



UNIVERSIDAD DON BOSCO

Facultad de ingeniería

Escuela de computación

Software de apoyo para el control de auditoría de sistemas con énfasis en el área de tecnología de información para la superintendencia de valores en El Salvador.

Trabajo de graduación para optar al título de:

Ingeniero en Ciencias de la computación.

Presentado por:

Joaquín Eduardo Bolaños Amaya.

Rosa Margarita Pineda Jovel.

Asesor:

Ing. Raúl Martínez.

Diciembre 2007, Soyapango, San Salvador, El Salvador.

ÍNDICE

Introducción	i
1 Anteproyecto	1
1.1 Antecedentes	2
1.1.1 Orígenes del estudio	2
1.2 Importancia de la investigación	3
1.2.1 Importancia del tema	3
1.2.2 Justificación	3
1.3 Proyección de Desarrollo Empresarial	4
1.4 Definición del Tema	4
1.4.1 Definición del Tema	4
1.4.2 Objetivos	4
1.4.2.1 Objetivo General	4
1.4.2.2 Objetivos Específicos	4
1.4.3 Alcances	5
1.4.4 Limitaciones	5
1.4.5 Delimitaciones	6
1.5 Marco Teórico	6
1.5.1 Marco histórico	6
1.5.1.1 Evolución histórica de las computadoras	6
1.1.1.2 Breve historia de la auditoría	7
1.5.2 Marco conceptual	8
1.5.2.1 Definición de auditoría	8
1.5.2.2 Clasificación de la auditoría	8
1.5.2.3 Generalidades de la auditoría	10
1.5.2.4 Alcance de la auditoría interna	11
1.5.2.5 Proceso general de realización de auditorías	11
1.5.2.6 Normas de la auditoría	13
1.5.2.6.1 La auditoría de sistemas y las Normas Internacionales	14
1.5.2.7 Generalidades de la auditoría informática	15
1.5.2.8 La auditoría asistida por computadora	17
1.5.2.9 Importancia de las herramientas en el desarrollo de sistemas	18

1.5.2.10 El Ciclo de Vida de Desarrollo de Sistemas de Información	18
1.5.2.10.1 Definición de un modelo de ciclo de vida	18
1.5.2.11 La Programación Orientada a Objetos y el UML	20
1.5.2.11.1 Conceptos de Programación Orientada a Objetos	20
1.5.2.11.2 El Lenguaje Unificado de Modelado (UML)	21
1.5.3 Marco Experimental	24
1.6 Contenido Capitular	25
1.7 Metodología de Investigación y Desarrollo	29
1.7.1 Investigación y Desarrollo	29
1.7.2 Técnicas de Recopilación de Datos	29
1.7.3 Aplicación del Ciclo de Vida de Desarrollo de Software	31
1.7.4 Resumen explicativo de actividades por etapa	31
1.7.4.1 Actividades de la etapa 1	31
1.7.4.2 Actividades de la etapa 2	32
1.8 Cronograma de Actividades	33
1.9 Presupuesto	34
1.9.1 Costos para el desarrollo del sistema	34
1.9.2 Costos de mano de obra	34
1.9.3 Costos de Licencias	34
1.9.4 Otros gastos y contabilización total	35
2 Análisis de Encuestas y Diseño de la Base de Datos	
2.1 Investigación Preliminar	37
2.1.1 Elaboración de encuestas	37
2.1.1.1 Objetivo de la Encuesta	37
2.1.1.2 Investigación de la población	37
2.1.1.3 Cálculo del tamaño de la muestra	38
2.1.1.4 Formato de las encuestas	39
2.1.1.5 Análisis de las encuestas	39
2.2 Diseño de la Base de Datos	53
2.2.1 Especificación del problema a resolver	54
2.2.2 Modelo Conceptual o de Entidad – Relación	55
2.2.2.1 Identificación de los conjuntos de entidades	55

2.2.2.2 Identificación de los conjuntos de interrelaciones	55
2.2.2.3 Primer Diagrama Entidad – Relación	56
2.2.2.4 Identificación de atributos	56
2.2.2.5 Selección de claves candidatas	57
2.2.2.6 Segundo Diagrama Entidad – Relación	58
2.2.3 Modelo Lógico o Relacional	58
2.2.3.1 Establecimiento de relaciones	59
2.2.3.2 Aplicación de Formas Normales	59
2.2.3.2.1 Primera Forma Normal (1FN)	60
2.2.3.2.2 Segunda Forma Normal (2FN)	62
2.2.4 Segunda aplicación del modelo Entidad – Relación	74
2.2.4.1 Tercer diagrama Entidad – Relación	78
2.2.4.2 Base de Datos – Diagrama Esquemático	79
2.3 Construcción de la Base de Datos	80
2.3.1 Diccionario de Datos	80
2.3.2 Elaboración de la Base de Datos en MySQL	92
2.3.2.1 Nombre de la Base de Datos	92
2.3.2.2 Especificación de las tablas	92
2.3.2.3 Creación de interrelaciones	92
2.3.2.4 Guión de creación de la Base de Datos	93
3 Diseño y Construcción del Sistema	111
3.1 Modelado de la Aplicación	112
3.1.1 Especificación del funcionamiento de la aplicación	112
3.1.2 Determinación de clases	114
3.1.3 Diagramación UML	117
3.1.3.1 Diagramas de casos de uso	117
3.1.3.2 Diagramas de clases	120
3.1.3.3 Diagramas de secuencia	125
3.1.3.4 Diagramas de actividad	137
3.1.3.5 Diagramas de colaboración	145
3.2 Construcción de la aplicación	149
3.2.1 Raíz del sitio	149

3.2.2 Página de autenticación	149
3.2.3 Estilos del sitio y recursos gráficos	150
3.2.4 Manejo de sesiones	150
3.2.5 Organización del contenido	150
3.2.6 Creación e inclusión de clases	151
3.2.7 Manejo de la programación del lado del cliente	151
3.2.8 Estructura del sitio	151
4 Documentación del Sistema	153
Manual de Usuario	187
Manual de Mantenimiento	200
Manual de Instalación	194
5 Conclusiones y Recomendaciones	211
Conclusiones	212
Recomendaciones	213
Anexos	214
Anexo 1: Formato de encuesta de investigación de hechos	215
Anexo 2: Artículo Técnico	219
Bibliografía y Fuentes de Consulta	223
A. Bibliografía	223
B. Recursos de Internet	224
Glosario	226

INTRODUCCIÓN

En el presente documento se desarrollan los aspectos de diseño y construcción del Software de Apoyo para la Auditoría de Sistemas de la Superintendencia de Valores de El Salvador.

Debido a que dicha institución se dedica a la supervisión del mercado de valores, es recomendable que utilicen software especializado para la realización de auditorías de sistemas. Nuestro país está dando pasos para lograr un mayor desarrollo tecnológico, sin embargo referente a las auditorías de sistemas escasea el software especializado. Por lo cual, este proyecto será una propuesta innovadora para la realización de las auditorías en la Superintendencia de Valores

El software diseñado es muy práctico y sencillo, siendo las actividades para su desarrollo:

- Capítulo I - Anteproyecto: En este capítulo se desarrollan los elementos históricos, conceptuales y experimentales que apoyan el desarrollo del proyecto. Además se da a conocer la metodología de investigación y desarrollo, el cronograma de actividades y el cálculo de presupuesto.
- Capítulo II - Análisis de encuestas y Diseño de la base de datos: Este capítulo explica el desarrollo y análisis de las encuestas. También se describe el proceso de diseño de la base de datos, aplicando el modelo entidad – relación y el modelo relacional.
- Capítulo III - Diseño y Construcción del Sistema: En este apartado se explica el proceso de modelado de la aplicación, utilizando UML. Además se describe la codificación del software.
- Capítulo IV: Documentación Técnica del Sistema: Aquí se muestra la información que soporta al software desarrollado, comprendida por: Manual de usuario, de mantenimiento y de instalación, a fin de dar una mejor comprensión del funcionamiento tanto al usuario como al programador.

CAPITULO I

ANTEPROYECTO

1.1 ANTECEDENTES

1.1.1 Orígenes del estudio

El interés del tema a desarrollar surgió de investigaciones anteriormente realizadas en el campo de la auditoría¹ de sistemas informáticos que realiza la Superintendencia de Valores de El Salvador.

En dichas actividades se observó la carencia de un software que apoyara los procedimientos formalmente establecidos para realizar auditorías de sistemas.

Para realizar las auditorías según tales procedimientos, se lleva a cabo actualmente una revisión manual, apoyada en una guía previamente diseñada. Posteriormente se procede a la cuantificación de los resultados y se extraen de los mismos las conclusiones que se reflejan en un informe final de auditoría.

Se advirtió el hecho que esta área es de las pocas que no están automatizadas en la Superintendencia de Valores. Existiendo una oportunidad para aplicar los conocimientos adquiridos y al mismo tiempo beneficiar a la entidad. Debido a que actualmente en la Superintendencia de Valores las auditorías se realizan sin hacer uso de formularios se ha considerado que el software a diseñar tenga contemplados estos elementos para facilitarle el trabajo al auditor.

El mercado de software comercial para el área de auditoría informática cuenta con reconocidos productos, los cuales, sin embargo, no están muy extendidos en nuestro medio. Para mayor información, sobre dichos paquetes de software, referirse al apartado *marco experimental* en este mismo documento.

¹ Cada vez que en este documento se emplee por primera vez un término técnico que no se define dentro de su contexto o con una nota al pie de la página, y que esté incluido en el glosario final, dicho término aparecerá en cursiva y subrayado.

1.2 IMPORTANCIA DE LA INVESTIGACIÓN

1.2.1 Importancia del tema

La importancia del tema radica en sistematizar los procedimientos de auditoría de sistemas que realiza la Superintendencia de Valores. El utilizar la capacidad intrínseca de la computadora actual para ejecutar aplicaciones rápidas, confiables y amigables con el usuario puede ahorrar tiempo de capacitación al auditor de sistemas y al mismo tiempo, darle mayor fluidez a su labor.

1.2.2 Justificación

En El Salvador, las técnicas de auditoría asistida por computadora, se han comenzado a expandir en las empresas, debido a que los datos hoy en día son de mucha importancia para las organizaciones; como prueba de ello se han creado varias superintendencias en los diferentes rubros por ejemplo (electricidad, finanzas y mercantil), para que controlen el cumplimiento a los procesos administrativos de las empresas del país.

La Superintendencia de Valores es una institución autónoma que se ha convertido en una entidad reguladora de gran importancia a nivel nacional, ya que en los últimos años ha aumentado el desarrollo del mercado de valores. Debido a esto sus actividades de auditoría en el área de sistemas de información han crecido y justifica la decisión de automatizar dicho trabajo que tradicionalmente se ha hecho manualmente.

Finalmente, ha de decirse que la entidad mencionada carece de un software diseñado a la medida del trabajo de auditoría de sistemas informáticos y de un manual de procedimientos de auditoría de sistemas. Estas carencias podrán ser solventadas con el desarrollo del sistema.

1.3 PROYECCIÓN DE DESARROLLO EMPRESARIAL

El desarrollo del proyecto será de utilidad a la Superintendencia de Valores de El Salvador, ya que sistematizará la labor del auditor de sistemas.

El beneficio del diseño del software para las auditorías de sistemas informáticos será para la entidad reguladora de la misma, debido a que contará con tecnología especializada para poder ejercer una supervisión eficiente. Actualmente no cuenta con un software como tal y considerando que el mercado de valores está teniendo un mayor desarrollo a nivel nacional, dicha institución deberá contar con el apoyo de tecnología informática.

1.4 DEFINICIÓN DEL TEMA

1.4.1 Definición del tema

Desarrollo de un software para el control de auditorías de sistemas con énfasis en el área informática para la Superintendencia de Valores de El Salvador.

1.4.2 Objetivos

1.4.2.1 Objetivo general

Desarrollar una aplicación informática que sistematice el proceso de auditorías de sistemas enfocadas en el área del software para la Superintendencia de Valores.

1.4.2.2 Objetivos específicos

- Identificar los recursos hardware y software con que cuenta la Superintendencia de Valores de El Salvador para el empleo de la aplicación a desarrollar.
- Determinar las entradas y salidas que se manejarán en la aplicación informática a diseñar.
- Modelar el flujo de datos y establecer el esquema entidad – relación de la aplicación.

- Proporcionar una interfaz interactiva por computadora que guíe al usuario a través del desarrollo de una auditoría informática.
- Realizar pruebas de funcionamiento sobre la aplicación desarrollada.
- Elaborar manuales de usuario, de mantenimiento y de instalación del software construido.
- Redactar un artículo técnico que sintetice los aportes e impacto obtenido con la ejecución del proyecto.

1.4.3 Alcances

- Las cédulas que manejará el software son: Cédula sumaria de Hardware, cédula sumaria de Software, cédula sumaria de políticas y procedimientos, cédula de hallazgos y cédula de marcas.
- El reporte que generará la aplicación será el informe de hallazgos.
- El software contará con documentación acorde como los manuales de usuario, de mantenimiento y de instalación.
- El software contará con capacidades de mantenimiento para la base de datos: adicionar, modificar, eliminar, consultar y almacenar registros.

1.4.4 Limitaciones

Algunas limitaciones que pueden surgir al momento del desarrollo de este proyecto son:

- Acceso restringido a información referente al desarrollo de las auditorías de sistemas que realiza la Superintendencia de Valores.
- El avance del proyecto se dará en la medida que la Superintendencia de Valores proporcione la información necesaria.
- Escasez de literatura adaptada a la realidad salvadoreña sobre el tema de software para auditoría de sistemas.

1.4.5 Delimitaciones

- Se tomarán como prototipos los procedimientos de auditoría de la Superintendencia para diseñar el software.
- El software operativo que soportará el funcionamiento de la aplicación a diseñar será Windows XP, debido a que bajo dicha plataforma trabaja la Superintendencia de Valores.
- Se usará PHP como software de programación del sistema y MySQL como gestor de base de datos. Esto debido a su calidad reconocida y a que posee licencia libre.
- El diseño lógico del software de especialización estará basado en la normativa de la Superintendencia de Valores.
- La ejecución del software prototipo a desarrollar será dentro de un servidor web montado a nivel local. Será decisión de la Superintendencia de Valores implantar el proyecto una vez terminado el mismo.

1.5 MARCO TEÓRICO

1.5.1 Marco histórico

1.5.1.1 Evolución histórica de las computadoras

La computadora, como la conocemos actualmente, se ha desarrollado en el transcurso del tiempo, desde las máquinas de tipo mecánico hasta las electrónicas. Para entender mejor esta evolución de las computadoras se han definido grupos de computadoras llamadas “generaciones”, los cuales tienen características comunes de diseño y rangos de tiempo. A continuación se describe un resumen de cada generación de las computadoras.

1ª. Generación: Se desarrolla entre 1940 y 1952. Es la época de los ordenadores que funcionaban a válvulas (tubos al vacío y relés) y el uso era exclusivo para el ámbito científico/militar.

