



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**SISTEMA DE AUDITORÍA INFORMÁTICA Y
CONTROL DE CENTROS DE TECNOLOGÍA DE
INFORMACIÓN EN SOFTWARE LIBRE**

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO EN COMPUTACIÓN

P R E S E N T A N:

**ROLANDO LIMA MACIEL
JÓSE ÁNGEL HARO JUÁREZ**

DIRECTOR DE TESIS: ING. HERIBERTO OLGUÍN ROMO

MÉXICO, D.F.

2008



AGRADECIMIENTOS

Dedico esta Tesis:

Con un profundo agradecimiento pero sobre todo amor a mis padres: Silvia Maciel Sierra y Darío Lima Moreno porque gracias a ellos he podido concluir este proyecto y llegar a una meta crucial en mi vida, a través de sus consejos y los valores que me han inculcado he podido lograr la disciplina y el compromiso para subir cada escalón necesario para la culminación de esta etapa de mi vida académica. A ti papá gracias por enseñarme a trabajar e inducirme el sentido de la responsabilidad desde muy pequeño. A ti mamá por toda la comprensión y el apoyo que siempre me has brindado y estar siempre pendiente de mi a lo largo de mi trayectoria escolar. A los dos por mantener unida y a flote nuestra familia, además de darnos todo lo que hemos necesitado tanto a mis hermanos como a mi.

A mis hermanos Silvia y Enrique por soportar las molestias que les pude haber causado como consecuencia de la culminación de este trabajo. A Silvia en especial por tomarse el tiempo para revisar la tesis escrita y ayudarme a mejorar la presentación de la misma.

A mis abuelos maternos: Lola y Joaquín, donde quiera que se encuentren, así como a mi abuela paterna: Joaquina. Por ser mis guías a través de sus consejos.

A mi tía Lupe y mis primos Fabiola y Juan, por formar parte de mi vida y ser prácticamente mi segunda familia. A Fabiola muchas gracias por su apoyo en la revisión ortográfica de la tesis escrita.

A Liz por acompañarme en el camino final hacia esta meta, apoyándome incondicionalmente, llenando de luz mi vida y devolviéndome la confianza en los momentos difíciles, por ayudarme a levantar cuando he caído, he impulsándome a lograr nuevos objetivos y enseñarme a ver nuevos horizontes. Te amo.

A mis tías, tíos y primos que crecieron conmigo y me brindaron su apoyo en todo momento.

A todos mis compañeros con los compartí un salón de clases, pero sobre todo a aquellos que me enseñaron que los conocimientos más trascendentales que se obtienen en la facultad se aprenden fuera de las aulas y me ayudaron a disfrutar mi vida académica. En especial a dos organizaciones estudiantiles que me marcaron gratamente de por vida, primero a HPC por introducirme a la filosofía con la que logré sacar adelante mi vida estudiantil pero sobre todo porque las personas que forman la organización se convirtieron en más que sólo compañeros de carrera, en una verdadera familia para mi. Por otro lado no puedo dejar de mencionar al PROTECO el cual me permitió tener un crecimiento exponencial no sólo académicamente sino como persona.

Al Ing. Heriberto Olguín Romo, por ser el director de la tesis y por todas las enseñanzas que me proporcionó durante todos estos años.

A mis compañeros de trabajo de quienes he recibido un apoyo extraordinario, en especial a mis Jefes lam Gerardo Moreno Virgen y Daniel Quezada Alvarado por todas las facilidades que me otorgaron para cumplir este objetivo. A Cinthia y Sergio les agradezco por invitarme a formar parte de este gran equipo.

A mí amada UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO y mi querida FACULTAD DE INGENIERÍA que me permitieron formarme como Ingeniero.

Rolando Lima Maciel

Mayo, 2008

Para mis padres ...

Mis guías y héroes.

Mis mas sinceros agradecimientos a mis padres, que han estado desde el principio y sin su esfuerzo y sacrificios no habría llegado hasta el final, a ti mamá que me cuidaste y me ayudaste en todo lo que estuvo a tu alcance y a ti papá que ahora te entiendo un poco mas y agradezco tus regaños y exigencias ya que gracias a estos logre terminar la carrera.

A ti Lucero que creciste conmigo y en la medida de tus posibilidades pusiste tu granito de arena para ayudarme.

A ti Lancho que fuiste abuelo y abuela y siempre estuviste con nosotros consintiéndonos.

A todos mis amigos que a pesar del tiempo aun siguen aquí, en especial a los ermitaños del CCH y a todos los Hp de la facultad con los que compartí salón de clase y seguimos adelante en nuestro paso por la carrera, especialmente a Rolando que me ayudo incondicionalmente a lo largo de la carrera y sin el no habría sido posible terminar este proyecto gracias.

A mi esposa e hijo que fueron y son mi fuerza para seguir adelante, gracias Mari por haber sido testigo desde hace 10 años de mi crecimiento y paso por la Universidad los Amo.

Expreso también mis agradecimientos a la Universidad Nacional Autónoma de México, a la Facultad de Ingeniería por ser el semillero de los mejores Ingenieros del país y especialmente al Programa de Tecnología en Cómputo (Proteco) el cual fue de gran importancia en mi paso por la Facultad de Ingeniería por todo lo que aprendí y me a sido de gran ayuda en el ámbito laboral.

Y final mente al Ingeniero Heriberto Olguín Romo, por haber dirigido esta tesis y por sus consejos y apoyo en mi paso por el Proteco.

Jóse Ángel Haro Juárez

Mayo, 2008

ÍNDICE

ÍNDICE

ÍNDICE	I
INTRODUCCIÓN.....	1
CAPÍTULO 1: SOFTWARE LIBRE	5
1.1 HISTORIA Y ANTECEDENTES DEL SOFTWARE LIBRE.....	7
1.2 DEFINICIÓN DE SOFTWARE LIBRE.....	8
1.3 PRIMEROS INDICIOS DEL SOFTWARE LIBRE.....	9
1.4 GNU Y RICHARD STALLMAN (NACE EL SOFTWARE LIBRE)	10
1.4.1 GNU	10
1.4.1.1 GNU/LINUX.....	11
1.5 LA FREE SOFTWARE FOUNDATION (FSF)	12
1.6 LA LICENCIA GPL.....	12
1.6.1 TÉRMINOS Y CONDICIONES PARA LA COPIA, DISTRIBUCIÓN Y MODIFICACIÓN DE UN PROGRAMA EN SOFTWARE LIBRE	12
CAPÍTULO 2: SOFTWARE PROPIETARIO	17
2.1 ANTECEDENTES PRELIMINARES SOBRE EL SOFTWARE PROPIETARIO	19
2.2 PRIMEROS IMPULSORES DEL SOFTWARE PROPIETARIO	19
2.2.1 BILL GATES	20
2.2.3 PAUL ALLEN.....	20
2.2.4 STEVE JOBS	21
2.2.5 STEVE WOZNIAK.....	21
2.3 EL GIGANTE MICROSOFT	21
2.4 APPLE	24
2.5 TIPOS DE LICENCIAS PARA SOFTWARE PROPIETARIO.....	25
2.5.1 EULA (ACUERDO DE LICENCIAS DE USUARIO FINAL)	25
2.5.2 CLUF (CONTRATO DE LICENCIAS DE USUARIO FINAL)	26
2.5.3 APSL (LICENCIA DE CÓDIGO FUENTE PÚBLICO DE APPLE)	26
2.5.3.1 FALTA DE RESPETO A LA PRIVACIDAD.....	26
2.5.3.1.1 CONTROL CENTRAL.....	26
2.5.3.2 POSIBILIDAD DE REVOCARLA EN CUALQUIER MOMENTO	27
2.6 LICENCIA DE CÓDIGO BINARIO (JAVA).....	27
CAPÍTULO 3: SOFTWARE LIBRE VS SOFTWARE PROPIETARIO	31
3.1 EL PASADO, SOFTWARE PROPIETARIO	33
3.2 VENTAJAS DEL SOFTWARE LIBRE	33
3.3 DESVENTAJAS DEL SOFTWARE LIBRE	36
3.4 VENTAJAS DEL SOFTWARE PROPIETARIO.....	37
3.4.1 VENTAJAS DEL SOFTWARE PROPIETARIO EN GENERAL	38
3.4.1.1 VENTAJAS DEL SOFTWARE MICROSOFT.....	38
3.4.2 DESVENTAJAS DEL SOFTWARE PROPIETARIO EN GENERAL	39
3.4.2.1 DESVENTAJAS DEL SOFTWARE DE MICROSOFT.....	40

CAPÍTULO 4: REINGENIERÍA DE SOFTWARE.....	43
4.1 ORIGEN DE REINGENIERÍA.....	45
4.2 DEFINICIÓN DE REINGENIERÍA	46
4.3 REINGENIERÍA DE SOFTWARE.....	46
4.4 REINGENIERÍA UN PROBLEMA DE INGENIERÍA.....	48
4.4.1 SOFTWARE ENGINEERING INSTITUTE (SEI)	49
4.4.1.1 MODELO DE CAPACIDAD Y MADUREZ.....	49
4.4.2 ETAPAS DE LA REINGENIERÍA	51
4.5 TIPOS DE REINGENIERÍA.....	53
4.5.1 INGENIERÍA INVERSA.....	53
4.5.2 REESTRUCTURACIÓN DE CÓDIGO	53
4.5.3 REESTRUCTURACIÓN DE DATOS	53
4.5.4 INGENIERÍA DIRECTA.....	54
4.6 HERRAMIENTAS DE REINGENIERÍA	54
4.6.1 HERRAMIENTAS CASE.....	55
CAPÍTULO 5: SAICOCETI EN PLATAFORMA MICROSOFT	57
5.1 LENGUAJE DE PROGRAMACIÓN	59
5.1.1 ANTECEDENTES HISTÓRICOS DE VISUAL BASIC.....	59
5.1.2 CARACTERÍSTICAS GENERALES DE VISUAL BASIC	61
5.1.3 VENTAJAS Y DESVENTAJAS DE VISUAL BASIC	61
5.1.4 USO DE VISUAL BASIC EN EL SAICOCETI	62
5.1.5 USO DE ASP EN EL SAICOCETI	63
5.1.5.1 CARACTERÍSTICAS GENERALES DE ASP.....	64
5.1.5.2 VENTAJAS Y DESVENTAJAS DE ASP	64
5.1.5.3 USO DE ASP EN EL SAICOCETI	64
5.2 MANEJADOR DE BASE DE DATOS	64
5.2.1 ANTECEDENTES HISTÓRICOS DE SQL SERVER	65
5.2.2 CARACTERÍSTICAS GENERALES DE SQL SERVER	66
5.2.3 VENTAJAS Y DESVENTAJAS DE SQL SERVER.....	66
5.2.4 USO DE SQL SERVER EN EL SAICOCETI.....	67
5.3 SERVIDOR WEB	67
5.3.1 ANTECEDENTES HISTÓRICOS DE IIS	67
5.3.2 CARACTERÍSTICAS GENERALES DE IIS	67
5.3.3 VENTAJAS Y DESVENTAJAS DE IIS	68
5.2.4 USO DE IIS EN EL SAICOCETI.....	69
5.4 ANÁLISIS GLOBAL DEL SAICOCETI.....	69
CAPÍTULO 6: SAICOCETI EN SOFTWARE LIBRE	71
6.1 LENGUAJE DE PROGRAMACIÓN	73
6.1.1 ANTECEDENTES HISTÓRICOS DE JAVA.....	73
6.1.2 CARACTERÍSTICAS GENERALES DE JAVA.....	74
6.1.3 MÁQUINA VIRTUAL DE JAVA.....	76
6.1.3.1 MOTOR DE EJECUCIÓN	78
6.1.3.2 VERIFICADOR DE JAVA.....	79
6.1.3.3 SOPORTE PARA MÉTODOS NATIVOS.....	79
6.1.3.4 CARGADOR DE CLASES	79

6.1.4	JAVA ENTERPRICE EDITION (JEE)	79
6.2	MANEJADOR DE BASE DE DATOS	80
6.2.1	ANTECEDENTES HISTÓRICOS DE POSTGRESQL.....	80
6.2.2	CARACTERÍSTICAS GENERALES DE POSTGRESQL.....	82
6.3	SERVIDOR WEB	82
6.3.1	ANTECEDENTES HISTÓRICOS DE TOMCAT	82
6.3.2	CARACTERÍSTICAS GENERALES DE TOMCAT	84
6.4	OTROS ELEMENTOS DEL SISTEMA	84
6.4.1	FRAMEWORKS	84
6.4.2	POOL DE CONEXIONES.....	86
CAPÍTULO 7: MANUAL TÉCNICO		87
7.1	ESTUDIO DE FACTIBILIDAD.....	89
7.2	CALENDARIO DE TRABAJO.....	90
7.3	RELACIÓN DE DOCUMENTOS	97
7.4	DIAGRAMA DE FLUJO DE DATOS	98
7.5	DICCIONARIO DE TÉRMINOS	102
7.6	DIAGRAMA DE CLASES.....	104
7.7	DICCIONARIO DE CLASES.....	105
7.8	DIAGRAMA DE SEGUIMIENTO DE SUCESOS.....	106
7.9	DIAGRAMA DE ESTADOS.....	112
7.10	ACTA DE ACUERDOS.....	113
7.11	SELECCIÓN E IDENTIFICACIÓN DE SOFTWARE Y HARDWARE	120
7.12	DIAGRAMA ENTIDAD-RELACIÓN.....	121
7.13	REGISTRO DE LLAVES SECUNDARIAS	122
7.14	DICCIONARIO DE DATOS	139
7.15	DESCRIPCIÓN DE PROCESOS	161
7.16	ENTRADAS Y SALIDAS	174
7.17	DIAGRAMA ESTRUCTURAL DEL SISTEMA.....	195
7.18	DESCRIPCIÓN DE PROGRAMAS Y RUTINAS	198
7.19	RESULTADO DE PRUEBAS	199
7.20	INTEGRACIÓN AL BANCO DE SISTEMAS.....	201
CAPÍTULO 8: MANUALES DE ADMINISTRACIÓN E INSTALACIÓN		203
8.1	REQUERIMIENTOS MÓDULO DESKTOP	205
8.1.1	REQUERIMIENTOS DE HARDWARE.....	205
8.1.2	REQUERIMIENTOS DE SOFTWARE	205
8.1.3	REQUERIMIENTOS ADICIONALES	205
8.2	INSTALACIÓN MÓDULO DESKTOP EN WINDOWS.....	205
8.2.1	ACCESO AL AUTORUN	205
8.2.2	INSTALACIÓN DEL SAICOCETI.....	206
8.2.3	CONFIGURACIÓN DE BASE DE DATOS.....	208
8.2.4	ACCESO AL SAICOCETI	209
8.3	INSTALACIÓN MÓDULO DESKTOP EN OTRO SISTEMA OPERATIVO.....	210
8.4	MANUAL DE USUARIO.....	210
8.4.1	ACCESO A EMPRESAS.....	211

8.4.2	AGREGAR EMPRESAS	211
8.4.3	BUSCAR EMPRESAS	212
8.4.4	LIMPIAR EMPRESAS	213
8.4.5	ACTUALIZAR EMPRESAS	213
8.4.6	ELIMINAR EMPRESAS	214
8.4.7	CREAR PDF DE EMPRESAS	215
8.4.8	ASIGNAR REPRESENTANTE	216
8.4.9	VER INFORMACIÓN DE CENTROS DE TECNOLOGÍA DE INFORMACIÓN (CTI).....	216
8.4.10	ASIGNAR AUDITORÍAS A LOS AUDITORES (PANTALLA DE AUDITORES)	216
8.4.11	PERSONAS EN CASO DE EMERGENCIA (PANTALLA DE AUDITORES)	217
8.4.12	MODIFICAR PREGUNTAS (PANTALLA DE AUDITORÍAS)	218
8.4.13	REPORTE (PANTALLA DE AUDITORÍAS POR REALIZAR)	221
8.4.14	ASIGNAR LÍDER (PANTALLA DE AUDITORÍAS POR REALIZAR)	225
8.4.15	CONSULTAR BITÁCORA (PANTALLA DE AUDITORÍAS POR REALIZAR)....	226
8.4.16	CAMBIAR CONTRASEÑA	227
8.4.17	ADMINISTRAR USUARIOS	227
8.5	MANUAL DEL SAICOCETI WEB	230
8.5.1	ACCESO AL SAICOCETI WEB	230
8.5.2	DESCRIPCIÓN DE LAS AUDITORÍAS DISPONIBLES EN EL SISTEMA.....	230
8.5.3	ENTRADA AL SAICOCETI WEB.....	231
8.5.3.1	DESCRIPCIÓN DE LAS AUDITORÍAS DISPONIBLES EN EL SISTEMA....	232
8.5.3.2	CONSULTAR AVANCES DE LAS AUDITORÍAS.....	232
8.5.3.3	INICIAR O REANUDAR AUDITORÍAS	233
8.5.3.3.1	SUGERIR PREGUNTAS ADICIONALES A LAS AUDITORÍAS EXISTENTES	234
8.5.3.3.2	SUGERIR AUDITORÍAS ADICIONALES A LAS AUDITORÍAS EXISTENTES	236
8.5.3.3.3	CONTINUAR AUDITORÍA O INICIARLA.....	237
8.5.3.3.4	ELEGIR GRUPO DE TRABAJO.....	238
8.6	MANUAL DE INSTALACIÓN DE LA BASE DE DATOS DEL SAICOCETI	239
8.6.1	REQUERIMIENTOS MÍNIMOS DE LA INSTALACIÓN.....	239
8.6.2	INSTALACIÓN DE LA BASE DE DATOS.....	239
CAPÍTULO 9: CONCLUSIONES		245
APÉNDICE 1 CLUF.....		249
APÉNDICE 2 LICENCIA DE JAVA.....		253
BIBLIOGRAFÍA		259

INTRODUCCIÓN

Sistema de auditoría informática y control a centros de tecnología de información (SAICOCETI).

INTRODUCCIÓN

El Sistema de Auditoría Informática y Control de Centros de Tecnología de Información (SAICOCETI), como cualquier otro sistema de cómputo empresarial, busca ser una herramienta en la simplificación de procesos relacionados con la administración y ejecución de auditorías informáticas mediante los diversos módulos que lo componen permite, desde la administración de usuarios y contenido de las auditorías, hasta la emisión de reportes ejecutivos de las auditorías aplicadas vía Web.

La implementación de este sistema es con el propósito de aprovechar un área de oportunidad que explotada de manera inteligente traerá grandes beneficios tanto a la Facultad de Ingeniería como a la propia UNAM en diversos niveles; se podrán obtener ingresos económicos extraordinarios por la realización de auditorías a empresas y dependencias de gobierno, hasta permitir una mejor formación de los miembros de la comunidad universitaria ya que se tendrá la oportunidad de que profesores y alumnos formen parte de los grupos de trabajo que realicen las auditorías.

Hoy en día las auditorías son un instrumento para que las empresas puedan medir la efectividad, tanto en sus procesos como en sus instalaciones, y de esta manera optimizar las ganancias que se obtienen con la misma cantidad o inclusive menos de recursos y es por ello que se considera una alternativa directa y rápida para mejorar.

Una limitante que se presentaba con la versión anterior del sistema, desarrollado en software propietario, es el hecho de que para la puesta en producción del mismo, con fines comerciales, era necesario hacer un pago de licencias por uso de diverso software requerido para su correcto funcionamiento, por lo anterior se decidió elaborar una nueva versión en software libre, rediseñando la estructura, ya que se incluyeron mejoras sustanciales. En este sentido el Ing. Heriberto Olguín Romo no sólo jugó el papel de director de tesis sino que es el usuario final principal al que va dirigido este producto, con quien se tuvieron juntas de definición de requerimientos y mejoras del sistema avaladas con sus respectivas minutas firmadas por los que formaron parte de las reuniones.

Otro punto por considerar dentro del desarrollo del sistema es que fue registrado como proyecto de investigación ante la Dirección General de Asuntos del Personal Académico (DGAPA) de la UNAM, a través del Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica (PAPIIT) el folio de registro asignado es: IN103007 y gracias al apoyo económico que se recibió fue posible adquirir dos equipos de cómputo, un servidor para desarrollo y otro servidor más robusto para producción. La parte escrita la integran nueve capítulos, en donde el:

Capítulo 1, Software libre, trata las características principales del software libre, tales como: antecedentes, sus principales exponentes y algunas corrientes de trabajo que se derivan de éste, ventajas que ofrece desarrollar software bajo esta corriente y las implicaciones legales que conlleva trabajar en el área de la programación.

En cuanto al *Capítulo 2, Software Propietario*, su finalidad es mencionar algunos antecedentes, características principales y las implicaciones legales que surgen de las licencias de uso de este tipo de software; en particular sus licencias o contratos CLUF (Contrato de Licencias de Usuario Final), PUR (Derechos de Uso del Producto) y APSL (Licencia Apple de Fuente Pública), así como el tipo de productos que usan éstas y las características principales que ofrecen.

En el *Capítulo 3, Software libre vs Software propietario*, se llevó a cabo un análisis minucioso sobre las ventajas y desventajas que conlleva usar software propietario o software libre, partiendo de los capítulos anteriores se profundizó en las características, rendimiento, popularidad, soporte, y otros tópicos que ayudaron para dejar en claro cuál es la tendencia actual en el área de software.

Para el *Capítulo 4, Reingeniería de Software*, se trató lo que es reingeniería de software. Para ello se menciona dónde se originó y algunos conceptos que se relacionan con el término, para después hablar de una institución que ha realizado grandes aportaciones en el campo de la ingeniería de software, posteriormente se hace un análisis de las diferentes etapas por las que se pasa cuando se realiza reingeniería de software, y por último explicar los tipos de reingeniería que se tienen, así como algunas herramientas que se emplean para la reingeniería de software.

En *este Capítulo 5, SAICOCETI en plataforma Microsoft*, se habla de los elementos que constituyeron la versión anterior del SAICOCETI tales como: el lenguaje de programación, el servidor de bases de datos y el servidor Web; se presentan características generales de cada uno de éstos y su relación con el sistema, para finalmente hacer un análisis global de la versión anterior.

Ya en el *Capítulo 6, SAICOCETI en software libre*, se abordan los temas relacionados con la elección de las tecnologías empleadas para desarrollar el SAICOCETI en software libre, tales como: lenguaje de programación, manejador de bases de datos y servidor Web, así como otras tecnologías incluidas en el nuevo sistema.

Los manuales se presentan en los dos siguientes capítulos: *Capítulo 7, Manual técnico*, en donde se tratan los detalles de la implementación del SAICOCETI; y en el *Capítulo 8, Manuales de administración e instalación*, se presentan tanto los manuales de administración, como los de instalación del SAICOCETI, con objeto de facilitar el manejo del mismo en sus diferentes módulos.

CAPÍTULO 1: SOFTWARE LIBRE

En el primer capítulo se dará a conocer las características principales del software libre tales como: antecedentes, sus principales exponentes y algunas corrientes de trabajo que se derivan de éste, ventajas que ofrece desarrollar software bajo esta corriente y las implicaciones legales que conlleva trabajar en el área de la programación.

1.1 HISTORIA Y ANTECEDENTES DEL SOFTWARE LIBRE

Antes de empezar el tema es conveniente saber el concepto de software libre, bastará con una sencilla definición y después se desarrollarán sus aspectos más a fondo: “software libre es aquel que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente.”¹

Como idea o concepto y como se conoce hoy en día surge a principios de la década de los ochenta del siglo pasado, sin embargo, tiene una historia de tiempo atrás para ser lo que es hoy.

A principios de los sesenta, cuando la computación comenzaba a tomar cierto auge, con los grandes distribuidores de hardware, un ejemplo es IBM como pionero de esta área, el software y el hardware se distribuían como una misma entidad y no se tenía idea de que algún día fuera de manera contraria. Cuando se adquiría el soporte técnico o el mantenimiento para los equipos se tenía acceso al catálogo de programas o aplicaciones, así como al código fuente y de esta forma era posible realizar modificaciones específicas o adecuar el software a las necesidades de cada empresa o persona que tuviese el hardware. Entre desarrolladores había un sentido de compañerismo, ya que si algún desarrollador conocía de algún programa apropiado o de alguna rutina o procedimiento le podía pedir el código a la persona que tuviera la aplicación que le interesara y sin ningún problema lo podía usar para adecuarlo a su sistema fuera de implicaciones legales como podría ser hoy en día.

Pero esto dio un giro total cuando IBM proporcionó la posibilidad de comenzar a distribuir su equipo por separado. Esto fue a principios de los setenta, y así surgió una cultura de colaboración. En consecuencia el software cobró valor por si mismo. Hardware y software se transformaron en algo totalmente diferente y separado. Es así como surge el software propietario o licenciado, distinción que será tema del siguiente capítulo.

A partir de la introducción anterior hay otro tema importante que se desarrolla casi en paralelo y es el sistema operativo Unix.

Es un sistema operativo que fue desarrollado por Thompson y Ricio (entre otros) en los laboratorios Bell de AT&T. A principios de los setenta, Unix llegó a muchas universidades con una licencia que permitía su uso para fines académicos, por consecuencia, también venía el código fuente con el que se le podían hacer modificaciones a la versión original y poder estudiarlo para ser impartido a los alumnos de las universidades, razón por la cual Unix se desarrolló de una forma rápida.

Lo anterior fue un panorama general sobre el surgimiento de lo que hoy en día es una corriente que está tomando mucho auge y que ha traído muchas otras implicaciones tanto técnicas como legales.

¹ http://es.wikipedia.org/wiki/Software_libre

1.2 DEFINICIÓN DE SOFTWARE LIBRE

Ubicado el tema central se puede empezar con dos definiciones importantes, que ayuden a tener claro de que se trata, puesto que en algunas ocasiones se confunde mucho este tema, dando como resultado un desarrollo poco próspero, comparándolo con su contraparte, el software propietario.

Primera definición: software libre se refiere a: “La libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De manera más precisa, se refiere a cuatro libertades de los usuarios del software: (1) Libertad de usar el programa con cualquier propósito. (2) Libertad de estudiar cómo funciona el programa, y adaptarlo a sus necesidades. El acceso al código fuente es una condición previa para esto. (3) Libertad de distribuir copias. (4) Libertad de mejorar el programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie. EL acceso al código fuente es requisito previo para esto.”²

Segunda definición: software libre una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. Suele estar disponible gratuitamente en Internet.

Sin embargo no es obligatorio, aunque conserve su carácter de libre, puede ser vendido comercialmente. Equivalentemente, el software gratis o gratuito (denominado usualmente freeware) incluye en algunas ocasiones el código fuente; sin embargo, este tipo de software no es libre en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.³

Con estas definiciones y lo que se ha estado presentando, es posible complementarlo con el siguiente diagrama de software libre.

² <http://www.gnu.org/philosophy/free-sw.es.html> Fecha de consulta: 21/diciembre/2006, 5:45 p.m.

³ http://es.wikipedia.org/wiki/Software_libre Fecha de consulta: 12/diciembre/2006, 7:37 p.m.

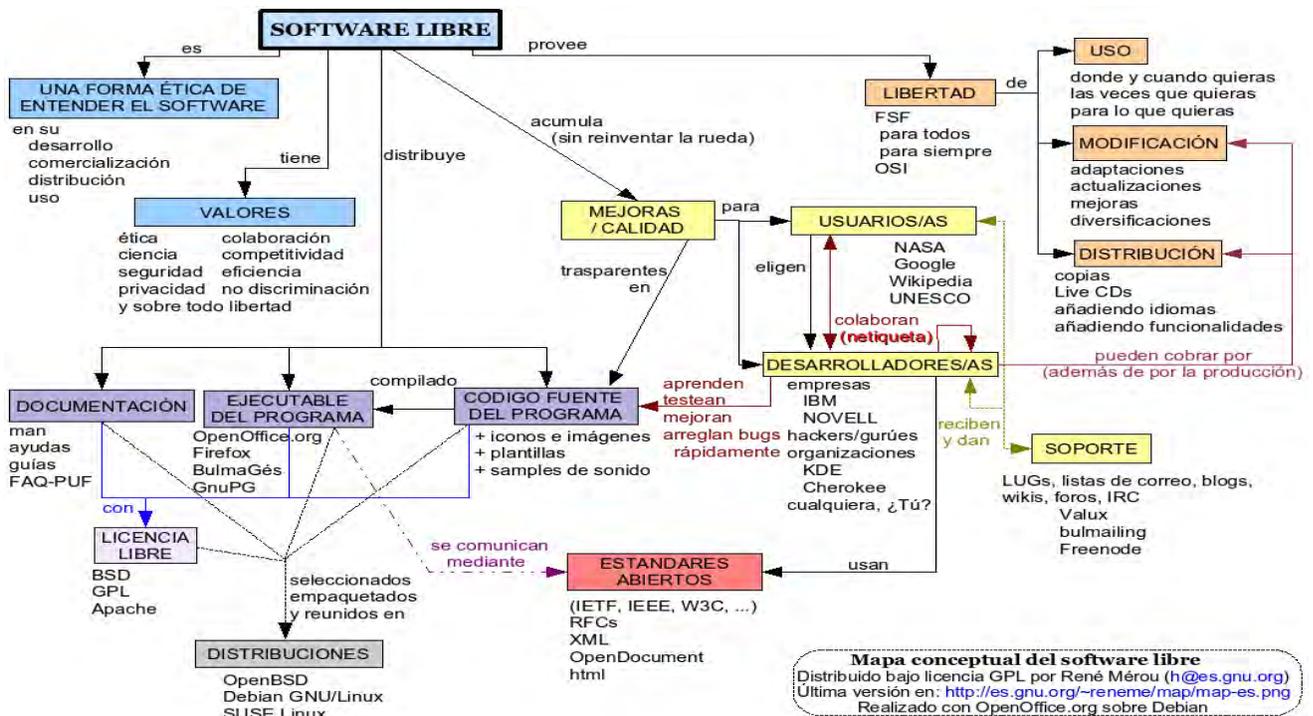


Figura 1. Mapa conceptual del software libre

Con lo anterior (definiciones y el diagrama), se puede decir que cuando se habla de software libre se hace referencia a la “libertad” y no a cuestiones de precio o gratuidad, esto se confunde ya que su denominación en inglés es “free software” y como es sabido la palabra tiene más de un significado y uno de ellos en el idioma inglés para “free” es “gratis”, por lo que es común que el público en general piense que se está hablando de software gratis cuando escuchan el término software libre o free software.

Por lo tanto, cada vez que se haga referencia al término de software libre, se manifiesta la idea de la libertad de poder hacer uso del software para cualquier fin, solamente con la condición de mantener los cuatro puntos que hacen que el desarrollo del software sea libre.

1.3 PRIMEROS INDICIOS DEL SOFTWARE LIBRE

En los antecedentes del software libre, se mencionó que la historia de éste va acompañada del nacimiento del sistema operativo Unix.

Ahora bien, todos estos antecedentes fueron un ensayo para lo que se venía a futuro. Todas estas disputas sobre la propiedad del software desencadenaron muchos problemas, a las personas que tenían que pagar por adquirir programas y no poder compartirlos, y a los desarrolladores que ya no podían partir de algo ya hecho para hacer mejoras (si adquirían un programa tenían que esperar a que se encontraran los errores, ya no podían tener acceso al código fuente).

Esto ocasionó el surgimiento del término software libre como tal, recordando que en un principio no era considerado así y al paso del tiempo esto fue cambiado.

1.4 GNU Y RICHARD STALLMAN (NACE EL SOFTWARE LIBRE)

El movimiento del software libre inicia como tal en el año de 1984, un empleado del Instituto Tecnológico de Massachussets (MIT), Richard Stallman, un desarrollador con una fama de hacker, compartía su código y sus inquietudes con la comunidad de programadores de esos tiempos, y decidió dejar su empleo en el MIT, por los acontecimientos ocurridos con la comercialización del software. Stallman incómodo de las limitaciones y los cambios que ocasionaba el software propietario decide lanzar un manifiesto donde sienta las bases y da a conocer sus motivos para crear una revolución en cuanto al manejo del software y volver a los orígenes donde lo que importaba era el sentido del compañerismo y es aquí donde nace puntualmente el software libre.⁴

1.4.1 GNU

Richard Stallman, a partir de su manifiesto y dando a conocer sus motivos del porque crear este movimiento, tuvo en mente un proyecto ambicioso para promover y fortalecer su idea, que era crear un sistema de software completamente libre al que denominó GNU; el significado es un acrónimo recursivo de GNU's no es Unix.

Al comienzo de este proyecto ya se contaba con sistemas de software como Tex y Emacs; que eran editores de texto.

Lo primero que Stallman comenzó a hacer fue escribir un compilador de lenguaje C, con características acordes a la definición de software libre el cual se llamo GCC, que hoy en día es uno de los compiladores del lenguaje C más populares entre los programadores orientados a esta filosofía.

El sistema operativo para el cual estaban creados, en un principio, toda esta serie de software era para Unix pues era una alternativa muy clara a los programas propietarios de esos tiempos y gracias a las características intrínsecas que hacían que el software fuera libre tenía cierta ventaja en cuanto al reporte y corrección de errores de las distintas versiones de los programas desarrollados bajo esta concepción, ya que las publicaciones o nuevas versiones de estos programas se publicaban casi al instante de liberar el software y como es una comunidad mundial cualquier persona lo podía hacer, en comparación con el software propietario en el cual había que esperar a que el grupo de desarrolladores que habían creado algún programa se dieran cuenta de los errores y los corrigieran hasta la siguiente

⁴ http://es.wikipedia.org/wiki/Software_libre

Fecha de consulta: 12/diciembre/2006, 4:44 p.m.

versión del programa, en algunos casos esto tardaba hasta un par de años, por lo cual algunas veces ya se perdía cierto interés de usar estos programas.

A principios de los noventa GNU ya era muy popular entre los desarrolladores, principalmente en las universidades, y algunas de sus aplicaciones ya estaban bastante mejoradas y podían competir perfectamente con su contraparte, el software propietario. Incluso algunas de esas aplicaciones ya eran de mejor calidad que las otras. También en estos tiempos ya habían logrado un sistema similar a Unix, aunque aún faltaba una de las piezas fundamentales para lograr su sistema operativo completamente libre, el Kernel, que es la parte del sistema operativo que es el núcleo del mismo y se encarga de interactuar con el hardware y hace que todo entre en armonía y funcione.

1.4.1.1 GNU/LINUX

Como se vio en el apartado anterior GNU, hasta ese momento, casi lograba el objetivo principal que Richard Stallman había planteado con la creación de su movimiento de software libre, que era lograr todo un sistema de software libre de propósito general, aunque aún faltaba el Kernel.

Justo es aquí cuando otro personaje importante para el software libre hace su aparición, Linus Torvalds quien publica sus intenciones de hacer un sistema libre parecido a MINIX; tiempo después saca a la luz la primera versión (0.01) y subsecuentemente cada semana sacaba mejoras hasta que finalmente en marzo de 1994 saca la versión (1.0) la cual ya era una versión estable.

Lo importante de esto es que Linus Torvalds construyó para sus sistemas el Kernel, el cual GNU tanto había estado buscando. Con esto, los desarrolladores de todo el mundo incorporan al Kernel de Torvalds las aplicaciones GNU disponibles hasta el momento y surge Linux (la historia de Linux es mucho más extensa, pero no es tema central que corresponde desarrollar), y es así como nace lo que se tendría por nombre GNU/LINUX.

Esto da lugar a que muchas empresas cambien sus modelos informáticos a uso de software libre, pues la calidad y competitividad que éste mostraba tenía una gran mejoría comparándolo con el software propietario.

Otro aspecto es que con el surgimiento de GNU/LINUX se ven beneficiadas todas las universidades, pues podían hacer uso de este sistema operativo para su estudio y modificación y así mismo también para la enseñanza de los procesos de un sistema operativo completo.

1.5 LA FREE SOFTWARE FOUNDATION (FSF)

Esta es una organización creada por Richard Stallman con el principal propósito de preservar y defender los ideales del software libre, así como obtener aportaciones para solventar el trabajo de las personas que se dedican a estos intereses.

Por otro lado, Stallman estableció los reglamentos éticos del software libre, y se dedicó a eliminar las restricciones concernientes a la redistribución y copia del software libre, y, en consecuencia, la promoción del uso de software libre en todas las áreas que sea posible ocupar programas de computadora. Otro propósito que tiene la FSF es el desarrollo de software para evitar el uso de programas que sean propietarios o no libres.

1.6 LA LICENCIA GPL

Como era de esperarse, en software libre había que seguir una cierta reglamentación a pesar de los ideales originales de éste, por lo que surgió la licencia pública GNU.

Generalmente, cuando se escucha hablar de una licencia de software es común pensar que generará una cantidad de restricciones que llevarían a perder las libertades de este movimiento de lucha.

Resulta así contradictorio hablar de una licencia para software libre pero esto aunque no lo parezca tiene bastante sentido.

La finalidad de crear una licencia para software libre es que se hagan valer las cuatro características esenciales que hacen que el software sea libre: copiar o distribuir el software y cobrar por este servicio si se quiere obtener algún tipo de remuneración aunque ésta no sea monetaria.

Por lo tanto, no solamente para esto fue necesaria la creación de este tipo de licencias, sino también para la protección de los autores o coautores de un programa en software libre, pues en los términos y cláusulas de las licencias GPL, que se verán más adelante, se hace explícito que si se recibe una copia modificada del programa original las modificaciones que se le hagan a este supuesto programa, ya sea que funcione o no, protege a los autores y conserva su reputación deslindándolos de los daños que esto pudiera ocasionar.

1.6.1 TÉRMINOS Y CONDICIONES PARA LA COPIA, DISTRIBUCIÓN Y MODIFICACIÓN DE UN PROGRAMA EN SOFTWARE LIBRE

Para dejar más claro el asunto y tener una mejor idea sobre las cláusulas, se citarán algunas de las más importantes de la licencia GPL (estas cláusulas son de la versión 2 que abarcan casi todo el software de la Free Software Foundation).

PRIMER CLÁUSULA

Esta licencia se aplica a cualquier programa u otro tipo de trabajo que contenga una nota colocada por el tenedor del copyright diciendo que puede ser distribuido bajo los términos de esta Licencia Pública General. En adelante, «Programa» se referirá a cualquier programa o trabajo que cumpla esa condición y «trabajo basado en el Programa» se referirá bien al programa o a cualquier trabajo derivado de él según la ley de copyright. Esto es, un trabajo que contenga el programa o una porción de él, en forma literal o con modificaciones y/o traducido en otro lenguaje. Por lo tanto, la traducción está incluida sin limitaciones en el término «modificación». Cada concesionario (licenciatario) será denominado «usted».⁵

SEGUNDA CLÁUSULA

Se puede copiar y distribuir copias literales del código fuente del programa, según lo has recibido, en cualquier medio, de forma adecuada y visible publicada en cada copia un anuncio de copyright adecuado y un repudio de garantía, mantenga intactos todos los anuncios que se refieran a esta licencia y a la ausencia de garantía, y proporcione a cualquier otro receptor del programa una copia de esta licencia junto con el programa. Puede cobrar un precio por el acto físico de transferir una copia, y puede, según su libre voluntad, ofrecer garantía a cambio de unos honorarios.⁶

TERCER CLÁUSULA

Puede modificar su copia o copias del programa o de cualquier porción de él, formando de esta manera un trabajo basado en el programa, y copiar y distribuir esa modificación o trabajo bajo los términos del apartado 1, antedicho, supuesto que además cumpla las siguientes condiciones: (1) Debe hacer que los archivos modificados lleven anuncios prominentes indicando que los ha cambiado y la fecha de cualquier cambio. (2) Debe hacer que cualquier trabajo que distribuya o publique y que en todo o en parte contenga o sea derivado del programa o de cualquier parte de él sea licenciada como un todo, sin carga alguna, a todas las terceras partes y bajo los términos de esta licencia. (3) Si el programa modificado lee normalmente órdenes interactivamente cuando es ejecutado, debe hacer que, cuando comience su ejecución para ese uso interactivo de la forma más habitual, muestre o escriba un mensaje que incluya un anuncio de copyright y un anuncio de que no se ofrece ninguna garantía (o por el contrario que sí se ofrece garantía) y que los usuarios pueden redistribuir el programa bajo estas condiciones, indicando al usuario cómo ver una copia de

⁵ <http://es.tldp.org/Otros/gples/gples.html>

⁶ *Ibíd.*

esta licencia. (Excepción: si el propio programa es interactivo pero normalmente no muestra ese anuncio, no se requiere que su trabajo basado en el Programa muestre ningún anuncio).⁷

CUARTA CLÁUSULA

No está obligado a aceptar esta licencia, ya que no la ha firmado. Sin embargo, no hay nada más que le proporcione permiso para modificar o distribuir el programa o sus trabajos derivados. Estas acciones están prohibidas por la ley si no acepta esta Licencia. Por lo tanto, si modifica o distribuye el programa (o cualquier trabajo basado en el programa), está indicando que acepta esta licencia para poder hacerlo, y todos sus términos y condiciones para copiar, distribuir o modificar el programa o trabajos basados en él.⁸

QUINTA CLÁUSULA

Si quiere incorporar partes del programa en otros programas libres cuyas condiciones de distribución son diferentes, le escribirá al autor para pedirle permiso. Si el software tiene copyright de la Free Software Foundation, escribe a la Free Software Foundation: algunas veces se hacen excepciones en estos casos. La decisión estará guiada por el doble objetivo de preservar la libertad de todos los derivados de nuestro software libre y promover el que se comparta y reutilice el software en general.⁹

Como se puede ver en este pequeño extracto de algunas de las cláusulas de la licencia GPL, hay toda una metodología para hacer una copia o redistribuir alguna mejora o agregar una porción de código para un programa que se esté desarrollando. Así como también la condición de aceptar o no la licencia, que comparada con las licencias de software propietario la GPL es bastante flexible.

Y como parte de la metodología que se menciona en párrafos anteriores he aquí más o menos qué pasos se tienen que seguir para convertir un programa en software libre y como consecuencia que cualquiera pueda hacer uso de él bajo las 4 reglas fundamentales del software libre.

Lo que se tiene que hacer es dar el siguiente formato o etiquetado al principio del encabezado del programa en cada archivo que contenga el programa:

1. Cada archivo debe contener al menos una línea de <<copyright>>, y un indicador donde se pueda encontrar la leyenda completa.

⁷ *Ibíd.*

⁸ *Ibíd.*

⁹ *Ibíd.*

2. Una línea en donde se le ponga el nombre del programa y una breve descripción sobre su funcionamiento.

3. Otra línea que contenga el siguiente formato: Copyright (C) 20aa <Nombre del autor>.

4. Y por último la siguiente leyenda: "Esté programa es software libre. Puede redistribuirlo y/o modificarlo bajo los términos de la Licencia Pública General de GNU según es publicada por la Free Software Foundation, bien de la versión 2 de dicha licencia o bien (según su elección) de cualquier versión posterior.

Este programa se distribuye con la esperanza de que sea útil, pero SIN NINGUNA GARANTÍA, incluso sin la garantía MERCANTIL implícita o sin garantizar la CONVENIENCIA PARA UN PROPÓSITO PARTICULAR. Véase la Licencia Pública General de GNU para más detalles.

Debería haber recibido una copia de la Licencia Pública General junto con este programa. Si no ha sido así, escriba a la Free Software Foundation, Inc., en 675 más Ave, Cambridge, MA 02139, EEUU.

También es necesario conseguir una renuncia de copyright para el programa, si es necesario

CAPÍTULO 2: SOFTWARE PROPIETARIO

La finalidad de este capítulo es mencionar algunos antecedentes, características principales y algunas de las implicaciones legales que surgen de las licencias de uso, de este tipo de software; en particular, las licencias o contratos CLUF (Contrato de Licencias de Usuario Final), PUR (Derechos de Uso del Producto) y APSL (Licencia Apple de Fuente Pública), así como el tipo de productos que usan estas licencias y las características principales que ofrecen.

2.1 ANTECEDENTES PRELIMINARES SOBRE EL SOFTWARE PROPIETARIO

El desarrollo del software propietario y del software libre, como se pudo apreciar en el capítulo anterior, tienen un nacimiento en común, se puede comparar con hermanos gemelos, que en un principio nacen y tienen el mismo origen y al paso del tiempo se convierten en dos entes totalmente diferentes, aunque en apariencia sean iguales, son distintos. Esto fue lo que ocurrió con el software propietario y el software libre, por lo tanto recordando un poco, en la década de los sesenta el software era solamente eso software no había distinción entre éstos, de hecho como ya se mencionó software y hardware eran una misma entidad.

El parte aguas a partir del cual el software se volvió privado, fue como ya se mencionó cuando IBM vio la posibilidad de empezar a separar lo que era una sola entidad; fue entonces cuando se empieza a cobrar por adquirir el software con aplicaciones muy específicas. A partir de ese momento surge el software propietario como tal.

IBM provoca que los compradores de hardware tengan la necesidad de pagar por algo a lo que no estaban habituados. Pero por si esto fuera poco los programas que compraban ya no tenían los derechos ni la posibilidad de adecuarlos a sus propias necesidades, pues se empezó a vender solamente lo que se conoce como programa ejecutable, el cual únicamente contenía el programa para usuario final como tal y se dejó de incluir el código fuente, teniendo como consecuencia que ya no se pudieran hacer modificaciones sobre éste. Esto, como lo se vio en el capítulo anterior, derivó en el movimiento de software libre.

Por otro lado, cobrar por adquirir software se volvió un negocio, esto provoco que surgieran compañías que solamente se dedicaran a la creación de programas para las computadoras (que eran utilizados en esa época), ejemplos visibles de este tipo de compañías son: IBM, Microsoft y Apple.

Aunque, como se presentará a continuación, no todas las compañías de software como las mencionadas anteriormente tuvieron desenlaces fructíferos, esto lo sé describiré en las siguientes líneas de una manera más clara con ayuda de antecedentes de los personajes involucrados en el tema, que de una forma u otra ayudaron y contribuyen hoy en día al surgimiento de distintos tipos de software.

2.2 PRIMEROS IMPULSORES DEL SOFTWARE PROPIETARIO

Los principales personajes que impulsaron la comercialización del software, como unidades individuales y no como compañías fueron dos personajes: William Henry Gates III mejor conocido como Bill Gates y Paul Allen.

Ahora entonces se expondrá una pequeña semblanza de las personas que fortalecieron y fueron los primeros impulsores del software propietario y los fundadores de la compañía más representativa, y líder en el campo del software.

2.2.1 BILL GATES

William Henry Gates III, uno de los personajes más conocidos en el mundo, cofundador de Microsoft compañía desarrolladora del sistema operativo para computadoras personales (PC) más usado en el mundo. Así es como se podría definir a grandes rasgos a Bill Gates. Entonces se hablará, de cómo fue que este personaje se convirtió en el líder empresarial más famoso de la actualidad.

Bill Gates nació en una familia acomodada, que le proporcionó una educación en centros de gran prestigio como la Escuela de Lakeside (1967–73) y la Universidad de Harvard 1973–77. Siempre trabajó en conjunto con su amigo Paul Allen. Se introdujo en el mundo de la informática formando un pequeño equipo dedicado a la realización de programas y éstos a su vez eran vendidos a empresas o a Administraciones Públicas.¹

2.2.3 PAUL ALLEN

Paul Allen es el cerebro detrás de la formación de Microsoft, por así decirlo. Nace en 1953 en Seattle y cursa la secundaria en un colegio de renombre, el Lakeside School.

El destino hizo que esta escuela se convirtiera en una de las primeras de los Estados Unidos en tener acceso a una computadora, es ahí donde conoce a un muchacho dos años menor que él, Bill Gates.

En 1969, Paul, Bill y dos estudiantes más decidieron fundar una empresa. Hasta el año 1970, se dedicaron a examinar computadoras de la firma Computer Center Corporation sin cobrar, teniendo como ventaja el acceso gratuito a las máquinas (en ese entonces no era tan fácil como lo es ahora).

Mientras Gates era admitido en Harvard, Allen, hijo de una maestra y de un bibliotecario de la universidad, se tuvo que conformar con la más humilde universidad estatal de Washington. En cualquier caso ninguno de los dos finalizaron sus estudios profesionales formalmente.

¹ <http://www.biografiasyvidas.com/biografia/g/gates.htm> Fecha de consulta: 27/enero/2007, 1:17 a.m.

2.2.4 STEVE JOBS

Steven Paul Jobs en términos comparativos, es para Apple lo que es Bill Gates para Microsoft. Jobs es un famoso empresario estadounidense, tal vez, se le conoce a últimas fechas por la renombrada y ya muy conocida marca de reproductores música digital iPod.

Toda la vida de Steve Jobs gira en torno a Silicon Valley (EE.UU.). Jobs es uno de los principales hombres del panorama tecnológico. Algo digno de tener en cuenta, ya que Apple, aunque es muy conocida en el ámbito de los sistemas operativos, no es una empresa especialmente importante.

La popularidad de Jobs y de Apple se debe a que son parte de la historia de la informática. Antes de crear Apple junto a Stephen Wozniak, Jobs pasó brevemente, y con pocos resultados, por el Reed College en Portland (Oregon), trabajó en Atari, empresa pionera en el mundo de los videojuegos, y por último trabajó para Hewlett-Packard.

2.2.5 STEVE WOZNIAK

Nació el 11 de agosto de 1950 en SiliconValley, California. El pequeño Woz se divertía jugando con la electrónica y montando radios, voltímetros y calculadoras en casa. A este niño, extremadamente brillante, la escuela le aburría, y en la universidad de Colorado tuvo su primer fracaso pues abandono sus estudios, para trabajar en Hewlett Packard. Wozniak conoció a Steve Jobs durante los años setenta, ambos pertenecían al Homebrew Computer Club.

Jobs convenció a Wozniak de crear una computadora personal más accesible que la Altair 8800 (la primera computadora personal aparecida en 1975). El primer prototipo de la Apple I fue creado en el dormitorio de Jobs, y construido en su garaje. En 1976 Jobs se deshizo de su coche y Wozniak vendió su calculadora HP para conseguir 1.300 dólares, el capital inicial de Apple Computer Inc., Nueva empresa dedicada a comercializar su Apple I por 666 dólares y dando como resultado en el primer año ventas que ascendieron a 774.000 dólares.¹

2.3 EL GIGANTE MICROSOFT

Ahora que ya se conoce quienes fueron los fundadores de Microsoft y una pequeña reseña de Gates y Allen, se verá cómo fue que surgió esta compañía líder en el desarrollo de software propietario.

¹ http://es.wikipedia.org/wiki/Stephen_Wozniak Fecha de consulta: 13/febrero/2007, 7:55 p.m.

Microsoft es un acrónimo de (Microcomputer Software), esta empresa es líder en un gran rubro de software de todo tipo, aunque en un principio solamente se dedicaba a la elaboración de software como interpretes de Basic, Fortran y Cobol. Tenían acuerdos con IBM para elaborar el sistema operativo para sus computadoras.

Posteriormente salió a la venta la primera computadora de IBM con el sistema operativo MS-DOS (Microsoft - Disk Operating System), en la década de los ochentas Microsoft sacó a la luz varias versiones de su sistema operativo, esto ayudo a tener muchas mejoras en el mismo.

Siempre que aparecía una nueva versión del DOS, se volvía más amigable con el usuario, hasta lograr un DOS con interfaz gráfica. Al principio no fueron tan populares, sino hasta que lograron obtener una versión considerablemente estable y operativa, dando como resultado la versión tres, con esta tuvieron un gran repunte en oficinas y lograron consolidarse paso a paso, formando una empresa con una estructura más solida.

En la década de los noventas, Microsoft ya era la empresa líder en el campo del software, liderazgo que le permitió a esta empresa tomar la decisión de terminar la colaboración estrecha que había sostenido hasta el momento con IBM y que en gran parte fue un puntal imprescindible en sus comienzos, sin esta decisión tal vez Microsoft no sería lo que es hoy.

Dicha separación de empresas ocasiono que Microsoft se decidiera a tener su propio sistema operativo, iniciando con Windows 3.11 y al llegar a la versión de Windows 95, la compañía tiene un parte aguas.

Microsoft logra así, aplastar a sus más cercanos competidores, pues Windows 95 se convierte en el principal sistema operativo que utilizaban la mayoría de las computadoras del mundo.

