las siguientes características:
- No debe aceptar valores nulos
- No pueden existir dos renglones con el mismo valor de la llave primaria
- No puede cambiarse este valor.
- PK asignada por el usuario (PK UA) De esta forma el usuario proporciona el valor para la llave primaria del registro insertado.
- PK asignada por el sistema (PK SA) El sistema proporciona automáticamente el valor de la llave primaria cuando el usuario inserta un registro.
- Llave foránea. Es una columna o grupo de columnas que es llave primaria en alguna otra tabla.
- Nulo. Un nulo significa ausencia de dato.

Las dos *reglas de integridad* tienen que ver precisamente con los conceptos antes mencionados y son:

- ❖ *Integridad de Relaciones*.- Donde ningún atributo que forme parte de una *llave primaria* puede aceptar *valores nulos*.
- ❖ *Integridad Referencial*.- La integridad referencial es un sistema de reglas que utilizan la mayoría de las bases de datos relacionales para asegurarse que los registros de tablas relacionadas son válidos y que no se borren o cambien datos relacionados de forma accidental produciendo errores de integridad.

Existen además dos formalismos importantes para el manejo de datos en el ámbito del modelo relacional, estos son el Álgebra relacional y el Cálculo relacional, de los que hablaremos a continuación:

- El *álgebra relacional* proporciona una serie de operaciones que pueden usarse para decir al sistema cómo *construir* la relación deseada a partir de las relaciones de la base de datos.
- El *cálculo relacional* proporciona una notación para formular la *definición* de la relación deseada en términos de las relaciones de la base de datos

Operadores

Los operadores del modelo relacional son de dos tipos: *operadores de actualización* y los *operadores del álgebra relacional*.

Para indicar cuáles son los operadores del álgebra relacional, primero definiremos las operaciones válidas dentro de los *operadores de actualización*, estas son; borrar, agregar o modificar. El manejo de las *llaves primarias* y *foráneas* incide directamente en procurar que no se violen las reglas de integridad, al determinar cómo han de manejarse los operadores de manera que al aplicar cualquiera de estas operaciones no se produzcan inconsistencias.

Ahora bien, se define como *álgebra relacional* a un lenguaje formal con una serie de operadores que trabajan sobre una o varias relaciones para obtener otra relación resultado, sin que cambien las relaciones originales. Tanto los operandos como los resultados son relaciones, por lo que la salida de una operación puede ser la entrada de otra operación. Esto permite anidar expresiones del álgebra, del mismo modo que se pueden anidar las expresiones aritméticas. A esta propiedad se le denomina *cerradura*⁴.

Por otra parte, para verificar si nuestro modelo es funcional y está listo para ser implementado se utiliza el método de normalización, el cual está basado en reglas llamadas formas normales, tema que se explica a continuación.

1.3.3.1.2 NORMALIZACIÓN

El modelo conceptual de datos obtenido mediante la técnica de entidad-relación se refina y convierte en un modelo lógico relacional, utilizando la normalización, lo que ofrece como

⁴ *Cerradura*: Las relaciones son cerradas bajo el álgebra, del mismo modo que los números son cerrados bajo las operaciones aritméticas.

resultado el conjunto de tablas a implantar en la base de datos. Su finalidad es reducir las inconsistencias y redundancias de los datos, facilitar el mantenimiento y evitar las anomalías en las manipulaciones de datos, su objetivo es obtener un modelo lógico normalizado que represente las entidades normalizadas y las interrelaciones existentes entre ellas.

El procedimiento de normalización consiste en someter a las tablas que representan entidades a un análisis formal, ya que una relación no normalizada es una relación que contiene uno o más grupos repetitivos, es decir, contienen datos redundantes. Si se cuenta con un alto nivel de normalización, se cuenta con calidad en la organización de los datos y menor peligro para la integridad de los mismos. Este procedimiento consiste en ir alcanzando formas normales de manera que se garantice que la representación de las relaciones y las entidades sean eficientes y acordes con la realidad, permitiendo responder adecuadamente consultas a la base de datos.

Todo el proceso se basa en que una primera relación universal plantearía enormes problemas de redundancia, consistencia e integridad de los datos, por lo que es necesario mejorar las relaciones. Estas mejoras deben dar como resultado tablas equivalentes y mejores que sus respectivas originales, y poseer siempre tres propiedades: conservación de la información (de atributos y de registros - tuplas -), conservación de dependencias y mínima redundancia de los datos. Las mejoras introducidas obligan a plantear hasta que Forma Normal es necesario llegar, es decir, a que nivel de depuración. Normalmente, es recomendable alcanzar la máxima Forma Normal.

Básicamente, las reglas de normalización son las técnicas para prevenir las anomalías en las tablas. Dependiendo de su estructura, una tabla puede estar en primera forma normal, segunda forma normal o en cualquier otra, existen cinco formas normales y podemos observarlas en la figura 3.

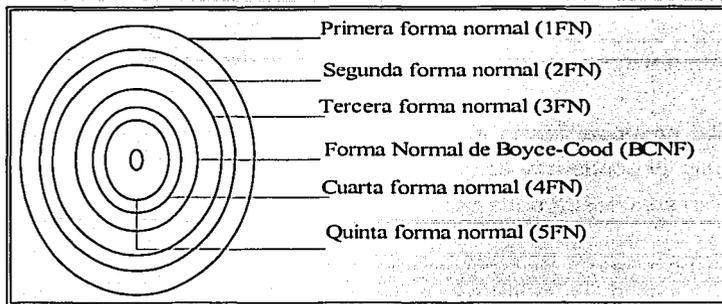


Figura 3. Formas Normales

La figura anterior muestra la secuencia de las formas normales, en principio debemos empezar la normalización por la parte externa y continuar hacia las internas.

A continuación se presenta una breve explicación de las tres primeras formas normales ya que son las que se utilizan en esta tesis.

Primera forma normal

Se considera que una relación se encuentra en la primera forma normal cuando cumple lo siguiente:

- Las celdas de las tablas poseen valores simples y no se permiten grupos ni arreglos repetidos como valores, es decir, contienen un solo valor por cada celda.
- Todos los ingresos en cualquier columna (atributo) deben ser del mismo tipo.
- Cada columna debe tener un nombre único, el orden de las columnas en la tabla no es importante.
- Dos filas o renglones de una misma tabla no deben ser idénticas, aunque el orden de las filas no es importante.
- Por lo general la mayoría de las relaciones cumplen con estas características, así que podemos decir que la mayoría de las relaciones se encuentran en la primera forma normal.

Segunda forma normal

Para definir la segunda forma normal requerimos saber qué es una *dependencia funcional*: Y esta dependencia consiste en identificar qué atributos dependen de otro(s) atributo(s), como se muestra en la siguiente figura:

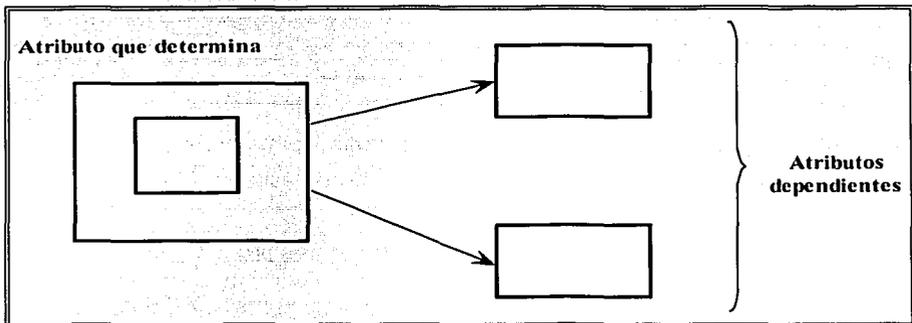


Figura 4. Relación entre Atributos.

- ✓ **Definición formal:** Una relación se encuentra en segunda forma normal, cuando cumple con las reglas de la primera forma normal y todos sus atributos que no son llaves dependen por completo de la llave. De acuerdo con esta definición, cada tabla que tiene un atributo único como llave, esta en segunda forma normal.

Tercera forma normal

Para definir formalmente la 3FN necesitamos definir dependencia transitiva: En una tabla bidimensional que tiene por lo menos 3 atributos (A,B,C) en donde $A \rightarrow B$, $B \rightarrow C$, pero B no determina a A.

- ✓ **Definición formal:** Aplicar la 3FN en una relación R consiste en eliminar la dependencia transitiva que queda en una segunda forma normal; en pocas palabras una relación está en tercera forma normal si está en segunda forma normal y no

existen dependencias transitivas entre los atributos, nos referimos a dependencias transitivas cuando existe más de una forma de llegar a referencias a un atributo de una relación.

Forma normal de Boyce Codd (FNBC)

- ✓ **Definición formal:** Una relación R esta en FNBC si y solo si cada determinante es una llave candidato.

Siendo x e y atributos de R, a x se le denomina determinante ya que x determina el valor de y . Dónde llave candidato es una superllave mínima y a su vez una superllave es un conjunto de atributos, que identifican como única a una entidad.

La 2FN y la 3FN eliminan las dependencias parciales y las dependencias transitivas de la llave primaria. Pero este tipo de dependencias todavía pueden existir sobre otras llaves candidatas, si éstas existen. La FNBC es más fuerte que la 3FN, por lo tanto, toda relación en FNBC está en 3FN.

La violación de la FNBC es poco frecuente ya que se da bajo ciertas condiciones que raramente se presentan. Se debe comprobar si una relación viola la FNBC si tiene dos o más claves candidatas compuestas que tienen al menos un atributo en común.

Uno de los componentes importantes del modelo de datos es el Diccionario de datos, ya que en él se define la estructura de la Base, este tema se explica a detalle a continuación.

1.3.4 DICCIONARIO DE DATOS

El diccionario de datos contiene las características de las entidades y atributos, que definen la estructura de la Base de Datos. El objetivo del diccionario de datos es:

Facilitar el control de cada una de las entidades y atributos que forman parte de la estructura de Base de Datos del Sistema.

Controlar dinámicamente la estructura de la interfase al usuario, para las diferentes pantallas del sistema.

TESIS CON
FALLA DE ORIGEN

Contenido de un registro del diccionario

Un diccionario de datos contiene datos del administrador de la base de datos, así como una descripción externa, conceptual e interna de la Base de datos. El diccionario de datos hace una descripción de entidades (registros), atributos (campos), referencias cruzadas, origen y significado de los elementos de los datos.

Un diccionario de datos también contiene sinónimos, homónimos, códigos de autorización y seguridad; además de especificar qué esquemas externos son usados y por qué programas, quienes son los usuarios y que autorizaciones tienen.

Ventajas de utilizar un Diccionario de Datos

El uso de un diccionario de datos facilita el manejo de la información tanto en sistemas chicos como en sistemas de gran tamaño, ya que los sistemas al sufrir cambios continuos dificultan el manejo de todos los datos, por eso se registra la información de los datos almacenados, ya sea sobre hoja de papel o usando procesadores de texto.

Para asignarle un solo significado a cada uno de los elementos y actividades del sistema, los diccionarios de datos proporcionan asistencia para asegurar significados comunes y registrando detalles adicionales relacionados con el flujo de datos en el sistema, de tal manera que todo pueda localizarse con rapidez.

Para documentar las características del sistema, incluyendo partes o componentes así como los aspectos que los distinguen. También es necesario saber bajo que circunstancias se lleva a cabo cada proceso y con que frecuencia ocurren. Produciendo una comprensión más completa. Una vez que las características están articuladas y registradas, todos los participantes en el proyecto tendrán una fuente común de información con respecto al sistema.

El diccionario de datos, también facilita el análisis de los detalles con la finalidad de evaluar las características y determinar donde efectuar cambios en el sistema, además determina si son necesarias nuevas características o si están en orden los cambios de cualquier tipo.

Por otra parte el diccionario de datos ayuda a localizar errores y omisiones en el sistema, detectan dificultades, y las presentan en un informe.

1.4 METODOLOGÍA PARA EL DESARROLLO DE SISTEMAS

1.4.1 DEFINICIÓN DE METODOLOGÍA

La metodología para el desarrollo de sistemas se define como: "un conjunto de procedimientos, técnicas y herramientas, que nos permiten estandarizar y sistematizar un sistema".

Las etapas de esta metodología son:

- ❖ Planeación
- ❖ Análisis
- ❖ Diseño
- ❖ Desarrollo

TESIS CON
FALLA DE ORIGEN

En la figura 5 mostraremos como están relacionadas estas etapas.

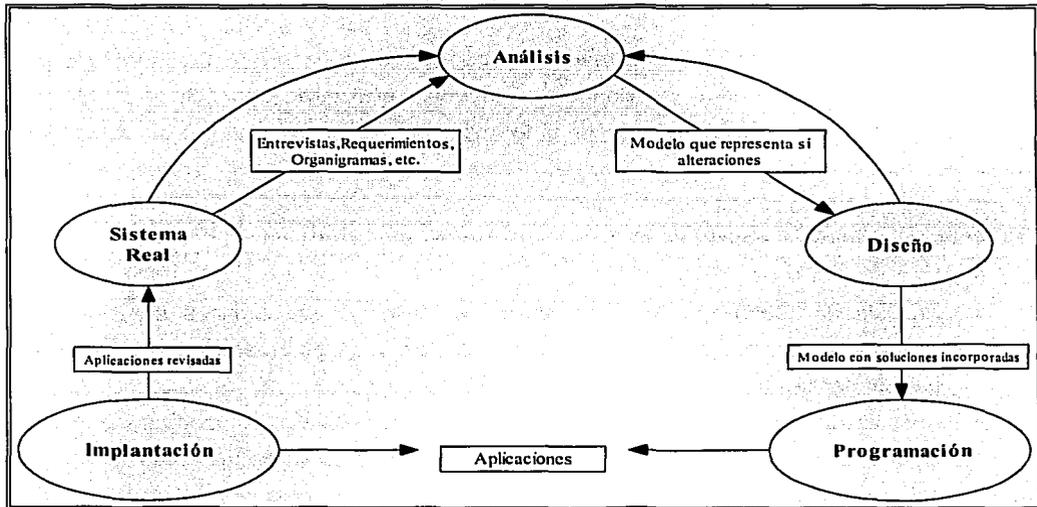


Figura 5. Ciclo de vida del desarrollo de sistemas.

En la figura anterior podemos observar el ciclo de vida de un sistema, se inicia con el análisis del mismo y partimos con un modelo sencillo hacia el diseño, después creamos un segundo modelo incluyendo las posibles soluciones del problema y comenzamos con la programación, cabe destacar que en la etapa de diseño es posible regresar a la etapa de análisis y así incrementar la información. Cuando ya se cuenta con un prototipo del sistema podemos realizar las pruebas pertinentes y afinar los detalles que requieran atención para entregar el producto terminado.

A continuación se describen cada una de las etapas que integran este ciclo de vida.

1.4.2 PLANEACIÓN

En esta etapa se involucra un análisis de la situación actual del sistema, así como de las necesidades y objetivos del mismo. Se realizan entrevistas con los usuarios del sistema en niveles estratégicos, técnicos y operativos, además de identificar dentro de toda la información las entidades que quedarán involucradas.

1.4.3 ANÁLISIS

Para esta etapa se analiza la información, basándonos en un proceso sistemático para razonar un problema dentro de una organización y encontrar una solución detallada. Además, el análisis permite establecer si la implantación de un sistema de información automatizado es posible.

El proceso de análisis consta de tres etapas fundamentales, que son:

- **Análisis de Objetivos:** En esta etapa se define el problema a solucionar, se entiende y razona para plantear una solución y se establecen los lineamientos para el desarrollo del sistema.
- **Análisis de Requerimientos:** Aquí se estudia con detalle la conducta que debe tener el sistema para ser útil a la organización, también se definen las restricciones del medio en el cual operará normalmente el sistema.

- **Análisis de Implantación:** Por último se determinan con precisión los costos y tiempos de desarrollo y se define el plan de acciones necesarias para diseñar, construir, probar e implantar el sistema.

En cada una de las etapas se define un conjunto de productos a obtener, y que están orientados a reflejar fielmente la operación esperada del sistema.

1.4.4 DISEÑO

En esta tercera etapa se decide lo que se tiene que hacer y la manera de hacerlo, es decir, se determinan las condiciones y restricciones tecnológicas necesarias para satisfacer la conducta esperada del sistema computacional.

El diseño es una actividad que se realiza una vez planteado el problema y construidos los modelos que representan el proceso. En esta etapa se incorporan los componentes en equipo y programas que forman parte del sistema, proporcionando así un anteproyecto para la implementación.

1.4.5 DESARROLLO

Finalmente, el desarrollo es una actividad que inicia una vez que se ha determinado lo que se tiene que hacer y la manera de hacerlo. En esta etapa se realiza la programación y las pruebas para verificar el correcto funcionamiento del sistema de acuerdo a lo planteado en el análisis y diseño.

1.5 SISTEMAS DE INFORMACIÓN VÍA INTERNET

Actualmente las empresas deben tener sistemas de información en línea para responder a las necesidades de la sociedad, siendo éste el medio idóneo para el intercambio de información, distribución de datos de todo tipo e interacción con otras personas. A continuación se presenta una breve introducción a los conceptos, medios (Browser, Página Web, etc.), y protocolos que proporcionan la transmisión de información en Internet.

TESIS CON
FALLA DE ORIGEN

1.5.1 INTERNET

Internet es tanto un conjunto de comunidades como un conjunto de tecnologías, y su éxito se puede atribuir a la satisfacción de las necesidades básicas de la comunidad y a la utilización de Internet de un modo efectivo para impulsar el crecimiento de la infraestructura de Internet. Es a la vez una oportunidad de difusión mundial, un mecanismo de propagación de la información y un medio de colaboración e interacción entre los individuos y sus computadoras, independientemente de su localización geográfica, en la figura 6 mostramos la red Internet.

Una característica importante de la red Internet es su descentralización, nadie gobierna Internet, así que cada red conectada conserva su independencia. Esto hace que sea necesario administrar una serie de recursos comunes distribuidos en todo el mundo, por lo que se ha creado una especie de servicio público denominado NIC ("Network Information Center"), el cual se encarga de la asignación de direcciones IP y de los registros de nombres de dominio (URL). El número total de usuarios de la Internet asciende hoy en día a varios millones, y su crecimiento es exponencial. Este alto nivel de conectividad ha creado un grado de comunicación, colaboración, acceso a la información e intercambio de recursos sin precedentes en la Historia de la Humanidad.

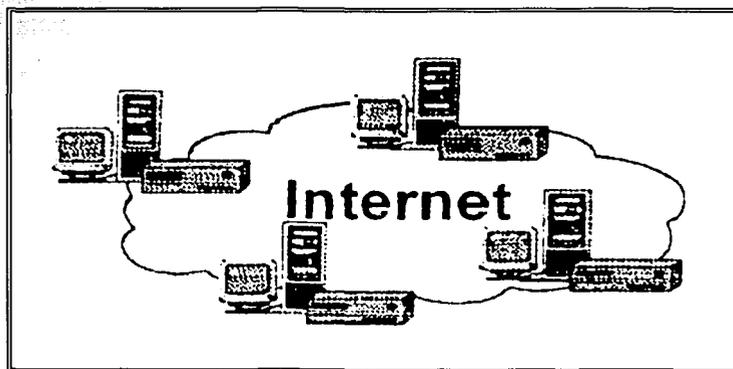


Figura 6. La red Internet

TESIS CON
FALLA DE ORIGEN

1.5.2 ARQUITECTURA CLIENTE SERVIDOR

La arquitectura Cliente-Servidor es utilizada por todas las aplicaciones de Internet y consta de los siguientes elementos:

- Un **cliente**, el cual funciona en la computadora del usuario, se comunica con el servidor remoto (computadora remota), y solicita a éste información.
- El **servidor**, que es una computadora remota tiene como tarea proporcionar la información solicitada por el cliente.
- Un **formato de comunicación**, que establece la manera en que el cliente y el servidor se comunican.

Los sistemas Cliente-Servidor pueden ser de muchos tipos, dependiendo de las aplicaciones que el servidor pone a disposición de los clientes. Entre ellos, existen:

- Servidores de Impresión, mediante el cual los usuarios comparten impresoras.
- Servidores de Archivos, con el cual los clientes comparten discos duros
- Servidores de Bases de Datos, donde existe una única base de datos
- Servidores de páginas Web, mediante los cuales podemos publicar información en Internet y realizar consultas de la misma.

A continuación profundizaremos en las características de los servidores de paginas Web.

1.5.3 SERVIDORES DE PAGINAS WEB

Los Servidores Web son aquellos que permiten a los clientes compartir datos, documentos y multimedia en formato HTML. Aunque se basa en la tecnología Cliente-Servidor, el servidor Web aporta ventajas adicionales en aspectos muy importantes como son:

- ❖ El Web se crea normalmente como un sistema abierto al que cualquiera puede contribuir y acceder desde cualquier punto de la red de la empresa; normalmente no requieren claves de acceso.

TESIS CON
FALLA DE ORIGEN

- ❖ Los Servidores Web regresan información con un simple clic del ratón a través de un proceso de hipervínculo.
- ❖ La información entregada puede ser de cualquier tipo (datos, documentos, multimedia, etc...), gracias a la utilización de los estándares de Internet. Esa información es únicamente de lectura pues el usuario no puede hacer cambios en el documento original de los datos.
- ❖ Dado que el servidor de Web es de fácil acceso, ello hace posible publicar información de forma instantánea en toda la organización mediante su almacenamiento en el servidor.
- ❖ Un servidor Web en la *Intranet*⁵ puede servir la misma copia de un archivo o documento a toda la organización, de la misma forma que un único servidor web en Internet puede servir al mundo entero
- ❖ La amplitud de la red suele ser mayor que otros sistemas Cliente- Servidor, pues la mayor parte de Intranet se construye sobre redes de Área Extensa (Wide Area Networks, WAN por sus siglas en inglés), más que sobre redes locales (Local Area Networks, LAN por sus siglas en inglés).

Un ejemplo de uno de los servidores de páginas web más importantes en la actualidad es Apache, así que hablaremos de sus características principales a continuación:

1.5.3.1 SERVIDOR WEB APACHE

Apache se basó originalmente en codificación e ideas basadas en el servidor HTTP. Esto ha desencaminado en un sistema que puede rivalizar (y probablemente sobrepasar) a casi cualquier otro servidor basado en UNIX - HTTP en cuanto a funcionalidad, eficacia y rapidez. Es uno de los mejores servidores de Webs utilizados en la red Internet desde hace mucho tiempo, siendo uno de los mayores triunfos del software libre adaptado a los nuevos protocolos.

⁵ *Intranet*. - Es una red de área local cuyos usuarios navegan dentro de la misma como si lo hiciesen a través de Internet.

Entre sus características destacan:

- ✓ Multiplataforma.
- ✓ Implementa los últimos protocolos, aunque se base en el http.
- ✓ Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo y con la API de programación de módulos.
- ✓ Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para solución de los mismos.
- ✓ Se desarrolla de forma abierta.
- ✓ Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.

Por otro lado, Apache es uno de los Servidores de páginas web más utilizados, posiblemente porque ofrece instalaciones sencillas para sitios pequeños y si se requiere es posible expandirlo hasta el nivel de los mejores productos comerciales. Cuando se requiere una aplicación implica lo siguiente:

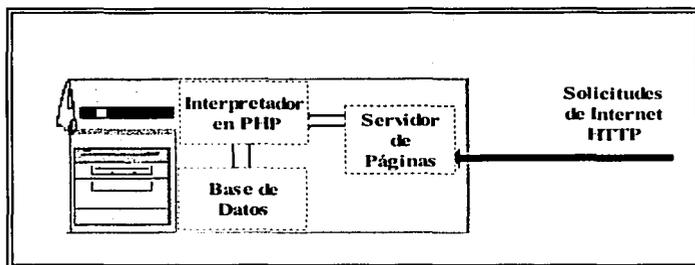


Figura 7. Aplicación en PHP

Cuando el servidor de páginas (Apache) recibe la requisición para "x" página, éste reconoce cuándo debe enviar un documento estático (HTML) o ejecutar algún tipo de aplicación, en la figura 3 se puede observar que la solicitud de "x" página, invoca un programa en Perl o PHP y este a su vez solicita información a una base de datos. Si no se tienen los suficientes recursos en cuanto a memoria y procesadores se refiere, seguramente caerá el servidor de páginas o bien se quemará el "Host" (computadora física)

TESIS CON
FALLA DE ORIGEN

por la demanda excesiva. Apache tiene tanto tiempo de desarrollo que han sido desarrolladas diferentes soluciones para evitar estas ineficiencias, algunas son:

- Es capaz de utilizar otros interpretadores y lenguajes como "Tcl", "PhP" y "Python".
- Puede conectarse directamente a una Base de datos.

1.5.4 PROTOCOLOS DE COMUNICACIÓN

En la actualidad vivimos la era de las comunicaciones, la cual se caracteriza por el gran número de tecnologías nuevas y en desarrollo, integradas en equipos y sistemas que influyen profundamente en la Industria de las comunicaciones y la sociedad. Sin embargo, para establecer dicha comunicación, contamos con los protocolos de comunicación comerciales los cuales utilizamos muchas veces aun sin darnos cuenta, es decir, nos ayudan a realizar tareas como lo son Internet, una transferencia por módem o una simple comunicación a un servicio en línea de algún banco.

Antes de introducirnos al tema de protocolos de comunicación será necesario definir el concepto de red y protocolo previamente.

Se entiende por red, un grupo de computadoras y dispositivos asociados, comunicados entre sí con el propósito de intercambiar y compartir datos, archivos, programas u operaciones.

En cuanto al término protocolo se define como un acuerdo que establece las reglas de cómo las computadoras deben comunicarse a través de las redes para reducir al mínimo los errores de transmisión. Por tanto diferentes tipos de redes se pueden comunicar a pesar de sus diferencias, porque los protocolos de cada una de ellas proveen formas y métodos de comunicación.

A continuación hablaremos de los Protocolos de Comunicación más importantes y/o los más comerciales, quienes determinan el contexto de intercambio de mensajes (correo electrónico), de las conexiones remotas⁶ (telnet), o de la transferencia de archivos FTP, entre otras actividades de la red.

⁶ Conexiones remotas.- Hace referencia a computadoras o redes no importando donde estén localizadas, tan solo que sean accesibles a través de Internet.

Entre los protocolos más importantes tenemos los siguientes:

TELNET (*Telecommunications Network*)

Es un protocolo estándar de sesión de trabajo remoto en internet, que permite la conexión hacia una computadora remota utilizando el principio de cliente-servidor, donde la computadora-cliente solicita abrir una sesión en la computadora remota (servicio o host). Durante la negociación, ambas computadoras se ponen de acuerdo en los parámetros que regirán la sesión. En conclusión podremos decir que telnet es un método para conectarse a una computadora remota mediante Internet, permitiendo utilizar programas y datos desde la computadora local.

FTP (*File Transfer Protocol*)

Es un protocolo que permite la transferencia de archivos, entre computadoras de distintos tipos y distintos sistemas operativos en el universo de Internet, es decir si se desea enviar información a algún usuario de Internet, se le puede enviar un correo electrónico, adjuntándole archivos, pero esto es bueno solo en el caso en que se desee enviar poca información. Si se desea transferir un archivo extenso es más conveniente usar FTP.

SMTP (*Simple Mail Transfer Protocol*)

Es un protocolo que se utiliza para dar servicio de correo electrónico desde la PC de un usuario sobre la conexión TCP/IP a un servidor remoto. Este protocolo define un conjunto de reglas permitiendo que dos programas se puedan enviar o recibir sobre la red. El protocolo SMTP permite realizar direccionamiento de los correos, es decir si una persona trabaja en distintas empresas y tiene asignada distintas direcciones de correo puede hacer que todos sus mensajes sean enviados a una misma casilla de correo en forma automática.

HTTP (*Hypertext Transfer Protocol*)

Es un protocolo que controla las peticiones y servicios de documentos HTML (HiperText Markup Language) además se utiliza para manejar la transferencia de páginas de hipertexto en la World Wide Web (WWW por sus siglas en inglés). Se entiende por hipertexto al texto que está codificado usando un sistema estándar llamado (HTML) disponible (archivo de texto, gráfico o sonidos).

PPP (*Point To Point Protocol*)

Es un protocolo que permite a una computadora establecer comunicación con una red de datos remota, convirtiéndose en un nodo de dicha red y pudiendo usar todos sus servicios tal cual lo haría si esa computadora estuviese conectada a la red directamente. Para lograr lo anterior la computadora utiliza un puerto serial, un módem y, como medio físico de enlace, una línea telefónica convencional.

DNS (*Domain Name System*)

El Sistema de Nombres de Dominios⁷ (DNS por sus siglas en inglés) es un conjunto de protocolos y servicios sobre una red TCP/IP, el DNS, se trata de un servicio de búsqueda de los datos de uso general, que se distribuyen y multiplican. Se utilizan principalmente para buscar direcciones IP en sistemas anfitriones (hosts) tomando como base los nombres de éstos, además permite a los usuarios de red utilizar nombres sencillos para comunicarse con otros equipos, en vez de memorizar y usar sus direcciones IP. Este sistema es muy usado en Internet y en muchas de las redes privadas actuales. Las utilerías como: browsers, servidores de Web, FTP y Telnet; utilizan DNS.

CGI (*Common Gateway Interface*)

La creación de programas del tipo CGI es realizada bajo el lenguaje de programación PERL o PHP, lenguaje que es utilizado muchas veces para establecer comunicación entre la computadora de la persona que se conecta para ver la página web (cliente) y la computadora donde se encuentran almacenadas las páginas (servidor). Concretamente,

⁷ Dominio.-Componente de la dirección de una computadora conectada a una red y, por extensión, a Internet, que sirve para su identificación y ubicación geográfica. Los dominios representan a la organización que los sustenta.

permite la generación de páginas dinámicas, generadas a través de la ejecución de un programa en el servidor de Internet.

El protocolo perteneciente a la capa física más importante es:

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

TCP/IP es un protocolo que engloba una familia de protocolos de comunicación, que determinan las reglas para enviar y recibir datos a través de las redes TCP/IP se originó cuando se necesitó comunicar diferentes computadoras de diferentes fabricantes, es el protocolo de comunicación básico de Internet e Intranets, y permite el acceso público a los protocolos de red, facilitando la integración entre ambientes de cómputo logrando así la comunicación punto-a-punto entre distintas computadoras de una red.

Los dos principales componentes de TCP/IP son:

- IP (Internet Protocol): define el protocolo de enrutamiento⁸ de los paquetes en la red, permite leer los paquetes y enviarlos a su destino, determina que cantidad de datos puede entrar en cada uno. Por otro lado, se le llama paquete a un grupo ordenado de datos y señales de control transmitidos a través de la red, como subconjunto de un mensaje grande.
- TCP: es el protocolo que corta los datos en paquetes de manera tal, que la red los pueda manejar eficientemente, verifica que todos los paquetes lleguen a su destino, ordenándolos a medida que los va recibiendo con la secuencia correcta y si un paquete está dañado reconstruye los datos a su forma original. Se lo llama orientado a conexión, porque establece una conexión lógica entre hosts antes de iniciar una conversación.

Además de los protocolos de comunicación, hablaremos también de uno de los servicios de información con mayor importancia en Internet, la World Wide Web.

⁸ Enrutamiento.- Es el proceso de dirigir los paquetes por vías alternativas hasta su destino.

1.5.5 WORLD WIDE WEB

La evolución de Internet en la década de los noventa ha sido espectacular, propiciando el surgimiento de otros servicios avanzados como el World Wide Web (Tejido Mundial), siendo este una red de recursos de información.

La Web se basa en tres mecanismos para hacer que estos recursos estén listos y a disposición de la mayor audiencia posible:

- ❖ Un esquema uniforme de nombres para localizar recursos en la Web (p.ej, URIs).
- ❖ Protocolos, para acceder a recursos con nombre en la Web (p.ej,HTTP).
- ❖ Hipertexto, para navegar fácilmente entre recursos (p.ej., HTML).

Durante una sesión normal de trabajo en WWW un cliente (navegador) solicita un documento de un servidor Web y una vez obtenido lo muestra al usuario que hizo la solicitud a través de páginas Web. Si esta página contiene un enlace a otra (en el mismo o en distinto servidor), y el usuario activa el enlace, el cliente Web efectuará otra petición y mostrará la nueva página. Para ello se hablará sobre páginas web a continuación.

1.5.5.1 CONCEPTO DE PAGINA WEB

Las páginas web son los documentos con los que se construye la WWW, y su número crece exponencialmente. Una página web es un documento multimedia, ya que puede contener información de distintos tipos de medios: textos, imágenes, sonidos, animaciones, etc. Y a ello se une la posibilidad de crear enlaces con otras páginas que pueden estar ubicadas en lugares remotos, éste tipo de páginas se convierten entonces en documentos hipermedia.