2ª. Generación: Va desde 1952 a 1964. Ésta surge cuando se sustituye la válvula por el transistor. En esta generación aparecen los primeros ordenadores comerciales, los cuales ya tenían una programación previa. Éstos interpretaban instrucciones en lenguaje de programación (Cobol, Fortran).

3ª. Generación: Se dio entre 1964 y 1971. Es la generación en la cual se comienzan a utilizar los circuitos integrados (IC); esto permitió por un lado abaratar costos y por el otro aumentar la capacidad de procesamiento reduciendo el tamaño físico de las máquinas. Por otra parte, esta generación es importante porque se da un notable mejoramiento en los lenguajes de programación y, además, surgen los programas utilitarios.

4ª. Generación: Se desarrolla entre los años 1971 y 1981. Esta fase de evolución se caracterizó por la integración de los componentes electrónicos, y esto dio lugar a la aparición del *microprocesador*, que es la integración de todos los elementos básicos del ordenador en un sólo circuito integrado.

5ª. Generación: comienza en 1981 y se caracteriza por el surgimiento de la *PC (Personal Computer, Computadora Personal)*, tal como se la conoce actualmente. Un suceso importante de esta etapa fue la aparición de procesadores Pentium, con capacidades de procesamiento multimedia.

1.5.1.2 Breve historia de la auditoría

La importancia de la auditoría es reconocida desde los tiempos más remotos, teniéndose conocimientos de su existencia ya en las lejanas épocas de la civilización sumeria.

Todavía se acredita que el término auditor evidenciando el título del que practica esta técnica, apareció a finales del siglo XVIII, en Inglaterra durante el reinado de Eduardo I. En diversos países de Europa, durante la edad media, muchas eran las asociaciones profesionales, que se encargaban de ejecutar funciones de auditoría, destacándose entre ellas los consejos Londinenses (Inglaterra), en 1.310, el Colegio

de Contadores, de Venecia (Italia), 1.581. La revolución industrial llevada a cabo en la segunda mitad del siglo XVIII, imprimió nuevas direcciones a las técnicas contables, especialmente a la auditoría, pasando a atender las necesidades creadas por la aparición de las grandes empresas.

Se puede decir que en la actualidad las auditorías siguen siendo de gran importancia, especialmente la de sistemas, debido a que el volumen de información contable o financiera ha crecido en proporción directa al desarrollo de las empresas, por lo que se vuelve necesario verificar el cumplimiento de las NIAS² con el apoyo de la tecnología.

1.5.2 Marco conceptual

1.5.2.1 Definición de auditoría.

¿Que es auditar? es el proceso de acumular y evaluar evidencia, realizado por una persona independiente y competente acerca de la información cuantitativa de una entidad económica específica, con el propósito de determinar e informar sobre el grado de correspondencia existente entre la información cuantificable y los criterios establecidos.

La auditoría es una de las aplicaciones de los principios científicos de la contabilidad, basada en la verificación de los registros patrimoniales de las haciendas, con el fin de observar su exactitud.

1.5.2.2 Clasificación de la auditoría.

La auditoría se clasifica en las siguientes áreas:

- Auditoría Fiscal.
- Auditoría Interna.
- Auditoría Administrativa.
- Auditoría Operacional.

² NIAS: Normas Internacionales de Auditoría.

- Auditoría Financiera.
- Auditoría Forense.
- Auditoría de Sistemas.

Auditoría Interna: Es el examen crítico, sistemático y detallado de un sistema de información de una unidad económica, realizado por un profesional con vínculos laborales con la misma, utilizando técnicas determinadas y con el objeto de emitir informes y formular sugerencias para el mejoramiento de la misma.

La auditoría interna abarca los tipos de:

- Auditoría Administrativa: Es la que se encarga de verificar, evaluar y promover el cumplimiento y apego al correcto funcionamiento de las fases o elementos del proceso administrativo y lo que incide en ellos es su objetivo también el evaluar la calidad de la administración en su conjunto.

- Auditoría Operacional: Es la que se encarga de promover la eficiencia en las operaciones, además de evaluar la calidad de las mismas y los procesos adecuados que se utilizan (flujos de procesos).

- Auditoría Financiera: Es el examen total o parcial de la información financiera y la correspondiente operacional y administrativa, así como los medios utilizados para identificar, medir, clasificar y reportar esa información.

- Auditoría Fiscal: Se realiza con el objeto de verificar el correcto cumplimiento de las obligaciones en materia tributaria de los contribuyentes.

- Auditoría Forense: Se define como una auditoría especializada en descubrir, divulgar y atestar sobre fraudes y delitos en el desarrollo de las funciones públicas considerándose un verdadero apoyo a la tradicional auditoría gubernamental, en especial ante delitos tales como: Enriquecimiento ilícito, peculado, cohecho, soborno, desfalco, malversación de fondos, prevaricato, conflicto de intereses.

- Auditoría de Sistemas: Es el proceso de revisión y evaluación, parcial o completo de los aspectos relacionados con el procesamiento automatizado de la información.

1.5.2.3 Generalidades de la auditoría

La auditoría interna se define como una actividad de evaluación independiente establecida dentro de una organización para examinar y evaluar sus actividades como un servicio para la organización.

Las normas de auditoría establecen que la función de auditoría interna es parte del “componente de seguimiento” del control interno de una organización.

El trabajo de la auditoría interna incluye no solo controles financieros, sino que también evalúan y prueban la efectividad de los controles diseñados para ayudar a la organización a satisfacer todos sus objetivos.

La auditoría interna ha evolucionado para satisfacer las necesidades de las empresas y de las organizaciones gubernamentales y entidades sin ánimos de lucro. Originalmente la necesidad de la auditoría interna surgió cuando los gerentes de las primeras grandes corporaciones reconocieron que no eran suficientes las auditorías anuales de los estados financieros.

Posteriormente, el papel de los auditores internos se amplió como resultado de las exigencias de las bolsas de valores. Estas exigencias dieron como resultado la ampliación de las responsabilidades del auditor interno. A medida que el tamaño de las organizaciones se hizo más grande y más complejo, estas encontraron problemas operacionales adicionales que se prestaron para ser resueltos por auditoría interna.

La ampliación de la auditoría interna a estas actividades operacionales exigió auditores internos con conocimiento especializado en otras disciplinas como economía, derecho, finanzas, estadística y computación.

1.5.2.4 Alcance de la auditoría interna

- Revisar la confiabilidad y la integridad de la información financiera y de las operaciones y el medio utilizado para identificar, medir clasificar y reportar esa información.
- Revisar los sistemas establecidos para garantizar el cumplimiento de esas políticas, planes, procedimientos, leyes y regulaciones que podrían tener un impacto significativo sobre las operaciones y los informes y determinar si la organización los está cumpliendo.
- Revisar los medios de protección de activos y verificar la existencia de esos activos, cuando sean necesario.
- Evaluar la economía y la eficiencia con la cual se emplean los recursos.
- Revisar operaciones o programas para asegurar si los resultados son consistentes con los objetivos y metas establecidas, y si las operaciones o programas se están realizando en forma planificada.

1.5.2.5 Proceso general de realización de auditorías.

La figura 1 esquematiza el proceso general de la realización de cualquier tipo de auditoría (no necesariamente informática).

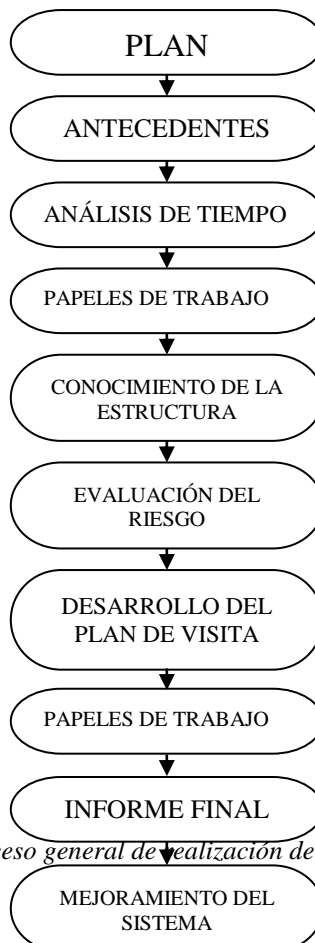


Figura 1: Proceso general de realización de auditorías.

Plan: Es la preparación del desarrollo de una auditoría.

Antecedentes: Es la información general de la empresa a auditar.

Análisis de tiempo: Tiempo estimado para el desarrollo de una auditoría.

Papeles de trabajo: Documentación que se ha recopilado durante el desarrollo de la auditoría.

Conocimiento de la estructura: Información conocida sobre la forma de cómo está organizada la empresa a auditar.

Evaluación del riesgo: Análisis del área a auditar.

Desarrollo del plan de visita: Ejecución del plan de auditoría.

Informe final: Informe sobre los hallazgos encontrados en el desarrollo de la auditoría.

Mejoramiento del sistema: Son las modificaciones realizadas al sistema, en base a las observaciones obtenidas en el resultado de la auditoría.

Las principales etapas del proceso de la auditoría son:

- Planeación
- Ejecución
- Elaboración de informe.

1.5.2.6 Normas de la auditoría

El desarrollo de una auditoría se basa en la aplicación de normas, técnicas y procedimientos de auditoría. Para nuestro caso, estudiaremos aquellas enfocadas a la auditoría en informática o sistemas.

Es fundamental mencionar que para el auditor de sistemas conocer los productos de software que han sido creados para apoyar su función aparte de los componentes de la propia computadora resulta esencial, esto por razones económicas y para facilitar el manejo de la información.

El auditor desempeña sus labores mediante la aplicación de una serie de conocimientos especializados que vienen a formar el cuerpo técnico de su actividad.

El auditor adquiere responsabilidades, no solamente con las personas que directamente contratan sus servicios, sino con un número de personas desconocidas para él, que van a utilizar el resultado de su trabajo como base para tomar decisiones.

La auditoría no es una actividad meramente mecánica, que implique la aplicación de ciertos procedimientos cuyos resultados, una vez llevados a cabo, son de carácter indudable.

La auditoría requiere el ejercicio de un juicio profesional, sólido maduro, para juzgar los procedimientos que deben seguirse y estimar los resultados obtenidos.

Las normas de auditoría son los requisitos mínimos de calidad relativos a la personalidad del auditor, al trabajo que desempeña y a la información que rinde como resultado de este trabajo.

Las normas de auditoría se clasifican en:

Normas personales: Son cualidades que el auditor debe tener para ejercer sin dolo una auditoría, basados en sus conocimientos profesionales, así como en un entrenamiento técnico, que le permita ser imparcial a la hora de dar sus sugerencias y está íntimamente relacionado a la ética.

Normas de ejecución del trabajo: Son la planificación de los métodos y procedimientos, tanto como papeles de trabajo a aplicar dentro de la auditoría.

Normas de información: Son el resultado que el auditor debe entregar a los interesados para que estos se den cuenta de la situación de la entidad como producto final de su trabajo, también conocido como informe o dictamen de auditoría.

1.5.2.6.1 La auditoría de sistemas y las Normas Internacionales.

Según la NIA 4001 se establece que el uso de la informática como ayuda de las técnicas de auditoría, provee una mayor eficiencia en el desempeño de dichos procedimientos.

De acuerdo a la norma DIPA³ 1001 párrafos 15 a 32, el control interno del ambiente informático puede ser mejorado utilizando los siguientes procedimientos:

- **Autorización de la administración para operación de microcomputadoras:** Se refiere a todas aquellas condiciones que norman el control y uso de las microcomputadoras desde el punto de vista de la administración.

- **Seguridad Física - Equipo:** Son todas aquellas medidas que previenen robos, daños físicos, accesos no autorizados o mal uso de las computadoras.

³ DIPA: Declaración Internacional de Prácticas de Auditoría

- **Seguridad Física – Medios removibles y no removibles:** Establecen medidas de protección para los programas y datos de una computadora, los cuales están almacenados en medios removibles o no removibles.
- **Seguridad de programas y de datos:** Se enumeran técnicas útiles para limitar el acceso a programas y datos solo a personal autorizado.
- **Integridad del Software y de los datos:** se describe la importancia del grado de exactitud y confiabilidad de los resultados producidos por los programas de aplicación, alimentación, procesamiento de datos y generación de informes.
- **Respaldo del hardware, software y datos:** Se refiere a los planes hechos por las entidades para obtener acceso a hardware, software y datos comparables en caso de falla, pérdida o destrucción.

1.5.2.7 Generalidades de la auditoría informática.

La auditoría de sistemas es el proceso de revisión y evaluación, parcial o completo de los aspectos relacionados con el procesamiento automatizado de la información⁴.

La auditoría de sistemas es fundamental para garantizar el correcto funcionamiento de los sistemas de información al proporcionar los controles necesarios que permiten garantizar la seguridad, integridad, disponibilidad y confiabilidad de los mismos.

La información es uno de los principales activos con los que cuentan las organizaciones, este recurso crítico involucra los siguientes elementos que deben ser protegidos⁵:

- Datos: Que son todos los objetos de la información.
- Aplicaciones: Son el conjunto de sistemas de información.
- Tecnología: Es el conjunto de hardware y software de base.

⁴ ISACA, 2002: Information Systems audit. And Control Association

⁵ COBIT, 1996: Control Objectives for Information and related Technology

- Instalaciones: Son los recursos necesarios para alojar a los sistemas de información.
- Recursos humanos: Es el personal relacionado directamente con el desarrollo y producción de los sistemas de información.

Los objetivos de la auditoría de sistemas son:

- Que el procesamiento de datos cumpla con las normas y procedimientos institucionales y legales vigentes.
- Que existan procedimientos adecuados para la selección, uso y resguardo de los recursos informáticos de la organización.
- Que la consistencia, confiabilidad y seguridad de los datos sean suficientes.
- Que esté asegurada la adecuada y eficaz operación de los sistemas informáticos de la organización.

Los auditores de sistemas de información⁶ examinan y evalúan el desarrollo, implementación, mantenimiento y operación de los componentes de sistemas automatizados y sus interfases con sistemas externos y no automatizados.

Se establecen tres tipos de funciones para los auditores de sistemas⁷ :

- Participar en la revisión del diseño, realización, implantación y explotación de las aplicaciones informáticas.
- Revisar y evaluar los controles implementados en los sistemas informáticos.
- Revisar y evaluar la eficacia, utilidad, fiabilidad y seguridad de los equipos e información.

1.5.2.8 La auditoría asistida por computadora

La utilización de equipos de computación en las instituciones, ha tenido una repercusión importante en el trabajo del contador público, no sólo en lo que se refiere

⁶ Según ISACA 2002

⁷ Mario Pattíni, 2003, Auditoría Informática, un enfoque práctico.

a los sistemas de información, sino también al uso de las computadoras en la auditoría. Al llevar a cabo auditorías donde existen sistemas computarizados, el profesional de la auditoría se enfrenta a problemas de muy diversa índole; uno de ellos, es la revisión de los procedimientos en sistemas administrativos (Control Interno) establecidos en la empresa que audita.