Teniendo casi acaparado o monopolizado el mercado de sistemas operativos, deciden incursionar en otros campos del software como la famosa y conocida enciclopedia electrónica Encarta, y, en el área de las herramientas de oficina electrónica, Office.

Otras áreas donde incursionó Microsoft es: en base de datos con dos de sus productos, Access y SQLServer, con los cuales se volvió bastante popular en el ámbito empresarial, en el área de los navegadores de Internet, se encuentra Internet Explorer, donde al igual que Windows y Office es casi un estándar, o al menos lo era.

Tampoco han dejado de lado el área de desarrollo, o creación de IDE's (Entorno de Desarrollo Integrado) estos son un conjunto de herramientas para programadores, crearon compiladores para el desarrollo de aplicaciones y han incursionado de manera importante con la suite de desarrollo Microsoft Visual Studio, que incluye Visual Basic, Visual C++, Visual C#, Visual J++ y otros más.

Microsoft no confiaba en el desarrollo de Internet; razón por la cual crea MSN (Microsoft Network), con esto pretendían establecer una alternativa más al Internet. Para hacer más

eficiente su nuevo producto adquirió el sistema de correo más popular del momento, Hotmail, integrándolo a sus sistemas operativos como Windows 95 y 98.

Pero obviamente ni siquiera Microsoft podía ser una alternativa para el Internet, por lo que tuvo que cambiar su estrategia y se avocó a tener aplicaciones muy específicas, como lo es el mensajero instantáneo, Messenger; a partir de esto Microsoft transformó su red para ser una más dentro de Internet y desarrolló Internet Explorer.

La tecnología que engloba o la hace una empresa fuerte a Microsoft es, la compra de esta misma tecnología y de sus aplicaciones, como la compra de QDOS a Tim Peterson por un precio muy bajo y al que luego nombrarían MS-DOS, éste sería el principio del gran emporio de Microsoft y otro ejemplo es la adquisición de Mosaic el cual fue el primer navegador para mostrar páginas web.

Por esto último Microsoft debe su gran emporio no solamente a la gran capacidad de intelecto por parte de sus fundadores, sino también a su gran visión para comprar tecnología que en su momento ellos no poseían y volverla parte de la empresa.

Por otro lado, Microsoft ha permanecido en la cima de ventas de software a nivel mundial, pues posee casi el 50 % de las aplicaciones para PC de escritorio. El dominio que tiene Microsoft es tan grande que en algunos países es necesario dominar herramientas electrónicas para oficina, Office, para poder obtener un empleo.

Otro aspecto que hace de Microsoft una empresa líder a nivel mundial es, haber encaminado su estrategia al desarrollo de:

- Compatibilidad del software con versiones anteriores; en concreto, se procura que las nuevas versiones de los sistemas operativos sean lo más compatibles posibles con las anteriores.
- Abundancia de asistentes y otros mecanismos para facilitar el trabajo de gente con escasos conocimientos de informática.
- Tener prioridad a la interoperabilidad entre los productos propios, por encima de la que haya con los ajenos.²

Tradicionalmente, se han distinguido 4 líneas de productos:

1. Sistemas operativos: MS-DOS, Windows.
2. Aplicaciones para la oficina (informática): MS Works, MS Office (Word, Excel, Access, PowerPoint, InfoPath, FrontPage, Outlook y Publisher).
3. Compiladores: MS Visual C++, MS Visual Basic.
4. Herramientas de desarrollo: .NET.

² <http://es.wikipedia.org/wiki/Microsoft> Fecha de consulta: 20/febrero/2007

2.4 APPLE

Si bien Apple en la actualidad, no es una de las compañías más emblemáticas o representantes en cuestión de software; en sus inicios como lo se verá a continuación fue, o sigue siendo quizá, una de las empresas más innovadoras y con mayor visión del mercado, con una serie de productos que han sido adelantados a su era y al paso del tiempo esas ideas son retomadas por otras compañías con un gran éxito.

Algo importante que se verá en las siguientes líneas es que Apple tiene una característica muy peculiar en algunos de sus sistemas operativos de sus computadoras, y es que algunas partes que conforman su sistema operativo contiene partes de software libre, así como algunas de las aplicaciones que ya vienen pre cargadas al comprar una computadora de Apple, y en otra parte también tienen su código regido bajo el software propietario.

Apple Inc., es una empresa innovadora y exitosa, que también a lo largo de su historia ha tenido fracasos de los que aprendió y resurgió para reestructurarse mejor y retomar el camino perdido.

Esta empresa fue creada por 2 personas; basta decir los nombres Steve Wozniak y Steve Jobs y recordar un poco de sus antecedentes y sus inicios para formar esta compañía.

La primera máquina de Apple denominada "Apple I" se vendió por 666.66 dólares. La Apple I se podía obtener en negocios informáticos o por correo, la empresa distribuidora estaba establecida en el garage de Jobs.

Woz y Jobs vendieron una cifra de 175 unidades, sin embargo, la propuesta no le interesó a Hewlett-Packard, ni a la empresa Atari, cuyos directivos se negaron a respaldar a Apple, así como Intel.

A comienzos de 1978, y a un año de haber sido declarada como corporación, Apple ya estaba generando ganancias logrando más de \$2 millones. Las ventas se duplicaron y quintuplicaron al año siguiente, de esta manera Apple ingresó en la revista Fortune 500, y a Jobs lo declararon "el millonario más joven de ese momento" (con tan solo 27 años), en la lista Fortune 400.

El proyecto que había diseñado Steve Wozniak vendió más de 300,000 unidades para fines de 1981, (cinco millones a lo largo de diecisiete años).

Tras varios fracasos con sus computadoras, la compañía se volcó en el pequeño proyecto de Jeff Raskin y Steve Jobs: el Macintosh. Tras numerosos retrasos y problemas, especialmente con el software, que estuvo listo hasta prácticamente el último momento, y tras una arriesgadísima campaña de publicidad, la computadora fue presentada el 24 de enero de 1984.

Las sospechas iniciales de venta, de medio millón de unidades vendidas hasta fin de año parecía que podrían cumplirse durante los primeros meses de 1984, pero poco a poco estas ventas se fueron a la baja provocando el pánico en Apple.

Entre los muchos motivos que hicieron desgastarse las ventas del Macintosh, se puede destacar el alto precio (\$2,495.00 USD), la escasa memoria RAM instalada en la computadora (sólo 128 Kbytes), la inclusión de una única unidad de disco, la falta de puertos de expansión y la falta de software. Otro punto importante por el que se incluye este tema, aparte de pertenecer a la historia del software propietario, es el episodio por el cual pasó esta empresa en el desarrollo de su Macintosh, pues Jobs robó las ideas de la empresa Xerox que era la primera en tener un entorno gráfico e implementar una interfaz de usuario más amigable, y lo implementó a su Macintosh.

Por otro lado, su fundación de bajo nivel se llama Darwin (es el sistema que subyace en Mac OS X) y tiene licencia APSL, una licencia open source y software libre, en sus últimas versiones; misma que se detalla en el siguiente punto. Sin embargo, las capas superiores del sistema (por ejemplo el subsistema gráfico en general) son código propietario.

2.5 TIPOS DE LICENCIAS PARA SOFTWARE PROPIETARIO

Ahora toda esta historia sobre lo que es Microsoft y sus productos, se puede pasar a un campo un tanto más delicado: el aspecto de las licencias que rigen la venta de software propietario. Dentro de las licencias más comunes y no tan conocidas para la venta del tipo de software, se encuentran: CLUF, EULA y adicionalmente se verá el tipo de licencia de la plataforma java.

2.5.1 EULA (ACUERDO DE LICENCIAS DE USUARIO FINAL)

En este tipo de contrato, el dueño de los derechos de un producto solicita al usuario final que reconozca tener conocimiento de las restricciones de los derechos de autor (copyright), patentes, etc. y para que acepte un contrato general para poder hacer uso del producto.

Las compañías utilizan este tipo de contratos para tener un mayor control del uso de sus productos.

A pesar de que contratos de tipo similar se encuentran regulados, en la mayoría de las veces, ya sea por la legislación del país de origen del producto, ya sea también por las legislaciones locales, estos contratos han sido muy atacados por diversas razones, entre las que destacan las siguientes:

- a) El conocimiento del contenido de los contratos es difícil antes de la compra del producto, ya que las cajas de los productos raramente contienen una copia completa del mismo, por lo que el comprador en la mayoría de las ocasiones conoce su contenido después de la compra.

b) En ocasiones se exige al usuario renunciar a realizar reclamos o demandas legales por diversos motivos tales como posibles daños producidos por el producto (o fallos en el) o aceptar la restitución de éstos en la forma y monto que la compañía decida.

c) Este tipo de acuerdo expresa qué tipos de usos se pueden dar y cuáles no al producto, ya que quien lo compra no es legalmente en ninguna forma dueño del producto, sino de una licencia para su uso, considerándose esto último por algunas personas, una limitación a los derechos del consumidor.

Estos acuerdos son unilaterales, pues el usuario no tiene más opción que aceptar o rechazar el contenido del mismo. (En algunos países existen organizaciones de protección al consumidor que previamente los autorizan).

2.5.2 CLUF (CONTRATO DE LICENCIAS DE USUARIO FINAL)

Los EULA y los CLUF son el equivalente a su versión en español respectivamente. El tipo de licencias CLUF rigen en la mayoría del software que distribuye Microsoft, para una mejor detalle véase el apéndice 1.

2.5.3 APSL (LICENCIA DE CÓDIGO FUENTE PÚBLICO DE APPLE)

Como se puede apreciar esta es una licencia de software libre pero la principal razón por la cual se menciona esta licencia es que tiene una peculiaridad y es que hasta que la versión 2 de esta licencia se puede considerar una licencia de software libre y sus antecesoras no, y esto se demuestra con un estudio que se hace de esta licencia.

2.5.3.1 FALTA DE RESPETO A LA PRIVACIDAD

La APSL no permite hacer una versión modificada y usarla para propósitos privados sin publicar los cambios.

2.5.3.1.1 CONTROL CENTRAL

A cualquiera que publique una versión modificada (o incluso utilice, a menos que lo haga con fines de investigación) se le exige que lo notifique a una organización específica, que resulta ser Apple.

2.5.3.2 POSIBILIDAD DE REVOCARLA EN CUALQUIER MOMENTO

La cláusula de finalización dice que Apple puede revocar esta licencia, prohibir y seguir usando todo o parte del software en el momento en el que alguien haga una acusación de violación de patente o derechos de autor.

De esta manera, Apple declina pelear por una patente en disputa (o cuya aplicabilidad al código en mano sea cuestionable). No se tendría la posibilidad de disputar la patente en un tribunal, pues previamente tendría que pelear también contra los derechos de autor de Apple.

Por lo cual con lo que se ha presentado anteriormente no cumple con los puntos básicos para considerarse software libre y es por esta razón principal mente, pero como se mencionó esto solamente ocurría para versiones de esta licencia inferiores a la versión 2.0.

La Licencia de Código Público de Apple (APSL) versión 2.0 está considerada como una licencia de software libre, en la versión 2.0 de la APSL, la definición de "Usos externos" ha sido reducida de modo que sea apropiada para el respeto a las libertades de los usuarios, la posición de la Free Software Foundation (FSF) siempre ha sido que la libertad del software libre está ante todo para los usuarios de dicho software. Las tecnologías, como las aplicaciones web, están cambiando la manera en que los usuarios interactúan con el software.

La APSL 2.0 persigue defender la libertad de aquellos que usen software de esta forma tan novedosa, sin dificultar excesivamente ni la privacidad de los usuarios ni la libertad para usar dicho software.

La FSF considera ahora a la APSL como una licencia de software libre con 2 problemas principales:

- No es un verdadero copyleft, porque permite enlazar con otros archivos que pueden ser completamente privativos.
- No es compatible con la GPL.

En resumidas cuentas Apple tiene una licencia la cual a pesar de que la FSF de GNU la consideran una licencia de software libre no lo es al 100% por las 2 consideraciones anteriores pero es una muy buena iniciativa.

Y recordando que solamente una parte del sistema operativo de Apple tiene esta licencia se combina con otras licencias para logran un producto híbrido el cual da como resultado el Mac OS X y sus versiones posteriores.

2.6 LICENCIA DE CÓDIGO BINARIO (JAVA)

Como un tema obligado se tiene que revisar la licencia de uso de la plataforma de desarrollo Java como se verá a continuación esta licencia se encuentra en el límite entre el bien y el mal por así decirlo.

A lo que hace referencia es que a diferencia de las licencias que se han analizado, donde solo depende del software Java es la excepción a la regla, ya que no solo se toma en cuenta el software sino que además hay que tomar en cuenta una serie de especificaciones que se deben cumplir, en estas es donde se describen las tecnologías que componen la plataforma. Las especificaciones Java Specification Request (JSR), son el derivado de un proceso establecido en la Java Community Process (JCP), la cual es una comunidad que permite a personas interesadas en involucrarse con la definición de la plataforma para versiones futuras, esta es la razón principal por la que la licencia de uso de Java, el software no sólo es lo más importante o el único punto de análisis, si no como una parte esencial de esta son las especificaciones de cada parte de Java, como cada una de las tecnologías como por mencionar algunas de estas especificaciones se encuentran: EJB 2.0, JSP y JVM, etc.

Cada tecnología de Java, tiene una especificación que ha sido analizada por un gran número de personas dedicadas a estos asuntos.

En primer lugar un grupo de expertos se dedica a la definición de la tecnología. Los documentos sobre los que trabajan son accesibles a todo el mundo a través de Internet, y cualquiera puede opinar y enviar recomendaciones al grupo de expertos.

Una vez que los expertos han llegado a un acuerdo en la especificación esta ha de ser aprobada por el comité en turno, y mediante votación pública en la que necesita una mayoría simple de los votos emitidos para aprobarse.

Si es aprobada pasa a una tercera fase, donde es toda la comunidad la que revisa y comenta la especificación y es así como se van desarrollando estas especificaciones a grandes rasgos.

En primera instancia es posible que surja la duda del considerar esta licencia en un capítulo de software propietario, si a simple vista se puede deducir que el desarrollo de estas especificaciones es como a evolucionado el software libre en cooperación mutua y con un espíritu de software libre, pero hay una pequeña condición que hace que esta licencia sea considerada de software propietario, y ahora es cuando queda más claro el por qué es tan importante el software que forma la plataforma de desarrollo de java como sus especificaciones.

En cuanto a la parte software de la plataforma, las ideas del movimiento del software libre, estudiar y modificar el código fuente del software sin restricciones, pueden entrar en conflicto con el principio básico de la plataforma Java: escríbelo una vez y ejecútalo en cualquier sitio, adaptación de "Write Once Run Anywhere".

Ahora es que pese a tener una licencia de uso propietaria sus especificaciones son claramente libres, es aquí donde se hace referencia a las primeras frases de este apartado, donde surge la gran incógnita de cómo catalogar a Java como parte del software libre o del software propietario, considerando estrictamente a los términos de la licencia de código binario de Java, es clasificado software propietario, aunque en esencia sea libre, para tener

una idea más concreta del tipo de licencia y restricciones de la licencia de código binario de Java, puede verse más a detalle las normas que rigen a Java en el extracto de licencia de código binario de Java que se anexa en el apéndice 2.

Como es posible apreciar con el análisis de algunas de las licencias más sobresalientes o controversiales de software propietario, ya se puede ver la diferencia que existe entre licenciar al software, ya sea libre o propietario, en el siguiente capítulo se resaltarán de forma más clara y con un conocimiento más amplio sobre estos temas las principales ventajas y desventajas que se tienen al tomar partido por una u otra corriente.

CAPÍTULO 3: SOFTWARE LIBRE VS SOFTWARE PROPIETARIO

A lo largo de este capítulo se tratará de hacer un análisis minucioso sobre las ventajas y desventajas que conlleva usar software propietario o software libre, partiendo de una idea específica basada en los dos capítulos anteriores en los cuales se describen las características de cada tipo de software, ahora se verá más a fondo las características, rendimiento, popularidad, soporte, y otros tópicos que sirvan para dejar en claro cuál es la tendencia actual en el área de software.

3.1 EL PASADO, SOFTWARE PROPIETARIO

Inicialmente se analiza el tipo de software que ha prevalecido en los últimos 30 años desde su surgimiento como una sola entidad; ahora, recordando el primer capítulo, es claro que al comenzar con este análisis surja la pregunta: ¿por qué tomar una alternativa diferente al software propietario?, y considerándolo de una manera fría y poco innovadora o hasta cierto punto conformista y poco informada, no hay razón por la cual migrar o probar nuevas alternativas diferentes de las conocidas: si es lo que ha funcionado.

¿Cuál será la razón por la que el software propietario ha prevalecido como líder en el mercado?; a pesar de lo mencionado en el capítulo anterior (“Software propietario”), las licencias propias de este tipo de software son restrictivas por no permitir al comprador de software hacer mucho, además de tener pocas garantías.

En este capítulo se presentan las deficiencias de este software, cuya vigencia ha sido dominante en las últimas tres décadas.

Precisamente, en este capítulo se verá por qué a pesar de esto, se ha sostenido como el tipo de software dominante en el mercado.

3.2 VENTAJAS DEL SOFTWARE LIBRE

El software libre presenta una serie de ventajas sobre el software propietario por los derechos que otorga a sus usuarios. Algunas de estas ventajas pueden ser más funcionales para los usuarios particulares y otras, para las empresas. A continuación se enlistan algunas de éstas.

1. El costo de compra y su libre uso.
2. El software, como mercancía, por lo general no está a la venta. Lo que el comprador adquiere, por medio de una transacción monetaria o sin ella, es una licencia respecto de los usos que puede dar a los programas en cuestión. El software no sólo tiene un precio de compra de licencia, también tiene costo de operación y mantenimiento.
3. Ahora, es importante para el comprador o, en este caso, para el usuario poder mantener los costos mencionados bajo control.

El usuario que adquiere software libre lo hace sin ninguna transacción monetaria o a muy bajo costo y con un conjunto de beneficios muy amplios. Cualquier persona con una PC lo puede usar software libre.

Para la mayoría de los usuarios el software libre es una opción atractiva por las libertades que éste garantiza sin necesidad de verse agobiados por el precio. Pero en el caso de empresas y del sector privado, el costo del software es un factor importante que llega a ser determinante en la elección de nuevos sistemas.

Entonces, se concluye por ahora que el software libre prácticamente carece de costo de licencia y por lo tanto, esta parte del presupuesto se puede destinar a mejores fines: mejoras, adaptación de los programas y el desarrollo tecnológico del software.

La mayor parte del software libre se basa en la creencia de que el usuario no es un usuario de escasos conocimientos computacionales, se parte de la confianza en los conocimientos y habilidad del usuario, un ejemplo muy claro de esto es el uso de los Shell de cualquiera de las distribuciones del sistema operativo Linux o Unix, donde al ejecutar cualquier comando no se cuenta con una niñera que esté pendiente cuando se ejecuta el comando.

Esta situación se observa desde el diseño del sistema operativo: decenas de pequeños comandos que hacen una tarea específica y se pueden conectar entre sí para efectuar tareas más complejas.

Ahora un pequeño fragmento del documento de Miguel de Icaza: El software libre tiene las siguientes características:

- Todo el mundo tiene derecho de usarlo sin costo alguno.
- Todo el mundo tiene derecho a acceder a su diseño y aprender de él. Es como obtener las instrucciones para construir un automóvil.
- Todo el mundo tiene derecho de modificarlo: si el software tiene limitaciones o no es adecuado para una tarea, es posible adaptarlo a necesidades específicas y redistribuirlo libremente.
- No tiene un costo asociado (gratuito).
- Es de libre distribución (cualquier persona puede regalarlo, venderlo o prestarlo).¹

Estos puntos que menciona Miguel de Icaza establecen una gran ventaja sobre el software propietario, pues para usarlo se debe pagar una licencia de uso y el usuario está atendido a las restricciones del fabricante o distribuidor.

El reciente auge del software libre se debe en mayor medida a Internet. Ésta ha permitido que las personas interesadas en los componentes del software libre se pongan fácilmente en contacto.

Internet actúa de esta manera como un catalizador que acelera el desarrollo y sintetiza el conocimiento en áreas muy específicas. Diferentes motivaciones impulsan a los contribuidores y desarrolladores a trabajar en el software libre, entre las más importantes están:

- Reutilización del conocimiento: esto permite que la gente vuelva a ocupar el conocimiento que se ha sintetizado en el software. En lugar de comenzar siempre desde el “inicio” (como lo promueve de manera eficiente el lenguaje de programación Java), siempre es posible empezar un proyecto a partir de un fundamento establecido. Esto equivale a la manera en la que la ciencia se desarrolla: no parte

1 <http://primates.ximian.com/~miguel/emexico2.html>

desde cero, parte de los descubrimientos previos y se innova con base en el conocimiento ya existente.

- El deseo de crear nuevo software.
- La posibilidad de tener el control del software. Es importante para aplicaciones de misión crítica tener un control total sobre posibles problemas en cualquier punto.
- El deseo de crear software más robusto.
- Crear aplicaciones de bajo costo.
- La posibilidad de adaptar el software a sus necesidades.
- Aprender alguna técnica de programación.

Una de las principales ventajas que ofrece el uso de software propietario está reflejada en los requerimientos de hardware, aunque es casi imposible generalizar, hay varios estudios y trabajos con los cuales se ha demostrado que el uso de software libre logra un mejor rendimiento que su contraparte (software propietario). El ejemplo más significativo de esto se encuentra en el ámbito del Internet pues alrededor de 70% de las aplicaciones y páginas web que se encuentran están alojadas en servidores con software libre en sus distintas distribuciones; esto se debe al uso sin necesidad de la interfaz gráfica, lo que reduce los requerimientos mínimos de uso, en consecuencia el costo de implementación se vuelve considerablemente más barato.

Otro punto por considerar es la relación usuario-proveedor, pues el software libre garantiza una independencia con respecto al proveedor gracias a la disponibilidad del código fuente. Cualquier empresa o profesional, con los conocimientos adecuados, puede seguir ofreciendo desarrollo o servicios para una aplicación específica.

En el mundo del software propietario, sólo el desarrollador de la aplicación puede ofrecer todos los servicios; en el software libre, como su denominación lo indica, el uso es libre: todo aquel que lo tiene en su poder tiene la posibilidad de usarlo cuantas veces quiera, en cuantas máquinas quiera, y con los fines que quiera. De esta manera, utilizándolo, el usuario se libera de toda dependencia de un proveedor único, y puede administrar su crecimiento y operación con total autonomía.

Uno de los grandes problemas en la industria del software propietario es la dependencia que se crea entre el fabricante con el cliente, este hecho se acentúa con especial gravedad cuando el fabricante no entrega el código fuente. El cliente queda atado inevitablemente al código para nuevas versiones y en general para cualquier mejora que necesite.

Si el programa usa un formato de almacenamiento estándar, entonces el usuario puede estar seguro de que en el futuro podrá seguir descifrando la información.

A menudo los proveedores de software propietario se ven obligados a dejar de fabricar un producto por un cambio drástico de las condiciones del mercado, o simplemente porque consideran que ya no podrán rentabilizar la inversión.

Disponiendo del código fuente, cualquier programador puede continuar su desarrollo y sus actualizaciones hasta que el cliente decida que es el momento adecuado para migrar a un nuevo sistema informático, esto ocurre muy a menudo en las empresas que están teniendo una migración de sus sistemas hacia software libre al darse cuenta de los beneficios que obtendrán con este cambio.

Esto ocurre porque el software propietario habitualmente se vende en forma de paquete, y en muchas ocasiones no se adapta a las necesidades específicas de las empresas. Es importante señalar que una gran parte de la industria del software se basa en desarrollar proyectos donde se requiere software personalizado.

Gracias al hecho de que se dispone del código fuente, el software libre permite personalizar los programas tanto como sea necesario hasta que cubran con puntualidad los requerimientos del usuario. La personalización es un área muy importante en la que el software libre puede responder mucho mejor que el software propietario y a costos mucho más razonables.

Un gran porcentaje de uso de software en los países es interno, para empresas y dependencias de la administración pública, que requieren un alto grado de personalización y donde el software puede proporcionar desarrollos más económicos.

3.3 DESVENTAJAS DEL SOFTWARE LIBRE

Como puede apreciarse son un sinnúmero las ventajas que se ofrecen con el uso del software libre, ahora se presentan algunas de las desventajas que trae consigo el software libre.

- Solamente proyectos importantes y de trayectoria tienen buen soporte, tanto de los desarrolladores como de los usuarios. Sin embargo, existen muchos proyectos más recientes que carecen del compromiso necesario por parte de sus usuarios o desarrolladores para que sean implementados de manera confiable. Estos proyectos importantes que tienen un excelente soporte cubren más del 90% de las necesidades de cómputo del usuario promedio.
- La curva de aprendizaje es mayor. Esto se debe en gran medida a lo mencionado anteriormente sobre el uso de software libre, que generalmente los usuarios de este tipo de software son gente con mucha experiencia computacional, pero si antes los usuarios ya usaron software propietario generalmente tardan más en aprender a usar algún tipo de software libre.
- El software libre no tiene una garantía específica o comprometida de parte del desarrollador de software libre.
- Se necesita dedicar recursos a la reparación de errores. Sin embargo en el software propietario es imposible reparar errores, hay que esperar a que saquen a la venta una versión más reciente.

- No existen compañías únicas que respalden toda la tecnología.
- Las interfaces gráficas de usuario (GUI) y la multimedia apenas se están estabilizando. Aunque hay un número cada vez mayor de usuarios que aseguran que las interfaces gráficas más populares en el software libre (KDE, GNOME y el manejador de ventanas WindowMaker) son ya lo suficientemente estables para el uso cotidiano y lo suficientemente amigables para los usuarios inexpertos de la informática.
- La mayoría de la configuración de hardware no es intuitiva. Se requieren conocimientos previos acerca del funcionamiento del sistema operativo y los fundamentos del equipo a conectar para lograr un funcionamiento adecuado. Sin embargo la documentación referente a la configuración del hardware es tan explícita y detallada que permite al usuario inexperto profundizar en el conocimiento de su hardware en muy pocas horas y una vez teniendo ese conocimiento la configuración se vuelve trivial aunque la mayoría de los casos el usuario no está acostumbrado a leer un manual de referencia por lo cual se vuelve un procedimiento tormentoso a sólo limitarse a dejar que el instalador lo haga.
- El usuario debe tener nociones de programación. La administración del sistema recae mucho en la automatización de tareas, esto se logra utilizando en muchas ocasiones lenguajes de programación como: perl, python, Shell scripts, etc.
Sin embargo, existen en la actualidad muchas herramientas visuales que permiten al usuario no técnico llevar a cabo tareas de configuración del sistema de una manera gráfica muy sencilla sin la necesidad de conocimientos de programación.
- En sistemas con acceso a Internet, se deben monitorear constantemente las correcciones de errores de todos los programas que contengan dichos sistemas, ya que son fuentes potenciales de intrusión. En el software propietario también se deben monitorear constantemente las correcciones de errores de todos los programas además es imposible reparar las vulnerabilidades (que en su mayoría son reparaciones triviales) por uno mismo, puesto que hay que esperar a que la compañía fabricante libere la actualización, en algunos casos hay que pagar dinero extra por obtener ésta.

La diversidad de distribuciones, métodos de empaquetamiento, licencias de uso, herramientas con un mismo fin, etc., pueden crear confusión en algunas personas acostumbradas a realizar algún tipo de procedimiento con una herramienta única monopolizada por el mercado.

3.4 VENTAJAS DEL SOFTWARE PROPIETARIO

Como en capítulos anteriores ineludiblemente hay que hacer una subdivisión al software propietario, donde entra el software de Microsoft ya que cuenta con la mayoría del software propietario en las computadoras del mundo.

3.4.1 VENTAJAS DEL SOFTWARE PROPIETARIO EN GENERAL

- El control de calidad. Las compañías productoras de software propietario por lo general tienen departamentos de control de calidad que llevan a cabo muchas pruebas sobre el software que producen.
- La difusión de publicaciones acerca del uso y aplicación del software. Existe gran cantidad de publicaciones, ampliamente difundidas, que documentan y facilitan el uso de las tecnologías provistas por compañías de software propietario, aunque el número de publicaciones orientadas al software libre va en aumento.
- Recursos a la investigación. Se destina una parte importante de los recursos a la investigación sobre los usos del producto.
- Hay un amplio campo de expansión de uso en universidades. Los planes de estudios de la mayoría de las universidades de México tienen tradicionalmente un enfoque marcado al uso de herramientas propietarias. Las compañías fabricantes ofrecen a las universidades planes educativos de descuento muy atractivos. De ahí que los recién egresados pueden comenzar su vida productiva utilizando estos productos de inmediato. No obstante, en los centros de estudio más prestigiados se observa un cambio en esta tendencia y otros lo adoptaron desde los inicios del software.
- Personal altamente capacitado. Se tienen contratados algunos programadores eficaces y con bastante experiencia.
- Software para aplicaciones muy específicas. Existe software propietario diseñado para estas aplicaciones el cual no existe en ningún otro lado más que con la compañía que lo produce.
- Uso común por los usuarios. El software propietario de marca conocida ha sido usado por muchas personas, es relativamente fácil encontrar a alguien que lo sepa usar.

3.4.1.1 VENTAJAS DEL SOFTWARE MICROSOFT

- Uso común por los usuarios. El sistema operativo Windows (en sus diferentes variantes) se encuentra ya instalado en la gran mayoría de las máquinas que se encuentran trabajando en México y en el mundo. Siendo más notable en las computadoras de escritorio, no como en el lado del servidor, donde los diferentes variantes de Unix y Linux mantienen mucho terreno como lo se menciona anteriormente.
- Curva de aprendizaje menor. Al utilizar productos fundamentados en los sistemas operativos de Microsoft, el tamaño de la curva de aprendizaje de la gente que ya utiliza esto disminuye al no tener que aprender el uso de un sistema operativo nuevo y de nuevas herramientas, aunque ya existen reportes de experiencias con usuarios

reales en quienes la curva de aprendizaje de GNOME o KDE fue mínima respecto a sus contrapartes en software propietario.

- Soporte de las herramientas por diversas compañías. Las herramientas de Microsoft son soportadas por una gran cantidad de compañías de todos tamaños a nivel nacional e internacional. Además existe una red de certificaciones que proveen de “credibilidad” a las soluciones creadas por cada compañía.
- Existen numerosas aplicaciones desarrolladas para la plataforma Win32 que no han sido portadas a otras plataformas. Aunque existen numerosas herramientas, libres o no, que facilitan la tarea de migración o reemplazos disponibles en las demás plataformas.
- Mejoras para desempeño en ambientes de red. Microsoft ha estado mejorando mucho sus productos, para que tengan un mejor desempeño en ambientes de red. Sin embargo aun mantienen un rezago importante, ya que estas mejoras no han sido probadas lo suficiente por el mercado y la falta de interés por la seguridad es evidente, ésta es una de las principales razones del uso de sistemas en software libre para ambientes de red.

3.4.2 DESVENTAJAS DEL SOFTWARE PROPIETARIO EN GENERAL

- Nulificación de desarrollo tecnológico de la industria nacional, respecto de la extranjera (las aplicaciones de consumo masivo se desarrollan en otros países).
- Dependencia a proveedores. En la mayoría de los casos el gobierno se hace dependiente de un solo proveedor.
- Descontinuación de una línea de software. Si una compañía fabricante de software es comprada por otra más poderosa, es probable que esa línea de software quede descontinuada y nunca más en la vida vuelva a tener una modificación.
- Quedar sin soporte técnico. Si la compañía fabricante del software propietario se va a la banca rota, el soporte técnico desaparece y por lo tanto la posibilidad de en un futuro tener versiones mejores de dicho software. No habría la posibilidad de corregir los errores. Los clientes que contrataron licencias para el uso de ese software quedan abandonados a su propia suerte.
- Imposibilidad de compartir. Si una dependencia de gobierno tiene funcionando exitosamente un sistema dependiente de tecnología propietaria no lo puede compartir con otras dependencias a menos que cada una de estas contrate todas las licencias necesarias.
- Ilegalidad de copias sin licencia para el efecto. Es ilegal hacer copias del software propietario sin antes haber contratado las licencias necesarias.
- Derecho exclusivo de innovación. La innovación es derecho exclusivo de la compañía fabricante. Si alguien tiene una idea innovadora con respecto a una aplicación

propietaria, tiene que elegir entre vender la idea a la compañía dueña de la aplicación o escribir desde cero su propia versión de una aplicación equivalente, para una vez logrado esto poder aplicar su idea innovadora.

- Ilegal o costosa la adaptación de un modulo del software a necesidades particulares. Es ilegal extender una pieza de software propietario para adaptarla a las necesidades particulares de un problema específico. En caso de que sea vitalmente necesaria tal modificación, es necesario pagar una elevada suma de dinero a la compañía fabricante, para que sea ésta quien lleve a cabo la modificación a su propio ritmo de trabajo y sujeto a su calendario de proyectos.
- Soporte técnico ineficiente. En la mayoría de los casos el soporte técnico es insuficiente o tarda demasiado tiempo en ofrecer una respuesta satisfactoria.
- Secreto del código fuente. El funcionamiento del software propietario es un secreto guardado celosamente por la compañía que lo produce. En muchos casos resulta riesgosa la utilización de un componente que es como una caja negra, cuyo funcionamiento es desconocido dando resultados impredecibles. En otros casos es imposible encontrar la causa de un resultado erróneo, producido por un componente cuyo funcionamiento se desconoce.
- Cursos de aprendizaje costosos. Es difícil aprender a utilizar eficientemente el software propietario sin haber asistido a costosos cursos de capacitación.

3.4.2.1 DESVENTAJAS DEL SOFTWARE DE MICROSOFT

- Sistemas operativos mono usuarios. Los sistemas operativos Windows son mono usuario, es decir, puede haber muchos usuarios pero sólo uno puede estar trabajando a la vez aunque a ultimas fechas han tratado de cambiarlo.
- Incompatibilidad con varias aplicaciones. Los service packs de Windows NT son incompatibles con muchas aplicaciones.
- Inestabilidad de las aplicaciones que corren sobre Windows. Las aplicaciones de Microsoft que corren sobre Windows también se consideran inestables.
- Sistemas operativos inestables en ambientes de red o condiciones de alto rendimiento. Por la experiencia, los sistemas operativos Windows se consideran inestables en ambientes de red, o en condiciones de alto rendimiento.
- Necesidad de contratar paquetes de soporte técnico. Es necesario contratar paquetes de soporte técnico para la resolución de problemas y su costo es muy elevado.
- Costo elevado de licencia. El costo de las licencias es muy elevado (aunque en ocasiones resulta ser más barato que las soluciones que ofrecen otras compañías productoras de software propietario).

- Oscuridad de la documentación sobre herramientas. La documentación de las herramientas para el desarrollo es muy limitada y oscura, debido a la falta de claridad en el funcionamiento que prefieren mantener.
- Descuido respecto a la seguridad. El descuido de Microsoft por la seguridad de sus sistemas operativos es evidente.
- Código fuente oculto. Cada componente de Microsoft es una caja negra cuyo funcionamiento detallado se desconoce.

CAPÍTULO 4: REINGENIERÍA DE SOFTWARE

En este capítulo se trata el concepto de lo que es reingeniería de software. Para ello se indicará dónde se originó y algunos conceptos que se relacionan con el término para después hablar de una institución que ha realizado grandes aportaciones en el campo de la ingeniería de software, y luego pasar a hacer un análisis de las diferentes etapas por las que se pasa cuando se realiza reingeniería de software, y por último explicar los tipos de reingeniería que se tienen, así como algunas herramientas que se emplean para la reingeniería de software.

4.1 ORIGEN DE REINGENIERÍA

El origen de la reingeniería, actividad que hoy en día es tan importante en diversos procesos con los que se interactúa en la vida diaria, reside en el ámbito de la administración, y de hecho, en muchos lugares del mundo es oficialmente conocida como reingeniería de procesos.

Es necesario retroceder al año 1898 cuando los Estados Unidos sostenían una guerra con España para poder conocer los orígenes de la reingeniería. Se tienen datos que la Marina de los Estados Unidos lanzó un total de 9500 proyectiles, pero solamente 121, que representan 1.3%, acertaron el blanco deseado, y lo más sorprendente es que en aquellos años representaba el máximo de eficiencia que se podía obtener. Cuatro años después, la Marina de los Estados Unidos se las ingenió para dar en el blanco con todos sus disparos, y, de hecho, la mitad de las balas podían acertar en un cuadrado de 1.7 metros de lado. Lo anterior se logró gracias a un oficial de artillería naval llamado William Sowden Sims quien es reconocido por ser el primero en llevar a cabo lo que hoy se conoce como reingeniería.

Apuntar un cañón en pleno mar abierto hace un siglo, era algo muy complicado, ya que tanto el cañón como el blanco se encontraban en movimiento constante. Los componentes que formaban parte de la artillería naval eran simples en esos tiempos: un cañón, una manivela para orientarlos hacia el ángulo deseado y un anteojo de larga vista sobre el cañón, con el fin de mantener el blanco en la mira. Y en base a los elementos que formaban parte del proceso, William Sowden Sims ideó una manera bastante sencilla para incrementar el número de proyectiles que acertaban en el blanco, mediante una compensación en la elevación y en el tiempo de balanceo del barco.

Para empezar, Sowden sugirió que la relación de los engranes fuera reglamentada para que de esta manera al artillero le fuera más fácil elevar o bajar el cañón y así le fuera posible seguir su objetivo, a pesar de los balanceos del barco. El siguiente paso fue que cambiar el lugar donde se ubicaba la mira del cañón y de esa manera el artillero no se distrajera al disparar para que así pudiera tener el blanco en la mira durante todo el proceso de disparo y por lo tanto abría fuego con la puntería continua.

Con base en sus propios cálculos, William Sowden Sims presentó una hipótesis sobre modificaciones que consideró ayudarían aumentando la precisión del número de tiros acertados en un 3000 %. Pero, aún más relevante que este increíble incremento era la manera de obtenerlo, ya que no había la necesidad de costos adicionales para cambiar o incrementar la tecnología empleada, y mucho menos el número de personas encargadas de la artillería, de manera que con el paso de un corto período de tiempo se llegó al 3000 % de incremento en la efectividad. Y éste es considerado el primer caso documentado que se tiene de aplicar el proceso reingeniería.

4.2 DEFINICIÓN DE REINGENIERÍA

Ya en el punto anterior se citó un caso en el se aplicó la reingeniería, además de que ilustra el hecho de que el concepto de reingeniería es aplicable en diversos ámbitos, lo que permite inferir una definición de lo que es la reingeniería, como un proceso a través del cual se tiene como resultado un cambio radical para obtener un mejor aprovechamiento de los recursos. Pero en realidad, hoy en día se tiene una gran variedad de definiciones que muchas veces los autores ajustan de acuerdo al objetivo al que van ligadas. A continuación se presenta la definición de reingeniería sobre la cual se va a trabajar:

*“Reingeniería es la actividad en la que los procesos son objeto de una revisión fundamental y rediseño radical, para lograr así la optimización de los flujos del trabajo y la productividad de una organización.”*¹

Ahora, un punto muy importante para poder lograr la completa comprensión de la definición anterior es especificar lo que se va a entender como un proceso:

*“Definimos un proceso de negocio como un conjunto de actividades que recibe uno o más insumos y crea un producto de valor para el cliente.”*²

Las anteriores definiciones, en principio, podrían parecer un poco alejadas de los propósitos para los cuales se quieren emplear, y es natural, debido a que como ya se comentó al inicio del capítulo, en sí los orígenes de la reingeniería yacen en la administración, y por ello se ve que las definiciones anteriores se relacionan más con esa disciplina. En el siguiente punto se verá a fondo el tipo de reingeniería que interesa analizar, la reingeniería de software.

4.3 REINGENIERÍA DE SOFTWARE

En el apartado anterior ya se definió en qué consiste el concepto general de reingeniería, ahora se va a definir lo que es la reingeniería aplicada a sistemas de software. Para tal objetivo, no se va a citar una definición de alguna de las fuentes consultadas, sino en base a dicho conjunto de definiciones se presenta la siguiente:

Reingeniería de software es el establecimiento y uso de principios de la ingeniería a fin de obtener ó transformar económicamente software, para que sea fiable, de fácil mantenimiento y que funcione eficientemente, a partir de sistemas heredados extrayéndoles los conocimientos que permitan crear un nuevo sistema que mejore al anterior con el fin de reemplazarlo.

Antes de entrar de lleno a la explicación de la definición que presentada hay que hacer notar que los sistemas de información heredados, en la mayoría de los casos, son el eje sobre el cual se sustentan las empresas. Un sistema de información heredado puede ser definido

¹ Manganelli, Raymond; Klein, Mark “Como Hacer reingeniería” Editorial Norma, Colombia CO. 1995

² Hammer, Michael & Champú, James “Reingeniería: Olvide lo que usted sabe sobre cómo debe funcionar una empresa ¡Casi todo ésta errado!” Editorial Norma. Bogotá. Co. 1994

como: "Cualquier sistema de información que significativamente se resiste a la modificación y evolución."³

Una vez que se han aclarado los conceptos que se emplean en la definición, se explicará detalladamente. Lo primero que se menciona es que la reingeniería de software es el establecimiento y uso de principios de la ingeniería, con estos aspectos se hace notar que para realizar un proceso de reingeniería es necesario emplear los fundamentos con los cuales se rige la ingeniería, de manera que permitan llevar a cabo el objetivo, o bien establecer otros nuevos que sirvan para el mismo fin. De tal manera que, como ya se detallará en el siguiente apartado la reingeniería es un problema de ingeniería.

Por otro lado, en la definición también se menciona que el establecimiento y uso de principios de la ingeniería tienen el fin de obtener o transformar económicamente software, esto porque regularmente los sistemas que son candidatos para aplicarles reingeniería son sistemas que representan un gran valor para las empresas y porque el reemplazo total del sistema no muchas veces es viable. Por lo que resulta necesario aplicar las técnicas para transformar un sistema actual a otro nuevo, además se tiene el otro lado de la moneda cuando el hecho de reemplazar el sistema es más económico que su transformación, todo depende de las circunstancias en las que se esté trabajando.

Posteriormente se aplica el proceso de reingeniería al sistema para que sea fiable, de fácil mantenimiento y que funcione eficientemente, dado que muchas veces los sistemas viejos fueron pensados para hardware y software, que estaban vigentes hace ya algunos años, por lo que su funcionamiento con los sistemas actuales puede resultar algunas veces conflictivo además de que el costo de mantenimiento puede ser, además de elevado, complicado de realizar, debido que las personas que construyeron los sistemas actualmente ya no se encuentran. Así el más importante de estos puntos es el hecho de que el sistema sea eficiente porque de lo contrario de nada serviría aplicar todo el proceso sin obtener esta característica.

También en la definición se dice que todo el proceso se debe realizar a partir de sistemas heredados extrayéndoles los conocimientos que permitan crear el nuevo sistema. Como ya se mencionó, los sistemas heredados son los que regularmente involucran aspectos críticos para una empresa. Por ello erróneamente se intenta postergar la aplicación del proceso de la reingeniería hasta que ya no se tiene otra solución. Para evitar esta situación se debe tener un amplio cuidado con estos sistemas, tomando en cuenta lo que realmente quieren las empresas, que el nuevo sistema siga haciendo lo mismo que el anterior, y no les importa cómo. En que esto se haga de la mejor manera posible es donde entra la tarea de los ingenieros en computación, y es por eso que a partir de los sistemas heredados se debe tomar lo que es de vital importancia en cuanto a su funcionalidad para que sean los

³ Brodie, Michel & Stonebraker; Michael "Migrating Legacy Systems: Gateways, Interfaces and Incremental Approach" Morgan Kaufmman. 1994

elementos a partir de los cuales se estructura el nuevo sistema. En lo que se refiere al trabajo que se realizó con del SAICOCETI básicamente sólo se tomaron los requerimientos del sistema previo así como algunos elementos de análisis, aplicando la reingeniería de directamente sobre la base de datos del sistema anterior, sin embargo se considera un elemento de gran relevancia en el proceso del desarrollo del nuevo sistema, razón por la cual se aborda el tema en el presente capítulo.

Finalmente, se menciona algo que va muy ligado con lo que se había mencionado antes, se espera que el nuevo sistema mejore al anterior con el fin de reemplazarlo, ya que de nada serviría tener un nuevo sistema construido con las últimas tecnologías si no se mejora la funcionalidad que tenía el sistema anterior.

4.4 REINGENIERÍA UN PROBLEMA DE INGENIERÍA

A lo largo de la investigación realizada para elaborar esta tesis, en algunos de los documentos había una referencia cuyo título resultó muy interesante “Reengineering: An Engineering Problem”, que traducido al español es el título que da nombre al presente apartado, es un documento elaborado por Peter H. Feiler en julio de 1993, que es una de las tantas publicaciones del “Software Engineering Institute” (SEI), en dicho documento se dice lo siguiente:

Mientras que las herramientas de reingeniería son útiles en algunos aspectos de reingeniería, la reingeniería no es más acerca de las herramientas de lo que es la ingeniería en sí. Tal y como la ingeniería implica un proceso disciplinado respaldado mediante los métodos de la ingeniería y herramientas automatizadas, la práctica de la reingeniería requiere de lo mismo. Es decir que la reingeniería es vista como un problema de ingeniería para el cual es necesario hacer un análisis cuantitativo del problema y empleando herramientas de la ingeniería.⁴

Con estas afirmaciones se quiere hacer notar que la manera en la que se debe resolver un proceso de reingeniería es la misma como se resolvería cualquier problema de ingeniería, así como a pesar de que actualmente ya se cuentan con herramientas especializadas para su empleo en la reingeniería, esto no quiere decir que esas herramientas sean las que definan la reingeniería. Sobre todo, también se indica que el procedimiento para la ingeniería debe ser el mismo que el que se emplea para resolver un problema de ingeniería, el cual debe incluir una metodología rigurosa auxiliándose de las herramientas disponibles. En conclusión, se puede decir que la reingeniería es un problema que debe ser abordado tal y como se aborda cualquier otro que se presenta en la ámbito de la ingeniería, Más adelante se detallará el cómo llevar a cabo ésto.

⁴ Feiler, Peter H. “Reengineering: An Engineering Problem”. Software Engineering Institute Pennsylvania E.U.A. 1993

4.4.1 SOFTWARE ENGINEERING INSTITUTE (SEI)

Ya en el punto anterior se hizo referencia al SEI y debido a que la importancia que ha tenido en el ámbito de la reingeniería de software es de gran relevancia se detallará un poco acerca del mismo.

El SEI fue fundado por el Congreso de los Estados Unidos en 1984 para desarrollar modelos de evaluación y mejora en el desarrollo de software, que dieran respuesta a los problemas que generaba al ejército estadounidense la programación e integración de los sub-sistemas de software en la construcción de complejos sistemas militares. EL SEI es financiado por el Departamento de Defensa de los Estados Unidos y administrado por la Universidad Carnegie Mellon, por lo tanto el SEI trabaja muy de cerca con organizaciones gubernamentales, pero también está en estrecho contacto con la industria y la academia para mejorar continuamente sus sistemas intensivos de software realizando las siguientes actividades principales:

1. Mantiene investigaciones orientadas a la exploración de posibles soluciones para los problemas de ingeniería de software.
2. Identifica y codifica soluciones de metodología y tecnología.
3. Evalúa y refina las soluciones a través de programas piloto que ayudan a la industria y al gobierno de Estados Unidos a resolver sus problemas.
4. Publica los métodos de los que mejores resultados ha obtenido.

El SEI tiene diversas áreas de trabajo las cuales incluyen: sistemas dinámicos, sistemas de producción en línea, supervivencia de sistemas enlazados en red y administración del proceso de ingeniería de software.

4.4.1.1 MODELO DE CAPACIDAD Y MADUREZ

A partir de su fundación, el SEI ha marcado la pauta para la realización de reingeniería inclusive traspasando el ámbito referente al software y creando modelos a seguir para la obtención de los objetivos que pretende alcanzar algún tipo de reingeniería, como es el caso del Modelo de Capacidad y Madurez o CMM (Capability Maturity Model), que es un modelo que permite realizar una valoración de los procesos que lleva a cabo una organización el cual fue desarrollado inicialmente para los procesos relativos al software.

A petición del gobierno de los Estados Unidos en 1986, el SEI elaboró la primera versión del modelo de madurez de procesos relativos al desarrollo de software, y ésta fue publicada en septiembre de 1987. Poco a poco se fue depurando el modelo hasta que llegó al CMM ó SW-CMM (CMM para software), en 1993 fue publicada la última versión de este modelo.

El modelo consiste en establecer un conjunto de actividades o procesos fundamentales englobados en lo que se denomina Áreas Clave de Proceso (KPA Key Process Area) y, por consecuencia, para cada área hay una serie de elementos fundamentales que se deben

respetar, ya que deben ser: definidas por un procedimiento documentado, provistas de los medios y formación necesarios, medidas, verificadas y ejecutadas en un modo sistemático, así como universal y uniforme. Es a partir de la efectividad en cada uno de los puntos mencionados que se le puede asignar a la organización una calificación denominada nivel de madurez y estos niveles se asignan de acuerdo a las siguientes condiciones:

1. Inicial. Las organizaciones a las que se les asigna este nivel no cuentan con una estabilidad en cuanto al ambiente para desarrollar y mantener el software, sin importar que se empleen técnicas correctas de ingeniería ya que hay una evidente falta de planeación y regularmente los proyectos exitosos se dan debido a esfuerzos personales, en la mayoría de los casos se presentan retrasos en las entregas así como gastos fuera del presupuesto inicial. El resultado de los proyectos que elaboran las organizaciones con este nivel es impredecible.

2. Repetible. Las organizaciones que poseen este nivel es porque cuentan con prácticas institucionalizadas para la gestión de sus proyectos, además de contar con elementos básicos para medir la efectividad, así como un seguimiento en lo que se refiere a la calidad. La forma de llevar la relación con los subcontratistas y clientes se hace a través de una gestión sistemática.

3. Definido. Aunado a la buena gestión de proyectos, las organizaciones a las que se les asigna este nivel es porque tienen procedimientos correctos para llevar a cabo la coordinación entre sus grupos, formación del personal, técnicas de ingeniería detalladas y un nivel alto en cuanto a la manera de medir los procesos. Algo común es que se implementa la técnica de revisión por pares, la cual consiste en que dos personas del mismo rango analicen el mismo objeto con la finalidad de comprobar su calidad.

4. Gestionado. Para obtener este nivel, las organizaciones disponen de un conjunto de métricas significativas, en lo que se refiere a la calidad y productividad, las cuales se toman sistemáticamente como referencia para tomar decisiones y gestionar los riesgos.

5. Optimizado. Las organizaciones con este nivel tienen como objetivo la mejora permanente de las actividades que realizan.

Las prácticas que son llevadas a cabo por cada área clave del proceso se dividen en cinco características comunes, las cuales permiten determinar qué tan efectivos, repetibles y duraderos son la implementación y la institucionalización de un proceso clave. Estas cinco características son: compromiso de la realización, capacidad de realización, actividades realizadas, mediciones de análisis y, finalmente, la verificación de la implementación.

En el año 2001 se presentó lo que se podría llamar la evolución del CMM que se denominó Capability Maturity Model Integration (CMMI) el cual difiere de su antecesor entre otras

cosas, porque el CCM permitía asignar niveles de madurez a cada área y el CMMI no, sino que lo maneja para toda la organización.

4.4.2 ETAPAS DE LA REINGENIERÍA

En este apartado se abordarán las diversas etapas del proceso de reingeniería. Sin embargo, antes de entrar de lleno al asunto es necesario aclarar la diferencia entre dos conceptos que con frecuencia se confunden, uno de ellos ya se ha definido claramente en cuanto a lo que es la reingeniería de programación, pero ahora se va a ver lo que es la ingeniería inversa: “es el proceso en el cual se analiza un sistema de software con el propósito de conocer su estructura y diseño a un nivel más alto de abstracción. Por ejemplo el proceso de extraer varios modelos de un sistema de software.”⁵

Entonces, como se puede observar en la definición anterior, la ingeniería inversa se relaciona de manera importante con la reingeniería de software, pero no es lo mismo, sino que es una herramienta muy recurrida en los procesos de reingeniería.