Las páginas web son diseñadas para ser accedidas a través de Internet, por lo cual serán visualizadas por distintos navegadores en máquinas con características muy diversas. Por ello resulta conveniente comprobar su correcta visualización al menos en las configuraciones más habituales. Cuando esto no sea posible se deberá indicar en la página la configuración más adecuada para visualizarla correctamente.

Por sus características, las páginas web representan un medio excelente para la publicación de información multimedia en Internet. Sin embargo, el uso de las páginas web no está restringido a la red. Las páginas se pueden visualizar localmente sin necesidad de conexión a la red. Para ello basta con disponer de un navegador, y como éste es ya un componente fijo en cualquier máquina se pueden utilizar las páginas web para distribuir cualquier tipo de información multimedia.

En una página web pueden colocarse distintos tipos de elementos. Los más habituales son los siguientes:

- Texto
- Imágenes
- Objetos multimedia
- Enlaces

El texto puede aparecer en forma de párrafo o de lista. Las imágenes complementan al texto proporcionando información visual. También pueden utilizarse para hacer la página más atractiva. En cuanto a los objetos multimedia, los más utilizados son sonidos, vídeos y animaciones. Estos objetos permiten crear efectos muy atractivos.

Los enlaces pueden estar asociados a cualquiera de los elementos anteriores y son los responsables de la auténtica potencia de las páginas web. Los enlaces permiten saltar de una página a otra sin importar el lugar donde ésta se encuentre. De esta forma podemos consultar un gran número de documentos relacionados con un mismo tema y que pueden estar dispersos por todo el planeta.

Todos los elementos mencionados deben estar colocados en la página de forma que la información que transmiten sea lo más clara posible. Para ello nos podemos servir de otros elementos como son las tablas y los frames o marcos. Estos elementos los podemos considerar como contenedores, pues contienen otros elementos, y nos ayudan a conseguir el formato de visualización que deseamos para ellos.

Las páginas Web se dividen en dos tipos, las páginas estáticas y las páginas dinámicas de las que hablaremos a continuación.

TESIS CON
FALLA DE ORIGEN

1.5.5.1.1 PAGINAS ESTÁTICAS

Inicialmente, las páginas Web eran estáticas, en el sentido de que, a efectos de usuario, el único proceso realizado era el de visualización de su contenido (escrito en lenguaje HTML) por parte del explorador del cliente.

Las páginas estáticas se siguen utilizando ampliamente debido a que forman la base necesaria para la presentación de datos en muchos tipos de situaciones. También influye decisivamente la sencillez con que se pueden crear, instalar y mantener.

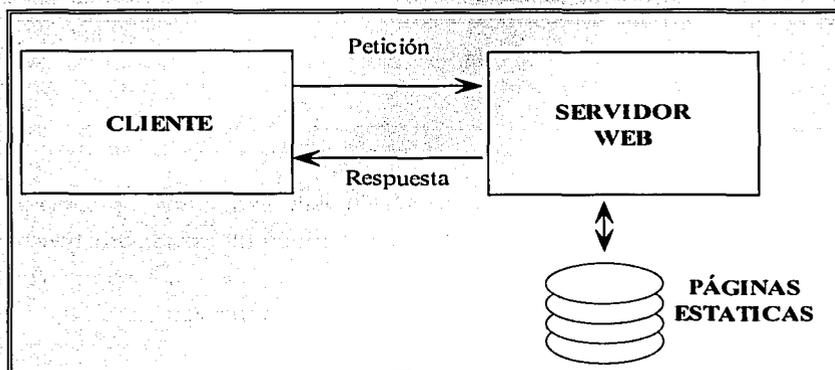


Figura 8. Páginas Estáticas

La figura anterior representa esquemáticamente la forma en que interactúan las páginas estáticas con el Cliente y el Servidor.

1.5.5.1.2 PAGINAS DINÁMICAS

En el momento en que se requiere una interacción mayor entre los usuarios y el sistema que soporta las páginas Web, surge la necesidad de reunir y procesar las peticiones del cliente con el fin de ofrecerle informaciones mejor dirigidas, escogidas y elaboradas. El procesamiento de intercambio de información entre los usuarios y el servidor de páginas Web, introdujo el concepto de páginas Web activas o dinámicas.

La capacidad de procesamiento que sustenta las páginas dinámicas se puede llevar a cabo siguiendo alguno de estos modelos:

- Procesamiento en el equipo del usuario: páginas activas en el cliente.
- Procesamiento en el equipo donde reside el servidor de Web: páginas activas en el servidor.
- Procesamiento mixto: páginas activas en el cliente + páginas activas en el servidor.

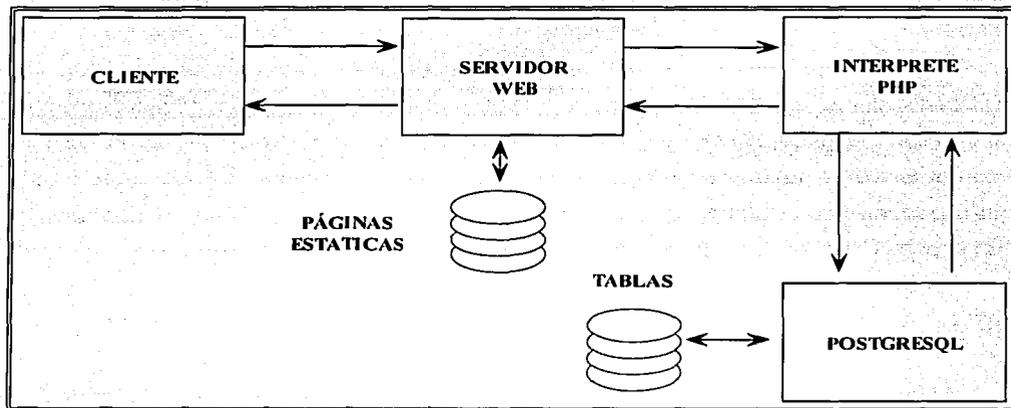


Figura 9. Páginas Dinámicas

La figura anterior representa el intercambio de información entre los usuarios y el servidor de páginas Web.

La ventaja principal de las páginas activas en el cliente es la descarga de trabajo que le proporciona al equipo servidor la posibilidad de traspasar cómputos a los usuarios. Otra importante ventaja se basa en el ahorro de comunicaciones (ancho de banda⁹) que se puede experimentar en muchas aplicaciones que ejecutan procesos en el cliente, y evitan de esta manera realizar continuos traspasos de información con el servidor de Web.

⁹ Ancho de Banda.- Velocidad a la que puede transmitir información un canal de comunicación, con independencia del soporte físico que se utiliza.

TESIS CON FALLA DE ORIGEN

1.5.6 LENGUAJES DE PROGRAMACIÓN PARA WEB

Existen diferentes lenguajes de programación que nos permiten crear de forma sencilla páginas estáticas y dinámicas en la Web, así como desarrollar aplicaciones para interactuar fácilmente con el usuario.

Entre los principales lenguajes tenemos:

HTML

HTML es el lenguaje básico para dar presentación o formato a textos en el que se desarrollan las páginas web, para que los navegadores como Netscape Navigator y MS Internet Explorer puedan desplegar las páginas a los visitantes en la web. Este formato permite la inclusión de imágenes y enlaces que el visitante puede utilizar para interactuar con la página y conseguir rápidamente la información que está buscando.

JAVA

Es un lenguaje de programación orientado a objetos, principalmente utilizado en Internet para desarrollar Applets, o pequeñas aplicaciones ejecutadas en la computadora del usuario, tales como: relojes en línea, pantallas de noticias rotativas, animaciones complejas.

JAVASCRIPT

Son lenguajes de programación basados en Java utilizados en muchas ocasiones para dar vida y movimiento a los botones programados para las páginas web. Además de las utilidades antes mencionadas, éste puede ser utilizado para realizar cálculos matemáticos, juegos, creación de páginas dinámicas en línea y pequeños programas.

SCRIPTS: Son pequeños fragmentos de código que el navegador interpreta para realizar determinadas tareas. Por ejemplo, en el caso más simple, puede buscar la fecha y hora del sistema y colocarla en la pantalla en forma de texto. En aplicaciones más sofisticadas, puede generar automáticamente complejas animaciones y efectos con el texto e imágenes.

PHP

PHP (Hypertext Preprocessor) es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor que está dirigido expresamente a construir páginas. Una de las virtudes de PHP es que puede correr como un CGI, y además contiene bibliotecas para funciones matemáticas y de bases de datos.

Un aspecto importante de PHP es que es un software libre, es decir, se tiene acceso al código fuente, se puede alterar, usar y redistribuir sin cargo alguno. Entre las características más importantes de este lenguaje se tiene que el código PHP no es un lenguaje de marcas como podría ser HTML, XML o WML, no se mezcla con las etiquetas HTML, ya que está acotado siempre por los símbolos <? y ?>, de forma que para programas compositores de páginas web son etiquetas que no soporta y las deja tal y como están. De esta forma, puede trabajarse a la vez en el diseño visual de la página y en la funcionalidad (programación) de la misma cómodamente.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, sin embargo para que las páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

A continuación se muestra un diagrama del funcionamiento de PHP

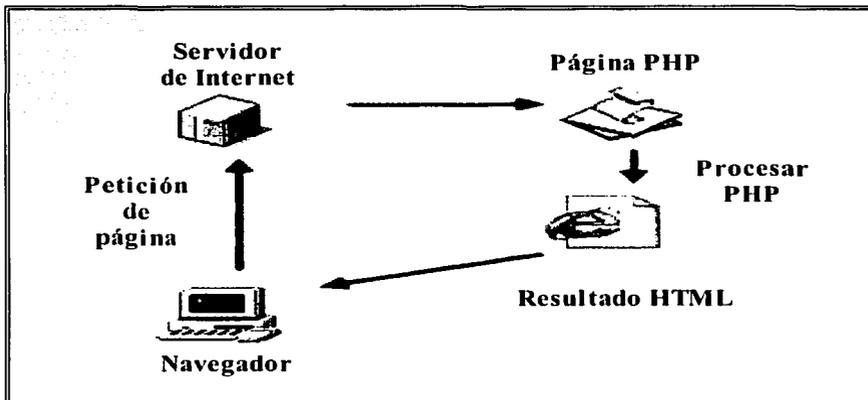


Fig. 10 Modelo funcional de PHP.

TESIS CON
FALLA DE ORIGEN

La figura 10 ejemplifica el modelo funcional de PHP, donde un programa es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página WML.

Funcionalidad

Hoy por hoy, PHP es competencia directa, dentro del campo de las páginas generadas dinámicamente, de las Active Server Pages (ASP, por sus siglas en inglés) que Microsoft implementa en su servidor web Internet Information Server (IIS, por sus siglas en inglés).

PHP ha sido desarrollado principalmente para usarse en conjunto con el servidor web Apache en cualquiera de las plataformas que éste soporta. No obstante, también puede ser compilado como un ejecutable más, para cualquiera de las plataformas en las que está soportado (Unix, Windows) y usarse como un CGI. De esta forma puede ser empleado con cualquier servidor web que soporte ejecución de CGIs en dichas plataformas. El lenguaje PHP ofrece interfaces para el acceso a la mayoría de las bases de datos comerciales y por ODBC a todas las bases de datos posibles en sistemas Microsoft, a partir de las cuales podremos editar el contenido de nuestro sitio. PHP es muy utilizado sobre todo con PostgreSQL y MySQL, bases de datos libres que en conjunto con Apache y PHP brindan una solución "libre" para hacer sitios web dinámicos de alta calidad, sin dependencia tecnológica de ninguna empresa en particular.

Además PHP soporta librerías para conectarse a un gran número de bases de datos entre ellas InterBase, dBase, mSQL, Sybase, MySQL, Oracle, Unix dbm, Informix y PostgreSQL.

También hablaremos acerca de los visualizadores que nos permiten acceder a la WWW desde cualquier computadora.

1.5.7 ACERCA DE LOS PRINCIPALES VISUALIZADORES

Para acceder al WWW se utilizan lo que se conocen como *browsers* (navegadores, visualizadores o clientes) y la información depositada en los Servidores Web de una

Intranet. El web browser permite al usuario seguir un hipervínculo o enlace, de manera que proporciona al servidor información sobre dónde está depositada la página que quiere ver, y automáticamente mandarle y ver en línea páginas web.

Los Web browsers son programas comerciales de software. Los dos de más utilización en la actualidad son los comercializados por Microsoft (Internet Explorer) y por Netscape Communications (Netscape Navigator). Ambos programas trabajan del mismo modo y ponen a disposición del usuario el mismo tipo de documentos: aquellos almacenados en formato HTML y otros.

Por último es necesario hacer mención del sistema operativo Linux ya que esta tesis estará basada en esta plataforma de desarrollo.

1.6 SISTEMA OPERATIVO LINUX

El sistema Operativo Linux se inició como el proyecto universitario de, Linus Torvalds, siendo en principio, una idea para desarrollar un Sistema Operativo basado en MINIX¹⁰. En la actualidad Linux se conoce como un *clon* de UNIX que varios millones de personas utilizan en todo el mundo, fruto del trabajo de miles de voluntarios que han contribuido a mejorar y añadir nuevas características al sistema.

Al igual que otros sistemas operativos, UNIX es un conjunto de programas de utilidad y un conjunto de instrumentos que permiten al usuario conectar y utilizar esas utilidades para construir sistemas y aplicaciones.

Los siguientes conceptos son comunes para todos los sistemas UNIX y LINUX, por lo cual se puede afirmar que éstos componen sus características principales.

Kernel: Es el componente principal residente en memoria del sistema operativo. Es un conjunto de programas que se encargan de proporcionar los recursos del sistema,

¹⁰ MINIX fue un Sistema Operativo desarrollado por Andrew Tanenbaum con el objetivo de formar a sus alumnos en los detalles de su construcción.

coordinar todos los detalles internos de la computadora además de asignar tareas y manejar el almacenamiento de datos. El usuario rara vez opera directamente con el kernel.

Shell ("Intérprete de Línea de Comandos"): Es un programa a través del cual los usuarios se comunican con el Kernel, traduce los comandos tecleados por el usuario y los convierte en instrucciones que puede entender el Kernel. También es un lenguaje de programación de alto nivel que puede utilizarse para crear aplicaciones completas. El shell puede soportar múltiples usuarios, múltiples tareas, y múltiples interfaces para sí mismo, los dos shells más populares son el BourneShell y el Cshell.

Sistema Multiusuarios: Soporta desde uno hasta más de 100 usuarios, ejecutando cada uno diferentes programas.

Sistema Multitareas¹¹: Permite la realización de más de una tarea a la vez. Pueden ejecutarse varias tareas en su interior, mientras se presta toda la atención al programa desplegado en la terminal.

Comunicación Entre Procesos: Permite el uso de conductos y filtros en la línea de comandos. Un conducto (*pipe*) redirige la salida de un programa para que se convierta en entrada de otro. Un filtro es un programa elaborado para procesar un flujo de datos de entrada y producir otro de datos de salida. Los conductos y filtros suelen usarse para unir utilerías y realizar alguna tarea específica.

Entrada Y Salida Independiente Del Dispositivo: Los dispositivos (como una impresora o una terminal) y los archivos en disco son considerados como archivos. Cuando se da una instrucción al sistema operativo, puede indicársele que envíe el resultado a cualquiera de los diversos dispositivos o archivos. Esta desviación recibe el nombre de redireccionamiento de la salida.

¹¹ **Multitarea:** Describe la habilidad de ejecutar varios programas al mismo tiempo. LINUX utiliza la llamada multitarea preventiva, la cual asegura que todos los programas que se están utilizando en un momento dado serán ejecutados, siendo el sistema operativo el encargado de ceder tiempo de microprocesador a cada programa.

Programas De Utilidad (Utilerías): El sistema operativo incluye una gran variedad de programas de utilidad que pueden ser fácilmente adaptadas para realizar tareas específicas. Estas utilerías son flexibles, adaptables, portables y modulares, y pueden ser usadas junto con filtros y redireccionamientos para hacerlos más poderosos.

Estructura De Archivos: La estructura de archivos está pensada para facilitar el registro de una gran cantidad de archivos. Utiliza una estructura jerárquica o de árbol que permite a cada usuario poseer un directorio principal con tantos subdirectorios como desee; también permite a los usuarios compartir archivos por medio de enlaces (links), que hacen aparecer los archivos en más de un directorio de usuario. Además, el sistema operativo permite proteger los archivos del usuario contra el acceso no autorizado.

Consolas Virtuales: Varias sesiones de login a través de la consola entre las que se puede cambiar con las combinaciones adecuadas de teclas (totalmente independiente del hardware de video). Se crean dinámicamente y puedes tener hasta 64.

Gestión Del Entorno De Red Local: En cuanto a servicios de red, incluye un completo soporte TCP/IP con soporte para la mayor parte de tarjetas de red actuales. Así mismo incluye soporte SLIP/PPP para el acceso a redes IP mediante puertos serie, tanto RS232, como ISDN, etc. También incluye soporte PLIP, para comunicación a través de puertos paralelos.

Están también, completamente soportados servicios NFS, FTP, TELNET, NNTP, SMTP, POP, IMAP, NIS y LDAP, así como servidores HTTP y HTTP seguro mediante SSL.

TESIS CON
FALLA DE ORIGEN

**TESIS CON
FALLA DE ORIGEN**



2 DESARROLLO DEL SISTEMA

2.1 ANÁLISIS

El paso previo al desarrollo de un nuevo sistema es analizar la condición actual del mismo y con ello la especificación de los requisitos que debe cumplir. Se deben especificar todas aquellas características que permitan definir los objetivos del sistema, es decir, la información que se obtendrá, los datos que se suministrarán, las funciones y el rendimiento esperado. En esta etapa el analista se asegura de distinguir entre lo que se "necesita" (elementos críticos para la realización) y lo que se "quiere" (elementos deseables pero no esenciales).

2.1.1 CONDICIÓN ACTUAL DEL SISTEMA

El grupo de Ingeniería Lingüística cuenta con un sistema en Internet para la búsqueda y recuperación de información de un Diccionario terminológico. El sistema cuenta con cinco bases de datos para su alimentación y explotación, estas son: Lingüística, Física, Destructive Phenomena, Metrología y Desastres.

El sistema permite el acceso desde Internet a tres niveles de usuario identificados por el sistema mediante un usuario (login) y una palabra clave (password), que son validadas por páginas dinámicas diseñadas en Perl. El primero de estos niveles es el de visitante, el cual ingresa con una clave pública al sistema concediéndosele permisos de consulta en las bases de datos. El segundo nivel existente es el de usuario de la base, quien fue dado de alta con ciertos permisos sobre la base como insertar y/o modificar, en el caso de querer borrar algún registro de las bases de datos el usuario manda un correo a un tercer nivel llamado Data Base Owner (DBO por sus siglas en inglés), con la petición de borrado.

El DBO tiene todos los permisos dentro de las bases de datos, y uno de los principales para el funcionamiento del sistema es el borrado de registros desde la interfaz, afectando de forma inmediata la base de datos.

TESIS CON
FALLA DE ORIGEN

2.1.2 DELIMITACIÓN DEL SISTEMA

Según Pressman¹¹ el análisis del sistema se realiza teniendo presentes los siguientes objetivos:

- ✓ identificar las necesidades del cliente
- ✓ evaluar la viabilidad del sistema
- ✓ asignar las funciones

A continuación se desglosa cada uno de los puntos anteriores permitiendo visualizar los alcances del sistema.

IDENTIFICACIÓN DE LAS NECESIDADES DEL CLIENTE

De acuerdo al análisis anterior y con base en los siguientes inconvenientes, el Grupo de Ingeniería Lingüística solicita rediseñar el sistema.

- El sistema no es portable a la plataforma Unix.
- Actualmente los usuarios del sistema son necesariamente usuarios de la base de datos, lo que implica que deben estar dados de alta en el DBMS. Cuando el DBO necesita agregar un nuevo usuario, forzosamente realiza una solicitud al Administrador del DBMS y no al Administrador del sistema.
- No se cuenta con una interfaz que permita crear o borrar bases de datos, respaldar las ya existentes o recuperar alguna base respaldada desde Internet.
- El sistema solo permite visualizar una base de datos a la vez y a partir de esta manipular los datos, pero en el caso de querer elegir otra base, es necesario salir del sistema o mantener otra sesión abierta.
- Así mismo, existen aplicaciones que hacen uso de las diferentes bases de datos pero actualmente no pueden ser ejecutados dentro del sistema. Por ejemplo los procesos de: Clustering, Distancia de edición de Levenshtein, y Contextos Definitorios.
- Los usuarios del sistema no cuentan con permisos de borrado de datos, por lo que deben solicitarlo mandando un correo electrónico al DBO para que este borre el dato.

¹¹ Ingeniería del Software, Roger S. Pressman, 3ª. Edición, McGraw Hill.

Después de establecer las necesidades del cliente, es necesario realizar un estudio de viabilidad del sistema, por lo que trataremos este tema a continuación.

VIABILIDAD DEL SISTEMA

El análisis de viabilidad se enfoca en cuatro áreas¹² explicadas a continuación.

- ❖ *Viabilidad económica:* Es una evaluación del costo del desarrollo frente al beneficio final producido por el sistema desarrollado.
- ❖ *Viabilidad técnica:* Es un estudio de la funcionalidad, del rendimiento y de las restricciones que pueden afectar a la posibilidad de realización de un sistema.
- ❖ *Viabilidad legal:* Determina cualquier infracción, violación o ilegalidad que pudiera resultar del desarrollo del sistema.
- ❖ *Alternativas:* Evalúa los enfoques alternativos para el desarrollo del sistema.

En esta tesis no fue necesario realizar un estudio de viabilidad, debido a que de antemano se habían aceptado los costos, establecido el software (software libre) y definido el tipo de sistema que se debía desarrollar. Por lo que la tesis se enfocará al diseño y desarrollo del sistema.

ASIGNACIÓN DE FUNCIONES

En el sistema propuesto es preciso contar con una interfaz Web, que maneje cuatro roles como son: Administrador del sistema, Administrador de bases de datos, Usuario del Sistema (dedicado a la alimentación de la base) y Visitante, estos tres últimos con posibilidad de interactuar con los procesos de Ingeniería Lingüística. Además es preciso contar con una base de datos global que incluya lógicamente todas las bases existentes a la fecha y que permita al usuario incluir nuevas bases.

Para los diferentes roles que maneja el sistema, se presentan a continuación cada una de las alternativas que pueden elegir.

¹² Ingeniería del Software, Roger S. Pressman, 3ª. Edición, McGraw Hill.

➤ **Administrador del sistema**

Cuando un usuario con privilegios de administrador ingrese al sistema, se mostrará un *menú de Administración del sistema* que le brindará las siguientes opciones:

- Dar de alta nuevas bases de datos desde la interfaz
- Borrar una o más bases de datos del sistema
- Dar de alta nuevos administradores del sistema
- Dar de alta nuevos Usuarios con la característica de ser propietarios de alguna base.
- Generar respaldos de la base de datos en el momento que así lo requiera, además establecer un respaldo periódico que será realizado por el sistema
- Podrá además recuperar los datos de la base eligiéndolos de un listado de respaldos anteriores
- Borrar físicamente usuarios de la base de datos

➤ **administrador de base de datos (DBO)**

El segundo menú, proporcionará una interfaz para el Administrador de base de datos (es decir el propietario de la base), las opciones son:

- Dar de alta nuevos usuarios en la base de datos
- Otorgar permisos a cada uno de los usuarios pertenecientes a su base
- Hacer un borrado lógico de usuarios que pertenecen a su base.
- Interactuar con los procesos de Ingeniería Lingüística utilizando los datos de las bases a las que tiene acceso

➤ **usuario del sistema**

La interfaz para el usuario le permitirá consultar y alimentar las bases de datos dependiendo de los permisos que le fueron otorgados e interactuar con los procesos de Ingeniería Lingüística.

➤ **visitante**

En el caso de los visitantes, se presentará una interfaz para la consulta de las bases públicas del sistema, además de una opción para solicitar la creación de una base de datos.

La siguiente etapa dentro del desarrollo del sistema es el diseño, por lo que se mencionarán las características más importantes a continuación.

2.2 DISEÑO

En esta etapa se hablará: del software propuesto para el desarrollo del sistema, de un diagrama jerárquico funcional, así como de la creación de la base de datos; también mostraremos la utilidad de los escenarios y las fichas técnicas dando ejemplos para el diseño de la interfaz de usuario.

2.2.1 PROPUESTA DE SOFTWARE

La propuesta de software considera: la plataforma a utilizar, el lenguaje de programación, el manejador de bases de datos y el servidor web necesario para visualizar el sistema desde Internet.

Considerando las necesidades del cliente, explicadas en la etapa de análisis, se decidió utilizar:

- El sistema operativo Linux Red Hat 7.2 como plataforma de desarrollo. De esta forma se contará con un sistema con capacidad de escalabilidad a los servidores Unix.
- Por otro lado, se buscó un lenguaje de programación que fuera ampliamente compatible con el sistema operativo. Dadas las características del lenguaje de programación PHP, y considerando que es un software libre, se eligió éste como lenguaje de desarrollo.
- En cuanto al manejador de bases de datos se utilizará PostgreSQL,
- Y como servidor web, Apache.

Con base en estas herramientas se diseñó el sistema que se presenta en esta tesis.

TESIS CON
FALLA DE ORIGEN

2.2.2 DIAGRAMA JERÁRQUICO FUNCIONAL DEL SISTEMA

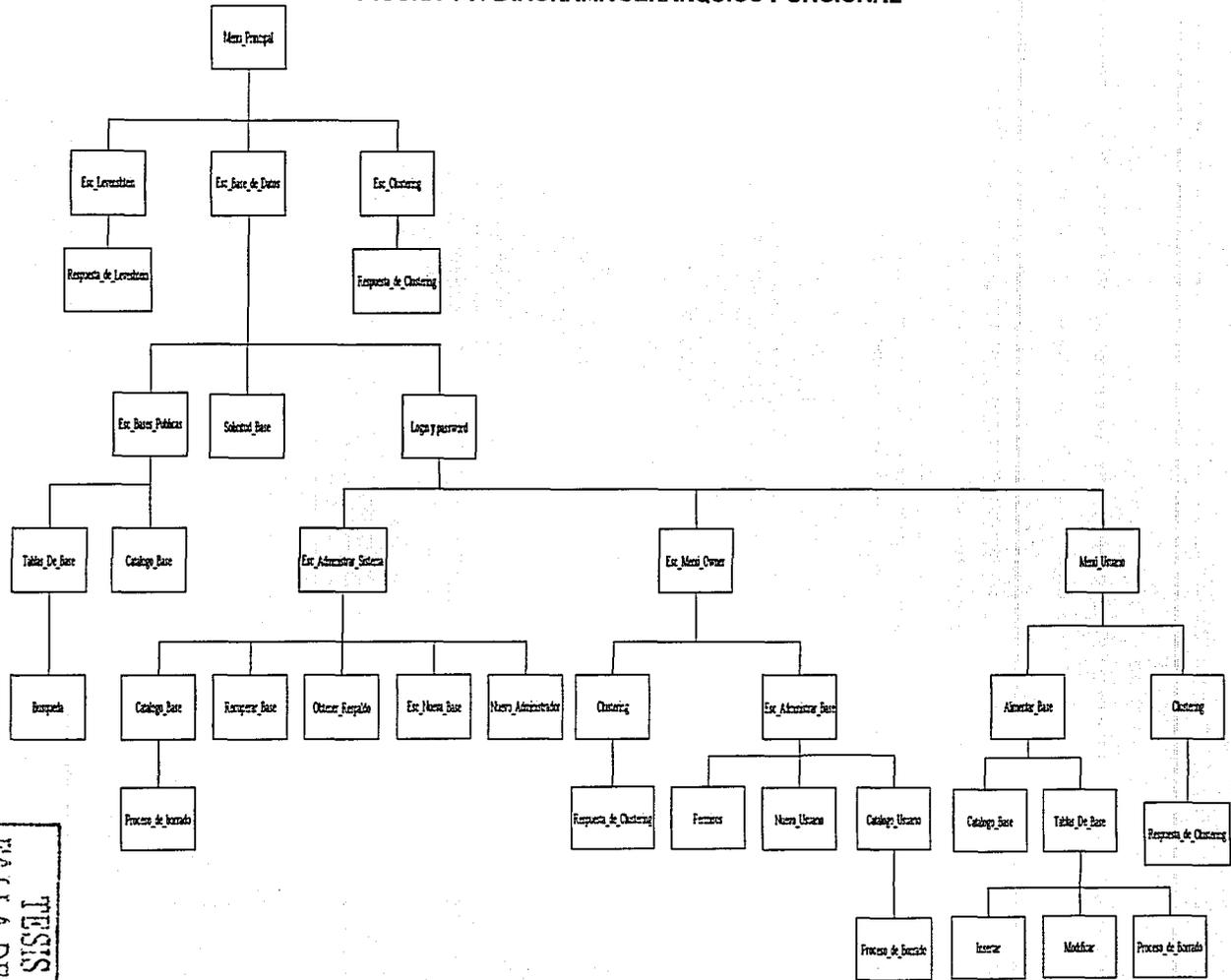
El diagrama jerárquico funcional, muestra los módulos que constituyen el sistema y esquematiza la relación que guardan entre sí. En la figura 11 se muestra el diagrama desarrollado para este sistema.

Como se observa en la figura 11, el usuario inicialmente visualiza una interfaz, llamada Menú Principal, en la que podrá interactuar con los procesos de Distancia de edición de Levenshtein y el proceso de Clustering. Además podrá ingresar al escenario de bases de datos, donde se muestra al usuario las opciones de: solicitud de base, bases públicas y la posibilidad de ingresar con un login y password para identificar permisos dentro de las bases. Por otro lado, cualquier persona, podrá consultar las bases públicas existentes en el sistema, así como enviar solicitudes para una nueva base de datos vía e-mail. El acceso mediante login y password proporciona tres tipos de menús (Administrar sistema, Menú del Owner y Menú de Usuario), donde podrán interactuar con cada uno de ellos, así como manipular los datos de las bases a las que tiene acceso.

El menú administrar sistema, ofrecerá al usuario la posibilidad de dar de alta nuevas bases de datos, borrar alguna base, dar de alta un nuevo administrador o borrarlo. Además de generar respaldos de la base de datos o recuperarlos. Cada uno de estos procesos será ejecutado desde Internet, así que es necesario un sistema de seguridad para esta etapa.

El menú del owner ofrecerá la interacción con los procesos de Ingeniería Lingüística, así como un submenú denominado "Administrar Base" donde el usuario podrá: dar de alta nuevos usuarios, borrarlos lógicamente del sistema, así como asignarles los permisos para cada una de sus bases. El menú de usuario mostrará una interfaz para la alimentación de la base de datos, en esta etapa los usuarios verán solo las bases de datos a las que tienen acceso, y contarán con permisos de consulta, borrado, inserción y modificación de los registros. Es importante mencionar que el sistema permitirá delegar responsabilidades de administración de las bases a los owners, así que el administrador del sistema no tendrá que preocuparse de ello.

FIGURA 11. DIAGRAMA JERÁRQUICO FUNCIONAL



TESIS CON
 FALLA DE ORIGEN

2.2.3 DISEÑO DE LA BASE DE DATOS

La base de datos que se diseñó, permitirá al usuario la manipulación de toda la información desde Internet, considerando los permisos que el usuario tenga dentro del sistema. El diseño partió de una base ya existente, tomando en cuenta las necesidades actuales y las consideraciones pertinentes para la migración de la información ya almacenada. Posteriormente se realizaron las adecuaciones a las necesidades del cliente. El diseño original se muestra en la figura 12:

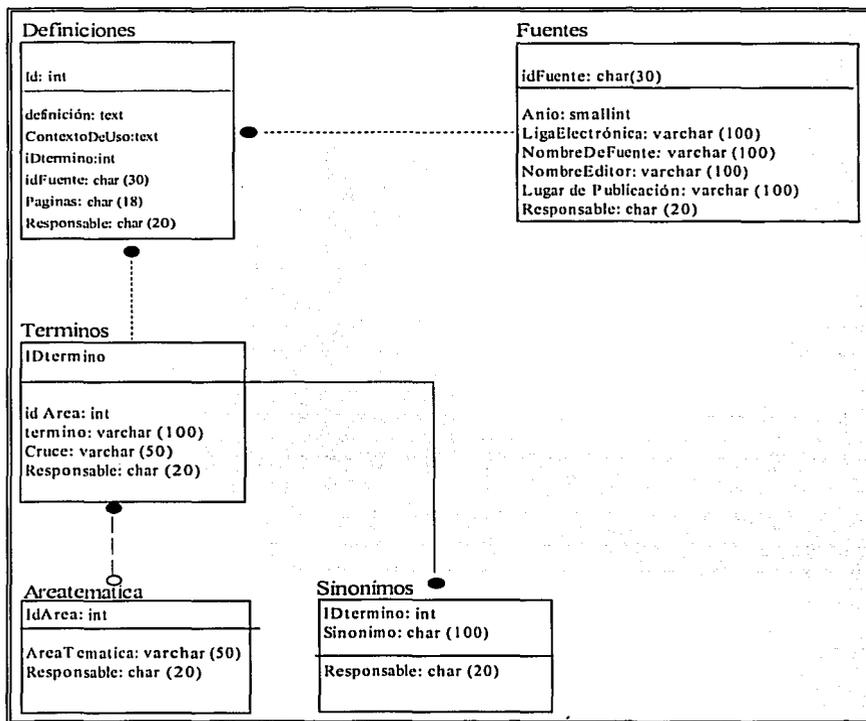


Figura 12. Diseño Original de la Base de Datos

La base de datos está utilizándose actualmente en el sistema Diccionario, pero, para permitir que el usuario administre el sistema y las bases desde Internet, se decidió modificar este diseño.