La utilización del computador como herramienta en el proceso de auditoría de sistemas se denomina técnicas de auditoría apoyadas en el computador (CAAT)⁸.

Las técnicas CAAT se pueden clasificar en las siguientes áreas:

1. CAAT para la auditoría de controles generales

- Controles de implantación
- Controles de seguridad de programas
- Controles de seguridad de bases de datos
- Controles de administración de operaciones
- Controles de programas de soporte del sistema

2. CAAT para la auditoría de aplicaciones

- Herramientas utilizadas para probar los procesos
 - o Pruebas de datos
 - o Facilidades de pruebas integradas
 - o Simulación en paralelo

3. CAAT que facilitan las funciones de administración de la auditoría de sistemas.

- Planificación de la auditoría
- Control s/ la ejecución de la auditoría
- Documentación de la auditoría
- Comunicación de resultados de la auditoría

⁸ CAAT: Técnicas de auditoría asistidas por computadora

1.5.2.9 Importancia de las herramientas en el desarrollo de sistemas

Las herramientas mejoran la forma en que ocurre el desarrollo y tienen influencia sobre la calidad del resultado final; extienden en tres formas la capacidad del analista de sistemas. Proporcionan el potencial para mejorar la productividad del analista, facilitan el desarrollo de procesos más eficaces y mejoran la calidad del sistema. En otras palabras, tanto el proceso de desarrollo de sistemas como el producto que se obtiene con él (Figura. 2), pueden mejorarse con el uso de herramientas apropiadas.

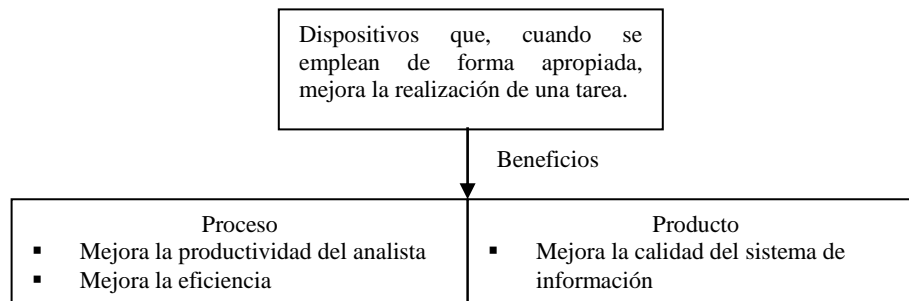


Figura 2. Beneficios obtenidos con el uso de herramientas en el desarrollo de sistemas de información

1.5.2.10 El Ciclo de Vida de Desarrollo de Sistemas de Información

1.5.2.10.1 Definición de un modelo de ciclo de vida.

Un modelo de ciclo de vida de software es una vista de las actividades que ocurren durante el desarrollo de la investigación, intenta determinar el orden de las etapas involucradas y los criterios de transición asociados entre estas etapas.

El modelo de ciclo de vida de nuestro software para auditorías de sistemas cumplirá con las siguientes funciones:

- Describirá las fases principales de desarrollo del software
- Ayudará a administrar el progreso del desarrollo
- Proveerá un espacio de trabajo para la definición de un detallado proceso de desarrollo del software

Así, el modelo que será utilizado para este proyecto servirá para ordenar las diversas actividades técnicas, por otra parte suministrará un marco para la administración del desarrollo y el mantenimiento, en el sentido en que permiten estimar recursos, definir puntos de control intermedios, monitorear el avance.

Modelo Cascada

Este es el modelo más utilizado por los analistas de sistemas. Se esquematiza en la figura 3. La visión del modelo cascada del desarrollo de software es muy fácil de comprender; dice que el desarrollo de software puede ser a través de una secuencia simple de fases. Cada fase tiene un conjunto de metas bien definidas, y las actividades dentro de una fase o quizás a una secuencia de metas de la fase. Las flechas muestran el flujo de información entre las fases. La flecha de avance muestra el flujo normal. Las flechas hacia atrás representan la retroalimentación.

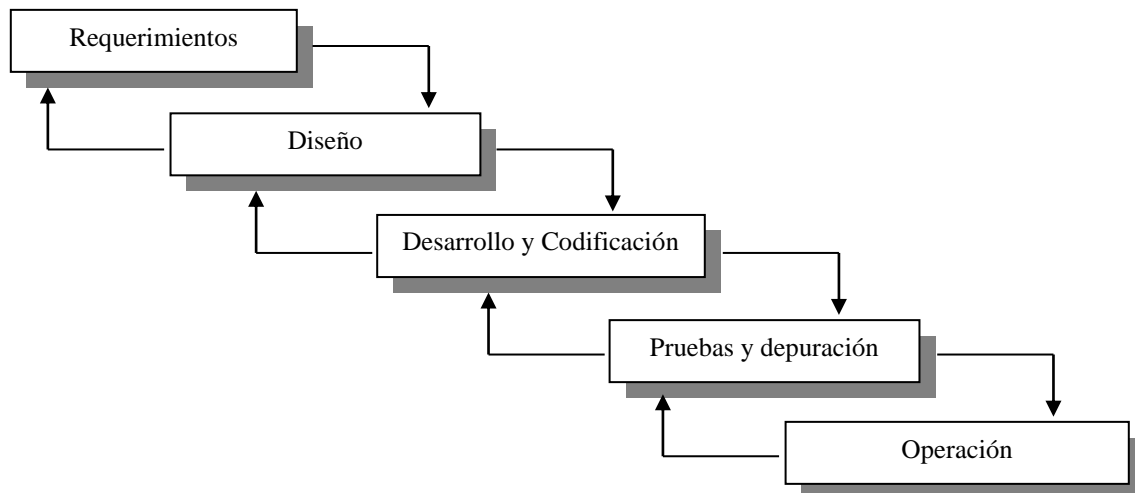


Figura 3: Esquema del modelo cascada del Ciclo de Vida de Desarrollo de Sistemas de Información

Las actividades de este modelo son:

- Investigación de requerimientos: Se utilizan técnicas de recopilación de datos (entrevistas, cuestionarios, observaciones) que permiten obtener información necesaria para definir las características y requerimientos del nuevo software.
- Diseño: Esta etapa tiene como objeto lograr la integración de los requerimientos del nuevo sistema y que estos sean más efectivos, para que los auditores se interesen por la aplicación.
- Desarrollo y Codificación: En esta etapa se convierte el diseño esquemático en la solución propuesta por medio de su codificación.
- Pruebas y Depuración: Realización de las pruebas piloto del sistema con el fin de asegurar que lo programado funcionará correctamente.
- Operación del sistema: Utilización formal del sistema por parte de los usuarios.

1.5.2.11 La Programación Orientada a Objetos y el UML

1.5.2.11.1 Conceptos de Programación Orientada a Objetos

La Programación Orientada a Objetos (POO) toma como modelo a los objetos reales para su contraparte en software. Toma ventaja de las relaciones entre clases, en donde objetos de cierta clase, tal como una clase de vehículos, tiene atributos y operaciones similares. Toma ventaja de las relaciones de herencia, e incluso de las relaciones de herencia múltiple, en donde las nuevas clases de objetos creadas se derivan mediante la absorción de atributos y operaciones de clases existentes y mediante la adición de sus propias características únicas.

La POO nos brinda una forma natural de ver el proceso de programación, a saber, mediante el modelado de objetos reales, sus atributos y comportamiento. La POO proporciona también comunicación entre los objetos. Tal como las personas se envían mensajes entre sí. Esto último se implementa a través de llamadas a métodos.

La POO encapsula datos (atributos) y funciones (métodos) en paquetes llamados objetos; los datos y las funciones de un objeto están íntimamente ligados. Los objetos tienen la propiedad de ocultar información. Esto significa que aunque los

objetos puedan “saber” cómo comunicarse con otros a través de interfaces bien definidas, por lo general, los objetos no saben cómo se implementan otros objetos. Los detalles de implementación se ocultan dentro de los mismos objetos.

El concepto básico de la POO es la clase, que es la estructura de programación que contiene los datos y los métodos que manejan dichos datos. En la actualidad, los lenguajes más populares poseen capacidades de trabajar con objetos. Ese es el caso de PHP 4, con el cual se trabajará el presente proyecto. Existen otros lenguajes que están completamente diseñados para trabajar con objetos, como Java y C#.

1.5.2.11.2 El Lenguaje Unificado de Modelado (UML)

El UML es un esquema de representación gráfica ampliamente utilizado para modelar sistemas orientados a objetos. Unifica los distintos esquemas de notación que existían a finales de la década de los ochenta.

Una de las características más atractivas de UML es su flexibilidad. UML se puede extender y es independiente de los muchos procesos de ADOO. Los modeladores en UML pueden desarrollar sistemas mediante el uso de distintos procesos; pero la idea de utilizar UML es que todos los diseñadores puedan expresar dichos procesos en un lenguaje gráfico común. UML es un conjunto extenso de diagramas. Entre los más utilizados tenemos:

Diagramas de colaboración: Describe las relaciones existentes entre los objetos. Indica qué objetos utilizan a qué otros objetos. Cualquier relación existente entre ellos se indica como una línea que conecta cuadros rectangulares. Si se desea, las invocaciones de los métodos pueden mostrarse en forma de flechas pequeñas que indican la dirección de la invocación. El nombre del método aparece al lado de la flecha. Este diagrama puede verse en la figura 4.



Figura 4: Ejemplo de diagrama de colaboración

Diagrama de clases: Este diagrama muestra las relaciones de herencia entre las clases, las subclases y superclases. Cada clase se muestra en forma de cuadro rectangular que contiene su nombre. Para mostrar que una clase extiende a otra, se dibuja una línea desde la subclase hasta la superclase, con la punta de flecha apuntando a la superclase. Este diagrama se muestra en la figura 5.

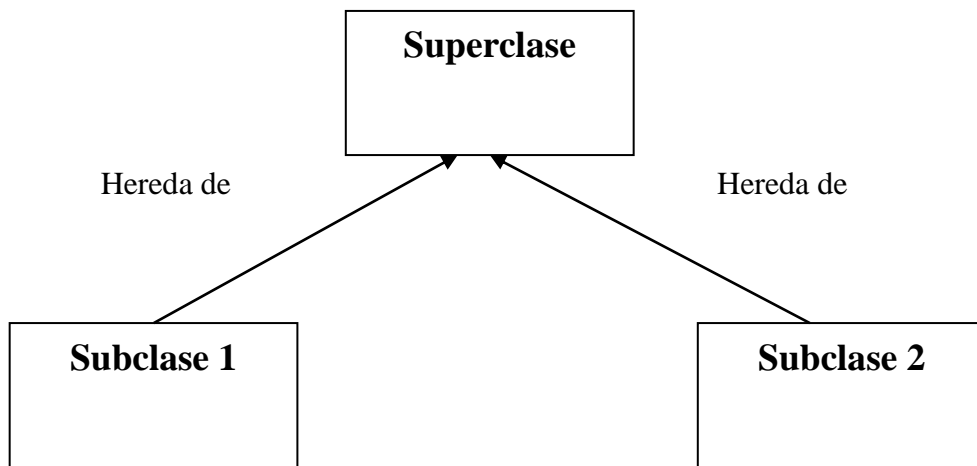


Figura 5: Diagrama de clases

Las clases de un diagrama de clases pueden describirse con más detalle, como lo muestra la figura 6. Este detalle incluye los datos (atributos) y funciones (métodos) de la clase.

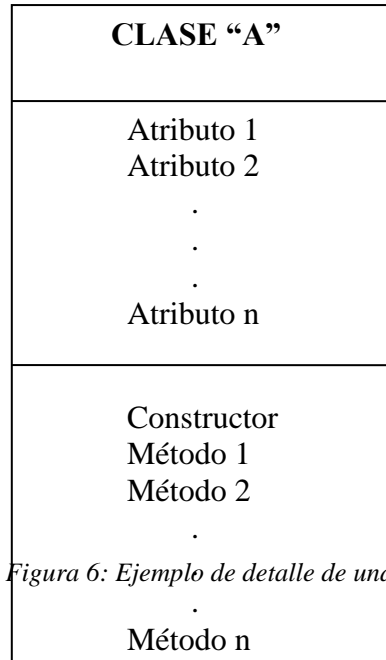
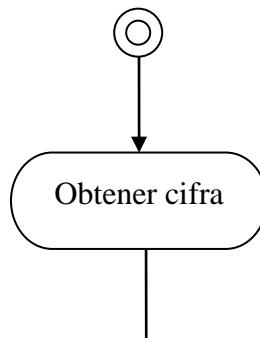


Figura 6: Ejemplo de detalle de una clase

Diagrama de actividad: Este tipo de gráfica describe una secuencia de actividades. Esto concuerda con el flujo usual de control dentro de un programa a medida que ejecuta una secuencia de instrucciones, incluyendo invocaciones a métodos. Un diagrama de actividad también puede mostrar:

- Condiciones (que corresponden a instrucciones *si.. entonces*).
- Ciclos (que corresponden a instrucciones *hacer para* y *hacer mientras*).
- Actividad concurrente (que corresponde a los subprocessos)

La figura 7 muestra un ejemplo de este diagrama:



cifra >= 10

cifra >= 10

Figura 7: Un diagrama de actividades de UML

1.5.3 Marco experimental⁹

Entre las aplicaciones de software comercial que el mercado ha producido para el campo de auditoría de sistemas, tenemos:

ERA: Su nombre proviene de Enfoque de Administración de Riesgos, por sus siglas en inglés. Como su nombre lo indica, este software sistematiza los enfoques de riesgos y el de auditoría interna así como también el planeamiento de auditoría con evaluaciones de riesgos del negocio y con criterios tradicionales basados en riesgos. Puede trabajar en plataforma Cliente/Servidor, basada en la Web, Multiusuario o Estándar

AutoAudit: Esta herramienta de computación es descrita como un sistema completo para la automatización de la función de auditoría, que soporta todo el proceso y flujo de trabajo, desde la fase de planificación, pasando por el trabajo de campo, hasta la preparación del informe final. Además del manejo de documentos y papeles de trabajo en forma electrónica. AutoAudit permite seguir la metodología de evaluación

⁹ Para esta sección se ocuparon las referencias de Internet especificadas en la última página de este documento.

de riesgos a nivel de entidad o de proceso, la planificación de auditorías y recursos, seguimiento de hallazgos, reportes de gastos y de tiempo, control de calidad, y cuenta con la flexibilidad de un módulo de reportes. Todos estos módulos están completamente integrados y los datos fluyen de uno a otro automáticamente.

Cobit Advisor: Utilizado por auditores internos y consultores, Cobit Advisor registra los resultados de los trabajos de auditoría interna y permite la comunicación de los resultados que tienen materialidad. Dos de sus más destacables características son: Generación de Riesgos, Controles, Observaciones, y Notas de Revisión y una amplia cantidad de informes, en una cantidad de formatos disponibles, fácilmente generables.