Habiendo dejado claro los conceptos, se analizará cómo se lleva a cabo regularmente un proceso de reingeniería tomando como marco conceptual el siguiente esquema:

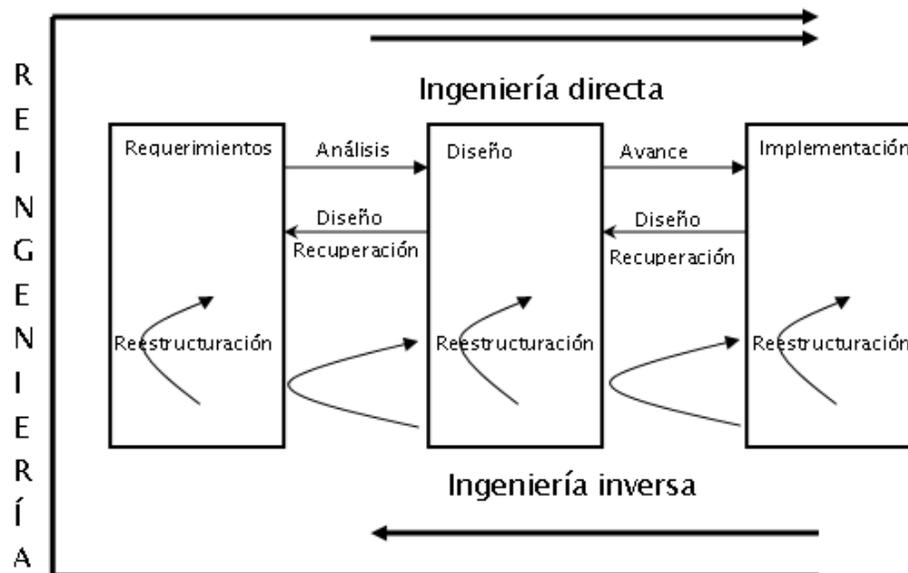


Figura 4.1. Diagrama conceptual de la reingeniería

En la Figura 4.1 es posible apreciar la comparación entre la realización de la ingeniería directa y la inversa, ya que para la ingeniería directa se parte de los requerimientos del sistema para después pasar a una etapa de diseño (con análisis previo) y, finalmente,

⁵ Feiler, Peter H. “Reengineering: An Engineering Problem” Software Engineering Institute Pennsylvania E.U.A. 1993

avanzar a la etapa de implementación. Pero para elegir cuándo se quiere aplicar ingeniería inversa se debe partir de la implantación del sistema, y a partir de ésta obtener su diseño y, finalmente, los requerimientos. Una vez que se ha pasado por esas etapas lo que hay que hacer es una reestructuración de cada uno de los elementos y aplicar el proceso de ingeniería directa una vez más, pero ahora sobre los requerimientos, análisis, diseño e implementación reestructurados de manera que esa es una visión general de cómo se puede abstraer el proceso de la reingeniería. Más adelante se detallará sobre algunas herramientas y métodos que ayudarán a este proceso. Ahora hay algo de gran importancia que se debe tener en cuenta es que la efectividad de la ingeniería es determinada de acuerdo a qué tan bien organizado se efectúa el manejo del proceso y es ahí donde entra la metodología como CMMI.

En el presente apartado se ha pretendido tomar la idea del documento del SEI: manejar la reingeniería como un problema de ingeniería y teniendo esa visión el hecho de que resolver un problema involucra que primero se debe entender. Es claro que si se comprenden las condiciones actuales de un sistema es más sencillo comprender en qué condiciones se quiere tener, de manera que sea posible elaborar un plan para llegar a cumplir los objetivos. En la siguiente figura se busca plantear de manera visual lo anterior:

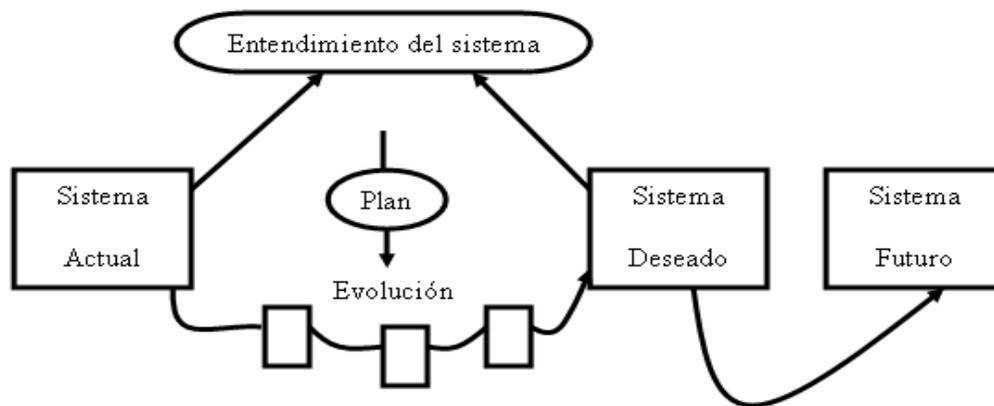


Figura 4.2. Entendimiento del sistema

El estado actual refleja las propiedades del sistema existente y el proceso mediante el cual dicho sistema fue “ingenierizado”, es decir, el cómo fue desarrollado y mantenido. Un subconjunto de esas propiedades actuales del sistema es indeseable dando la pauta del problema que debe ser resuelto.

El entendimiento del sistema muestra el proceso de creación y mantenimiento del mismo a través del análisis. La evolución del sistema representa la actividad ingenieril de la migración hacia el producto final que se desea. Basado en el entendimiento tanto actual del

sistema como del deseado, así como en posibilidades tecnológicas para aplicar a la reingeniería un análisis de los costos de la aplicación de la misma, dan como resultado un plan de reingeniería. Durante la ejecución de este plan, se debe considerar que en algunas ocasiones es necesario efectuar cambios en el mismo debido al surgimiento de nuevas tecnologías o posibles cambios en el presupuesto.

4.5 TIPOS DE REINGENIERÍA

Hasta el momento ya se ha hablado de lo que es la reingeniería, y de cómo se puede aplicar, ahora se muestra una descripción de los estilos más recurridos para cuando se aplica.

4.5.1 INGENIERÍA INVERSA

Ya en el apartado anterior se definió brevemente en qué consiste la ingeniería inversa, a estas alturas sólo queda agregar que se elige este método porque nunca se llegó a desarrollar una especificación, por lo tanto, lo que se necesita es extraer la información del diseño arquitectónico y de proceso e información de los datos a través de una representación del programa con un nivel de abstracción más elevado del código fuente.

4.5.2 REESTRUCTURACIÓN DE CÓDIGO

Este tipo de reingeniería es muy común. Se emplea cuando los sistemas heredados cuentan con una arquitectura lo suficientemente sólida como para mantenerla, sin embargo, presenta problemas debido a que la manera en la que fueron codificadas cada una de las partes que lo componen es difícil de entender, comprobar así como mantener, por lo tanto a lo que se procede es a reestructurar el código de los módulos que están dando problemas. La manera en cómo se efectúa este proceso es mediante algunas herramientas de reestructuración (de las cuales se detallará más en el siguiente apartado), que ayudan a detectar las violaciones en cuando a la forma de emplear las estructuras de la programación estructurada y regeneran el código de manera automática (también se puede hacer en forma manual), y posteriormente se revisa de nuevo el código para comprobar que no existan anomalía en cuanto a su funcionalidad.

4.5.3 REESTRUCTURACIÓN DE DATOS

Hay muchos casos en los cuales la arquitectura de los datos de un programa influye aún más que el propio código fuente dando como resultado que, cuando se posee una estructura de datos débil traerá como consecuencia que el sistema sea difícil de adaptar y mejorar. El proceso de la reestructuración de datos se realiza a un nivel mucho más alto de abstracción en comparación con el que se realiza en la reestructuración de código, generalmente va antecedida de la aplicación de la ingeniería inversa. Por lo que se debe analizar de manera minuciosa la arquitectura de datos original para que partir de ella se

definan los nuevos modelos de datos que se necesitan. En una aplicación con este tipo de reingeniería implica que se efectúen ya sea cambios en el código o en la arquitectura completa del sistema, debido a ya que la arquitectura de datos tiene una gran influencia sobre el programa.

4.5.4 INGENIERÍA DIRECTA

El empleo de este tipo de reingeniería va muy ligado al empleo de CASE (más adelante se detallará que son este tipo de herramientas) el cual actualmente provee de lo que se podría llamar un “motor de reingeniería” al que le se inserta un programa viejo, lo analiza o a veces reestructura, para finalmente efectuar algunas mejoras en cuanto a la calidad de software. Se le llama ingeniería directa debido a que todo se ve como el proceso normal para resolver cualquier otro problema de ingeniería. El problema que se tiene actualmente con uso de herramientas CASE es que regularmente solamente se puede aplicar sobre un conjunto reducido de tecnologías que van muy ligadas entre sí.

4.6 HERRAMIENTAS DE REINGENIERÍA

En algunos de los apartados anteriores de este capítulo se ha mencionado la existencia de que hay algunas herramientas que facilitan varias tareas, que son algo monótonas y repetitivas, de manera que permiten que el equipo de trabajo se concentre en las actividades que realmente merecen mayor atención. Una manera de clasificar este tipo de herramientas es la siguiente:

1. Herramientas de ingeniería inversa: se emplean generalmente para obtener especificaciones a partir del código fuente y generar los modelos gráficos de análisis y diseño de una manera estructurada, así como listas de utilización y algunos otros datos más de diseño.
2. Herramientas de reestructuración y análisis de código: con ayuda de este tipo de herramientas es posible analizar la sintaxis de un programa, de manera que se obtiene una gráfica de control de flujo y automáticamente se obtiene un programa estructurado.
3. Herramientas de reingeniería para sistemas en línea: se recurre a este tipo de herramientas para modificar sistemas de bases de datos en línea.

La mayoría de las herramientas mencionadas presentan restricciones en cuanto a que solamente son aplicables a lenguajes de programación específicos. Hay algo que se debe tener presente, ya que si bien las herramientas ayudan a ahorrar tiempo ésto no implica que los ingenieros ya no participen en el proceso, debido a que siempre hay que hacer revisiones de los productos que dan las herramientas.

La tendencia de hoy en día, en cuanto a la creación de herramientas de ingeniería inversa, es hacer uso de la inteligencia artificial para obtener mejores resultados.

4.6.1 HERRAMIENTAS CASE

Ya se mencionó anteriormente que una de las herramientas que se utilizan recurrentemente en la reingeniería son las del tipo Computer Aided Software Engineering (CASE), que se crearon con la intención de incrementar la productividad y el control de calidad en cualquier etapa de la construcción de un software.

El uso de herramientas CASE ha permitido: la aplicación práctica de metodologías, sencillez para la realización de prototipo y desarrollo de aplicaciones conjuntas, simplificar el mantenimiento del software, mejorar y estandarizar la documentación, aumentar la portabilidad de las aplicaciones y facilitar la reutilización de componentes de software. Todo lo anterior son elementos de gran ayuda que hoy en día son muy valorados en el desarrollo de los sistemas de software.

No hay un criterio definido mediante el cual se puedan clasificar las herramientas CASE. Se ha intentado realizar algunas clasificaciones de acuerdo a las plataformas que soportan, las fases del ciclo de vida del sistema que cubren, la arquitectura de las aplicaciones o bien, de acuerdo a su funcionalidad. Hoy en día este tipo de herramientas tienen una tendencia hacia dos tipos de integración: (1) mediante una integración de datos que permita disponer de las herramientas CASE como diferentes estructuras de diccionarios para el intercambio de datos o bien (2) mediante la integración de varias herramientas CASE de tal forma sean capaces invocar a otras de manera que interactúen entre sí.

CAPÍTULO 5: SAICOCETI EN PLATAFORMA MICROSOFT

En este capítulo se habla de los componentes que componen la versión anterior del SAICOCETI tales como: el lenguaje de programación, el servidor de bases de datos y el servidor Web; se presentan características generales de cada uno de estos elementos y su relación con el sistema, para finalmente hacer un análisis global de la versión anterior.

5.1 LENGUAJE DE PROGRAMACIÓN

El SAICOCETI inicial fue programado mediante el uso del lenguaje Visual Basic de Microsoft, ya sea de manera directa para los módulos de administración o bien con el empleo de los llamados Visual Scripts para los módulos Web mediante Active Server Pages (ASP).

Visual Basic fue desarrollado por Alan Cooper para la compañía Microsoft. Es un lenguaje de programación, un dialecto del lenguaje BASIC, principalmente, pero con muchas más prestaciones añadidas en comparación con éste. La primera versión de este lenguaje fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico de manera que el proceso de crear interfaces gráficas fuera sencillo y rápido.

5.1.1 ANTECEDENTES HISTÓRICOS DE VISUAL BASIC

El nacimiento de Visual Basic data desde el año 1964, cuando nació el lenguaje de programación Beginner's All purpose Symbolic Instruction Code, mejor conocido como BASIC. El propósito de la creación de esta herramienta fue ayudar al aprendizaje de los principiantes en el campo de la programación, de manera que les fuera posible crear programas con un lenguaje muy similar al que se utiliza en la vida diaria (en el idioma inglés), con instrucciones sumamente sencillas, intuitivas y escasas. Para el año de su nacimiento las características de este lenguaje cubrían todas las necesidades presentes en aquella época, ya que las computadoras estaban basadas en transistores, el número de ciclos de trabajo eran alrededor de 10,000 por cada segundo y la cantidad de almacenamiento de memoria era de sólo unos cuantos kilobytes.

Posteriormente, el BASIC sufrió de un fenómeno similar al que en un momento dado se vio inmerso el lenguaje C, ya que surgieron varias versiones del mismo lenguaje. Aunado al gran auge que por los años 70 tenían lenguajes como FORTRAN y COBOL hubo la necesidad de crear un estándar para que pudiera competir con estos lenguajes, y por ende, se creó la normativa del BASIC STANDARD.

Unos años después junto con la aparición de las computadoras personales comerciales en los años 80, el BASIC retomó su lugar como lenguaje pensado para principiantes. Con el auge de las PC, se liberaron algunas versiones más del BASIC, pero en lugar de ayudar a su popularidad estas versiones a lo único que contribuyeron fue a que se perdiera la confianza que tenían en él muchos programadores, ya que presentaba un gran número de desventajas, respecto a otros lenguajes de la época, como PASCAL, C, ó CLIPPER. Entre los claros inconvenientes que se presentaban los más relevantes son:

- No era un lenguaje estructurado: Probablemente, éste fue el mayor de sus defectos, ya que por el hecho de no ser un lenguaje estructurado, los códigos resultantes eran muy difíciles de mantener y depurar; al grado de que se generó un movimiento

mundial para terminar con la práctica de la programación no estructurada representada por el famoso “goto”.

- Falta de compiladores confiables: Ligado a un buen lenguaje de programación siempre debe haber un buen compilador, pues de nada sirve tener un buen lenguaje de programación si no se cuenta con una herramienta fiable que permita depurar errores así como obtener ejecutables que no provoquen problemas graves. Efectos que no ocurrían con el BASIC, con la evolución tecnológica se fueron incrementando las necesidades y BASIC resultó insuficiente para satisfacerlas.
- Falta de herramientas para intercambio de información: Algo muy importante que se presentó con la evolución de la tecnología en el ámbito de la computación fue la imperiosa interacción que debían tener muchos de los sistemas. De tal manera que, para cumplir sus cometidos, un programa ya no podía satisfacer las necesidades por sí mismo, sino que necesitaba del apoyo de más sistemas y el BASIC no tenía medios para apoyar estas tareas.
- Ausencia de librerías: Algo imprescindible en cualquier lenguaje de programación hoy en día es la ayuda que representan para el programador las librerías, ya que representan un gran ahorro de tiempo y esfuerzo en cuanto a lo que se refiere a la programación de varias tareas comunes. Y, como el BASIC no apoyaba en este aspecto a los programadores dejó de ser una opción viable.
- Incapacidad para programación de bajo nivel: Con el aumento en las capacidades de las computadoras ya no bastaba con tener un control sobre “la capa superficial”, sino que para explotar todas las nuevas características era necesario poder controlar las computadoras a nivel de registros, puertos físicos, etc. y los lenguajes que permitían esto fueron los que dominaron.

Debido a los motivos anteriores, y algunos otros más, los usuarios del BASIC lo abandonaron totalmente; al grado de que en una posterior versión denominada Quick-BASIC, en la cual se corregían muchos de los problemas que presentó en entregas anteriores, pasó prácticamente inadvertida de no ser por que Microsoft incluyó una pequeña versión de prueba en su sistema operativo MS-DOS. En esta versión ya se manejaba como un lenguaje estructurado que permitía crear programas modularmente y podía competir con lenguajes de alto nivel. Pero debido al sistema gráfico de Windows el uso de MS-DOS disminuyó notablemente, y por ende, la poca presencia que tenía el BASIC.

Debido a toda la serie de fracasos anteriormente citados se buscó darle un nuevo enfoque al BASIC aprovechando toda la nueva amplia gama de posibilidades que presentaba el sistema operativo Windows, tales como el intercambio de información, librerías, drivers, controladores, manejo de bases de datos, etc. Y fue así como surgió Visual Basic, el cual incluía un compilador con un entorno totalmente gráfico para el desarrollo de aplicaciones con interfaces gráficas de usuario.

La versión actual de Visual Basic se basa en la plataforma .NET y mantiene muchas de las características de anteriores versiones, a pesar de lo cual hay un gran número de diferencias que lo hacen incompatible con las entregas anteriores.

5.1.2 CARACTERÍSTICAS GENERALES DE VISUAL BASIC

Como ya se ha mencionado, uno de los principales motivos por los cuales muchas personas son partidarias del empleo de Visual Basic hoy en día, es por la gran sencillez que tiene en cuanto a su manejo y no se necesitan conocimientos muy profundos de programación para comenzar a trabajar.

Es un lenguaje de programación orientado a eventos, en el cual la programación del flujo del programa se planea en base a las acciones que el usuario lleva a cabo durante su ejecución. Para la facilidad de la creación de las interfaces gráficas de usuario Visual Basic se apoya en un motor centrado en el manejo de formularios. La innovación principal que introdujo, y posteriormente, algunos otros lenguajes adoptaron, fue la inclusión de una clase especial de dll (Biblioteca de vínculos dinámicos: un archivo que contiene funciones que se pueden llamar desde aplicaciones u otras dll).

Las dll fueron llamadas inicialmente vbx, y hoy en día conocidos como ocx que permiten contener toda la funcionalidad de un control y facilitar su rápida incorporación.

La sintaxis es una derivación del BASIC, y a través del tiempo se fueron incorporando sentencias necesarias para el correcto manejo de la programación estructurada, además cuenta con una limitada implementación para programar, orientada a objetos, ya que los formularios y controles se manejan como objetos y el empleo del polimorfismo mediante el uso de interfaces pero no admite la herencia.

No se requiere el uso de los apuntadores y brinda una manera muy sencilla para el manejo de las cadenas, característica básica en cualquier lenguaje de programación. Cuenta también con una gran variedad de bibliotecas que permiten el acceso a bases de datos por medio de Open DataBase Connectivity (ODBC).

El compilador incluido con el lenguaje crea archivos ejecutables que para funcionar necesitan una dll la cual provee de las funcionalidades implementadas en el lenguaje, aunadas a otras que facilitan el acceso a funciones del sistema operativo y la integración con otros sistemas.

5.1.3 VENTAJAS Y DESVENTAJAS DE VISUAL BASIC

Visual Basic ha sido empleado principalmente para la creación de aplicaciones relacionadas con la gestión de empresas, debido a la gran rapidez con la que se puede construir un programa que utilice una base de datos sencilla. Y, como ya se ha indicado, debido a su gran facilidad de manejo hay gran abundancia de programadores en este lenguaje. Pero cada vez más, los requerimientos para satisfacer las necesidades en cuanto a la tecnología

informática van aumentando de manera que la velocidad y potencia de los sistemas debe ser cada día mayor, y el proceso de mantenimiento de un sistema debe ser mejorado.

En relación con lo anterior, Visual Basic no ha salido muy bien librado, ya que ha sido descalificado por muchas grandes empresas, debido a que mencionan la falta estructuración de términos, ausencia de una real programación orientada a objetos, y qué decir de sus fallas repentinas en proyectos a gran escala provocadas por la gran cantidad de código que emplea, además de que los programas creados son muy lentos en relación a los que se elaboran en otros lenguajes de programación.

Por otro lado, un punto a favor de Visual Basic es la documentación que proporciona a través de MSDN, una herramienta completaría para los desarrolladores que proporciona código de ejemplo y documentación, no solamente de Visual Basic sino también para otros productos de Microsoft.

Otras de las ventajas de Visual Basic es que permite hacer un tratamiento de los mensajes que arroja el sistema operativo Windows, y gran parte del trabajo en cuanto al diseño de los formularios ya está realizado. Sin embargo, los controles personalizados no ayudan a mejorar la potencia de Windows.

El hecho de que sea un software, cuyo propietario es Microsoft, provoca que solamente las personas involucradas en la compañía decidan qué tanto evoluciona el lenguaje. Existe sólo un compilador y al mismo tiempo Integrated Development Environment (IDE); gran limitante, ya que en lugar de adaptar las herramientas a sus necesidades los programadores deben adaptar sus necesidades a la herramienta.

Otras desventajas con las que hay que lidiar cuando se trabaja con Visual Basic son que: no permite exportar el código a otras plataformas que no sean Windows, los ejecutables son bastantes lentos, la sintaxis es poco flexible, no permite un tratamiento de procesos, el manejo de errores no sigue los patrones estructurados, no incluye operadores a nivel de bits, no permite el manejo de memoria dinámica y la que probablemente pueda ser su mayor desventaja, en cuanto a otros lenguajes de programación, es que se debe pagar una licencia para usarlo.

5.1.4 USO DE VISUAL BASIC EN EL SAICOCETI

El lenguaje Visual Basic fue empleado para la construcción del módulo de administración del SAICOCETI, junto con otras tecnologías propias de Microsoft como ADO (Active Data Objects), una herramienta que permite realizar la comunicación con las bases de datos que se apoya en una interfaz de bajo nivel llamada OLE DB. Ambas tecnologías van muy ligadas ya que ADO necesita de OLE DB para poder funcionar. ADO es la base de la comunicación para el SAICOCETI con Sql Server.

Otros de los elementos básicos en el módulo de administración es ActiveX, el cual es una combinación de tecnología y herramientas orientadas a objetos, que se basa principalmente en el Modelo de Objeto Componente (COM por Component Object Model). Elementos a partir de los cuales se centralizó el control de muchas de las tareas del sistema, sobre todo, las que tienen que ver con la interacción con la base de datos. El problema que afecta al sistema es que para algunas pantallas se muestran acciones que no tienen ninguna funcionalidad para una pantalla dada. Además se emplean una serie de controles que fueron creados por compañías privadas por los cuales hay que pagar.

5.1.5 USO DE ASP EN EL SAICOCETI

En el inicio de la popularización de la World Wide Web, las páginas Web no eran capaces de otra cosa, más que de mostrar sitios estáticos en los cuales no había muchas posibilidades de mostrar vistosos diseños, y mucho menos pensar acerca de contenidos multimedia.

Lo anterior, poco a poco se fue volviendo un gran problema ya que para actualizar un sitio Web se requería un proceso largo y tedioso debido a la gran cantidad de páginas que regularmente los componían, por ello surgió la necesidad de idear alguna manera mediante la cual fuera posible agregar contenido dinámico a las páginas Web.

Para lograr que las páginas Web fueran dinámicas se utilizaron dos vertientes principales: una que trabaja del lado del cliente y la otra del lado del servidor. La primera vertiente fue el DHTML (HTML dinámico), con este tipo de páginas fue posible que las páginas alteraran su forma una vez cargadas en el navegador (lado del cliente). En contra parte del lado del servidor la tecnología que surgió en primera instancia fue CGI (Common Gateway Interface), esta tecnología permite al programador crear un programa en C, Pascal, Perl o Python, y mediante algunas sentencias específicas el programa es capaz de leer los contenidos enviados mediante los formularios HTML y procesarlos de forma que la presentación enviada a las computadoras clientes se adaptaba de acuerdo al entorno para cada cliente definido dinámicamente.

El problema de utilizar lenguajes como C o Pascal para generar contenido Web dinámico fue que estos lenguajes de programación no fueron creados para tal propósito, por lo que se presentó el inconveniente que ya mencionó en el apartado anterior de este capítulo: que en lugar de adaptar las herramientas a las necesidades, eran las necesidades las que se tenían que adaptar a las herramientas. Y así fueron surgiendo nuevas tecnologías y la propuesta de Microsoft fue ASP (Active Server Pages), la cual estuvo muy ligada al surgimiento del producto que presentó la misma compañía para trabajar como servidor Web IIS (Internet Information Server), del cual se tratarán detalles más adelante.

5.1.5.1 CARACTERÍSTICAS GENERALES DE ASP

ASP es una tecnología del lado del servidor, pero a diferencia de los CGI, ya no se trata de un programa almacenado en el servidor Web que atiende las peticiones realizadas por los clientes, sino que es una combinación de los códigos HTML y ASP (Visual Script).

La manera en la que trabaja ASP es la siguiente: cuando un cliente hace una petición a un servidor Web de ASP se ejecuta el código que contiene la página, posteriormente lo único que se envía al cliente es el código HTML resultado de la ejecución de las sentencias propias de ASP y así se consigue alterar el contenido antes de enviarlo, por lo que no se necesita ejecutar nada en el cliente.

5.1.5.2 VENTAJAS Y DESVENTAJAS DE ASP

Es muy sencillo entender cómo trabaja ASP, ya que al ser un derivado de Visual Basic es posible que desarrollares Web principiantes puedan crear sitios completos. Otra ventaja que se tiene es el hecho de que, al igual que Visual Basic, cuenta con la documentación MSDN de Microsoft y el entorno gráfico que se proporciona con el lenguaje.

También es posible compartir una gran variedad de recursos con Visual Basic, Visual C++, Visual J++, Delphi o cualquier otro lenguaje de programación que sea compatible con la tecnología COM de Microsoft, y por ende, migrar aplicaciones ya existentes a un entorno Web.

Una de las desventajas de esta tecnología es que está basada en Visual Basic; si bien le agrega muchas funcionalidades de ese lenguaje, también hereda los problemas que ya se citaron, y el hecho de ser un software propietario de Microsoft implica pagar una licencia para su uso, ya que si bien las páginas de ASP se pueden crear desde un bloc de notas, al menos se debe pagar lo que se refiere al servidor Web, el cual solamente se encuentra disponible para el sistema operativo Windows y por lo tanto no se puede migrar el código a otras plataformas. También si se quiere el editor gráfico, hay que pagar por el uso del IDE (Microsoft Interdev).

5.1.5.3 USO DE ASP EN EL SAICOCETI

En la versión del SAICOCETI Web se emplea ASP para agregar el dinamismo necesario para la presentación del sitio, así como las tareas básicas para validar el acceso de los usuarios: cambio de password, realización de las auditorías, recibir sugerencias de los auditores, mediante el empleo de conexión a bases de datos para obtener y almacenar información.

5.2 MANEJADOR DE BASE DE DATOS

SQL Server de Microsoft fue el manejador de base de datos empleado para realizar el manejo de la información de SAICOCETI, el cual es un sistema administrador de Bases de

Datos relacionales basadas en la arquitectura cliente–servidor. Este sistema como la mayoría de los productos de Microsoft se destaca por la gran facilidad en su uso, debido a la interfaz gráfica que proporciona, por lo que no se necesita tener conocimientos muy profundos acerca de la administración de las bases de datos.

5.2.1 ANTECEDENTES HISTÓRICOS DE SQL SERVER

Sql Server es un producto que ya lleva algunos años en el mercado comercializado bajo la firma de Microsoft, sin embargo, no es un producto original de esta compañía ya que realmente hasta la versión 6.5 era más bien un producto sublicenciado, porque en sí el responsable de la creación del producto fue SysBase (compañía con experiencia probada en el mercado).

Debido a que Microsoft no tenía la capacidad de construir un manejador de bases de datos, que pudiera competir con los que predominaban en aquella época, decidió comprar los derechos de uso de SysBase para poder lanzarlo bajo su marca, y el trabajo Microsoft se limitó a cambiar el aspecto gráfico de las herramientas tanto para los clientes como para el servidor.

Según el acuerdo que tenía Microsoft con SysBase si deseaba una modificación de fondo en el sistema debía hacer la petición a la gente de SysBase porque debido a los términos del contrato no tenía permitido realizar modificación alguna sobre el código del manejador. Por lo anterior, para la liberación de la versión 7, Microsoft decidió desligarse completamente de SysBase y manufacturar su propio producto sin necesidad de depender de otra compañía, pero en esta ocasión ya no partían de cero ya que a pesar de que se menciona que Sql Server 7 fue un producto totalmente nuevo en relación con su antecesor, se basaron mucho en la experiencia adquirida durante el periodo de sublicenciamiento para la elaboración del producto.

Muchos expertos en el tema mencionan que el corto tiempo en el que desarrollaron Sql Server 7, el cual era en realidad la primera versión del software, es una situación digna de resaltar, ya que dotaron al sistema de una robustez con la que no se contaba en las anteriores entregas. Para la siguiente entrega (Sql Server 2000) se agregaron aun más, lo que permitió que el producto pudiera competir en algunos ámbitos con Oracle (Empresa líder en este mercado).

Actualmente ya se comercializa la versión 2005, pero muchas empresas siguen trabajado con la 2000 debido a que resultó una herramienta muy confiable que hasta la fecha sigue atendiendo las necesidades de ciertos negocios.

5.2.2 CARACTERÍSTICAS GENERALES DE SQL SERVER

Sql Server es una herramienta fácil de utilizar para construir, administrar e implementar aplicaciones de negocios, lo cual trae consigo el hecho de que se pone a disposición un modelo de programación rápido y sencillo para los desarrolladores, eliminando la administración de la base de datos para operaciones estándar y a la vez proporcionando herramientas sencillas de manejar para operaciones regularmente complicadas.

Sql Server cuenta con: administración multiservidor o en una sola consola, ejecución y alertas de trabajo basadas en eventos, seguridad integrada, scripting administrativos, automatización de tareas rutinarias, también proporciona elementos importantes para el manejo de warehouse, que es una colección de datos en la cual se integra la información de una institución y que se emplea como soporte para la toma de decisiones.

5.2.3 VENTAJAS Y DESVENTAJAS DE SQL SERVER

Cuando se trata de hacer una comparación de Sql Server regularmente se realiza tomado como punto de referencia a Oracle, que es el líder en el mercado, y un aspecto cuestionado en Sql Server es la facilidad que se tiene en cuanto a su uso.

El problema con la situación anterior es que la configuración de la instalación se hace con características que no son las óptimas y por lo tanto se puede provocar que se presenten muchas fallas en ambientes de trabajo exigentes. Por lo tanto, por el hecho de que la instalación se realice de manera muy sencilla, en comparación con otras herramientas empresariales no implica que no se tenga que contratar a una persona especializada para poner a punto el sistema, lo cual es un error que recurrentemente se comete.

Si bien muchos de los productos Microsoft han sido muy criticados por su falta de confiabilidad, de acuerdo con las opiniones de muchos expertos, Sql Server 2000 es el mejor producto que ha presentado esta compañía, al grado que supera en algunos aspectos a Oracle, pero aun así para volúmenes elevados de información (terabytes) no ofrece el rendimiento y confiabilidad necesarios.

Entre las grandes desventajas que tiene, es el pago de licencias, ya que al ser un software cuyo propietario es Microsoft se deben pagar las licencias correspondientes, las cuales incluyen el considerar Windows Server, ya que si bien se puede instalar en otras versiones de Windows, el número de conexiones y el rendimiento para necesidades empresariales son muy limitadas. Por otro lado, al ser un sistema con arquitectura cliente-servidor se deben pagar las licencias para cubrir la instalación tanto del servidor como del número de clientes que se necesite.

Y de nuevo, la otra gran desventaja que presenta el Sql Server es el hecho de que solamente está disponible para la plataforma Windows, lo cual limita en gran medida cuando se necesita realizar una migración a otras plataformas.

5.2.4 USO DE SQL SERVER EN EL SAICOCETI

La manera en cómo está ligado el Sql Server con el SAICOCETI es que fue el manejador de bases de datos seleccionado para el sistema, debido a su fácil integración tanto con ASP como con Visual Basic se emplearon una serie de controles predeterminados, los cuales hicieron que no se tuviera que entrar en detalles en cuanto a la programación de las tareas propias del manejo de la base de datos. El sistema fue construido empleando Sql Server 7 aunque también se hicieron algunas pruebas con Sql Server 2000 sin tener complicación alguna en cuanto al funcionamiento de la base de datos.

5.3 SERVIDOR WEB

Como se mencionó anteriormente, el servidor Web empleado para el SAICOCETI fue el Internet Information Services (IIS), el cual no es solamente un servidor Web sino que es un sistema que brinda un conjunto de servicios para el sistema operativo Windows. Este servicio permite que una computadora se convierta en un servidor de Internet o Intranet, lo que facilita la publicación de páginas Web, ya sea a nivel local o remota. El servidor Web se basa en varios módulos que le dan capacidad para procesar distintos tipos de contenidos como ASP o inclusive como PHP o Perl.

5.3.1 ANTECEDENTES HISTÓRICOS DE IIS

La primera aparición del IIS fue con Windows NT 3.51 ya que se agregó como parte de las actualizaciones para el sistema operativo. Fue una versión que recibió una gran cantidad de críticas debidas al sin número de problemas que provocaba en comparación con sus competidores directos como los "Enterprise Servers" de Netscape y el servidor Apache.

Posteriormente, la versión 2.0 fue incluida con Windows NT 4.0 y la 3.0 se puso a disposición por medio de una actualización para el mismo sistema operativo.

En 1998 la versión 4.0 se puso a disposición en el option pack del NT 4.0 y a partir de la versión 5.0 ya fue incluido con el Windows 2000 y después la versión 5.1 en Windows XP Professional, así como la versión 6.0 en Windows Server 2003 y XP Profesionalx64 Edition. La última versión es la versión 7.0 la cual ésta disponible con Windows Vista y Windows Server "Longhorn".

5.3.2 CARACTERÍSTICAS GENERALES DE IIS

Con el paso del tiempo Microsoft ha mejorado sustancialmente al IIS, los avances han sido sobre todo en seguridad y rendimiento. Las mejoras realizadas en seguridad están orientadas al aprovechamiento de nuevas tecnologías de cifrado y autenticación mediante certificados del cliente y el servidor.

Una de las maneras en que se asegura la información es a través de Secure Socket Layer (SSL), pues proporciona un método para transferir datos entre el cliente y el servidor de una manera segura, además de que también es posible realizar una comprobación del cliente antes de permitirle que inicie una sesión como usuario de los servicios del servidor. También cuenta con autenticación mediante servidores proxy o servidores de seguridad que permiten comprobar a los usuarios de una manera confiable.

A partir de la versión 5.0 también es posible impedir que los usuarios tengan acceso a recursos para los cuales no tienen derechos, ésto a partir de una lista restricciones asociadas a una dirección IP específica.

La administración es muy sencilla ya que se realiza por medio de los asistentes de seguridad, así también se pueden definir permisos de acceso en directorios virtuales e inclusive sobre los archivos.

5.3.3 VENTAJAS Y DESVENTAJAS DE IIS

Una de las ventajas principales es que, para la administración del sistema, no es necesario tener conocimientos avanzados, ya que todo se realiza mediante la consola gráfica incorporada en el mismo, aunque en las últimas versiones ya se incluyen algunos detalles de la configuración en archivos de tipo XML, para resolver más fácilmente los problemas que se presenten con el servidor, aunque en este caso si es necesario contar con un mayor grado de conocimientos técnicos, ya que de lo contrario no se comprenderá en contenido de los archivos.

Con el paso del tiempo se han ido mejorando problemas de versiones anteriores del sistema y uno de los detalles notables de la versión 6.0 es que puede detectar automáticamente las pérdidas de memoria y diversos errores, así como reiniciar procesos cuando sea necesario, mientras va colocando en espera otras solicitudes. También se pueden definir grupos de aplicaciones para ejecutar ciertos sitios Web independientes de otros, de manera que si uno falla los demás siguen funcionando.

Otra ventaja que presenta el sistema es el BITS (servicio de transferencia inteligente en segundo plano), mediante el cual es posible realizar la transferencia de grandes cantidades de información sin disminuir el rendimiento de la red, lo anterior se logra enviando pequeños paquetes de datos empleando el ancho de banda no utilizado en el momento de la transferencia.

Entre las grandes desventajas que se tienen al emplear IIS, es el hecho de que sólo se puede utilizar sobre la plataforma Windows y, por ende, hay que realizar el pago de las licencias correspondientes. Algunas personas consideran que IIS es prácticamente gratis debido a que ya viene incluido con Windows, pero entonces hay que tener en cuenta que se debe pagar al menos licencia del sistema operativo Windows que lo contenga.

Por otro lado, a lo largo de todas sus versiones, IIS ha presentado grandes problemas de seguridad, sobretodo, en lo que se refiere a los agujeros que se han quedado descuidados en el sistema. En el año 2001 este problema fue muy grave, al grado de que se recomendó que los sistemas que trabajaran en IIS fueran migrados a otro tipo de tecnología, debido a que un gusano llamado "Nimda", provocó que la gente de Microsoft reescribiera completamente el IIS para su versión 6.0.

Aunado a lo anterior se puede indicar que un año antes se detectó un problema que pudo haber tenido resultados muy severos. El problema fue que resultó propenso a permitir que se incrustaran archivos en el servidor mediante la cadena: "../"; en peticiones Web especialmente construidas era posible escalar directorios en el servidor inclusive hasta acceder carpetas que no eran de contenido Web, el problema anterior se presentaba tanto para el servicio Web como el de ftp.

Muchos de los problemas que ha tenido a lo largo de todas sus versiones se han corregido, pero cada vez surgen más. El detalle es que sería un producto mucho más eficiente si en lugar de preocuparse por resolver los problemas hubiera un esfuerzo mayor en evitarlos.

5.2.4 USO DE IIS EN EL SAICOCETI

IIS está ligado con el SAICOCETI ya que emplea ASP. Por otro lado, si bien se mencionó que es posible crear las páginas ASP desde un bloc de notas en caso de que se quiera emplear el entorno gráfico incluido en la suite, de Microsoft Visual Studio (Visual InterDev para realizar la creación y depuraciones de los proyectos ASP) necesariamente se debe tener instalado el IIS ya que de lo contrario no será posible trabajar.

5.4 ANÁLISIS GLOBAL DEL SAICOCETI

Si bien el SAICOCETI cumple con las reglas del negocio establecidas para su construcción, queda bastante limitado, por el hecho de que si en momento dado se pretende cambiar la plataforma sobre la que trabaja es imposible, ya que al estar completamente basado en tecnologías Microsoft, no hay compatibilidad con otras plataformas de manera que el sistema ésta atado a la plataforma Microsoft.

Por otro lado, se tiene el problema del pago de licencias ya que al emplearse software, cuyo propietario es Microsoft, es necesario pagar por su uso y se deben considerar los gastos que implican desde la etapa de construcción hasta la de producción, pues se debe pagar por el Visual Basic, el Visual InterDev y por un sistema operativo Windows en el cual se puedan instalar las aplicaciones anteriores, así como IIS tanto para depuración como producción, y qué decir del Sql Server para el cual se deben pagar por separados los derechos de uso del servidor y del cliente.

Otro punto a tomar en cuenta es que como primera versión del sistema presenta algunas fallas en su funcionamiento, además de que han surgido nuevas funcionalidades que son necesarias incluir en el mismo.

Por lo tanto, se decidió realizar una actualización del sistema, así como una reestructuración para que sea posible agregarle portabilidad.

CAPÍTULO 6: SAICOCETI EN SOFTWARE LIBRE

En este capítulo se abordan los temas relacionados con la elección de las tecnologías empleadas para realizar la migración del SAICOCETI a software libre tales como: lenguaje de programación, manejador de bases de datos y servidor Web, así como otras tecnologías empleadas en el nuevo sistema.

6.1 LENGUAJE DE PROGRAMACIÓN

Debido a que el principal objetivo del presente trabajo de tesis es realizar el cambio de plataforma de un sistema a software libre, la selección de los componentes utilizados se hizo cuidando el cumplimiento de este fin, dando como resultado que el lenguaje de programación elegido fue Java, el cual es un producto de Sun Microsystems, ya que cumple con los requisitos que se desean para lograr cumplir los objetivos de esta tesis, dado que es un lenguaje por el cual no se debe pagar licencia alguna para emplearlo, además de que tanto la confiabilidad como portabilidad está garantizada debido la manera en como está estructurado.

6.1.1 ANTECEDENTES HISTÓRICOS DE JAVA

La historia de los inicios de Java se remonta a diciembre de 1990 cuando Patrick Naughton, un ingeniero de Sun Microsystems, presentaba su renuncia a esta empresa, pero como respuesta lo que recibió fue una propuesta por parte de Sun para unirse a un grupo de trabajo encargado del desarrollo de una nueva tecnología, el grupo al cual lo invitaron iba a ser autónomo de la línea directiva de Sun, además de que se mantendría en secreto, las características del proyecto le interesaron a Naughton.

Posteriormente en 1991 se formó el grupo de trabajo cuyos integrantes fueron: Bill Joy, Andy Bechtolsheim, Wayne Rosing, Mike Sheridan, James Gosling y Patrick Naughton, y llegaron a la conclusión que de acuerdo a las perspectivas que tenían sobre las tendencias para el futuro, se presentaría una evolución de los aparatos electrodomésticos, de tal forma que su objetivo se estableció en la creación de un lenguaje de programación que permitiera interactuar con electrodomésticos de manera práctica y sencilla, fue así como nació lo que se denominó proyecto Stealth, que posteriormente fue renombrado como proyecto Green y fue puesto en marcha en febrero de 1991.

Como propuesta inicial, el equipo de trabajo del proyecto Green, consideró emplear como base el lenguaje C++ y realizarle algunas modificaciones de manera que se adaptara a sus necesidades, pero debido a que el lenguaje presentaba muchos inconvenientes para lo que tenían en mente se inclinaron por desarrollar un lenguaje desde cero, el cual recibió inicialmente el nombre de "Oak". Los objetivos que se fijaron para el desarrollo del nuevo lenguaje eran: que fuera independiente de la plataforma (1) y por ello se optó por un lenguaje interpretado, el lenguaje debía ser robusto y al mismo tiempo sencillo (2) de manera que ayudara a disminuir los errores de los programadores y una de las soluciones que encontraron fue eliminar la herencia múltiple, lo anterior dio como resultado un lenguaje que tenía grandes similitudes con C y C++ pero no estaba ligado a un hardware específico. Debido a que ya existía en el mercado un producto llamado "Oak" se vieron forzados a cambiar el nombre a "Java".

Para agosto de 1991 ya se tenían los primeros programas escritos en el nuevo lenguaje y se dedicaron a crear un prototipo llamado “Star 7” el cual consistía en un dispositivo parecido a una PDA, cuyo objetivo era permitir responder una llamada telefónica desde cualquier lugar; en septiembre de 1992 realizaron una demostración de sus avances incluyendo un sistema operativo y algunas librerías, y a partir de la demostración Sun decidió fundar una empresa filial llamada “FirstPerson” con el fin de comercializar la nueva tecnología.

Pero lamentablemente para Sun, no encontraron un verdadero plan de negocio, ya que a las empresas a las que les presentaron su iniciativa no les resultaba redituable incluirla en sus productos debido a que aumentaba los costos de una manera muy importante. Por lo tanto el personal de SUN se dio a la tarea de buscar nuevas alternativas para poder colocar su producto de manera exitosa en el mercado, y aprovecharon el auge que por aquella época (junio de 1994) comenzaba a tener Internet debido al aumento de sitios Web pero sobre todo la creación del navegador Mosaic que permitió acceder a los sitios de manera gráfica y tomando esa idea crearon un navegador llamado “HotJava”.

En mayo del año 1995, se anunció que la versión alpha de Java, que para ese entonces solamente corría en sistemas Solaris, iba a ser agregada al navegador de Netscape, que en ese momento era el que más se utilizaba. Posteriormente con la segunda versión alpha en julio del mismo año ya contaba con soporte para Windows 95 de manera que en enero 1996 Sun creó JavaSoft para desarrollar de lleno nueva la tecnología dando como resultado que en el mismo mes apareciera la versión 1.0 del Java Development Kit (JDK).

6.1.2 CARACTERÍSTICAS GENERALES DE JAVA

Hoy en día Java es considerado por muchos expertos como el lenguaje más avanzado, debido a que permite la implementación de sistemas orientados a objetos de manera sencilla con un entorno robusto, proporciona una sofisticada librería con elementos que ahorran gran cantidad de tiempo, además creación de aplicaciones Web de manera sencilla. El concepto de “objeto” que maneja Java es bastante sencillo, se tiene facilidad de ampliarlo, y también se brinda la posibilidad de seguir manejando los llamados elementos primitivos como son los números enteros, números reales, caracteres y algunos otros tipos de datos simples.

Un elemento muy destacado es que a pesar de su simpleza se logró una riqueza semántica muy destacada ya que aunque cada tarea se puede implementar con un número muy limitado de opciones, el programador tiene a su disposición un gran potencial en cuanto a lo que se refiere al manejo de las expresiones.

Java verifica el código al momento de compilarse, pero también realiza una revisión al momento de llevar a cabo la ejecución ayudando a que se corrijan los errores en la etapa adecuada, debido a que la sintaxis para el manejo de los tipos de datos y las declaraciones

es bastante restringida, es posible detectar una gran cantidad de errores en tiempo de compilación.

En lo que se refiere al manejo de memoria, Java se encarga de liberar automáticamente los bloques de memoria que ya no son ocupados por el usuario, auxiliándose del llamado colector de basura (garbage colector), que hace las operaciones correspondientes cuando un espacio en memoria ya no ésta siendo referenciado por ninguna variable sea liberado y se ponga a disposición del sistema, de manera que el programador ya no tiene que ocuparse por liberar explícitamente los registros de memoria solicitados dinámicamente.

El manejo de los errores es completamente orientado a objetos, a través de las llamadas “excepciones”, lo cual tiene como consecuencia que se haga un tratamiento más sencillo y natural de los mismos. También Java proporciona conjuntos de clases llamados “paquetes” que contienen abstracciones básicas para facilitar la implementación de los programas bastante complejos; entre los paquetes más utilizados están los que ayudan a la creación de las interfaces gráficas de usuario, que permiten crear elementos gráficos para diversas plataformas sin necesidad de cambiar el código.

Algo que ha hecho que los programadores se inclinen al empleo de Java es la gran facilidad para aprender el lenguaje, ya que el único requisito realmente indispensable, es conocer la teoría de la programación orientada a objetos y con un poco de tiempo es suficiente conocer las características básicas propias del lenguaje.

Un elemento importante que pone a Java por encima de otros lenguajes es la amplia colección por la que está conformada la Application Programming Interface (API) de Java la cual contiene paquetes que son de mucha ayuda para simplificar tareas comunes. Entre las ventajas que proporciona el API de Java son los conjuntos de clases que permiten establecer de manera sencilla conexiones de red entre computadoras o bien crear aplicaciones multiproceso, sin necesidad de depender directamente del sistema operativo sobre el que se esté trabajando.

Java fue diseñado para que los programas puedan ser ejecutados en forma correcta independiente de la plataforma en la que fueron creados, ya que utiliza compilación en una representación intermedia que recibe el nombre de códigos byte, que pueden ser interpretados por cualquier sistema operativo que cuente con la máquina virtual de Java, pero el hecho de necesitar un intérprete para ejecutarse hace que el tiempo necesario para correr un programa sea más lento que en otros lenguajes.

En lo que se refiere a la seguridad los diseñadores de Java se propusieron cuidar varios aspectos, desde restringir el manejo de la memoria controlando la asignación de los recursos de memoria y no dejándolos al programador, hasta la introducción de utilerías que permiten manejar encriptación, así como también considerando algunas restricciones para que no puedan ser creados de manera sencilla virus u otros elementos perjudiciales para los sistemas de cómputo.

6.1.3 MÁQUINA VIRTUAL DE JAVA

Java es un lenguaje que se dice es compilado y la vez interpretado debido a que en la creación de los programas intervienen los dos procesos. Se tienen lenguajes que son totalmente interpretados (como por ejemplo JavaScript), estos lenguajes poseen las cualidades de ser altamente portables y seguros, pero son menos eficientes que los que se ejecutan por medio de una máquina virtual, ya que para realizar la ejecución de programas totalmente interpretados se debe realizar el análisis léxico y sintáctico en tiempo de ejecución, dando como resultado una sobrecarga del sistema, llegando al grado de que la diferencia de tiempo al ejecutar una tarea entre un lenguaje completamente interpretado y uno compilado llega a ser inmensamente superior. Si bien los lenguajes que emplean una máquina virtual no llegan a ser tan rápidos como los netamente compilados, si disminuyen de manera importante la diferencia, esto se debe a que como ya se mencionó, usan una arquitectura de código intermedio, de manera que la ejecución del programa es dividida en dos fases: en la primera se realiza el análisis léxico y sintáctico del código fuente para generar el programa en instrucciones del procesador virtual y la segunda fase en la cual a partir del código intermedio generado en la etapa anterior la máquina virtual realiza las tareas necesarias para que se ejecute sobre la arquitectura (de hardware y software) que se está empleando.

La máquina virtual de Java (MVJ) es el núcleo del lenguaje, y por lo tanto no es posible ejecutar un programa en Java sin haber primero ejecutado alguna implementación de la MVJ, ya que en ella se encuentra el motor que realmente realiza la ejecución del programa convirtiéndose en la clave para la eficiencia, portabilidad y seguridad que proporciona Java, de manera que la ejecución de los programas se realizan sobre un procesador virtual.

La representación de los códigos de instrucción Java (bytecode) es simbólica de manera que los desplazamientos de los índices de los métodos que contiene un programa no son constantes sino que son cadenas de caracteres o nombres simbólicos que son resueltos la primera vez que se ejecuta el método, el nombre simbólico del método se busca dentro del archivo de clases (.class) y se determina el valor numérico del desplazamiento para su ejecución, este valor es guardado de manera que en los futuros accesos al método el tiempo requerido es menor, permitiendo introducir un nuevo método o sobrescribir uno existente en tiempo de ejecución, sin afectar o romper la estructura del código.

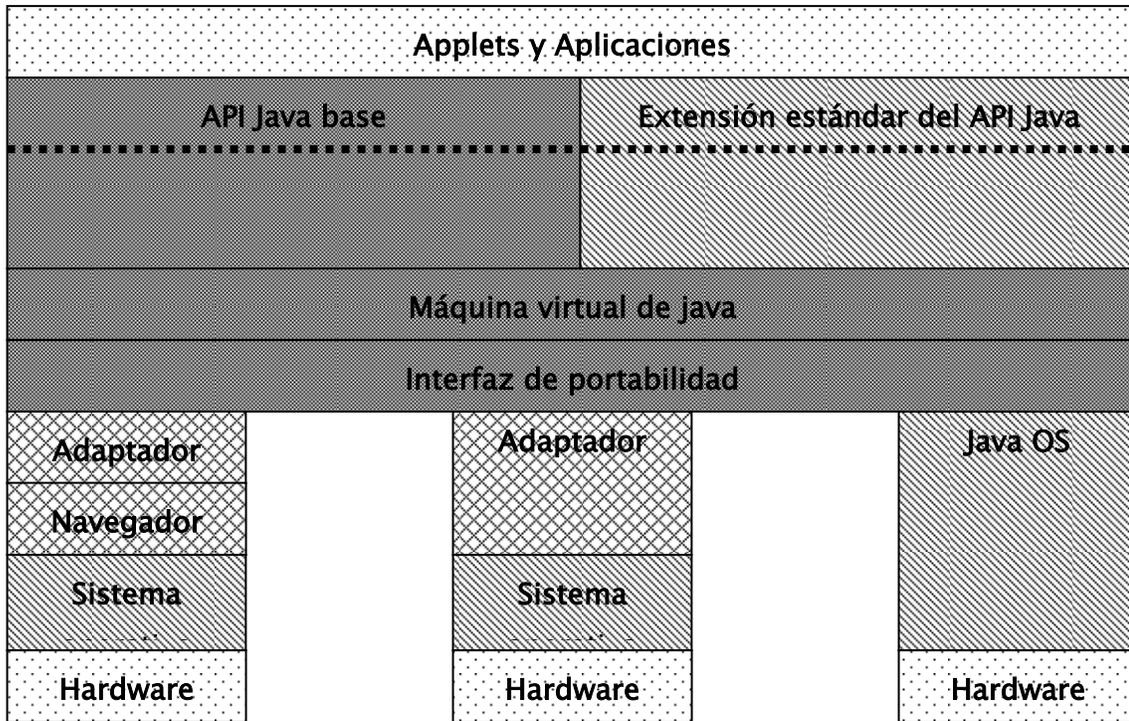


Figura 5.1. Estructura de Java.