Se partió de 5 entidades principales mostradas en la figura 12, en las que el usuario consulta y modifica los datos. Para lograr la etapa de administración y el enlace a las aplicaciones existentes de Ingeniería Lingüística, fue necesario adicionar otras entidades importantes para este proceso, a continuación hablaremos de ellas.

2.2.3.1 DEFINICIÓN DE ENTIDADES

La definición de entidades se enfocó a corregir las deficiencias identificadas en la etapa de análisis, logrando con ello, satisfacer las necesidades del usuario. A continuación mostramos un listado inicial de las entidades propuestas así como sus atributos.

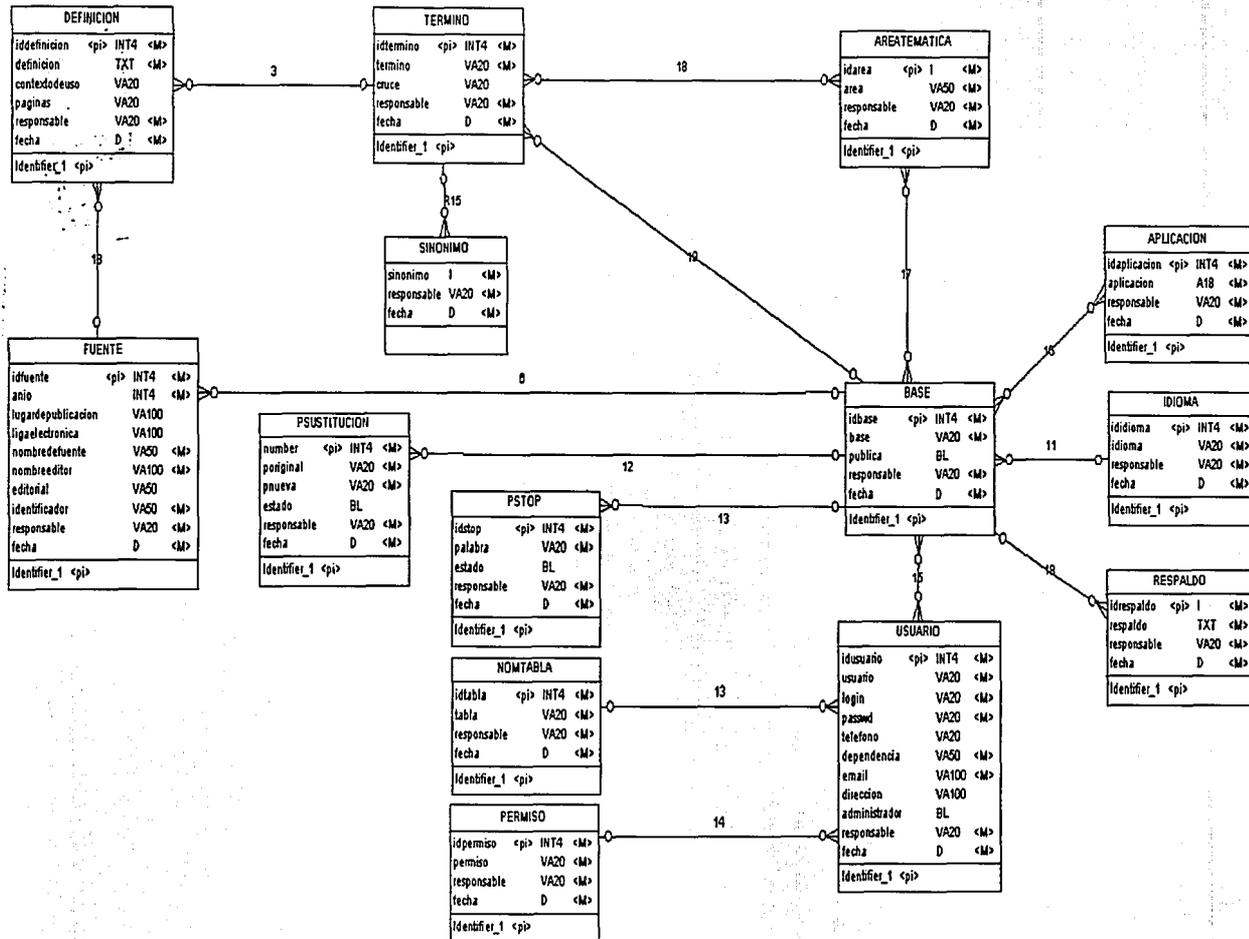
Entidades del Diccionario.

Nombre	Atributos	Nombre	Atributos
APLICACION	Idaplicacion aplicacion responsable fecha	USUARIO	Idusuario usuario passwd telefono dependencia email direccion administrador responsable fecha
NOMTABLA	Idntabla tabla responsable fecha		
PERMISO	Idpermiso permiso responsable fecha	PSTOP	Idpstop palabra estado responsable fecha
BASE	Idbase base owner responsable fecha	PSUSTITUCION	Idsustitucion poriginal pnueva estado responsable fecha
IDIOMA	Ididioma idioma responsable fecha		

Para el diseño del diagrama Entidad-Relación se utilizó la herramienta Power Designer, porque permite verificar el cumplimiento de las reglas de normalización en el diseño. La figura 13 muestra la relación existente entre las entidades antes definidas.

TESIS CON
FALLA DE ORIGEN

Figura. 13 Diagrama Entidad-Relación del Sistema Diccionario



Para corregir el diagrama anterior y romper las relaciones muchos a muchos que existían, fue necesario adicionar nuevas entidades, estas fueron nombradas según las tablas que estaban involucradas. Cabe destacar que en el caso de relaciones M:M las llaves primarias de las entidades involucradas pasan como llaves foráneas a la entidad que las rompe esto con el fin de mantener la relación entre ambas.

Entidades Adicionales

Nombre	Atributos	Nombre	Atributos
AREATERM	idarea fk1 idtermino fk2 responsable fecha	USUARIOBASE	Idusuariobase pk Idusuario fk1 Idbase fk2 owner responsable fecha
AREABASE	Idbase fk1 Idarea fk2 responsable fecha	RELACION	Idusuariobase fk1 Idtabla fk2 Idpermiso fk3 responsable fecha
BASEAPLICACION	Idbase fk1 Idaplicacion fk2 responsable fecha		

La entidad adicional llamada usuariobase, permite identificar dentro del sistema a aquellos usuarios que tendrán la característica de ser dueños de una base u owners. Estas nuevas entidades contribuyen además a que la base de datos cumpla con las reglas de normalización necesarias para su creación.

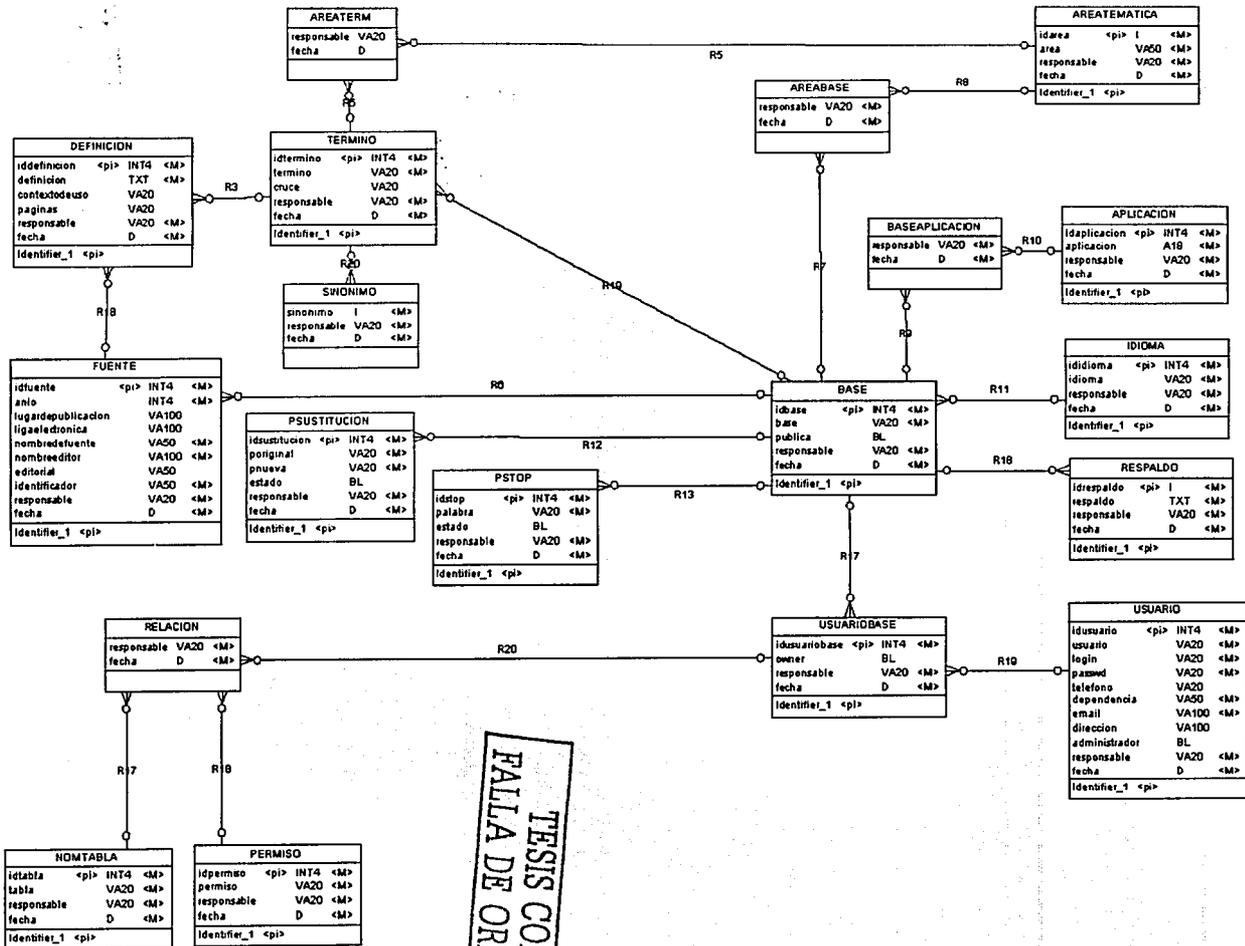
2.2.3.2 DISEÑO CONCEPTUAL Y FÍSICO

En la figura 14 podemos observar el diagrama conceptual de la base de datos, en este diseño están relacionadas las tablas originales, las propuestas y las adicionales, así como aquellos atributos que representan las llaves primarias de cada tabla.

En la figura 15 se presenta el diseño físico de la base de datos, éste representa el diseño final a partir de cual se generará el script para la creación de la base.

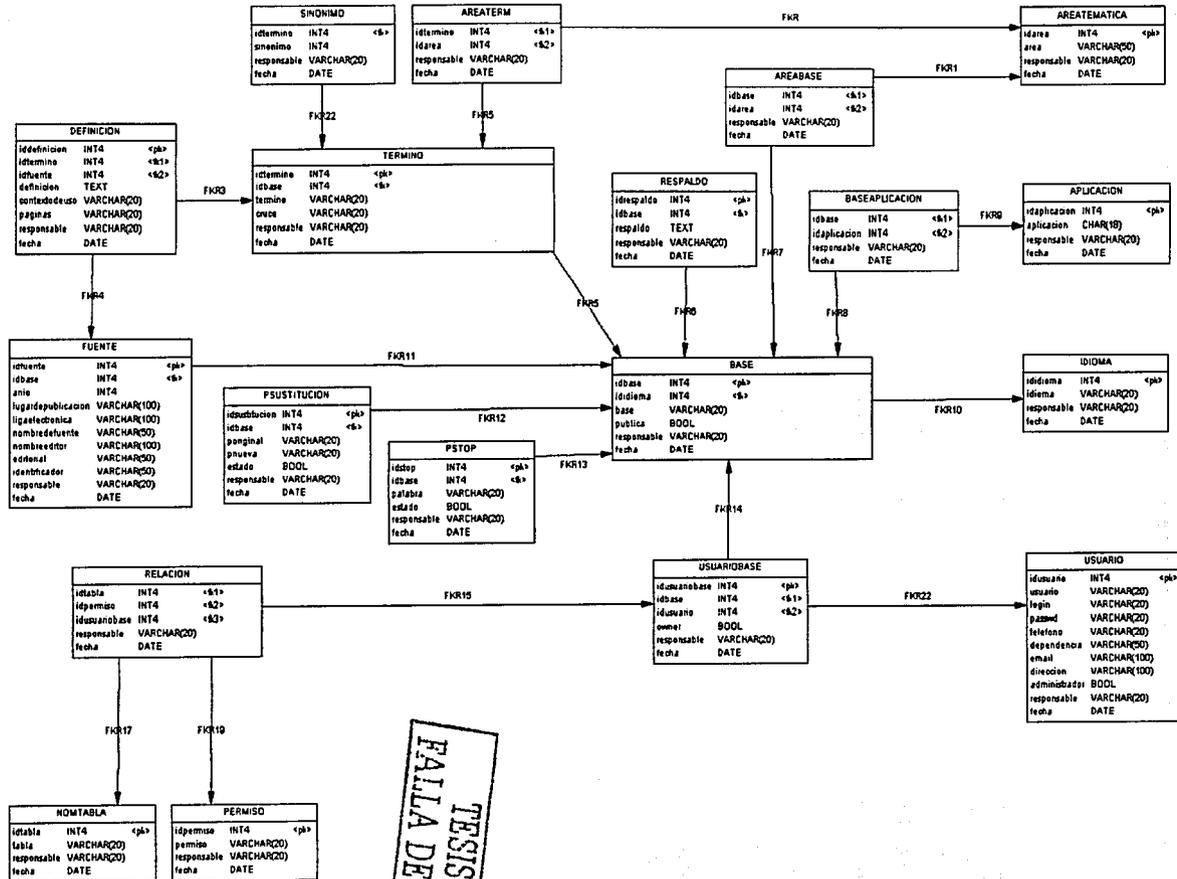
TESIS CON FALLA DE ORIGEN

FIGURA 14. DISEÑO CONCEPTUAL DE LA BASE DE DATOS



TESIS CON
 FALLA DE ORIGEN

FIGURA 15. DISEÑO FÍSICO DE LA BASE DE DATOS



TESIS CON
 FALLA DE ORIGEN

2.2.4 DICCIONARIO DE DATOS

El diccionario de datos correspondiente a la base antes diseñada, se muestra a continuación, se incluyen las validaciones pertinentes para los campos que lo requieran.

El diccionario muestra:

- Nombre de la Entidad
- Descripción de la Entidad
- Nombre de los atributos
- Tipo de dato que almacena
- Descripción de los atributos

Nombre de la Entidad	Definición de la Entidad	Nombre del Atributo	Tipo	Definición del Atributo
APLICACIÓN	Contiene los nombres de los procesos de Ingeniería Lingüística con los que puede interactuar el sistema.	idaplicacion	INT (4), not null	Identificador de la aplicación. Idaplicacion Aplicacion 1 Proceso de Clustering 2 Distancia de edición de Lev.
		aplicacion	CHAR(18), not null	Nombre de la aplicación.
		responsable	VARCHAR(20), not null	Nombre del responsable de la última modificación a los datos de la entidad aplicacion.
		fecha	DATE , not null	Fecha de la última modificación de los datos de la entidad aplicacion. responsable fecha Admin1 12-03-2002 Usuario1 23-05-2002
AREATEMATICA	Entidad que contiene los nombres de las áreas temáticas existentes en el sistema.	idarea	INT (4), not null	Identificador del área temática. Idarea Area 1 Proceso de Clustering 2 Distancia de edición de Lev.
		area	CHAR(18), not null	Nombre del área temática.
		responsable	VARCHAR(20), not null	Nombre del responsable de la última modificación a los datos de la entidad areatematica.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad areatematica.
BASE	Contiene las bases existentes en el sistema.	ibase	INT (4), not null	Identificador de la base de datos. ibase base 1 Física 2 Phenomena 3 Metrología
		base	VARCHAR(20), not null	Nombre de la base de datos.

		publica	BOOLEAN	Indicador booleano que informa si la base de datos permite el acceso publico o no. base publica Fisica T Phenomena F
		responsable	VARCHAR(20), not null	Nombre del responsable de la última modificación a los datos de la entidad base.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad base.
DEFINICION	Entidad que contiene las definiciones correspondientes a los términos de la base de datos.	iddefinicion	INT (4), not null	identificador de la definición.
		definicion	TXT, not null	Definición del termino.
		contextodeuso	VARCHAR(20)	Contexto de uso del termino.
		paginas	VARCHAR(20)	Numero de paginas de la definición.
		responsable	VARCHAR(20), not null	Nombre del responsable de la última modificación a los datos de la entidad definicion.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad definicion.
		FUENTE	Contiene las fuentes utilizada en la base de datos.	idfuente
anio	INT (4), not null			Año de publicación.
lugarpublicacion	VARCHAR(100)			Nombre del lugar donde fue publicada la fuente.
ligaelectronica	VARCHAR(100)			Liga electrónica de la fuente.
nombredelfuente	VARCHAR(50), not null			Fuente correspondiente a la definición del termino.
nombrededitor	VARCHAR(100), not null			Nombre del Editor de la fuente.
editorial	VARCHAR(50)			Nombre de la editorial de la fuente.
identificador	VARCHAR(50), not null			Abreviación del nombre de la fuente. Idfuente identificador 1 DicEnc1985 2 EncBri1964
responsable	VARCHAR(20) not null			Nombre del responsable de la última modificación a los datos de la entidad fuente.
fecha	DATE, not null			Fecha de la última modificación de los datos de la entidad fuente.
IDIOMA	Entidad que contiene los Idiomas utilizados en el sistema.	ididioma	INT (4), not null	identificador del idioma.
		idioma	VARCHAR(20) not null	Idiomas manejados en el sistema. Idioma idioma 1 español 2 ingles 3 francés
		responsable	VARCHAR(20) not null	Nombre del responsable de la última modificación a los datos de la entidad idioma.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad idioma.
NOMTABLA	Entidad que contiene los nombres de las tablas existentes en la base de datos.	idtabla	INT (4), not null	Identificador de la tabla.
		tabla	VARCHAR(20) not null	Nombre de la tabla. Idtabla tabla 1 aplicacion 2 aretematica 3 base
		responsable	VARCHAR(20) not null	Nombre del responsable de la última modificación a los datos de la entidad nomtabla.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad nomtabla.
PERMISO	Contiene los nombres de los permisos que el usuario	idpermiso	INT (4), not null	identificador del permiso.
		permiso	VARCHAR(20), not null	Nombre de los permisos. Idpermiso permiso 1 consulta 2 insercion

	tendrá sobre las tablas del sistema.	responsable	VARCHAR(20) not null	Nombre del responsable de la última modificación a los datos de la entidad permiso.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad permiso.
PSTOP	Entidad que contiene las palabras de stop de los términos.	idstop	INT (4), not null	Identificador de la palabra de stop.
		palabra	VARCHAR(20), not null	Palabra de stop. Idstop palabra 1 la 2 de
		estado	BOOLEAN	Indica si la palabra de stop esta en uso.
		responsable	VARCHAR(20) not null	Nombre del responsable de la última modificación a los datos de la entidad pstop.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad pstop.
PSUSTITUCION	Entidad que contiene las palabras de sustitucion.	idsustitucion	INT (4), not null	Identificador de la palabra de sustitución.
		poriginal	VARCHAR(20), not null	Palabra original.
		pnueva	VARCHAR(20), not null	Palabra por la que fue sustituida la palabra original. poriginal pnueva año año
		estado	BOOLEAN	Indica si la palabra de sustitucion esta en uso.
		responsable	VARCHAR(20) not null	Nombre del responsable de la última modificación a los datos de la entidad pstop.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad pstop.
RESPALDO	Entidad que contiene los respaldos del sistema en archivos txt..	idrespaldo	INT (4), not null	Identificador del respaldo.
		respaldo	TXT	Archivo que contiene los respaldos del sistema.
		responsable	VARCHAR(20) not null	Nombre del responsable del respaldo del sistema.
		fecha	DATE, not null	Fecha de creación del respaldo.
SINONIMO	Entidad que contiene los sinónimos correspondientes a los términos de las bases de datos	sinonimo	INT, not null	Identificador del sinónimo, basado en los términos existentes de las bases de datos. Sinónimo 1 4 6
		responsable	VARCHAR(20) not null	Nombre del responsable de la última modificación de los datos de la entidad sinonimo.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad sinonimo.
TERMINO	Entidad que contiene los sinónimos correspondientes a los términos de las bases de datos	idtermino	INT, not null	Identificador del termino.
		termino	VARCHAR(20) not null	Nombre del termino. Idtermino termino 1 telescopio 2 perro 3 sexo
		cruce	VARCHAR(20)	Cruce del termino.
		responsable	VARCHAR(20) not null	Nombre del responsable de la última modificación de los datos de la entidad termino.
		fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad termino.
USUARIO	Entidad que agrupa toda la información	idusuario	INT (4) not null	Identificador de usuario.
		usuario	VARCHAR(20) not null	Nombre del usuario.
		login	VARCHAR(20) not null	Nombre que identifica a una persona como usuario del sistema.

referente a los usuarios del sistema.	password	VARCHAR(20) not null	Palabra clave que permite acceder al sistema con seguridad. Login password Usuario1 Uc21t1i Admin1 At14 l l i
	telefono	VARCHAR(20)	Teléfono del usuario.
	dependencia	VARCHAR(50), not null	Nombre de la dependencia donde trabaja el usuario.
	email	VARCHAR(100), not null	Correo electrónico del usuario.
	direccion	VARCHAR(100)	Dirección donde vive el usuario.
	administrador	BOOLEAN	Indicador booleano que determina si el usuario es o no administrador del sistema
	responsable	VARCHAR(20) not null	Nombre del responsable de la última modificación a los datos de la entidad usuario.
	fecha	DATE, not null	Fecha de la última modificación de los datos de la entidad usuario.

A continuación se explican las características del diseño de escenarios y fichas técnicas para la interfaz de usuario:

2.2.5 INTERFAZ WEB PARA EL DICCIONARIO ELECTRÓNICO

El sistema a desarrollar será capaz de proporcionar un fácil acceso a la base de datos a través del Web, confiable y seguro, permitiendo que los usuarios autorizados alimenten la base de datos de manera simultánea, sin restricción de acceso, y permitiendo la consulta a quien visite la página Web.

Por otro lado, se llevará a cabo el diseño de una interfaz que permita al usuario manipular la base de datos, proporcionando cuatro tipos de menús (Administradores, Owners – propietarios de base--, usuarios y visitantes), logrando con ello, delegar responsabilidades en el proceso. Se habla de la creación de una base de global que incluye de forma lógica, todas las bases de datos existentes a la fecha y que permitirá agregar nuevas bases en la ejecución del programa.

En el apéndice 1, se presenta de manera detallada la metodología para el desarrollo de sitios Web aplicada en este tesis.

Esta metodología permitirá al desarrollador entender que es lo que el sistema va a realizar y como es que interactúan entre si las partes que componen el sistema, la relación que hay entre estas partes y los procesos que intervienen en un momento dado para que el

sistema realice las funciones que se esperan. Además, dará una visión general de los scripts o códigos que se usaran para realizar una acción específica.

Se explicará el diseño de escenarios y fichas técnicas utilizados para esta tesis, así como una breve definición de cada uno.

Un **escenario** nos ayuda a entender la relación que existe entre páginas Web teniendo como resultado los diferentes comportamientos que surgen al momento de interactuar de una página a otra.

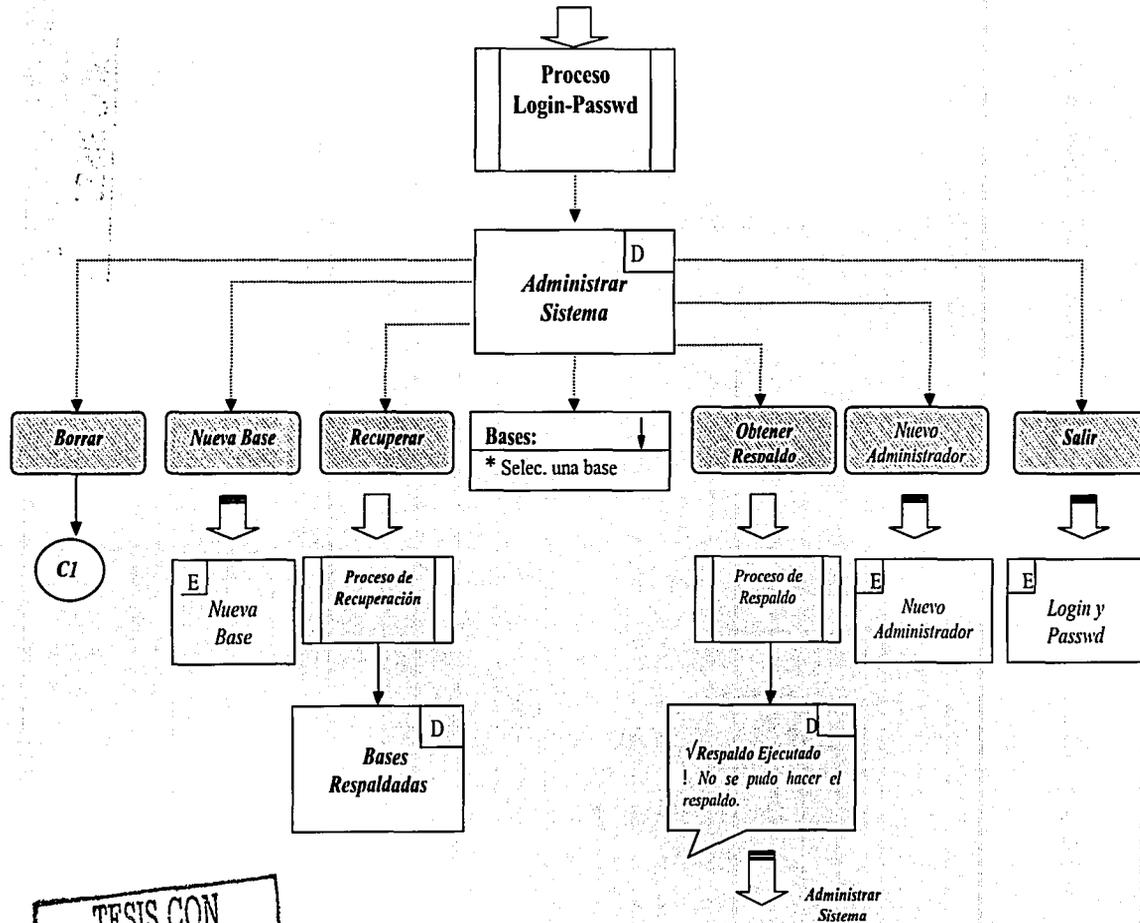
Una **ficha técnica** es la descripción general del contenido de una página Web. Sirve para mostrarnos las propiedades de la página, sus acciones y su interacción con otras páginas, así como los scripts generados.

En la figura 16 se muestra un ejemplo representativo de los Escenarios. El escenario "Administrar Sistema" ejemplifica una de las etapas más importantes del sistema, ya que ofrece al usuario la posibilidad de crear y borrar bases de datos desde Internet, así como respaldar la información del sistema periódicamente y/o recuperarla cuando sea necesario. El usuario podrá crear y borrar administradores desde esta interfaz.

Por otro lado, el sistema ofrecerá al usuario un menú llamado bases publicas, mostrado en el escenario de la figura 17 donde el usuario podrá manipular los datos de las bases a las que tiene acceso, y dependiendo de los permisos que le fueron asignados podrá insertar, borrar o modificar. Como se puede observar este escenario muestra una Frame o Marco como pagina principal ya que es necesario tener a la vista una lista de bases existentes, así como las tablas y datos correspondientes a cada una de ellas.

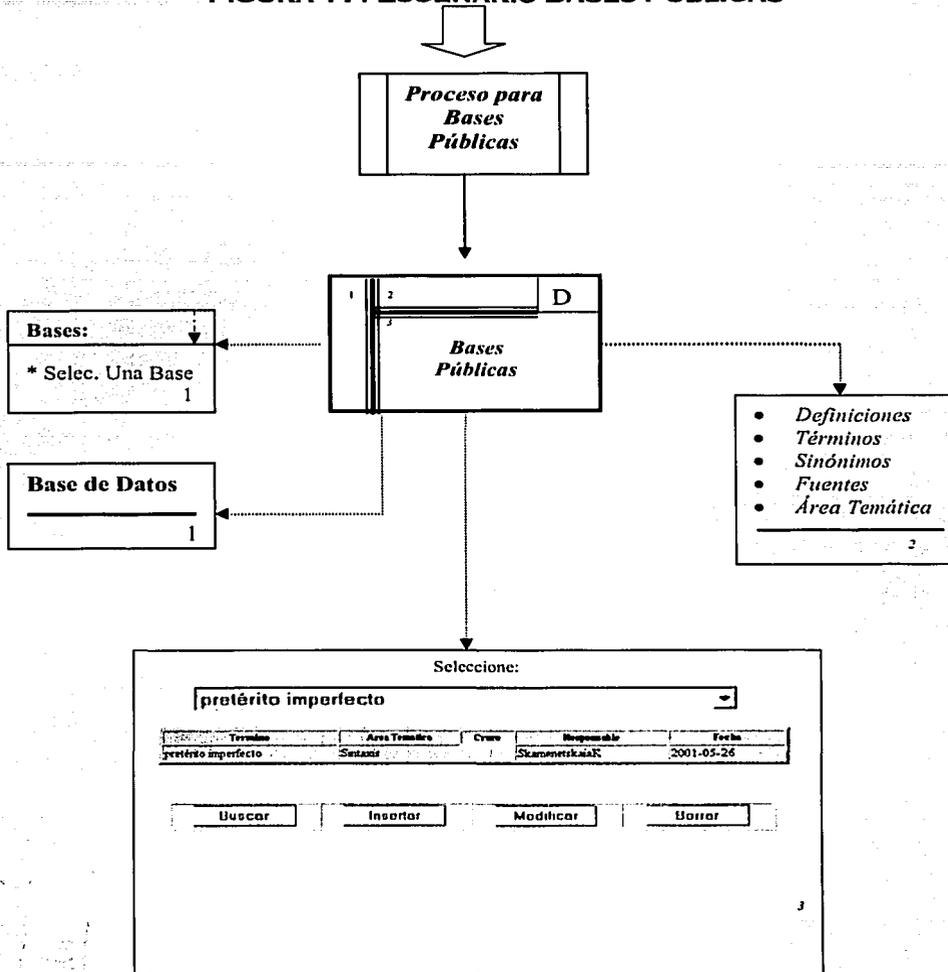
Los escenarios del sistema quedan explicados a detalle en el apéndice 1 de esta tesis.

FIGURA 16. ESCENARIO ADMINISTRAR SISTEMA



TESIS CON FALLA DE ORIGEN

FIGURA 17. ESCENARIO BASES PÚBLICAS



TESIS CON FALLA DE ORIGEN

Fichas técnicas. Como una forma de sintetizar la información contenida en una página web, es necesario la utilización de descriptores breves que permitan dar información con la cual se pueda construir la página conociendo los elementos con los que cuenta y la acción que éstos generan, así como saber que código (scripts) puede ser reutilizable. Esta información esta contenida en las fichas técnicas por lo que a continuación se ejemplifica la etapa de administrar el sistema.

- ✓ **Página Dinámica**
- ◆ **Nombre:** Administrar Sistema
- ◆ **Entrada:** Proviene del proceso Login y Passwd
- ◆ **Propiedades:**

- **Catalogo**

Nombre	Función
Bases	Despliega las bases existentes en el Sistema.