1.6 CONTENIDO CAPITULAR

CAPITULO I: Anteproyecto

1.1. Antecedentes

1.1.1. Orígenes del estudio

1.2. Importancia de la investigación

1.2.1. Importancia

1.2.2. Justificación

1.3. Proyección de desarrollo empresarial

1.4. Definición del tema

1.4.1. Definición del tema

1.4.2. Objetivos

1.4.2.1. General

1.4.2.2. Específicos

1.4.3. Alcances

1.4.4. Limitaciones

1.4.5. Delimitaciones

1.5. Marco teórico

1.5.1. Marco histórico

1.5.1.1. Evolución histórica de las computadoras.

- 1.5.1.2. Breve historia de la auditoría.
- 1.5.2. Marco conceptual
 - 1.5.2.1. Definición de auditoría.
 - 1.5.2.2. Clasificación de la auditoría.
 - 1.5.2.3. Generalidades de la auditoría
 - 1.5.2.4. Alcance de la auditoría interna
 - 1.5.2.5. Proceso general de realización de auditorías.
 - 1.5.2.6. Normas de la auditoría
 - 1.5.2.6.1. La auditoría de sistemas y las normas internacionales
 - 1.5.2.7. Generalidades de la auditoría informática.
 - 1.5.2.8. La auditoría asistida por computadora
 - 1.5.2.9. Importancia de las herramientas en el desarrollo de sistemas
 - 1.5.2.10. El Ciclo de Vida de Desarrollo de Sistemas de Información
 - 1.5.2.10.1. Definición de un modelo de ciclo de vida
 - 1.5.2.11. La Programación Orientada a Objetos y el UML
 - 1.5.2.11.1. Conceptos de Programación Orientado a Objetos
 - 1.5.2.11.2. El Lenguaje Unificado de Modelado (UML)
- 1.5.3. Marco experimental
- 1.6. Esquema de solución
- 1.7. Metodología de investigación y desarrollo
 - 1.7.1. Investigación de campo
 - 1.7.2. Técnicas de recopilación de datos
 - 1.7.3. Aplicación del Ciclo de Vida de Desarrollo de Software
 - 1.7.4. Resumen explicativo de actividades por etapa
 - 1.7.4.1. Actividades de la etapa 1:
 - 1.7.4.2. Actividades de la etapa 2
- 1.8. Cronograma de actividades
- 1.9. Presupuesto
 - 1.9.1. Costos para el desarrollo del sistema
 - 1.9.2. Costos de mano de obra
 - 1.9.3. Costos de Licencias
 - 1.9.4. Otros gastos

CAPITULO II: Análisis de Encuestas y Modelado de la Base de Datos

2.1 Investigación de Hechos

2.1.1 Elaboración de encuestas

- 2.1.1.1 Objetivo de la encuesta
- 2.1.1.2 Investigación de la población
- 2.1.1.3 Cálculo del tamaño de la muestra
- 2.1.1.4 Formato de las encuestas
- 2.1.1.5 Análisis de las encuestas.

2.2 Diseño de la Base de Datos

2.2.1 Especificación del problema a resolver

2.2.2 Modelo Conceptual o de Entidad – Relación

- 2.2.2.1 Identificación de los conjuntos de entidades
- 2.2.2.2 Identificación de los conjuntos de interrelaciones
- 2.2.2.3 Primer Diagrama Entidad – Relación
- 2.2.2.4 Identificación de Atributos
- 2.2.2.5 Selección de claves candidatas
- 2.2.2.6 Segundo diagrama Entidad – Relación

2.2.3 Modelo Lógico o Relacional

- 2.2.3.1 Establecimiento de relaciones
- 2.2.3.2 Aplicación de formas normales
 - 2.2.3.2.1 Primera Forma Normal (1FN)
 - 2.2.3.2.2 Segunda Forma Normal (2FN)

2.2.4 Segunda aplicación del modelo Entidad – Relación

- 2.2.4.1 Estableciendo nuevas interrelaciones
- 2.2.4.2 Tercer diagrama Entidad – Relación
- 2.2.4.3 Base de datos – Diagrama Esquemático

2.3 Construcción de la Base de Datos

2.3.1 Diccionario de Datos

2.3.2 Elaboración de la Base de Datos en MySQL

- 2.3.2.1 Nombre de la Base de Datos

- 2.3.2.2 Especificación de las tablas
- 2.3.2.3 Creación de relaciones
- 2.3.2.4 Guión de creación de la Base de Datos

CAPITULO III: Modelado y Construcción de la Aplicación

3.1 Modelado de la aplicación

3.1.1 Especificación del funcionamiento de la aplicación

3.1.2 Determinación de clases

3.1.3 Diagramación UML

3.1.3.1 Diagramas de casos de uso

3.1.3.2 Diagramas de clases

3.1.3.3 Diagramas de secuencia

3.1.3.4 Diagramas de actividad

3.1.3.5 Diagramas de colaboración

3.1.4 Construcción de la Aplicación

CAPITULO IV: Documentación del Sistema

4.1 Manual de usuario

4.2 Manual de mantenimiento

4.3 Manual de instalación

Glosario

Conclusiones

Recomendaciones

Anexos

1. Formato de encuesta de investigación de hechos
2. Artículo Técnico: Experiencia profesional del proyecto

1.7 METODOLOGÍA DE INVESTIGACIÓN Y DESARROLLO

1.7.1 Investigación de campo

Se llevarán a cabo revisiones de la metodología empleada en el trabajo de un auditor de sistemas, con el fin de obtener información de procedimientos y documentación que constituirán la base de conocimientos del sistema informático a diseñar.

1.7.2 Técnicas de recopilación de datos

Las técnicas de recopilación de datos que se ocuparán son:

Entrevistas:

Con la guía de entrevistas se buscará obtener la información requerida, a través del Jefe del Departamento de Supervisión y Control de la Superintendencia de Valores, ya que en dicho departamento se realizan las auditorías de sistemas. Entre los puntos principales que contendrá el cuestionario estará la metodología que aplican para realizar las auditorías, los recursos que utilizan y su opinión sobre el software para auditorías.

Encuestas:

Con la encuesta que se pretende pasar a los auditores de sistemas de la Superintendencia de Valores, se quiere adquirir datos interesantes sobre la auditoría y opiniones acerca del uso de un software de control de auditorías de sistemas como herramienta de apoyo.

La técnica de la encuesta es muy utilizada por los investigadores. Permite recopilar información en el lugar de los hechos, mediante opiniones de personas involucradas de alguna manera con la problemática que se investiga.

Una de sus características principales es hacer las mismas preguntas a todos los sujetos involucrados en la muestra de una población determinada, por medio de la información interrogativa escrita.

La información recopilada por medio de esta técnica se utiliza para hacer un análisis de la problemática en estudio e identificar y cuantificar la magnitud del fenómeno de investigación.

Se puede aplicar personalmente, por correo o por medios de comunicación electrónica.

Básicamente se puede hablar de dos tipos de preguntas en un cuestionario; preguntas cerradas y abiertas. Las preguntas cerradas pueden clasificarse como sigue:

- Preguntas cerradas: Ofrecen dos opciones a la persona encuestada del tipo Sí – No.
- Preguntas cerradas con tres opciones.
- Preguntas cerradas categorizadas: En este tipo de preguntas se obtienen respuestas más objetivas del encuestado, porque se le presenta más de una opción, para que pueda responder con más facilidad.

En las preguntas abiertas, este tipo de preguntas es muy fácil para formularla; pero se necesita un plan específico para tabular; de lo contrario se dificulta mucho su tabulación.

Existe un tipo que combina preguntas abiertas y cerradas, llamado *pregunta en cascada*. Al usar este tipo de preguntas es recomendable formular las preguntas cerradas con solo dos alternativas, para no tener dificultad en la tabulación de los datos.

La encuesta contiene los siguientes elementos:

- Nombre de la institución.
- Título del tema que se quiere observar.
- Objetivo de la encuesta.
- A quién va dirigida.
- Agradecimiento al entrevistado por la información proporcionada.
- Instrucción de las preguntas del cuestionario.
- Contenido de las preguntas.

Una vez recolectada la información, se procederá a la tabulación de los datos. Con el resultado se espera respaldar la justificación del software.

1.7.3 Aplicación del Ciclo de Vida de Desarrollo de Software

Se utilizará el modelo en cascada del Ciclo de Vida de Desarrollo de Software. Para mayor información de las etapas que se seguirán en esta metodología, consultar el apartado 5.2.10.1 “Definición de un modelo de Ciclo de Vida” en este mismo documento. En este modelo se harán un máximo de tres retrocesos por cada etapa del ciclo.

1.7.4 Resumen explicativo de actividades por etapa

1.7.4.1 Actividades de la etapa 1

Esta etapa puede denominarse: *Determinación de requerimientos y diseño del software* y las actividades a llevar a cabo en la misma serán las siguientes:

- ***Investigación de Campo.*** Realización de encuestas y entrevistas al jefe del Departamento de Supervisión y Control de la Superintendencia de Valores y a profesionales de la auditoría de sistemas; utilizando cuestionarios como instrumentos para recolectar información con el fin de determinar requerimientos para el diseño.
- ***Recopilación de información.*** Las fuentes de información a consultar son las siguientes: recursos bibliográficos, consultas profesionales e Internet.
- ***Investigación permanente.*** Investigación en diferentes fuentes durante el período de desarrollo del proyecto.¹⁰
- ***Redacción y edición de documento.*** Se elaborará el documento de acuerdo a la recopilación de información, encuesta e investigación realizada.
- ***Creación de interfaz de usuario.*** Desarrollo del entorno con el cual el usuario interactuará. En esta el usuario encontrará disponible las diferentes opciones que proporcionará el sistema. La interfaz de usuario será construida usando el

¹⁰ Consultar la sección de Bibliografía al final del documento.

lenguaje PHP 4, debido a su calidad de software libre y a su naturaleza orientada a las aplicaciones Internet.

- **Diseño y creación de base de datos.** La base de datos será construida y diseñada en MySQL 5.0, debido a su licencia libre de costo y a su adaptabilidad con el software PHP.
- **Diseño de Algoritmos.** Se describirán los procesos principales que manejará el software.
- **Construcción del software.** Se desarrollarán las funciones principales de los módulos que compondrán el software

1.7.4.2 Actividades de la etapa 2:

Esta segunda etapa puede denominarse: *Construcción del software y documentación externa*. Las actividades que se realizarán son:

- **Construcción del software.** Se completará la programación de los módulos que componen el software.
- **Creación de manuales de usuario.** Se crearán los manuales necesarios para la documentación externa del software: De usuario, de mantenimiento y de instalación.
- **Validación de resultados.** Consistirá en la realización de pruebas al software en base a datos previamente definidos, considerando a la Superintendencia como empresa piloto, con el fin de evaluar la funcionalidad de éste y validar los resultados provistos por el software.

1.8 CRONOGRAMA DE ACTIVIDADES

1.9 PRESUPUESTO

1.9.1 Costos para el desarrollo del sistema

Lunes a Viernes = 4hrs diarias

Sábado y Domingo = 16hrs

Meses de Duración del Proyecto = 8

Por lo tanto:

1 Semana = 36 hrs.

Horas x mes = 36 hrs. x 4 semanas

Horas x mes = 144 hrs.

Tiempo Total = 1152 hrs. al año

Salario devengado tomando como base un salario de:

Salario x mes = \$ 1000

Salario / días laborales = \$ 1000 / 30 días = \$ 33.33

Salario / horas laborales = \$ 33.33 / 8horas = \$ 4.17

Salario x hora = \$4.17 la hora

1.9.2 Costos de mano de obra

Precio = Tiempo total x Salario por hora

Precio = 1152 hrs. x \$ 4.17

Precio = \$ 4,803.84 al año

1.9.3 Costos de Licencias

En este rubro se encuentran los costos de las licencias del software a utilizar como herramienta de desarrollo.

Herramienta	Costo
PHP	<i>Gratuito</i>
MySQL 5.0	<i>Gratuito</i>
Windows XP	\$165
Total Costos	\$165

Tabla 2: Costos de Licencias Software a ocupar en el proyecto

1.9.4 Otros gastos y contabilización total

Rubro en el que se agrupan todos aquellos gastos adicionales en los que se incurre y que no pertenecen a los dos grupos anteriores.

Rubros	Cantidad	Precio Unitario	Costos mensuales (\$)	Costos proyecto(8 meses) x
Computadora portátil	1	\$ 1,300		
Electricidad			\$ 30.00	\$ 210.00
Internet			\$ 15.00	\$ 35.00
Papelería :				
Fotocopias	500	0.03		\$15.00
Compra de CDs	7	0.57		\$ 3.99
Compra de Diskette				
Papel Bond	4			\$ 10.00
Empastados y Anillados				\$ 25.00
Transporte				\$ 60.00
Impresión				\$ 40.00
Otros				\$ 30.00
Total:				\$ 1,728.00

Tabla 3: Otros gastos del proyecto

Total Otros Gastos = \$ 1,728.00

Precio Total = Costo Mano de Obra + Costo de Licencias + Total Otros Gastos

Precio Total = \$ 4,803.84 + \$ + \$435 + \$ 1,728.00

Precio Total = \$ 6,966.84

CAPITULO II

ANALISIS DE ENCUESTAS

Y

DISEÑO DE LA BASE DE DATOS

2.1 INVESTIGACIÓN PRELIMINAR

2.1.1 Elaboración de encuestas

2.1.1.1 Objetivo de la encuesta

El objetivo de la encuesta realizada fue determinar la necesidad de un software especializado que sirva de apoyo para el trabajo del auditor de sistemas.

2.1.1.2 Investigación de la población

En esta etapa se realizó una investigación acerca del total de auditores de sistemas legalmente registrados a nivel nacional en las instituciones: ASPROC¹¹ y FESIARA¹², las cuales se encargan de llevar un registro actualizado de los profesionales en computación legalmente autorizados a nivel nacional. Sin embargo, se nos manifestó que dichas instituciones no poseen dicho dato; por lo que se optó por la investigación vía telefónica a las instituciones que se muestran en la tabla 4, junto con los respectivos datos obtenidos.

NOMBRE DE LA INSTITUCIÓN	NUMERO DE AUDITORES DE SISTEMAS
Ministerio de Hacienda	19
Corte de Cuentas de El Salvador	20
Banco Central de Reserva de El Salvador	5
Superintendencia de Valores de El Salvador	1
Bolsa de Valores de El Salvador	1
TOTAL	46

Tabla 4: Número de auditores de sistemas por institución

2.1.1.3 Cálculo del tamaño de la muestra

Una vez obtenido el total de la población, se procedió a realizar el cálculo para obtener el tamaño de la muestra, la cual se obtuvo por medio de la siguiente fórmula:

¹¹ ASPROC: Asociación Salvadoreña de Profesionales en Computación.