Como se puede apreciar en la Figura 5.1 la capa de software que implementa la máquina virtual de Java oculta los detalles inherentes a la plataforma, ahora bien para que la plataforma de Java sea implementada sobre una amplia gama de sistemas operativos y procesadores, se implementaron dos capas de software, la primera es dependiente de la plataforma la cual funciona como adaptador, mientras que la segunda es independiente de la plataforma y se llama interfaz de portabilidad, de manera que la única parte que se tiene que crear cuando se construye una MVJ para una nueva plataforma es el adaptador. La MVJ se apoya en el sistema operativo para hacer el manejo de ventanas, red, sistema de archivos entre otros elementos.

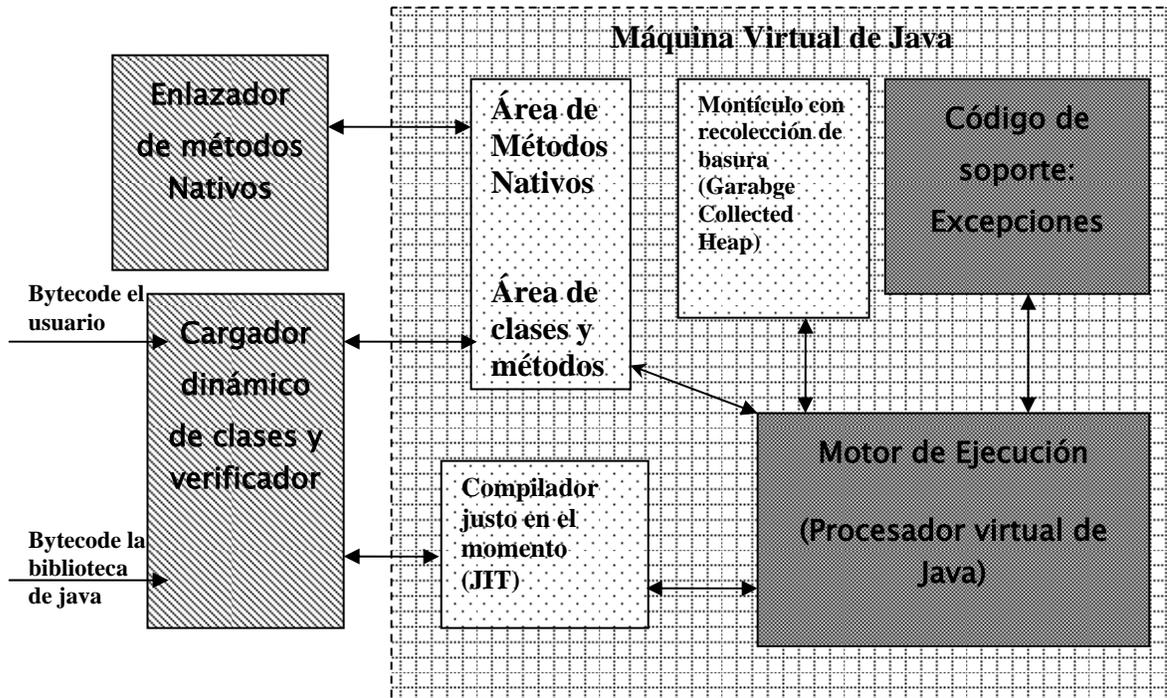


Figura 5.2. Sistema de Java en tiempo de ejecución.

SUN Microsystems utiliza el término de “Máquina Virtual de Java”, para referirse a la especificación abstracta de una máquina de software para ejecutar programas escritos en Java. La especificación de esta máquina virtual, define detalles tales como el formato de los archivos de clases o la semántica de las instrucciones que componen la máquina virtual, las implementaciones de esta especificación se le conoce como MVJ. Los elementos típicos de un sistema en tiempo de ejecución son los mostrados en la Figura 5.2, ejemplos comunes de sistemas de ejecución son los navegadores Netscape, Internet Explorer ó bien el programa Java que se proporciona con el kit de desarrollo de Java.

6.1.3.1 MOTOR DE EJECUCIÓN

El motor de ejecución es la parte del hardware o software encargada de ejecutar las instrucciones contenidas en los códigos de operación (bytecodes) que implementan los métodos de Java. Las primeras versiones de la MVJ consistían simplemente de un intérprete de códigos de operación, en las versiones más recientes se emplea una tecnología conocida como generación de código, justo en el momento (Just in Time code generation) que se encarga de convertir el código de operación a código nativo, el cual se ejecuta directamente sobre el hardware, este código nativo se genera únicamente la primera vez que se ejecuta el código de operación Java, dando como resultado que en las posteriores ejecuciones de los programas la velocidad de respuesta sea mayor.

6.1.3.2 VERIFICADOR DE JAVA

Como ya se ha mencionado anteriormente una las características más relevantes de Java es la seguridad que brinda, ya que al momento de llevar a cabo la ejecución de un programa justo después de que se obtiene una clase, la MVJ se encarga de verificar la estructura del archivo, que todos los registros tengan la longitud correcta, pero sobre todo se asegura de que cada uno de los códigos de operación sean seguros de ejecutar.

6.1.3.3 SOPORTE PARA MÉTODOS NATIVOS

Los programas en Java pueden invocar métodos que no estén implementados en lenguaje Java, contenidos en código nativo, por ejemplo almacenados en DLL de Windows o las bibliotecas compartidas SO de Solaris, pero la MVJ es capaz de incluir el código necesario para cargar dinámicamente y ejecutar el código nativo. Una vez que se enlaza el módulo que contiene el código nativo, el procesador virtual atrapa las llamadas a éste y se encarga de invocarlo. Este proceso también incluye la modificación de los argumentos de la llamada, para adecuarlos al formato que requiere el código nativo, además de transferirle el control de la ejecución. Cuando el código nativo termina, el módulo de soporte para métodos nativos se encarga de recuperar los resultados y de adecuarlos al formato de la máquina virtual Java. De manera análoga, el módulo de soporte para código nativo se encarga de canalizar una llamada hecha desde un procedimiento o método nativo, a un método escrito en Java.

6.1.3.4 CARGADOR DE CLASES

Los programas Java están completamente estructurados en clases. Por lo tanto, una función muy importante en tiempo de ejecución, es cargar, enlazar e inicializar clases dinámicamente, de manera que sea posible instalar componentes de software en tiempo de ejecución. El proceso de cargado de las clases se realiza hasta que haya una petición para una clase específica.

La MVJ utiliza dos mecanismos para cargar las clases. Uno consiste en un cargador de clases del sistema, cuya función es cargar todas las clases estándar de Java, así como la clase cuyo nombre se ingresa vía la línea de comandos. Es importante hacer notar que el cargador de clases es uno de los recursos que protege el administrador de seguridad. Los cargadores especificados por los programadores, pueden definir la localización remota de donde se cargarán las clases o asignar atributos de seguridad apropiados para sus aplicaciones particulares.

6.1.4 JAVA ENTERPRICE EDITION (JEE)

Hasta el momento se han tratado las características que componen la estructura general de Java, pero no se ha comentado el hecho de que Java se puede manejar en varios tipos de “presentaciones”, una de ellas es la versión estándar de Java conocida como JSE, que es lo

que se ha abarcado hasta el momento, pero no es suficiente para atender todas las necesidades de los requerimientos de una aplicación tipo empresarial, para la cual ya es necesario tener módulos Web, o establecer comunicaciones con las bases de datos, para ello se tiene que recurrir a otra “presentación” de Java, la tecnología Java 2 Enterprise Edition (J2EE).

J2EE define el estándar para desarrollar aplicaciones empresariales multicapas simplificándolas al basarlas en componentes modulares estandarizados, proporcionarles un conjunto de servicios a esos componentes y manejar varios detalles del comportamiento de la aplicación automáticamente sin una programación complicada.

J2EE agrega soporte total para componentes Enterprise JavaBeans, el API de Java Servlets, Java Server Pages y la tecnología XML, lo cuales son elementos básicos para la construcción de aplicaciones Web. El estándar J2EE incluye una especificación completa y pruebas de compatibilidad para asegurar la portabilidad de las aplicaciones a través de una amplia variedad de sistemas empresariales existentes capaces de soportar J2EE.

6.2 MANEJADOR DE BASE DE DATOS

Actualmente en el mundo del software libre hay dos manejadores de bases de datos que son los que predominan por sus características: MySQL y PostgreSQL. Después de hacer un balance fue seleccionado el segundo ya que además de brindar mayores ventajas sobre MySQL se adapta de una mejor manera a necesidades que se tienen que cubrir para el SAICOCETI.

6.2.1 ANTECEDENTES HISTÓRICOS DE POSTGRESQL

PostgreSQL en un inicio fue llamado simplemente Postgres, fue desarrollado en la universidad de California en Berkeley, por un profesor de ciencias llamado Michael Stonebraker. La aparición de Postgres representó una gran aportación en el mundo de la informática ya que brindó la posibilidad de salirse de la clásica estructura del modelo relacional para la implementación de las bases de datos y proporcionó la alternativa de emplear un modelo “objeto-relacional” en el cual además de tener todas las ventajas del modelo relacional también se puede combinar con el paradigma orientado a objetos, lo que lo hace más versátil que otros manejadores de bases de datos.

Durante ocho años hubo un proceso muy activo para el desarrollo de Postgres, en ese tiempo se le fueron añadiendo muchas características que hoy en día lo ponen al nivel de los líderes en el mercado, como son: adición de reglas, procedimientos almacenados, tipos de datos extensibles con índices y el ya mencionado concepto objeto-relacional; poco tiempo después fue comercializado y se transformó en “Illustra” el cual fue comprado posteriormente por Informix e integrado a su servidor.

En 1995 dos estudiantes de Stonebraker, Andrew Yu y Jolly Chen, remplazaron el lenguaje de consulta POSTQUEL propio de Postgres a otro que consistía en un subconjunto de sentencias del estándar SQL y renombraron el sistema como Postgres95. En 1996 Postgres95 se separó de la academia y tomó su propio camino hacia el mundo del código abierto gracias a un grupo de desarrolladores externos a Berkeley que vislumbraron el futuro promisorio del sistema y decidieron continuar con el desarrollo por su cuenta y lo transformaron de una manera radical, en cinco años lograron obtener consistencia y uniformidad al código base, crearon exámenes detallados de regresión para asegurarse de la calidad, repararon una gran cantidad de errores, agregaron nuevas características, y redondearon el sistema llenando grandes vacíos que había en lo que se refiere a documentación tanto para desarrolladores como para usuarios.

El fruto de su labor fue un nuevo manejador de base de datos que con el paso del tiempo se ha ganado una muy buena reputación. Con la inclusión al mundo del código abierto, así como con la adición de las nuevas características, el sistema adquirió su nombre actual PostgreSQL.

En solamente cuatro años el sistema fue mejorado prácticamente en todos los aspectos, gracias a que al ser de código abierto recibió aportaciones de miles de desarrolladores alrededor del mundo, de las mejoras más destacadas se tienen las siguientes:

- Control de concurrencia multiversión. Consiste en remplazar la metodología del bloqueo de niveles de las tablas por un sofisticado sistema de control de concurrencia multiversión, el cual permite que se atiendan peticiones de la lectura de la información consistente mientras se realiza escritura, permitiendo además hacer respaldos mientras el servidor se encuentra activo.
- Características SQL. Se incluyeron muchas de las bondades que provee SQL incluyendo subconsultas, valores defaults, constrains, llaves primarias y foráneas, alias para identificación, coherencia de las literales de cadenas, conversión de datos, etc.
- Mejora de los tipos de datos predefinidos. Se incluyeron un amplio rango de nuevos tipos de datos incluyendo una amplia gama para el manejo de fechas y tiempos así como los tipos de datos geométricos.
- Velocidad. El tiempo de respuesta se mejoró entre 20 y 40 por ciento, y el tiempo necesario para iniciar el servidor se redujo en un orden del 80 por ciento.

En los siguientes cuatro años posteriores a su aparición, se entregaron de la versión 7.0 a la 7.4 a las cuales les fueron agregando elementos que hicieron al sistema un producto cada vez más atractivo, de manera que poco a poco las grandes corporaciones comenzaron a considerar a PostgreSQL como parte de proyectos cruciales de manera que realizaron contribuciones a su desarrollo, dando origen a la versión 8.0 y para la versión 8.1 fue integrada por Sun Microsystems en su sistema operativo. Actualmente muchas organizaciones así como agencias gubernamentales de U.S.A emplean PostgreSQL, y de

hecho se dice que hoy en día es muy difícil encontrar una agencia gubernamental que no lo use por lo menos uno de sus departamentos.

6.2.2 CARACTERÍSTICAS GENERALES DE POSTGRESQL

PostgreSQL ofrece grandes ventajas sobre muchos otros manejadores de base de datos ya que al no pagar una licencia para su uso, se puede hacer uso libre del sistema, además de que al ser de código abierto se puede adaptar a las necesidades específicas de cada negocio. Soporta operadores, trabaja con la especificación SQL99 e incluye características avanzadas tales como joins y un elemento básico para los manejadores de bases de datos que es la integridad referencial.

PostgreSQL permite extender su funcionalidad ya que es posible emplear varios lenguajes de programación para la creación de procedimientos almacenados, además de que actualmente se tienen un gran número de interfaces para interactuar con él mismo como: Object Pascal, Python, Perl, PHP, ODBC, Java, Ruby, TCL, C/C++ y Pike.

La característica de PostgreSQL conocida como Write Ahead Logging incrementa la dependencia al registro de cambios antes de que éstos sean escritos en la base de datos; Ello garantiza que en el caso de que el servidor se caiga, existirá un registro de las transacciones a partir del cual es posible restaurar la base de datos, lo que representa un gran beneficio en el caso de fallas, ya que cualesquiera de los cambios que no fueron escritos en la base de datos pueden ser recuperados usando la información que fue previamente registrada. Una vez que el sistema ha quedado restaurado, un usuario puede continuar trabajando desde el punto en que lo dejó antes de la falla.

En general, muchos especialistas en el tema han concluido que PostgreSQL es un magnífico manejador de bases de datos, capaz de competir con muchos manejadores comerciales.

6.3 SERVIDOR WEB

La selección del servidor Web para el SAICOCETI no fue para nada difícil ya que a pesar de que hoy en día se tienen disponibles varias opciones de servidores Web, el que mejor cumple con todas las características y propósitos del presente trabajo de tesis es, el servidor Apache Tomcat, ya que es un sistema de software libre que proporciona el soporte necesario para trabajar con los diversos elementos de la versión empresarial de Java, además de ser muy confiable.

6.3.1 ANTECEDENTES HISTÓRICOS DE TOMCAT

La historia de Tomcat se remonta a la implementación de referencia de la API Java Servlet de Sun, ya que no se iba a publicar una especificación a menos que tuviera una implementación de referencia, la cual es un software en el que se basan las empresas para crear prototipos de aplicaciones y poner a prueba los conceptos y la utilización de una nueva tecnología. Sun

desarrolló la implementación de referencia inicial de la API Java Servlet y la presentó con el nombre de Java Servlet Web Development Kit (JSWDK).

Pero ocurrió que en octubre de 1999, el desarrollador de JSWDK convenció a Sun de que donara el código fuente de JSWDK a la Apache Software Foundation. Dos meses más tarde veía la luz Tomcat 3.0. En Apache Software Foundation se percataron de las grandes posibilidades de ese código para desarrollar productos de código libre con la misma calidad de productos comerciales y crearon el Proyecto Jakarta. Es decir que Jakarta nace del desarrollo de soporte para las tecnologías Java orientadas al mundo Web, como Java Servlets y JSP que fueron la respuesta de Sun a los lenguajes y plataformas de desarrollo como CGI-Perl, ColdFusion, ASP, PHP, etc. que, junto a los motores de base de datos, permiten la creación de aplicaciones Web. Sun ofreció estas respuestas basadas en Java con las ventajas de permitir el desarrollo de aplicaciones Web utilizando orientación a objetos más la oportunidad de separar las capas lógicas de la de despliegue y una fácil adaptabilidad a los distintos motores de base de datos utilizando JDBC como capa intermedia entre las aplicaciones Web y los datos.

Todo esto sucedía mientras Apache ya era conocida y muy utilizada debido a su servidor Web “Apache”, que estaba creciendo muy rápido en el mercado gracias a su política Open Source y a que tenía un muy buen rendimiento en el sistema operativo Linux.

Por lo tanto, la gente de Apache y la comunidad que la ocupaba vio la necesidad de integrar las soluciones Servlet y JSP al servidor Web por lo que nació JServ, que era un módulo para Apache y que, agregándose, se podían ejecutar aplicaciones programadas en esos lenguajes.

Pero debido a que Java es un lenguaje orientado a objetos, se estaba viendo que muchas soluciones se repetían en las aplicaciones para Internet, por lo que aprovechando el poder la orientación a objetos, la popularidad de Java dentro de los programadores y a la reutilización de código se llegó a la decisión de crear un proyecto independiente, donde se pudieran integrar todas estas soluciones fácilmente aumentando el poder y la rapidez en el desarrollo de aplicaciones. La idea era mantener un orden ya que esto mejoraba la calidad de las aplicaciones, permitían una mejor manutención y escalabilidad. Las licencias garantizaban acceso al código fuente y economizaban las soluciones, siendo ambas grandes ventajas comparadas, por ejemplo, a las soluciones provistas por Microsoft.

Actualmente se cuenta con cinco versiones diferentes de Tomcat que van de la versión 3.0 a la 6.0 y sus principales características son las siguientes:

- Tomcat 3.x (distribución inicial). Implementado a partir de las especificaciones Servlet 2.2 y JSP 1.1.
- Tomcat 4.x. Implementado a partir de las especificaciones Servlet 2.3 y JSP 1.2, que contaba con Java Management Extensions (JMX), JSP Y administración basada en Struts.

- Tomcat 5.x. Implementado a partir de las especificaciones Servlet 2.4 y JSP 2.0, que incluyó una capa envolvente nativa para Windows y Unix ayudando a la integración de las plataformas.
- Tomcat 6.x. Implementado a partir de las especificaciones Servlet 2.5 y JSP 2.1

6.3.2 CARACTERÍSTICAS GENERALES DE TOMCAT

En las primeras versiones de Tomcat se decía que el servidor era recomendable solamente para aplicaciones básicas y sitios Web con poco tráfico, pero conforme se han ido entregado las nuevas versiones esta afirmación ha cambiado radicalmente de manera que hoy en día se tienen enormes sitios Web que están alojando potentes aplicaciones sobre un servidor Tomcat.

Aunque Tomcat no es directamente compatible con todas las API de J2EE (como JDBC, JNDI, JavaMail, RMI, JMS, XML y EJB), todas esas API, con la notable excepción de EJB, están disponibles agregando archivos Java (JAR) que se pueden obtener gratuitamente. El hecho de que Tomcat no es compatible con EJB, se puede solucionar mediante el empleo de JBoss, un servidor EJB de código libre que se integra muy bien con Tomcat.

Las grandes ventajas que tiene Tomcat son: es completamente gratis, fácil de instalar, es posible ejecutarlo en máquinas pequeñas, es compatible con las API más recientes de Java, ocupa muy poco espacio y dado que el número de usuarios es bastante grande, siempre habrá alguien capaz de arreglar lo que para los demás ser un error muy complejo. Dicho de otra forma, la solidez de Tomcat se basa en que miles de desarrolladores contribuyen con código.

6.4 OTROS ELEMENTOS DEL SISTEMA

Con el propósito de aumentar el rendimiento del sistema, así como para reducir los tiempos de desarrollo, se emplearon una serie de herramientas adicionales a las que brinda el lenguaje Java, como son: un pool de conexiones y los frameworks Ibatis, Abator así como ltext.

6.4.1 FRAMEWORKS

Un framework en contexto de la programación es un conjunto de aplicaciones que ayudan a realizar tareas específicas de una manera sencilla sin la necesidad de saber a detalle cómo se realizan sino más que nada saber cómo se tiene acceso a la ejecución de las tareas.

Un framework se puede ver como una abstracción de una máquina, cuya construcción fue basada en ciclos de experiencia, para resolver un problema en un contexto, a diferencia de un “patrón” que es para resolver un problema en cualquier contexto, es decir que un framework condensa experiencia. De manera que solamente es necesario a limitarse a

instanciar esa máquina (framework) pero no se tiene la necesidad inmediata de conocer cada detalle de ella.

La ventaja que ofrecen los frameworks es que además de ser posible conocer una nueva arquitectura y poder copiar o aprender cómo funciona una “máquina”, el tiempo de desarrollo se disminuye, ya que simplifican en gran escala el trabajo. Alguien que tiene que completar un proyecto rápidamente y quizás en un lenguaje que no conoce, podría fácilmente aprender un framework además del lenguaje y centrarse en la solución del problema del negocio. Esto no implica que no sea posible llevar a cabo adaptaciones o extensiones a esa máquina, inclusive cambiar los elementos que la componen, pero lo anterior es posible llevarlo a cabo cuando ya se tiene a la máquina funcionando y se han realizado los ciclos de experiencia con el framework.

No obstante hay que tener en cuenta el costo y riesgo de que instanciar algo abstracto es mucho más grande a medida que se incrementan los grados de abstracción. Por ejemplo, clonar es mucho más económico que crear un objeto, utilizar un objeto de una clase concreta es más económico que uno de una clase más abstracta, construir un objeto a partir de una clase es mucho más económico que emplear un framework y así sucesivamente. Es por eso que usar una máquina construida por un tercero obliga a tener que ajustarse a ciertas reglas.

Actualmente debido a la gran popularidad y aceptación que ha tenido el lenguaje Java existen una gran cantidad de frameworks relacionados con el lenguaje, con los que es posible apoyarse para realizar tareas que en un momento dado podrían parecer muy complejas o bien ahorrar tiempo en su desarrollo así como aumentar la eficiencia.

Para tener un mejor control sobre la construcción y funcionamiento del SAICOCETI, se empleó el modelo de capas, este modelo a grandes rasgos lo que indica es que hay que tener separado el sistema en 3 capas:

- Capa de presentación. Con la que tiene interacción directa el usuario y engloba interfaces gráficas de usuario, páginas Web, etc.
- Capa de domino. Encargada de llevar a cabo toda la lógica del negocio.
- Capa de persistencia. Encargada de llevar a cabo todo el manejo del almacenamiento de la información.

En este aspecto para el desarrollo del sistema se empleó un framework llamado “Ibatis”, el cual es capaz de realizar todas las tareas relacionadas con la capa de persistencia, llevando a cabo toda la comunicación necesaria con la base de datos mediante la configuración de archivos XML, que en la mayoría de los casos van muy ligados a la estructura de la base de datos que se emplea y para simplificar la creación de estos archivos de configuración se utilizó otro framework llamado “Abator”, el cual es capaz de que a partir de las tablas que componen la base de datos, crear de manera automática casi todos los archivos de

configuración XML, así como el código fuente asociado a la capa de persistencia y algunos elementos muy útiles para la capa de dominio.

Finalmente uno de los elementos importantes de cualquier sistema es la elaboración y presentación de reportes y este requisito fue cumplido con otro framework llamado "itext" que permite la elaboración de documentos pdf con ayuda de Java.

6.4.2 POOL DE CONEXIONES

En la generalidad de los casos las aplicaciones de software consumen mayor tiempo en los accesos a base de datos, por lo tanto la comunicación con las bases de datos es un punto crucial que se debe tener en cuenta de tal manera que se logre un mejor rendimiento del sistema. Es por eso que para el desarrollo del SAICOCETI se emplea un "Pool" de conexiones a la base datos, de manera que no es necesario conectarnos y desconectarnos cada vez que se realice una operación en la base de datos.

El esquema clásico de JDBC implica que se realice una conexión a la base de datos por cada operación, repitiéndose el esquema conexión-operación-desconexión una y otra vez. Si bien esta forma de trabajar es válida, resulta ineficiente, ya que se están desperdiciando ciclos de ejecución en cada conexión y desconexión. La manera de solucionar esto es crear un número determinado de conexiones que se mantengan siempre activas y cuando se tenga que llevar a cabo alguna operación en la base de datos simplemente se invoca a una conexión disponible. Cuando un objeto Java requiere una conexión, la fuente de datos establece comunicación con el Pool de conexiones y éste le entrega una conexión lógica. Esta conexión lógica finalmente es recibida por el objeto Java y cuando desea cerrar la conexión, la fuente de datos establece de nuevo comunicación con el Pool de conexiones y devuelve la conexión lógica.

Hay varias maneras de configurar un Pool de conexiones, ya que se pueden hacer aplicaciones propias que hagan el manejo de las conexiones anteriormente descrito o bien se puede emplear alguno de los múltiples frameworks que se tienen disponibles hoy en día para no preocuparse por los detalles, es por ello que se aprovechó otras de las ventajas que brinda el framework "Ibatis", y se configuró de manera que el SAICOCETI se comunica con la base de datos empleando las ventajas que brinda el usar un Pool de conexiones.

CAPÍTULO 7: MANUAL TÉCNICO

En este capítulo se presentan los detalles de la implementación del SAICOCETI.



ESTUDIO DE FACTIBILIDAD

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA01-SAL-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV: SAICOCEI HOJA: 1 DE: 1

1. Definición de objetivos.

- Se debe realizar la migración del SAICOCEI en plataforma Microsoft a software libre.
- Se deben realizar mejoras por definir al sistema
- Se debe registrar el sistema con derechos de autor a la facultad de ingeniería de la UNAM

2. Determinación de recursos.

2.1. Factibilidad técnica.

Se cuenta con dos servidores adquiridos con apoyo de la DGAPA: uno para desarrollo del sistema y otro para la puesta en producción del mismo los cuales cumplen con los requerimientos necesarios para llevar a cabo el proyecto, así como con la capacidad de respuesta necesaria para pruebas y crecimiento del sistema.

2.2. Factibilidad económica.

El costo del desarrollo completo del sistema es cero debido a que será producto de un trabajo de tesis.

El costo de software es cero debido a que se utilizará software libre.

El costo hardware adicional es cero debido a que no se requiere más equipo del adquirido con apoyo de la DGAPA.

Los costos de papelería (hojas blancas, impresora, etc.) se cubrirán con el apoyo económico proporcionado por la DGAPA por la tanto no se emplearan recursos extra de la facultad de ingeniería de la UNAM.

2.3. Factibilidad operacional.

Con la implementación del SAICOCEI en software libre se reducirán los costos de operación del sistema debido a que no hay necesidad de pagar licencia alguna por software adicional.

Se contará con un sistema amigable y de fácil uso para la administración y control de las auditorías y de los participantes en los diversos niveles.

3. Ventajas y desventajas de la implantación del sistema.

La ventaja principal es que no se deben realizar costos adicionales debido a que el desarrollo se llevará a cabo mediante un trabajo de tesis. Se contará con un sistema en software libre registrado con derechos para la facultad de ingeniería.

4. Conclusiones.

Se concluye la migración del SAICOCEI se llevará a cabo debido a todas las ventajas que proporciona.

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo

FECHA: 10/12/2007

CALENDARIO DE TRABAJO



ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA02-SAL-01
 Proyecto: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV: SAICOCETI HOJA: 3 DE: 7

FECHA INICIO: 08/09/2006 FECHA TÉRMINO: 16/11/2007 TIEMPO TOTAL PROGRAMADO: 32 SEMANAS (X) TIEMPO TOTAL REAL: 62

DESCRIPCIÓN DE LA ACTIVIDAD	RESPON.	TIEMPO	TIEMPO EN: DÍAS () SEMANAS (X)													
			4	4	4	4	4	4	4	4	4	5	5	5	5	
Para sugerir nuevas preguntas a las auditorías ya existentes se debe preguntar a que auditoría se desea realizar la sugerencia así como el número de preguntas a sugerir (se 1 a 16) una vez realizado lo anterior, se mostrará un par de textboxés por cada nueva pregunta a sugerir, en una columna corresponderá a el punto a verificar en campo y la otra a como realizar la observación, una vez que el auditor termine de proporcionar sus sugerencias deben de enviarse al servidor.	José Ángel Haro Juarez	ESTIMADO	4	4	4	4	4	4	4	4	4	4	5	5	5	5
		REAL	0	1	2	3	4	5	6	7	8	9	0	1	2	3
Para proponer un nuevo tipo de auditoría se debe solicitar el nombre de la nueva auditor así como una explicación de la misma una vez que se han proporcionado, se deben enviar al servidor.	José Ángel Haro Juarez	ESTIMADO														
		REAL														

OBSERVACIONES: Los tiempos se elevaron debido al aumento de requerimientos

ELABORÓ: Rolando Lima Maciel AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 25/11/2007 FECHA: 10/12/2007



CALENDARIO DE TRABAJO

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA02-SAL-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV: SAICOCETI HOJA: 6 DE: 7

FECHA INICIO: 08/09/2006 FECHA TÉRMINO: 16/11/2007 TIEMPO TOTAL PROGRAMADO: 32 SEMANAS (X) TIEMPO TOTAL REAL: 62

DESCRIPCIÓN DE LA ACTIVIDAD	RESPON.	TIEMPO	TIEMPO EN: DÍAS () SEMANAS (X)																	
			1	2	3	4	5	6	7	8	9	10								
Debe ser posible registrar auditorías que deben tener un número único de identificación, nombre así como las preguntas que contiene, cada una de las preguntas debe tener un número, sugerencia y si esta activa o no a partir de los datos anteriores debe ser posible agregar, borrar o cambiar los datos de una auditoría así como realizar búsqueda a partir de las auditorías. También debe ser posible imprimir los datos de la pantalla.	Rolando Lima Maciel	ESTIMADO	1	1	1	1	1	1	1	1	1	2								
		REAL	0	1	2	3	4	5	6	7	8	9	0							
Se debe poder hacer un registro de auditorías por realizar la cual debe tener la información relacionada con el nombre de la empresa a la que se realizará la auditoría el CTI, el tipo de auditoría, fecha de inicio, fecha de término, clave para seguimiento vía Internet a si como un password, tiempo y costo, una sección para comentarios generales, también se debe poder seleccionar los auditores que participaran en la auditoría a partir de los datos anteriores debe ser posible agregar.	Rolando Lima Maciel	ESTIMADO																		
		REAL																		

OBSERVACIONES: Los tiempos se elevaron debido al aumento de requerimientos

ELABORÓ: Rolando Lima Maciel AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 25/11/2007 FECHA: 10/12/2007

RELACIÓN DE DOCUMENTOS



ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA03-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 1

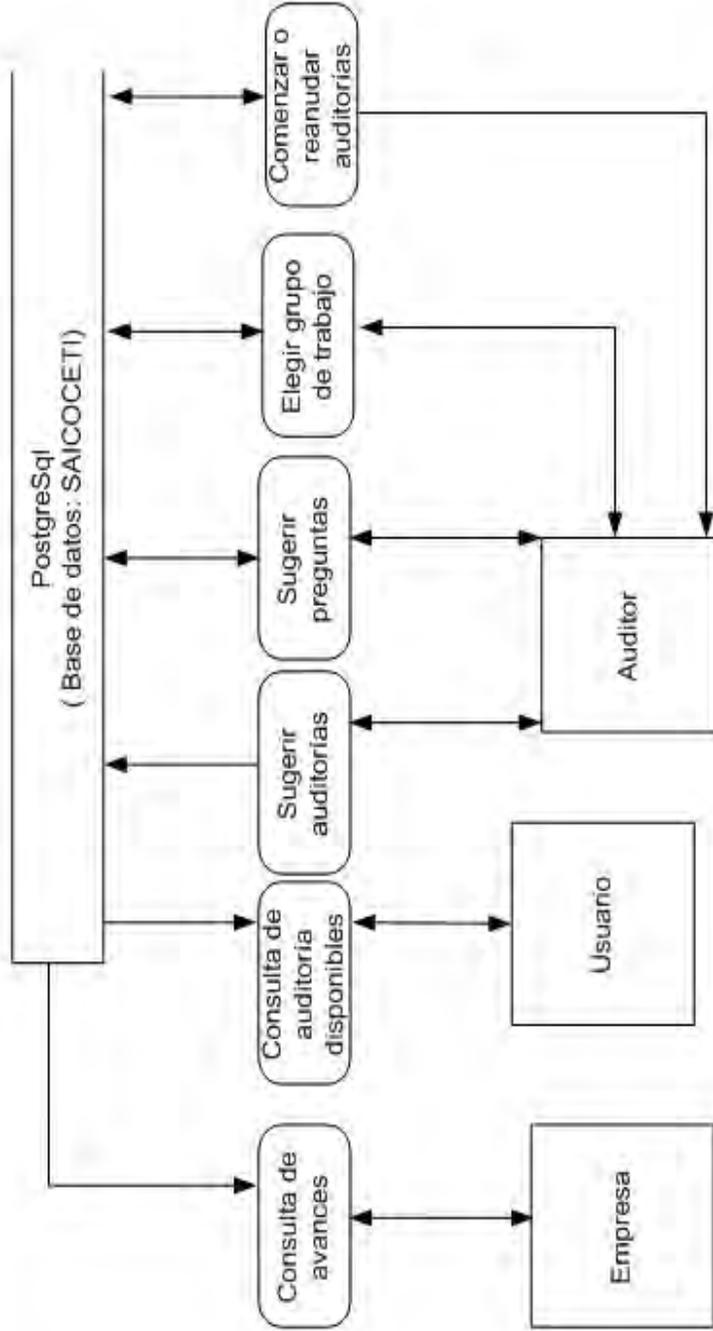
NOMBRE	FUENTE QUE ORIGINA EL DOCUMENTO	DESCRIPCIÓN DEL DOCUMENTO	FECHA EN QUE SE RECOPIÓ
DA01	Análisis	Estudio de factibilidad	03/12/2007
DA02	Análisis	Calendario de trabajo	03/12/2007
DA03	Análisis	Relación de documentos	03/12/2007
DADMF04	Análisis	Diagrama de flujo de datos	03/12/2007
DA05	Análisis	Diccionario de términos	03/12/2007
DADME06	Análisis	Diagrama de clases	03/12/2007
DADME07	Análisis	Diccionario de clases	03/12/2007
DADMD08	Análisis	Diagrama de seguimiento de sucesos	03/12/2007
DADMD09	Análisis	Diagrama de estados	03/12/2007
DA10	Análisis	Acta de acuerdos	03/12/2007
DD01	Diseño	Selección e identificación de software y hardware	03/12/2007
DD02	Diseño	Diagrama entidad-relación	03/12/2007
DD03b	Diseño	Registro de llaves secundarias	03/12/2007
DD04	Diseño	Diccionario de datos	03/12/2007
DD05	Diseño	Descripción de procesos	03/12/2007
DD06	Diseño	Entradas y salidas	03/12/2007
DD07	Diseño	Diagrama estructural del sistema	03/12/2007
DP01	Programación	Descripción de programas y rutinas	03/12/2007
DI01	Pruebas	Resultado de pruebas	03/12/2007
DI02	Liberación	Integración al banco de sistemas institucionales	03/12/2007

ELABORÓ: Rolando Lima Maciel AUTORIZÓ: Heriberto Olguin Romo

FECHA: 25/11/2007 FECHA: 10/12/2007

DIAGRAMA DE FLUJO DE DATOS

ÁREA: Análisis CVE: PROYECTO: IN103007 CVE. REFERENCIA: DADMF04-SAL-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 4



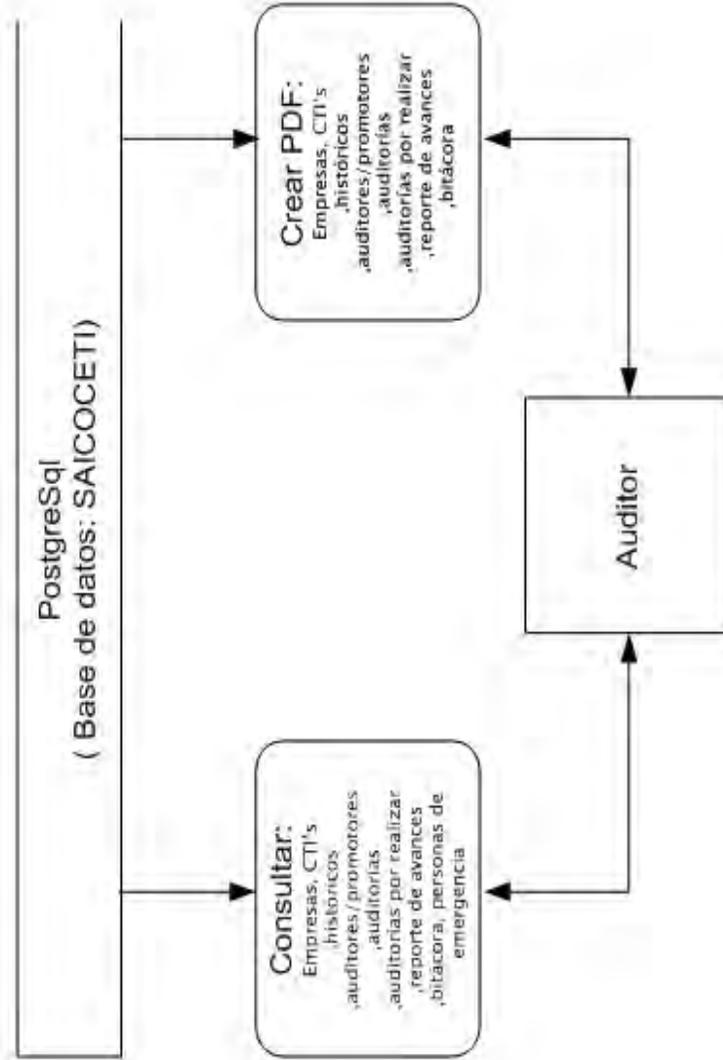
ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007



DIAGRAMA DE FLUJO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DADMF04-SAI-01
 PROYECTO Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 2 DE: 4



ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007

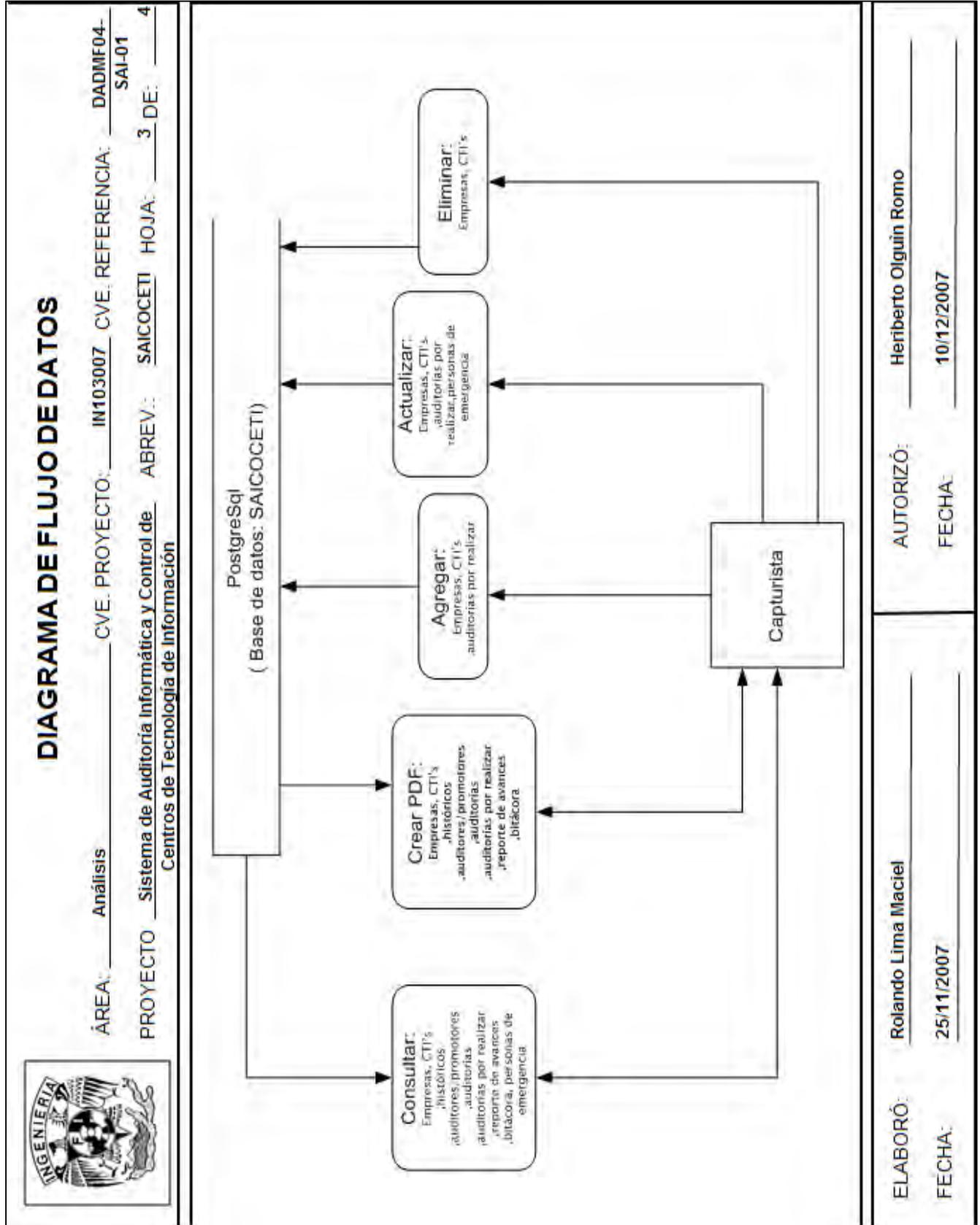
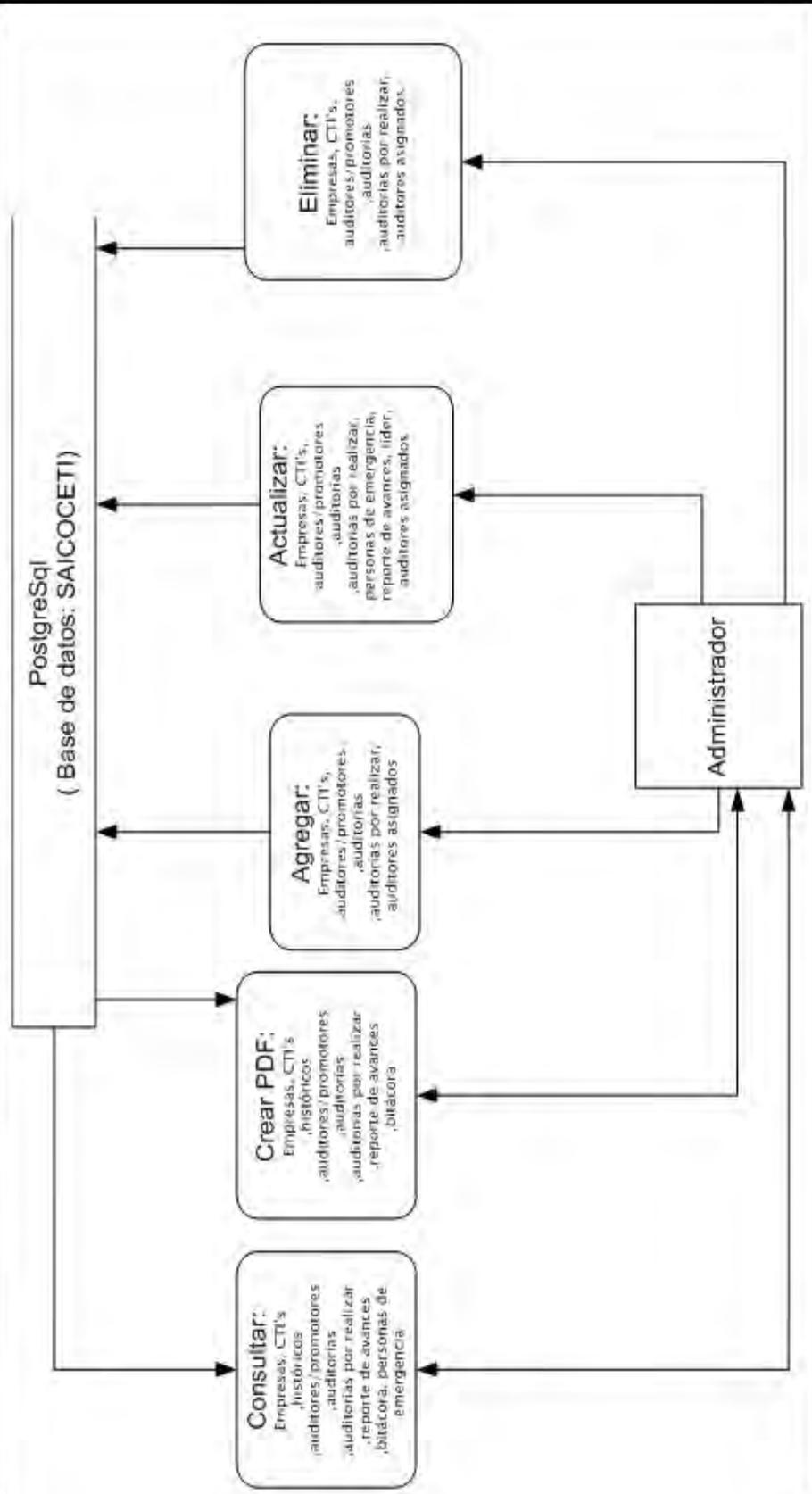


DIAGRAMA DE FLUJO DE DATOS



ÁREA: Análisis CVE. PROYECTO: INT03007 CVE. REFERENCIA: DADMF04
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 4 DE: 4
 SAI-01



ELABORÓ: <u>Rolando Lima Maciel</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u>
FECHA: <u>25/11/2007</u>	FECHA: <u>10/12/2007</u>



DICCIONARIO DE TÉRMINOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA05-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 2

TÉRMINO	DESCRIPCIÓN
Consulta de avances	Proceso en el cual una empresa consulta los avances de la auditoría que contrató
Consulta de auditoría disponibles	Proceso en el cual un usuario consulta las auditorías disponibles para contratación
Sugerir auditorías	Proceso en el cual un auditor sugiere preguntas a las auditorías existentes
Sugerir preguntas	Proceso en el cual un auditor sugiere nuevas auditorías
Elegir grupo de trabajo	Proceso en el cual un líder de auditoría selecciona su grupo de trabajo
Comenzar o reanudar auditorías	Proceso en el cual los auditores capturan las observaciones de las auditorías que se les asignaron
Consultar empresas	Proceso en el cual se muestra la información de las empresas registradas en el SAICOCETI
Consultar CTI's	Proceso en el cual se muestra la información de los CTI's registrados en el SAICOCETI
Consultar históricos	Proceso en el cual se muestra la información de las participaciones de los auditores en proyectos
Consultar auditores/promotores	Proceso en el cual se muestra la información de los auditores ó promotores registrados en el SAICOCETI
Consultar auditorías por realizar	Proceso en el cual se muestra la información de las auditorías por realizar registradas en el SAICOCETI
Consultar reporte de avances	Proceso en el cual se muestra la información de las observaciones registradas por los auditores en el SAICOCETI
Consultar bitácora	Proceso en el cual se muestra la información del tiempo empleado por los auditores para registrar sus observaciones en el SAICOCETI
Consultar personas de emergencia	Consulta la información de las personas a las que hay que informar si ocurre una emergencia con un auditor
Crear PDF Empresas	Creación de un archivo PDF con la información de la empresas registradas en el SAICOCETI
Crear PDF CTI's	Creación de un archivo PDF con la información de los CTI's registrados en el SAICOCETI
Crear PDF históricos	Creación de un archivo PDF con la información de las participaciones de los auditores en proyectos
Crear PDF auditores/promotores	Creación de un archivo PDF con la información de los auditores ó promotores registrados en el SAICOCETI
Crear PDF auditorías	Creación de un archivo PDF con la información de las auditorías registradas en el SAICOCETI
Crear PDF auditorías por realizar	Creación de un archivo PDF con la información de las auditorías por realizar registradas en el SAICOCETI
Crear PDF reporte de avances	Creación de un archivo PDF con las observaciones realizadas por los auditores
Crear PDF bitácora	Creación de un archivo PDF del tiempo empleado por los auditores para registrar sus observaciones
Agregar empresas	Proceso en el cual se guardan los datos de una nueva empresa en el SAICOCETI
Agregar CTI's	Proceso en el cual se guardan los datos de un nuevo CTI en el SAICOCETI

ELABORÓ: Rolando Lima Maciel AUTORIZÓ: Heriberto Olguin Romo

FECHA: 25/11/2007 FECHA: 10/12/2007



DICCIONARIO DE TÉRMINOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA05-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 2 DE: 2

TÉRMINO	DESCRIPCIÓN
Agregar auditores/promotores	Proceso en el cual se guardan los datos de un nuevo auditor o promotor en el SAICOCETI
Agregar auditorías	Proceso en el cual se guardan los datos de una nueva auditoría en el SAICOCETI
Agregar auditorías por realizar	Proceso en el cual se guardan los datos para realizar nueva auditoría
Agregar auditores asignados	Proceso en el cual se guardan se definen las auditorías que un auditor puede llevar a cabo
Actualizar empresas	Proceso en el cual se actualizan los datos de un empresa
Actualizar CTI's	Proceso en el cual se actualizan los datos de un CTI
Actualizar auditores/promotores	Proceso en el cual se actualizan los datos de un auditor o un promotor
Actualizar auditorías	Proceso en el cual se actualizan los datos de una auditoría
Actualizar auditorías por realizar	Proceso en el cual se actualizan los datos de una auditoría por realizar
Actualizar personas de emergencia	Proceso en el cual se actualizan los datos de las personas de emergencia
Actualizar reporte de avances	Proceso en el cual se actualizan las observaciones realizadas por los auditores
Actualizar líder	Proceso en el cual se asigna ó invita a un auditor como líder para un proyecto
Actualizar auditores asignados	Proceso en el cual se actualizan las auditorías que un auditor puede llevar a cabo
Eliminar empresas	Proceso en el cual se eliminan los datos de una empresa del SAICOCETI
Eliminar CTI's	Proceso en el cual se eliminan los datos de un CTI del SAICOCETI
Eliminar auditores/promotores	Proceso en el cual se eliminan los datos de un auditor o promotor del SAICOCETI
Eliminar auditorías	Proceso en el cual se eliminan los datos de una auditoría del SAICOCETI
Eliminar auditorías por realizar	Proceso en el cual se eliminan los datos de una auditoría por realizar del SAICOCETI
Eliminar auditores asignados	Proceso en el cual se eliminan las auditorías que un auditor puede llevar a cabo

ELABORÓ: <u>Rolando Lima Maciel</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u>
FECHA: <u>25/11/2007</u>	FECHA: <u>10/12/2007</u>

	DIAGRAMA DE CLASES
ÁREA: <u> Análisis </u>	CVE: <u>PROYECTO: IN103007</u> CVE: <u>REFERENCIA: DA03-SAI-01</u>
PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u>	ABREV.: <u>SAICOCETI</u> HOJA: <u>1</u> DE: <u>1</u>
El diagrama de clases se entrega en formato JPG	
ELABORÓ: <u>Rolando Lima Maciel</u> FECHA: <u>25/11/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>



DICCIONARIO DE CLASES

ÁREA: Análisis CVE PROYECTO: IN103007 CVE. REFERENCIA: DADME07-SAI
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 1

NOMBRE:	ATRIBUTOS	DESCRIPCIÓN DE LA CLASE
SUBCLASE DE:		
HERENCIA:		
- NOMBRE Y TIPO DE MÉTODO		DESCRIPCIÓN DEL MÉTODO
El presente formato se sustituye por JavaDoc		

ELABORÓ: <u>Rolando Lima Maciel</u> FECHA: <u>25/11/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
---	--