- **Botón**

Nombre	Asociado	Función
Borrar	A un Escenario llamado: Administrar Sistema.	Invoca un Script llamado: "Proceso de Borrado para Administrar Sistema".
Nueva Base	A un Escenario llamado: Administrar Sistema.	Realiza un proceso que abre la página estática Nueva Base.
Recuperar	A un Escenario llamado: Administrar Sistema.	Invoca un Script llamado: "Recuperar".
Obtener respaldo	A un Escenario llamado: Administrar Sistema.	Invoca un Script llamado: "Proceso para Obtener Respaldo".
Nuevo Administrador	A un Escenario llamado: Administrar Sistema.	Muestra la página dinámica <u>Nuevo administrador</u> en la que se puede dar de alta a un nuevo Administrador.
Salir	A un Escenario llamado: Administrar Sistema.	Cierra la Página dinámica de Administrar Sistema y abre la página Login y Passwd.

- ✓ **Scripts**

- ◆ **Nombre:** Proceso de Borrado para Administrar Sistema
- ◆ **Descripción:** Ejecuta un proceso que muestra la página dinámica Respuesta de Borrado en la cual el Administrador puede borrar una base de datos
- ◆ **Parámetros que recibe:** El nombre de la página de la que proviene en este caso "Administrar Sistema" el nombre de la base dado por el Catalogo "Bases", el

nombre de la tabla en este caso "Bases" y el elemento a Borrar, en este caso el nombre de la base de datos dada por el catalogo "Bases".

- **Nombre:** Recuperar
- **Descripción:** Se conecta a la tabla Respaldos para seleccionar los nombres de las bases que han sido respaldadas y las muestra en la página dinámica Bases respaldadas
- **Nombre:** *Proceso para Obtener Respaldo*
- **Descripción:** Ejecuta un proceso que generará un respaldo de la base de datos, creando un archivo txt que se colocara dentro de la tabla Respaldos. Mostrará también un cuadro de diálogo que indique si el proceso se ejecutó satisfactoriamente o no.
- **Parámetros que recibe:** Información dada por el Catalogo de nombre: "Bases", con el nombre de la base a respaldar.

Todas las fichas técnicas utilizadas para el diseño de esta tesis se explican a detalle en el apéndice 2 de esta tesis.

La utilidad de los escenarios y fichas técnicas antes explicados, se aborda en el siguiente capítulo, donde se hablará de la etapa de implantación del Sistema.

3 IMPLANTACIÓN

3.1 INTRODUCCIÓN

En la etapa de implantación se exponen los puntos principales para la creación del sistema, iniciando con la elección de nuestra plataforma operativa y continuando con la creación de la base de datos y la interfaz de usuario.

A partir de la propuesta de software y el diseño del sistema explicados en el capítulo anterior, la implantación de éste se llevo a cabo en Linux Red Hat 7.2 utilizando el lenguaje de programación PHP 4.0.6, el manejador de bases de datos PostgreSQL 7.1 y el servidor web Apache 1.3.

3.1.1 ESPECIFICACIONES DE HARDWARE

Para instalar Linux, fue necesario conocer los requerimientos mínimos del hardware, previniendo de esta forma alguna falla que pudiera presentarse en la red. Estas características se enlistan a continuación:

- Procesador Intel 386 y posteriores, SPARC, Alpha, PowerPC, etc.
- Mínimo 4MB de memoria
- de 150 a 200 Mb en disco Duro

De acuerdo a lo anterior fue posible instalar el servidor en una computadora con las siguientes especificaciones:

DISPOSITIVO	CARACTERÍSTICAS
Marca	Hewlett Packard
Procesador	Intel Pentium 550MHz
Memoria RAM	48 MB
Disco duro	2.39 GB

3.1.2 SEGURIDAD DEL SERVIDOR

Se inicio con la instalación de Linux, tomando como referencia los manuales de Red Hat existentes en Internet, así como parte de la bibliografía mencionada en esta tesis.

La seguridad del servidor se implantó con Apache y el protocolo de capa de enchufes seguros (Secure Sockets Layer Protocol) o SSL, proporcionando entre el cliente y el servidor, una comunicación segura, con una autenticación mutua usando firmas digitales para comprobar la integridad de los mensajes y la encriptación para la privacidad. Posteriormente se configuró PHP y PostgreSQL, para dar inicio a la creación de la base de datos.

3.2 CREACIÓN DE LA BASE DE DATOS

Una vez instalados los paquetes de PostgreSQL, Apache y PHP, fue necesario dar de alta los servicios de httpd y postgresql con los siguientes comandos:

```
$ service httpd start  
$ service postgresql start
```

Para instaurar la base de datos, fue necesario crear la cuenta de superusuario de PostgreSQL, la cual usa por defecto el nombre de "postgres". Este usuario no cuenta con privilegios de root (administrador) dentro del sistema, reduciendo así los riesgos de inseguridad.

Una vez que el usuario existe, es necesario ingresar con esa cuenta, de manera que pueda crearse la base de datos haciendo uso del siguiente comando:

```
$ createdb DICCIONARIO
```

Para llevar a cabo la creación de las tablas de la base, acorde a la definición de entidades y modelo de datos explicados en la etapa de análisis, se creó un script llamado diccionario.sql en PowerDesigner 9.0, el cual fue ubicado en el directorio /var/www/html del servidor, que es el directorio que indica el archivo de configuración del servicio httpd, el httpdconf. Para ejecutar el script fue necesario ingresar a la base de datos y utilizar los siguientes comandos:

```
$ psql DICCIONARIO postgres  
$ \i diccionario.sql
```

Este último comando, lee el script "diccionario" y ejecuta todas las sentencias que están dentro de él. A continuación se muestra parte del código utilizado para la creación de la base de datos, en él se ejemplifica la creación de las tablas principales de la base, los atributos correspondientes y el establecimiento de las llaves primarias y foráneas respectivas.

```

/*=====*/
/* Database name: DICCIONARIO */
/* DBMS name: PostgreSQL 7 */
/* Created on: 14/04/2002 06:39:34 p.m. */
/*=====*/

/* Table: APLICACION */
/*=====*/
create table APLICACION (
IDAPLICACION INT4 not null,
APLICACION CHAR(18) not null,
RESPONSABLE VARCHAR(20) not null,
FECHA DATE not null,
constraint PK_APLICACION primary key (IDAPLICACION)
);

/*=====*/
/* Table: AREATEMATICA */
/*=====*/
create table AREATEMATICA (
IDAREA INT4 not null,
AREA VARCHAR(50) not null,
RESPONSABLE VARCHAR(20) not null,
FECHA DATE not null,
constraint PK_AREATEMATICA primary key (IDAREA)
);

/*=====*/
/* Table: IDIOMA */
/*=====*/
create table IDIOMA (
IDIOMA INT4 not null,
IDIOMA VARCHAR(20) not null,
RESPONSABLE VARCHAR(20) not null,
FECHA DATE not null,
constraint PK_IDIOMA primary key (IDIOMA)
);

/*=====*/
/* Table: PERMISO */
/*=====*/
create table PERMISO (
IDPERMISO INT4 not null,
PERMISO VARCHAR(20) not null,
RESPONSABLE VARCHAR(20) not null,
FECHA DATE not null,
constraint PK_PERMISO primary key (IDPERMISO)
);

```

TESIS CON
FALLA DE ORIGEN

```

/*=====*/
/* Table: USUARIO */
/*=====*/
create table USUARIO (
IDUSUARIO      INT4          not null,
USUARIO        VARCHAR(20)   not null,
LOGIN          VARCHAR(20)   not null,
PASSWD        VARCHAR(20)   not null,
TELEFONO       VARCHAR(20)   null,
DEPENDENCIA   VARCHAR(50)   not null,
EMAIL         VARCHAR(100)  not null,
DIRECCION     VARCHAR(100)  null,
ADMINISTRADOR  BOOL         null,
RESPONSABLE   VARCHAR(20)   not null,
FECHA         DATE          not null,
constraint PK_USUARIO primary key (IDUSUARIO)
);

```

```

/*=====*/
/* Table: BASE */
/*=====*/
create table BASE (
IDBASE        INT4          not null,
IDIOMA        INT4          null,
BASE          VARCHAR(20)   not null,
PUBLICA       BOOL         null,
RESPONSABLE   VARCHAR(20)   not null,
FECHA         DATE          not null,
constraint PK_BASE primary key (IDBASE),
constraint FK_BASE_RELATIONS_IDIOMA foreign key (IDIOMA)
references IDIOMA (IDIOMA)
on delete restrict on update restrict
);

```

```

/*=====*/
/* Table: PSTOP */
/*=====*/
create table PSTOP (
IDSTOP        INT4          not null,
IDBASE        INT4          null,
PALABRA       VARCHAR(20)   not null,
ESTADO        BOOL         null,
RESPONSABLE   VARCHAR(20)   not null,
FECHA         DATE          not null,
constraint PK_PSTOP primary key (IDSTOP),
constraint FK_PSTOP_RELATIONS_BASE foreign key (IDBASE)
references BASE (IDBASE)
on delete restrict on update restrict
);

```

```

/*=====*/
/* Table: PSUSTITUCION */
/*=====*/
create table PSUSTITUCION (
IDSUSTITUCION INT4          not null,
IDBASE        INT4          null,
PORIGINAL     VARCHAR(20)   not null,
PNUEVA        VARCHAR(20)   not null,
ESTADO        BOOL         null,
RESPONSABLE   VARCHAR(20)   not null,
FECHA         DATE          not null,

```

```

constraint PK_PSUSTITUCION primary key (IDSUSTITUCION),
constraint FK_PSUSTITU_RELATIONS_BASE foreign key (IDBASE)
  references BASE (IDBASE)
  on delete restrict on update restrict
);

/*=====*/
/* Table: RESPALDO */
/*=====*/
create table RESPALDO (
  IDRESPALDO      INT4      not null,
  IDBASE          INT4      null,
  RESPALDO       TEXT      not null,
  RESPONSABLE    VARCHAR(20) not null,
  FECHA          DATE      not null,
  constraint PK_RESPALDO primary key (IDRESPALDO),
  constraint FK_RESPALDO_RELATIONS_BASE foreign key (IDBASE)
    references BASE (IDBASE)
    on delete restrict on update restrict
);

/*=====*/
/* Table: SINONIMO */
/*=====*/
create table SINONIMO (
  IDTERMINO      INT4      null,
  SINONIMO       INT4      not null,
  RESPONSABLE    VARCHAR(20) not null,
  FECHA          DATE      not null,
  constraint FK_SINONIMO_RE_22_TERMINO foreign key (IDTERMINO)
    references TERMINO (IDTERMINO)
    on delete restrict on update restrict
);

/*=====*/
/* Table: FUENTE */
/*=====*/
create table FUENTE (
  IDFUENTE       INT4      not null,
  IDBASE         INT4      null,
  ANIO           INT4      not null,
  LUGARDEPUBLICACION VARCHAR(100) null,
  LIGAELECTRONICA VARCHAR(100) null,
  NOMBREDEFUENTE VARCHAR(50) not null,
  NOMBREEDITOR   VARCHAR(100) not null,
  EDITORIAL      VARCHAR(50) null,
  IDENTIFICADOR  VARCHAR(50) not null,
  RESPONSABLE    VARCHAR(20) not null,
  FECHA          DATE      not null,
  constraint PK_FUENTE primary key (IDFUENTE),
  constraint FK_FUENTE_RELATIONS_BASE foreign key (IDBASE)
    references BASE (IDBASE)
    on delete restrict on update restrict
);

/*=====*/
/* Table: DEFINICION */
/*=====*/
create table DEFINICION (
  IDDEFINICION  INT4      not null,

```

```

IDTERMINO          INT4          null,
IDFUENTE           INT4          null,
DEFINICION         TEXT          not null,
CONTEXTODEUSO     VARCHAR(20)   null,
PAGINAS            VARCHAR(20)   null,
RESPONSABLE       VARCHAR(20)   not null,
FECHA              DATE          not null,
constraint PK_DEFINICION primary key (IDDEFINICION),
constraint FK_DEFINICI_R_3_TERMINO foreign key (IDTERMINO)
references TERMINO (IDTERMINO)
on delete restrict on update restrict,
constraint FK_DEFINICI_RELATIONS_FUENTE foreign key (IDFUENTE)
references FUENTE (IDFUENTE)
on delete restrict on update restrict
);

```

3.3 MIGRACIÓN DE DATOS

Es conveniente mencionar que la base de datos que utilizaba el Grupo de Ingeniería Lingüística, estaba instalada en un servidor Windows 2000 y un manejador de bases de datos SQL Server, por lo que se tuvo que recurrir a la utilización de una herramienta que nos permitiera hacer la migración de los datos existentes en SQL Server a una plataforma Linux y un manejador de bases de datos PostgreSQL.

El proceso de migración se llevo a cabo mediante la opción de exportación que tiene SQL Server, los datos eran almacenados en archivos de texto usando delimitadores¹³.

PgAdmin reflejó ser la herramienta requerida para la migración de los datos de la base, ya que puede instalarse en computadoras que tengan Windows 2000 o superior además de que utiliza PostgreSQL ODBC driver (psqlODBC) para comunicarse con el servidor PostgreSQL. Proporciona una interfaz principal que permite; manipular los datos de la base por medio de la ejecución de sentencias sql, editarlos y de esta manera exportarlos. En la etapa de migración, los archivos txt fueron adecuados a las características de la nueva base, se le adicionaron datos que no existían para que fueran compatibles. A continuación se muestra un ejemplo de los datos contenidos en la tabla areatemática y como se realizaron los cambios correspondientes para la migración.

¹³ Un delimitador puede ser una coma, un espacio en blanco, o comillas.

Tipos de datos:

La tabla AREATEMATICA contiene cuatro campos de validación que deben ser insertados en cada ocasión que se da de alta un nuevo registro(idarea, area, responsable y fecha). Para realizar la exportación fue necesario adicionar entre cada campo el delimitador coma (,) así como el delimitador comillas (") para especificar que los campos son del tipo character o date. Un ejemplo de los datos obtenidos de SQL Server fue:

1, Dialectología, RZacarias

Al realizar la adecuación correspondiente se obtuvieron los siguientes registros listos para ser migrados a la base de datos:

1,"Dialectología"," RZacarias","2002-05-13 13:47:34"

2,"Lexicología"," RMaldonados","2002-05-13 13:47:34"

3.4 CREACIÓN DE LA INTERFAZ DE USUARIO

A partir del diagrama jerárquico funcional presentado en el capítulo anterior, se obtuvo el panorama general del sistema que se necesitaba implantar. La aplicación se realizó con el lenguaje PHP, el código consta de unas 10,000 líneas distribuidas en 15 archivos.

El diseño del sistema estuvo basado en la creación de 4 diferentes menús para los usuarios (administrador, owner, usuario y visitante), permitiendo de esta forma que el ingreso a las bases de datos y su modificación quedara restringido a aquellos usuarios con permisos. Fue necesaria la creación de un menú principal que permitiera a toda persona un recorrido por el sistema, así como la consulta de las bases de datos públicas, este menú es el Menú Principal y consta de cuatro opciones de navegación, tres de éstas brindan el acceso a los procesos de Ingeniería Lingüística (Clustering, Distancia de edición de Levenshtein, y Contextos Definitorios), además del acceso a las bases de datos del sistema, ver figura 18.

TESIS CON
FALLA DE ORIGEN

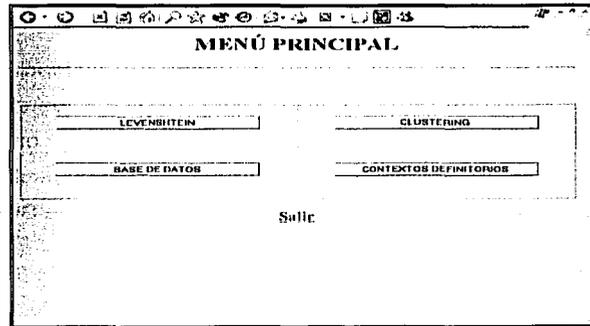
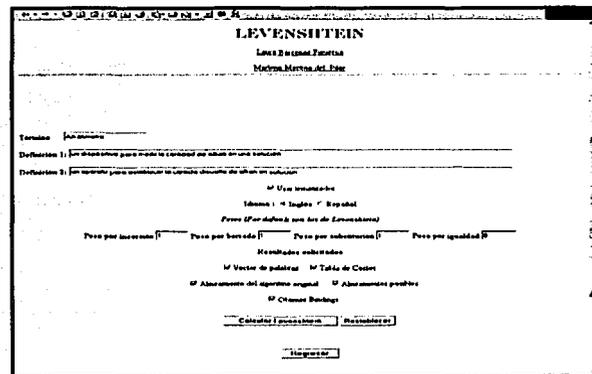


Figura 18. Menú Principal

3.4.1 VISITANTE

Cuando un usuario del tipo visitante ingresa al sistema podrá navegar dentro de éste, utilizando los procesos de Ingeniería Lingüística como Distancia de Edición de Levenshtein (ver figura 19).



TESIS CON FALLA DE ORIGEN

Figura 19. Proceso de Levenshtein

En el proceso de Levenshtein, el usuario ingresará dos definiciones diferentes acerca de un mismo término para obtener como salida un vector de palabras, tabla de costos, alineamiento del algoritmo original, alineamientos posibles y bindings, según haya elegido.

El documento que detalla este proceso se encuentra en una tesis de Maestría¹⁴ a disposición de los interesados.

De igual forma el usuario podrá interactuar con el proceso de Clustering (figura 20) y Contextos Definitivos.

Figura 20. Proceso de Clustering

Al ingresar a la opción de base de datos, el usuario visualizará otro menú (figura 21) donde podrá ingresar a la página de bases públicas y consultar los datos que se encuentren disponibles, además si así lo requiere, podrá enviar una solicitud de base de datos vía Internet a los administradores del sistema, quienes evaluarán la petición y regresarán de ser el caso un correo con los datos de la base ya creada.

A continuación se muestra un ejemplo en la figura 22, ejemplificando lo que el usuario vería al ingresar a la opción de solicitud de base:

TESIS CON
FALLA DE ORIGEN

¹⁴ Castillo Hernández Gabriel, *Algoritmo Revisado para la extracción automática de agrupamientos semánticos*.

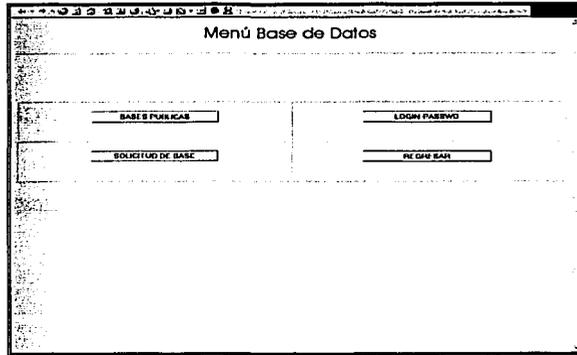


Figura 21. Menú Base de Datos

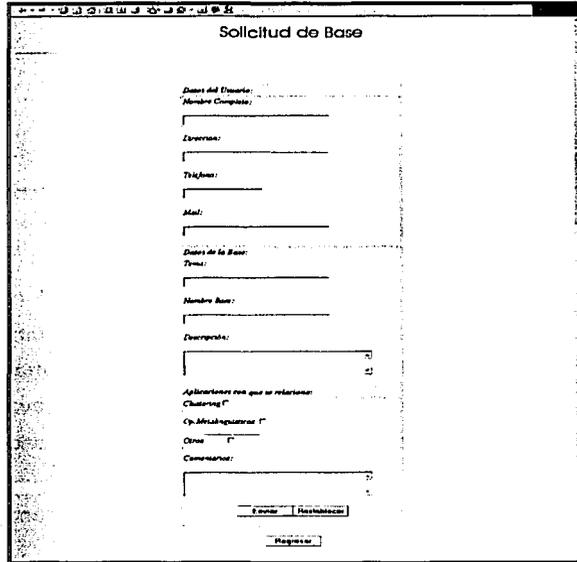


Figura 22. Solicitud de Base

Algunos de los datos que se piden en esta pantalla son obligatorios, ya que tratamos de evitar el acarreo de basura por Internet. La creación de la base de datos solicitada, necesariamente tiene que tener un fin académico explícito, ya que los datos dentro de la base son de gran utilidad para la investigación del Instituto de Ingeniería.

TESIS CON FALLA DE ORIGEN

El ingreso a las bases de datos mediante login y password ofrece al usuario tres opciones: Menú Administrar Sistema (si sé que el usuario tiene la característica de ser administrador), Menú Owner (si el usuario es propietario de una o más bases de datos) y Alimentar base (para los usuarios que manipulan la información dentro de la base), ver figura 23.

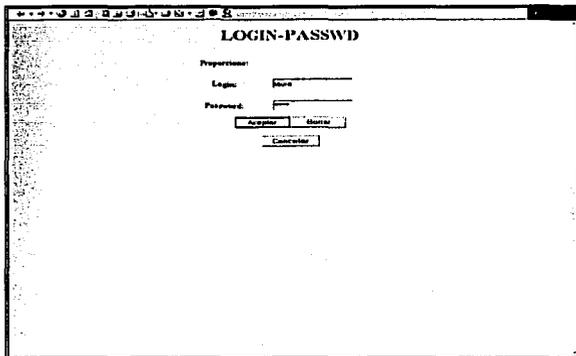


Figura 23. Login y Password

El código que determina cual menú deberá aparecer se muestra a continuación.

Conexión_users.php

```
<?php
/*realiza la conexión a la base de datos */
$conexion=pg_connect("dbname=DICCIONARIO user=**** password=*****")
or die("<center> No se pudo establecer la conexión con el servidor de BD </center>");
$sql="select * from usuario where login='$login' and passwd='$passwd'";
$sql_result=pg_exec($conexion,$sql) or die ("No se pudo ejecutar la búsqueda");
$num =pg_numrows($sql_result);
$i=0;

/*si la conexión se estableció con éxito entonces comienza la validación del login de usuario */
while($i<$num){
    $renglon=pg_fetch_array($sql_result,$i);
    $idusuario=$renglon["idusuario"];
    $administrador=$renglon["administrador"];
    $usuario=$renglon["login"];
    $passwd=$renglon["passwd"];
    $i++;
}
if(isset($nvar)or($usuario === $login && $passwd === $passwd ))
{
```

TESIS CON FALLA DE ORIGEN

```

/*Determina si el campo de administrador esta habilitado */
if($administrador=="t")
{
/* Genera la pagina de Administrador */
}
if($owner=="t")
{
/* genera el menú de owner
}
else
{
/* Genera la pagina de Alimentar base */
}
pg_close($conexion); /* cierra la conexion */
.....

```

3.4.2 ADMINISTRADOR

En el caso de que el sistema determine que el usuario que ingreso es un administrador , se genera la página de Administrar Sistema (figura 24), brindándole al usuario la posibilidad de crear bases de datos o borrarlas desde ésta misma pantalla, además de generar respaldos del sistema en cualquier momento, así como dar de alta o borrar administradores.

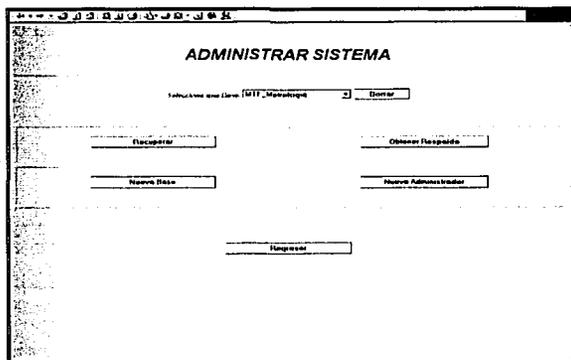


Figura 24. Administrar Sistema

Cabe destacar que el sistema tiene un proceso de alarma que determina si el usuario ha dejado de interactuar con el sistema, por lo que la pagina caduca y la conexión establecida se elimina, esto fue creado para evitar que conexiones de administradores, owners y usuarios queden abiertas, así que pasado un tiempo sin uso, es necesario volver a ingresar un login y password.

TESIS CON
FALLA DE ORIGEN

Cuando un administrador da de alta una nueva base de datos aparece la pantalla de la figura 25, donde será necesario ingresar los datos necesarios para que ingresen al sistema.

The screenshot shows a web browser window with the title 'NUEVA BASE'. The form contains the following elements:

- Form title: **NUEVA BASE**
- Input fields:
 - Nombre: _____
 - Usuario: _____
 - Password: _____
 - Host: _____
 - Servidor: _____
- Label: **Responsable administrador**
- Text input field: _____
- Buttons: **Cancelar** and **Registrar**

Figura 25. Nueva Base

EL proceso de dar de alta un nuevo administrador es similar al anterior, y la figura 26 lo ejemplifica.

The screenshot shows a web browser window with the title 'NUEVO ADMINISTRADOR'. The form contains the following elements:

- Form title: **NUEVO ADMINISTRADOR**
- Input fields:
 - Nombre de Usuario: _____
 - Login: _____
 - Password: _____
 - Direccion: _____
 - Telefono: _____
 - Email: _____
 - Responsable de: _____
- Label: **Responsable de**
- Text input field: _____
- Buttons: **Cancelar** and **Registrar**

Figura 26. Nuevo Administrador

3.4.3 OWNER

Cuando el usuario que ingresa al sistema es un owner, podrá tener acceso al proceso de Clustering antes explicado, así como a una etapa importante en el sistema, la administración de bases de datos (figura 27).

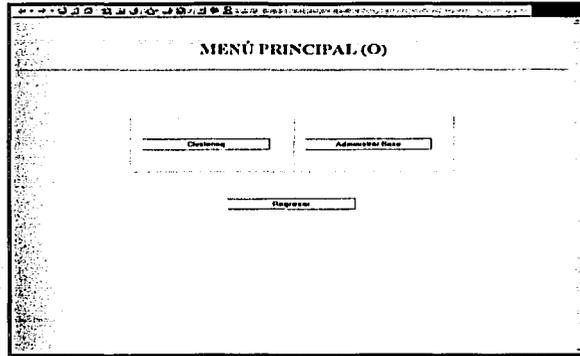


Figura 27. Menú Owner

La etapa de administración permite al owner establecer permisos sobre sus tablas, así como dar de alta a aquellos usuarios que estarán encargados de alimentarla.

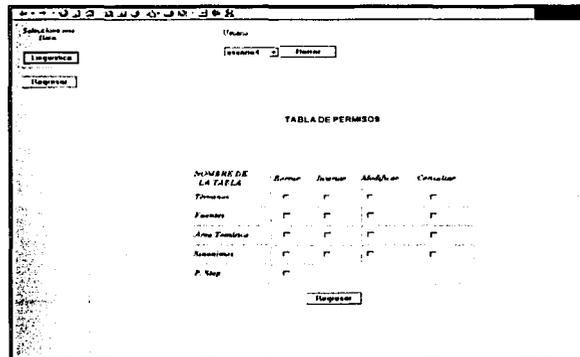


Figura 28. Administrar Base

A continuación se muestra una parte del código que muestra al owner las bases de datos a las que tiene acceso y los usuarios que alimentan estas bases.

Catalogo Base.php

```
<?php
```

```
switch ($pagina)
```

```
{
  case owner:
```

```
    $conexion=pg_connect("dbname=Diccionario user=**** password=****")
    or die("<center> No se pudo establecer la conexión con el servidor de BD </center>");
```

```
    $sql="select base* from base, usuariobase, usuario where usuario.login='$login' and
    usuario.idusuario=usuariobase.idusuario and usuariobase.idbase=base.idbase";
```

```
    $sql_result=pg_exec($conexion,$sql) or die ("No se pudo ejecutar la búsqueda");
    $num =pg_numrows($sql_result);
```

```
    echo"<form method='POST' action='Base_usuario.php?idbase=$idbase'
    target='Cabecera'>";
```

```
    echo"<p '>Seleccione una Base</p>";
```

```
    echo"<table border=1>";
```

```
    echo"<tr>";
```

```
    $i=0;
```

```
    while($i<$num)
```

```
    {
        $renglon=pg_fetch_array($sql_result,$i);
        $idbase=$renglon["idbase"];
        $base=$renglon["base"];

```

```
        echo"<p align='center'><input type='submit' name=base value=$base ></p>";
```

```
        echo"<input type='hidden' name='idbase' value='".htmlspecialchars($idbase)."'>";
```

```
        echo"<tr>";
        $i++;
    }
```

```
    echo"</table>" ;
```

```
    echo"</form>";
```

```
    echo"<form method=POST action='Esc_Login_Passwd.php' target='_parent'>";
```

```
    echo"<p align='center'><input type='submit' value='Regresar'> </p>";
    echo"</form>";
```

```
    break;
```

```
    default:
```

```
        echo "";
```

```
    break;
```

```
    }
```

```
?>
```

TESIS CON
FALLA DE ORIGEN

Catalogo_usuario.php

```

<?php
    $conexion=pg_connect("dbname=DICCIONARIO user=**** password=****")
    or die("<center> No se pudo establecer la conexion con el servidor de BD </center>");

    $sql="select usuario.login, usuario.idusuario from usuario, usuariobase where
    usuario.idusuario=usuariobase.idusuario and usuariobase.idbase=$idbase and
    usuariobase.owner='f'";

    $sql_result=pg_exec($conexion,$sql) or die ("No se pudo ejecutar la busqueda");

    $num =pg_numrows($sql_result);

    echo"<div align='center'>";
    echo"<table >";
    echo"<tr>";
    echo"<td width='205' height='82' valign='middle'>";
    echo"<form method='POST' U-File='C:\inetpub\Clustering\Tesis_private\form_results.txt'>";
    echo"<font >Usuario: <p>";
    $i=0;
    while($i<$num)
        {
            $renglon=pg_fetch_array($sql_result,$i);
            $idusuario=$renglon["idusuario"];
            $login=$renglon["login"];
            echo"<option value=$idusuario>$login";
            $i++;
        }
    echo"</select><input type='submit' value='Borrar' name='B1' >";
    echo"</form>";
    echo"<p></td>";
    echo"</tr></table>";
    echo"</div>";
?>

```

3.4.4 USUARIO DE BASE

Por otro lado el sistema ofrece tanto a los visitantes como a los usuarios de base, una página muy similar donde podrán consultar y modificar los datos de las bases a las que tienen permiso. Esta página se llama Alimentar base y se muestra en la figura 29.

ESTA TESIS NO SALE
DE LA BIBLIOTECA

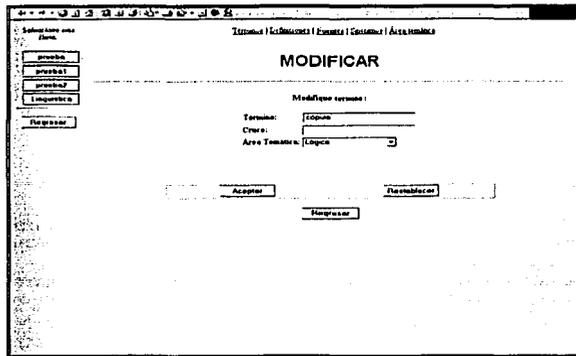


Figura 33. Modificar

El proceso de borrado se ejemplifica en la siguiente figura, donde antes de ejecutar el proceso, el sistema confirma que el usuario realmente quiere llevar a cabo la acción que esta a punto de realizar.

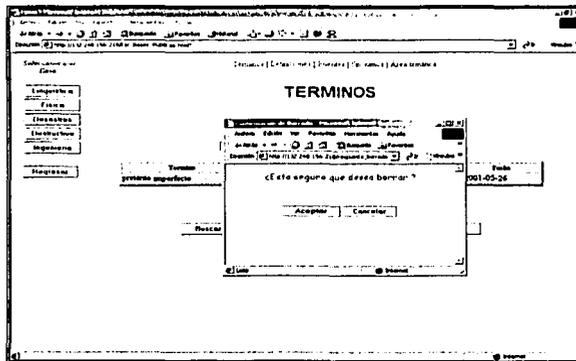


Figura 34. Borrar

TESIS CON
FALLA DE ORIGEN

Como se ha visto anteriormente, el sistema consta con una serie de páginas que permiten al usuario navegar por el sistema con seguridad, además de manipular los datos de la base en tiempo real. Los procesos de insertar, modificar, y consultar se realizan a través del siguiente código.