¹² FESIARA: Federación Salvadoreña de Ingenieros, Arquitectos y Ramas Afines.

$$n = \frac{Z^2 pq N}{(N - 1)E^2 + Z^2 pq}$$

Donde:

n = Tamaño de la muestra.

Z = 1.96 → Coeficiente de confianza. Representa el área de la curva normal para un 95% de probabilidad que las respuestas de la encuesta no sean altamente diferentes.

p = 0.5 → Proporción de ocurrencia. Este valor se asume a 0.5 por defecto para estudios con un tamaño de población pequeño como la de nuestro estudio.

q = 0.5 → Proporción de no ocurrencia: Resulta de la diferencia 1 – p.

N = 46 → Tamaño de la población = 46

E = 0.09 → Error muestral: Es un valor bajo debido a que se espera que los resultados no varíen mucho y el tamaño de la población es pequeño.

Sustituyendo los valores en la ecuación y desarrollando:

$$n = \frac{(1.96)^2 (0.5) (0.5) (46)}{(46 - 1)(0.09)^2 + (1.96)^2 (0.5) (0.5)}$$

$$n = 33$$

El resultado obtenido en el tamaño de la muestra es: 33. Es decir, se encuestarán a 33 elementos de la población. Debido a que no se tuvo acceso a las instituciones que se dedican propiamente a la realización de auditorías de sistemas, se optó por seleccionar una muestra entre un grupo de profesionales en el área de la informática (ingenieros en sistemas).

2.1.1.4 Formato de las encuestas

El formato de la encuesta que se llevó a cabo puede revisarse en el anexo 1 del presente documento.

2.1.1.5 Análisis de las encuestas

1 Que es la auditoría de sistemas.

Que es para usted auditoría de sistemas?	Resultados por ítem
Revisión y evaluación de la eficacia y la eficiencia de un sistema.	12
Evaluación de objetivos, metas y políticas informáticas	2
Verificación de estándares y licencias.	5
Gestión controladora de técnicas y procedimientos informáticos.	8
Evaluación de casos de uso del sistema.	1
No sabe/ No responde	3

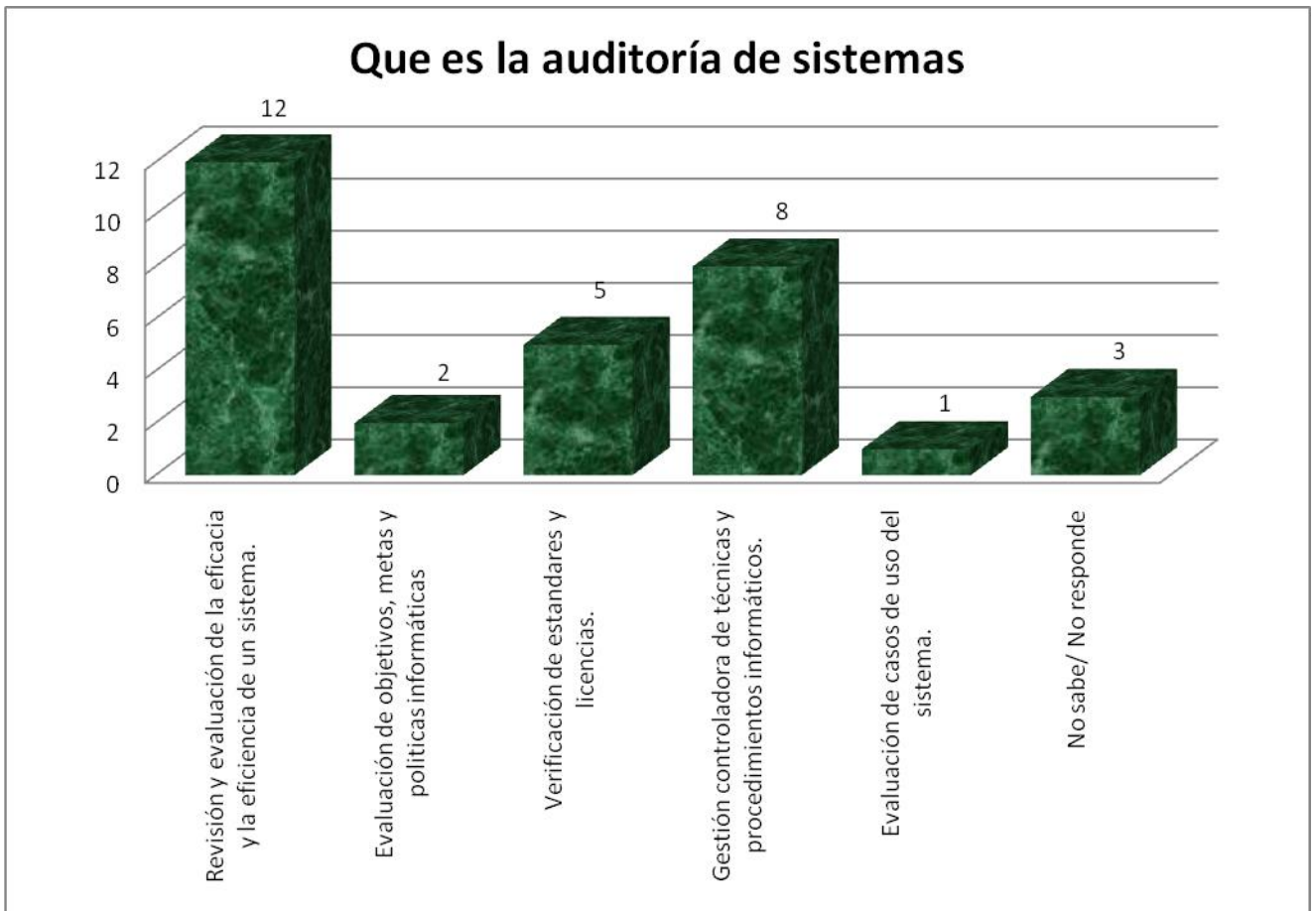


Figura 8: Pregunta 3 de la encuesta

La opinión mayoritaria entiende que la auditoría de sistemas es la verificación del buen funcionamiento de un sistema informático. No obstante, el concepto se queda corto, por lo que el resto de opiniones lo complementan más.

2 Conocimiento sobre las funciones del auditor de sistema.

¿Conoce cuáles son las funciones de un auditor de sistemas?, si las conoce, mencione algunas.	Resultados por ítem
Verificación, control y diagnóstico de datos.	8
Evaluación de procesos y políticas informáticas.	9
Revisión y pruebas de la seguridad y funcionamiento del sistema.	4
Desarrollo de informes y programas de auditoría en sistemas.	3
Revisión de documentación del sistema.	1
No sabe/ No responde	7

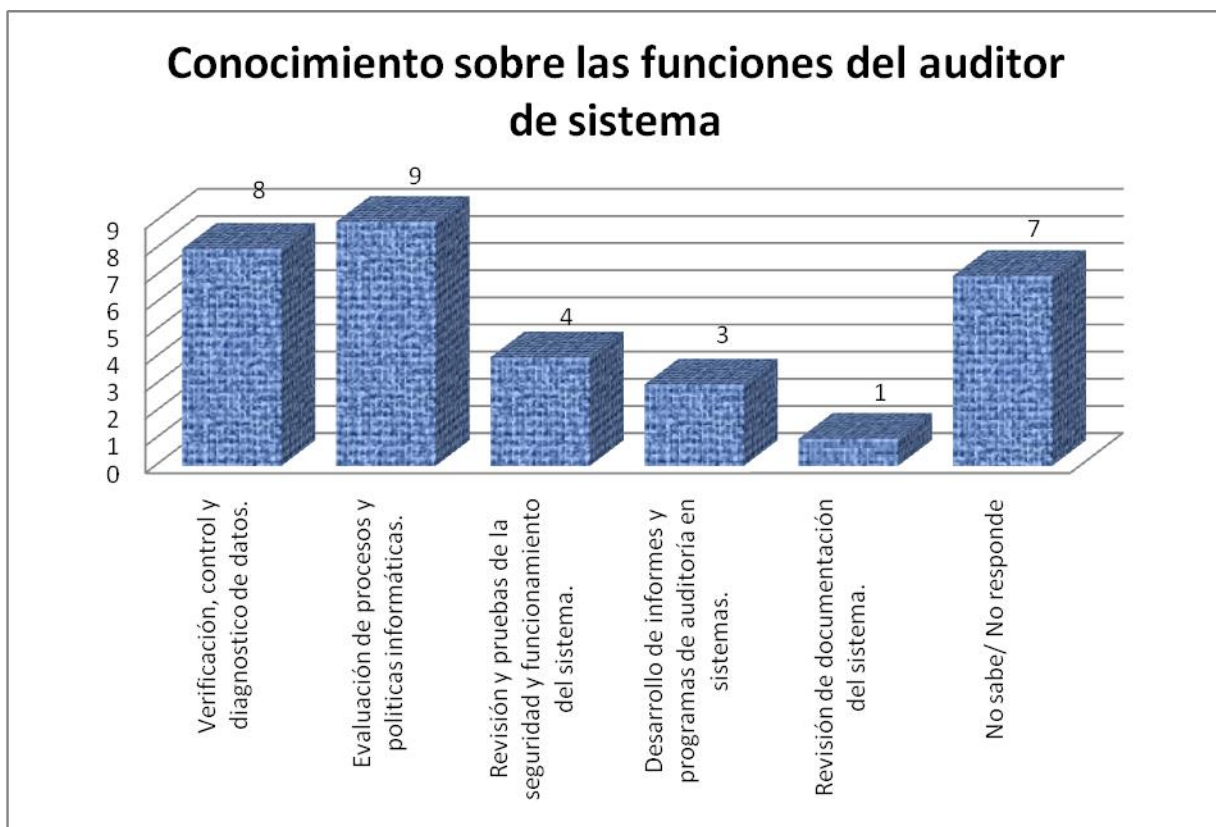


Figura 9: Pregunta 4 de la encuesta

Tomando como base los resultados de la pregunta anterior, los encuestados entienden que las funciones del auditor de sistemas son las siguientes: evaluar y controlar las actividades informáticas de las entidades. Asimismo señalan las pruebas y el diagnóstico como parte importante de esta labor.

3 Aspectos que se evalúan en la A. S.

Marque cuales aspectos conoce se evalúan cuando lleva a cabo una auditoria de sistemas	Resultados por ítem
Instalaciones	20
Respaldo de datos	22
Validación de datos	18
Acceso a personal no autorizado	19
Interfaz del sistema	11
Licencias	27
Contraseñas	24
Tipo de software	14
Protección contra desastres	19
Capacitación del personal	8
Soporte técnico	9
Captura de datos	15
Sistemas Operativos	12
Documentación del sistema	26
Presupuesto	7
Otros	0
No conozco	1

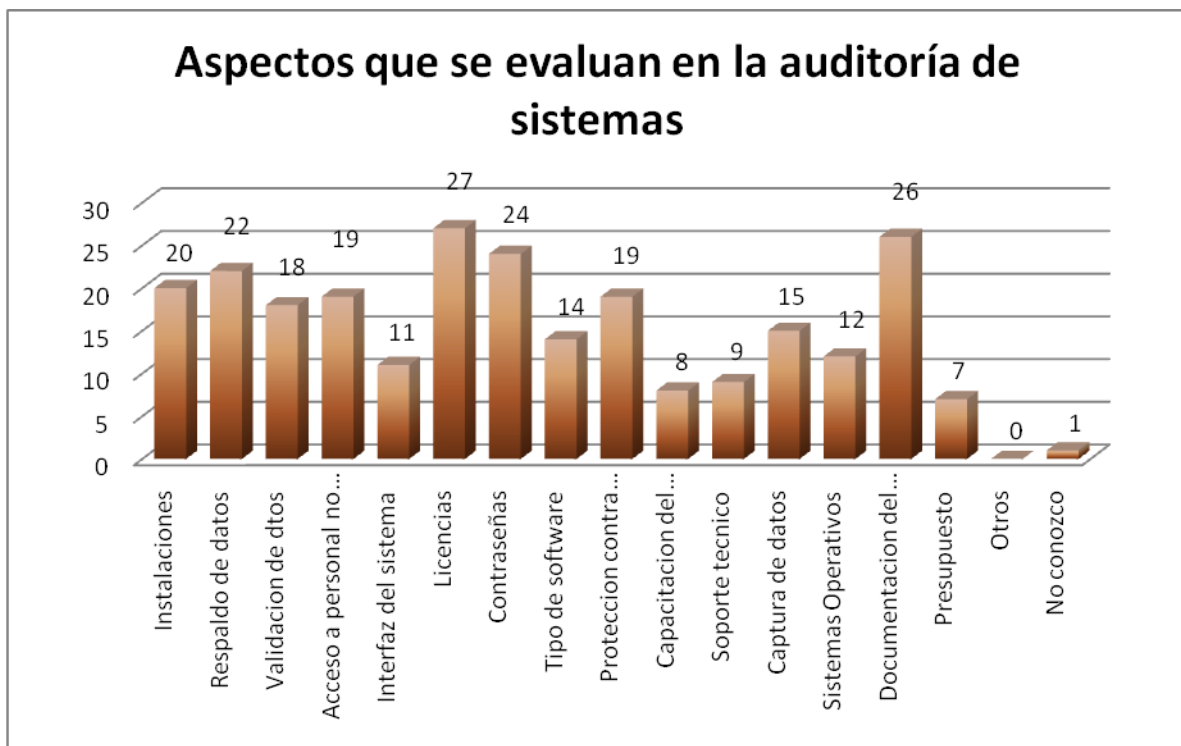


Figura 10: Pregunta 5 de la encuesta

Los resultados de esta pregunta están casi balanceados, así podemos ver que en lo referente a los aspectos a evaluar en una auditoría de sistemas, las opciones más elegidas tienen que ver con aspectos de hardware, seguridad, código del software (validaciones) y documentación.