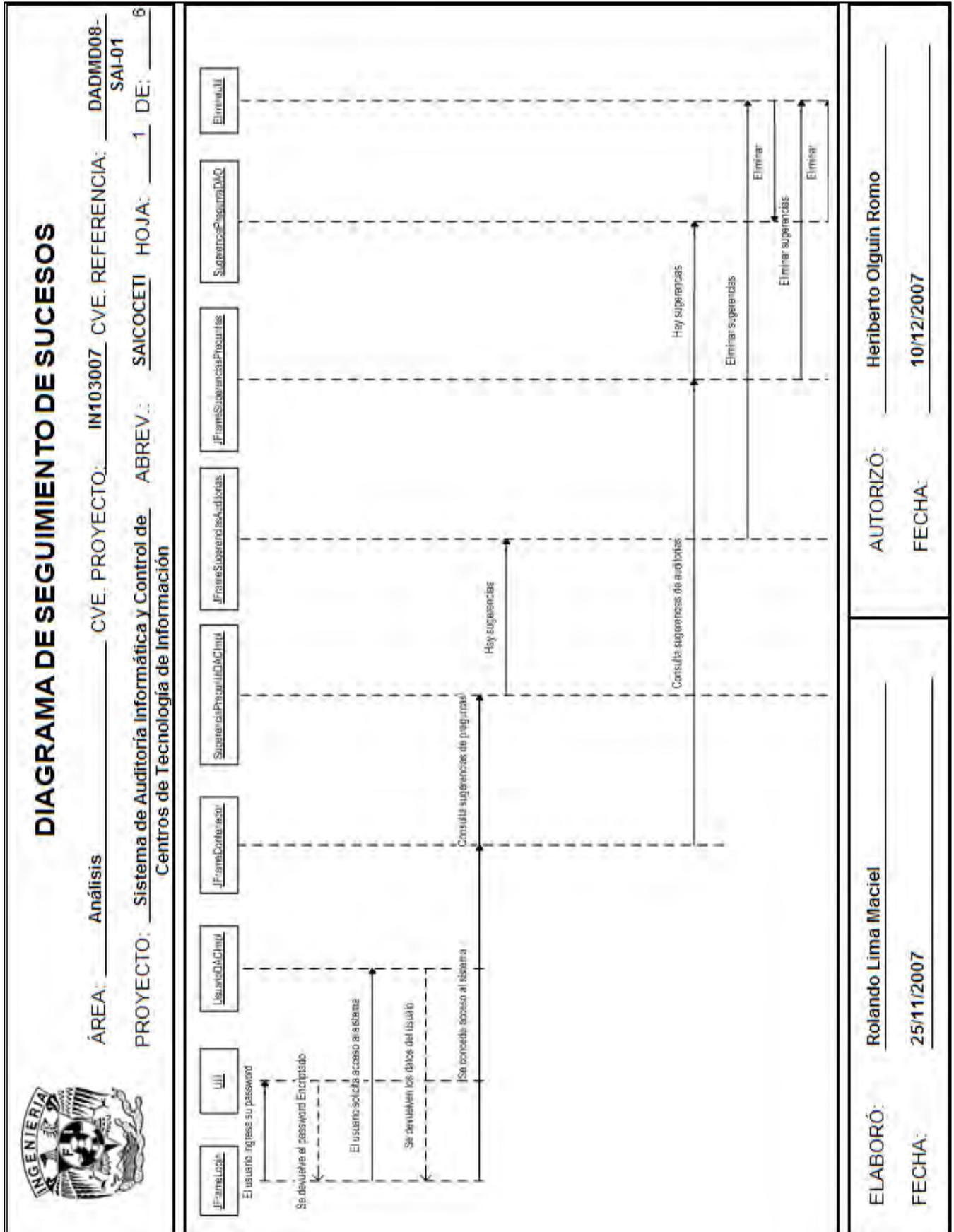
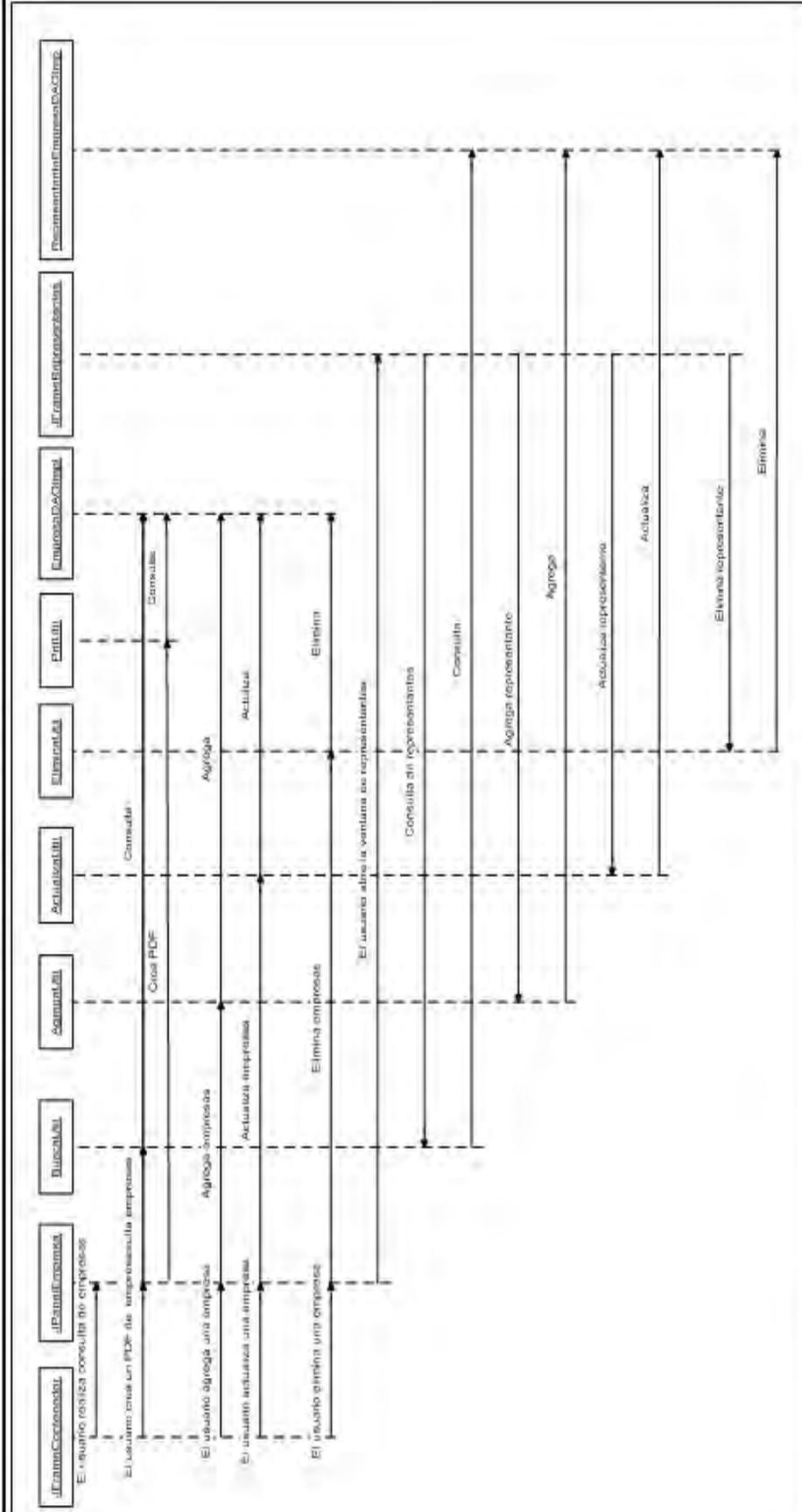




DIAGRAMA DE SEGUIMIENTO DE SUCESOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DADMD08-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 2 DE: 6



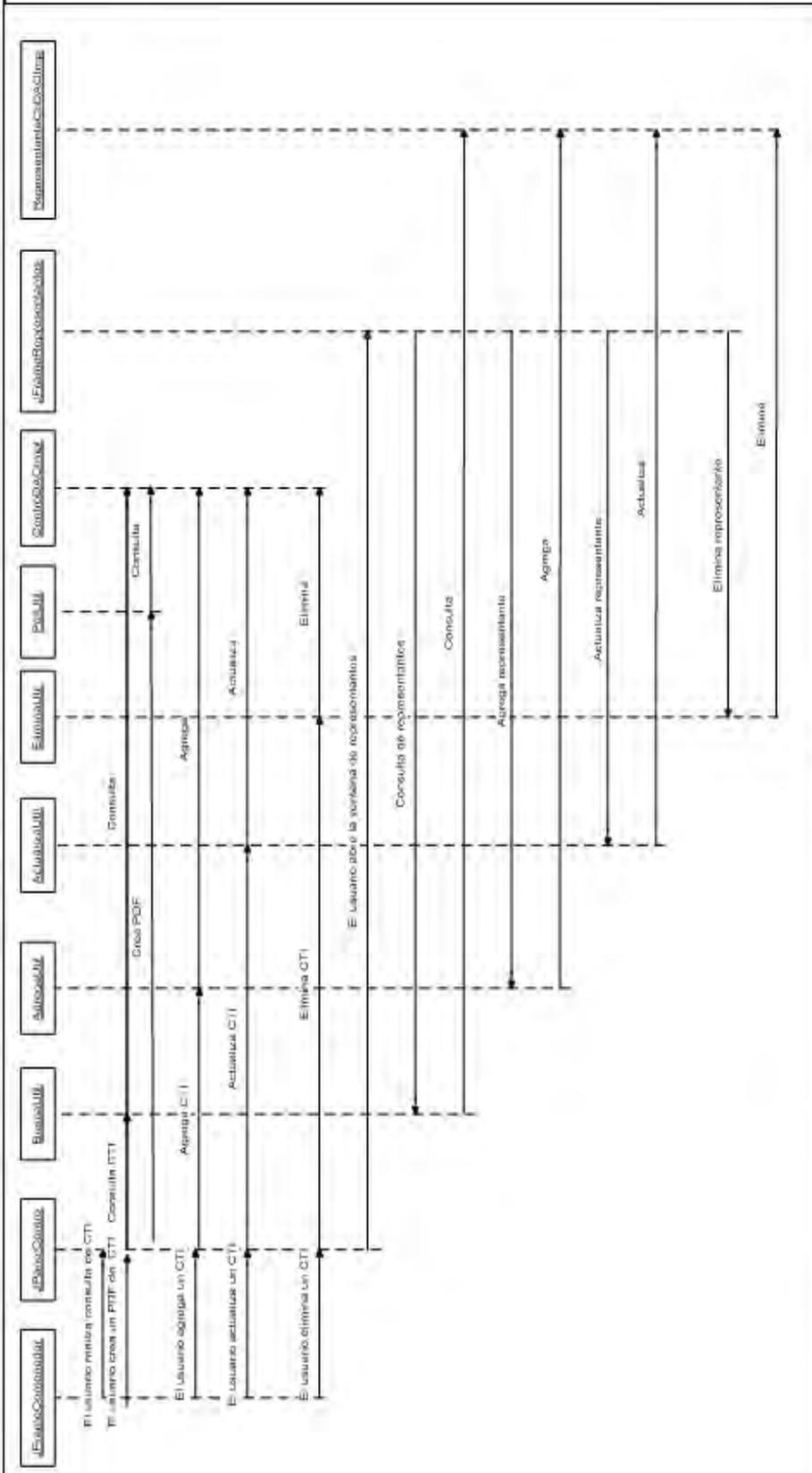
ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007



DIAGRAMA DE SEGUIMIENTO DE SUCESOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DADMD08-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 3 DE: 6

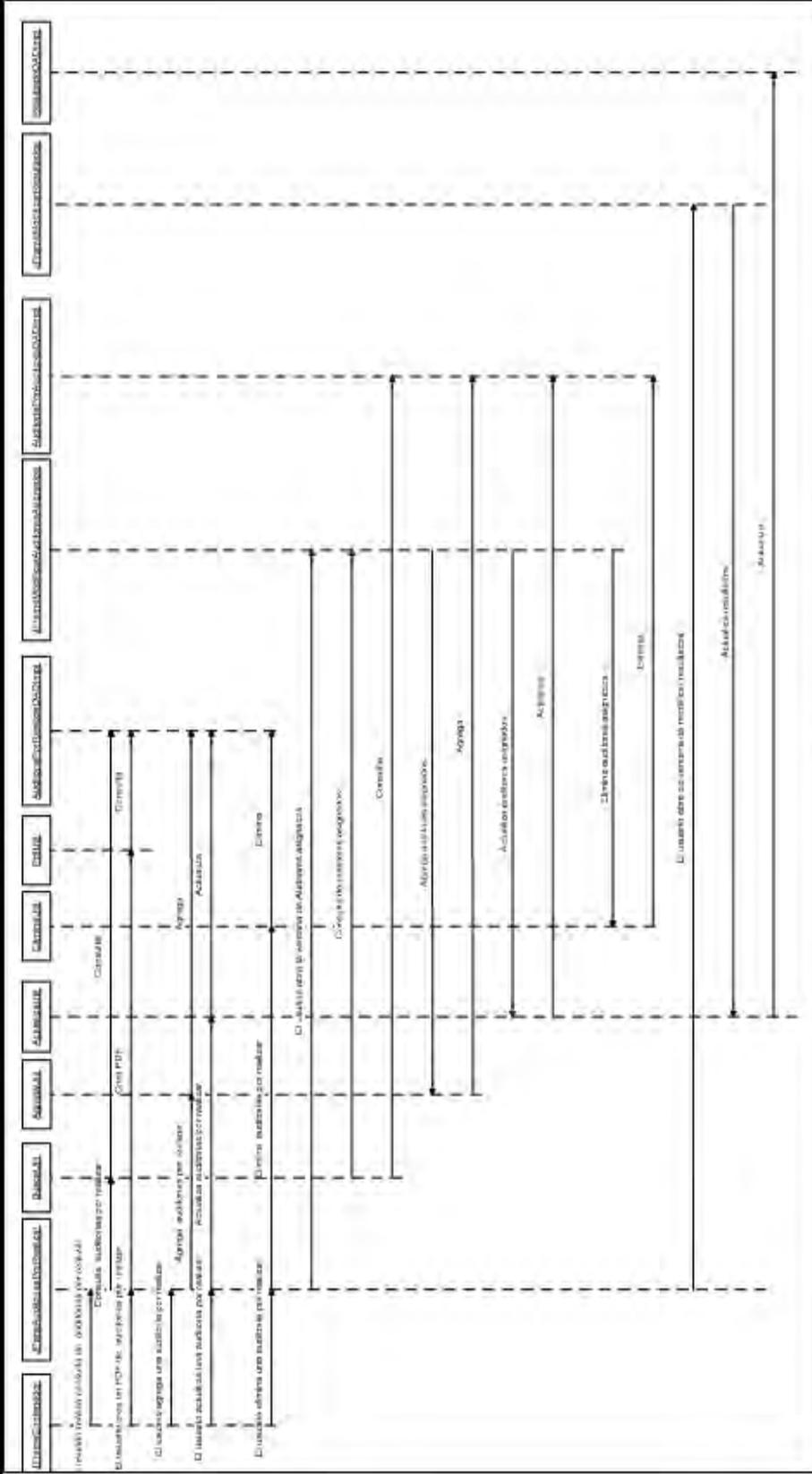


ELABORÓ: Rolando Lima Maciel AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 25/11/2007 FECHA: 10/12/2007



DIAGRAMA DE SEGUIMIENTO DE SUCESOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DADMD08-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 5 DE: 6



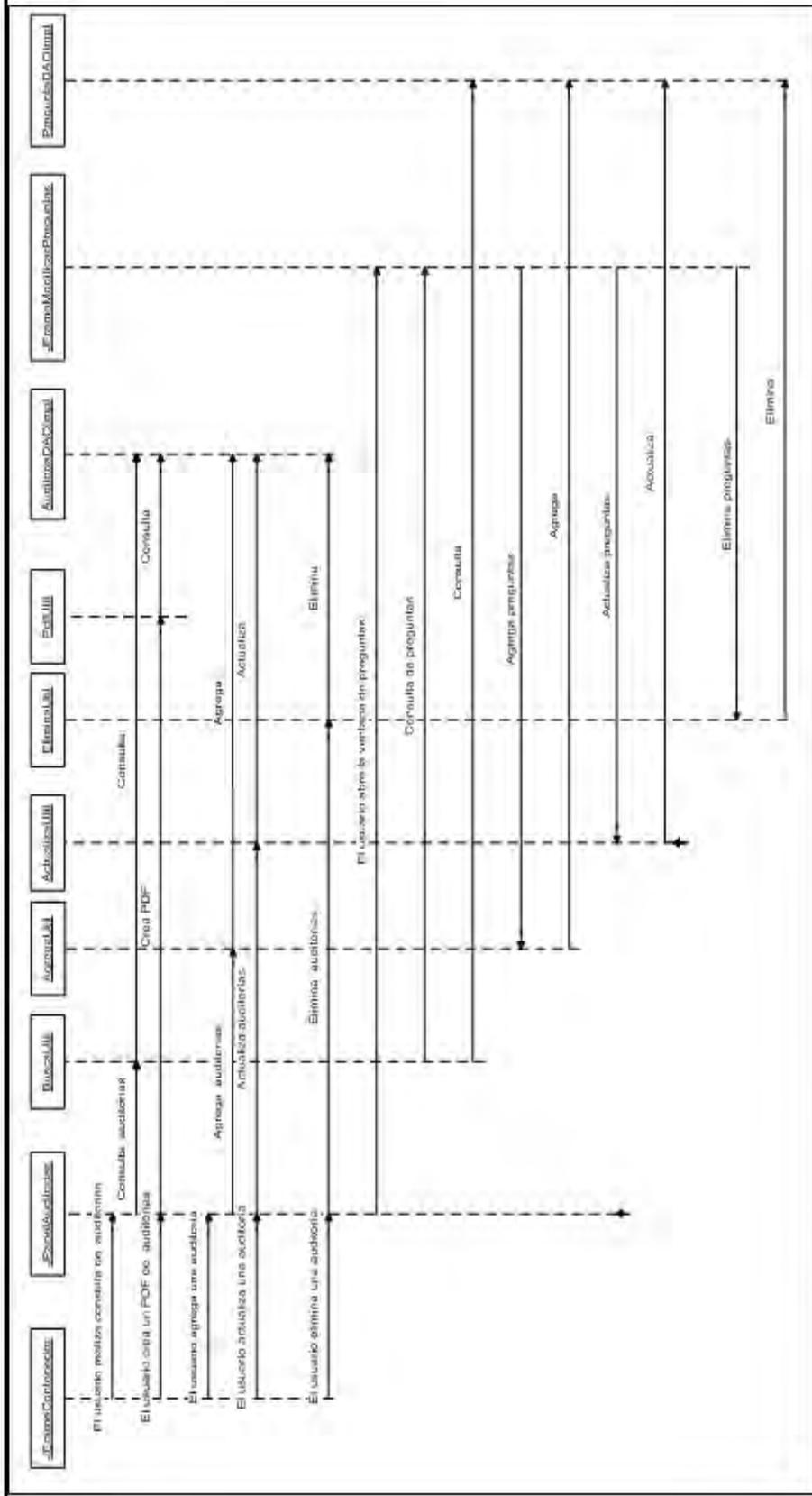
ELABORÓ: Rolando Lima Maciel AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 25/11/2007 FECHA: 10/12/2007



DIAGRAMA DE SEGUIMIENTO DE SUCESOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DADMD08-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 6 DE: 6



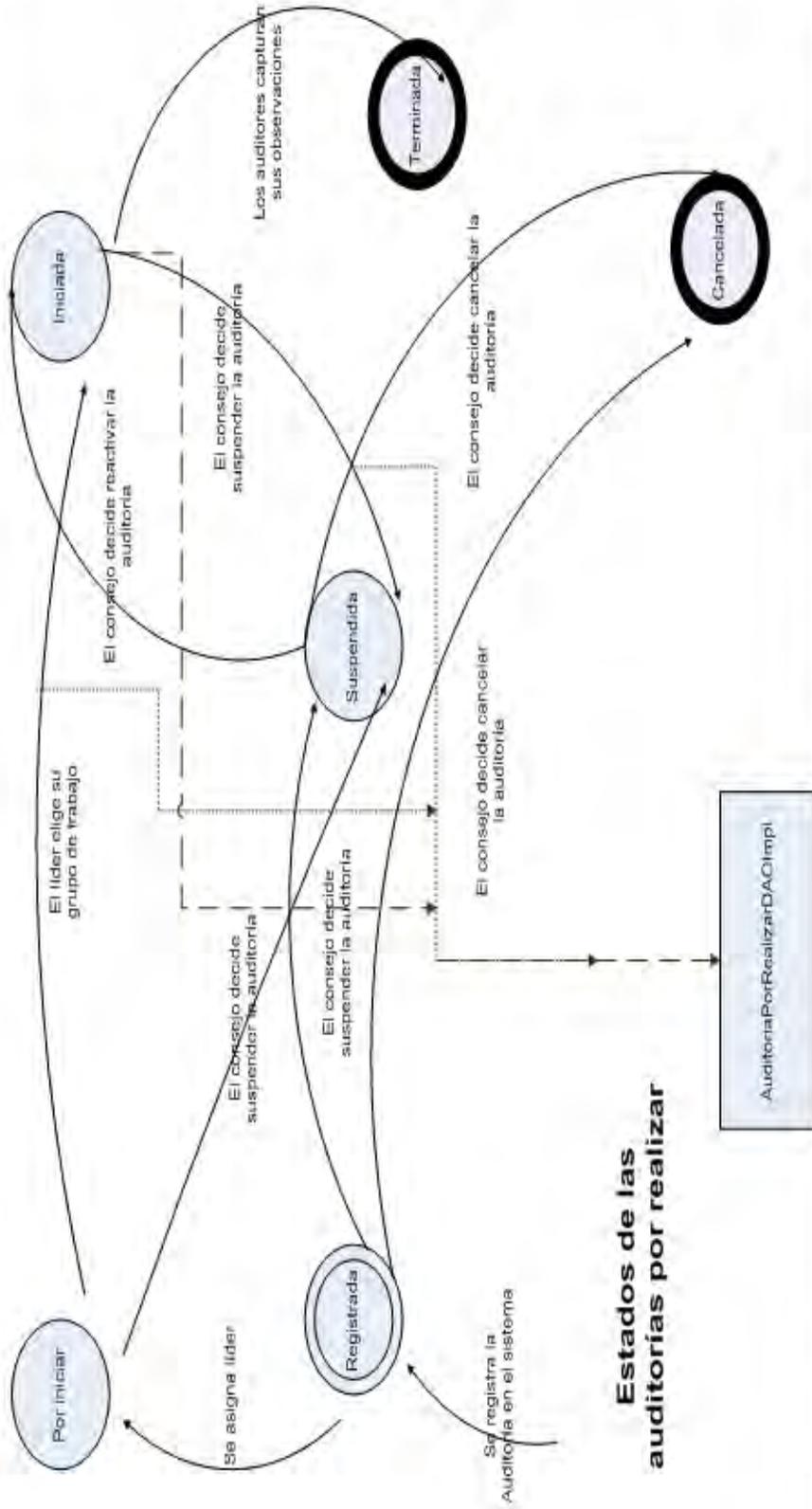
ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007

DIAGRAMA DE ESTADOS



ÁREA: Análisis CVE: PROYECTO: IN103007 CVE: REFERENCIA: DADM09-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 1



ELABORÓ: Rolando Lima Maciel AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 25/11/2007 FECHA: 10/12/2007



ACTA DE ACUERDOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA10-SAL-01
 PROYECTO Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 7

ASISTENTES:

Ing. Heriberto Olguín Romo (Director de tesis)
 José Ángel Haro Juárez (tesista)
 Rolando Lima Maciel (tesista)

PUNTOS TRATADOS:

- Pantalla de aditores
- Pantalla de aditorias
- Pantalla de aditorias por realizar
- Otros puntos
- Puntos extras a valorar

CONCLUSIONES Y ACUERDOS:

En la siguiente página

ELABORÓ:

 Rolando Lima Maciel

FECHA:

 06/12/2007

AUTORIZÓ:

 Heriberto Olguín Romo

FECHA:

 10/12/2007

ACTA DE ACUERDOS	
 <p> ÁREA: <u>Análisis</u> CVE. PROYECTO: <u>IN103007</u> CVE. REFERENCIA: <u>DA10-SAI-01</u> PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u> ABREV.: <u>SAICOCETI</u> HOJA: <u>2</u> DE: <u>7</u> </p>	
ASISTENTES:	
Ing. Heriberto Olguín Romo (Director de tesis)	
José Ángel Haro Juárez (tesista)	
Rolando Lima Maciel (tesista)	
CONCLUSIONES Y ACUERDOS:	
<p>Pantalla de auditores: Cambiar el título de la pantalla "Auditores" por "Auditores/Promotores" Adecuar la pantalla de "Auditores" para que haga el manejo de promotores también, así como hacer las adecuaciones pertinentes en la base de datos.</p> <p>En la pantalla de auditorías: Se cambia el símbolo # por la palabra "Número" Reemplazar "Auditoria" por "Auditoria" Reemplazar "Auditorias" por "Auditorias"</p> <p>En la pantalla de auditorías por realizar: Folio 4 dígitos. Se debe agregar sub-pantallas con las diferentes funciones Para la captura es fijo lo siguiente:</p> <ul style="list-style-type: none"> • Empresa • Centro • Auditoría • Promotor (nuevo atributo) 	
ELABORÓ: <u>Rolando Lima Maciel</u>	AUTORIZÓ: <u>Heriberto Olguín Romo</u>
FECHA: <u>06/12/2007</u>	FECHA: <u>10/12/2007</u>

ACTA DE ACUERDOS	
	<p>ÁREA: <u> Análisis </u> CVE. PROYECTO: <u> IN103007 </u> CVE. REFERENCIA: <u> DA10-SAL-01 </u></p> <p>PROYECTO: <u> Sistema de Auditoría Informática y Control de Centros de Tecnología de Información </u> ABREV.: <u> SAICOCETI </u> HOJA: <u> 3 </u> DE: <u> 7 </u></p>
ASISTENTES:	
Ing. Heriberto Olguín Romo (Director de tesis)	
José Ángel Haro Juárez (tesista)	
Rolando Lima Maciel (tesista)	
CONCLUSIONES Y ACUERDOS:	
<p>Para selección la selección del promotor se deben mostrar en una lista desplegable la cual contenga todos los promotores y auditores registrados en el sistema y se mostrará "(P)" para indicar si es promotor o bien "(A)" para en caso de que sea auditor.</p> <p>Para la pantalla de históricos se debe indicar si la auditoría ya se encuentra terminada o esta en proceso.</p> <p>Debe haber una sub-pantalla para realizar la asignación del líder de la auditoría por realizar en cuestión, en la cual se debe mostrar una lista con los auditores que tienen conocimiento del tipo de auditoría que se va aplicar con la condición de que en ese momento este participando en dos o menos auditorías. Se debe mostrar:</p> <ul style="list-style-type: none"> *Nombre completo *Número de participaciones en el tipo de auditoría. *Nivel con el que ha participado. <p>Del listado que se muestre se tendrá la posibilidad de seleccionar a un grupo de auditores para invitarlos a ser los líderes en la auditoría y por consecuencia enviarles un mail de aviso.</p> <p>Una vez elegido el líder se le enviará un mail de aviso de su asignación solicitándole el programa de actividades del plan de trabajo. Y a los demás auditores que no se seleccionaron como líderes se deberá enviarles una notificación de que el líder ya fue asignado.</p> <p>El líder elegirá a su grupo de trabajo en el modulo web, en donde se le mostrará un listado de los auditores especialistas en la auditoría en cuestión y podrá elegir un grupo de ellos para formen parte de su equipo de trabajo asignandoles un nivel (senior, junior ó asistente)</p>	
ELABORÓ: <u> Rolando Lima Maciel </u>	AUTORIZÓ: <u> Heriberto Olguín Romo </u>
FECHA: <u> 06/12/2007 </u>	FECHA: <u> 10/12/2007 </u>

	<h2 style="margin: 0;">ACTA DE ACUERDOS</h2>
ÁREA: <u>Análisis</u> CVE. PROYECTO: <u>IN103007</u> CVE. REFERENCIA: <u>DA10-SAI-01</u>	PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u> ABREV.: <u>SAICOCETI</u> HOJA: <u>4</u> DE: <u>7</u>
ASISTENTES:	
Ing. Heriberto Olguín Romo (Director de tesis)	
José Ángel Haro Juárez (tesista)	
Rolando Lima Maciel (tesista)	
CONCLUSIONES Y ACUERDOS:	
La definición de los estatus de las auditorías por realizar queda de esta manera: • Por iniciar: Desde que se invita a los promotores a ser líderes hasta que selecciona su grupo de trabajo • Iniciada: Desde la selección del grupo de trabajo por parte del líder hasta que se entrega del informe ejecutivo. • Terminada: A la entrega del informe ejecutivo. Para realizar cancelación o suspensión sólo puede ser por el consejo de la empresa y para cualquiera de los dos casos se debe realizar una captura de comentarios. Para futuras versiones: • Considerar la generación automática del contrato de la auditoría. • Al dar de alta a un auditor capturar el nivel mínimo con el que quiere participar en las auditorías • Quitar las recomendaciones personales del auditor de los reportes ejecutivos • Crear una pantalla para consulta de históricos. Pantalla de Auditorías • Se reemplazará la palabra "Clave" por "folio" • Se elimina el costo • Los encabezados de la tabla que muestra las preguntas se cambian a esta manera: Pregunta o acción Guía o sugerencia al auditor Número de pregunta o acción Activa	
ELABORÓ: <u>Rolando Lima Maciel</u>	AUTORIZÓ: <u>Heriberto Olguín Romo</u>
FECHA: <u>06/12/2007</u>	FECHA: <u>10/12/2007</u>



ACTA DE ACUERDOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA10-SAL-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 5 DE: 7

ASISTENTES:

- Ing. Heriberto Olguín Romo (Director de tesis)
- José Ángel Haro Juárez (tesista)
- Rolando Lima Maciel (tesista)

CONCLUSIONES Y ACUERDOS:

- Cuando se suspenda a un auditor capturar las observaciones del motivo de la suspensión.
 - Se reemplazará la palabra "Dirección" por "Domicilio".
 - El folio de la auditoría se mostrará con 3 dígitos.
 - El folio de las preguntas será 3 dígitos(auditoría) – 3 dígitos (preguntas).
 - En la subpantalla de modificación de preguntas sólo se cambia el texto de los títulos para que coincidan con los de la tabla anteriormente definidos.
 - Poner el ordenamiento de las preguntas y mostrarlas por las que están activas y las que no.
- Pantalla de Auditorías por realizar(inicio).
- Se agrega el envío de mails, para invitar a los auditores a participar en una auditoría.
 - Al registrar una auditoría, se debe ingresar quien va a ser el líder el cual podrá decidir las personas que participaran con el en la auditoría
 - Se agrega el estatus de "En asignación de personal".
- Puntos extras a valorar:

ELABORÓ: <u>Rolando Lima Maciel</u>	AUTORIZÓ: <u>Heriberto Olguín Romo</u>
FECHA: <u>06/12/2007</u>	FECHA: <u>10/12/2007</u>

	<h2 style="margin: 0;">ACTA DE ACUERDOS</h2>
<p>ÁREA: <u>Análisis</u> CVE. PROYECTO: <u>IN103007</u> CVE. REFERENCIA: <u>DA10-SAI-01</u></p> <p>PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u> ABREV.: <u>SAICOCETI</u> HOJA: <u>6</u> DE: <u>7</u></p>	
<p>ASISTENTES:</p> <p>Ing. Heriberto Olguín Romo (Director de tesis)</p> <p>José Ángel Haro Juárez (tesista)</p> <p>Rolando Lima Maciel (tesista)</p>	
<p>CONCLUSIONES Y ACUERDOS:</p> <ul style="list-style-type: none"> • Se reemplazará la palabra "Calve" por "Folio". • El folio del auditor se manejará con dos dígitos (00). • El código postal se valida que siempre tenga 5 dígitos (contando los ceros a la izquierda). • El RFC para personas se valida como 4 letras- 6 dígitos- 3 caracteres alfanuméricos. • Se cambia la organización de los campos a esta manera. • Se cambia el texto de "Auditorías asignadas" por "Auditor en". • Únicamente se mostrarán las auditorías asignadas al auditor y se permitirá agregar/eliminar las auditorías asignadas. • Agregar captura de dos personas para aviso en caso de emergencia con: Nombre, tel. móvil, tel casa. Pantalla de Empresas • El RFC para empresas se valida como 3 letras- 6 dígitos- 3 caracteres alfanuméricos. • Se reemplazará la palabra "Dirección" por "Domicilio". • Mostrar únicamente los representantes de la empresa seleccionada en una ventana aparte donde se podrán agregar los representantes de manera individual a cada empresa. (Poner un botón para acceso a la ventana), se deben tener los siguientes datos: cargo en la empresa, título, nombre, tel, tel móvil, email. • Poner una liga que lleve a la pantalla de CTI para a mostrar los CTI de la empresa. Pantalla de CTI • Se cambia el texto de "Nombre del centro" por "Nombre CTI". 	
<p>ELABORÓ: <u>Rolando Lima Maciel</u></p> <p>FECHA: <u>06/12/2007</u></p>	<p>AUTORIZÓ: <u>Heriberto Olguín Romo</u></p> <p>FECHA: <u>10/12/2007</u></p>



ACTA DE ACUERDOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DA10-SAI-01

PROYECTO Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 7 DE: 7

ASISTENTES:

- Ing. Heriberto Olguín Romo (Director de tesis)
- José Ángel Haro Juárez (tesista)
- Rolando Lima Maciel (tesista)

CONCLUSIONES Y ACUERDOS:

- NO se agregan los campos para RFC y CURP por los CTI.
- Mostrar únicamente los representantes del CTI seleccionado en una ventana aparte donde se podrán agregar los representantes de manera individual a cada CTI. (Poner un botón para acceso a la ventana), se deben tener los siguientes datos: cargo en la empresa, título, nombre, tel, tel móvil, email.
- Mostrar los centros organizados por folio del centro.
- Poner un botón que nos cambie a la pantalla de empresas conteniendo la información de la empresa al que pertenece el centro.
- Se cambiarán los escudos de la UNAM y FI.
- La pantalla de representantes se elimina debido a que estarán incluidos en empresas y CTI.

COMPROMISOS:

ACTIVIDAD	RESPONSABLE	FECHAS
Desarrollo módulo Web	José Ángel Haro Juárez	31/10/2007
Desarrollo módulo de administración	Rolando Lima Maciel	31/10/2007

ELABORÓ: Rolando Lima Maciel

FECHA: 06/12/2007

AUTORIZÓ: Heriberto Olguín Romo

FECHA: 10/12/2007

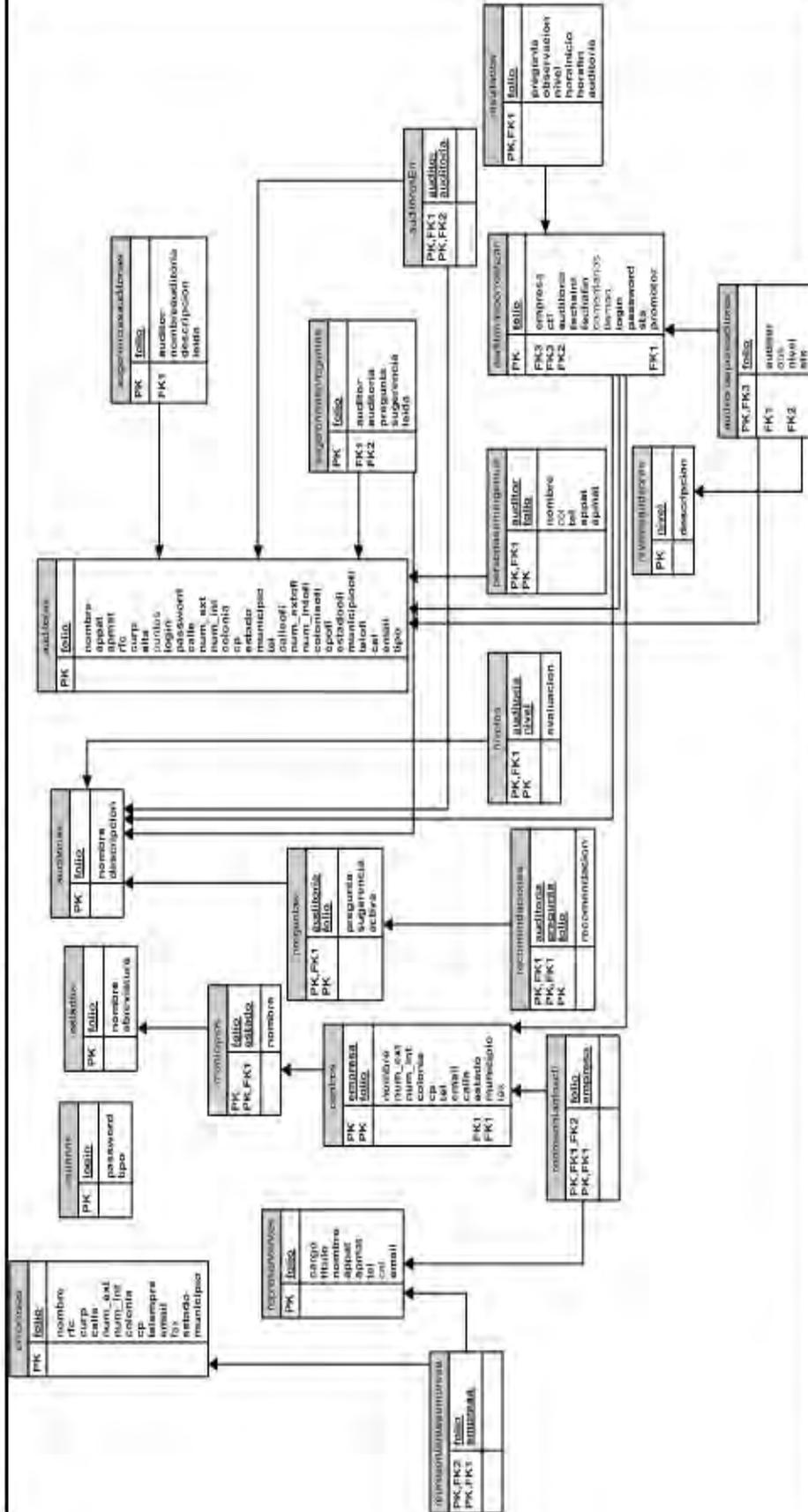
 <h2 style="margin: 0;">SELECCIÓN E IDENTIFICACIÓN DE SOFTWARE Y HARDWARE</h2>	
ÁREA: <u> Diseño </u>	CVE. PROYECTO: <u> IN103007 </u> CVE. REFERENCIA: <u> DD01-SAL-01 </u>
PROYECTO: <u> Sistema de Auditoría Informática y Control de Centros de Tecnología de Información </u> ABREV.: <u> SAICOCETI </u> HOJA: <u> 1 </u> DE: <u> 1 </u>	
SOFTWARE/HARDWARE	USO
<p>160 GB en disco duro</p> <p>4 GB Memoria RAM</p> <p>Procesador Intel™ Core®2 Duo E4300 (1.80GHz/800MHz/2MB L2) 525W</p> <p>Tarjeta de red</p> <p>LINUX</p> <p>IP real</p> <p>Java 1.5.x</p> <p>Apache tomcat</p> <p>PostgreSQL</p>	<p>Almacenamiento la información de las empresas, centros de tecnología de información, auditores, promotores, auditorías, auditorías por realizar así como del software requerido para el servidor web y el manejador de base de datos.</p> <p>Atender las peticiones realizadas al servidor con un rendimiento adecuado tanto para el servidor web como para manejador de bases de datos.</p> <p>Atender las peticiones realizadas al servidor con un rendimiento adecuado tanto para el servidor web como para manejador de bases de datos.</p> <p>Acceso al sistema desde cualquier lugar que cuente con una conexión a Internet.</p> <p>Administración de las correcto funcionamiento del servidor</p> <p>Acceso al sistema desde cualquier lugar que cuente con una conexión a Internet.</p> <p>Ejecución del servidor apache tomcat.</p> <p>Permitir a los auditores capturar sus observaciones así como a las empresas consultar los avances de las auditorías contratadas.</p> <p>Manejo del almacenamiento la información de las empresas, centros de tecnología de información, auditores, promotores, auditorías, auditorías por realizar.</p>
ELABORÓ: <u> Rolando Lima Maciel </u>	AUTORIZÓ: <u> Heriberto Olguín Romo </u>
FECHA: <u> 25/11/2007 </u>	FECHA: <u> 10/12/2007 </u>



DIAGRAMA ENTIDAD-RELACION

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD02-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 1



ELABORÓ: Rolando Lima Maciel AUTORIZÓ: Heriberto Olguin Romo

FECHA: 25/11/2007 FECHA: 10/12/2007



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 17

NOMBRE DE LA TABLA: LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(auditor)	auditores	Uno a muchos
(auditoriasistema)	auditorias	Uno a muchos
(auditoriasugerida)	sugerencias auditorias	Uno a muchos

ELABORÓ: José Ángel Haro Juárez

FECHA: 06/12/2007

AUTORIZÓ: Heriberto Olguín Romo

FECHA: 10/12/2007



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 2 DE: 17

NOMBRE DE LA TABLA: Auditores

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(estado, municipio)	Estados Municipios	Uno a Muchos
(estadoof, municipioof)	Estados Municipios	Uno a Muchos

ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
--	--



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 4 DE: 17

NOMBRE DE LA TABLA: Auditorías por auditores

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(auditor)	auditores	Uno a muchos
(folio)	auditorías por realizar	Uno a muchos
(nivel)	niveles auditores	Uno a muchos

ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguín Romo</u> FECHA: <u>10/12/2007</u>
--	--



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 5 DE: 17

NOMBRE DE LA TABLA: Auditorias por realizar

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(promotor)	auditores	Uno a muchos
(empresa, cti)	centros	Uno a muchos
(auditoria)	auditorias	Uno a muchos

ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
--	--



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 6 DE: 17

NOMBRE DE LA TABLA: Centros

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(municipio, estado)	municipios	Uno a uno
(empresa)	empresas	Uno a muchos

ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguín Romo</u> FECHA: <u>10/12/2007</u>
--	--



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 12 DE: 17

NOMBRE DE LA TABLA: Recomendaciones por respuesta

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(recomendacion, pregunta, auditoria)	recomendaciones	Uno a uno
(respuesta, pregunta)	resultados	Uno a muchos

ELABORÓ: <u> José Ángel Haro Juárez </u> FECHA: <u> 06/12/2007 </u>	AUTORIZÓ: <u> Heriberto Olguín Romo </u> FECHA: <u> 10/12/2007 </u>
--	--



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 13 DE: 17

NOMBRE DE LA TABLA: Representantes

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(cti, empresa)	centros	Uno a muchos
(folio)	representantes	Uno a uno

ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
--	--



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO: Sistema de Auditoria Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 14 DE: 17

NOMBRE DE LA TABLA: Representantes empresas

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(empresa)	empresas	Uno a uno
(folio)	representantes	Uno a muchos

ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
--	--



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 16 DE: 17

NOMBRE DE LA TABLA: Sugerencias/Auditorias

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(folio)	auditorias por realizar	Uno a muchos
(pregunta, auditoria)	preguntas	Uno a muchos

ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
--	--



REGISTRO DE LLAVES SECUNDARIAS

ÁREA: Análisis CVE. PROYECTO: IN10300 CVE. REFERENCIA: DD03b-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 17 DE: 17

NOMBRE DE LA TABLA: Sugerenciaspreguntas

LLAVE SECUNDARIA	RELACIÓN CON LA TABLA	CARDINALIDAD DE LA RELACIÓN
(auditor)	auditores	Uno a muchos
(auditoriasistema)	auditorias	Uno a muchos
(auditoriasugerida)	sugerenciasauditorias	Uno a muchos

ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
--	--



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD04-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAI HOJA: 1 DE: 22

NOMBRE DEL ARCHIVO		DESCRIPCIÓN				OBSERVACIONES
AUDITORES	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	
	folio	int	8	NOT NULL	Pk	Identificador de los auditores registrados
	nombre	varchar	59	NOT NULL		Nombre del auditor
	apmat	varchar	50	NOT NULL		Apellido paterno del auditor
	apmat	varchar	50	NOT NULL		Apellido materno del auditor
	rfc	varchar	15	NOT NULL		Rfc del auditor
	curp	varchar	20	NOT NULL		Curp del auditor
	alta	timestamp		NOT NULL		Fecha de Alta de la auditoría
	puntos	int	8	NULL		Puntos asignados al auditor
	login	varchar	20	NOT NULL		Login del auditor para realizar las auditorías
	password	varchar	20	NOT NULL		Password del auditor para realizar las auditorías
	calle	varchar	20	NOT NULL		Calle
	num_ext	varchar	15	NOT NULL		Número exterior
	num_int	varchar	15	NOT NULL		Número interior de la ubicación de la auditoría
	colonia	varchar	35	NOT NULL		Colonia
	cp	varchar	8	NOT NULL		Código postal
	estado	int	8	NOT NULL	fk1	Estado
	municipio	int	8	NOT NULL	fk1	Municipio
	tel	varchar	20	NOT NULL		Teléfono del auditor
	calleofi	varchar	20	NOT NULL		Calle de la oficina del auditor
	num_extofi	varchar	15	NOT NULL		Número exterior de la oficina del auditor
	num_intofi	varchar	15	NOT NULL		Número interior de la oficina del auditor
	coloniaofi	varchar	35	NOT NULL		Colonia de la oficina del auditor
	cpofi	varchar	8	NOT NULL		Código postal de la oficina del auditor
	estadooofi	int	8	NOT NULL	fk2	Estado de la oficina del auditor
	municipioofi	int	8	NOT NULL	fk2	Municipio de la oficina del auditor
	telofi	varchar	20	NOT NULL		Teléfono de la oficina del auditor
	cel	varchar	20	NOT NULL		Teléfono celular del auditor
	email	varchar	50	NOT NULL		Correo electrónico del auditor
	tipo	varchar	2	NOT NULL		Tipo del auditor

ELABORÓ: José Ángel Haro Juárez AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 06/12/2007 FECHA: 10/12/2007

	<h3>DICCIONARIO DE DATOS</h3>																		
ÁREA: <u> Análisis </u>	CVE. PROYECTO: <u> IN103007 </u>	CVE. REFERENCIA: <u> DD04-SAI-01 </u>																	
PROYECTO: <u> Sistema de Auditoría Informática y Control de Centros de Tecnología de Información </u>	ABREV.: <u> SAICOCETI </u>	HOJA: <u> 2 </u> DE: <u> 22 </u>																	
NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																	
AUDITORESEN	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e0f0ff;"> <th style="text-align: left;">Nombre del campo</th> <th style="text-align: left;">Tipo de dato</th> <th style="text-align: left;">Precisión</th> <th style="text-align: left;">Nulo / No Nulo</th> <th style="text-align: left;">Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>auditor</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Pk,Fk1</td> </tr> <tr> <td>auditoria</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Pk,Fk2</td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	auditor	int	8	NOT NULL	Pk,Fk1	auditoria	int	8	NOT NULL	Pk,Fk2	Identificador de la tabla Identificador de la tabla		
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave															
auditor	int	8	NOT NULL	Pk,Fk1															
auditoria	int	8	NOT NULL	Pk,Fk2															
ELABORÓ: <u> José Ángel Haro Juárez </u>	AUTORIZÓ: <u> Heriberto Olguin Romo </u>			FECHA: <u> 06/12/2007 </u>															
FECHA: <u> 06/12/2007 </u>	FECHA: <u> 10/12/2007 </u>																		

	<h3>DICCIONARIO DE DATOS</h3>																											
ÁREA: <u> Análisis </u>	CVE PROYECTO: <u> IN103007 </u>	CVE REFERENCIA: <u> DD04-SAI-01 </u>																										
PROYECTO: <u> Sistema de Auditoría Informática y Control de Centros de Tecnología de Información </u>	ABREV.: <u> SAICOCETI </u>	HOJA: <u> 3 </u> DE <u> 22 </u>																										
NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																										
AUDITORIAS	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="background-color: #e0f0ff;">Nombre del campo</th> <th style="background-color: #e0f0ff;">Tipo de dato</th> <th style="background-color: #e0f0ff;">Precisión</th> <th style="background-color: #e0f0ff;">Nulo / No Nulo</th> <th style="background-color: #e0f0ff;">Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>folio</td> <td>int</td> <td></td> <td></td> <td></td> </tr> <tr> <td>nombre</td> <td>text</td> <td>8</td> <td>NOT NULL</td> <td>Pk</td> </tr> <tr> <td>descripcion</td> <td>text</td> <td></td> <td>NOT NULL</td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	folio	int				nombre	text	8	NOT NULL	Pk	descripcion	text		NOT NULL					NOT NULL		Identificador de la auditoría Nombre de la auditoría Descripción de la auditoría	
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																								
folio	int																											
nombre	text	8	NOT NULL	Pk																								
descripcion	text		NOT NULL																									
			NOT NULL																									
ELABORÓ: <u> José Ángel Haro Juárez </u>	FECHA: <u> 06/12/2007 </u>	AUTORIZÓ: <u> Heriberto Olguin Romo </u>	FECHA: <u> 10/12/2007 </u>																									



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IM103007 CVE. REFERENCIA: DD04-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 4 DE: 22

NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																														
AUDITORIA SPORAUDITORES	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e0f0ff;"> <th>Nombre del campo</th> <th>Tipo de dato</th> <th>Precisión</th> <th>Nulo / No Nulo</th> <th>Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>auditor</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Pk,Fk2</td> </tr> <tr> <td>obs</td> <td>varchar</td> <td>255</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>nivel</td> <td>varchar</td> <td>2</td> <td>NOT NULL</td> <td>Fk3</td> </tr> <tr> <td>sts</td> <td>char</td> <td>1</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Pk,Fk1</td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	auditor	int	8	NOT NULL	Pk,Fk2	obs	varchar	255	NOT NULL		nivel	varchar	2	NOT NULL	Fk3	sts	char	1	NOT NULL		folio	int	8	NOT NULL	Pk,Fk1	<p>Identificador de los auditores registrados</p> <p>Observaciones del auditor</p> <p>Nivel del auditor</p> <p>Estatus del auditor</p> <p>Identificador de la tabla</p>
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																												
auditor	int	8	NOT NULL	Pk,Fk2																												
obs	varchar	255	NOT NULL																													
nivel	varchar	2	NOT NULL	Fk3																												
sts	char	1	NOT NULL																													
folio	int	8	NOT NULL	Pk,Fk1																												

ELABORÓ: José Ángel Haro Juárez

FECHA: 06/12/2007

AUTORIZÓ: Heriberto Olquin Romo

FECHA: 10/12/2007



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD04-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 5 DE: 22

NOMBRE DEL ARCHIVO	DESCRIPCIÓN				OBSERVACIONES
AUDITORIA SPORREALIZAR	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave
	empresa	int	8	NOT NULL	FK1
	cti	int	8	NOT NULL	FK1
	auditoria	int	8	NOT NULL	FK2
	fechaini	timestamp		NOT NULL	
	fechafin	timestamp		NOT NULL	
	comentarios	text		NULL	
	tiempo	timestamp	20	NULL	
	login	varchar	40	NOT NULL	
	password	varchar	40	NOT NULL	
	sis	char	1	NOT NULL	
	folio	int	8	NOT NULL	PK
	promotor	int	8	NOT NULL	FK

Identificador de los auditores registrados
 Identificador del centro de tecnología
 Identificador de las auditorías
 Fecha de inicio
 Fecha de fin
 Comentarios del auditor
 Tiempo estimado de duración de la auditoría
 Login de la empresa relacionada al auditor para consultar avances
 Password de la empresa relacionada al auditor para consultar avances
 Estatus de la auditoría
 Identificador de la auditoría
 Identificador del promotor asociado

ELABORÓ: José Ángel Haro Juárez AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 06/12/2007 FECHA: 10/12/2007