Operación.php

```

<?
switch ($opcion)
{
    case Insertar:
        echo"base=$base<p>";
        echo"termino=$termino<p>";
        echo"cruce=$cruce<p>";
        echo"login=$login<p>";
        echo"fecha=$fecha<p>";
        echo"pagina=$pagina<p>";*/
        pg_close($conexion);
        $conexion=pg_connect("dbname=DICCIONARIO user=marlene password=marlene") or
        die("<center> No se pudo establecer la conexion con el servidor de BD </center>");
        $sql="select max(idtermino) from termino";
        $sql_result=pg_exec($conexion,$sql) or die ("No se pudo ejecutar la busqueda del termino");
        $num =pg_numrows($sql_result);
        echo"$idtermino";
        echo"";
        break;

    case Modificar:
        echo"base=$base<p>";
        echo"termino=$termino<p>";
        echo"cruce=$cruce<p>";
        echo"login=$login<p>";
        echo"fecha=$fecha<p>";
        echo"pagina=$pagina<p>";*/
        pg_close($conexion);
        $conexion=pg_connect("dbname=DICCIONARIO user=marlene password=marlene") or
        die("<center> No se pudo establecer la conexion con el servidor de BD </center>");
        $sql="select max(idtermino) from termino";
        $sql_result=pg_exec($conexion,$sql) or die ("No se pudo ejecutar la busqueda del termino");
        $num =pg_numrows($sql_result);
        echo"$idtermino";
        echo"";
        break;

    case Consultar:
        echo"base=$base<p>";
        echo"termino=$termino<p>";
        echo"cruce=$cruce<p>";
        echo"login=$login<p>";
        echo"fecha=$fecha<p>";
        echo"pagina=$pagina<p>";*/
        pg_close($conexion);
        $conexion=pg_connect("dbname=DICCIONARIO user=marlene password=marlene") or
        die("<center> No se pudo establecer la conexion con el servidor de BD </center>");
        $sql="select max(idtermino) from termino";
        $sql_result=pg_exec($conexion,$sql) or die ("No se pudo ejecutar la busqueda del termino");
        $num =pg_numrows($sql_result);
        echo"$idtermino";
        echo"";
        break;

    case Regresar:
        echo"tabla=$tabla";
        echo"base=$base";
        echo"<href='\"Datos.php?base=$base&&tabla=$tabla\" target='\"Contenido\"'>";
        break;
}
?>

```

4 CONCLUSIONES

Los objetivos fundamentales que dieron origen al desarrollo de la presente tesis han sido cubiertos satisfactoriamente. Como resultado de ello podemos mencionar que se ha obtenido una interfaz integral para la creación y mantenimiento de diccionarios terminológicos; interfaz que proporciona un acceso sencillo, confiable, seguro y cómodo a los diccionarios a través del web.

El sistema implantado cuenta con cuatro tipos de usuario: el administrador del sistema, el administrador de la base de datos, el usuario y el visitante. Cada tipo de usuario tiene funciones específicas, estas funciones van desde la simplemente la consulta hasta la creación y mantenimiento de nuevos diccionarios.

Una de las mejoras más importantes del sistema, es que con base en los diferentes tipos de usuarios, el sistema permite delegar responsabilidades de administración de las bases. De esta manera el administrador del sistema no tiene que estar constantemente dando de alta nuevos usuarios, pues delega esta responsabilidad en el administrador de la base de datos, quien a su vez, para dar de alta un usuario, lo agrega a una tabla por medio de una interfaz y no por medio de las tablas y utilerías del sistema manejador de base de datos. Este hecho permite mejorar significativamente la carga de responsabilidades y la seguridad del sistema.

Por otra parte entre las funciones del Administrador de la base de datos destacan: dar de alta nuevos usuarios, borrarlos lógicamente del sistema, así como asignarles los permisos para cada una de sus bases.

El sistema planteado permite el acceso y ejecución de las herramientas de Lingüística Computacional como son: El proceso de Clustering y Contextos Definitorios además de las herramientas que se requieran agregar posteriormente.

En cuanto al desarrollo del sistema e implantación, las personas que integran el equipo de trabajo deben tener asignadas tareas específicas: alguien que analice completamente los

requerimientos del sistema y efectúe los diagramas y esquemas necesarios, otro más que realice el diseño e implantación de la base de datos, y que se dedique a programar los esquemas o diagramas resultante, y finalmente alguien más que pruebe la funcionalidad del sistema.

Finalmente, como todo sistema, se espera que evolucione a fin de agregar nuevas funcionalidades y mejorar las ya existentes, corrigiéndose en el camino aquellos problemas y defectos encontrados durante su operación.

BIBLIOGRAFÍA

Libros

- * *Edición Especial Linux. Máxima Seguridad*, Pearson Educación, S.S., Madrid, 2000
- * F. Korth Henry, *Fundamentos de Bases de Datos*, McGraw-Hill, 2ª. Edición
- * Laudon Kenneth C., *Administración de los Sistemas de información*, 3ª. Edición
- * Lucas Gómez Angel, Romero García Paloma, *Diseño y Gestión de sistemas de Bases de datos*, Editorial Paraninfo
- * Luke Welling, Laura Thompson , *PHP and MySQL Web development*
- * Roger S. Pressman, *Ingeniería del Software*, McGraw-Hill, 3ª. Edición

Sitios Web

- * http://jornadas.sourceforge.net/dardo_guidobono/ProgAvanzadaPHP
- * <http://lucas.hispalinux.es/Postgresql-es/web/navigable/tutorial/x56.html>
- * http://www.gda.itesm.mx:8080/www/tw2001_archivos/frame.htm
- * <http://www.geocities.com/SiliconValley/Campus/2208/WEapache.html>
- * <http://www.htmlweb.net/linux/linux.html>
- * <http://www.inode.es/aulas/informatica/browsers.html>
- * <http://www.monografias.com/trabajos5/basede/basede.shtml>
- * http://www.mundofree.com/joanmmas/Diseno_Web/Edicion_web/DHTML/dhtml.html
- * <http://www.ucm.es/info/Psyap/Prieto/alum9798/intranet01/internet.htm>
- * <http://www.ucm.es/info/Psyap/Prieto/alum9798/intranet01/techweb.htm>

METODOLOGÍA PARA EL DESARROLLO DE SITIOS WEB

En el desarrollo de sistemas es necesario el uso de alguna metodología que permita al desarrollador entender de una mejor forma que es lo que el sistema va a realizar y como es que interactúan entre si las partes que componen al sistema, la relación que hay entre estas y los procesos que intervienen en un momento dado para que el sistema realice las funciones que se esperan. Además, da una visión general de los scripts o códigos que se usaran para realizar una acción específica, como por ejemplo validar. Por esta razón se considera conveniente hacer una representación esquemática del sistema.

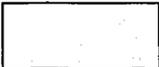
Se considerará algún tipo de simbología, así como los conceptos de escenarios y excepciones que esquematicen un planteamiento de la interfaz Web. Comenzaremos con definir los conceptos de escenario y excepción.

Un **escenario** nos ayuda a entender la relación que existe entre páginas Web teniendo como resultado los diferentes comportamientos que surgen al momento de interactuar de una página a otra. Entendiendo como comportamientos las distintas acciones que se pueden presentar en una página.

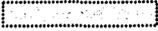
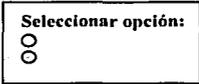
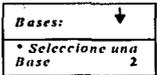
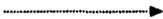
Una **excepción** es aquella que surge cuando existe algún tipo de interferencia o error inesperado que puede ser o no producido por el sistema al momento de que el usuario realice un envío de información que el mismo tecleo.

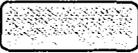
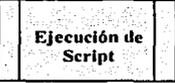
A continuación se define la simbología que utilizada para hacer la representación esquemática de los escenarios y excepciones de la Interfaz Web del Diccionario onomasiológico para el Instituto de Ingeniería.

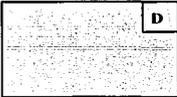
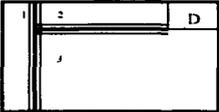
SIMBOLOGÍA PROPUESTA:

SIMBOLO	DEFINICIÓN
	<p>Página Web Estática. Es una página que presenta información estática (sin cambios) para el usuario, (formularios, imágenes, frames, etc.) o presenta opciones de que permitan navegar al usuario a través del sistema.</p>

TESIS CON
FALLA DE ORIGEN

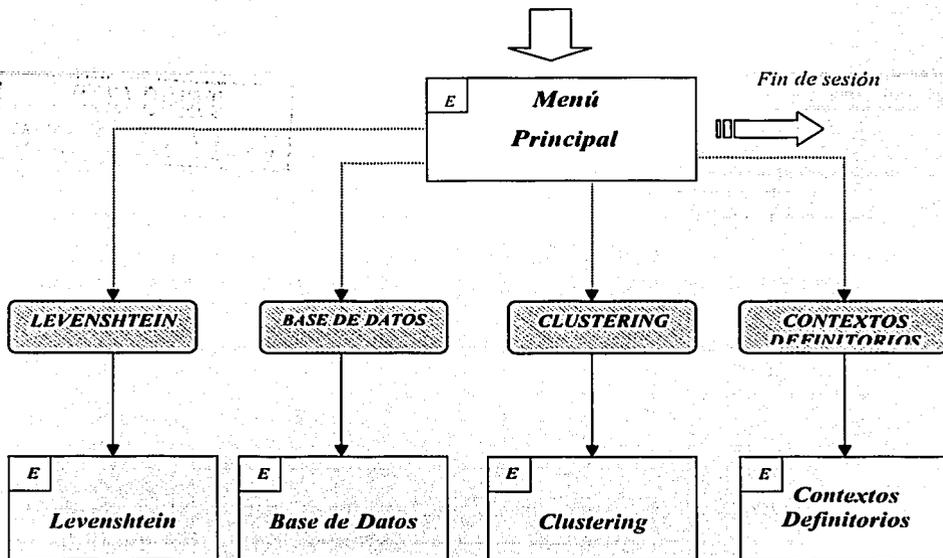
	<p>Escenario. Conjunto de páginas que nos permite analizar los diferentes comportamientos que se tienen en el sistema. Sólo aparece cuando se hace la definición del mismo.</p>
	<p>Caja de texto. Propiedad de la página que permite introducir información que es de interés para el usuario (como un formulario).</p>
	<p>Despliegue de liga. Representación que se hace para indicar el traslado de una página Web a otra.</p>
	<p>Mensaje página Web. Es el mensaje que presenta una página Web que puede ser o no dinámica al momento de realizar algún tipo de operación realizada por parte del usuario.</p>
	<p>Acceso de Información. Indica que la Información proporcionada por el usuario entrará a la base de datos para ser dada de alta o bien para verificar si esta existe.</p>
	<p>Botón. Es una propiedad de la página Web que presenta una opción a ser ejecutada por el usuario.</p>
	<p>Mensaje de Java Script. Es un mensaje dinámico proporcionado por el sistema, si la información que se envía (mediante formularios) es válida para el sistema.</p>
	<p>Liga. Es una propiedad de la página Web que permite al usuario ir al escenario que se indica en el texto sobre la línea (hipertexto). Aparece el punto arriba de la línea si hay más de una liga, exceptuando listas de resultados y paginación.</p>
	<p>Caja de selección. Propiedad de la página Web que permite seleccionar alguna opción de interés para el usuario.</p>
	<p>Catálogo. Propiedad de la página Web que permite al usuario elegir una sola opción a través de una cortina desplegable que presenta un conjunto de opciones que son de interés para el usuario.</p>
	<p>Propiedad de la página. Representación que se utiliza para indicar que elementos son parte de una página Web.</p>

	Entrada de escenario con login y password. Representación que se hace para indicar al usuario que debe de introducir su login y password para poder tener acceso a una página Web.
	Entrada de escenario normal. Representación que se utiliza cuando el usuario desea entrar a un conjunto de páginas Web que están relacionadas entre si.
	Salida de escenario. Representación que se utiliza para indicar al usuario la salida hacia un escenario especificado por el texto que le sigue. Nota: Este símbolo representa una y solo una salida en un escenario.
	Regresar a la misma página. Es una propiedad de la página Web que permite al usuario ir a la misma página cuando se ha ejecutado alguna acción.
	Mensaje de Java Script. Es un mensaje que aparecerá dentro de la página Web con el fin de proporcionar algún tipo de información útil para el usuario.
	Botón de aceptación. Propiedad de la página Web que permite al usuario confirmar la información que se presenta en la página.
	Botón de cancelación. Propiedad de la página Web que permite cancelar algún tipo de información que en el instante no es de interés ara el usuario.
	Botón de limpiar. Propiedad de la página Web que permite limpiar el formulario de una página Web después de haber metido información en este.
	Botón. Propiedad de la página Web que permite ejecutar una acción determinada indicada por el nombre del botón.
	Botón opcional. Propiedad de la página Web que aparecerá sólo si se cumple la o las condiciones especificadas por el texto aunado.
	Ejecución de un Proceso. Se ejecuta un script o código que realiza alguna búsqueda, elección, inserción o modificación en la base de datos del sistema.

	<p>Página Web Dinámica. Es una página que puede generar y presentar con un mismo código, varios resultados, dependiendo de los valores que le hayan sido dados.</p>
	<p>Excepción. Se refiere a un proceso que surge cuando existe un error inesperado, debido al tecleo de datos por parte del usuario. El símbolo # representa el número de excepción por escenario.</p>
	<p>Conector. Representa la conexión que se hace con otra parte del diagrama que se encuentre ubicado en otra página. El # representa el número de conector.</p>
	<p>Frame. Es un marco, que representa un conjunto de páginas independientes mostradas en una misma ventana. Estas páginas pueden ser dinámicas o estáticas.</p>

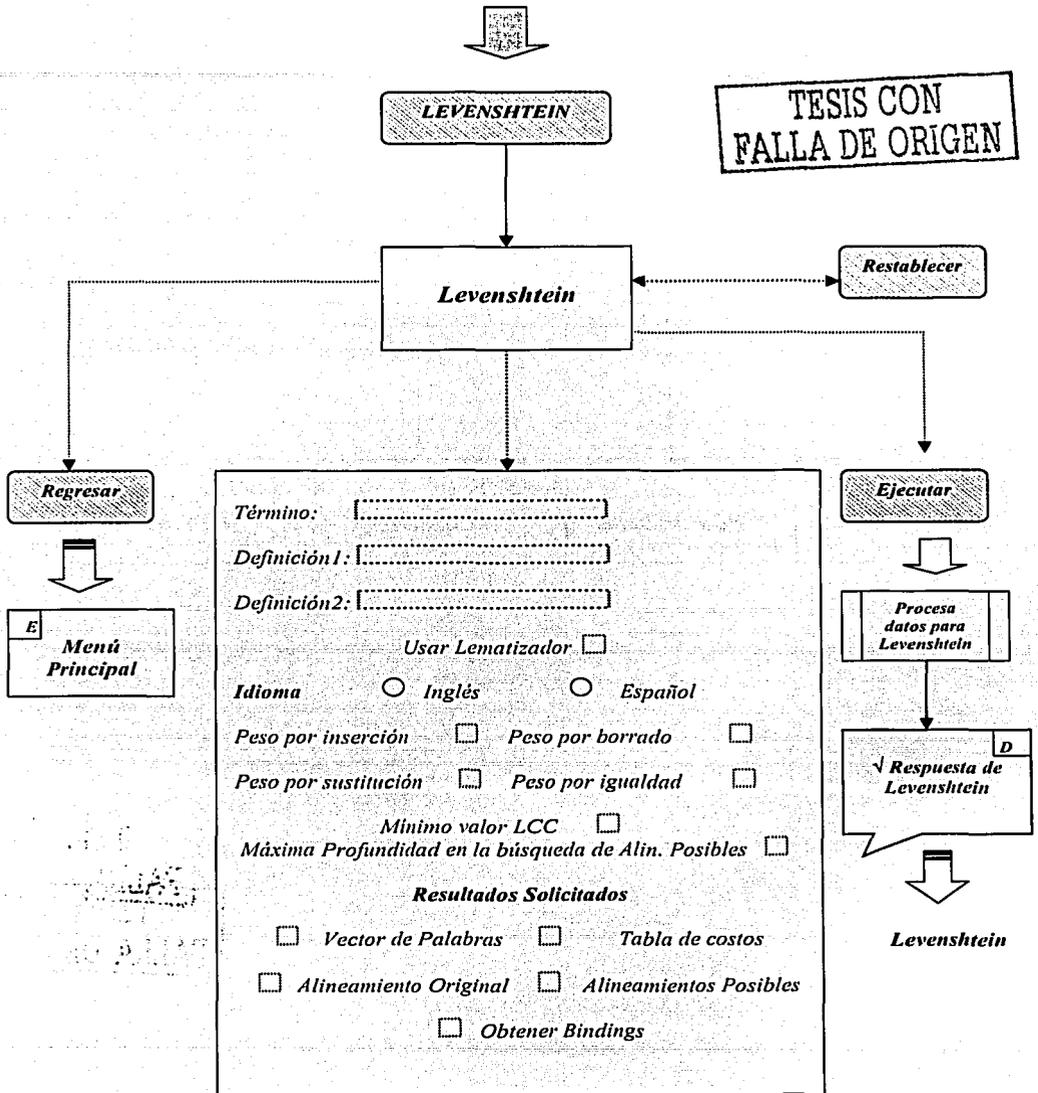
A continuación se muestran los escenarios creados para el sistema:

Escenario Menú Principal

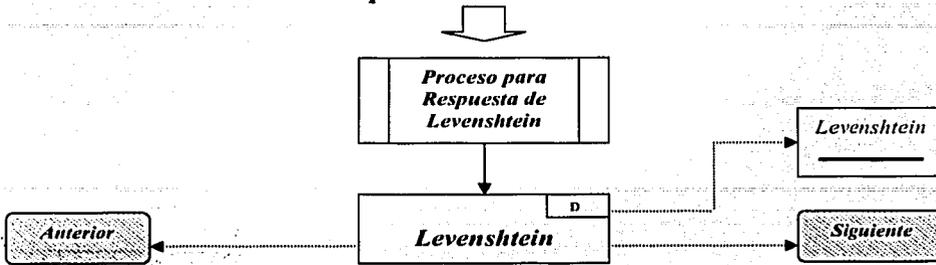


TESIS CON FALLA DE ORIGEN

Escenario Levenshtein



Respuesta de Levenshtein



Fecha del ejecutable: 11/09/2001 02:38:08 p.m.

Levenshtein distance

Gabriel Castillo Hernández

gch@pomas.lingen.unam.mx

Utiliza lematizador en inglés

DefD Full Form

al	l	imetro	un	dispositivo	para	medir	la	cantidad	de	alkali	en	luna	solucion
----	---	--------	----	-------------	------	-------	----	----------	----	--------	----	------	----------

DefD Stemmed Form

al	l	imetro	un	dispositivo	para	medir	la	cantidad	de	alkali	en	luna	solucion
----	---	--------	----	-------------	------	-------	----	----------	----	--------	----	------	----------

Dimensión de la matriz de pesos: [12x12]

Matriz de pesos de Levenshtein

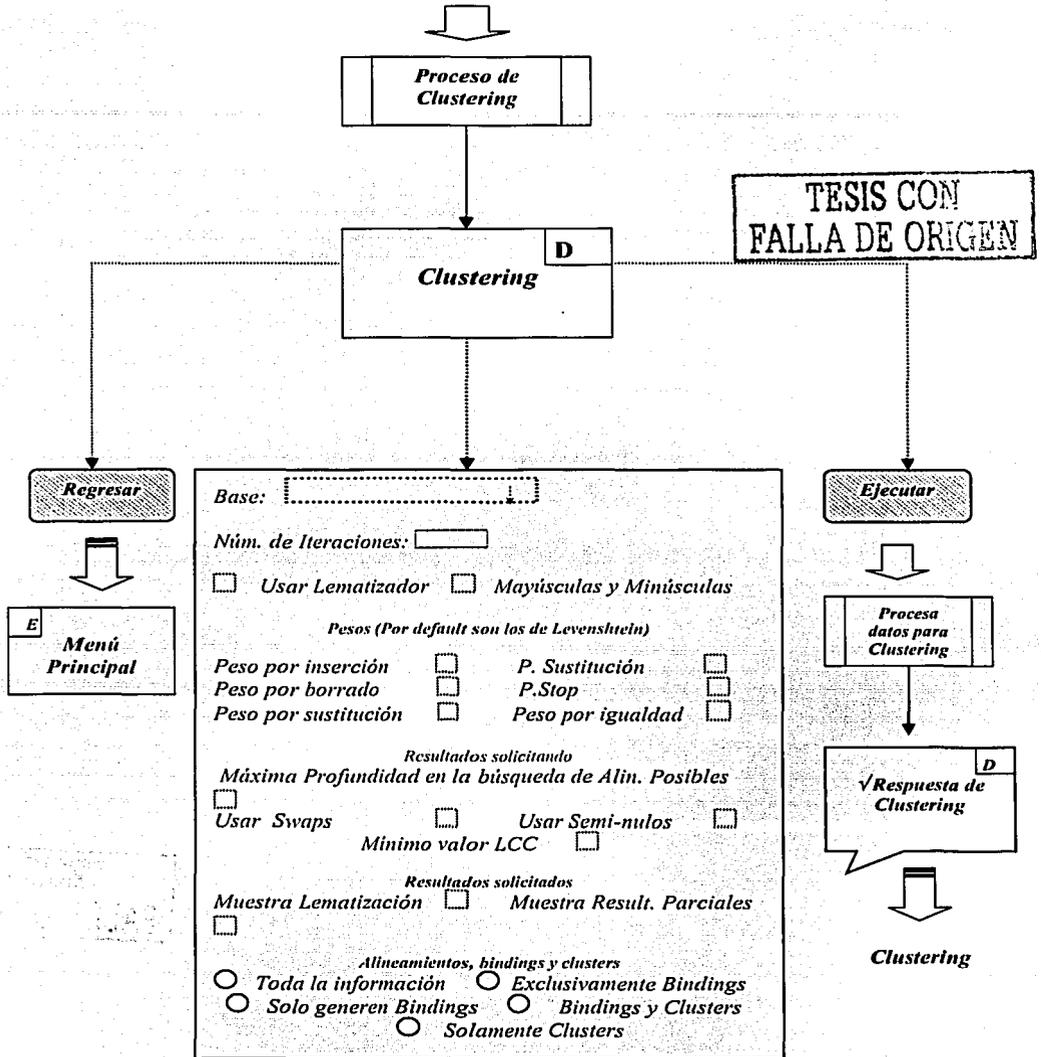
0	1	2	3	4	5	6	7	8	9	10	11	12
1	0	1	2	3	4	5	6	7	8	9	10	11
2	1	0	1	2	3	4	5	6	7	8	9	10
3	2	1	1	2	3	4	5	6	7	8	9	10
4	3	2	2	3	4	5	6	7	8	9	10	11
5	4	3	3	4	5	6	7	8	9	10	11	12
6	5	4	4	5	6	7	8	9	10	11	12	13
7	6	5	5	6	7	8	9	10	11	12	13	14
8	7	6	6	7	8	9	10	11	12	13	14	15
9	8	7	7	8	9	10	11	12	13	14	15	16
10	9	8	8	9	10	11	12	13	14	15	16	17
11	10	9	9	10	11	12	13	14	15	16	17	18
12	11	10	10	11	12	13	14	15	16	17	18	19

Peso por Inserción: 1
 Peso por Borrada: 1
 Peso por Substitución: 1
 Peso por Igualdad: 0
 Algoritmo Original*

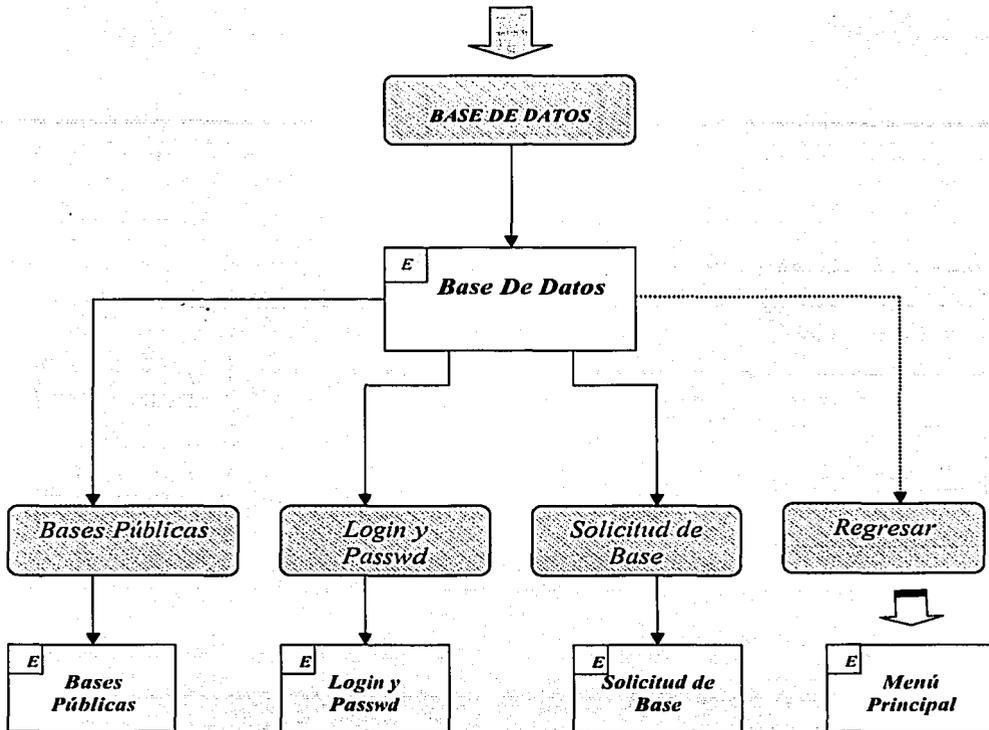
Def 1	al	l	imetro	un	dispositivo	para	medir	la	cantidad	de	alkali	en	luna	solucion
Def 2	al	l	imetro	un	aparato	para	establecer	la	cantidad	disuelta	de	alkali	en	solucion
Costos	0	0	1	1	2	2	3	4	4	4	4	5	6	
Possible binding	NO	NO	SI	NO	SI	NO	NO	NO	NO	NO	NO	NO	NO	NO
lce	0	0	4	0	1	0	0	4	0	0	0	0	0	
Tipo	equal	equal	matched	equal	matched	equal	null	matched	equal	equal	equal	null	equal	

TESIS CON FALLA DE ORIGEN

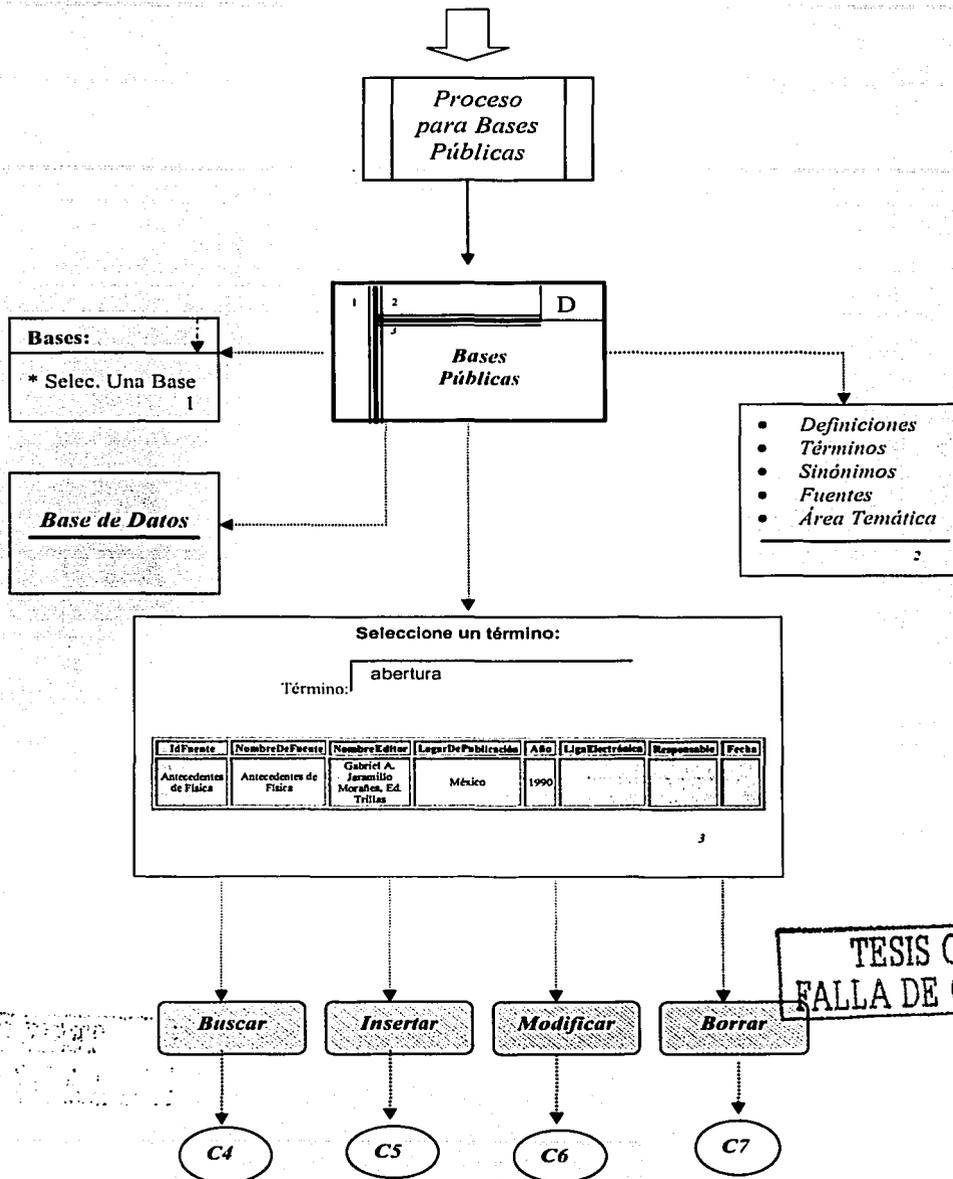
Escenario Clustering (P)



Escenario Base de Datos



Escenario Bases Públicas



Escenario Bases Públicas (continuación)

Seleccione una fuente:

Fuente:

ID Fuente	Nombre De Fuente	Nombre Editor	Lugar De Publicación	Año	Liga Electrónica	Responsable	Fecha
Anticodigos de Pláza	Anticodigos de Pláza	Gabriel A. Jaramillo Morales, Ed. Trillas	México	1990			

3

Seleccione un Área Temática:

Término:

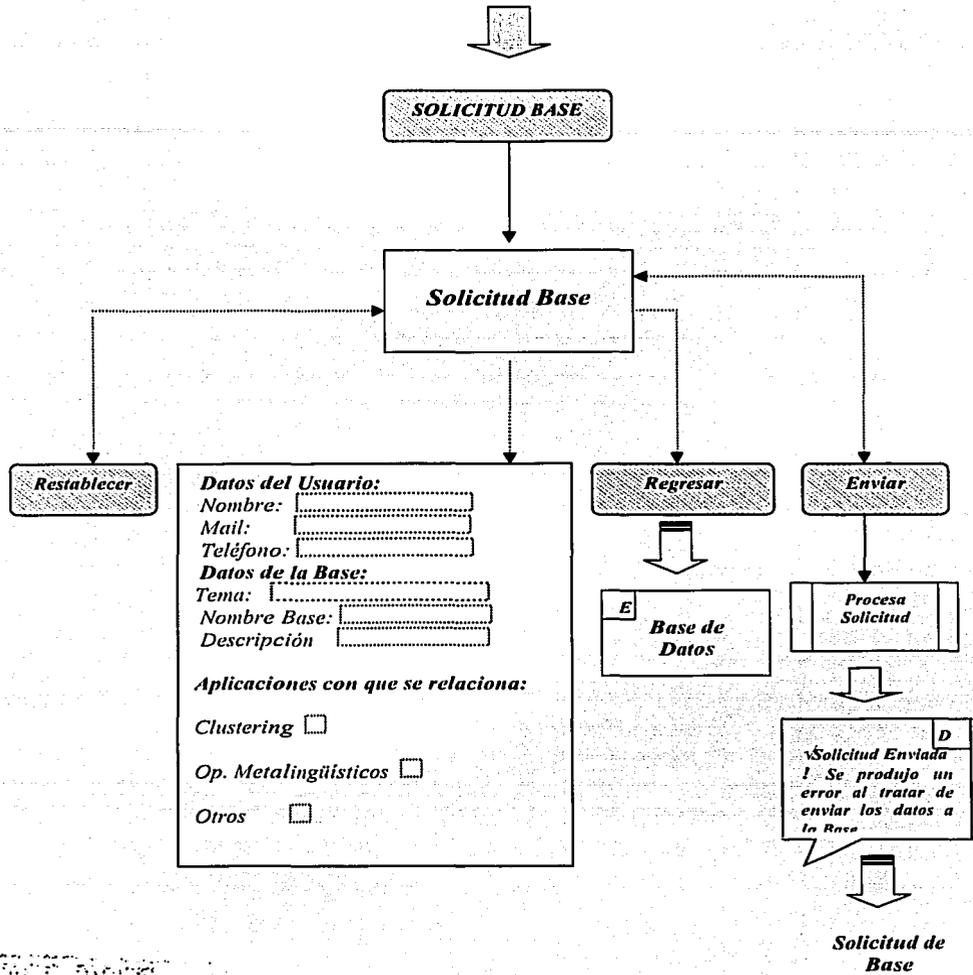
Área:

Sinonimo	Modificar	Eliminar	Responsable	Fecha

3

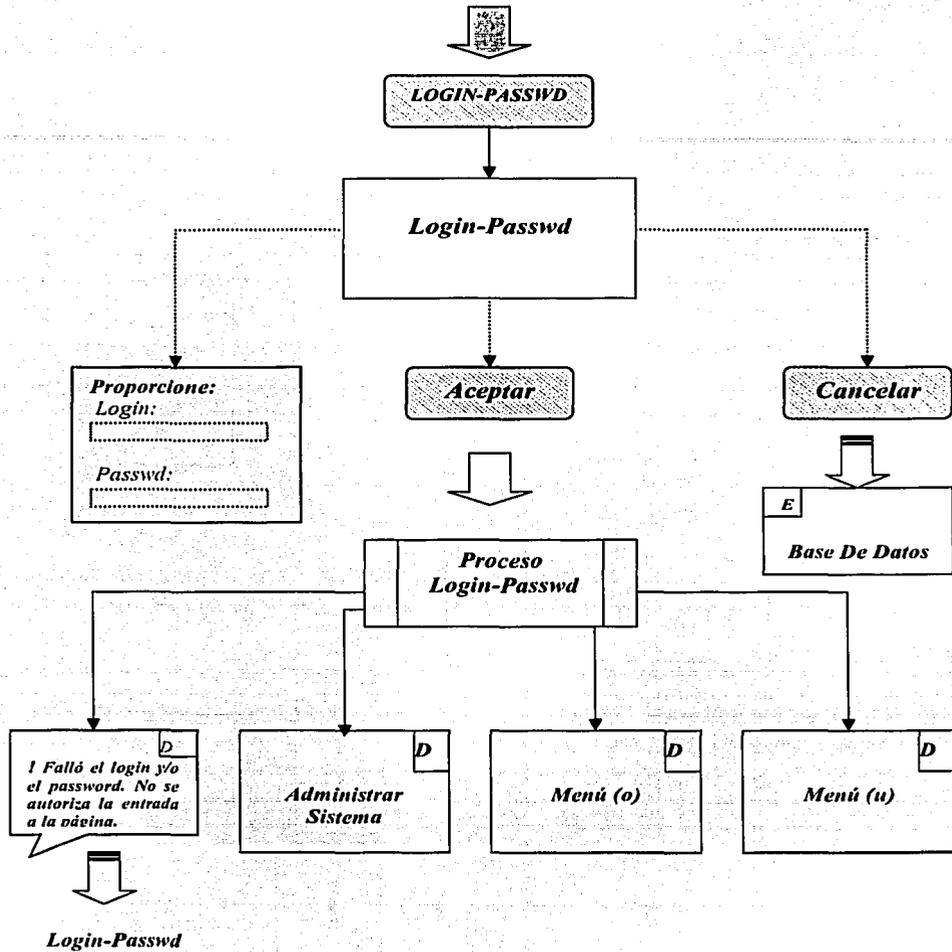
TESIS CON
 FALLA DE ORIGEN

Escenario Solicitud para Base



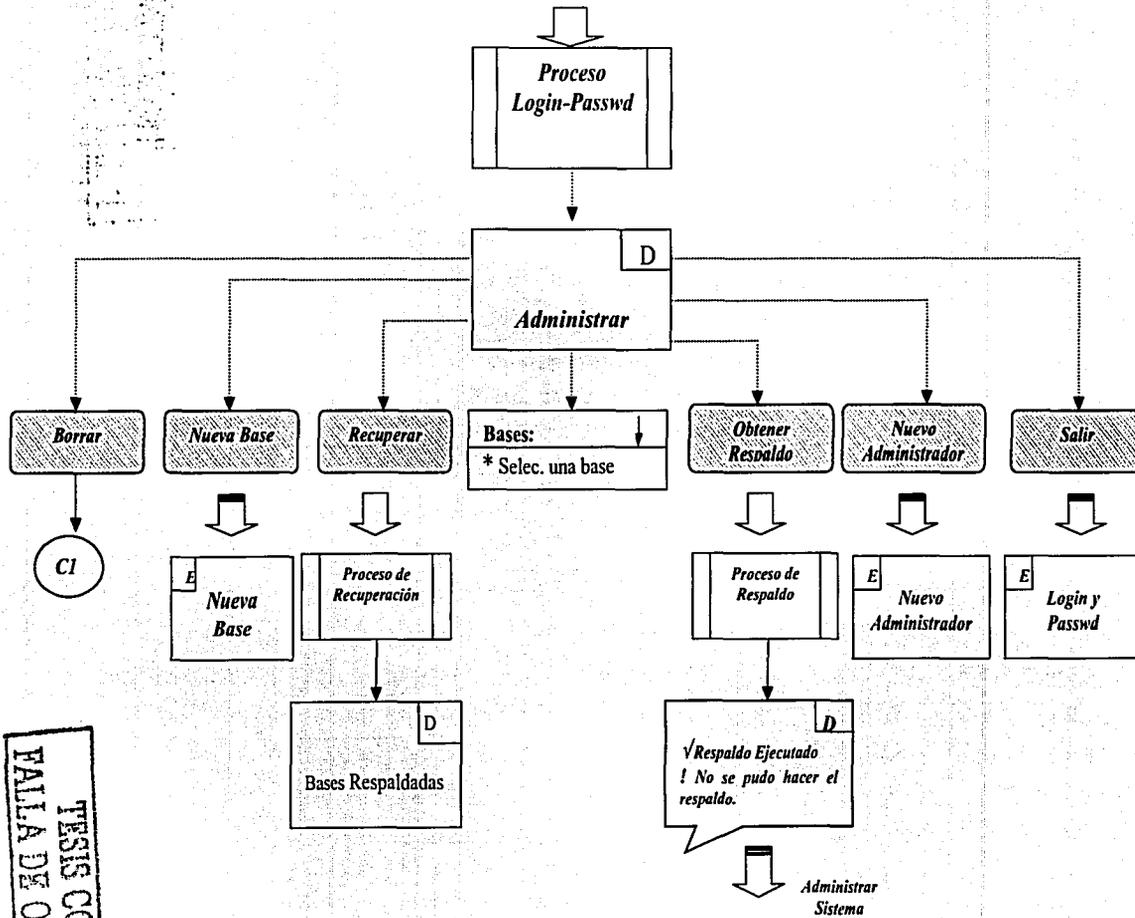
TESIS CON FALLA DE ORIGEN

Escenario Login y Password (a),(o),(u)



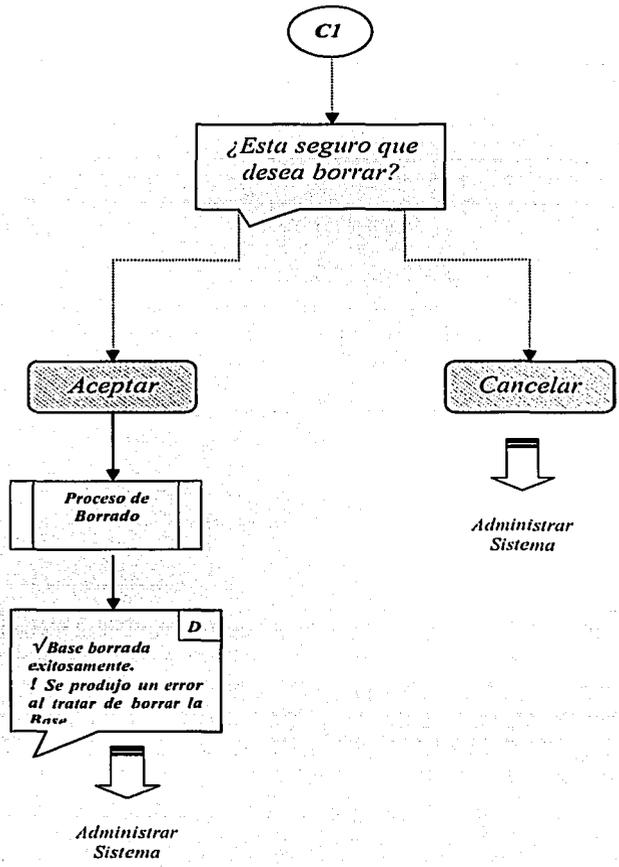
TESIS CON FALLA DE ORIGEN

Escenario Administrar Sistema



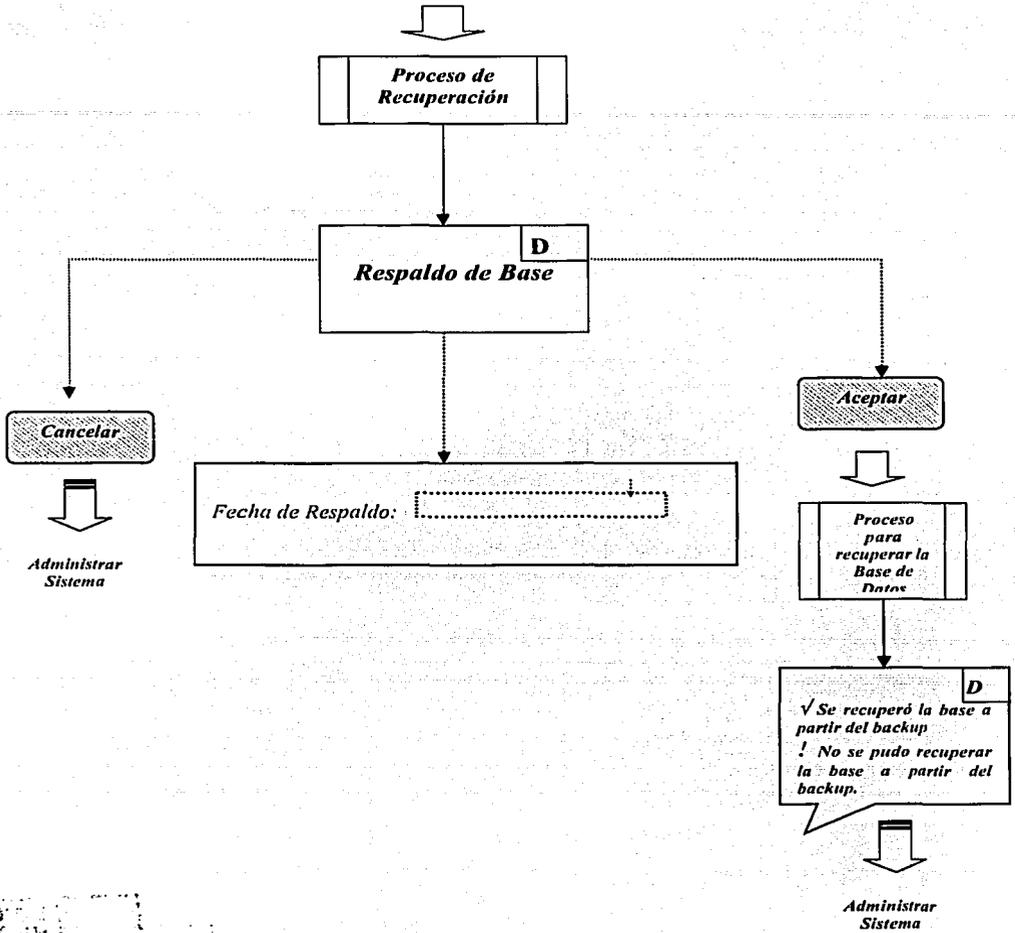
**TESIS CON
FALLA DE ORIGEN**

Respuesta de Borrado Para Escenario Administrar Sistema



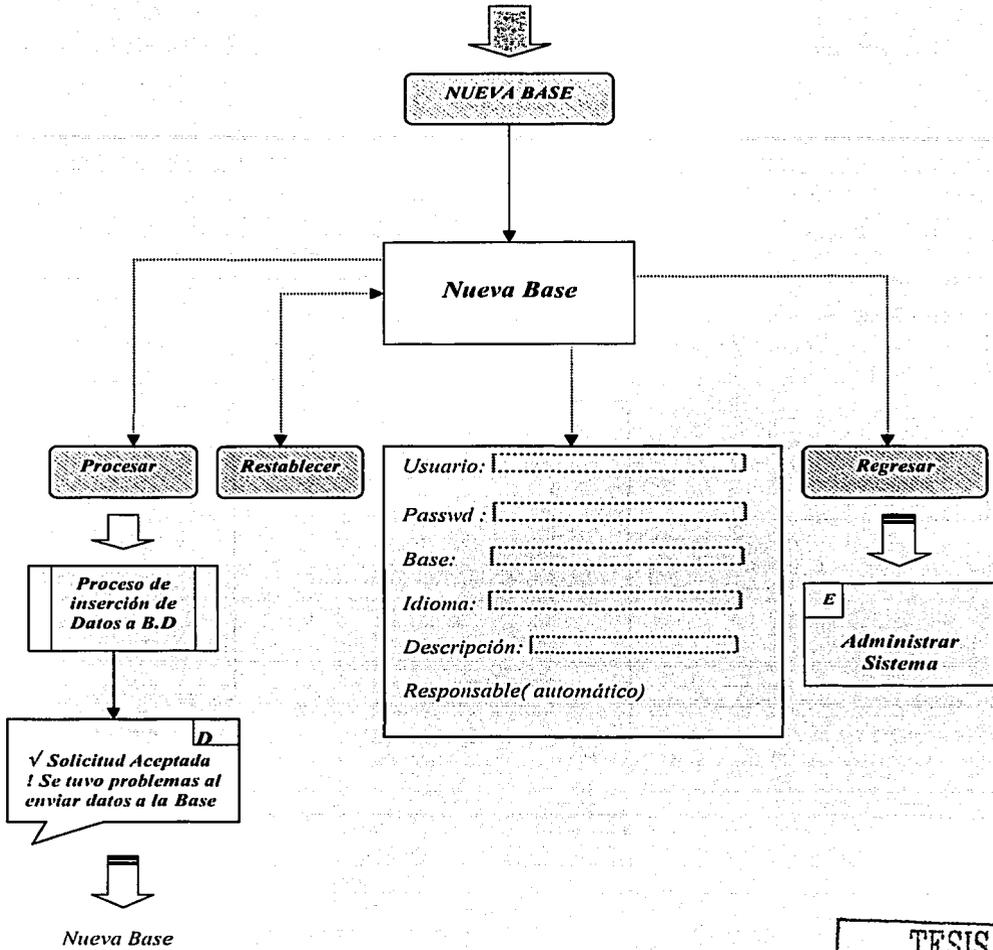
TESIS CON FALLA DE ORIGEN

Bases Respaldadas



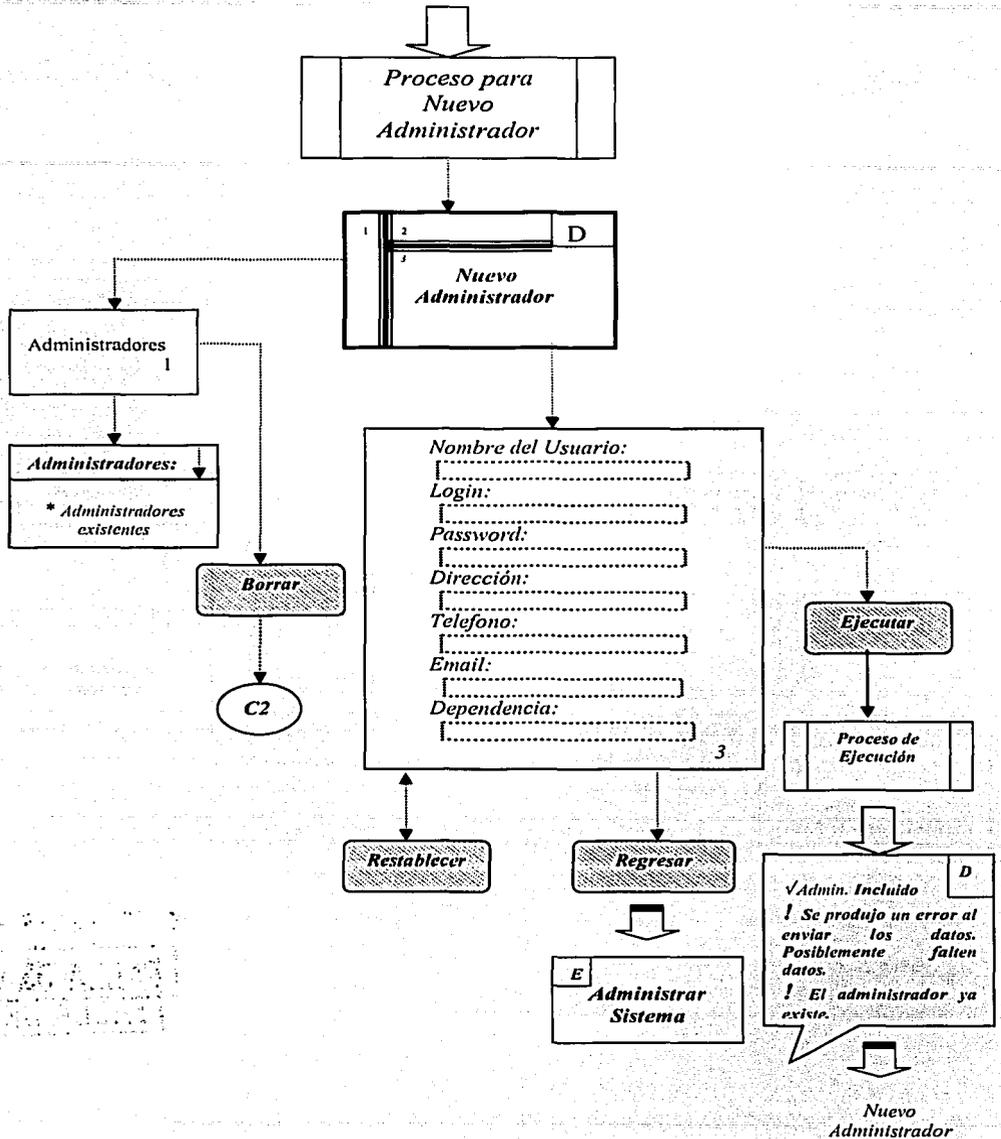
TESIS CON FALLA DE ORIGEN

Escenario Nueva Base

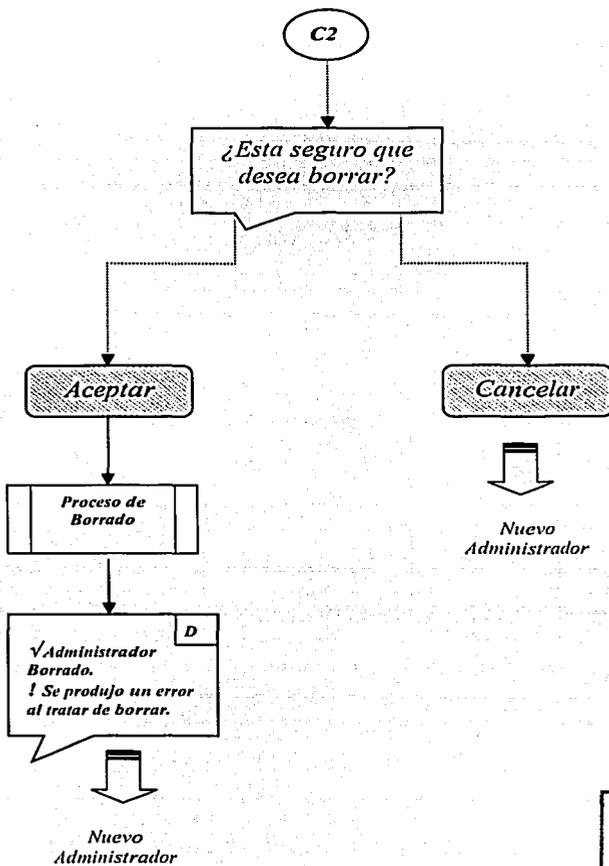


TESIS CON FALLA DE ORIGEN

Escenario Nuevo Administrador

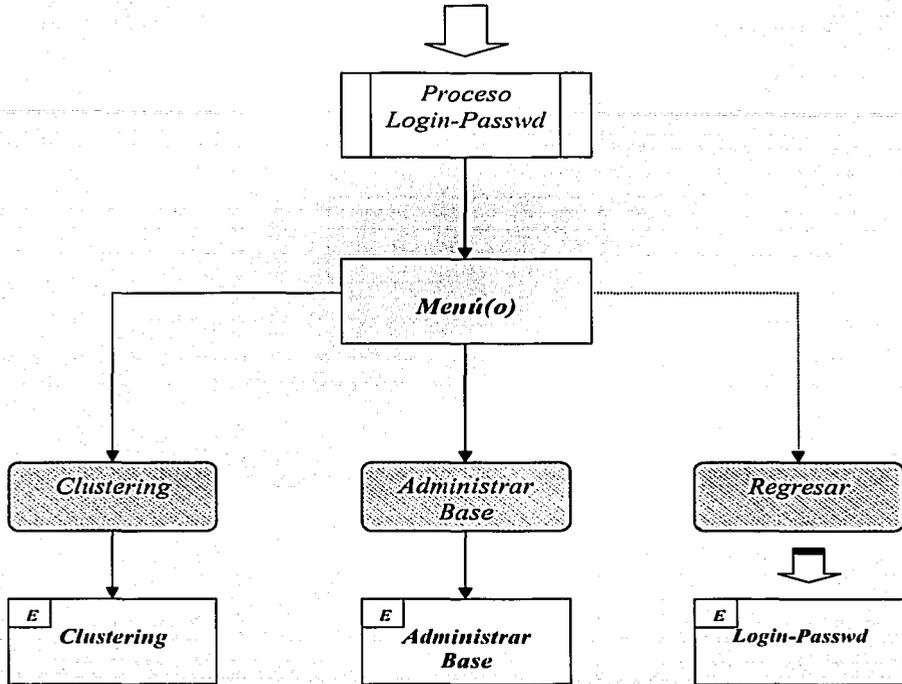


Respuesta de Borrado Para Escenario Nuevo Administrador



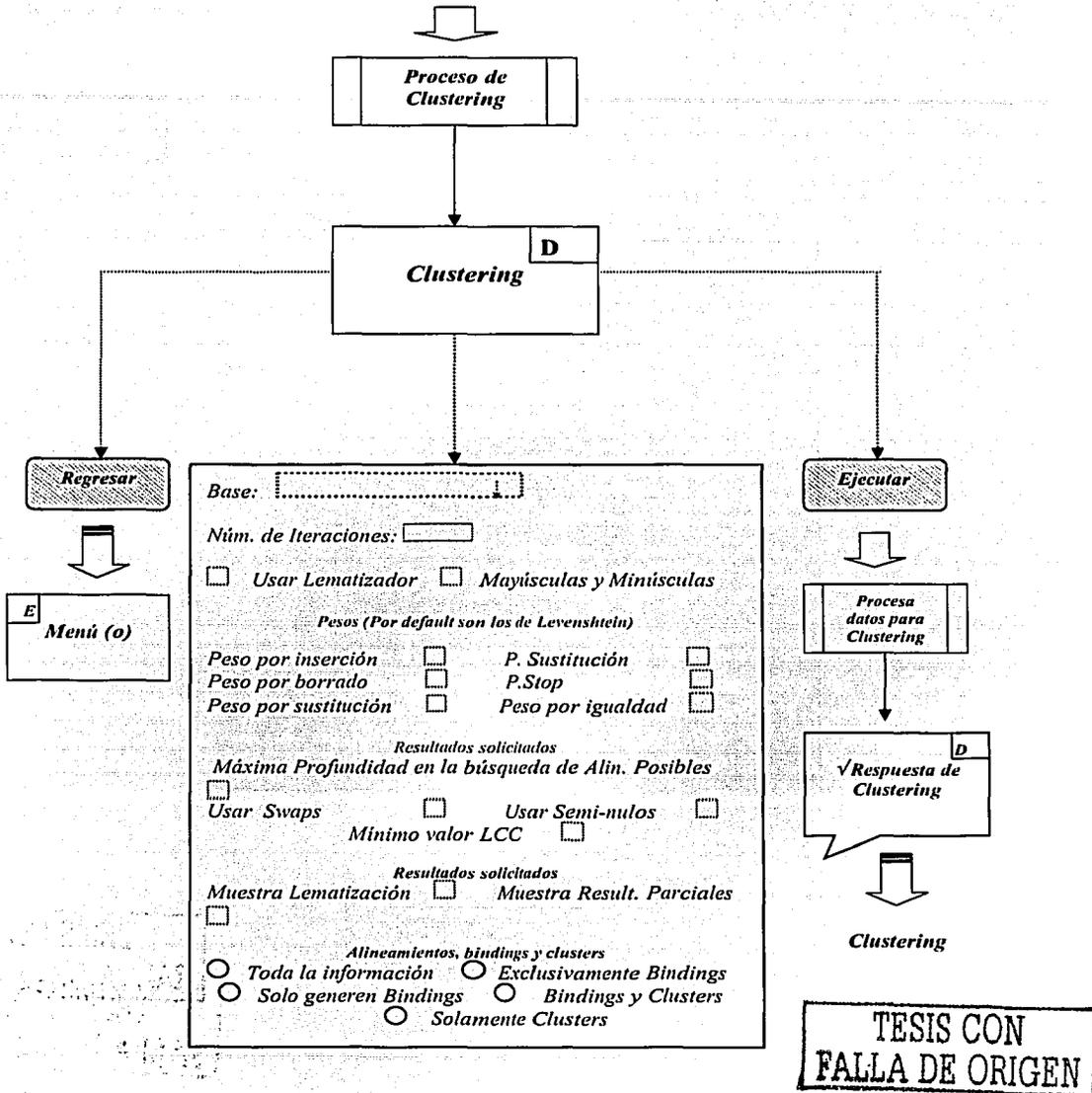
TESIS CON FALLA DE ORIGEN

Página del Menú (o)

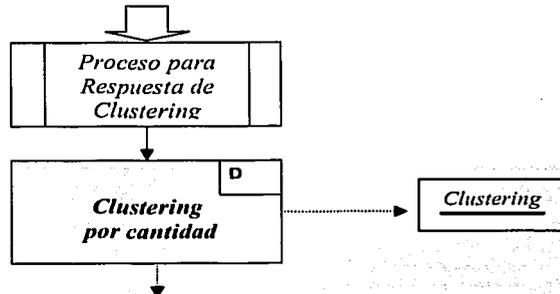


TESIS CON FALLA DE ORIGEN

Escenario Clustering



Respuesta de Clustering por cantidad (una ejecución)



Fecha del eje catabol: 11/09/2001 02:30:00 p.m.

Atendiendo procesamiento de la base.
 Conexión a TABASCO
 La base se cargó con éxito.
 Conexión e inicio de ejecución

Plan: Inscricion 1 Horario 1 Subsección 1 Igualdad 0
 No de formatos para secuencias de monedas
 Códigos terminados en letras

Terminos de grupo: 253
 Formatos para analizar: 1
 Híndago general: 0

Resultado:

[[1] Inscricion (secuencia) N=10 - 3

Detalle de las modalidades lingüísticas por las clases socioeconómicas elevadas 1910 Códice Masas Diccionario

Full Form

Inscricion	Secuencia	Id	Id	Modalidad	Lingüística	Grupo	Id							

Normalized Form

Inscricion	Secuencia	Id	Id	Modalidad	Lingüística	Grupo	Id							

Detalle de la dicción: 1910 Códice Masas Diccionario

Full Form

Inscricion	Secuencia	Id	Id	Diccion

Normalized Form

Inscricion	Secuencia	Id	Id	Diccion

La modalidad lingüística más elevada en un continuo* 1910 Códice Masas Diccionario

Full Form

Inscricion	Id	Modalidad	Lingüística	Grupo	Elevada	Id	Id	Id	Id	Id

Normalized Form

Inscricion	Id	Modalidad	Lingüística	Grupo	Elevada	Id	Id	Id	Id	Id

Forma Inf 1 [[1910 Códice Masas Diccionario] Forma Inf 2 [[1910 Códice Masas Diccionario]

Almacenamiento Original*

Inf 1	Inscricion	Secuencia	Id	Id	Modalidad	Lingüística	Grupo	Id											
Inf 2																			
Formas	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
Formas banding	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not
Id	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
Tipo	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal

Híndago:

No existieron ningún elemento el vector

Forma Inf 1 [[1910 Códice Masas Diccionario] Forma Inf 2 [[1910 Códice Masas Diccionario]

Almacenamiento Original*

Inf 1	Inscricion	Secuencia	Id	Id	Modalidad	Lingüística	Grupo	Elevada	Id										
Inf 2																			
Formas	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
Formas banding	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not	Not
Id	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
Tipo	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal	equal

Híndago:

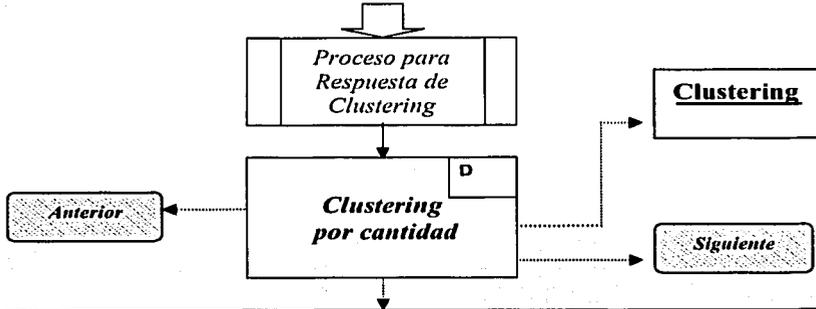
No continúan ningún elemento el vector

Cadena debug = ""

Resultado:

TESIS CON FALLA DE ORIGEN

Respuesta de Clustering por cantidad (varias ejecuciones)



Fecha del eje estable: 11/09/2001 02:10:00 p.m.
 Almacenado procesamiento de la base.
 Conexión a TABASCO...
 La base se cargó con éxito 1...
 Conexión a Limpieza...
 Para: Instancia 1 Herramienta 1 Subinstancia 1 Instancia 0
 No se detectó ningún error de ejecución.
 Última herramienta en ejecución

Terminó de ejecutarse: 251
 Errores procesados: 1
 Mensajes generados: 0
 Reprocesar

El resultado (archivo) N110 - 1
 Dicho de las modalidades lingüísticas propias de las áreas socioculturalmente alineadas. 11011 Códice Masivo Diccionario
 Falla form

Modalidad form	Alcance	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Relativo a la dicción	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas

Relativo a la dicción: 11011 Códice Masivo Diccionario
 Falla form

Modalidad form	Alcance	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Relativo a la dicción	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas

La modalidad lingüística más elevada en un contexto: 11011 Códice Masivo Diccionario
 Falla form

Modalidad form	Alcance	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Relativo a la dicción	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas

Forma Def 1 (11011 Códice Masivo Diccionario) Forma Def 2 (11011 Códice Masivo Diccionario)
 Almacenamiento Original*

Def 1	Alcance	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Def 1	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Def 2	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Alcance	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Double bracketing	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Def	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Tipo	equal	small	small	equal	equal	equal	small	matched	matched	matched	matched	matched