4 Conocimiento de la documentación que se elabora al finalizar la auditoría

¿Conoce qué tipo de documentación elabora al final de una auditoría?	Resultados por ítem
Acta	13
Informe de hallazgos	26
Informe gerencial	11
Nota con hallazgos	4
No conozco	4

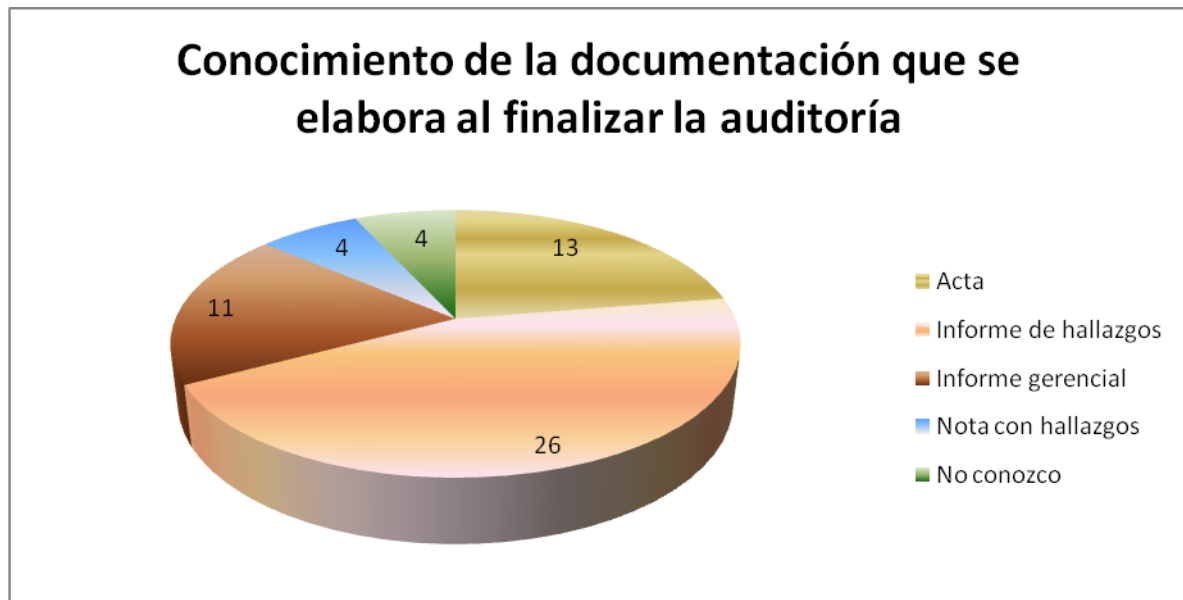


Figura 11: Pregunta 6 de la encuesta

El informe de hallazgos fue la opción con mayor puntaje en cuanto a la documentación que se elabora al finalizar una auditoría de sistemas. Los encuestados coinciden que este es el informe más importante. El software a desarrollar generará este informe acompañado del acta, de forma automática.

5 Tiempo promedio que se tarda una persona en elaborar dichos documentos sin software especializado.

En promedio ¿cuanto estima usted que se tarda una persona en elaborar dichos documentos, sin software especializado?	Resultados por ítem
Menos de 1 hora	0
Entre 1 y 2 horas	3
Entre 2 y 3 horas	1
Entre 3 y 4 horas	1
Entre 4 y 5 horas	3
Entre 5 y 6 horas	2
Entre 6 y 7 horas	1
Más de 7 horas	18
No sabe/No contesta	3



Figura 12: Pregunta 7 de la encuesta

Como puede observarse en la figura 12, el tiempo de elaboración de informes es relativamente alto, por lo que el software a desarrollar será una herramienta de apoyo para minimizar ese tiempo.

6 Necesidad de utilizar formularios para realizar las auditorías de sistemas.

Si usted trabajara como auditor de sistemas estima necesario utilizar alguna clase de formularios para realizar dicha tarea.	Resultados por ítem
si	29
no	1
No sabe/No contesta	2

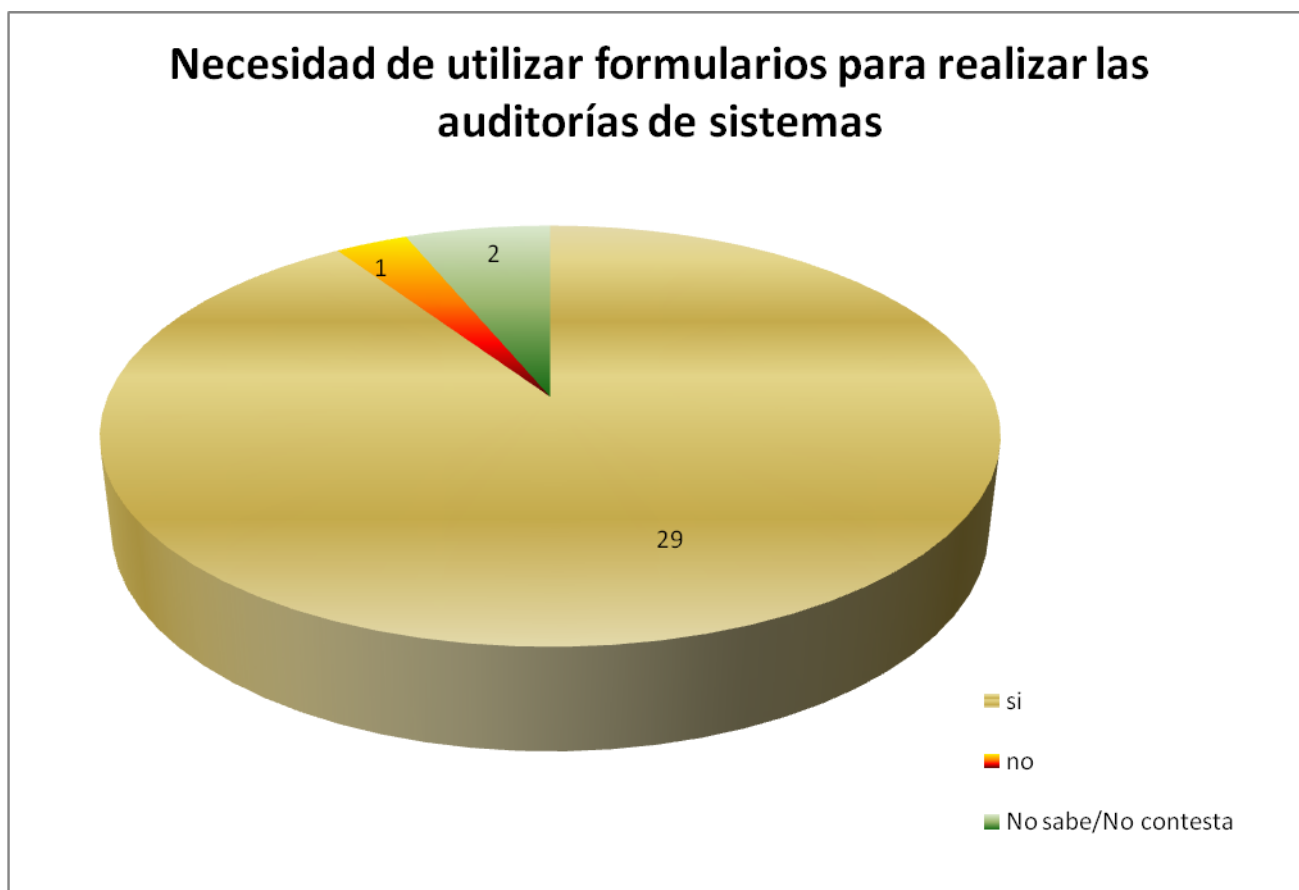


Figura 13: Pregunta 8 de la encuesta

La gran mayoría de encuestados estima necesarios los formularios para la auditoría de sistemas. Es por ello que en el software a desarrollar, estos formularios serán implementados bajo HTML.

7 Experiencia en el uso de la computadora.

¿Cuanta experiencia tiene usando computadora?	Resultados por ítem
Nada	
Poca	1
Normal	7
Mucha	24



Figura 14: Pregunta 9 de la encuesta

Esta pregunta tiene por objetivo confirmar la experiencia de los encuestados con la herramienta tecnológica que se ocupará para el presente proyecto.

8 Experiencia con el uso de Internet.

¿Está usted familiarizado con el uso de Internet?	Resultados por ítem
Si	32
No	0
No sabe/No contesta	0

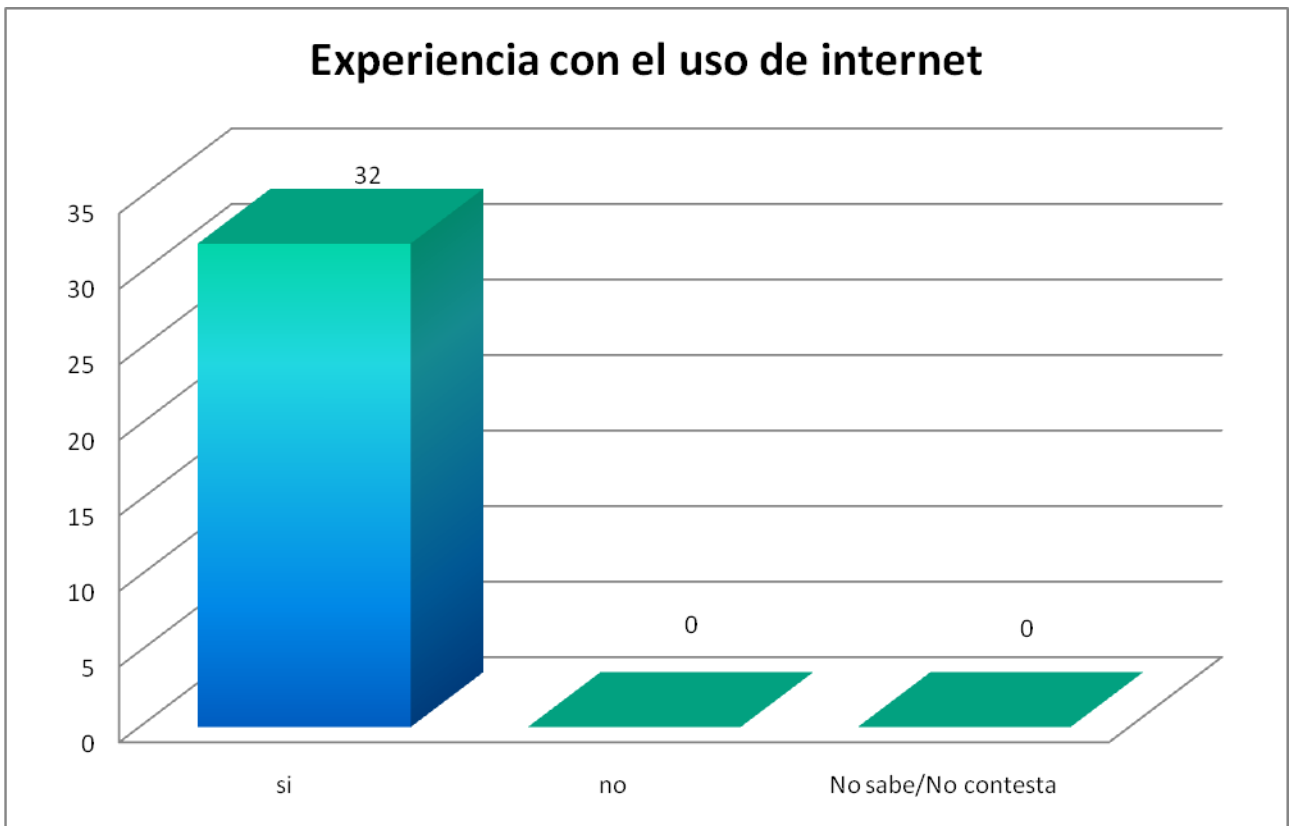


Figura 15: Pregunta 10 de la encuesta

Así como en la figura 15, los puntos de vista de los encuestados demuestran la gran aceptación de Internet como herramienta de trabajo. Por ello, el software a construir será diseñado para ambiente World Wide Web.

9 Formas de utilización de Internet.

¿Para qué utiliza el Internet?	Resultados por ítem
Consultar sus papeles de trabajo	21
Investigar información de la empresa a auditar	11
Sitio Web de mi empresa	18
Contacto con las empresas fiscalizadas	8
Otro uso	8
No sabe/No contesta	4

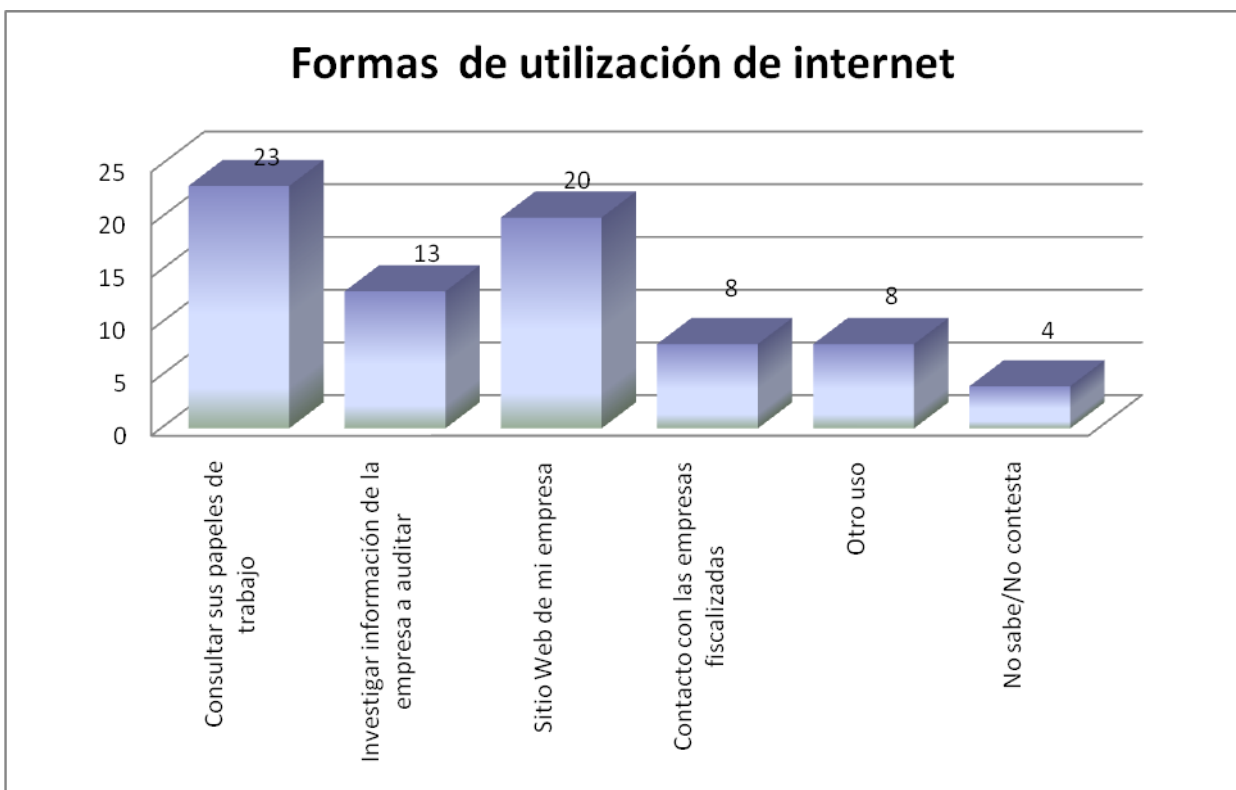


Figura 16: Pregunta 11 de la encuesta

Estas opiniones nos confirman que el uso de la Web combinado con las consultas de papeles de trabajo es aceptado mayoritariamente. Tal es la idea de la aplicación a diseñar.