DICCIONARIO DE DATOS

ÁREA: Análisis

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información

CVE PROYECTO: IM103007

ABREV: SAICOCETI

CVE REFERENCIA: DD04-SAI-01

HOJA: 6 DE 22

NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																																																																						
CENTROS	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e1f5fe;"> <th>Nombre del campo</th> <th>Tipo de dato</th> <th>Precisión</th> <th>Nulo / No Nulo</th> <th>Tipo de Llave</th> </tr> </thead> <tbody> <tr><td>folio</td><td>int</td><td>8</td><td>NOT NULL</td><td>Pk</td></tr> <tr><td>empresa</td><td>int</td><td>8</td><td>NOT NULL</td><td>Pk,Fk2</td></tr> <tr><td>nombre</td><td>varchar</td><td>20</td><td>NOT NULL</td><td></td></tr> <tr><td>num_ext</td><td>varchar</td><td>15</td><td>NOT NULL</td><td></td></tr> <tr><td>num_int</td><td>varchar</td><td>15</td><td>NOT NULL</td><td></td></tr> <tr><td>colonia</td><td>varchar</td><td>35</td><td>NOT NULL</td><td></td></tr> <tr><td>cp</td><td>varchar</td><td>8</td><td>NOT NULL</td><td></td></tr> <tr><td>tel</td><td>varchar</td><td>20</td><td>NOT NULL</td><td></td></tr> <tr><td>email</td><td>varchar</td><td>50</td><td>NOT NULL</td><td></td></tr> <tr><td>calle</td><td>varchar</td><td>20</td><td>NOT NULL</td><td></td></tr> <tr><td>estado</td><td>int</td><td>8</td><td>NOT NULL</td><td></td></tr> <tr><td>municipio</td><td>int</td><td>8</td><td>NOT NULL</td><td>FK1</td></tr> <tr><td>fax</td><td>varchar</td><td>20</td><td>NULL</td><td>FK1</td></tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	folio	int	8	NOT NULL	Pk	empresa	int	8	NOT NULL	Pk,Fk2	nombre	varchar	20	NOT NULL		num_ext	varchar	15	NOT NULL		num_int	varchar	15	NOT NULL		colonia	varchar	35	NOT NULL		cp	varchar	8	NOT NULL		tel	varchar	20	NOT NULL		email	varchar	50	NOT NULL		calle	varchar	20	NOT NULL		estado	int	8	NOT NULL		municipio	int	8	NOT NULL	FK1	fax	varchar	20	NULL	FK1	<p>Identificador de la tabla</p> <p>Identificador de la empresa</p> <p>Nombre del centro</p> <p>Número exterior de la dirección del centro</p> <p>Número interior de la dirección del centro</p> <p>Colonia de ubicación del centro</p> <p>Código postal de la ubicación del centro</p> <p>Teléfono del centro</p> <p>Correo electrónico del centro</p> <p>Calle de ubicación del centro</p> <p>Estado de residencia del centro</p> <p>Municipio de residencia del centro</p> <p>Fax del centro</p>
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																																																																				
folio	int	8	NOT NULL	Pk																																																																				
empresa	int	8	NOT NULL	Pk,Fk2																																																																				
nombre	varchar	20	NOT NULL																																																																					
num_ext	varchar	15	NOT NULL																																																																					
num_int	varchar	15	NOT NULL																																																																					
colonia	varchar	35	NOT NULL																																																																					
cp	varchar	8	NOT NULL																																																																					
tel	varchar	20	NOT NULL																																																																					
email	varchar	50	NOT NULL																																																																					
calle	varchar	20	NOT NULL																																																																					
estado	int	8	NOT NULL																																																																					
municipio	int	8	NOT NULL	FK1																																																																				
fax	varchar	20	NULL	FK1																																																																				

ELABORÓ: José Ángel Haro Juárez

FECHA: 06/12/2007

AUTORIZÓ: Heriberto Olguín Romo

FECHA: 10/12/2007



DICCIONARIO DE DATOS

ÁREA: Análisis CVE PROYECTO: IN103007 CVE REFERENCIA: DD04-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 7 DE 22

NOMBRE DEL ARCHIVO		DESCRIPCIÓN			OBSERVACIONES
EMPRESAS	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave
	folio	int	8	NOT NULL	Pk
	nombre	varchar	50	NOT NULL	
	rfc	varchar	15	NOT NULL	
	curp	varchar	20	NOT NULL	
	calle	varchar	20	NOT NULL	
	num_ext	varchar	15	NOT NULL	
	num_int	varchar	15	NOT NULL	
	colonia	varchar	35	NOT NULL	
	cp	varchar	8	NOT NULL	
	telempre	varchar	20	NOT NULL	
	email	varchar	50	NOT NULL	
	fax	varchar	50	NULL	
	estado	int	8	NOT NULL	Fk1
	municipio	int	8	NOT NULL	Fk1

ELABORÓ: José Ángel Haro Juárez AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 06/12/2007 FECHA: 10/12/2007

	<h2 style="margin: 0;">DICCIONARIO DE DATOS</h2>																							
ÁREA: <u> Análisis </u>	CVE. PROYECTO: <u> IM103007 </u>	CVE. REFERENCIA: <u> DD04-SAI-01 </u>																						
PROYECTO: <u> Sistema de Auditoría Informática y Control de Centros de Tecnología de Información </u>	ABREV.: <u> SAICOCETI </u>	HOJA: <u> 8 </u> DE: <u> 22 </u>																						
NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																						
ESTADOS	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e1f5fe;"> <th style="text-align: left;">Nombre del campo</th> <th style="text-align: left;">Tipo de dato</th> <th style="text-align: left;">Precisión</th> <th style="text-align: left;">Nulo / No Nulo</th> <th style="text-align: left;">Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1</td> </tr> <tr> <td>abreviatura</td> <td>varchar</td> <td>5</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>nombre</td> <td>varchar</td> <td>35</td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	folio	int	8	NOT NULL	PK1	abreviatura	varchar	5	NOT NULL		nombre	varchar	35	NOT NULL		Identificador del estado Abreviatura del estado Nombre completo del estado		
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																				
folio	int	8	NOT NULL	PK1																				
abreviatura	varchar	5	NOT NULL																					
nombre	varchar	35	NOT NULL																					
ELABORÓ: <u> José Ángel Haro Juárez </u>	AUTORIZÓ: <u> Heriberto Olguin Romo </u>			FECHA: <u> 06/12/2007 </u>																				
FECHA: <u> 06/12/2007 </u>	FECHA: <u> 10/12/2007 </u>																							



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD04-SAL-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 9 DE: 22

NOMBRE DEL ARCHIVO	DESCRIPCIÓN				OBSERVACIONES																	
MUNICIPIOS	<table border="1"> <thead> <tr> <th>Nombre del campo</th> <th>Tipo de dato</th> <th>Precisión</th> <th>Nulo / No Nulo</th> <th>Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1</td> </tr> <tr> <td>estado</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1,FK1</td> </tr> <tr> <td>nombre</td> <td>varchar</td> <td>35</td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	folio	int	8	NOT NULL	PK1	estado	int	8	NOT NULL	PK1,FK1	nombre	varchar	35	NOT NULL		Identificador del municipio Identificador del estado Nombre del municipio
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																		
folio	int	8	NOT NULL	PK1																		
estado	int	8	NOT NULL	PK1,FK1																		
nombre	varchar	35	NOT NULL																			

ELABORÓ: José Ángel Haro Juárez AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 06/12/2007 FECHA: 10/12/2007

	DICCIONARIO DE DATOS																							
ÁREA: <u>Análisis</u>	CVE. PROYECTO: <u>IN103007</u>	CVE. REFERENCIA: <u>DD04-SAI-01</u>																						
PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u>	ABREV.: <u>SAICOCETI</u>	HOJA: <u>10</u> DE: <u>22</u>																						
NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																						
NIVELES	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e1f5fe;"> <th>Nombre del campo</th> <th>Tipo de dato</th> <th>Precisión</th> <th>Nulo / No Nulo</th> <th>Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>nivel</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Pk1</td> </tr> <tr> <td>evaluacion</td> <td>text</td> <td></td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>auditoria</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Fk1</td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	nivel	int	8	NOT NULL	Pk1	evaluacion	text		NOT NULL		auditoria	int	8	NOT NULL	Fk1	Identificador del nivel Evaluacion crrespondiente Identificador de la auditoria		
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																				
nivel	int	8	NOT NULL	Pk1																				
evaluacion	text		NOT NULL																					
auditoria	int	8	NOT NULL	Fk1																				
ELABORÓ: <u>José Ángel Haro Juárez</u>	FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u>	FECHA: <u>10/12/2007</u>																					



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IM103007 CVE. REFERENCIA: DD04-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 11 DE: 22

NOMBRE DEL ARCHIVO		DESCRIPCIÓN				OBSERVACIONES														
NIVELESAUDITORES	<table border="1"> <thead> <tr> <th>Nombre del campo</th> <th>Tipo de dato</th> <th>Precisión</th> <th>Nulo / No Nulo</th> <th>Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>nivel</td> <td>varchar</td> <td>2</td> <td>NOT NULL</td> <td>PK1</td> </tr> <tr> <td>descripcion</td> <td>varchar</td> <td>255</td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	nivel	varchar	2	NOT NULL	PK1	descripcion	varchar	255	NOT NULL		Identificador del nivel	Descripción del nivel		
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																
nivel	varchar	2	NOT NULL	PK1																
descripcion	varchar	255	NOT NULL																	
ELABORÓ:	<u> José Ángel Haro Juárez </u>	AUTORIZÓ:	<u> Heriberto Olguin Romo </u>	FECHA:	<u> 10/12/2007 </u>															
FECHA:	<u> 06/12/2007 </u>	FECHA:	<u> 10/12/2007 </u>																	



DICCIONARIO DE DATOS

ÁREA: Análisis

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información

CVE. PROYECTO: IM103007

ABREV.: SAICOCETI

CVE. REFERENCIA: DD04-SAI-01

HOJA: 12 DE: 22

NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																																								
<p>PERSONA SEMERGENCIA</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e0f0ff;"> <th>Nombre del campo</th> <th>Tipo de dato</th> <th>Precisión</th> <th>Nulo / No Nulo</th> <th>Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>auditor</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1,FK1</td> </tr> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1</td> </tr> <tr> <td>nombre</td> <td>varchar</td> <td>35</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>cel</td> <td>varchar</td> <td>20</td> <td></td> <td></td> </tr> <tr> <td>tel</td> <td>varchar</td> <td>20</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>appat</td> <td>varchar</td> <td>50</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>apmat</td> <td>varchar</td> <td>50</td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	auditor	int	8	NOT NULL	PK1,FK1	folio	int	8	NOT NULL	PK1	nombre	varchar	35	NOT NULL		cel	varchar	20			tel	varchar	20	NOT NULL		appat	varchar	50	NOT NULL		apmat	varchar	50	NOT NULL		<p>Identificador del auditor relacionado Identificador de la tabla Nombre de la persona de emergencia Celular de la persona de emergencia Teléfono de la persona de emergencia Apellido paterno Apellido materno</p>	
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																																						
auditor	int	8	NOT NULL	PK1,FK1																																						
folio	int	8	NOT NULL	PK1																																						
nombre	varchar	35	NOT NULL																																							
cel	varchar	20																																								
tel	varchar	20	NOT NULL																																							
appat	varchar	50	NOT NULL																																							
apmat	varchar	50	NOT NULL																																							

ELABORÓ: José Ángel Haro, Juárez

FECHA: 06/12/2007

AUTORIZÓ: Heriberto Olguin Romo

FECHA: 10/12/2007



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IM103007 CVE. REFERENCIA: DD04-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información SAICOCETI HOJA: 13 DE: 22
 ABREV.: _____

NOMBRE DEL ARCHIVO	DESCRIPCIÓN				OBSERVACIONES
RECOMENDACIONES	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave
	auditoria	int	8	NOT NULL	Pk1, Fk1
	pregunta	text		NOT NULL	
	folio	int	8	NOT NULL	Pk1
	sugerencia	text		NOT NULL	
	activa	bool		NOT NULL	

Identificador de la auditoria a la que se aplica la recomendación
 Pregunta
 Identificador de la tabla
 Sugerencia
 Estatus de la recomendación activa o no

ELABORÓ: José Ángel Haro Juárez AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 06/12/2007 FECHA: 10/12/2007

	<h2 style="margin: 0;">DICCIONARIO DE DATOS</h2>			DD04-SAL-01																									
ÁREA: <u>Análisis</u>	CVE. PROYECTO: <u>IN103007</u>	CVE. REFERENCIA: _____	HOJA: <u>14</u> DE: <u>22</u>																										
PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u>	ABREV.: _____	SAICOCETI _____	HOJA: _____ DE: _____																										
NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																											
RECOMENDACIONES	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e0f0ff;"> <th style="text-align: left;">Nombre del campo</th> <th style="text-align: left;">Tipo de dato</th> <th style="text-align: left;">Precisión</th> <th style="text-align: left;">Nulo / No Nulo</th> <th style="text-align: left;">Tipo de Liave</th> </tr> </thead> <tbody> <tr> <td>auditoria</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1,FK1</td> </tr> <tr> <td>pregunta</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1,FK1</td> </tr> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1</td> </tr> <tr> <td>recomendaciones</td> <td>text</td> <td></td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Liave	auditoria	int	8	NOT NULL	PK1,FK1	pregunta	int	8	NOT NULL	PK1,FK1	folio	int	8	NOT NULL	PK1	recomendaciones	text		NOT NULL		Identificador de la auditoria Identificador de la pregunta Identificador de la tabla Recomendaciones		
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Liave																									
auditoria	int	8	NOT NULL	PK1,FK1																									
pregunta	int	8	NOT NULL	PK1,FK1																									
folio	int	8	NOT NULL	PK1																									
recomendaciones	text		NOT NULL																										
ELABORÓ: <u>José Ángel Haro Juárez</u>	FECHA: <u>06/12/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u>	FECHA: <u>10/12/2007</u>																										



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD04-SAL-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 15 DE 22

NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																									
RECOMENDACIONESPORRESPUESTA <table border="1"> <thead> <tr> <th>Nombre del campo</th> <th>Tipo de dato</th> <th>Precisión</th> <th>Nulo / No Nulo</th> <th>Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>recomendacion</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1</td> </tr> <tr> <td>respuesta</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1,FK2</td> </tr> <tr> <td>pregunta</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK1,FK2</td> </tr> <tr> <td>auditoria</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>FK1</td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	recomendacion	int	8	NOT NULL	PK1	respuesta	int	8	NOT NULL	PK1,FK2	pregunta	int	8	NOT NULL	PK1,FK2	auditoria	int	8	NOT NULL	FK1	<p>Identificador de las recomendaciones Identificador de la respuesta Identificador de la pregunta Identificador de la auditoría</p>	<p>AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u></p> <p>ELABORÓ: <u>José Ángel Haro Juárez</u> FECHA: <u>06/12/2007</u></p>
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																							
recomendacion	int	8	NOT NULL	PK1																							
respuesta	int	8	NOT NULL	PK1,FK2																							
pregunta	int	8	NOT NULL	PK1,FK2																							
auditoria	int	8	NOT NULL	FK1																							



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD04-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 16 DE: 22

NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																																																		
REPRESENTANTES	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e0f0ff;"> <th>Nombre del campo</th> <th>Tipo de dato</th> <th>Precisión</th> <th>Nulo / No Nulo</th> <th>Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Pk</td> </tr> <tr> <td>cargo</td> <td>varchar</td> <td>50</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>titulo</td> <td>varchar</td> <td>50</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>nombre</td> <td>varchar</td> <td>50</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>apmat</td> <td>varchar</td> <td>50</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>apmat</td> <td>varchar</td> <td>50</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>tel</td> <td>varchar</td> <td>20</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>cel</td> <td>varchar</td> <td>20</td> <td>NULL</td> <td></td> </tr> <tr> <td>email</td> <td>varchar</td> <td>50</td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	folio	int	8	NOT NULL	Pk	cargo	varchar	50	NOT NULL		titulo	varchar	50	NOT NULL		nombre	varchar	50	NOT NULL		apmat	varchar	50	NOT NULL		apmat	varchar	50	NOT NULL		tel	varchar	20	NOT NULL		cel	varchar	20	NULL		email	varchar	50	NOT NULL		<p>Identificador de la tabla Cargo del representante Titulo del representante Nombre del representante Apellido paterno del representante Apellido materno del representante Teléfono del representante Teléfono celular del representante Correo electrónico del representante</p>
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																																																
folio	int	8	NOT NULL	Pk																																																
cargo	varchar	50	NOT NULL																																																	
titulo	varchar	50	NOT NULL																																																	
nombre	varchar	50	NOT NULL																																																	
apmat	varchar	50	NOT NULL																																																	
apmat	varchar	50	NOT NULL																																																	
tel	varchar	20	NOT NULL																																																	
cel	varchar	20	NULL																																																	
email	varchar	50	NOT NULL																																																	

ELABORÓ: José Ángel Haro Juárez

FECHA: 06/12/2007

AUTORIZÓ: Heriberto Olguin Romo

FECHA: 10/12/2007

 <h2 style="margin: 0;">DICCIONARIO DE DATOS</h2>																					
ÁREA: <u>Análisis</u>	CVE. PROYECTO: <u>IM103007</u>																				
PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u>	CVE. REFERENCIA: <u>DD04-SAI-01</u>																				
ABREV.: <u>SAICOCEI</u>	HOJA: <u>17</u> DE: <u>22</u>																				
NOMBRE DEL ARCHIVO	DESCRIPCIÓN																				
REPRESENTANTE SCTI	OBSERVACIONES																				
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e0f0ff;"> <th style="text-align: left;">Nombre del campo</th> <th style="text-align: left;">Tipo de dato</th> <th style="text-align: left;">Precisión</th> <th style="text-align: left;">Nulo / No Nulo</th> <th style="text-align: left;">Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Pk,Fk</td> </tr> <tr> <td>empresa</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Fk</td> </tr> <tr> <td>cti</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	folio	int	8	NOT NULL	Pk,Fk	empresa	int	8	NOT NULL	Fk	cti	int	8	NOT NULL		Identificador de la tabla Identificador de la empresa Identificador del centro de tecnología
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																	
folio	int	8	NOT NULL	Pk,Fk																	
empresa	int	8	NOT NULL	Fk																	
cti	int	8	NOT NULL																		
ELABORÓ: <u>José Ángel Haro Juárez</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u>																				
FECHA: <u>06/12/2007</u>	FECHA: <u>10/12/2007</u>																				

	<h3>DICCIONARIO DE DATOS</h3>																		
ÁREA: <u>Análisis</u>	CVE. PROYECTO: <u>IN103007</u>	CVE. REFERENCIA: <u>DD04-SAI-01</u>																	
PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u>	ABREV.: <u>SAICOCETI</u>	HOJA: <u>18</u>	DE: <u>22</u>																
NOMBRE DEL ARCHIVO		DESCRIPCIÓN																	
REPRESENTANTE SEMPRESAS		OBSERVACIONES																	
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="background-color: #e0f0ff;">Nombre del campo</th> <th style="background-color: #e0f0ff;">Tipo de dato</th> <th style="background-color: #e0f0ff;">Precisión</th> <th style="background-color: #e0f0ff;">Nulo / No Nulo</th> <th style="background-color: #e0f0ff;">Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>PK, Fk</td> </tr> <tr> <td>empresa</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Fk</td> </tr> </tbody> </table>		Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	folio	int	8	NOT NULL	PK, Fk	empresa	int	8	NOT NULL	Fk	Identificador de la tabla Identificador de la empresa		
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave															
folio	int	8	NOT NULL	PK, Fk															
empresa	int	8	NOT NULL	Fk															
ELABORÓ: <u>José Ángel Haro Juárez</u>		AUTORIZÓ: <u>Heriberto Olguin Romo</u>																	
FECHA: <u>06/12/2007</u>		FECHA: <u>10/12/2007</u>																	



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IM103007 CVE. REFERENCIA: DD04-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 19 DE: 22

NOMBRE DEL ARCHIVO		DESCRIPCIÓN				OBSERVACIONES
RESULTADOS	Nombre del campo	Tipo de dato	Precisión	Null / No Null	Tipo de Llave	
	pregunta	int	8	NOT NULL	Fk2	Identificador de la pregunta
	observacion	text		NOT NULL	Fk1	Observación realizada por el auditor
	nivel	int	8	NOT NULL		Nivel de la pregunta asignada por el auditor
	horainicio	time		NOT NULL		Hora en la que comenzo el auditor a contestar una pregunta
	horafin	time		NOT NULL		Hora en la que termino el auditor a contestar una pregunta
	folio	int	8	NOT NULL	Pk,Fk	Identificador de la tabla
	auditoria	int	8	NOT NULL	Fk2	Identificador de la auditoría relacionada

ELABORÓ: José Ángel Haro Juárez AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 06/12/2007 FECHA: 10/12/2007

	<h2 style="margin: 0;">DICCIONARIO DE DATOS</h2>																																
ÁREA: <u> Análisis </u>	CVE. PROYECTO: <u> IM103007 </u>	CVE. REFERENCIA: <u> DD04-SAI-01 </u>																															
PROYECTO: <u> Sistema de Auditoría Informática y Control de Centros de Tecnología de Información </u>	ABREV.: <u> SAICOCETI </u>	HOJA: <u> 20 </u> DE: <u> 22 </u>																															
NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																															
SUGERENCIA SAUDITORIAS	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #e0f0ff;"> <th style="text-align: left;">Nombre del campo</th> <th style="text-align: left;">Tipo de dato</th> <th style="text-align: left;">Precisión</th> <th style="text-align: left;">Nulo / No Nulo</th> <th style="text-align: left;">Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>folio</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>Pk</td> </tr> <tr> <td>auditor</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td>FK1</td> </tr> <tr> <td>nombreauditoria</td> <td>text</td> <td></td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>descripcion</td> <td>text</td> <td></td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>leida</td> <td>bool</td> <td></td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	folio	int	8	NOT NULL	Pk	auditor	int	8	NOT NULL	FK1	nombreauditoria	text		NOT NULL		descripcion	text		NOT NULL		leida	bool		NOT NULL		Identificador de las sugerencias por preguntas Identificador del auditor que sugirió la pregunta Nombre de la auditoría sugerida Descripción de la auditoría Campo de control sobre si ya fue leída la pregunta o no en el sistema	
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																													
folio	int	8	NOT NULL	Pk																													
auditor	int	8	NOT NULL	FK1																													
nombreauditoria	text		NOT NULL																														
descripcion	text		NOT NULL																														
leida	bool		NOT NULL																														
ELABORÓ:	FECHA:	AUTORIZÓ:	FECHA:																														
<u> José Ángel Haro Juárez </u>	<u> 06/12/2007 </u>	<u> Heriberto Olguín Romo </u>	<u> 10/12/2007 </u>																														



DICCIONARIO DE DATOS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD04-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 21 DE: 22

NOMBRE DEL ARCHIVO	DESCRIPCIÓN				OBSERVACIONES
SUGERENCIA SPREGUNTAS	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave
folio	int		8	NOT NULL	Pk
auditor	int		8	NOT NULL	FK1
auditoriasistema	int		8	NOT NULL	FK2
pregunta	text			NOT NULL	Pregunta sugerida por el auditor
sugerencia	text			NOT NULL	Sugerencia realizada por el auditor
leida	bool			NOT NULL	Campo de control sobre si ya fue leida la pregunta o no en el sistema
auditoriasugerida	int		8	NULL	Auditoria sugerida por el auditor

ELABORÓ: José Ángel Haro Juárez AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 06/12/2007 FECHA: 10/12/2007

 <h2 style="margin: 0;">DICCIONARIO DE DATOS</h2>																						
ÁREA: <u> Análisis </u> CVE. PROYECTO: <u> IM103007 </u> CVE. REFERENCIA: <u> DD04-SAI-01 </u> PROYECTO: <u> Sistema de Auditoría Informática y Control de Centros de Tecnología de Información </u> ABREV.: <u> SAICOCETI </u> HOJA: <u> 22 </u> DE: <u> 22 </u>																						
NOMBRE DEL ARCHIVO	DESCRIPCIÓN	OBSERVACIONES																				
USUARIOS	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="background-color: #e0f0ff;">Nombre del campo</th> <th style="background-color: #e0f0ff;">Tipo de dato</th> <th style="background-color: #e0f0ff;">Precisión</th> <th style="background-color: #e0f0ff;">Nulo / No Nulo</th> <th style="background-color: #e0f0ff;">Tipo de Llave</th> </tr> </thead> <tbody> <tr> <td>login</td> <td>varchar</td> <td>15</td> <td>NOT NULL</td> <td>Pk</td> </tr> <tr> <td>password</td> <td>varchar</td> <td>40</td> <td>NOT NULL</td> <td></td> </tr> <tr> <td>tipo</td> <td>int</td> <td>8</td> <td>NOT NULL</td> <td></td> </tr> </tbody> </table>	Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave	login	varchar	15	NOT NULL	Pk	password	varchar	40	NOT NULL		tipo	int	8	NOT NULL		Login del usuario registrado Password del usuario registrado Tipo de usuario registrado
Nombre del campo	Tipo de dato	Precisión	Nulo / No Nulo	Tipo de Llave																		
login	varchar	15	NOT NULL	Pk																		
password	varchar	40	NOT NULL																			
tipo	int	8	NOT NULL																			
ELABORÓ: <u> José Ángel Haro Juárez </u> FECHA: <u> 06/12/2007 </u>	AUTORIZÓ: <u> Heriberto Olguin Romo </u> FECHA: <u> 10/12/2007 </u>																					

DESCRIPCIÓN DE PROCESOS



ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD05-SAI-01

PROYECTO: Sistema de Auditoria Informática y Control de Centros de Tecnología de Información ABREV. SAICOCET: HOJA: 1 DE 13

NOMBRE DEL PROCESO: Manejo de empresas

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Folio RFC Teléfono Fax Nombre CURP Email Calle Colonia Estado N° Ext N° Int C.P. Municipio	- Registro de una empresa en el sistema - Consulta de los datos de una empresa. - Archivo PDF de empresas	Administración de los datos relacionados con las empresas, mediante este proceso es posible almacenar, actualizar y eliminar información de las empresas registradas en el sistema, así como realizar consultas y crear archivos PDF.	Empresas Manejo de representantes	DD06-SAI-01 (Hoja 1) DD05-SAI-01 (Hoja 6)

ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD05-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV. SAICOCETI: 7 HOJA: 2 DE: 13

NOMBRE DEL PROCESO: Manejo de CTI		REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO
Empresa Folio Teléfono Fax Nombre Email Calle Colonia Estado N° Ext N° Int C.P. Municipio	- Registro de un CTI en el sistema - Consulta de los datos de un CTI - Archivo PDF de CTI	Administración de los datos relacionados con los CTI, mediante este proceso es posible almacenar, actualizar y eliminar información de los CTI registrados en el sistema, así como realizar consultas y crear archivos PDF.
		SUBPROCESOS UTILIZADOS CTI Manejo de representantes
		DD06-SAI-01 (Hoja 2) DD05-SAI-01 (Hoja 6)

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguín Romo

FECHA: 10/12/2007



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño CVE. PROYECTO: IM103007 CVE. REFERENCIA: DD05-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV. SAICOCETI: 7 HOJA: 3 DE: 13

NOMBRE DEL PROCESO: Manejo de auditores

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Folio RFC CURP Apellido paterno Apellido materno Nombre Fecha de alta Teléfono Tel. móvil Email Tipo Login Password Puntos Domicilio particular Domicilio oficina	<ul style="list-style-type: none"> - Registro de un auditor en el sistema - Consulta de los datos de un auditor - Archivo PDF de auditores 	Administración de los datos relacionados con los auditores, mediante este proceso es posible almacenar, actualizar y eliminar información de los auditores registrados en el sistema, así como realizar consultas y crear archivos PDF.	<ul style="list-style-type: none"> Auditores Manejo de personas en caso de emergencia Manejo de auditorías permitidas 	<ul style="list-style-type: none"> DD06-SAI-01 (Hoja 3) DD05-SAI-01 (Hoja 7) DD05-SAI-01 (Hoja 8)

<p>ELABORÓ: _____</p> <p>FECHA: _____</p>	<p style="text-align: center;">Rolando Lima Maciel</p> <p>AUTORIZÓ: _____</p> <p>FECHA: _____</p>
<p style="text-align: right;">Heriberto Olguin Romo</p>	



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD05-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV. SAICOCETI: 7 HOJA: 4 DE: 13

NOMBRE DEL PROCESO: Manejo de auditorías

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Folio Nombre Descripción	<ul style="list-style-type: none"> - Registro de una auditoría en el sistema - Consulta de los datos de una auditoría - Archivo PDF de auditoría 	Administración de los datos relacionados con las auditorías, mediante este proceso es posible almacenar, actualizar y eliminar información de las auditorías registradas en el sistema, así como realizar consultas y crear archivos PDF.	auditorías Manejo de preguntas	DD06-SAI-01 (Hoja 4) DD05-SAI-01 (Hoja 9)

ELABORÓ: <u>Rolando Lima Maciel</u> FECHA: <u>25/11/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
---	--



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño _____ **CVE. PROYECTO:** IN103007 _____ **CVE. REFERENCIA:** DD05-SAI-01 _____
PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información _____ **ABREVIATURA:** SAICOCETI _____ **HOJA:** 6 _____ **DE:** 13 _____

NOMBRE DEL PROCESO: Manejo de representantes

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Empresa ó CTI Folio Apellido paterno Apellido materno Nombre Teléfono Tel. móvil Email Cargo Título	- Registro de un representante en el sistema	Administración de los datos relacionados con los representantes de una empresa o CTI, mediante este proceso es posible almacenar, actualizar y eliminar información de los representantes registrados en el sistema.	Representantes	DD06-SAI-01 (Hoja 8)

ELABORÓ: _____ FECHA: 25/11/2007	AUTORIZÓ: _____ FECHA: 10/12/2007
Rolando Lima Maciel	Heriberto Olguin Romo

DESCRIPCIÓN DE PROCESOS



ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD05-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV. SAICOCETI 7 HOJA: 7 DE: 13

NOMBRE DEL PROCESO: **Manejo de personas en caso de emergencia**

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Auditor Apellido paterno (referencia principal) Apellido materno (referencia principal) Nombre (referencia principal) Teléfono (referencia principal) Tel. móvil (referencia principal) Apellido paterno (referencia principal) Apellido materno (referencia principal) Nombre (referencia principal) Teléfono (referencia principal) Tel. móvil (referencia principal)	- Registro de personas en caso de emergencia	Administración de los datos relacionados con las personas en caso de emergencia de un auditor o promotor, mediante este proceso es posible almacenar y actualizar información de las personas en caso de emergencia registradas en el sistema.	Registro de persona en caso de emergencia	DD06-SAI-01 (Hoja 9)

ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD05-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV. SAICOCETI: 8 DE: 13

NOMBRE DEL PROCESO: Manejo de auditorías permitidas			
INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS
Auditor Auditoría	- Conjunto de auditorías permitidas al auditor	Selección del conjunto de auditorías que un auditor tiene permitido llevar a cabo.	Asignación de auditorías DD06-SAI-01 (Hoja 10)
		REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO	

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo

FECHA: 10/12/2007



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD05-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV. SAICOCETI ⁷ HOJA: 9 DE: 13

NOMBRE DEL PROCESO: **Manejo de preguntas**

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Auditoría Número de pregunta o acción Pregunta o acción Guía o sugerencia al auditor Activa	- Conjunto de preguntas contenidas en una auditoría	Manejo del conjunto de preguntas que componen una auditoría. Se lleva a cabo el registro de nuevas preguntas así como actualización y eliminación de las mismas del sistema	Modificar preguntas Manejo de recomendaciones	DD06-SAI-01 (Hoja 11) DD05-SAI-01 (Hoja 11)

ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD05-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV. SAICOCETI: 7 HOJA: 10 DE: 13

NOMBRE DEL PROCESO: **Manejo de líder**

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Auditoría por realizar Auditor	- Líder a cargo de una auditoría por realizar	Selección del líder que será el responsable del manejo de una auditoría por realizar	Modificar auditores asignados	DD06-SAI-01 (Hoja 12)

ELABORÓ: Rolando Lima Maciel
FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
FECHA: 10/12/2007



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño CVE. PROYECTO N103007 CVE. REFERENCIA: DD05-SAI-01

PROYECTO: Sistema de Auditoria Informática y Control de Centros de Tecnología de Información ABREVS/AICOCETI 7 HOJA: 11 DE 13

NOMBRE DEL PROCESO: **Manejo de recomendaciones**

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Auditoría Número de pregunta o acción Número de recomendación Recomendación	- Conjunto de recomendaciones contenidas en una pregunta	Manejo del conjunto de recomendaciones que componen una pregunta Se lleva a cabo el registro de nuevas recomendaciones así como actualización y eliminación de las mismas del sistema.	Recomendaciones	DD06-SAI-01 (Hoja 13)

ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño **CVE. PROYECTO:** IN103007 **CVE. REFERENCIA:** DD05-SAI-01
PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información **ABREV. SAICOCET:** **HOJA:** 12 **DE:** 13

NOMBRE DEL PROCESO: Manejo de reporte			
INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS
Auditoría por realizar Número de pregunta o acción Observación	- Reporte de avances para las empresas	Actualización de las observaciones realizadas por los auditores así como de las recomendaciones asignadas a una pregunta o acción de los resultados registrados por los auditores	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO Recomendaciones por respuesta DD06-SAI-01 (Hoja 14) Modificar reporte DD06-SAI-01 (Hoja 15)

ELABORÓ: _____

FECHA: 25/11/2007

AUTORIZÓ: _____

FECHA: 10/12/2007

Rolando Lima Maciel Heriberto Olguín Romo



DESCRIPCIÓN DE PROCESOS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD05-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV. SAICOCETI HOJA: 13 DE: 13

NOMBRE DEL PROCESO: **Continuar auditoría o iniciarla**

INSUMOS DE ENTRADA	PRODUCTO	DESCRIPCIÓN DEL PROCESO	SUBPROCESOS UTILIZADOS	REFERENCIA DONDE SE DESCRIBE CADA SUBPROCESO
Auditoría por realizar Auditor Número de pregunta o acción Comentarios Nivel Hora de inicio Hora de término	- Observaciones de los auditores a las preguntas que componen una auditoría	Los auditores registran en el sistema las observaciones generadas como resultado de la auditoría en cuestión	Continuar auditoría o iniciarla	DD06-SAI-01 (Hoja 16)

ELABORÓ: <u>Rolando Lima Maciel</u> FECHA: <u>25/11/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
---	--



ENTRADAS Y SALIDAS

ÁREA: _____ Diseño _____ CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 3_B DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: Audidores		<input checked="" type="checkbox"/> ENTRADA	<input checked="" type="checkbox"/> SALIDA
TABLA PRINCIPAL ASOCIADA AL MÓDULO: audidores			
OTRAS TABLAS RELACIONADAS: municipios, estados			
CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VÁLIDOS
auditores.cptof	Edición, despliegue	Números	Se válida que siempre se ingresen cinco números
auditores.estadoof	Despliegue, edición	No aplica (selección por lista)	No aplica (selección por lista)
auditores.municipioof	Despliegue, edición	No aplica (selección por lista)	No aplica (selección por lista)
auditores.telof	Edición, despliegue	Números	Válida números y espacios en blanco
auditores.email	Edición, despliegue	Letras ó números ó " " ó " " ó "@"	Se válida que se tenga la siguiente secuencia: cualquier combinación entre letras, números, " " ó " " posteriormente "@" seguido de cualquier combinación entre letras, números, " " ó " "
auditores.cel	Despliegue	No aplica (selección por lista)	No aplica (selección por lista)
auditores.tipo	Despliegue	No aplica (selección por lista)	No aplica (selección por lista)
municipios.nombre	Despliegue	No aplica (selección por lista)	No aplica (selección por lista)
estados.nombre	Despliegue	No aplica (selección por lista)	No aplica (selección por lista)
auditores.num_intofi	Edición, despliegue	Sin restricción	Cualquier carácter
auditores.coloniaofi	Edición, despliegue	Sin restricción	Cualquier carácter
auditores.calleofi	Edición, despliegue	Sin restricción	Cualquier carácter
auditores.num_extofi	Edición, despliegue	Sin restricción	Cualquier carácter
auditores.tel	Edición, despliegue	Números	Válida números y espacios en blanco

ELABORÓ:	Rolando Lima Maciel	AUTORIZÓ:	Heriberto Olguín Romo
FECHA:	25/11/2007	FECHA:	10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 4 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: **Auditorías** ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: **auditorías**

OTRAS TABLAS RELACIONADAS: **No aplica**

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VÁLIDOS
auditorias_folio	Edición, despliegue	Números	De 0 a 9223372036854775807
auditorias_nombre	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco
auditorias_descripcion	Edición, despliegue	Sin restricción	Cualquier carácter

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguín Romo

FECHA: 10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 6 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: **Bitácora** ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: **resultados**

OTRAS TABLAS RELACIONADAS:

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VÁLIDOS
empresas.nombre	Despligue	Letras y espacios en blanco	No aplica
centros.nombre	Despligue	Letras y espacios en blanco	No aplica
auditorias.nombre	Despligue	Letras y espacios en blanco	No aplica
resultados.horainicio	Despligue	dd/Nombre mes/aaaa hh:mm:ss	No aplica
resultados.horafin	Despligue	dd/Nombre mes/aaaa hh:mm:ss	No aplica
tiempo total	Despligue	dd/Nombre mes/aaaa hh:mm:ss	Se obtiene restando la hora de inicio a la hora final

ELABORÓ:

Rolando Lima Maciel

AUTORIZÓ: Heriberto Olguin Romo

FECHA:

25/11/2007

FECHA:

10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 7 DE: 19

NOMBRE DEL MÓDULO: Reporte		<input type="checkbox"/> ENTRADA	<input checked="" type="checkbox"/> SALIDA
TABLA PRINCIPAL ASOCIADA AL MÓDULO: resultados			
OTRAS TABLAS RELACIONADAS: empresas, centros, auditorias, preguntas, auditorias por realizar			
CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VALIDOS
empresas.nombre	Despligue	Letras y espacios en blanco	No aplica
centros.nombre	Despligue	Letras y espacios en blanco	No aplica
auditorias.nombre	Despligue	Letras y espacios en blanco	No aplica
preguntas.pregunta	Despligue	Letras y espacios en blanco	No aplica
resultados.observacion	Despligue	Sin restricción	No aplica
resultados.nivel	Despligue	Números	No aplica
auditorias por realizar.fechaini	Despligue	dd/Nombre mes/aaaa	No aplica

ELABORÓ: <u>Rolando Lima Maciel</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u>
FECHA: <u>25/11/2007</u>	FECHA: <u>10/12/2007</u>



ENTRADAS Y SALIDAS

ÁREA: _____ Diseño _____ CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 8 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: **Representantes** ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: **representantes**

OTRAS TABLAS RELACIONADAS: **empresas, centros, representantes.cti, representante empresa**

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VÁLIDOS
representantes.folio	Edición, despliegue	Números	De 0 a 92233372036854775807
representantes.cargo	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco
representantes.titulo	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco
representantes.nombre	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco
representantes.appat	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco
representantes.apmat	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco
representantes.tel	Edición, despliegue	Números	Válida números y espacios en blanco
representantes.cel	Edición, despliegue	Números	Válida números y espacios en blanco
representantes.email	Edición, despliegue	Letras ó números ó " " ó " " ó "@"	Se válida que se tenga la siguiente secuencia: cualquier combinación entre letras, números, " " ó " " posteriormente "@" seguido de cualquier combinación entre letras, números y " "
representantesempresa.folio	Edición, despliegue	Números	De 0 a 92233372036854775807
representantesempresa_empresa	Edición, despliegue	Números	De 0 a 92233372036854775807
representantescti.folio	Edición, despliegue	Números	De 0 a 92233372036854775807
representantescti_empresa	Edición, despliegue	Números	De 0 a 92233372036854775807
representantescti.cti	Edición, despliegue	Números	De 0 a 92233372036854775807

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo

FECHA: 10/12/2007



ENTRADAS Y SALIDAS

ÁREA: _____ Diseño _____ CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 9 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: Registro de persona en caso de emergencia ENTRADA SALIDA
 TABLA PRINCIPAL ASOCIADA AL MÓDULO: personasemergencia

OTRAS TABLAS RELACIONADAS: No aplica

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VÁLIDOS
personasemergencia.auditor	Edición, despliegue	Números	De 0 a 9223372036854775807
personasemergencia.folio	Edición, despliegue	Números	De 0 a 9223372036854775807
personasemergencia.nombre	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco
personasemergencia.cel	Edición, despliegue	Números	Válida números y espacios en blanco
personasemergencia.tel	Edición, despliegue	Números	Válida números y espacios en blanco
personasemergencia.appat	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco
personasemergencia.apmat	Edición, despliegue	Letras y espacios en blanco	Letras y espacios en blanco

ELABORÓ: _____

Rolando Lima Maciel

FECHA: _____

25/11/2007

AUTORIZÓ: _____

Heriberto Olguín Romo

FECHA: _____

10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 10 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: **Asignación de auditorías** ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: **auditoresen**

OTRAS TABLAS RELACIONADAS: **auditorias**

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VÁLIDOS
auditorias.nombre	Despliegue	Letras y espacios en blanco	No aplica
auditoresen.auditor	Edición, despliegue	Números	De 0 a 9223372036854775807
auditoresen.auditoria	Edición, despliegue	Números	De 0 a 9223372036854775807

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguín Romo

FECHA: 10/12/2007



ENTRADAS Y SALIDAS

ÁREA: _____ Diseño _____ CVE. PROYECTO: IM103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 11 DE: 19

NOMBRE DEL MÓDULO: **Modificar preguntas** ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: **preguntas**

OTRAS TABLAS RELACIONADAS: **No aplica**

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VALIDOS
preguntas.folio	Edición, despliegue	Números	De 0 a 9223372036854775807
preguntas.pregunta	Edición, despliegue	Sin restricción	Cualquier carácter
preguntas.sugerencia	Edición, despliegue	Sin restricción	Cualquier carácter
preguntas.activa	Edición, despliegue	No aplica (selección por lista)	No aplica (selección por lista)

ELABORÓ: _____

Rolando Lima Maciel

AUTORIZÓ: _____

Heriberto Olguin Romo

FECHA: _____

25/11/2007

FECHA: _____

10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE: PROYECTO: IN103007 CVE REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 12 DE 19
 Centros de Tecnología de Información

NOMBRE DEL MÓDULO: **Modificar auditores asignados/ Seccionar grupo de trabajo** ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: **auditoriasporaauditores**

OTRAS TABLAS RELACIONADAS: **auditores**

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VALIDOS
auditor	Edición, despliegue	Números	De 0 a 9223372036854775807
nivel	Edición, despliegue	No aplica (selección por lista)	No aplica (selección por lista)
sts	Edición, despliegue	No aplica (selección por lista)	No aplica (selección por lista)
folio	Edición, despliegue	Números	De 0 a 9223372036854775807
auditores.nombre	Despliegue	Letras y espacios en blanco	No aplica

ELABORÓ:

Rolando Lima Maciel

AUTORIZÓ: Heriberto Olguin Romo

FECHA:

25/11/2007

FECHA: 10/12/2007



ENTRADAS Y SALIDAS

ÁREA: _____ Diseño _____ CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 13 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: Recomendaciones ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: recomendaciones

OTRAS TABLAS RELACIONADAS:

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VÁLIDOS
recomendaciones.auditoria	Edición	Números	De 0 a 9223372036854775807
recomendaciones.pregunta	Edición	Números	De 0 a 9223372036854775807
recomendaciones.folio	Edición, despliegue	Números	De 0 a 9223372036854775807
recomendaciones.recomendación	Edición, despliegue	Sin restricción	Cualquier carácter

ELABORÓ: _____

Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: _____ Heriberto Olguín Romo

FECHA: 10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 14 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: Recomendaciones por respuesta ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: recomendaciones por respuesta

OTRAS TABLAS RELACIONADAS: recomendaciones

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VÁLIDOS
recomendaciones por respuesta comendacion	Edición, despliegue	Números	De 0 a 9223372036854775807
recomendaciones por respuesta spuesta	Edición, despliegue	Números	De 0 a 9223372036854775807
recomendaciones por respuesta egunta	Edición, despliegue	Números	De 0 a 9223372036854775807
recomendaciones por respuesta ditoria	Edición, despliegue	Números	De 0 a 9223372036854775807
recomendaciones.recomendacion	Despliegue	Letras y espacios en blanco	No aplica

ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguín Romo
 FECHA: 10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE. PROYECTO: IM103007 CVE REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 15 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: **Modificar reporte** ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: **resultados**

OTRAS TABLAS RELACIONADAS: **preguntas**

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VALIDOS
resultados.pregunta	Despliegue	No aplica (selección por lista)	No aplica (selección por lista)
resultados.observacion	Edición, despliegue	Sin restricción	Cualquier carácter
resultados.nivel	Edición, despliegue	No aplica (selección por lista)	No aplica (selección por lista)
preguntas.pregunta	Despliegue	Letras y espacios en blanco	No aplica

ELABORÓ:

Rolando Lima Maciel

AUTORIZÓ: Heriberto Olguin Romo

FECHA:

25/11/2007

FECHA:

10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 16 DE: 19

NOMBRE DEL MÓDULO: Continuar auditoría o iniciarla		<input checked="" type="checkbox"/> ENTRADA	<input checked="" type="checkbox"/> SALIDA
TABLA PRINCIPAL ASOCIADA AL MÓDULO: resultados			
OTRAS TABLAS RELACIONADAS: auditorias, preguntas			
CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VALIDOS
auditoria.nombre	Despliegue	Letras y espacios en blanco	No aplica
preguntas.pregunta	Despliegue	Sin restricción	No aplica
preguntas.sugerencia	Despliegue	Sin restricción	No aplica
resultados.pregunta	Edición, despliegue	Números	De 0 a 92233372036854775807
resultados.observacion	Edición, despliegue	Números	De 0 a 92233372036854775807
resultados.nivel	Edición, despliegue	Sin restricción	Cualquier carácter
resultados.horainicio	Edición, despliegue	Números	De 0 a 92233372036854775807
resultados.horafin	Edición, despliegue	hh:mm	Números
resultados.folio	Edición, despliegue	hh:mm	Números
resultados.auditoria	Edición, despliegue	Números	De 0 a 92233372036854775807

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo

FECHA: 10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE. PROYECTO: IM103007 CVE. REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 17 DE: 19
Centros de Tecnología de Información

NOMBRE DEL MÓDULO: Históricos ENTRADA SALIDA

TABLA PRINCIPAL ASOCIADA AL MÓDULO: auditoresporauditores

OTRAS TABLAS RELACIONADAS: auditores, auditoria

CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO	VALORES VALIDOS
auditores.nombre	Despliegue	Letras y espacios en blanco	No aplica
auditoresporauditores.nivel	Despliegue	Letras y espacios en blanco	No aplica
auditoresporauditores.sts	Despliegue	Letras y espacios en blanco	No aplica
auditoresporauditores.obs	Despliegue	Letras y espacios en blanco	No aplica
auditores.nombre	Despliegue	Letras y espacios en blanco	No aplica
auditores.appat	Despliegue	Letras y espacios en blanco	No aplica
auditores.apmat	Despliegue	Letras y espacios en blanco	No aplica

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo

FECHA: 10/12/2007



ENTRADAS Y SALIDAS

ÁREA: Diseño CVE: PROYECTO: IN103007 CVE REFERENCIA: DD06-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de ABREV.: SAICOCETI HOJA: 18 DE 19
Centros de Tecnología de Información

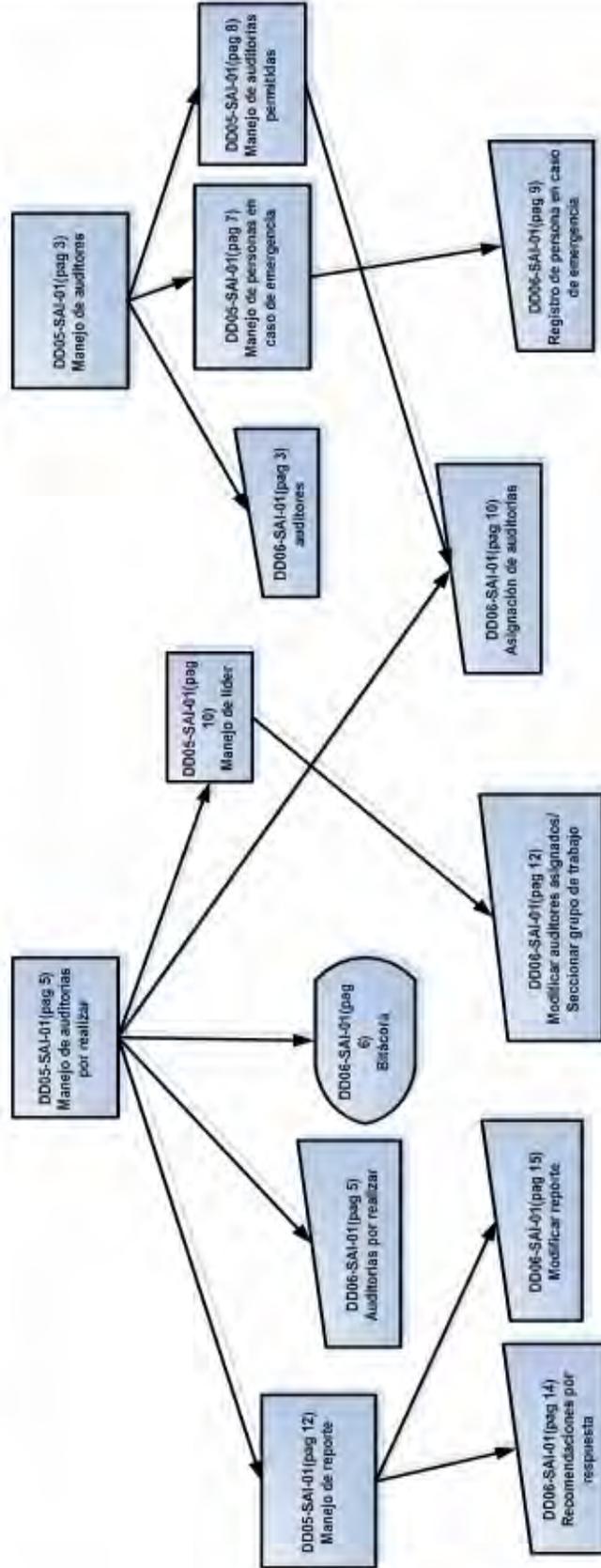
NOMBRE DEL MÓDULO: Sugerir preguntas adicionales		<input checked="" type="checkbox"/> ENTRADA <input checked="" type="checkbox"/> SALIDA
TABLA PRINCIPAL ASOCIADA AL MÓDULO: sugerenciaspreguntas		
OTRAS TABLAS RELACIONADAS: auditorias		
CAMPOS Y VARIABLES	OPERACIONES ASOCIADAS	FORMATO
auditorias.nombre	Despliegue	Letras y espacios en blanco
sugerenciaspreguntas.folio	Edición, despliegue	Números
sugerenciaspreguntas.auditor	Edición, despliegue	Números
sugerenciaspreguntas.auditoriasis	Edición, despliegue	Números
sugerenciaspreguntas.pregunta	Edición, despliegue	Números
sugerenciaspreguntas.sugerencia	Edición, despliegue	Sin restricción
sugerenciaspreguntas.leida	Edición, despliegue	Falso o verdadero
sugerenciaspreguntas.auditoriasul	Edición, despliegue	Números
		VALORES VALIDOS
		No aplica
		De 0 a 9223372036854775807
		Cualquier carácter
		true ó false
		De 0 a 9223372036854775807

ELABORÓ: <u>Rolando Lima Maciel</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u>
FECHA: <u>25/11/2007</u>	FECHA: <u>10/12/2007</u>



DIAGRAMA ESTRUCTURAL DEL SISTEMA

ÁREA: Diseño CVE PROYECTO: IN10300Z CVE REFERENCIA: DD07-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de ABREV: SAICOCETI HOJA: 1 DE: 3
Centros de Tecnología de Información