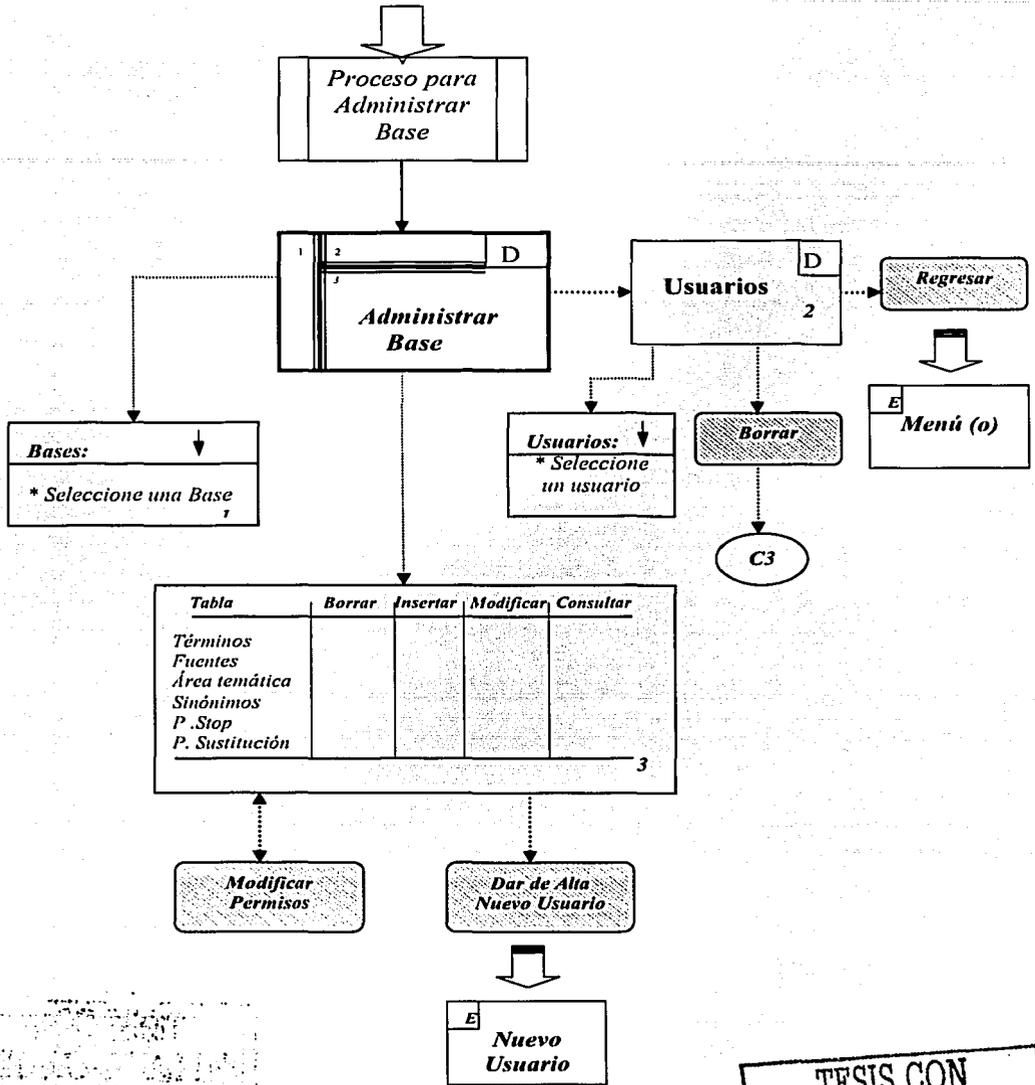
Findings
 No contiene ningún elemento el vector
 Forma Def 1 (11011 Códice Masivo Diccionario) Forma Def 2 (11011 Códice Masivo Diccionario)
 Almacenamiento Original*

Def 1	Alcance	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Def 1	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Def 2	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Alcance	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Double bracketing	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Def	Es	Es	Es	modalidades	lingüísticas	propias	de	Es	Es	Alcance	modalidades	lingüísticas
Tipo	equal	small	small	equal	equal	equal	small	matched	matched	matched	matched	matched

Findings
 No contiene ningún elemento el vector
 Cadena debug = ""
 Reprocesar

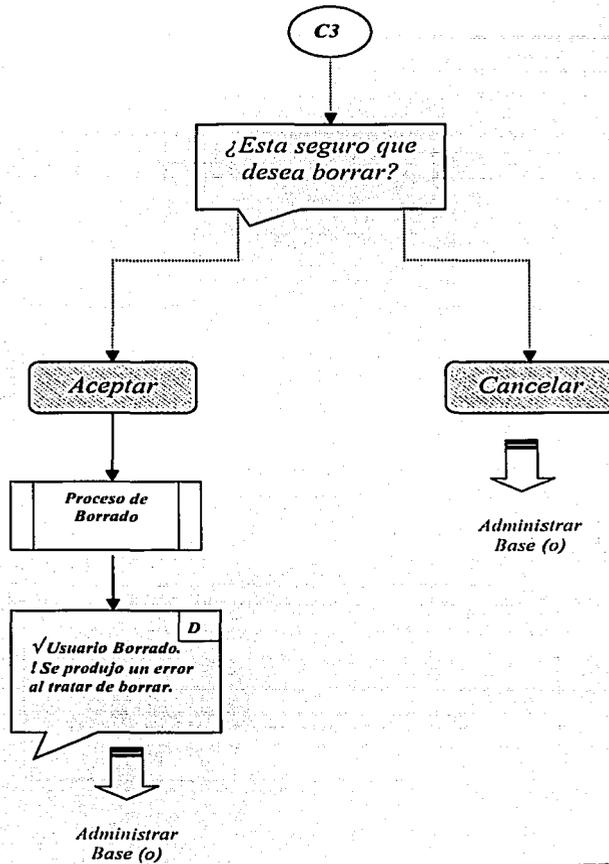
TESIS CON FALLA DE ORIGEN

Escenario Administrar Base (o)



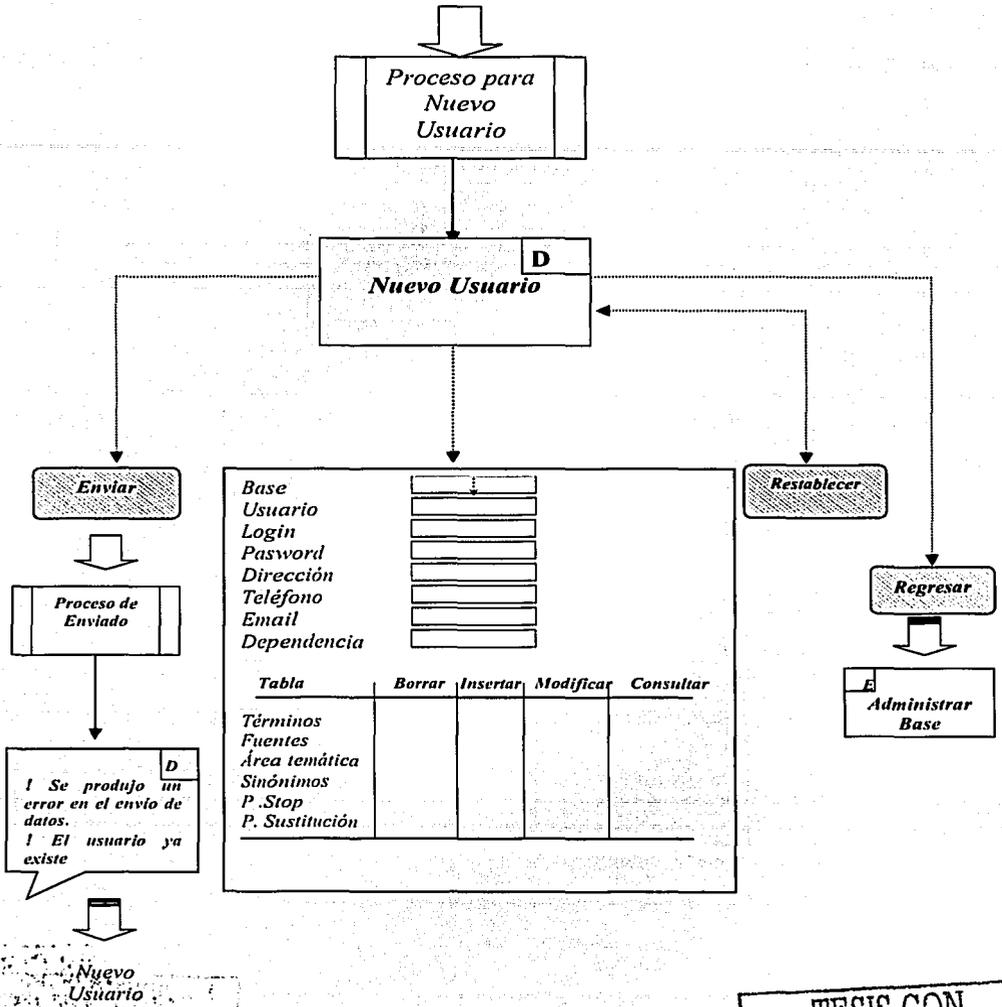
TESIS CON FALLA DE ORIGEN

Respuesta de Borrado Para Escenario Administrar Base (o)



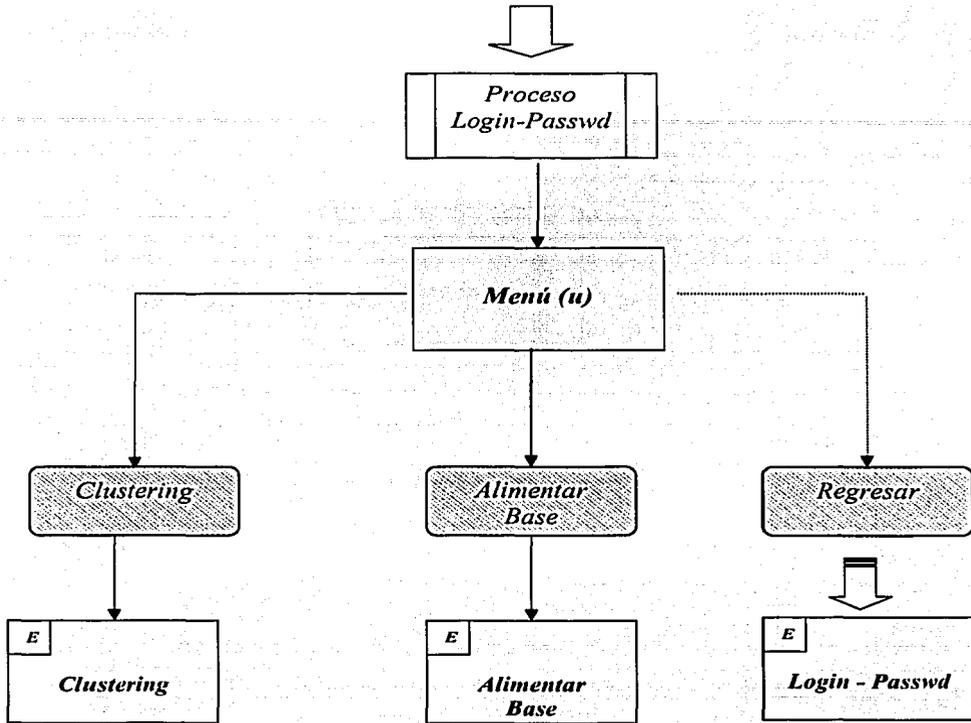
TESIS CON
FALLA DE ORIGEN

Escenario dar de Alta Nuevo Usuario (o)



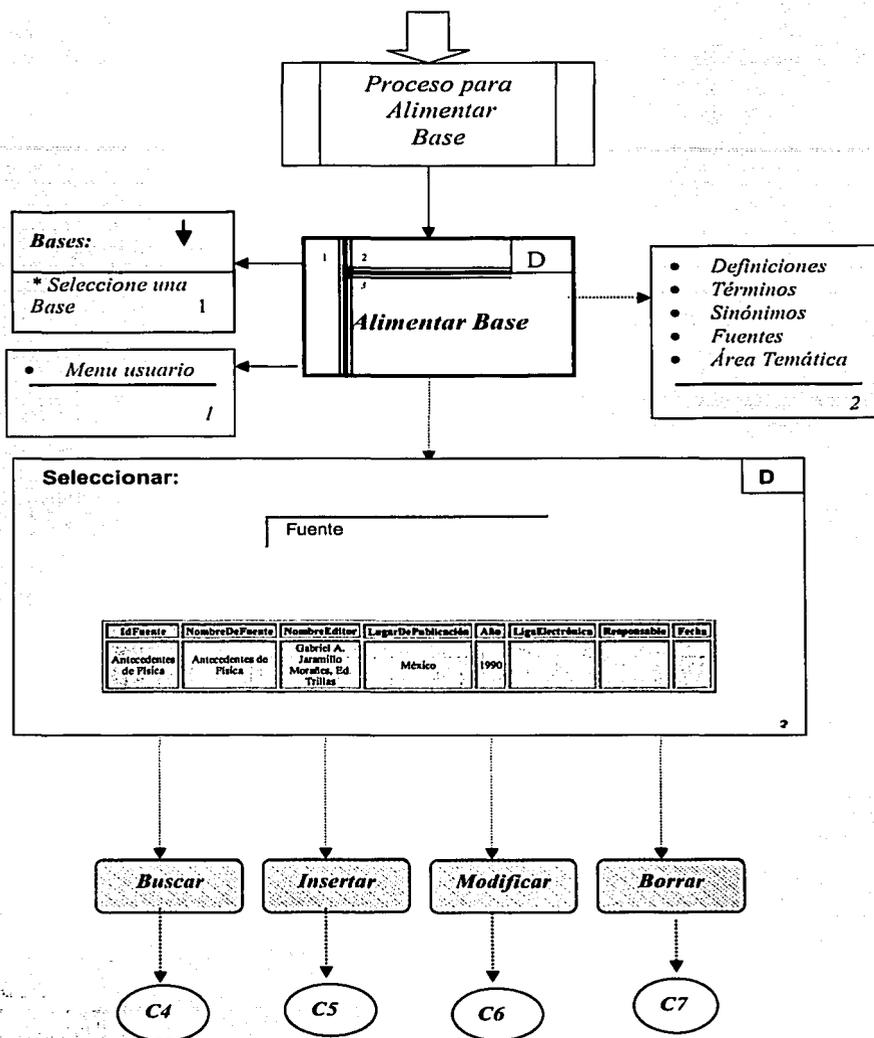
TESIS CON FALLA DE ORIGEN

Página de Menú (u)



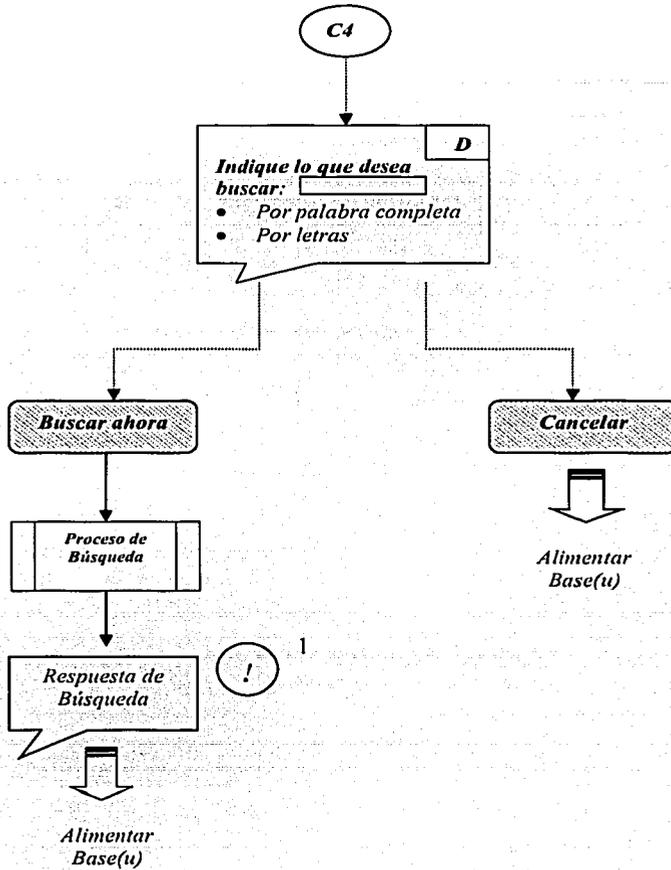
TESIS CON FALLA DE ORIGEN

Escenario Alimentar Base (u)

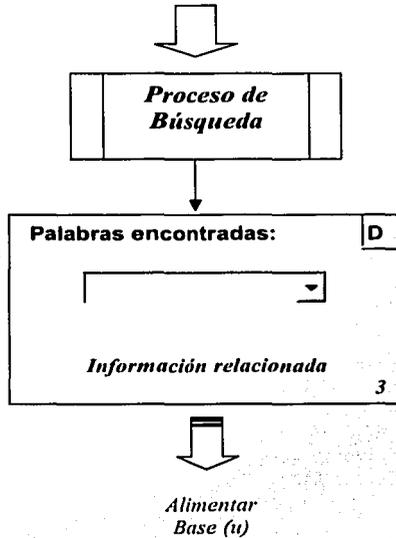


TESIS CON FALLA DE ORIGEN

Botón Buscar Para Escenario Alimentar Base (u)

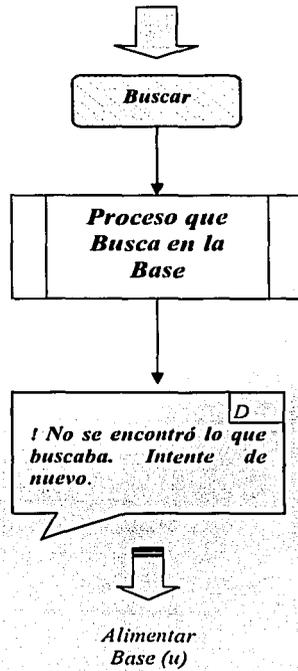


Respuesta de Búsqueda Para Escenario Alimentar Base (u)



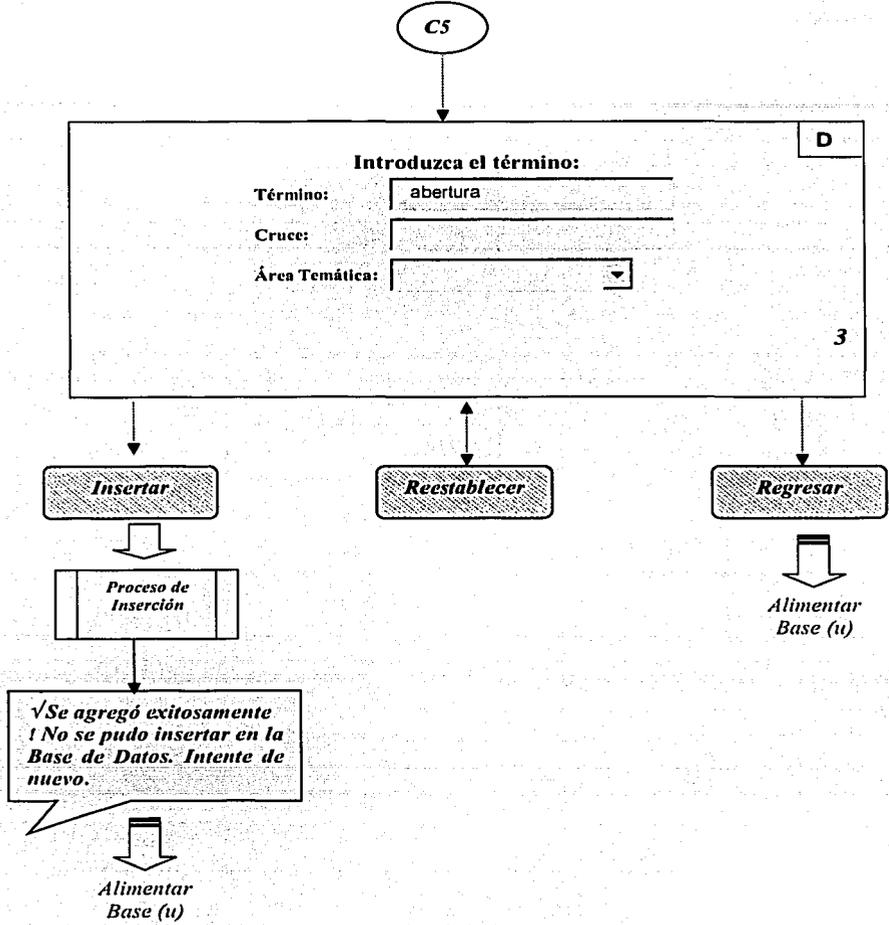
TESIS CON
FALLA DE ORIGEN

Excepción No.1 del Escenario de Alimentar Base (u)



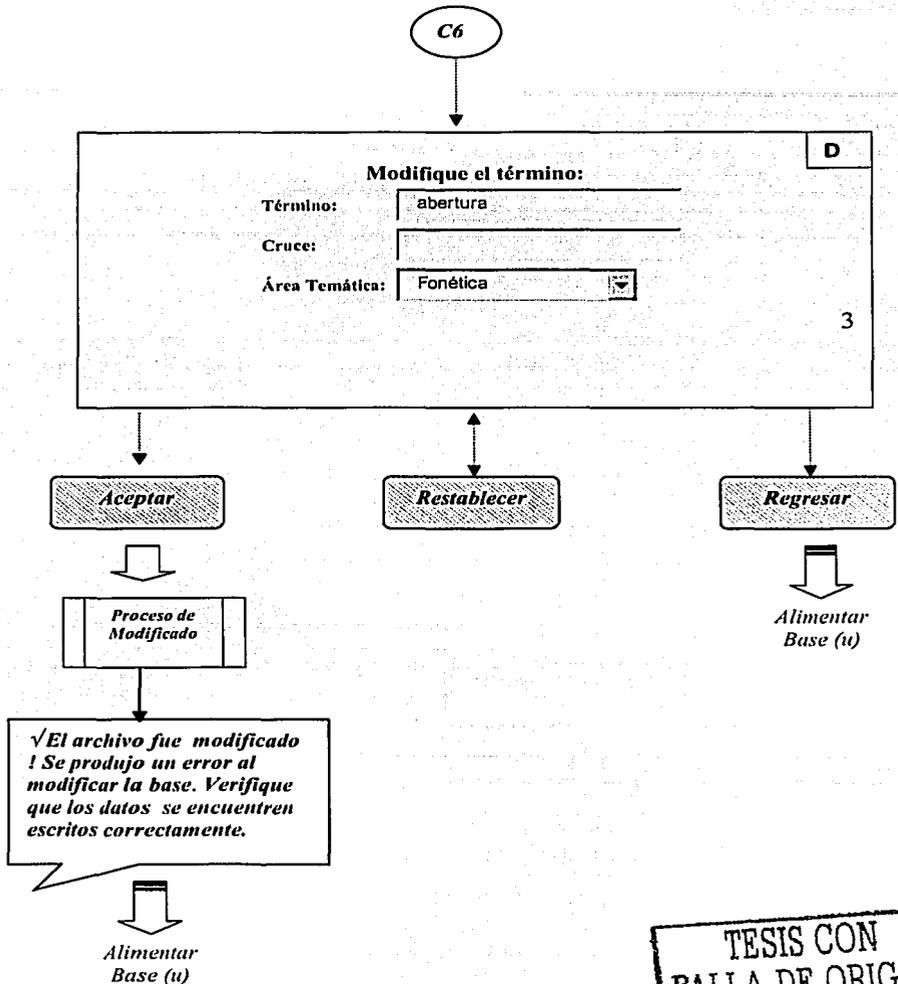
TESIS CON FALLA DE ORIGEN

Página Insertar Para el Escenario Alimentar Base (u)

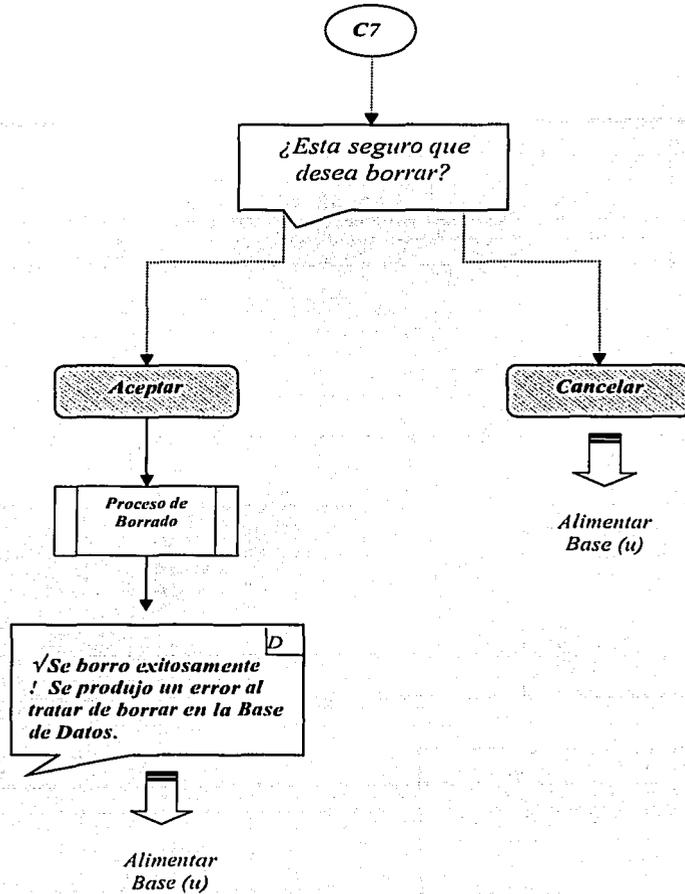


TESIS CON FALLA DE ORIGEN

Página Modificar Para Escenario Alimentar Base (u)



Respuesta de Borrado Para Escenario Alimentar Base (u)



TESIS CON
FALLA DE ORIGEN

FICHAS TÉCNICAS DEL SISTEMA

Una ficha técnica es la descripción general del contenido de una página Web. Sirve para mostrarnos las propiedades de la página, sus acciones y su interacción con otras páginas, así como los scripts generados. Dentro de las fichas técnicas describimos las páginas estáticas, los scripts que éstas generan y las páginas dinámicas. Para esto se ha seguido la siguiente descripción:

Una **página estática** contendrá las siguientes descripciones:

- **Nombre:** Se refiere al nombre de la página Web estática.
- **Entrada:** Hace referencia al nombre o los nombres de las páginas de dónde la página estática proviene.
- **Propiedades:** Son los elementos con los que cuenta la página Web. Se coloca el nombre de la propiedad, y aunado el nombre particular de cada elemento y la función del mismo.
- **Salida:** Indica la acción que se ha de realizar para salir de la página Web indicando el nombre de la página Web a la que se dirige.

Un **Script** contendrá:

- **Nombre:** Se refiere al nombre del Script.
- **Descripción:** Se refiere al o los procesos que el código genera y hace referencia al nombre de la página dinámica que es generada por el Script.
- **Parámetros que recibe:** Son los datos que se han seleccionado o tecleado en las propiedades de una página Web y que el Script recibe para ejecutar el o los procesos del Script.

Una **página dinámica** contendrá:

- **Nombre:** Se refiere al nombre de la página Web dinámica.
- **Generada por:** Hace referencia al nombre o los nombres de los scripts de dónde surge la página dinámica.

- **Propiedades:** Son los elementos con los que cuenta la página Web. Se coloca el nombre de la propiedad, y aunado el nombre particular de cada elemento y la función del mismo.
- **Salida:** Indica la acción que se ha de realizar para salir de una página Web indicando el nombre de la página Web a la que se dirige.

A continuación se ejemplificarán las fichas técnicas del sistema, basadas en los escenarios vistos en el apéndice 1.

Escenario Menú Principal

✓ **Página Estática**

- ◆ **Nombre:** Menú Principal
- ◆ **Entrada:** Página de inicio que se le mostrará al usuario final
- ◆ **Propiedades:**
 - **Botón**

Nombre	Asociado	Función
Levenshtein	A un Escenario llamado: Menú principal	Abre el Escenario Levenshtein.
Base de Datos	A un Escenario llamado: Menú principal	Abre la página estática Base de Datos para ejecutar otros procesos utilizando la base de datos del sistema.
Clustering	A un Escenario llamado: Menú principal	Invoca un Script llamado: Proceso de Clustering.
Operadores Metalingüísticos	A un Escenario llamado: Menú principal	Abre la página estática de Operadores Metalingüísticos

- ◆ **Salida:** Dando un clic en la liga llamada Salir

✓ **Scripts**

- ◆ **Nombre:** *Proceso de Clustering*
- ◆ **Descripción:** Se conecta al sistema para seleccionar de la tabla Bases, aquellas que pertenecen al Owner o al Usuario y mostrarlas en la página Dinámica "Clustering" (Públicas, Owner o Usuario).
- ◆ **Parámetros que recibe:** Información que teclea el usuario al proporcionar su login y passwd, en caso de que el proceso se ejecute desde el menú principal, se mostrarán solo aquellas bases que tengan activado el identificador "Públicas", de lo contrario aparecerán las que pertenezcan al usuario.

Escenario Levenshtein

✓ *Página Estática*

- **Nombre:** Levenshtein
- **Entrada:** Proviene de la siguiente página
 - ❖ Menú Principal (Página Estática)
- **Propiedades:**
 - **Caja de texto**

Nombre	Función
Término	Permite teclear el Término para el proceso de Levenshtein.
Definición1	Permite teclear la primera definición para el término de Levenshtein.
Definición2	Permite teclear la segunda definición para el término de Levenshtein.
Peso por inserción	Permite teclear el valor del peso por Inserción.
Peso por Borrado	Permite teclear el valor del peso por Borrado.
Peso por Sustitución	Permite teclear el valor del peso por Sustitución.
Peso por Igualdad	Permite teclear el valor del peso por Igualdad.

▪ **Casilla de verificación**

Nombre	Función
Usar Lematizador	Habilita el Lematizador
Vector de Palabras	Habilita el vector de palabras
Tabla de Costos	Habilita la tabla de costos
Alineamiento del Algoritmo Original	Habilita el alineamiento original
Alineamientos posibles	Muestra todos los alineamientos
Obtener Bindings	Habilita los Bindings

▪ **Casilla de Opción para Idioma**

Nombre	Función
Inglés	El proceso de Levenshtein utilizará como idioma el Inglés
Español	El proceso de Levenshtein utilizará como idioma el Español

▪ **Botón**

Nombre	Asociado	Función
Restablecer	A un Escenario llamado: Levenshtein.	Permite limpiar los datos de la página.
Regresar	A un Escenario llamado: Levenshtein.	Cierra la pagina de Levenshtein y abre la de Menú Principal.

Ejecutar	A un Escenario llamado: Levenshtein.	Invoca un Script llamado: Proceso para la Respuesta de Levenshtein.
----------	--------------------------------------	---

✓ *Scripts*

- **Nombre:** Proceso para la Respuesta de Levenshtein
- **Descripción:** Genera los datos de la respuesta de Levenshtein, los cuales son presentados en una página dinámica llamada: Respuesta de Levenshtein.
- **Parámetros que recibe:** Información que teclea el usuario en las opciones dentro de la página de Levenshtein.

Respuesta de Levenshtein

✓ *Página Dinámica*

- **Nombre:** Respuesta de Levenshtein
- **Entrada:** Proviene del "Proceso para la Respuesta de Levenshtein"
- **Información:**
 - **Liga**

Nombre	Función
Levenshtein	Permite regresar al escenario que tiene por nombre Levenshtein

- **Tabla**

Nombre	Función
Presentación de Resultados	Mostrar los resultados del proceso de Levenshtein de acuerdo a lo que el usuario tecleo en su formulario.

- **Botón**

Nombre	Función
Anterior	Permite seguir la secuencia de la información de los resultados de Levenshtein. El botón esta en función de la cantidad de resultados que se presenten.
Siguiente	Permite regresar a la secuencia anterior de la información de los resultados de Levenshtein. El botón esta en función de la cantidad de resultados que se presenten.

TESIS CON
 FALLA DE ORIGEN

Escenario Clustering

✓ *Página Dinámica*

- ◆ **Nombre:** Clustering
- ◆ **Entrada:** Proviene de la siguiente página
 - ❖ Menú Principal (Página Estática)
 - ❖ Menú(o) (Página Estática)
 - ❖ Menú(u) (Página Estática)

- ◆ **Información:**

- **Catálogo**

Nombre	Función
Base	Permite seleccionar la base con la que se va a trabajar

- **Caja de texto**

Nombre	Función
Num. Iteraciones	Permite teclear el Numero de Iteraciones

- **Casilla de Información**

Nombre	Función
Lematizador	Habilita el lematizador
Mayúsculas y Minúsculas	Habilita las mayúsculas y Minúsculas
Peso por inserción	Habilita peso por inserción
Peso por borrado	Habilita peso por borrado
Peso por sustitución	Habilita peso por sustitución
Peso por igualdad	Habilita Peso por igualdad
P. sustitución	Habilita palabras de sustitución
P. Stop	Habilita palabras de Stop
Máxima Profundidad	Solicita resultados con máxima profundidad
Swaps	Habilita swaps
Semi-nulos	Habilita Semi-nulos
Mínimo valor LCC	Habilita mínimo valor LCC
Muestra Lematización	Solicita resultados mostrando lematización
Muestra Resultados Parciales	Muestra resultados parciales

- **Casilla de Opción**

Nombre	Función
Toda la información	Muestra toda la información
Exclusivamente Bindings	Muestra la Información de Bindings
Solo generen Bindings	Muestra los que generen Bindings
Bindings y Clusters	Muestra Bindings y Clusters
Solamente Clusters	Muestra solamente Clusters

▪ **Botón**

Nombre	Asociado	Función
Regresar	A un Escenario llamado: Clustering.	Cierra la pagina de Clustering y abre la de Menú Principal, Menú(o) ó Menu(u).
Ejecutar	A un Escenario llamado: Clustering.	Invoca un Script llamado: Proceso para la Respuesta de Clustering.

✓ **Scripts**

- **Nombre:** Proceso para la Respuesta de Clustering
- **Información:** Genera los datos de la respuesta de Clustering, los cuales son presentados en una página dinámica llamada: Respuesta de Clustering ya sea por una o varias ejecuciones.
- **Parámetros que recibe:** Información que teclea el usuario en las opciones de la página "Clustering"

Respuesta de Clustering

✓ **Página Dinámica**

- **Nombre:** Respuesta de Clustering
- **Entrada:** Proviene de "Proceso para la Respuesta de Clustering "
- **Propiedades:**

▪ **Liga**

Nombre	Función
Clustering	Permite regresar al escenario que tiene por nombre Clustering, correspondiente al login y password que tecleó el usuario.