10 Conocimiento sobre software para asistir la auditoría de sistemas.

¿Conoce usted algún software para asistir la auditoría de sistemas?	Resultados por ítem
Si	12
No	19
No sabe/No contesta	1

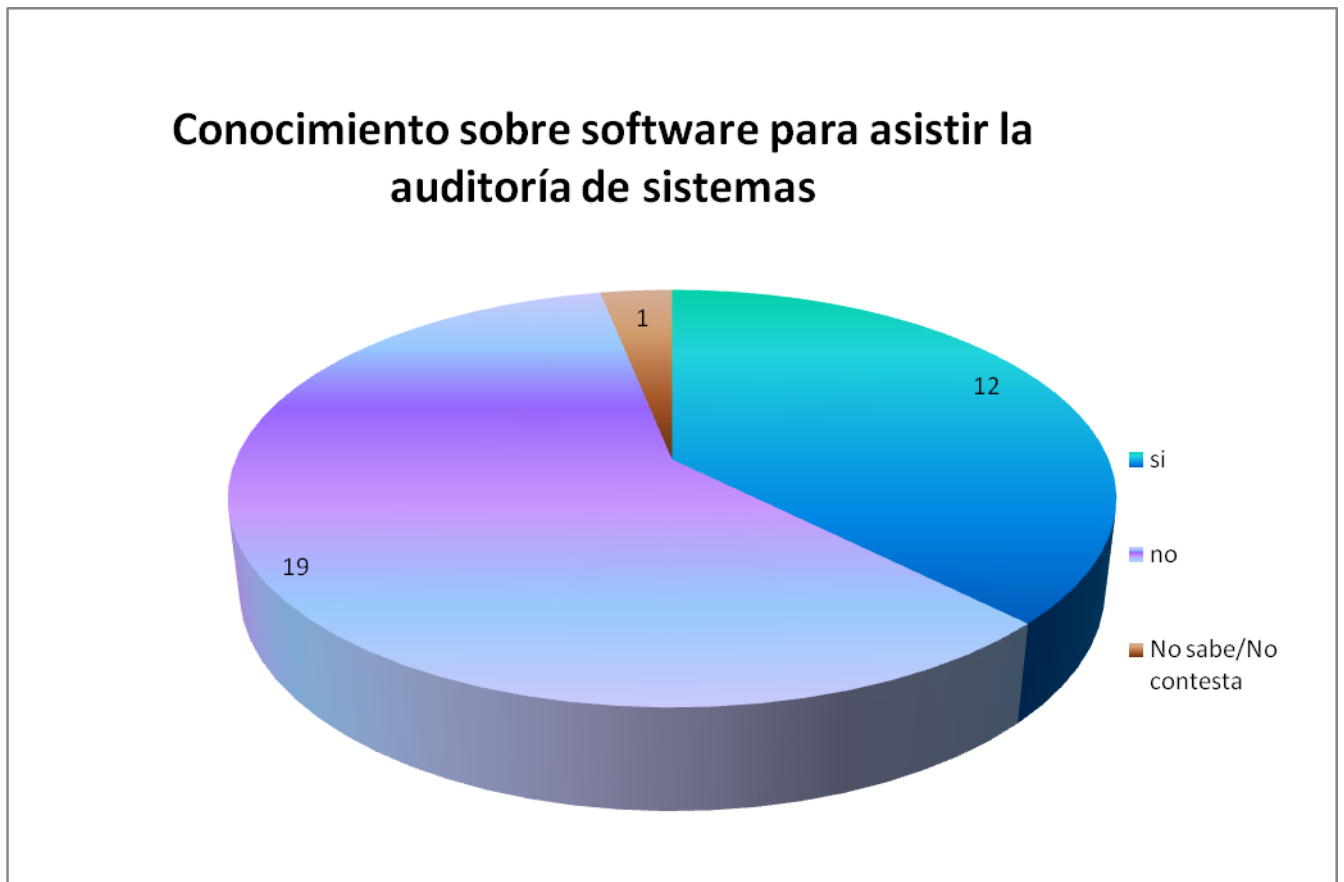


Figura 17: Pregunta 12 de la encuesta

Según las respuestas de la figura 17, el software de auditoría de sistemas ha tenido poco auge en nuestro medio. Este proyecto busca suplir esa deficiencia.

11 ¿Cual software es el que conoce para asistir la auditoría?

Si la respuesta es si, seleccione cuales conoce.	Resultados por ítem
Auto Audit	1
Era	0
Cobit Advisor	2
Word	7
Excel	8
Otros	2
No sabe/No contesta	19

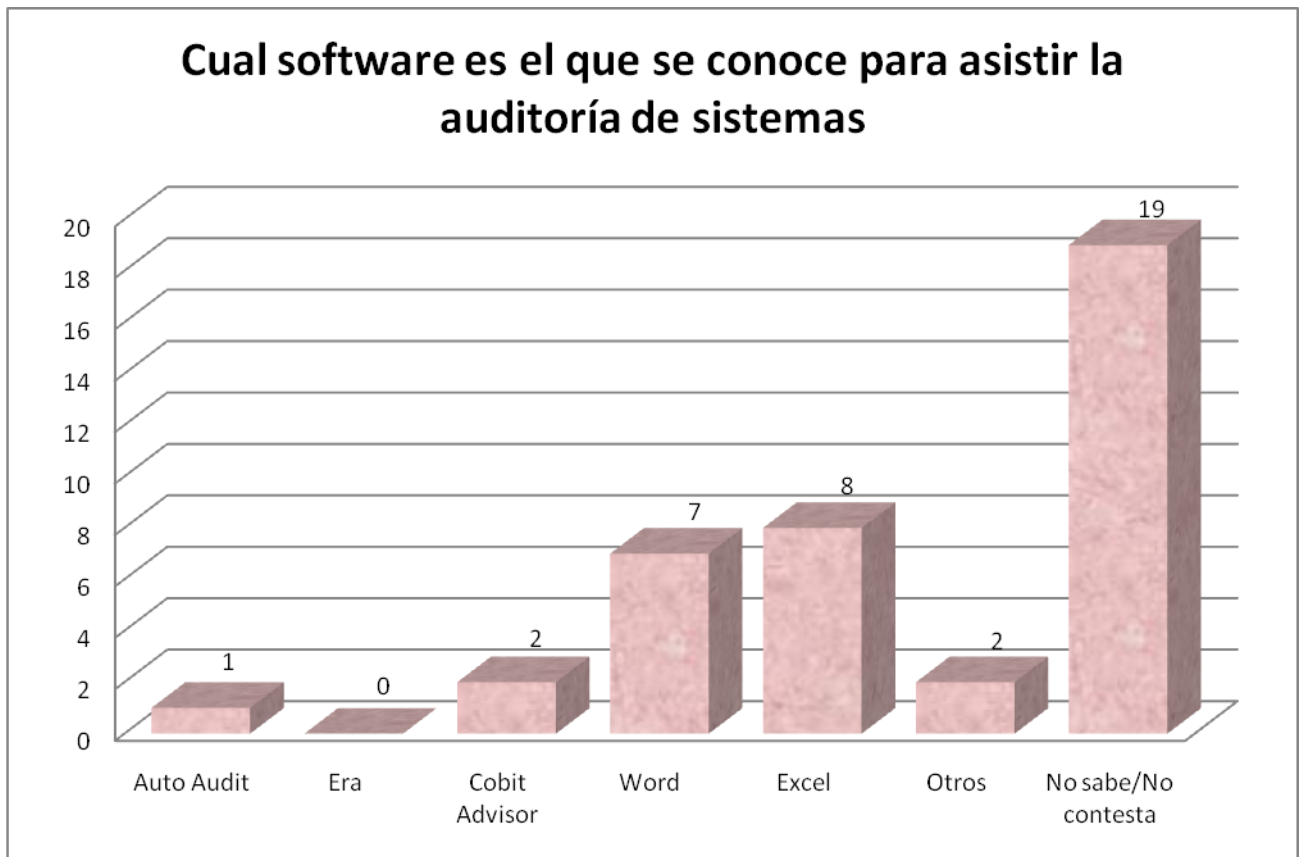


Figura 18: Pregunta 13 de la encuesta

Aunque existe muy buen software plenamente dedicado a la auditoría de sistemas; estos no han tenido mucha distribución en nuestro país, como lo muestran los resultados de esta encuesta, donde las herramientas ofimáticas tienen mayor presencia.

12 Limitantes para que el auditor de sistema utilice software especializado.

¿Cuál cree que es el mayor impedimento para que un auditor de sistemas use software especializado en su labor?	Resultados por ítem
Alto costo	24
Falta de experiencia en el manejo de computadoras	7
Falta de computadora	5
Otros	4



Figura 19: Pregunta 14 de la encuesta

Los resultados de esta pregunta muestran que el alto costo del software especializado es la mayor limitante para la popularización de su uso. El sistema a construir reducirá costos pues se basa en herramientas de libre distribución.

13 Características que debe tener un software para que sirva de apoyo a un auditor de sistemas.

¿Que características considera usted que debe tener un software para que le sirva de apoyo a un usuario auditor de sistemas?	Resultados por ítem
Facilidad de uso	24
Agilidad en la obtención de resultados	24
Bajo costo	18
Interfaz amigable	23
Otros	4

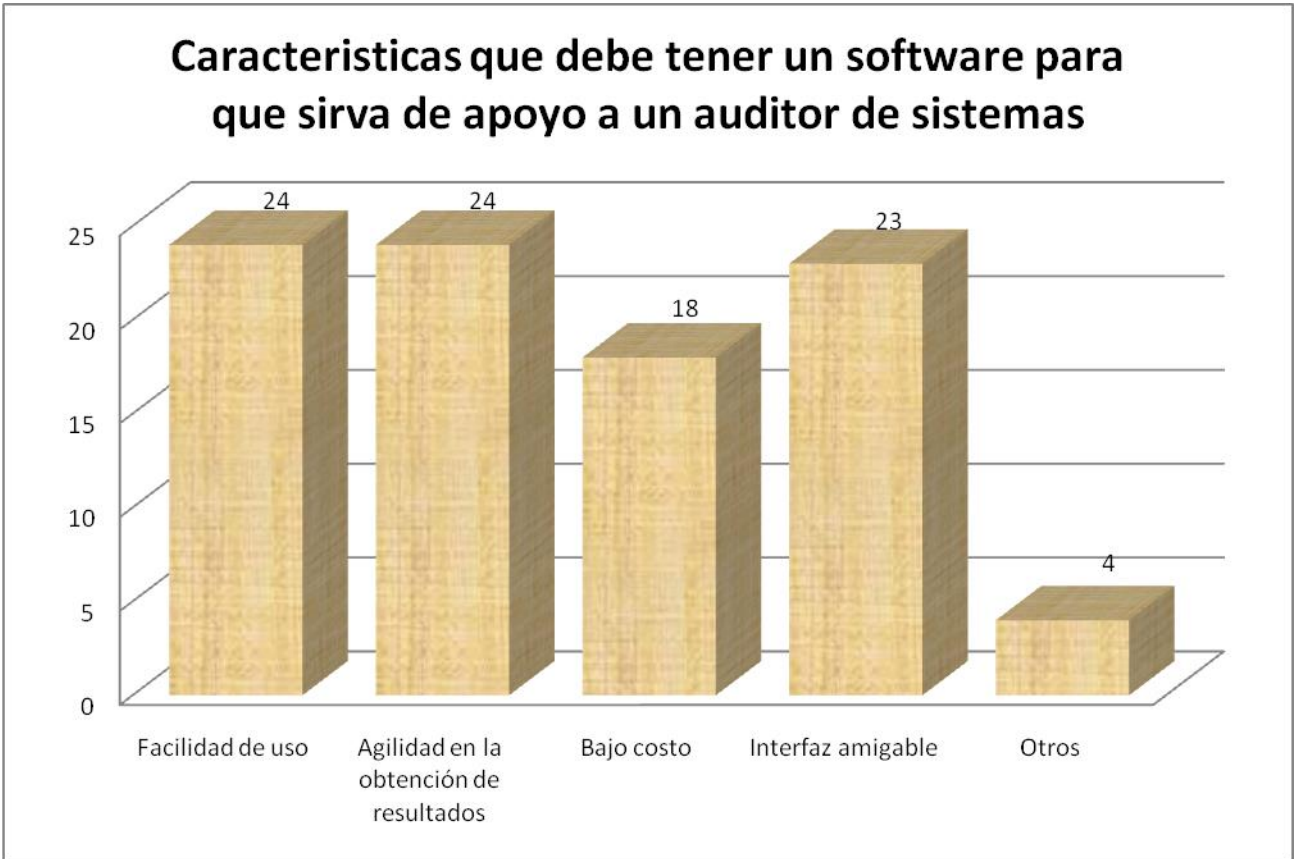


Figura 20: Pregunta 15 de la encuesta

Las opiniones de este punto están parejas en cuanto a la facilidad de uso, la eficiencia y el diseño de interfaz, seguidos de cerca por el bajo costo. Éstas serán metas a lograr en el sistema a desarrollar.

2.2 DISEÑO DE LA BASE DE DATOS

2.2.1 Especificación del problema a resolver

El sistema a desarrollar contendrá cuatro programas de auditoría. Dichos programas serán los siguientes:

- Programa para Almacenes Generales de Depósitos.
- Programa para Bolsas de Valores.
- Programa para Casas Corredoras de Bolsa.
- Programa para Centrales de Depósito de Valores.

Cada programa de auditoría estará conformado por varias cédulas de auditoría. Éstas serán:

- Cédula sumaria de planeación
- Cédula sumaria de hardware.
- Cédula sumaria de software.
- Cédula sumaria de políticas y procedimientos.
- Cédula sumaria de seguridad.
- Cédula sumaria de hallazgos.
- Cédula sumaria de marcas.

Las cinco primeras cédulas contendrán un conjunto de preguntas relativas al área a evaluar. Cada una de ellas incluirá un conjunto único de preguntas.

Para obtener una descripción sobre el contenido que se manejará en cada una de las cédulas anteriores revisar el apartado

3.1.1 Especificación del funcionamiento de la aplicación.

El sistema a desarrollar se delimitará dentro de las auditorías de cuatro tipos de instituciones: Almacenes Generales de Depósito, Casas Corredoras de Bolsa, Centrales de Depósito de Valores y Bolsas de Valores. Cada una de éstas tendrá uno o varios papeles de auditoría. De estas instituciones solo existirá una Central de Depósito y una Bolsa de Valores. Podrá haber más de un Almacén General de Depósito y más de una Casa Corredora de Bolsa.

No todas las preguntas de las cédulas aplicarán a todas las instituciones, ya que habrá, por ejemplo, preguntas de la cédula de hardware que apliquen sólo a los Almacenes, otras de la misma cédula que apliquen sólo para la Central de Depósito de Valores. En otras palabras, todas las instituciones usarán las mismas cédulas; pero las preguntas de dichas cédulas se aplicarán dependiendo del tipo de institución. Asimismo, existirán preguntas que apliquen para todas las entidades. Cada cédula clasificará sus preguntas por categorías. Una categoría albergará varias preguntas y cada pregunta sólo podrá incluirse en una categoría específica.