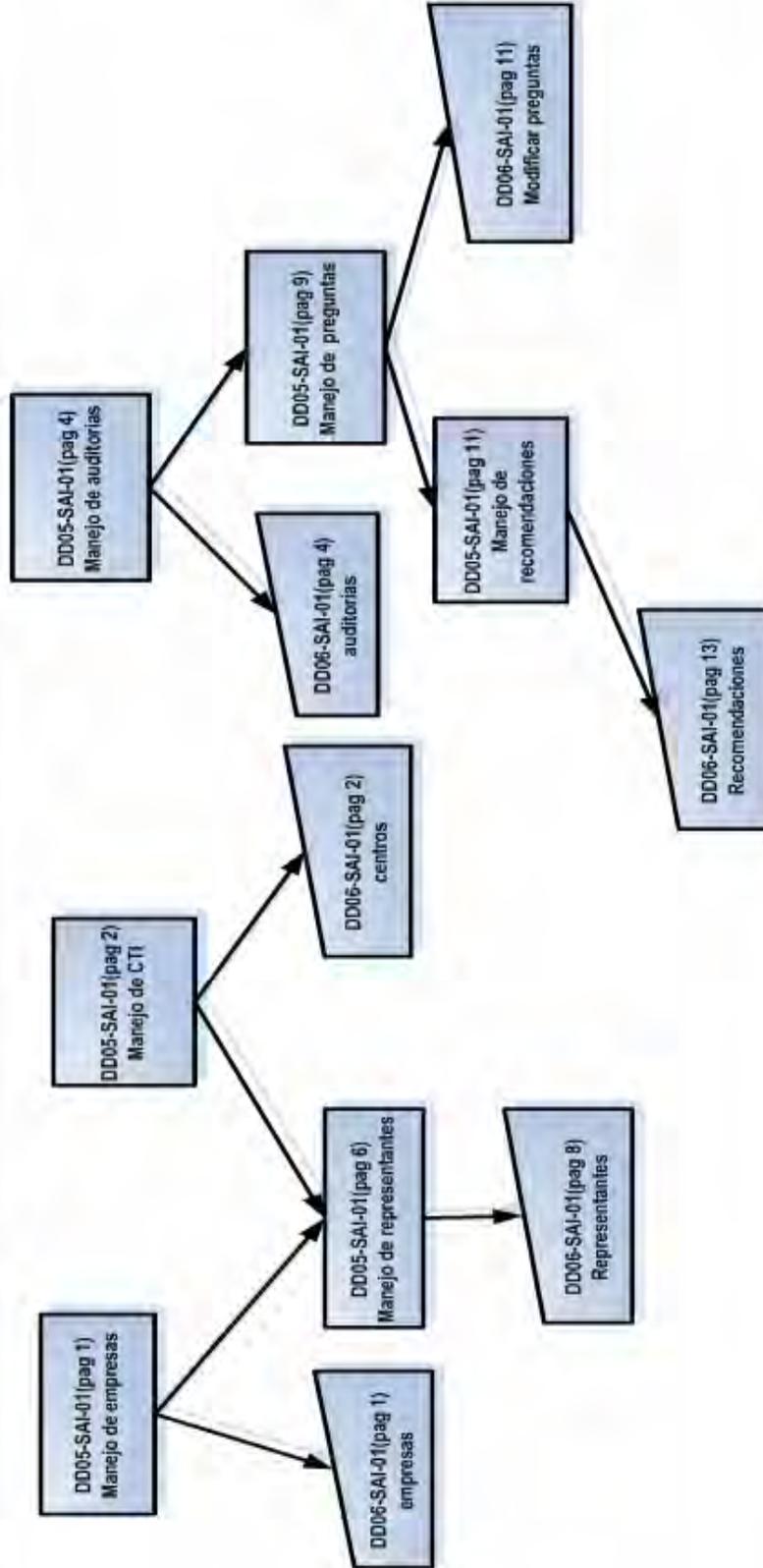
ELABORÓ: Rolando Lima Maciel
 FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
 FECHA: 10/12/2007

DIAGRAMA ESTRUCTURAL DEL SISTEMA



ÁREA: Diseño CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD07-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 2 DE: 3



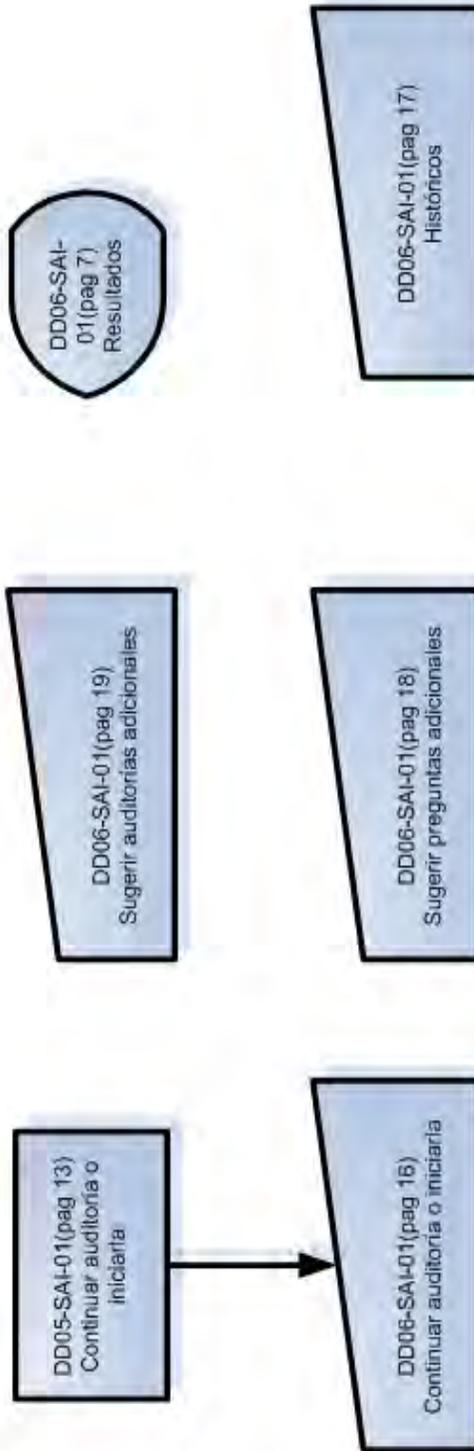
ELABORÓ: <u> Rolando Lima Maciel </u>	AUTORIZÓ: <u> Heriberto Olguin Romo </u>
FECHA: <u> 25/11/2007 </u>	FECHA: <u> 10/12/2007 </u>



DIAGRAMA ESTRUCTURAL DEL SISTEMA

ÁREA: _____ Diseño _____ CVE. PROYECTO: IN103007 CVE. REFERENCIA: DD07-SAI-01

PROYECTO Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCEI HOJA: 3 DE: 3



ELABORÓ: Rolando Lima Maciel
FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguin Romo
FECHA: 10/12/2007

DESCRIPCIÓN DE PROGRAMAS Y RUTINAS



ÁREA: Programación CVE. PROYECTO: IN103007 CVE. REFERENCIA: DP01-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE: 1

NOMBRE ELEMENTO	PROG./RUTINA	NOMBRE ARCHIVO		Parámetros de entrada		Valor de retorno		CLAVE PROCESO	Nº. DE LINEAS CÓDIGO	DESCRIPCIÓN
		Nombre	Tipo	Desc.	Nombre	Tipo	Desc.			
Se sustituye por JavaDoc										

ELABORÓ: <u>Rolando Lima Maciel</u> FECHA: <u>25/11/2007</u>	AUTORIZÓ: <u>Heriberto Olguin Romo</u> FECHA: <u>10/12/2007</u>
---	--

RESULTADO DE PRUEBAS



ÁREA: Pruebas CVE. PROYECTO: IN103007 CVE. REFERENCIA: DI01-SAI-01

PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE 2

DESCRIPCIÓN DE LAS PRUEBAS REALIZADAS

- 1.- Se validó el acceso al sistema de administración, se ingresó un login y password inválidos y posteriormente otros válidos.
- 2.- Se realizó el registro de una empresa, un CTI, un auditor, una auditoría (con una pregunta) y una auditoría por realizar (con asignación de líder) verificando todas las validaciones indicadas en el formato DD06-SAI-01.
- 3.- Se realizaron búsquedas de empresas, CTI, históricos, auditores, auditorías y auditorías por realizar y posteriormente se crearon los archivos pdf de cada una de la busquedas.
- 4.- Se realizó la actualización de una empresa, un CTI, un auditor, una auditoría (con una pregunta) y una auditoría por realizar (con asignación de líder) verificando todas las validaciones indicadas en el formato DD06-SAI-01.
- 5.- Se realizó la eliminación de una empresa, un CTI, un auditor, una auditoría (con una pregunta) y una auditoría por realizar (con asignación de líder).
- 6.- Se validó el acceso al sistema web para empresas y auditores.
- 7.- Se probó que una empresa pudiera consultar avances

DESCRIPCIÓN DE RESULTADOS

- 1.- Para este proceso se ejecutó el sistema y se ingresó como login la palabra "casa" y password "asac" y el sistema negó el acceso, posteriormente se ingreso el login admin y password "nimda" y se desplegó la pantalla principal del sistema.
- 2.- Para esta prueba se agregaron nuevos registros en el sistema (especificados en la sección de descripción de las pruebas realizadas) en primera instancia se ingresaron datos inválidos de acuerdo a lo indicado en el formato DD06-SAI-01 y el sistema no permitió agregar los registros posteriormente se ingresaron datos correctos y el sistema permitió agregar los nuevos registros.
- 3.- Se realizaron consultas de cada una de las pantallas (especificados en la sección de descripción de las pruebas realizadas) dejando todos los campos vacíos y el sistema devolvió todos los datos registrados en la base de datos. Posteriormente de crearon los archivos en formato pdf los cuales contenían todos los datos mostrados en cada una de las pantallas.
- 4.- Para esta prueba se actualizaron datos de registros existentes en el sistema (especificados en la sección de descripción de las pruebas realizadas) en primera instancia se ingresaron datos inválidos de acuerdo a lo indicado en el formato DD06-SAI-01 y el sistema no permitió actualizar los registros posteriormente se ingresaron datos correctos y el sistema permitió llevar a cabo la actualización.
- 5.- Se eliminaron datos de registros existentes en el sistema (especificados en la sección de descripción de las pruebas realizadas) y posteriormente se realizó una consulta y se comprobó que la eliminación de los registros.

NECESIDAD DE NUEVAS PRUEBAS
 PRUEBAS SATISFACTORIAS

FECHA PARA LAS SIGUIENTES PRUEBAS: N/A

ELABORÓ: Rolando Lima Maciel

FECHA: 25/11/2007

AUTORIZÓ: Heriberto Olguín Romo

FECHA: 10/12/2007

	<h2 style="margin: 0;">RESULTADO DE PRUEBAS</h2>
<p>ÁREA: Pruebas CVE. PROYECTO: <u>IN103007</u> CVE. REFERENCIA: <u>DI01-SAI-01</u></p> <p>PROYECTO: <u>Sistema de Auditoría Informática y Control de Centros de Tecnología de Información</u> ABREV.: <u>SAICOCETI</u> HOJA: <u>2</u> DE: <u>2</u></p>	<p style="text-align: center;">DESCRIPCIÓN DE LAS PRUEBAS REALIZADAS</p> <p>8.- Se probó que un auditor realizara sugerencias de auditorías. 9.- Se probó que un auditor realizara sugerencias de preguntas. 10.- Se realizó el proceso de registro de observaciones.</p>
<p style="text-align: center;">DESCRIPCIÓN DE RESULTADOS</p> <p>6.(continuación).- Se ingreso a la sección de auditores y se escribió como login "promotor" con password "rotomorp" y no se permitió el acceso posteriormente se escribió el login "auditor" y el password "rotidua" y se desplegó la pantalla principal de auditores. 7.- En la pantalla de empresa se accedió a la opción de "Consultar avances" y se desplegó la lista de preguntas para las cuales ya existían observaciones. 8.- En la pantalla de auditores se accedió a la opción para sugerir nuevas auditorías y se ingresaron los datos solicitados y se registraron los datos en la base de datos, verificando lo anterior realizando la consulta. 9.- En la pantalla de auditores se accedió a la opción para sugerir nuevas preguntas y se ingresaron los datos solicitados y se registraron los datos en la base de datos, verificando lo anterior realizando la consulta. 10.- En la pantalla de auditores se accedió a la opción para iniciar o continuar una auditoría y se ingresaron los datos necesarios para registrar las observaciones para tres preguntas y se registraron los datos en la base de datos, verificando lo anterior realizando la consulta</p>	<p>FECHA PARA LAS SIGUIENTES PRUEBAS: <u>N/A</u></p>
<p><input type="checkbox"/> NECESIDAD DE NUEVAS PRUEBAS</p> <p><input checked="" type="checkbox"/> PRUEBAS SATISFACTORIAS</p>	<p>AUTORIZÓ: <u>Heriberto Olguin Romo</u></p> <p>FECHA: <u>10/12/2007</u></p>
<p>ELABORÓ: <u>Rolando Lima Maciel</u></p> <p>FECHA: <u>25/11/2007</u></p>	<p>AUTORIZÓ: <u>Heriberto Olguin Romo</u></p> <p>FECHA: <u>10/12/2007</u></p>



INTEGRACIÓN AL BANCO DE SISTEMAS

ÁREA: Análisis CVE. PROYECTO: IN103007 CVE. REFERENCIA: DI02-SAI-01
 PROYECTO: Sistema de Auditoría Informática y Control de Centros de Tecnología de Información ABREV.: SAICOCETI HOJA: 1 DE 1

RESPONSABLE DEL SISTEMA

NOMBRE: Heriberto Olguín Romo TELÉFONO: N/A
 PUESTO: N/A FAX: N/A
 DOMICILIO: N/A
 E-MAIL: N/A TEL. CEL.: N/A

DESCRIPCIÓN DEL SISTEMA

El SAICOCETI emplea solamente herramientas de software libre para que la Facultad de Ingeniería de la UNAM pueda llevar a cabo auditorías a centros de tecnología de información empelando como la herramienta central el sistema a construir. El sistema será una combinación entre diversas tecnologías ya que contará con módulos para trabajar vía Web y computadoras personales.

Dentro de los principales componentes de software libre que se usan son:
 Lenguaje de programación "Java", Manejador de base de datos "Postgres"
 Framework "Ibatis"

ELEMENTOS ENTREGADOS

		FORMAS:			
<input checked="" type="checkbox"/>	PROGRAMAS FUENTE	<input checked="" type="checkbox"/> DA03	<input checked="" type="checkbox"/> DD02	<input checked="" type="checkbox"/> DD04	<input checked="" type="checkbox"/> DP01
<input checked="" type="checkbox"/>	PROGRAMAS EJECUTABLES	<input checked="" type="checkbox"/> DA04	<input checked="" type="checkbox"/> DD03A	<input checked="" type="checkbox"/> DD05	
<input checked="" type="checkbox"/>	MANUAL TÉCNICO	<input checked="" type="checkbox"/> DA05	<input checked="" type="checkbox"/> DD03B	<input checked="" type="checkbox"/> DD06	
<input checked="" type="checkbox"/>	MANUAL DEL USUARIO	<input checked="" type="checkbox"/> DD01	<input checked="" type="checkbox"/> DD03C	<input checked="" type="checkbox"/> DD07	

OTROS: _____

ENTREGÓ: José Ángel Haro Juárez

FIRMA: _____

RECIBIÓ: Heriberto Olguín Romo

FIRMA: _____

CAPÍTULO 8: MANUALES DE ADMINISTRACIÓN E INSTALACIÓN

En este capítulo se presentan tanto los manuales de administración como los de instalación del SAICOCETI, con objeto de facilitar el manejo del mismo en sus diferentes módulos.

8.1 REQUERIMIENTOS MÓDULO DESKTOP

A continuación se presentan los requerimientos para llevar a cabo correctamente la instalación del sistema.

8.1.1 REQUERIMIENTOS DE HARDWARE

- Procesador: Pentium II o superior.
- Memoria RAM: 64 MB mínimo 128 MB recomendable.
- Espacio en Disco Duro: 10 MB.
- Tarjeta de red o módem para acceso a Internet.

8.1.2. REQUERIMIENTOS DE SOFTWARE

- Sistema Operativo: Cualquier sistema operativo que soporte Java 1.5 o superior.
- Paquetería:
 - Máquina virtual de Java 1.5 o superior.
 - Navegador de Internet.
 - Visor de archivos PDF.
 - Visor de archivos RTF.

8.1.3. REQUERIMIENTOS ADICIONALES

- Acceso a Internet (obligatorio)

8.2 INSTALACIÓN MÓDULO DESKTOP EN WINDOWS

En esta sección se presenta el proceso de instalación del SAICOCETI en el sistema operativo Windows.

8.2.1 ACCESO AL AUTORUN

Mediante el CD que contiene el software de instalación se mostrará el siguiente menú:



Figura 8.1. Menú del autorun.

Descripción de las opciones:

- Manual de instalación: Acceso al presente documento.
- Manual de Usuario: Acceso al manual del usuario.
- Instalación de Java: Instalación de Java en caso de que no esté instalado en la computadora.
- Instalación del SAICOCETI: Instalación del software asociado con el SAICOCETI.
- Salir: Para salir de la ventana que muestra el menú.

8.2.2 INSTALACIÓN DEL SAICOCETI

Simplemente dar click en la opción de Instalación de SAICOCETI; a continuación del siguiente menú se muestran las pantallas del proceso:



Figura 8.2. Instalación del SAICOCETI.

Una vez realizado lo anterior se mostrará la siguiente pantalla:



Figura 8.3. Inicio de instalación.

Dar click en el botón de “siguiente” y se verá la pantalla:



Figura 8.4. Selección del directorio de instalación.

En la pantalla anterior se pregunta en qué directorio se desea instalar el SAICOCETI, es posible, si se desea, cambiar el directorio que muestra por defecto, una vez que se especificó el directorio de instalación hay que dar click en el botón de “siguiente” y se mostrará la pantalla:



Figura 8.5. Creación del directorio de instalación.

En la pantalla anterior se indica que el directorio de instalación no existe, y se pregunta si se quiere crear, hay que dar click en el botón de “Sí”, y luego se mostrará:



Figura 8.6. Grupo de programas de instalación.

En la ventana anterior se pregunta acerca del grupo de programas en el que se desea aparezca el SAICOCETI en el menú de inicio, hay que dejar todo tal y como se muestra para posteriormente dar click en botón de “siguiente” y aparece: la ventana:



Figura 8.7. Resumen de instalación.

En la que se muestra el resumen de la instalación, de nuevo hay que dar click en el botón de “siguiente” posteriormente se llevará a cabo la instalación del SAICOCETI, y cuando finalice, se verá la siguiente pantalla:



Figura 8.5. Fin de instalación.

En esta pantalla se indica que la instalación ha finalizado y por último dar click en el botón que dice “Finalizar”.

8.2.3 CONFIGURACIÓN DE BASE DE DATOS

En la ruta de instalación del SAICOCETI hay que localizar un archivo llamado “siacoceti.properties” y abrirlo con cualquier editor de texto plano y se verán las siguientes líneas:

```
server=localhost
base=SAICOCETI
```

Se debe editar la primera línea y sustituir la palabra “localhost” por la dirección del servidor de base de datos del SAICOCETI, proporcionada por el administrador del sistema.

8.2.4 ACCESO AL SAICOCETI

Una vez instalado el sistema SAICOCETI, la manera de acceder al programa es la siguiente: En el menú “Inicio” seleccionar la opción de “Todos los programas” posteriormente hay que buscar el submenú llamado “SAICOCETI” y finalmente dar click en la opción “SAICOCETI” representada por el siguiente icono:



Figura 8.9. Ícono del SAICOCETI.

Tal y como se muestra en la siguiente pantalla:



Figura 8.10. Menú del SAICOCETI.

NOTA: Para que el SAICOCETI funcione se deberá tener acceso a Internet.

Después de haber realizado los pasos anteriores se mostrará la siguiente pantalla:



Figura 8.11. Inicio del SAICOCETI.

En la pantalla anterior se debe proporcionar un Login y Password, en caso de no contar con una cuenta de acceso es necesario contactar al administrador del SAICOCETI, para que se asigne una cuenta.

Si la información tecleada para el Login y Password es incorrecta se mostrará un mensaje como el que sigue:

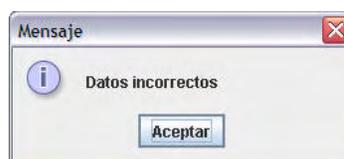


Figura 8.12. Error de Acceso.

Pero si la información es correcta se mostrará la siguiente pantalla:

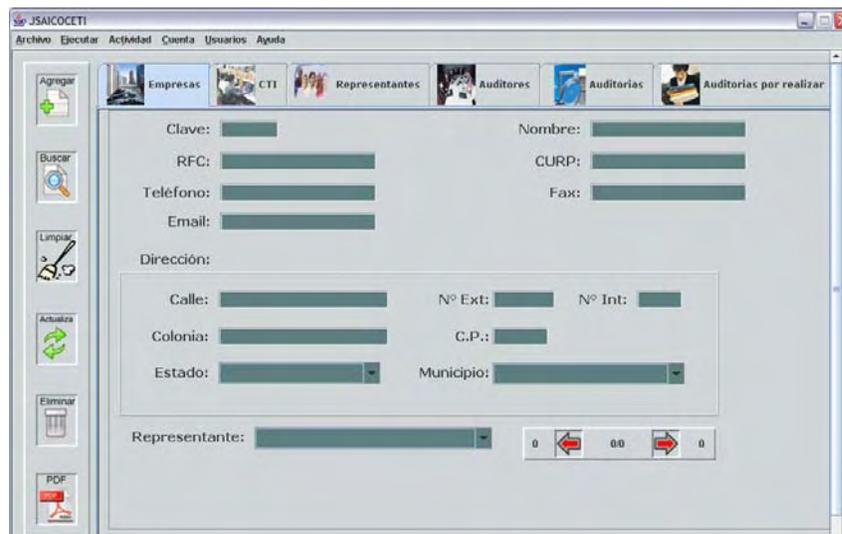


Figura 8.13. Pantalla principal.

Si se presenta alguna duda sobre el funcionamiento del sistema es posible consultar la ayuda mediante el menú “ayuda” y posteriormente en la opción “manual de usuario”, o bien, con las teclas ALT+Y y luego ALT+M.

8.3 INSTALACIÓN MÓDULO DESKTOP EN OTRO SISTEMA OPERATIVO

Debido a que el SAICOCETI fue construido con Java, es posible ejecutarlo en cualquier sistema operativo que cuente con soporte para Java. El procedimiento de instalación es muy sencillo ya que consiste en los tres pasos siguientes:

1. Instalar Java versión 1.5 ó superior (Solamente en caso de no tenerlo previamente instalado).
2. Descomprimir los archivos del SAICOCETI, los cuales se proporcionan en tres formatos: zip, tar y gz.
3. Ejecutar la siguiente sentencia en la carpeta donde de descomprimieron los archivos del SAICOCETI:

```
java -jar JSIACOCETI.jar
```

8.4 MANUAL DE USUARIO

A continuación se describe como llevar a cabo las tareas generales del sistema, como son: agregar, buscar actualizar o eliminar datos, lo anterior solamente se explicará con la sección de empresas ya que el procedimiento a seguir es el mismo con todas las pantallas. Además se darán detalles otros procesos más específicos del sistema.

8.4.1 ACCESO A EMPRESAS

Para acceder a esta pantalla dar click sobre la pestaña titulada como Empresas:



Figura 8.14. Pantalla principal del SAICOCETI.

O bien, seleccionando el menú de Actividad y posteriormente la opción de Empresas:



Figura 8.15. Acceso a empresas.

Otra opción es presionando las teclas ALT+T y después ALT+E.

8.4.2 AGREGAR EMPRESAS

Todos los datos son obligatorios, a excepción del Folio, el cual se asigna en forma automática. El Fax es opcional. Para seleccionar la Entidad Federativa se hace mediante la lista que presenta el combo:

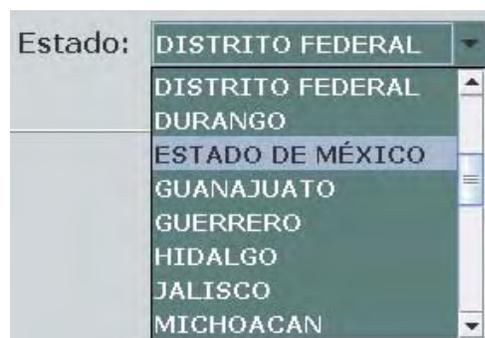


Figura 8.16. Selección de Entidad Federativa.

Después de haber elegido una Entidad Federativa, podrá seleccionarse el Municipio respectivo:



Figura 8.17. Selección del Municipio.

Realizados los pasos anteriores dar click en el botón Agregar para guardar los datos de la empresa.



Figura 8.18. Agregar.

Observando lo anterior se podrán guardar los datos de la empresa exitosamente. En caso de que se presente algún problema, se mostrará un mensaje indicando el error.

8.4.3 BUSCAR EMPRESAS

Se pueden mostrar todos los datos de una Empresa, tecleando cualquiera de ellos.



Figura 8.19. Buscar.

Para localizar puntualmente una empresa se tecléa el Folio. Para que se muestren todas las empresas por número de Folio, simplemente dar click en el botón de Buscar. Si un dato se repite en varias empresas se mostrarán estas, por número de Folio.



Figura 8.20. Buscar Empresa.

En el siguiente panel se indica el registro 9 de un total de 47 y es posible desplazarse al registro que se requiera.

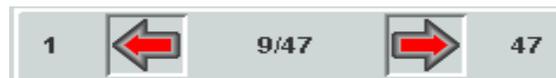


Figura 8.21. Panel de desplazamiento.

8.4.4 LIMPIAR EMPRESAS

Mediante el botón "Limpiar" se inicializan todos los campos, con objeto de introducir nuevos datos.



Figura 8.22. Limpiar.

8.4.5 ACTUALIZAR EMPRESAS

Todos los datos son obligatorios, a excepción del Fax que es opcional.



Figura 8.23. Actualizar.

Se puede actualizar cualquiera de los datos de la empresa a excepción del Folio. Para actualizar se recomienda primero hacer una búsqueda por alguno de los campos y posteriormente la actualización requerida.

Una vez realizados los cambios se mostrará una pantalla como la siguiente:

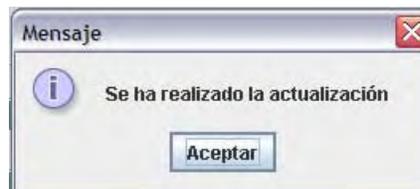


Figura 8.24. Actualización de Empresas.

8.4.6 ELIMINAR EMPRESAS

Para eliminar una Empresa todos sus datos deben estar presentes. Hay que tomar en cuenta que también se eliminarán todos los datos de los centros de cómputo de dicha empresa. Deberá darse click en el botón "Eliminar".



Figura 8.25. Eliminar.

A continuación aparecerá la siguiente ventana preguntando la confirmación de la eliminación:

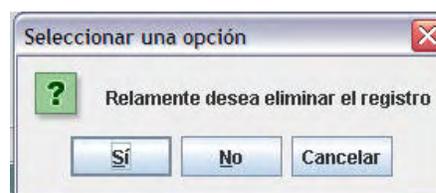


Figura 8.26. Confirmación de eliminación.

Si se acepta, se eliminarán los datos de la empresa.



Figura 8.27. Eliminación de empresa.

8.4.7 CREAR PDF DE EMPRESAS

En el supuesto de que se requiera imprimir la información de una Empresa, los datos de la misma se imprimirán en un archivo PDF, dando click en el botón "PDF".



Figura 8.28. Crear archivo PDF.

A continuación se mostrará una ventana en la cual se preguntará la ruta en la que se desea guardar el archivo:

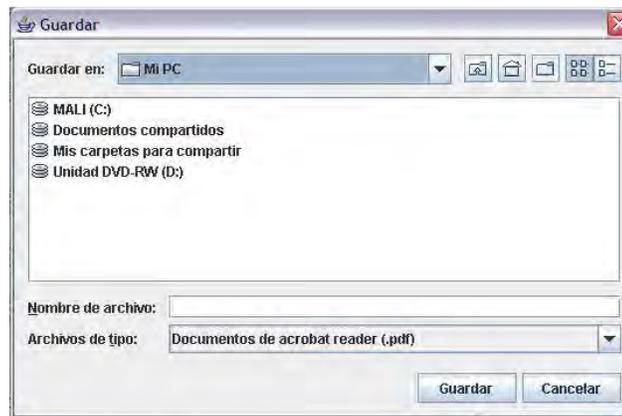


Figura 8.29. Ruta para guardar archivo PDF.

Una vez seleccionada la ruta e ingresado el nombre del archivo, éste se abrirá en el visor de PDF.

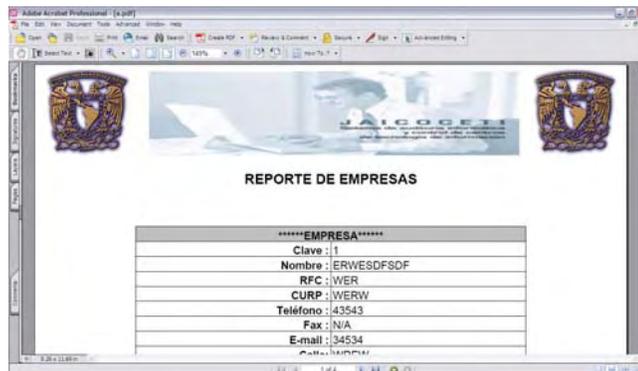


Figura 8.30. PDF de empresas.

Es importante tener en cuenta que se debe tener instalado un visor de archivos PDF para poder visualizar el documento.

8.4.8 ASIGNAR REPRESENTANTE

Ya registrada una empresa, es posible asignar a los representantes de la misma. Para ello se mostrará una pantalla como la siguiente:



Figura 8.31. Representantes de empresa.

Los datos relacionados con los representantes de la empresa se pueden ingresar, consultar o eliminar; para ello basta con teclear la información solicitada y posteriormente seleccionar la acción a realizar

8.4.9 VER INFORMACIÓN DE CENTROS DE TECNOLOGÍA DE INFORMACIÓN (CTI)

Una vez que se ha cargado la información de una empresa es posible visualizar los datos relacionados con los CTI que pertenecen a la misma, basta con dar click en esta opción y se mostrará de manera automática la pestaña de la sección de CTI con los datos de los CTI que pertenecen a la empresa.



Figura 8.32. Ver los CTI de las empresas.

8.4.10 ASIGNAR AUDITORÍAS A LOS AUDITORES (PANTALLA DE AUDITORES)

Para asignar las auditorías que un auditor puede realizar es necesario dar click en el botón de asignación de auditor en:



Figura 8.33. Asignar auditorías.

Y se mostrará la siguiente pantalla:



Figura 8.34. Pantalla de asignación de auditorías.

En la pantalla anterior se muestran las auditorías registradas en el sistema. Para asignar una auditoría hay que marcarla y dar click en Agregar, o bien si lo que se desea eliminar es la auditoría asociada al auditor hay que dar click en Eliminar.

8.4.11 PERSONAS EN CASO DE EMERGENCIA (PANTALLA DE AUDITORES)

Para asignar a los contactos en caso de emergencia con un Auditor/Promotor, se debe dar click en la opción Emergencia:



Figura 8.35. Personas de emergencia.

Y mostrará una ventana como la siguiente:

Figura 8.36. Pantalla para personas de emergencia.

En la pantalla anterior se deben ingresar los datos requeridos y posteriormente dar click en actualizar para que la información se guarde.

8.4.12 MODIFICAR PREGUNTAS (PANTALLA DE AUDITORÍAS)

Una vez realizada una búsqueda hay que dar click en Modificar Preguntas, se mostrará una ventana como la siguiente:



Figura 8.37. Pantalla modificar preguntas.

En la pantalla anterior es posible actualizar, eliminar o agregar preguntas para una auditoría.



Figura 8.38. Selección de preguntas.

Una vez seleccionado el número de pregunta se cargará la pregunta y la sugerencia:

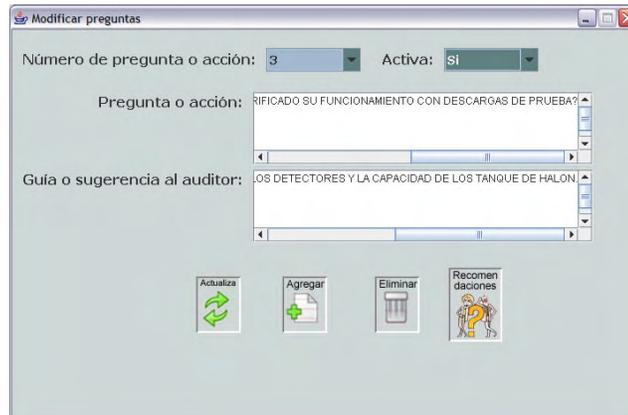


Figura 8.39. Modificación de preguntas.

Realizados los cambios que se desean se debe dar click en el botón Actualiza para guardar los cambios.

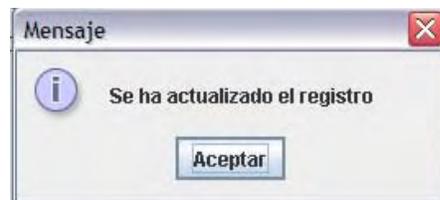


Figura 8.40. Actualización de preguntas.

Si se quiere eliminar una pregunta hay que utilizar el botón correspondiente y aparecerá una ventana preguntando la confirmación de la eliminación:

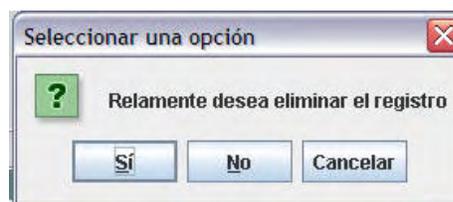


Figura 8.41. Confirmación de eliminación de preguntas.

Si se acepta se eliminará el registro mostrado en la pantalla:



Figura 8.42. Eliminación de preguntas.

Para agregar una nueva pregunta es necesario ingresar el texto de la pregunta así como para sugerencia, el folio de la pregunta se genera de manera automática por el sistema:

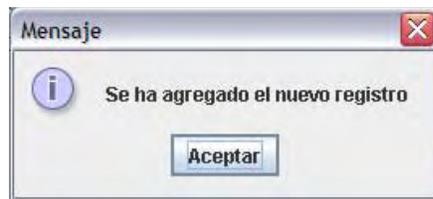


Figura 8.43. Inserción de preguntas.

Para acceder a la sección de recomendaciones, una vez que se ha seleccionado una pregunta se debe dar click en el botón Recomendaciones:



Figura 8.44. Pantalla de recomendaciones.

Y se desplegará una ventana como la siguiente:

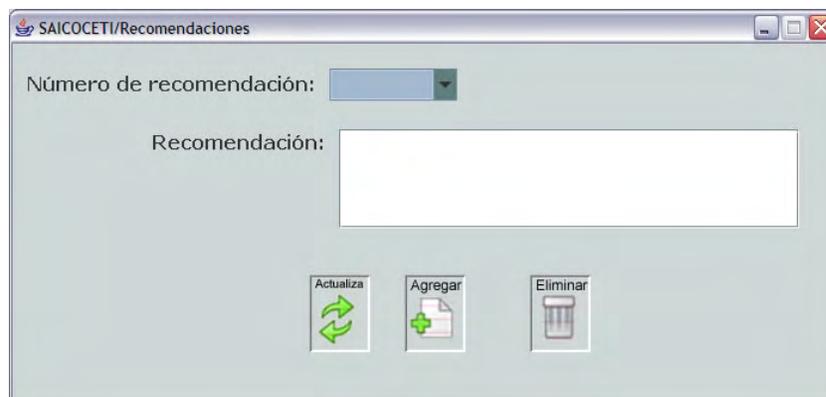


Figura 8.45. Recomendaciones de preguntas.

La pantalla anterior tiene un funcionamiento similar al explicado para el manejo de preguntas, pero aplicado a las recomendaciones.

8.4.13 REPORTE (PANTALLA DE AUDITORÍAS POR REALIZAR)

Para consultar el reporte de las observaciones realizadas por los auditores vía Internet se debe dar click en la opción de Reporte y se mostrará una ventana como la siguiente:

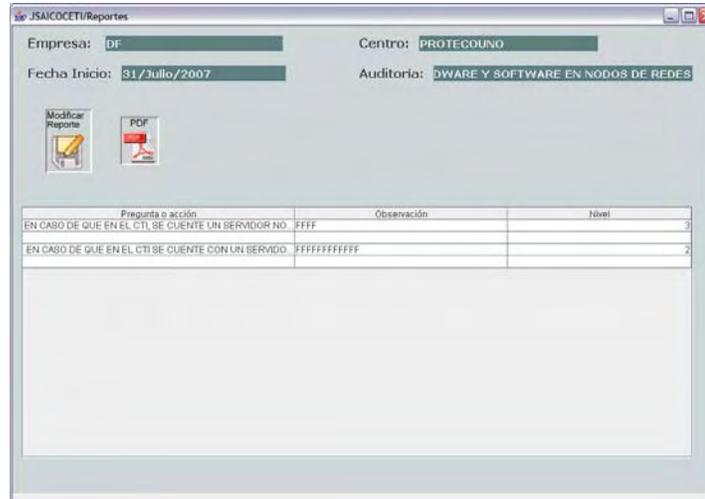


Figura 8.46. Reporte de avances.

Dando click en la opción de Modificar Reporte es posible modificar las recomendaciones realizadas por los auditores:



Figura 8.47. Modificar reporte.

Se mostrará una ventana como la siguiente con el listado de las preguntas que han sido evaluadas por los auditores:

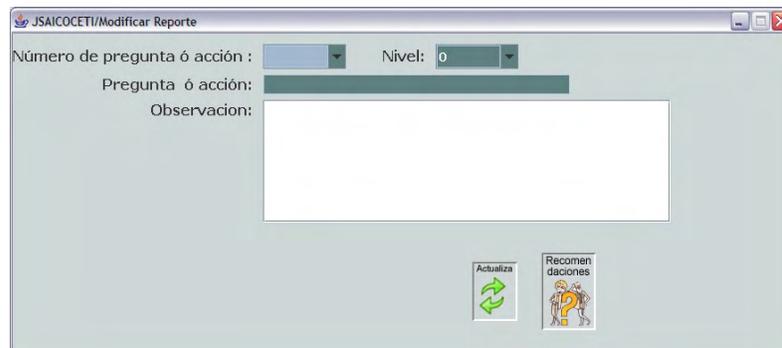


Figura 8.48. Selección de la pregunta a modificar.

Al seleccionar una de ellas la información se cargará de manera automática:



Figura 8.49. Editar reporte.

Para guardar los cambios hay que dar click en el botón Actualiza.

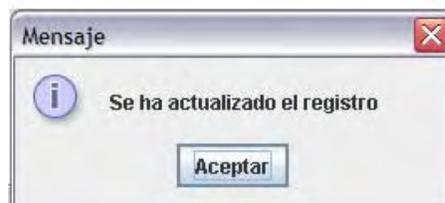


Figura 8.50. Actualización de reporte.

Al dar click en el botón Recomendaciones es posible agregar recomendaciones a las observaciones hechas por los auditores.



Figura 8.51. Asignar recomendaciones.

Se mostrará una ventana como la siguiente:

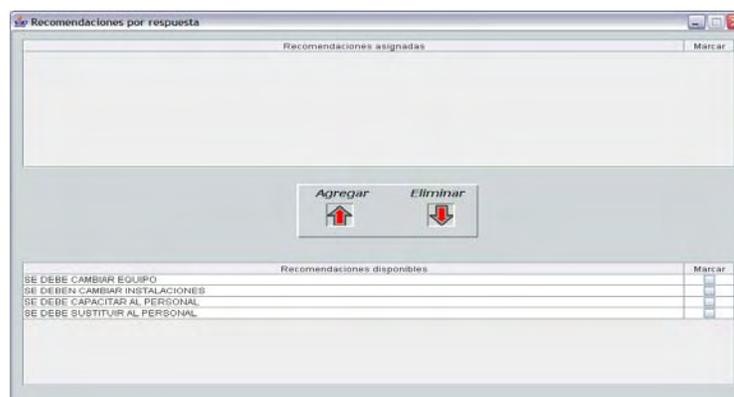


Figura 8.52. Agregar recomendaciones.

Marcando las recomendaciones que desean y dando click en el botón de Agregar se asignan las recomendaciones a la pregunta:

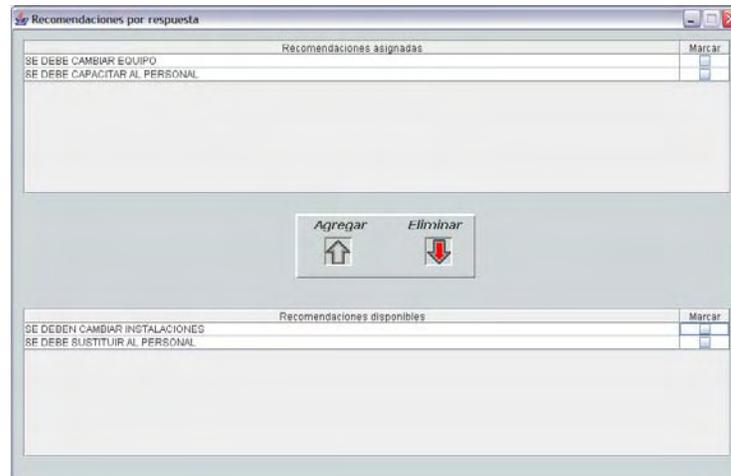


Figura 8.53. Eliminar recomendaciones.

Para eliminar recomendaciones asignadas se deben marcar y posteriormente dar click en el botón de eliminar.

Dando click en a opción de PDF se mostrará una ventana como la siguiente:



Figura 8.54. Crear PDF del reporte.

En la cual hay que indicar la ruta para guardar el reporte y una vez proporcionada la ruta se mostrará el reporte tal y como se muestra en la siguiente página.

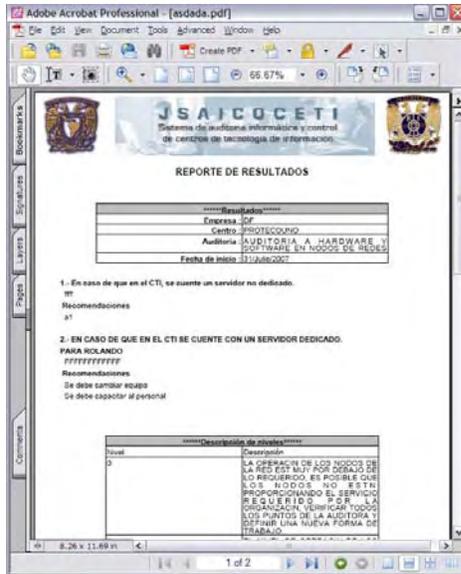


Figura 8.55 PDF del reporte.

Los administradores del sistema tendrán la opción de crear el reporte en formato RTF:

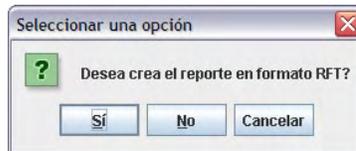


Figura 8.56 Opción de reporte en RTF.

En caso de dar click en el botón de Sí, se creará un reporte como el siguiente:

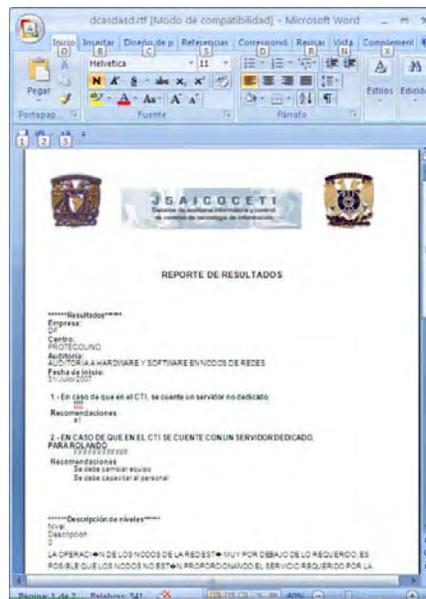


Figura 8.57 Reporte en RTF.

8.4.14 ASIGNAR LÍDER (PANTALLA DE AUDITORÍAS POR REALIZAR)

Al dar click al botón de Asignar Líder se mostrará una ventana como la siguiente:

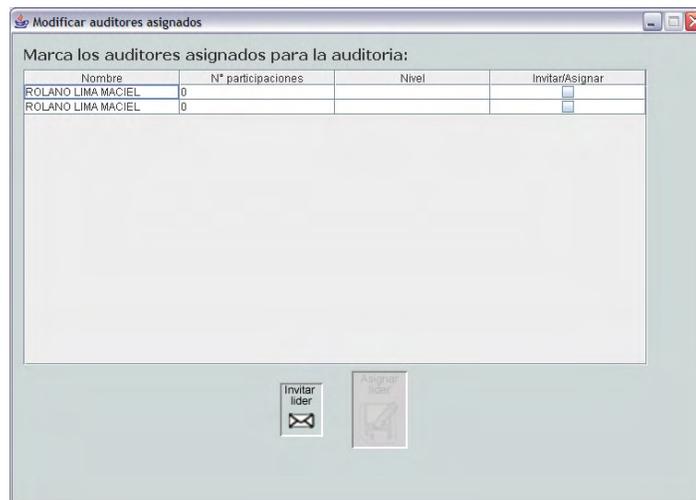


Figura 8.58 Invitación para líder.

En la pantalla anterior se muestra la lista de los auditores que pueden llevar a cabo la auditoría en cuestión así como el número de participaciones y el nivel con el que han participado.

Para invitar a un grupo de auditores a ser líderes basta con marcarlos y posteriormente dar click en la opción de Invitar líder.

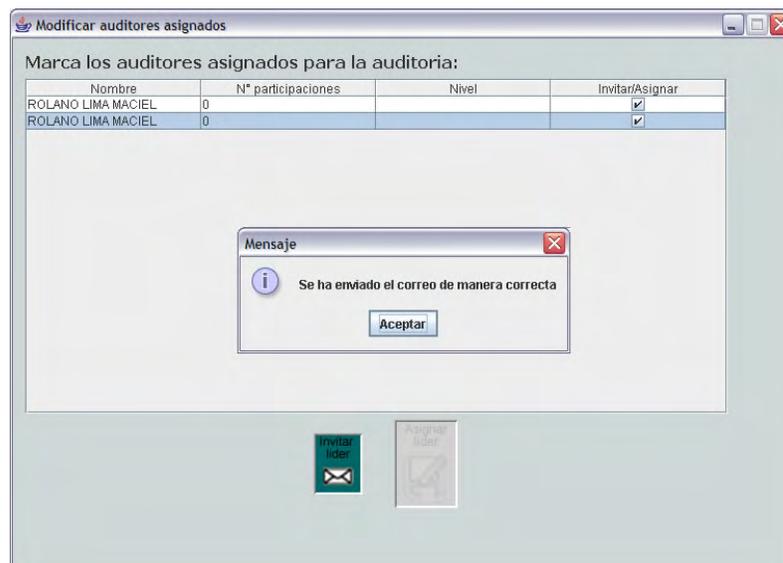


Figura 8.59 Envío de correo para invitación.

Una vez que se ha enviado la invitación se habilitará la opción de Asignar líder, es necesario seleccionar solamente uno de los auditores mostrados en la lista para enviarle el correo

electrónico que fue asignado como líder y a los restantes una notificación de que no fueron asignados.

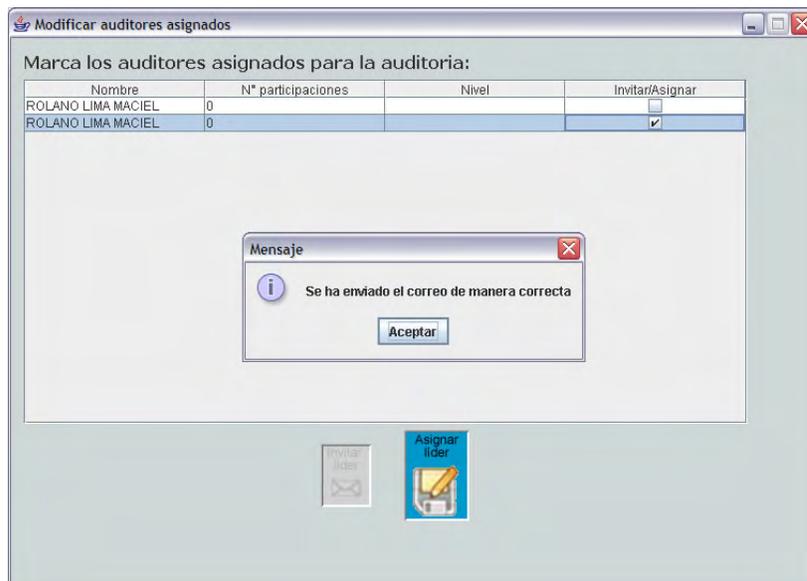


Figura 8.60 Selección de líder.

8.4.15 CONSULTAR BITÁCORA (PANTALLA DE AUDITORÍAS POR REALIZAR)

Se mostrará una ventana como la siguiente:

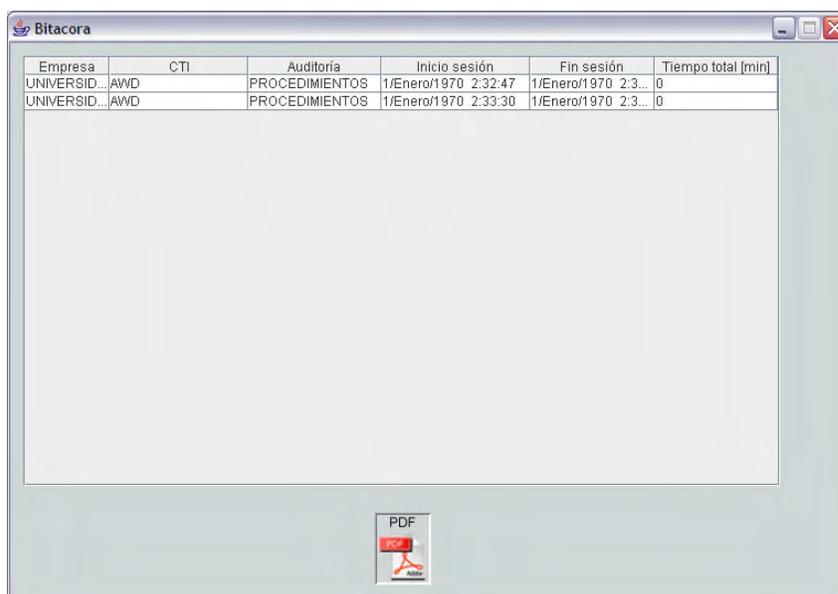


Figura 8.61 Consulta de bitácora.

Donde se muestran los datos asociados a los movimientos realizados por los auditores los cuales se pueden exportar a PDF.

8.4.16 CAMBIAR CONTRASEÑA

En caso de que requiera cambiar la contraseña se debe acceder al menú Cuenta y luego a la opción: Cambiar tu contraseña:



Figura 8.62 Acceso a cambio de contraseña.

O bien con presionando las teclas ALT+C un par de veces, realizado lo anterior se mostrará una pantalla como la siguiente:

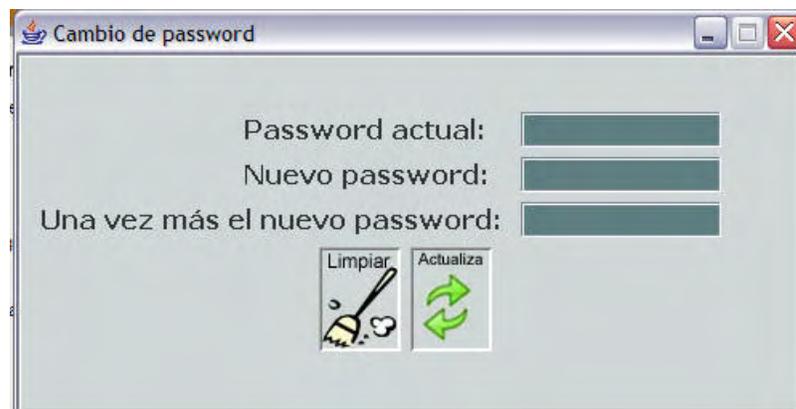


Figura 8.63 Pantalla de cambio de contraseña.

Una vez que se han ingresado los datos hay que dar click en el botón de actualizar para que los cambios se lleven a cabo.

8.4.17 ADMINISTRAR USUARIOS

Para acceder a esta opción basta con seleccionar el menú Usuarios y después en la opción Administrar Usuarios tal y como se muestra en la siguiente imagen:



Figura 8.64 Acceso a la administración de usuarios.

O bien presionando las teclas ALT+U y posteriormente con ALT + A y una vez realizado lo anterior se mostrará una pantalla de la siguiente página.