▪ **Tabla**

Nombre	Función
Presentación de Resultados	Mostrar los resultados del proceso de Clustering de acuerdo a lo que el usuario tecleo en su formulario.

▪ **Botón**

Nombre	Función
Anterior	Permite seguir la secuencia de la paginación de los resultados de Clustering. El botón esta en función de la cantidad de resultados que se presenten en la respuesta de varias ejecuciones.

Siguiente	Permite regresar a la secuencia anterior de la paginación de los resultados de Clustering. El botón esta en función de la cantidad de resultados que se presenten en la respuesta de varias ejecuciones.
-----------	--

Escenario Base de Datos

✓ *Página Estática*

- **Nombre:** Base de Datos
- **Entrada:** Proviene de la siguiente página
 - ❖ Menú Principal (Pagina. Estática)
- **Propiedades:**
 - **Botón**

Nombre	Asociado	Función
Bases públicas	A un Escenario llamado: Base de Datos.	Abre la página dinámica Bases Públicas para ejecutar otros procesos utilizando las bases públicas del sistema.
Login y passwd	A un Escenario llamado: Base de Datos.	Abre la pagina estática Login y Passwd, para que el usuario pueda consultar y actualizar las bases que tiene asignadas.
Solicitud de Base	A un Escenario llamado: Base de Datos.	Abre la página estática de Solicitud de Base.
Regresar	A un Escenario llamado: Base de Datos.	Abre la página estática de Menú principal.

Escenario Bases Públicas

✓ *Página Dinámica*

- **Nombre:** Bases Públicas
- **Entrada:** Proviene de la página
 - ❖ Escenario Base de Datos (Página Estática)
- **Propiedades:**
 - **Frame**

Nombre	Función
Bases Públicas	Permite interactuar con las páginas dinámicas de nombre: <u>Tablas de Base(Frame1)</u> , <u>Catálogo (Frame2)</u> y <u>Datos (Frame3)</u> .

Tablas de Base

✓ *Página Dinámica*

- **Nombre:** Tablas de Base
- **Entrada:** Proviene de las páginas
 - ❖ Bases Públicas(Página. Dinámica)
 - ❖ Alimentar Base (Página Dinámica)
- **Propiedades:**

- **Liga**

Definiciones	Invoca un Script llamado: Proceso para Definiciones.
Términos	Invoca un Script llamado: Proceso para Términos.
Fuentes	Invoca un Script llamado: Proceso para Fuentes.
Sinónimos	Invoca un Script llamado: Proceso para Sinónimos.
Área Temática	Invoca un Script llamado: Proceso para Área Temática.

✓ *Scripts*

- **Nombre:** Proceso para Definiciones
- **Descripción:** Se conecta a la base de datos seleccionada en la página dinámica Catálogo, selecciona todos los campos de la tabla definiciones y los presenta en la página dinámica del frame3, logrando con ello que el usuario pueda interactuar con las Definiciones existentes en la Base.
- **Parámetros que recibe:** Información dada por las ligas de Tablas de Base, el nombre de la base dada por el Catálogo Bases y un identificador de la página anterior(Bases Públicas o Alimentar Base).

- **Nombre:** Proceso para Términos
- **Descripción:** Se conecta a la base de datos seleccionada en la página dinámica Catálogo, selecciona todos los campos de la tabla Términos y los presenta en la página dinámica del frame3, logrando con ello que el usuario pueda interactuar con los Términos existentes en la Base de Datos.
- **Parámetros que recibe:** Información dada por las ligas de Tablas de Base, el nombre de la base dada por el Catálogo Bases y un identificador de la página anterior(Bases Públicas o Alimentar Base).

- **Nombre:** Proceso para Fuentes
- **Descripción:** Se conecta a la base de datos seleccionada en la página dinámica Catálogo, selecciona todos los campos de la tabla Fuentes y los presenta en la página dinámica del frame3, logrando con ello que el usuario pueda interactuar con las Fuentes existentes en la Base de Datos.

- **Parámetros que recibe:** Información dada por las ligas de Tablas de Base, el nombre de la base dada por el Catálogo Bases y un identificador de la página anterior(Bases Públicas o Alimentar Base).
- **Nombre:** Proceso para Sinónimos
- **Descripción:** Se conecta a la base de datos seleccionada en la página dinámica Catálogo, selecciona todos los campos de la tabla Sinónimos y los presenta en la página dinámica del frame3, logrando con ello que el usuario pueda interactuar con los Sinónimos existentes en la Base de Datos.
- **Parámetros que recibe:** Información dada por las ligas de Tablas de Base, el nombre de la base dada por el Catálogo Bases y un identificador de la página anterior(Bases Públicas o Alimentar Base).
- **Nombre:** Proceso para Área Temática
- **Descripción:** Se conecta a la base de datos seleccionada en la página dinámica Catálogo, selecciona todos los campos de la tabla Área temática y los presenta en la página dinámica del frame3, logrando con ello que el usuario pueda interactuar con las Áreas existentes en la Base de Datos.
- **Parámetros que recibe:** Información dada por las ligas de Tablas de Base, el nombre de la base dada por el Catálogo Bases y un identificador de la página anterior(Bases Públicas o Alimentar Base).

Catálogo

✓ *Página Dinámica*

- **Nombre:** Catalogo
- **Entrada:** Proviene de la página
 - ❖ Bases Públicas(Pagina Dinámica)
 - ❖ Administrar Base (Página Dinámica)
 - ❖ Alimentar Base(o) (Página Dinámica)
 - ❖ Alimentar Base (u)(Página Dinámica)
- **Propiedades:**
 - **Catalogo**

Nombre	Función
Bases	Desplegará las Bases a las que puede acceder el usuario.

▪ **Liga (sólo cuando viene de Bases Públicas)**

Nombre	Función
Base de Datos	Cierra la pagina que se encuentra en el frame3 ya sea(Bases públicas, Administrar Base o Alimentar Base(o)ó(u)) y abre el escenario Base de Datos.

(sólo cuando viene de Alimentar Base(o) o (u))

Nombre	Asociado	Función
Buscar	A una página dinámica llamada: Catalogo.	Invoca un Script llamado: Proceso para Buscar.

✓ **Scripts**

- **Nombre:** Proceso para buscar
- **Descripción:** Abre la página Dinámica Página Buscar, en la cual podremos seleccionar en que forma queremos realizar dicha búsqueda
- **Parámetros que recibe:** Información dada por las ligas de Tablas de Base, el nombre de la base dada por el Catálogo Bases y un identificador de la página anterior (Bases Públicas o Alimentar Base).

Datos

✓ **Página Dinámica**

- **Nombre:** Datos
 - **Entrada:** Proviene de la página
 - ❖ Bases Públicas(Página. Dinámica)
 - **Propiedades:**
- **Formulario** ??tendrá que explicarse qué tiene este formulario

Nombre	Función
--------	---------

~~*Pag. 6 de escenarios. Falta Link~~

Escenario Solicitud Para Base

✓ *Página Dinámica*

- **Nombre:** Solicitud para Base
- **Entrada:** Proviene de
 - ❖ Escenario Base de Datos
- **Propiedades:**
 - **Caja de texto**

Nombre	Función
Usuario	Permite teclear el Nombre del Usuario.
Mail	Permite teclear el Mail del Usuario.
Teléfono	Permite teclear el Teléfono de la Base.
Tema Base	Permite teclear el Tema de la Base .
NombreBase	Permite teclear el Nombre de la Base.
Descripción	Permite teclear la descripción del funcionamiento de la Base.

- **Casilla de verificación**

Nombre	Función
Clustering	Permite relacionarse con la aplicación de Clustering.
Op. Metalingüísticos	Permite relacionarse con la aplicación de Op. Metalingüísticos.
Otros	Permite relacionarse con otras aplicaciones en el sistema.

- **Botón**

Nombre	Asociado	Función
Restablecer	A un Escenario llamado: Solicitud para Base.	Permite limpiar los datos del formulario Solicitud par Base .
Regresar	A un Escenario llamado: Solicitud para Base.	Cierra la pagina de Solicitud para Base y abre la de Base de Datos.
Enviar	A un Escenario llamado: Solicitud para Base.	Invoca un Script llamado: Proceso de envió.

✓ *Scripts*

- **Nombre:** Proceso de envió
- **Descripción:** Envía un correo electrónico al Administrador del Sistema, con la solicitud de la Base, y regresa un mensaje si el proceso estuvo bien o una excepción si no.
- **Parámetros que recibe:** Información que teclea el usuario en las cajas de Texto.

Escenario Login y Passwd

✓ *Página Estática*

- **Nombre:** Login y Passwd
- **Entrada:** Proviene de la siguiente página
 - ❖ Base de Datos (Pagina. Estática)
- **Propiedades:**

▪ Caja de texto

<i>Nombre</i>	<i>Función</i>
<i>Login</i>	<i>Permite teclear el Login del Usuario.</i>
<i>Passwd</i>	<i>Permite teclear el Passwd del Usuario.</i>

▪ Botón

<i>Nombre</i>	<i>Asociado</i>	<i>Función</i>
<i>Aceptar</i>	<i>A un Escenario llamado: Login y Passwd.</i>	<i>Invoca un Script llamado: Proceso de Login y passwd.</i>
<i>Cancelar</i>	<i>A un Escenario llamado: Login y Passwd.</i>	<i>Cierra la Página estática de Login y Passwd y abre la página Base de Datos.</i>

✓ *Scripts*

- **Nombre:** Proceso de Login y Passwd
- **Descripción:** Se conecta al sistema para buscar en la tabla "Usuarios" el login y passwd dados por el usuario. De encontrarse en esta tabla, checa si el usuario se encuentra en la Tabla de Administradores para abrir la página Dinámica Administrar Sistema, de no ser así verifica en la tabla UsuarioBase para abrir la página Estática Menú(o), y por último solo abre la página estática de Menú(u), también puede abrir una excepción si los datos son incorrectos.
- **Parámetros que recibe:** Información dada por el usuario en las cajas de texto.

TESIS CON
 FALLA DE ORIGEN

Administrar Sistema

✓ *Página Dinámica*

- **Nombre:** Administrar Sistema
- **Entrada:** Proviene del proceso Login y Passwd
- **Propiedades:**

▪ **Catalogo**

Nombre	Función
Bases	Despliega las Bases existentes en el Sistema.

▪ **Botón**

Nombre	Asociado	Función
Borrar	A un Escenario llamado: Administrar Sistema.	Invoca un Script llamado: "Proceso de Borrado para Administrar Sistema".
Nueva Base	A un Escenario llamado: Administrar Sistema.	Realiza un proceso que abre la página estática Nueva Base.
Recuperar	A un Escenario llamado: Administrar Sistema.	Invoca un Script llamado: "Recuperar".
Obtener respaldo	A un Escenario llamado: Administrar Sistema.	Invoca un Script llamado: "Proceso para Obtener Respaldo".
Nuevo Administrador	A un Escenario llamado: Administrar Sistema.	Muestra la página dinámica <u>Nuevo administrador</u> en la que se puede dar de alta a un nuevo Administrador.
Salir	A un Escenario llamado: Administrar Sistema.	Cierra la Página dinámica de Administrar Sistema y abre la página Login y Passwd.

✓ *Scripts*

- **Nombre:** Proceso de Borrado para Administrar Sistema
- **Descripción:** Ejecuta un proceso que muestra la página dinámica Respuesta de Borrado en la cual el Administrador puede borrar una Base de Datos
- **Parámetros que recibe:** El nombre de la página de la que proviene en este caso "Administrar Sistema" el nombre de la base dado por el Catalogo "Bases", el nombre de la tabla en este caso "Bases" y el elemento a Borrar, en este caso el nombre de la base de Datos dada por el catalogo "Bases".
- **Nombre:** Recuperar
- **Descripción:** Se conecta a la tabla Respaldos para seleccionar los nombres de las Bases que han sido respaldadas y las muestra en la página dinámica Bases respaldadas
- **Nombre:** Proceso para Obtener Respaldo

- **Descripción:** Ejecuta un proceso que generará un respaldo de la Base de Datos, creando un archivo txt que se colocara dentro de la tabla Respaldos. Mostrará también un cuadro de diálogo que indique si el proceso se ejecutó satisfactoriamente o no.
- **Parámetros que recibe:** Información dada por el Catalogo de nombre: "Bases", con el nombre de la base a respaldar.

Bases Respaldadas

✓ ***Página Dinámica***

- **Nombre:** Bases Respaldadas
- **Entrada:** Proviene de la página Administrar Sistema
- **Propiedades:**

▪ **Catalogo**

Nombre	Función
Bases	Despliega el nombre de las bases respaldadas y la fecha de este respaldo.

▪ **Botón**

Nombre	Asociado	Función
Aceptar	A una página dinámica llamada: Bases Respaldadas.	Invoca un Script llamado: "Proceso para Recuperar La Base de Datos".
Cancelar	A una página dinámica llamada: Bases Respaldadas.	Cierra la página <u>Bases Respaldadas</u> y abre la página <u>Administrar Sistema</u> .

✓ ***Scripts***

- **Nombre:** Proceso para Recuperar La Base de Datos
- **Descripción:** Ejecuta un proceso que recupera la base de Datos que fue Modificada y mostrará un cuadro de diálogo que indique si el proceso se ejecutó o no satisfactoriamente.
- **Parámetros que recibe:** Información dada por el Catalogo de nombre: "Bases", con el nombre de la base a respaldar.

Respuesta de Borrado

✓ *Página Dinámica*

- **Nombre:** Respuesta de Borrado
- **Entrada:** Proviene de los siguientes Scripts
 - ❖ Proceso de Borrado para Administrar Sistema
 - ❖ Proceso de Borrado para Administradores
 - ❖ Proceso de Borrado para Usuarios
 - ❖ Proceso de Borrado Registros de la Base
- **Propiedades:**

▪ Cuadro de Diálogo

Nombre	Función
Respuesta de Borrado	Permite cerciorar que dicha información quiere ser borrada.

▪ Botón

Nombre	Asociado	Función
Aceptar	A una página estática llamada: Respuesta de Borrado.	Invoca un Script llamado: Proceso de Borrado.
Cancelar	A una página estática llamada: Respuesta de Borrado.	Invoca un Script llamado: Proceso para cancelar Borrado.

✓ *Scripts*

- **Nombre:** Proceso de Borrado
- **Descripción:** Selecciona del sistema el nombre de la base en "tabla Bases"; además del nombre de la tabla en la que se encuentra el elemento y por último el nombre del elemento. Regresa un mensaje de éxito o error según sea el caso.
- **Parámetros que recibe:** Información dada por los procesos que la mandan llamar, el primer argumento es el nombre de la página de la que proviene, el segundo el nombre de la base de Datos del usuario, el tercero el nombre de la tabla que debe seleccionar y el cuarto y último el nombre del elemento a borrar; puede darse el caso de que el nombre de la base sea nulo, por lo que ese parámetro no se toma en cuenta .
- **Nombre:** Proceso para cancelar Borrado
- **Descripción:** Cierra la página Respuesta de Borrado y abre la página de la que proviene.

- ♦ **Parámetros que recibe:** El nombre de la página de que proviene dado por los procesos que la generan.

Escenario Nueva Base

✓ *Página Estática*

- ♦ **Nombre:** Nueva Base
- ♦ **Entrada:** Proviene de la siguiente página
 - ❖ Administrar Sistema(Página. Dinámica)
- ♦ **Propiedades:**

▪ **Caja de texto**

Nombre	Función
Usuario	Permite teclear el Nombre del Usuario.
Passwd	Permite teclear el Passwd del Usuario.
Base	Permite teclear el Nombre de la Base.
Idioma	Permite teclear el Idioma de la Base .
Descripción	Permite teclear la descripción del funcionamiento de la Base.

▪ **Botón**

Nombre	Asociado	Función
Procesar	A un Escenario llamado: Nueva Base.	Invoca un Script llamado: Proceso para Nueva Base.
Restablecer	A un Escenario llamado: Nueva Base.	Limpia los datos del formulario Nueva Base .
Regresar	A un Escenario llamado: Nueva Base.	Cierra la página Nueva Base y abre la página dinámica de Administrar Sistema.

✓ *Scripts*

- ♦ **Nombre:** Proceso para Nueva Base
- ♦ **Descripción:** Crea una nueva Base, insertando el nombre de ésta en la tabla Bases del sistema adicionando además el nombre del Owner en la tabla usuarios y adicionando su información en la tabla UsuarioBase además regresa un mensaje se éxito o error según sea el caso.
- ♦ **Parámetros que recibe:** Información dada por las cajas de Texto.

Escenario Nuevo Administrador

✓ *Página Dinámica*

- **Nombre:** Nuevo Administrador
- **Entrada:** Proviene de la siguiente página
 - ❖ Administrar Sistema(Pagina. Dinámica)
- **Propiedades:**
 - **Frame**

Nombre	Función
Nuevo Administrador	Permite interactuar con las páginas de nombre: <u>Administradores</u> (Frame1) en las que podemos consultar los Administradores del Sistema y así mismo borrarlos si así se requiere; <u>Alta Nuevo Admin.</u> (Frame3) donde se proporcionan los datos del nuevo administrador.

Página Administradores

✓ *Página Estática*

- **Nombre:** Administradores
- **Entrada:** Proviene de la página
 - ❖ Nuevo Administrador(Pagina. Dinámica)
- **Propiedades:**
 - **Catálogo de Página Administradores**

Nombre	Función
Administradores	Desplegará los Administradores existentes en el sistema.

- **Botón**

Nombre	Asociado	Función
Borrar	A una página llamada: Administradores.	Invoca un Script llamado: "Proceso de Borrado para Administradores".

✓ *Scripts*

- **Nombre:** Proceso de Borrado para Administradores
- **Descripción:** Ejecuta un proceso que muestra la página dinámica Respuesta de Borrado en la cual el Administrador puede borrar un Administrador Existente.
- **Parámetros que recibe:** El nombre de la página de la que proviene en este caso "Administradores", el nombre de la tabla en este caso "Administradores" y el elemento

a Borrar, en este caso el nombre del Administrador dada por el catalogo "Administradores".

Alta Nuevo Administrador

✓ *Página Estática*

- ♦ **Nombre:** Alta Nuevo Admin.
- ♦ **Entrada:** Proviene de la página
 - ❖ Nuevo Administrador(Pagina. Dinámica)
- ♦ **Propiedades:**
 - **Caja de texto**

Nombre	Función
Nombre del Usuario	Permite teclear el Nombre del Usuario.
Login	Permite teclear el Login del Usuario.
Password	Permite teclear el Passwd del Usuario.
Dirección	Permite teclear la Dirección del Usuario .
Teléfono	Permite teclear el Teléfono del Usuario.
Email	Permite teclear la dirección electrónica del Usuario.
Dependencia	Permite teclear la Dependencia a la que pertenece el Usuario.

- **Botón**

Nombre	Asociado	Función
Restablecer	A un Formulario llamado: Nuevo Administrador.	Permite limpiar los datos de la caja de Texto.
Regresar	A un Formulario llamado: Nuevo Administrador.	Abre la página dinámica Administrar Sistema.
Ejecutar	A un Formulario llamado: Nuevo Administrador.	Invoca un Script llamado: Proceso para Nuevo Administrador.

✓ *Scripts*

- ♦ **Nombre:** Proceso para Nuevo Administrador
- ♦ **Descripción:** Inserta un Nuevo Administrador en el sistema, y adiciona en la tabla de Usuarios todos los datos relacionados con él.
- ♦ **Parámetros que recibe:** Información dada por las Cajas de Texto

Página del Menú (o)

✓ *Página Estática*

- **Nombre:** Pág. del Menú (o)
- **Entrada:** Proviene del "Proceso Login-Passwd"
- **Propiedades:**
 - **Botón**

Nombre	Asociado	Función
Clustering	A la Pág. Menú (o)	Invoca un Script llamado "Proceso de Clustering".
Administrar Base	A la Pág. Menú (o)	Invoca un Script llamado "Proceso de Administrar Base (o)".
Regresar	A la Pág. Menú (o)	Regresa al Escenario Login-Paswwd cerrando la pág. del Menú(o)

✓ *Scripts*

- El script correspondiente a Clustering y el Escenario de Clustering, son explicados anteriormente.
- **Nombre:** Proceso de Administrar Base
- **Descripción:** Ejecuta un proceso que realiza una conexión al sistema, la cuál muestra las bases disponibles por medio de la página dinámica Catálogo en el Frame1. Dentro del Frame2 hace una conexión a la página dinámica Catálogo Usuarios, los muestra y permite borrarlos, en la página dinámica Permisos del Frame3 despliega un cuadro de datos donde muestra los permisos que tiene cada usuario para que a su vez se puedan modificar dichos permisos.
- **Parámetros que recibe:** La información que recibe es la selección que se hace de la base y el usuario.

Escenario Administrar Base (o)

✓ *Página Dinámica*

- **Nombre:** Administrar Base (o)
- **Entrada:** Proviene de la siguiente página
 - ❖ Página del Menú (o)
- **Propiedades:**
 - **Frame**

Nombre	Función
Administrar	Permite interactuar con tres páginas que son: <u>Catálogo</u> (Frame1),

Base	<u>Catálogo Usuarios</u> (Frame2) y <u>Permisos</u> en las cuales podemos Administrar los permisos de la Base seleccionada y otorgar y/o modificar dichos permisos. Además se puede seleccionar un usuario y darlo de baja.
------	---

* La página Catálogo esta previamente definida en Escenario Bases Públicas

Página Catálogo Usuarios

✓ *Página Dinámica*

- **Nombre:** Catálogo Usuarios
- **Entrada:** Proviene de la página
 - ❖ Administrar Base(Pagina. Dinámica)
- **Propiedades:**
 - Catálogo

Nombre	Función
Usuarios	Despliega los nombres de los usuarios de las bases

- **Botón**

Nombre	Asociado	Función
Regresar	A un Escenario llamado: Administrar Base.	Cierra la pagina de Administrar Base y abre la de página de menú (o).
Borrar	A un Escenario llamado: Administrar Base.	Invoca un Script llamado: "Proceso de Borrado para Usuarios".

✓ *Scripts*

- **Nombre:** Proceso de Borrado para Usuarios
- **Descripción:** Ejecuta un proceso que muestra la página dinámica Respuesta de Borrado en la cual el Owner puede borrar un Usuario Existente.
- **Parámetros que recibe:** El nombre de la página de la que proviene en este caso Catálogo Usuarios , el nombre de la base dado por la página (Catalogo), el nombre de la tabla en este caso "Usuarios" y el nombre del elemento a Borrar, en este caso el nombre del Usuario dada por el Catalogo "Usuarios".

Página Permisos

✓ *Página Dinámica*

- ◆ **Nombre:** Permisos
- ◆ **Entrada:** Proviene de la página
 - ❖ Administrar Base(Página. Dinámica)
- ◆ **Propiedades:**
 - **Formulario**

Nombre	Función
Administrar Base	Permite modificar los permisos de las tablas de términos, fuentes, área temática, sinónimos, p. stop y p. sustitución

- **Botón**

Nombre	Asociado	Función
Modificar Permisos	A un Escenario llamado: Administrar Base.	Permite modificar los permisos del formulario
Dar de Alta Nuevo Usuario	A un Escenario llamado: Administrar Base.	Cierra el escenario Administrar Base y ejecuta un Script llamado "Nuevo Usuario".

✓ *Scripts*

- ◆ **Nombre:** Nuevo Usuario
- ◆ **Descripción:** Ejecuta un proceso que realiza una conexión al sistema la cual muestra las bases disponibles para el Owner por medio de un catálogo.
- ◆ **Parámetros que recibe:** La información que recibe es la selección que se hace de la tabla bases.

Escenario Alta de Nuevo Usuario

✓ *Página Dinámica*

- ◆ **Nombre:** Alta de Nuevo Usuario
- ◆ **Entrada:** Proviene de la siguiente página
 - ❖ Administrar Base
- ◆ **Propiedades:**
 - **Catálogo**

Nombre	Función
Base	Permite seleccionar la base con la que se va a trabajar

- **Caja de texto**

Nombre	Función
Usuario	Permite teclear el Nombre del nuevo usuario
Login	Permite teclear el login del nuevo usuario
Password	Permite teclear el password del nuevo usuario
Dirección	Permite teclear la dirección de el nuevo usuario
Teléfono	Permite teclear el teléfono de el nuevo usuario
E-mail	Permite teclear el E-mail de el nuevo usuario
Dependencia	Permite teclear la Dependencia de origen de el nuevo usuario

- **Formulario**

Nombre	Función
Tabla de permisos	Permite asignar los permisos de el nuevo usuario sobre las tablas de términos, fuentes, área temática, sinónimos, p. stop y p. Sustitución

- **Botón**

Nombre	Asociado	Función
Restablecer	A un Escenario llamado: Alta de Nuevo Usuario.	Permite limpiar los datos tecleados en la caja de texto .
Regresar	A un Escenario llamado: Alta de Nuevo Usuario.	Cierra la pagina de Alta de Nuevo Usuario y abre la de Administrar Base.
Enviar	A un Escenario llamado: Alta de Nuevo Usuario.	Invoca un Script llamado: Enviar

- ✓ **Scripts**

- **Nombre:** Enviar
- **Descripción:** Ejecuta un proceso que envía los datos del nuevo usuario a la Base de Datos y regresa un mensaje si el proceso tuvo una excepción .
- **Parámetros que recibe:** Información que teclea el usuario para dar de alta al nuevo usuario.

Escenario Alimentar Base

- ✓ **Página Dinámica**

- **Nombre:** Alimentar Base
- **Entrada:** Proviene de la siguiente página
 - ❖ Página del Menú (o)
 - ❖ Página del Menú (u)
- **Propiedades:**

- **Frame**

Nombre	Función
Alimentar Base	Permite interactuar con tres páginas que son: <u>Tablas de Base (Frame1)</u> , <u>Catálogo (Frame2)</u> y <u>Alimentar (Frame3)</u> en las cuales podemos Alimentar la Base seleccionada, elegir las tablas que se quieren consultar y a su vez borrar, insertar, buscar o modificar datos de dicha base.

* Las páginas Tablas de Base y Catálogo, están previamente definidas en el Escenario Bases Públicas así como los scripts respectivos.

Página Alimentar

- ✓ **Página Dinámica**

- **Nombre:** Alimentar
- **Entrada:** Proviene de la página
 - ❖ Alimentar Base (o)(Página. Dinámica)
 - ❖ Alimentar Base (u)(Página. Dinámica)
- **Propiedades:**

- **Formulario**

Nombre	Función
Alimentar Base	Muestra la información de la liga seleccionada en la página <u>Tablas de Base</u> y permite modificar y/o consultar información.

- **Menú Desplegable**

Nombre	Función
Términos Disponibles	Permite seleccionar un término, correspondiente a la base y liga elegidas.

- **Botón**

Nombre	Asociado	Función
Insertar	A un Escenario llamado: Alimentar Base.	Reemplaza la información presentada en el frame3 por una página dinámica llamada: <u>Página insertar</u> la cual nos permite insertar nuevos términos.
Modificar	A un Escenario llamado: Alimentar Base.	Reemplaza la información presentada en el frame3 por una página dinámica llamada: <u>Página Modificar</u> la cual nos permite cambiar términos existentes.
Borrar	A un Escenario llamado: Alimentar Base.	Invoca un Script llamado: "Proceso de Borrado para Alimentar".
Regresar	A un Escenario	Cierra la página de Alimentar Base y abre la de página

llamado: Alimentar Base.	de menú (o).
--------------------------	--------------

✓ **Scripts**

- ♦ **Nombre:** Proceso de Borrado para Alimentar
- ♦ **Descripción:** Ejecuta un proceso que muestra la página dinámica Respuesta de Borrado en la cual el Usuario puede borrar un Registro de la Base.
- ♦ **Parámetros que recibe:** El nombre de la página dado por(página Alimentar) , el nombre de la base (Catalogo), el nombre de la tabla(página tablas de Base)y el nombre del elemento a Borrar, en este caso el nombre del Registro dado por el menú desplegable dentro de la página Alimentar .

Página Buscar

✓ **Página Dinámica**

- ♦ **Nombre:** *Página Buscar*
- ♦ **Entrada:** Proviene de la siguiente página
 - ❖ Alimentar (Página Dinámica)
- ♦ **Propiedades:**

- **Cuadro de Diálogo**

Nombre	Función
Buscar	Permite indicar lo que se quiere buscar ya sea por palabra completa o por letras.

- **Botón**

Nombre	Asociado	Función
Buscar ahora	Al cuadro de diálogo de la página buscar	Invoca un Script llamado: "Proceso de Búsqueda"
Cancelar	Al cuadro de diálogo de la página buscar	Cierra la página Buscar y abre la página Alimentar Base.

✓ **Scripts**

- ♦ **Nombre:** Proceso de Búsqueda
- ♦ **Descripción:** Ejecuta la búsqueda de palabras, ya sea por letras o por palabra completa y muestra los resultados en la página dinámica Respuesta de Búsqueda reemplazando dicha información en el frame3. En caso de ocurrir una excepción se muestra en un cuadro de diálogo que es generado por el script.
- ♦ **Parámetros que recibe:** La información que recibe es la palabra que teclea el usuario para que sea buscada.

Respuesta de Búsqueda del Escenario Alimentar Base

✓ *Página Dinámica*

- **Nombre:** Respuesta de Búsqueda
- **Entrada:** Proviene de la siguiente página
 - ❖ *Página Buscar*
- **Propiedades:**

▪ **Formulario**

Nombre	Función
Resultados de la búsqueda	Muestra los resultados de la búsqueda en la <u>Página Alimentar</u> y la información relacionada con la misma además permite modificar y/o consultar información.

- *Los Botones de Insertar, Modificar, Borrar y Regresar correspondientes a la Respuesta de búsqueda, se presentan en la Página Alimentar y están previamente definidos en dicha página.*

Página Insertar

✓ *Página Dinámica*

- **Nombre:** Página Insertar
- **Entrada:** Proviene de la siguiente página
 - ❖ *Alimentar (Página Dinámica)*
- **Propiedades:**

▪ **Formulario**

Nombre	Función
Introduzca el término	Permite introducir información con respecto al nuevo término que se quiere agregar.

▪ **Botón**

Nombre	Asociado	Función
Aceptar	A la Página Insertar	Invoca el Script llamado: "Proceso de inserción"
Limpiar	A la Página Insertar	Limpia los campos de término y cruce
Regresar	A la Página Insertar	Cierra la Página Insertar y abre el escenario Alimentar Base

✓ **Scripts**

- **Nombre:** Proceso de inserción
- **Descripción:** Ejecuta un proceso que realiza una conexión a las tablas de términos, definiciones, fuentes, sinónimos y área temática, donde la información proporcionada se agrega a dichas tablas, en caso contrario generará una excepción que indique si el proceso se ejecutó o no satisfactoriamente por medio de un cuadro de diálogo.
- **Parámetros que recibe:** Información que teclea el usuario al dar de alta un nuevo término.

Página Modificar del Escenario Alimentar Base✓ **Página Dinámica**

- **Nombre:** Página Modificar
- **Entrada:** Proviene de la siguiente página
 - ❖ Alimentar (Página Dinámica)
- **Propiedades:**
 - **Formulario**

Nombre	Función
Modifique término	el Permite modificar cierta información con respecto a los términos que existen en la base.

▪ **Botón**

Nombre	Asociado	Función
Aceptar	A la Página Modificar	Invoca un Script llamado: " Proceso de Modificado"
Limpiar	A la Página Modificar	Limpia los campos de término y cruce
Regresar	A la Página Modificar	Cierra la Página Modificar y abre el escenario Alimentar Base

✓ **Scripts**

- **Nombre:** Proceso de Modificado
- **Descripción:** Ejecuta un proceso que realiza una conexión a las tablas de términos, definiciones, fuentes, sinónimos y área temática, donde la información proporcionada se modifica en dichas tablas en caso contrario generará una excepción que indique si el proceso se ejecutó o no satisfactoriamente por medio de un cuadro de diálogo.
- **Parámetros que recibe:** Información que teclea el usuario al dar de alta un nuevo término.

Página del Menú (u)

✓ *Página Estática*

- **Nombre:** Pág. del Menú (u)
- **Entrada:** Proviene del "Proceso Login-Passwd"
- **Propiedades:**

▪ **Botón**

Nombre	Asociado	Función
Clustering	A la Pág. Menú (u)	Invoca un Script llamado "Proceso de Clustering".
Alimentar Base	A la Pág. Menú (u)	Invoca un Script llamado "Proceso de Alimentar Base(u)".
Regresar	A la Pág. Menú (u)	Regresa al Escenario Login-Paswwd cerrando la pág. del Menú(u)

* *Los procesos de Clustering y Alimentar base así como sus páginas respectivas están previamente definidas.*