Figura 8.65 Pantalla de administración de usuarios.

En la pantalla anterior es posible: Agregar, Actualizar o Eliminar un usuario, para ello es suficiente con ingresar los datos solicitados y llevar a cabo la operación que se desee, solamente los administradores del sistema tendrán acceso a la función anterior.

El sistema cuenta con tres tipos de usuarios los cuales tienen los siguientes privilegios de acuerdo a su rol:

ROL/Actividad	Capturistas				
	Agregar	Buscar	Actualizar	Eliminar	Crear Pdf
Sección					
Empresas	X	X	X	X	X
CTI	X	X	X	X	X
Históricos	----	X	----	----	X
Audidores/Promotores	----	X	----	----	X
Auditorías	----	X	----	----	X
Auditorías por realizar	X	X	X	----	X
Reporte de Avances	----	x	----	----	x
Consultar bitácora	----	x	----	----	x
Modificar Auditores asignados	----	----	----	----	----
Personas de emergencia	----	x	X	----	----
Asignar líder	----	----	----	----	----

ROL/Actividad	Auditores				
	Agregar	Buscar	Actualizar	Eliminar	Crear Pdf
Sección					
Empresas	----	X	----	----	X
CTI	----	X	----	----	X
Históricos	----	X	----	----	X
Auditores/Promotores	----	X	----	----	X
Auditorías	----	X	----	----	X
Auditorías por realizar	----		----	----	
Reporte de Avances	----	X	----	----	X
Consultar bitácora	----	X	----	----	X
Modificar Auditores asignados	----	----	----	----	----
Personas de emergencia	----	X	----	----	----
Asignar líder	----	----	----	----	----

ROL/Actividad	Administradores				
	Agregar	Buscar	Actualizar	Eliminar	Crear Pdf
Sección					
Empresas	X	X	X	X	X
CTI	X	X	X	X	X
Históricos	----	X	----	----	X
Auditores/Promotores	X	X	X	X	X
Auditorías	X	X	X	X	X
Auditorías por realizar	X	X	X	X	X
Reporte de Avances	----	X	X	---	X
Consultar bitácora	----	X	---	---	X
Modificar Auditores asignados	X	----	X	X	---
Personas de emergencia	----	X	X	---	---
Asignar líder	----	----	X	---	---

Con base en los datos anteriores se debe elegir el perfil del usuario que se agrega, de acuerdo a las labores que va a desempeñar en el SAICOCETI.

8.5 MANUAL DEL SAICOCETI WEB

8.5.1 ACCESO AL SAICOCETI WEB

La forma de acceder al sistema web es por medio de Internet, a través de la dirección proporcionada por los administradores del sistema (Figura 8.66).



Figura 8.66 Pantalla informativa del SAICOCETI WEB

Una vez en la página inicial se tendrán diferentes opciones:

1. Descripción de las auditorías disponibles en el sistema.
2. Descripción de las auditorías disponibles en el sistema.
3. Entrada al SAICOCETI WEB.
4. Correo de dudas o sugerencias al sistema.
5. Ligas de interés.

8.5.2 DESCRIPCIÓN DE LAS AUDITORÍAS DISPONIBLES EN EL SISTEMA

A través de esta pantalla se podrán consultar las auditorías que se pueden realizar, presentando el nombre de la auditoría y la descripción de la misma en formato PDF (Figura 8.67).

NOMBRE DE LA AUDITORÍA		DETALLE DE LA AUDITORÍA
1 →	AUDITORÍA A EQUIPOS DE INFRAESTRUCTURA	Ver
	AUDITORIA A HARDWARE Y SOFTWARE DE SERVIDORES	Ver
	AUDITORIA A HARDWARE Y SOFTWARE DE TELECOMUNICACIONES	Ver
	AUDITORIA A HARDWARE Y SOFTWARE EN NODOS DE REDES	Ver
	AUDITOR?A A CONTRATO DE MANTENIMIENTO A EQUIPOS DE C?MPUTO, TELECOMUNICACIONES O INFRAESTRUCTURA	Ver
	AUDITOR?A A CONTRATO DE PRESTACI?N DE SERVICIOS PROFESIONALES (DESARROLLO DE APLICACIONES Y/O MANTENIMIENTO)	Ver
	AUDITOR?A A CONTRATOS DE COMPRA-VENTA DE EQUIPOS DE C?MPUTO, TELECOMUNICACIONES O DE INFRAESTRUCTURA	Ver
	PROCEDIMIENTOS	Ver
SISTEMAS DE APLICACIONES	Ver	

Figura 8.67 Pantalla de descripción de las auditorías disponibles en el sistema.

1. Nombre de la auditoría.
2. Liga para ver la descripción completa de la auditoría seleccionada.

Al dar click sobre una de las ligas relacionadas a una auditoría se abrirá una ventana emergente la cual presentara la descripción de la auditoría en formato PDF (Figura 8.68).

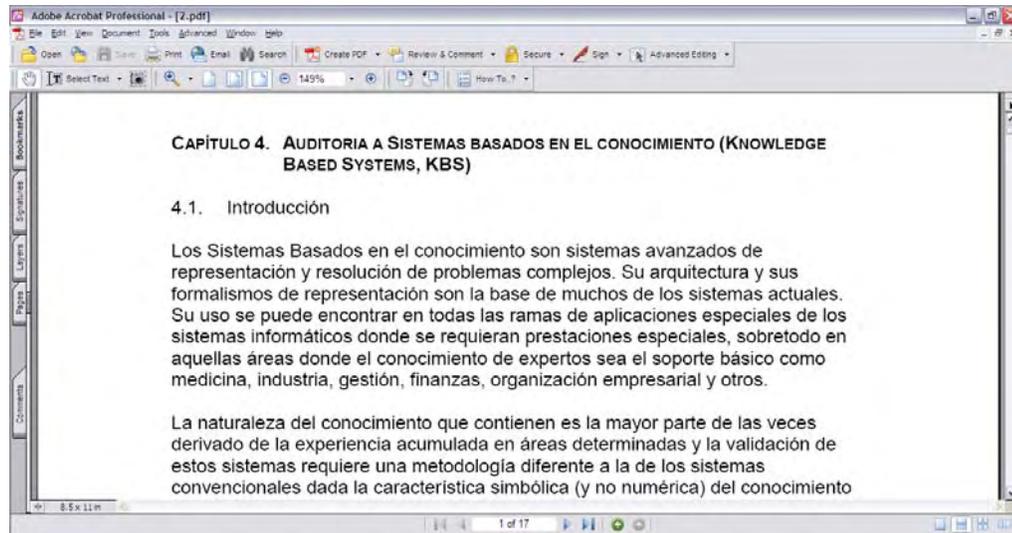


Figura 8.68 Pantalla de Descripción de las auditorías en formato PDF.

8.5.3 ENTRADA AL SAICOCETI WEB

Dentro de esta pantalla es donde los auditores y las empresas podrán interactuar con el sistema (Figura 8.69).



Figura 8.69 Pantalla inicial del SACOCETI WEB.

1. Descripción de auditorías.
2. Consultar avances de la auditoría.
3. Comenzar o reanudar auditorías.

8.5.3.1 DESCRIPCIÓN DE LAS AUDITORÍAS DISPONIBLES EN EL SISTEMA

Esta sección que se muestra es la misma referencia que se describe en el punto 8.5.2.

8.5.3.2 CONSULTAR AVANCES DE LAS AUDITORÍAS

En la pantalla inicial del SAICOCETI WEB (Figura 8.69) se muestran dos imágenes una correspondiente a “Empresas” y otra para “Auditores”, para poder consultar los avances de las auditorías se tiene que dar click sobre la imagen de Empresas se abrirá una ventana emergente (1) en la cual se podrá teclear el login y password asignados en el SAICOCETI Desktop (Figura 8.70) para acceder a consultar los avances de la auditoría (Figura 8.71).



Figura 8.70 Pantalla para poder teclear login y password.



Figura 8.71 Pantalla de consulta de avances de las auditorías.

En la figura 8.71 se muestran las siguientes referencias:

1. Nombre de la empresa.
2. Nombre del centro.
3. Observaciones generadas por el sistema.
4. Observaciones generadas por los auditores.

8.5.3.3 INICIAR O REANUDAR AUDITORÍAS

Para poder iniciar o reanudar auditorías se tiene que dar click sobre la imagen de Auditorías, se abrirá una ventana emergente en la cual se podrá teclear el login y password asignados en el SAICOCETI Desktop (Figura 8.72) para acceder a iniciar o reanudar auditorías (Figura 8.73).



Figura 8.72 Pantalla para poder teclear login y password.

Esta pantalla (Figura 8.73) esta organizada por varias pestañas y al dar click sobre alguna de ellas se cambiara dinámicamente a la sección seleccionada.



Figura 8.73 Pantalla para iniciar o reanudar auditorías.

Los puntos mostrados en la Figura 8.73 son:

1. Sugerir preguntas adicionales a las auditorías existentes.
2. Sugerir auditorías adicionales a las auditorías existentes.
3. Continuar auditoría o iniciarla.
4. Elegir grupo de trabajo.
5. Salir

8.5.3.3.1 SUGERIR PREGUNTAS ADICIONALES A LAS AUDITORÍAS EXISTENTES

En esta pantalla es posible sugerir preguntas adicionales a las existentes, se elige una “auditoría” del combo, se da click sobre la barra de número de “preguntas a sugerir” para que se habilite la opción de seleccionar el número de preguntas a sugerir por medio del combo de “número de preguntas” y dar click en el botón de “enviar”, esto abrirá una ventana emergente donde se podrán capturar las preguntas adicionales(Figura 8.74a, 8.74b) con base en los parámetros de la pantalla anterior (Figura 8.75).

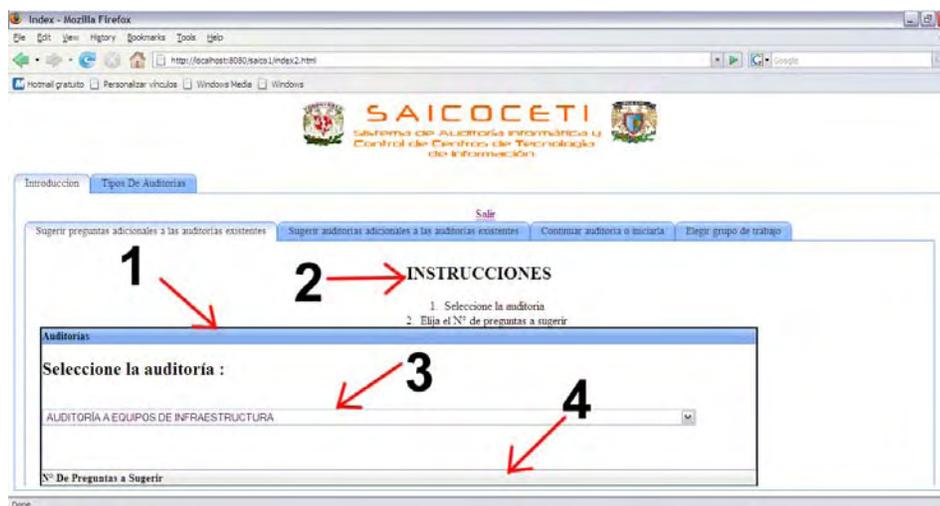


Figura 8.74a Pantalla para sugerir preguntas adicionales a las auditorías existentes.

1. Instrucciones de la pantalla.
2. Barra de auditorías.
3. Combo de auditorías.
4. Barra de número de preguntas a sugerir.

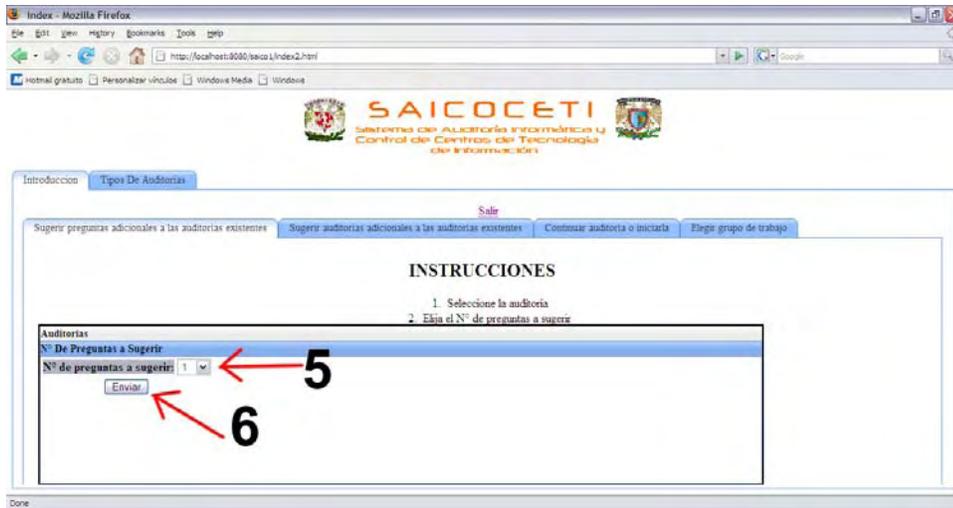


Figura 8.74b Pantalla para sugerir preguntas adicionales a las auditorías existentes.

5. Combo de número de preguntas a sugerir.
6. Botón enviar.

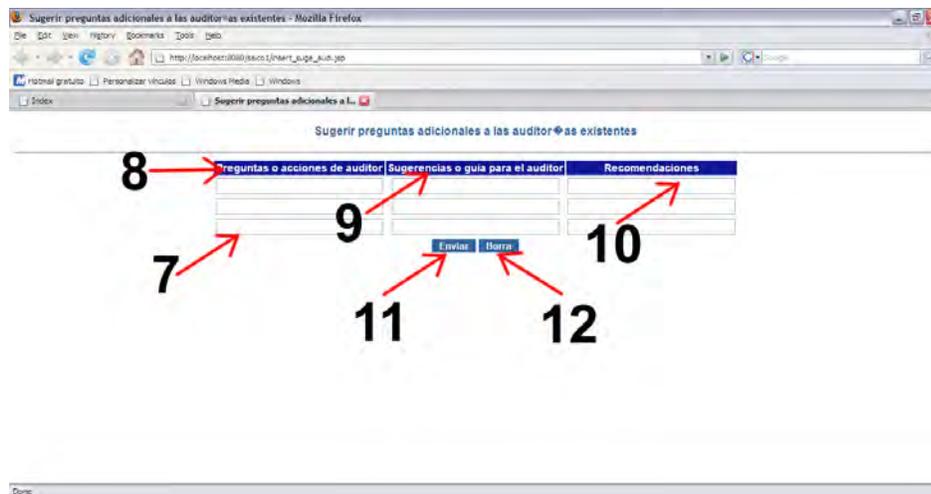


Figura 8.75 Pantalla de captura para sugerir preguntas adicionales a las auditorías existentes.

7. Pantalla de captura de preguntas o acciones, sugerencias o guías y recomendaciones.
8. Preguntas o acciones de auditor.
9. Sugerencias o guía para auditor.
10. Recomendaciones.
11. Botón enviar, este botón envía los campos capturados al sistema para su análisis.
12. Botón borrar, limpia los campos capturados en caso de error.

8.5.3.3.2 SUGERIR AUDITORÍAS ADICIONALES A LAS AUDITORÍAS EXISTENTES

Dentro de esta pantalla es posible sugerir auditorías adicionales a las auditorías existentes, se elige el número de preguntas por medio del combo para la nueva auditoría que se esta proponiendo con su descripción (Figura 8.76).

Una vez seleccionado el número de preguntas y dando click en el botón de enviar, lo que abrirá una ventana emergente en la cual se podrá capturar la información mencionada anteriormente (Figura 8.77).



Figura 8.76 Pantalla para sugerir auditorías adicionales a las auditorías existentes.

1. Combo de número de preguntas a sugerir.
2. Botón enviar.

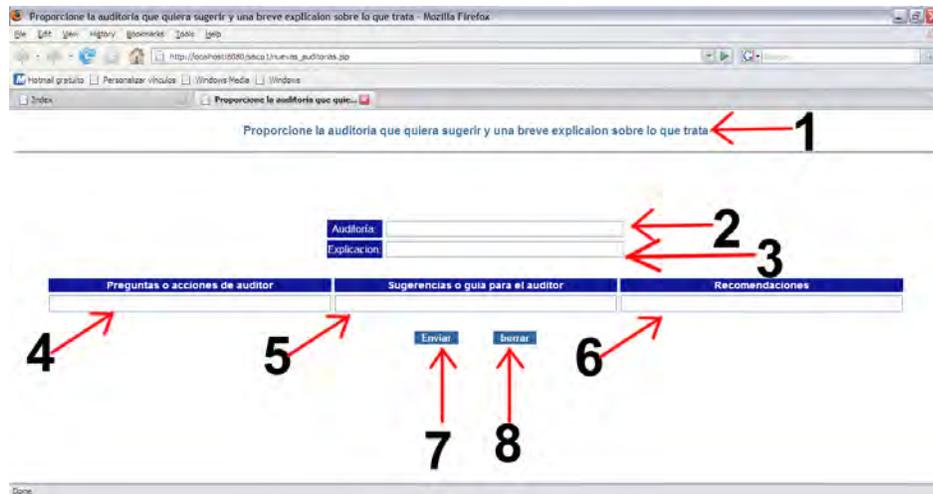


Figura 8.77 Pantalla de captura para sugerir auditorías adicionales a las auditorías existentes.

Las referencias mostradas en la Figura 8.77 son:

1. Indicaciones de la pantalla.
2. Nombre de la auditoría.
3. Explicación de la auditoría.
4. Preguntas o acciones de auditor.
5. Sugerencias o guía para el auditor.
6. Recomendaciones.

8.5.3.3 CONTINUAR AUDITORÍA O INICIARLA

Dentro de esta pantalla se podrá iniciar una auditoría asignada a un auditor o continuarla si fuere el caso.

Se presentara una tabla donde se le mostraran al auditor todas las auditorías en las que esté asignado, ya sean auditorías nuevas o auditorías que ya está trabajando el auditor (Figura 8.78).



Figura 8.78 Pantalla de selección de auditorías a trabajar.

Por medio del botón de radio se podrá seleccionar la auditoría de que desee trabajar, al dar click sobre este botón se abrirá una ventana emergente en la cual se mostrara una nueva tabla con información referente a la auditoría que se haya seleccionado en la pantalla anterior, y es aquí donde el auditor podrá calificar y emitir sus comentarios de la preguntas que se le presenten (Figura 8.79).



Figura 8.79 Pantalla de trabajo de auditorías.

1. Numero de pregunta.
2. Pregunta o acción.
3. Sugerencia o guía.
4. Comentarios.
5. Nivel.
6. Hora de inicio.
7. Hora de fin.
8. Liga para descargar las preguntas presentadas en pantalla.
9. Botón para enviar y guardar la evaluación a las preguntas.

8.5.3.3.4 ELEGIR GRUPO DE TRABAJO

Esta pantalla sólo se le presentará al auditor que sea líder en una auditoría determinada, al resto de los auditores no les aparecerá esta pestaña.

El auditor podrá elegir de una lista de auditores los candidatos para formar parte de su grupo de trabajo en la auditoría, por medio de un botón tipo check se podrá marcar auditores para poder enviarles un mail de invitación para participar en la auditoría (Figura 8.80).



Figura 8.80 Pantalla de elección del grupo de trabajo.

1. Nombre del auditor.
2. Nivel del auditor.
3. Botón de mail de invitación.
4. Combo de nivel propuesto a participar.
5. Casilla de selección.

8.6 MANUAL DE INSTALACIÓN DE LA BASE DE DATOS DEL SAICOCETI

8.6.1 REQUERIMIENTOS MÍNIMOS DE LA INSTALACIÓN

Requerimientos de hardware (Servidor):

Procesador Pentium III o superior.

50 MB de espacio libre en disco duro más el necesario para el almacenamiento de la información.

256 MB de memoria RAM.

Tener una IP real.

Requerimientos de software (Servidor):

- Sistema operativo Linux o Unix en cualquier distribución.
- PostgreSQL 8.1.5 o superior.
- pgAdmin III.

8.6.2 INSTALACIÓN DE LA BASE DE DATOS

El CD de instalación tiene un archivo llamado SAICOCETI.backup, este archivo es necesario para poder crear la estructura de la base de datos.

El primer paso es conectarse al servidor de la base de datos dando doble click sobre la imagen de la base para poder entrar (Figura 8.81).

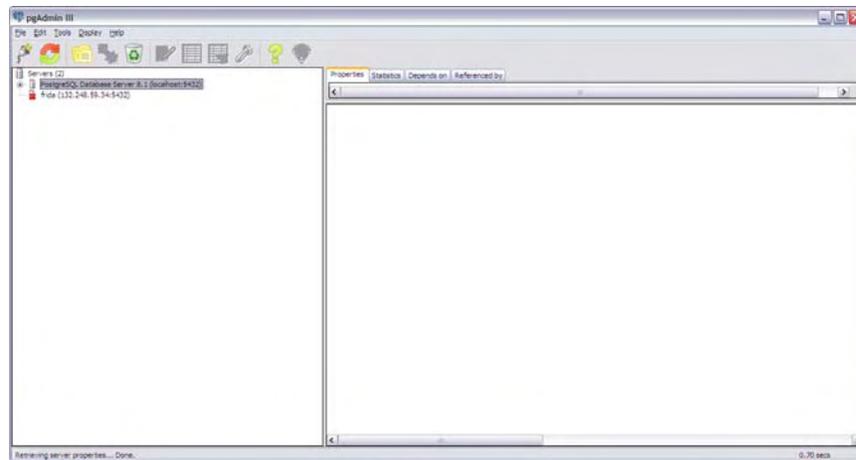


Figura Servidores 8.81

Posteriormente se desplegará un menú con varias opciones y se debe seleccionar “Databases”, se da click derecho sobre la imagen de “Databases” y se elige la opción “New Databases” (Figura 8.82).

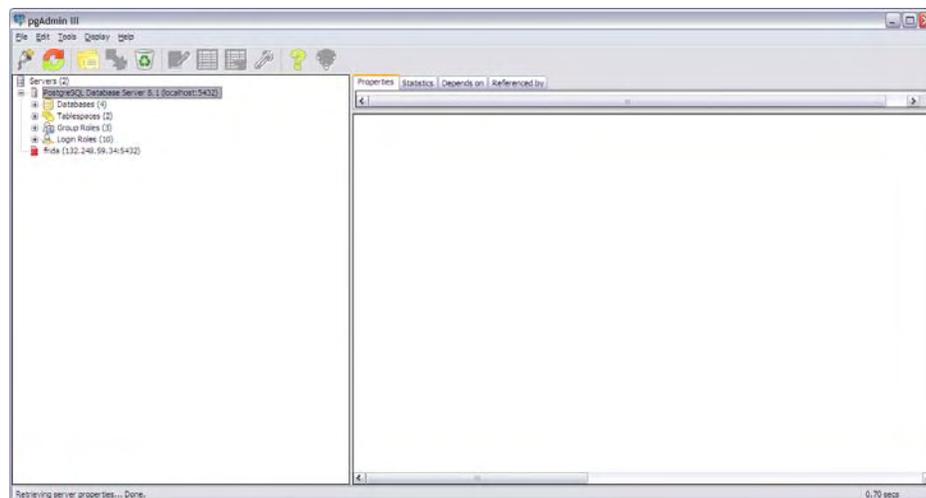


Figura Base de datos 8.82

Al hacer esto se abrirá una ventana en la cual se podrán capturar los campos necesarios para la base de datos para el SAICOCETI (Figura 8.83), los campos necesarios de captura serán los siguientes:

1. Name: SAICOCETI
2. Owner: root
3. Encoding: SLQ_ASCII
4. Template: postgres
5. Tablespace: pg_default

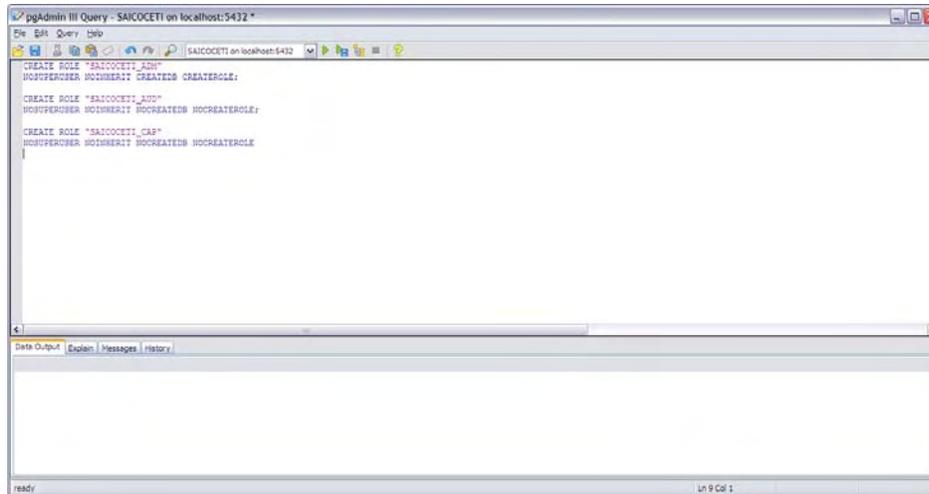


Figura Creación de roles 8.85

Una vez tecleados los comandos se debe seleccionar el menú e ir a la opción “Query” y en el submenú seleccionar “Execute”.

Si todo ha ido bien hasta este paso, ahora es el momento de usar el archivo que creará la estructura de la base de datos por medio del archivo SAICOCETI.backup, el cual se ejecutará dando click derecho sobre la base de datos que se acaba de crear SAICOCETI y elegir la opción “Restore” (Figura8.87).

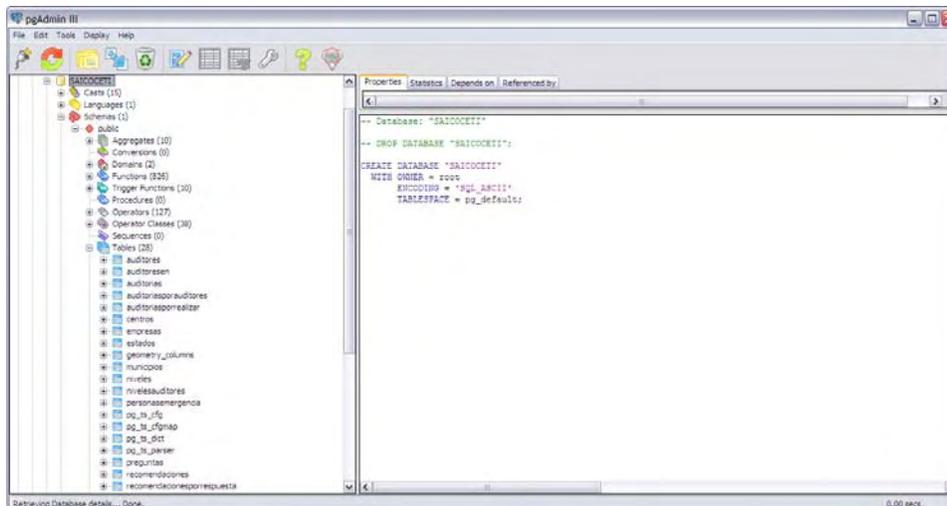


Figura Restauración de la base de datos 8.87

Esto mostrará una de ventana en la cual se tendrá la opción de seleccionar el origen del archivo SAICOCETI.backup y por último se elijé la opción “Ok” como se muestra a continuación (Figura 8.88).

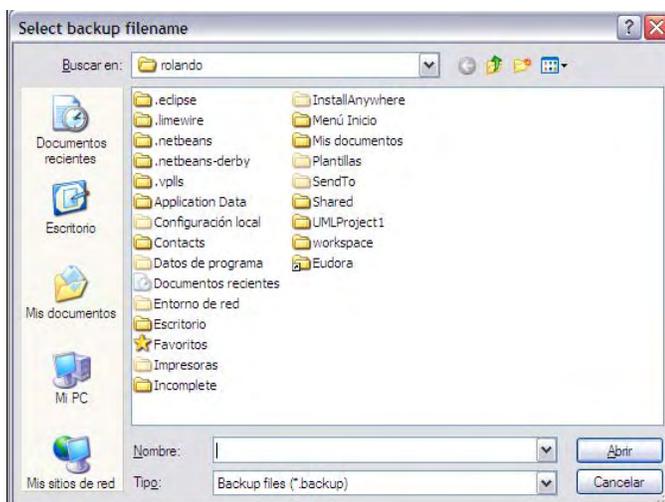


Figura 8.88

El último paso es seleccionar “Tools” de la barra de opciones y del menú desplegado elegir “Query tool” (Figura 8.89), donde se tendrá que teclear los siguientes comandos:

```
CREATE ROLE s_administrador LOGIN PASSWORD
9450476b384b32d8ad8b758e76c98a69' CREATEROLE
VALID UNTIL 'infinity' IN ROLE "SAICOCETL_ADM";
INSERT INTO USUARIOS VALUES('administrador','9450476b384b32d8ad8b758e76c98a69',3);
```

CAPÍTULO 9: CONCLUSIONES

El presente proyecto dio como resultado la migración del SAICOCETI a software libre, de manera que para la operación del sistema no se emplearán recursos económicos adicionales por concepto de software requerido por el sistema.

Los beneficios más relevantes del sistema son los siguientes:

- No se tiene que realizar pago alguno por el uso de herramientas de software para desarrollo y administración:
 - Para llevar a cabo la administración de la base de datos y del servidor Web se cuentan con herramientas gratuitas de excelente calidad proporcionadas por PostgreSQL y Apache Tomcat.
 - Para el mantenimiento se cuentan con entornos de desarrollo sofisticados disponibles sin costo por Sun Microsystems.
- Independencia del sistema operativo ya que al estar construido en Java puede funcionar siempre y cuando se cuente con la máquina virtual de Java.
- Cuenta con envío automático de notificaciones vía correo electrónico para los auditores.
- Se cuenta con elaboración de reportes en formato PDF así como un preinforme ejecutivo en formato RTF para disminuir el tiempo en la elaboración del mismo.
- Se incrementa la seguridad del sistema a través del empleo de algoritmos de encriptación.
- Se permite al líder de proyecto elegir su equipo de trabajo.
- La Facultad de Ingeniería de la UNAM es dueña del sistema.

Como todo sistema, el SAICOCETI puede ser mejorado mediante el establecimiento de un programa de mantenimiento y manejo de cambios al mismo, y posteriormente extender su alcance y funcionalidad. Durante la elaboración de este proyecto se detectó que la mejor área de oportunidad para explotar y mejorar el sistema es elaboración de auditorías a empresas públicas y privadas permitiendo aumentar los ingresos económicos de la Facultad de ingeniería así como de la propia UNAM.

APÉNDICE 1 CLUF

APÉNDICE 1 CLUF

“CONTRATO DE LICENCIA PARA EL USUARIO FINAL IMPORTANTE. LEA DETENIDAMENTE: este Contrato de Licencia para el Usuario Final (CLUF) constituye un acuerdo legal entre usted (sea persona física o jurídica) y el fabricante (fabricante) del sistema informático de computación o componente de sistema informático o de computación (hardware) con el que usted adquirió el producto o productos de software de Microsoft antes identificados (software).

El SOFTWARE incluye software de Microsoft y puede incluir medios relacionados, materiales impresos, documentación "en pantalla" o electrónica y servicios basados en Internet.

Tenga en cuenta, sin embargo, que todo el software, documentación o servicios web incluidos en el SOFTWARE, o que sean accesibles a través del SOFTWARE, y que estén acompañados de sus propios contratos de licencia o condiciones de uso, se rigen por esos contratos o condiciones de uso en lugar de por este CLUF. Los términos de una copia impresa del CLUF, que puede acompañar al SOFTWARE, sustituyen y anulan los términos de todo CLUF "en pantalla".

El CLUF es válido y concede los derechos de licencia para usuario final SÓLO si el SOFTWARE es genuino y se incluye un Certificado de Autenticidad (Certificate of Authenticity) original como parte del SOFTWARE”.

“Al instalar, copiar, descargar, tener acceso o de otra manera utilizar el SOFTWARE, usted acepta quedar obligado por los términos del CLUF.

Si no está de acuerdo con estos términos, no podrá utilizar ni copiar el SOFTWARE y deberá ponerse en contacto cuanto antes con el Fabricante, para obtener instrucciones acerca de cómo devolver el producto o los productos no utilizados de acuerdo con los requisitos de devolución del fabricante.”

“1. OTORGAMIENTO DE LICENCIA.

El Fabricante le concede los siguientes derechos, con la condición de que cumpla todos los términos y condiciones de este CLUF:

* Instalación y uso. Excepto en lo dispuesto expresamente de otra forma en el presente CLUF, usted puede instalar, utilizar, tener acceso, exhibir y ejecutar sólo una (1) copia del SOFTWARE en el EQUIPO. El SOFTWARE no puede ser utilizado por más de dos (2) procesadores a la vez en el EQUIPO, a menos que se indique un número mayor en el Certificado de Autenticidad.

Puede permitir a un máximo de diez (10) ("máximo de conexiones") equipos u otros dispositivos electrónicos (cada uno de ellos un "Dispositivo") conectarse al EQUIPO para utilizar los servicios del SOFTWARE sólo para servicios de archivo e impresión, servicios de

Internet Information Server y acceso remoto (incluyendo el uso compartido de conexiones y los servicios de telefonía).

El máximo de diez (10) conexiones incluye cualquier conexión indirecta realizada mediante "multiplexado" u otro software o hardware que agrupe o agregue conexiones.

Excepto que se permita lo contrario a continuación, usted no podrá utilizar el dispositivo para usar, tener acceso, exhibir o ejecutar el SOFTWARE, la interfaz de usuario del SOFTWARE u otro software ejecutable que resida en el EQUIPO."

*Software como componente del equipo: transferencia. NO PODRÁ COMPARTIR, TRANSFERIR NI USAR SIMULTÁNEAMENTE ESTA LICENCIA EN DIFERENTES DISPOSITIVOS.

El SOFTWARE se concede bajo licencia con el HARDWARE como un único producto integrado y sólo puede utilizarse con el HARDWARE. Si el SOFTWARE no va acompañado de nuevo HARDWARE, no podrá utilizar el SOFTWARE.

Todos los derechos que se conceden en el presente CLUF podrán transferirse de manera definitiva, sólo como parte de una transferencia o venta permanente del HARDWARE, con la condición de que usted no conserve ninguna copia en caso de que transfiera el SOFTWARE completo (incluyendo todas las partes componentes, los medios y los materiales impresos, cualquier actualización, este CLUF y el Certificado de Autenticidad (Certificate of Authenticity)) y de que el destinatario acepte los términos del CLUF. Si el SOFTWARE es una actualización, toda cesión deberá incluir también todas las versiones anteriores del SOFTWARE. "

Ya con este pequeño extracto del CLUF se puede ver que esta licencia esta encaminada a proteger a la empresa distribuidora de Software y dejando solamente la posibilidad de solamente usar el software y ya, con lo cual no hay posibilidad de hacer algún reclamo en algunos casos.

APÉNDICE 2 LICENCIA DE JAVA

APÉNDICE 2

Contrato de Licencia de Código Binario de Sun Microsystems, Inc.
JAVA SE RUNTIME ENVIRONMENT (JRE) VERSION 6

“SUN MICROSYSTEMS, INC. (EN ADELANTE DENOMINADO “SUN”) LE CONCEDE LA LICENCIA DEL SOFTWARE DEFINIDO A CONTINUACIÓN ÚNICAMENTE CON LA CONDICIÓN DE QUE USTED ACEPTE TODOS LOS TÉRMINOS ESTIPULADOS EN EL PRESENTE CONTRATO DE LICENCIA DE CÓDIGO BINARIO Y TÉRMINOS DE LICENCIA ADICIONALES (EN CONJUNTO DENOMINADOS “CONTRATO”). POR FAVOR, LEA EL CONTRATO DETENIDAMENTE. AL DESCARGAR O INSTALAR ÉSTE SOFTWARE, USTED ACEPTA LOS TÉRMINOS DEL PRESENTE CONTRATO. INDIQUE SU ACEPTACIÓN SELECCIONANDO EL BOTÓN “ACCEPT” (ACEPTAR) SITUADO AL PIE DEL CONTRATO. SI USTED NO ESTÁ DISPUESTO A COMPROMETERSE CON TODOS LOS TÉRMINOS DEL PRESENTE CONTRATO, SELECCIONE EL BOTÓN “DECLINE” (REHUSAR) SITUADO AL PIE DE ÉSTE CONTRATO A FIN DE DETENER EL PROCESO DE DESCARGA O INSTALACIÓN.

1. DEFINICIONES. “Software” significa el identificado arriba en forma binaria, cualquier otro material en formato legible por equipos informáticos (incluso, pero sin limitarse a ello, bibliotecas, archivos fuente, archivos de cabecera y archivos de datos), cualquier actualización o corrección de errores suministrada por Sun, y cualquier manual de usuario, guía de programación y otra documentación que le haya sido proporcionada por Sun bajo el presente Contrato. “Programas” significa los applets y aplicaciones de Java concebidos para ejecutarse en la plataforma Java Platform, Standard Edition (Java SE) en computadoras y servidores de escritorio compatibles con Java de uso general.
2. LICENCIA DE USO. En virtud de los términos y condiciones dispuestos en el presente Contrato, incluidas, entre otras, las Restricciones de la Tecnología Java de los Términos de Licencia Adicionales, Sun le concede, sin tarifa de licencia, una licencia limitada, no exclusiva e intransferible para la reproducción y el uso interno del Software completo y sin modificaciones con el único propósito de ejecutar Programas. Las licencias adicionales para desarrolladores y/o editores se otorgan en los Términos de Licencia Adicionales.
3. RESTRICCIONES. El software es confidencial y se encuentra protegido por derechos de autor (Copyright). Sun y/o sus licenciantes mantienen la titularidad del Software, así como todos los derechos de propiedad intelectual asociados. Queda prohibido modificar, descompilar o utilizar técnicas de ingeniería inversa en el Software, a

menos que se estipule lo contrario en la legislación aplicable. El licenciataria acepta que el Software con licencia no se ha diseñado para ser utilizado en el diseño, construcción, funcionamiento o mantenimiento de cualquier instalación nuclear, ni se tiene la intención de usarlo para dichos fines. Sun Microsystems, Inc. renuncia a cualquier garantía explícita o implícita de adecuación del Software para dichos fines. El presente Contrato no otorga ningún tipo de derecho, título o propiedad sobre o respecto a las marcas comerciales o de servicio, logotipos o nombres comerciales de Sun o de sus licenciantes. Las restricciones adicionales para las licencias de los desarrolladores y/o editores se estipulan en los Términos de Licencia Adicionales.

4. **GARANTÍA LIMITADA.** Sun garantiza que los medios en los que se proporciona el Software (si los hubiera) se encontrarán libres de defectos en los materiales o de fabricación, siempre y cuando se den circunstancias normales de uso, durante un período de noventa (90) días a partir de la fecha de adquisición, que se demostrará con la presentación de una copia del recibo de compra. Excepto en los casos especificados anteriormente, el Software se suministra "TAL CUAL". El único recurso a su disposición y la responsabilidad total de Sun de conformidad con la presente garantía limitada radicarán en el derecho que Sun se reserva para determinar la sustitución de los medios del Software o la devolución del importe abonado por este. Cualquiera de las garantías implicadas en el Software están limitadas a un término de 90 días. Algunos Estados no permiten limitaciones en la duración de las garantías implícitas, de modo que lo expresado anteriormente podría no corresponder para usted. Esta garantía limitada le otorga a usted derechos legales específicos. Es posible que tenga otros derechos que varían de un Estado a otro.
5. **EXCLUSIÓN DE GARANTÍAS.** A MENOS QUE EN EL PRESENTE CONTRATO SE ESTIPULE LO CONTRARIO, SE RENUNCIA A TODAS LAS CONDICIONES, MANIFESTACIONES Y GARANTÍAS EXPLÍCITAS O IMPLÍCITAS, INCLUIDA CUALQUIER GARANTÍA IMPLÍCITA DE COMERCIALIZACIÓN, IDONEIDAD PARA UN PROPÓSITO DETERMINADO O PARA LA CONTRAVENCIÓN DEL PRESENTE CONTRATO, SALVO EN AQUELLOS CASOS EN LOS QUE ESTAS EXCLUSIONES CAREZCAN DE VALIDEZ JURÍDICA.

TÉRMINOS ADICIONALES DE LA LICENCIA

Estos Términos Adicionales de la Licencia amplían o modifican los términos del Contrato de Licencia de Código Binario. Los términos en mayúsculas que no se definan en los presentes Términos Adicionales mantendrán el mismo significado que se les ha atribuido en el Contrato de Licencia de Código Binario. Los presentes Términos Adicionales sustituyen cualquier término del Contrato de Licencia de Código Binario o de cualquier licencia contenida dentro del Software con los que sean contradictorios o incongruentes.

-
- A. Uso interno del software y otorgamiento de la licencia de desarrollo. Sujeto a los términos y condiciones de este Acuerdo y a las restricciones y excepciones que se establecen en el archivo “README” del Software que se incorpora al presente por referencia, incluidas, sin limitación, las Restricciones de la Tecnología Java de estos Términos Adicionales, Sun le otorga una licencia no exclusiva, no transferible y limitada sin derechos de licencia, para reproducir y usar de manera interna el Software completo y sin modificar a los efectos de diseñar, desarrollar y probar sus Programas.
- B. Licencia para la distribución del Software. En virtud de los términos y condiciones del presente Contrato y las restricciones y excepciones estipuladas en el archivo README del Software, incluidas, aunque no exclusivamente, las Restricciones de la Tecnología Java de estos Términos Adicionales, Sun le concede una licencia limitada, no exclusiva e intransferible, sin tarifa de licencia, para reproducir y distribuir el Software, siempre y cuando: i) distribuya el Software completo y sin modificar y únicamente integrado como parte de sus Programas y con el sólo objeto de ejecutarlos, ii) los Programas añadan una funcionalidad sustancial y primaria al Software, iii) no distribuya software adicional con el propósito de sustituir cualquier componente del Software iv) no elimine ni modifique las notificaciones ni los avisos de propiedad incluidos en el Software; v) distribuya el Software sólo mediante un contrato de licencia que proteja los intereses de Sun de conformidad con los términos establecidos en el Contrato, y vi) acuerde defender e indemnizar a Sun y a sus licenciantes por cualquier daño, costo, responsabilidad, transacción extrajudicial o gasto (incluidos honorarios de abogados) que se deriven de cualquier reclamación, litigio o acción de terceros como consecuencia del uso o distribución de cualquiera o de todos los Programas y/o Software.
- C. Restricciones de la tecnología Java. Usted se compromete a no crear, modificar ni alterar el desempeño, ni autorizar a sus licenciatarios para crear, modificar ni alterar el desempeño de clases, interfaces ni subpaquetes que en cualquier modo se identifiquen como “Java”, “javax”, “sun” o similares según especifique Sun en cualquier designación de la convención de denominación.
- D. Código fuente. El Software puede contener código fuente que, a menos que se otorgue una licencia expresa para otros fines, se proporciona únicamente con fines de referencia en virtud de los términos del presente Contrato. El código fuente no podrá redistribuirse a menos que así se estipule explícitamente en el presente Contrato.”

BIBLIOGRAFÍA

http://es.wikipedia.org/wiki/Imagen:Mapa_conceptual_software_libre.png

Fecha de consulta: 12 de diciembre 2006 (4:47 p.m.)

http://es.wikipedia.org/wiki/Software_libre

Fecha de consulta: 12 de diciembre 2006 (6:24 p.m.)

<http://www.fsf.org/>

Fecha de consulta: 20 de diciembre 2006 (10:08 a.m.)

<http://www.gnu.org/philosophy/free-sw.es.html>

Fecha de consulta: 21 de diciembre 2006 (11:49 p.m.)

<http://www.biografiasyvidas.com/biografia/g/gates.htm>

Fecha de consulta: 27 de enero 2007 (11:52 a.m.)

http://es.wikipedia.org/wiki/Bill_Gates

Fecha de consulta: 13 de febrero 2007 (11:23 p.m.)

<http://etsiit.ugr.es/alumnos/mlii/Paul%20Allen.htm>

Fecha de consulta: 13 de febrero 2007 (8: 13 p.m.)

<http://www.codigolibre.org/GPL-vs-EULA/cluf-eula-traduccion.pdf>

Fecha de consulta: 13/febrero/2007 (11:12 a.m.)

http://es.wikipedia.org/wiki/Stephen_Wozniak

Fecha de consulta: 13/febrero/2007 (9:36 a.m.)

http://es.wikipedia.org/wiki/Steve_Jobs

Fecha de consulta: 16/febrero/2007 (11:38 p.m.)

http://es.wikipedia.org/wiki/Mac_OS_X (10:14 p.m.)

Fecha de consulta: 16/febrero/2007 (12:14 a.m.)

http://es.wikipedia.org/wiki/Safari_%28navegador%29

Fecha de consulta: 17/febrero/2007 (11:58 a.m.)

http://www.laflecha.net/perfiles/tecnologia/steve_jobs?page=1

Fecha de consulta: 18/febrero/2007 (11:58 a.m.)

<http://www.gnu.org/philosophy/historical-apsl.es.html>

Fecha de consulta: 18/febrero/2007 (9:17 a.m.)

<http://www.gnu.org/philosophy/apsl.es.html>

Fecha de consulta: 18/febrero/2007 (11:12 a.m.)

<http://www.opensource.org/licenses/apsl-2.0.php>

Fecha de consulta: 20/febrero/2007 (5:33 p.m.)

<http://es.wikipedia.org/wiki/Microsoft>

Fecha de consulta: 20/febrero/2007 (8:387 p.m.)

http://groups.google.com.mx/group/microsoft.public.es.windowsxp/browse_thread/thread/92687f91321561e/0e3233215b0dccb7?lnk=st&q=primeras+empresas+de+software&rnum=9&hl=es#0e3233215b0dccb7

Fecha de consulta: 20/febrero/2007 (11:54 p.m.)

<http://jcp.org/en/home/index>

Fecha de consulta: 20/febrero/2007 (2:41 a.m.)

<http://www.java.com/es/download/license.jsp>

Fecha de consulta: 18/febrero/2007 (6:58pa.m.)

<http://primates.ximian.com/~miguel/emexico2.html>

Fecha de consulta 20/Febrero/2007, 2:29 p.m.

<http://www.sei.cmu.edu/about/about.html>

Fecha de consulta: 17/diciembre/2006 (10:40 a.m.)

http://www.um.es/docencia/barzana/IAGP/Enlaces/CASE_principales.html

Fecha de consulta 20/diciembre/2006 (08:55 p.m.)

<http://personales.com/cuba/cardenas/internet/case.html>

Fecha de consulta 22/diciembre/2006 (01:00 a.m.)

http://www.svetlian.com/dll/articulos_descripcion_dll.htm

Fecha de consulta: 27 de diciembre 2006 (8:23 p.m.)

http://www.google.com.mx/url?sa=X&start=7&oi=define&q=http://www.optimize.es/conten/glossary/letram.jsp&usg=__cKdAkL7khRX4wGgCUAbXGk65ESw

Fecha de consulta: 28 de diciembre 2006 (9:15 p.m.)

<http://www.monografias.com/trabajos2/guiavb/guiavb.shtml>

Fecha de consulta: 27 de diciembre 2006 (8:35 p.m.)

<http://www.dooyoo.es/archivos-ordenadores/microsoft-visual-basic/433549/>

Fecha de consulta: 29 de diciembre 2006 (10:00 a.m.)

http://es.wikipedia.org/wiki/Visual_Basic

Fecha de consulta: 31 de diciembre 2006 (11:08 a.m.)

http://mx.preciomania.com/search_getprod.php/masterid=38986

Fecha de consulta: 31 de diciembre 2006 (1:20 p.m.)

<http://www.abcdatos.com/tutoriales/tutorial/17781.html>

Fecha de consulta: 01 de enero 2007 (2:45 p.m.)

<http://es.wikipedia.org/wiki/IIS>

Fecha de consulta: 02 de enero 2007 (3:00 p.m.)

<http://mundogeek.net/archivos/2004/10/04/una-no-tan-breve-historia-de-java/>

Fecha de consulta: 01 /Feb/2007, 9:00 p.m

<http://mundogeek.net/archivos/2004/10/04/una-no-tan-breve-historia-de-java/>

Fecha de consulta: 01 /Feb/2007, 9:35 p.m

http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/I_2.htm

Fecha de consulta: 01 /Feb/2007, 10:30 p.m.

<http://distritos.telepolis.com/java/lib/documentos/historia.htm>

Fecha de consulta: 03 /Feb/2007, 9:00 a.m.

http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/I_3.htm

Fecha de consulta: 03 /Feb/2007, 13:15 p.m.

<http://www.iec.csic.es/CRIPTONOMICON/java/quesjava.html>

Fecha de consulta: 06/Feb/2007, 10:35 p.m.

<http://www.revista.unam.mx/vol.1/num2/art4/>

Fecha de consulta: 06/Feb/2007, 10:55 p.m.

<http://www.postgresql.org/about/history>

Fecha de consulta: 07/Feb/2007, 8:30 p.m.

http://www.netpecos.org/docs/mysql_postgres/x15.html

Fecha de consulta: 10/Feb/2007, 10:15 p.m.

<http://es.tldp.org/Postgresql-es/web/navigable/tutorial/x56.html>

Fecha de consulta: 12/Feb/2007, 9:25 p.m.

<http://www.postgresql.org/about/advantages>

Fecha de consulta: 12/Feb/2007, 9:35 p.m.

<http://www.sobl.org/traduccion/practical-postgres/node19.html>

Fecha de consulta: 15/Feb/2007, 10:15 p.m.

http://es.wikipedia.org/wiki/Java_2_Enterprise_Edition

Fecha de consulta: 15/Feb/2007, 10:30 p.m.

http://snoopconsulting.com/snoop_es/index.php?option=com_content&task=view&id=132&Itemid=163

Fecha de consulta: 15/Feb/2007, 11:15 p.m.

<http://es.wikipedia.org/wiki/Tomcat>

Fecha de consulta: 15/Feb/2007, 11:40 p.m.

<http://www.tejedoresdelweb.com/307/article-5541.html>

Fecha de consulta: 17/Feb/2007, 9:45 a.m.

<http://www.help400.es/asp/scripts/nwart.asp?Num=131&Pag=10&Tip=T>

Fecha de consulta: 17/Feb/2007, 12:00 p.m.

<http://www.als-es.com/home.php?location=recursos/articulos/optimizacion-acceso-BBDD-pool-conexiones>

Fecha de consulta: 18/Feb/2007, 11:00 a.m.

http://www.proactiva-calidad.com/java/jndi/jndi_server_xml.html

Fecha de consulta: 18/Feb/2007, 1:00 p.m.

<http://www.forosdelweb.com/showthread.php?t=402380>

Fecha de consulta: 19/Feb/2007, 9:00 p.m.

<http://www.portalfox.com/index.php?name=News&file=article&sid=1271&mode=nested&order=0&thold=0>

Fecha de consulta: 19/Feb/2007, 10:30 p.m.

<http://www.java.com/es/download/license.jsp>

Fecha de consulta: 18/febrero/2007, 2:56 p.m.

Olguin, Romo, H. "Dirección, Organización y Administración de Centros de Tecnología de Información". México, UNAM, Facultad de Ingeniería, 2005, Apéndice H.

Manganelli, Raymond; Klein, Mark "Como Hacer reingeniería" Editorial Norma, Colombia CO. 1995.

Hammer, Michael & Champú, James "Reingeniería: Olvide lo que usted sabe sobre como debe funcionar una empresa ¡Casi todo ésta errado!" Editorial Norma. Bogotá. Co. 1994

Brodie, Michel & Stonebraker; Michael "Migrating Legacy Systems: Gateways, Interfaces and Incremental Approach" Morgan Kaufmman. 1994

Feiler, Peter H. "Reengineering: An Engineering Problem"
Software Engineering Institute Pennsylvania E.U.A. 1993

[CMM 01] CMMI Product Team "CMMI® for Development, Version 1.2"
Software Engineering Institute Pennsylvania E.U.A. 2006

Sea, Maldonado, Horacio, "Sistema de Información para Auditoria Informática Y control de Centro de tecnología de Información"
Trabajo de tesis UNAM, 1995