El sistema podrá ser utilizado por varios usuarios, cada uno de los cuales poseerá un rol específico. Cada rol determinará diferentes privilegios de operación del sistema. Los tres roles permitidos serán:

- *Administrador*: Este rol tendrá la facultad de agregar, consultar, modificar registros de la base de datos de instituciones y además tendrá todos los privilegios del rol *operativo*. Sólo podrá existir un usuario con este rol dentro del sistema.
- *Operador*: Este rol permitirá crear, consultar, modificar y guardar trabajos de auditoría (llamados también *papeles de trabajo*). También tendrá permiso para consultar datos de las instituciones agregadas al sistema.
- *Consulta*: Con este rol, el usuario podrá acceder a la información de los papeles de trabajo, sin la posibilidad de llevar a cabo ninguna modificación en la misma.

El usuario, al momento de hacer su trabajo de auditoría podrá seleccionar aquellas preguntas de las cédulas que considere necesarias para el programa que va a desarrollar durante la realización de auditoría y desechar aquellas que no apliquen al caso. A esta selección personalizada se le llamará plantilla.

2.2.2 Modelo Conceptual o de Entidad - Relación

El modelado conceptual o de entidad – relación consiste en identificar y seleccionar los conjuntos de entidades del problema a diseñar y en buscar las *interrelaciones* existentes entre dichas entidades. Para diagramar dicho modelo se emplean los diagramas Entidad – Relación.

2.2.2.1 Identificación de los conjuntos de entidades

Las entidades que se reconocen en un inicio, a partir de la especificación anterior son:

- Papel de trabajo
- Institución
- Cédula
- Usuario
- Plantilla

2.2.2.2 Identificación de los conjuntos de interrelaciones

- Cada papel de trabajo se refiere a una sola institución.
- Cada institución puede tener varios papeles de trabajo.
- Cada papel de trabajo contiene más de una cédula.
- Cada cedula de auditoría puede ser usada en varios papeles de trabajo en una misma institución.
- Varias cédulas de auditoría pueden ser utilizadas por varios usuarios en sus papeles de trabajo.
- Cada papel de trabajo puede ser utilizado por varios usuarios (aunque de diferente forma, dependiendo del rol de los mismos).

2.2.2.3 Primer Diagrama Entidad – Relación

La figura 21 esquematiza los conjuntos de interrelaciones descritos en el apartado anterior. En ella se muestran las entidades representadas por rectángulos y los conjuntos de interrelaciones representadas por rombos. Las relaciones de varios a varios se representan por las letras N y M en los rombos. Las relaciones de uno a varios se representan por 1 y N en las relaciones de los rombos.

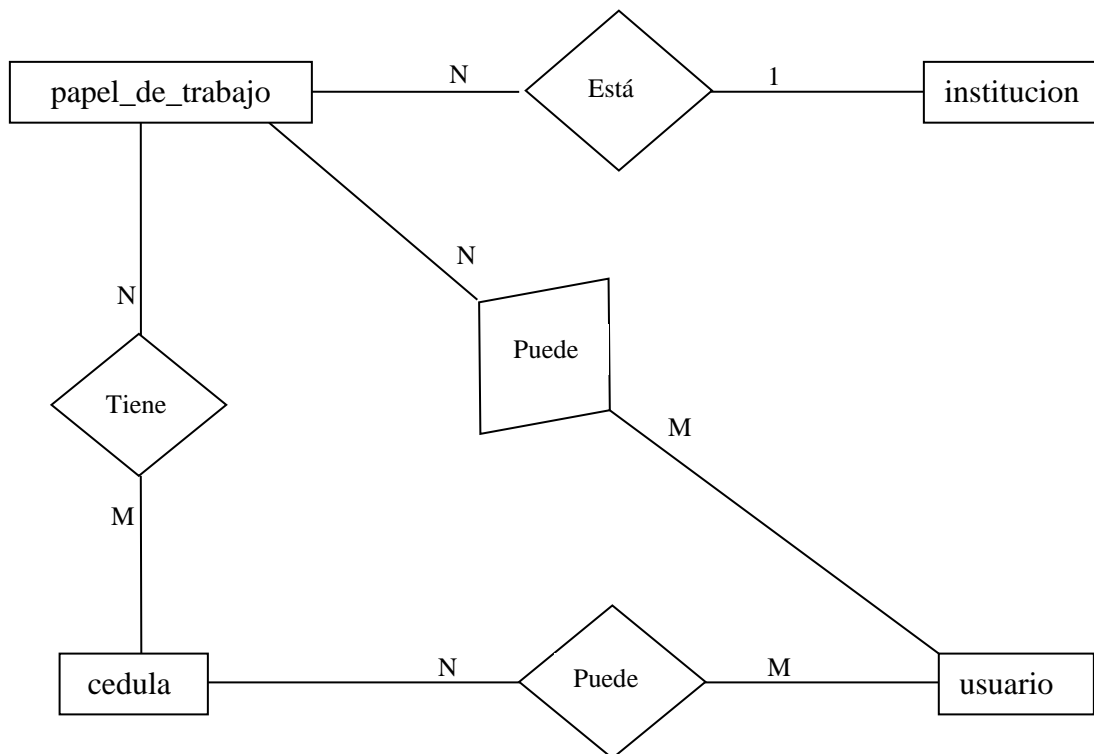


Figura 21: Primer diagrama entidad – relación

La entidad *plantilla* se tratará y agregará luego de haber mejorado y normalizado el diseño anterior.

2.2.2.4 Identificación de atributos

Los atributos para cada una de las entidades consideradas en el diagrama 8 son los siguientes:

- Entidad *papel de trabajo*: Nombre del papel, nombre del usuario, fecha de creación, fecha de actualización, nombre de la institución, descripción.

- Entidad *institución*: Nombre de la institución, tipo de institución, dirección, teléfono, nombre del contacto.
- Entidad *cédula*: Nombre de la cédula, preguntas de la cédula, respuestas de la cédula.
- Entidad *usuario*: Nombre del usuario, cargo del usuario, tipo de usuario, identificador, contraseña.

2.2.2.5 Selección de claves candidatas

En esta etapa seleccionaremos aquel atributo o conjunto de atributos que identifican de manera única a cada entidad. De acuerdo a esto, tenemos las siguientes claves principales:

- Entidad *papel de trabajo*: Nombre del papel, descripción.
- Entidad *institución*: Nombre de la institución, dirección, teléfono.
- Entidad *cédula*: Nombre de la cédula.
- Entidad *usuario*: identificador, contraseña

2.2.2.6 Segundo diagrama Entidad – Relación

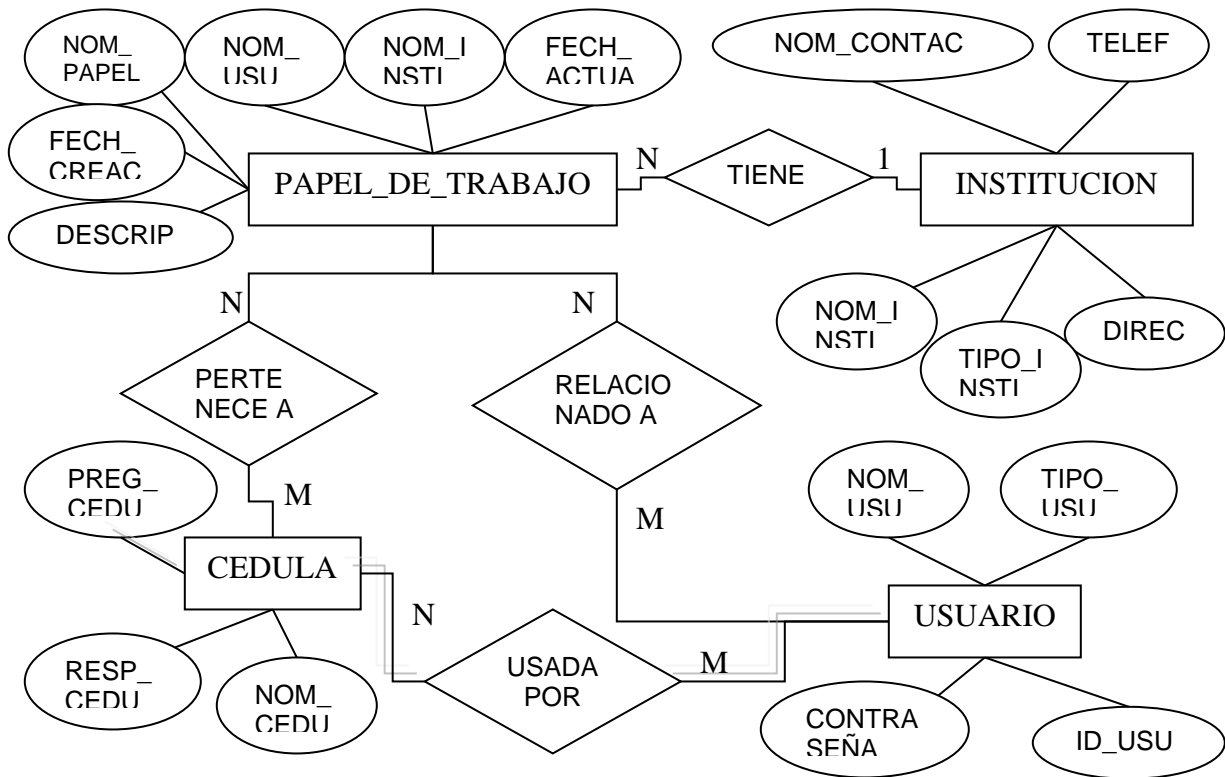


Figura 22: Segundo Diagrama Entidad - Relación

La figura 22 representa el diagrama entidad – relación de la figura 21; pero con los atributos adicionados a cada entidad. Todavía no se ha destacado ningún campo clave en las entidades, ya que se hará hasta que tengamos la base normalizada.

2.2.3 Modelo Lógico o Relacional

En este apartado se continuará con el diseño de la base, aplicando el llamado modelo lógico o relacional a los resultados obtenidos en el modelo entidad – relación de la sección anterior.

El modelo relacional agrupa a los conjuntos de entidades en unidades llamadas *tuplas* o *relaciones*, las cuales contienen los atributos de cada entidad. Esta representación es facilita el proceso de *normalización* de la base de datos. Dicho proceso se revisa en el apartado **2.2.3.2 Aplicación de Formas Normales**.

2.2.3.1 Establecimiento de relaciones

En esta etapa se hará uso de la notación relacional para expresar las entidades anteriormente definidas en el modelo entidad – relación.

Las relaciones del modelo son las siguientes:

Papel de trabajo (nombre del papel, nombre del usuario, fecha de creación, fecha de actualización, nombre de la institución, descripción).

Institución (nombre de institución, tipo de institución, dirección, teléfono, nombre del contacto).

Cédula (nombre de la cédula, preguntas de la cédula, respuestas de la cédula).

Usuario (Nombre del usuario, tipo de usuario, identificador, contraseña).

2.2.3.2 Aplicación de formas normales

El proceso de normalización consiste en verificar el cumplimiento de las características de integridad de la base de datos, a fin de evitar las redundancias e inconsistencias. En caso de no cumplir dichas características, se aplica un proceso llamado *aplicación de formas normales*.

Por *integridad de la base de datos* se entiende aquella condición en la cual se mantienen coherentes las interrelaciones entre registros y las claves principales se mantienen sin repetirse.

Por *redundancia en la base de datos* se entiende aquella situación en la cual los mismos datos se almacenan más de una vez en la misma base de datos.

La *inconsistencia* en la base de datos se da como producto de la redundancia, y se pone de manifiesto cuando en una tabla de la base de datos se lleva a cabo una actualización; pero dicho cambio no se refleja en las demás tablas que de una u otra forma se relacionan con la primera tabla.

La inconsistencia y la redundancia de datos son siempre características a eliminar en las bases de datos.

2.2.3.2.1 Primera Forma Normal (1FN)

En esta primera forma normal se buscará *atomizar* los registros de la base de datos, es decir, que no hayan dos registros diferentes con la misma información en un campo. Para ello, se dividirán algunos atributos en otros más pequeños.

En la relación *Papel de trabajo* el atributo *nombre del usuario* puede desglosarse en los siguientes atributos:

- Primer apellido del usuario.
- Segundo apellido del usuario.
- Primer nombre del usuario.
- Segundo nombre del usuario.

El atributo *fecha de creación* puede dividirse en tres atributos más:

- Día de creación.
- Mes de creación.
- Año de creación

El atributo *fecha de actualización* puede dividirse en tres atributos más:

- Día de actualización
- Mes de modificación
- Año de modificación

En la relación *Institución* el atributo *nombre del contacto* puede desglosarse como sigue:

- Primer apellido del contacto.
- Segundo apellido del contacto.
- Primer nombre del contacto.
- Segundo nombre del contacto.

En la relación *Institución* el atributo *dirección* puede descomponerse en los siguientes atributos:

- Calle
- Avenida
- Colonia
- Número de local

En la relación *Usuario*, el atributo *nombre del usuario* puede descomponerse como sigue:

- Primer apellido del usuario.
- Segundo apellido del usuario.
- Primer nombre del usuario.
- Segundo nombre del usuario.

Las relaciones finales después de aplicar esta forma normal quedan:

- *Papel de trabajo* (nombre del papel, día de creación, mes de creación, año de creación día de actualización, mes de actualización, año de actualización, primer apellido del usuario, segundo apellido del usuario, primer nombre del usuario, segundo nombre del usuario, nombre de la institución, descripción del papel).
- *Institución* (nombre de institución, tipo de institución, calle, avenida, colonia, número de local, teléfono, primero apellido del contacto, segundo apellido del contacto, primer nombre del contacto, segundo nombre del contacto).
- *Cédula* (nombre de la cédula, preguntas de la cédula, respuestas de la cédula)
- *Usuario* (primer apellido del usuario, segundo apellido del usuario, primer nombre del usuario, segundo nombre del usuario, cargo del usuario, tipo de usuario, identificador, contraseña)

No todas estas divisiones de atributos se mantendrán en la base de datos final. Esto se debe a que MySQL maneja tipos de datos capaces de incluir en un solo campo

tres atributos diferentes, por ejemplo, los atributos *día de creación*, *mes de creación*, y *año de creación*, pueden incluirse en un solo campo denominado *fecha de creación*, del tipo de dato MySQL *DATE*, el cual ha sido concebido para almacenar datos que representen fechas.

2.2.3.2.2 Segunda Forma Normal (2FN)

Con esta forma normal se busca crear una correspondencia uno a uno entre cada atributo no clave y los atributos clave de cada entidad. Esto se logra examinando las dependencias funcionales de cada una de las relaciones y creando nuevas relaciones a partir de aquellos atributos que no verifiquen dichas dependencias. A la situación de cumplimiento de una dependencia se representa así:

Atributo clave → Atributo no clave

El incumplimiento de una dependencia funcional se escribe así:

Atributo clave → | Atributo no clave

Las dependencias funcionales para la relación *Papel de trabajo* son:

Nombre del papel → Fecha de creación

Nombre del papel → Fecha de actualización

Nombre del papel → Nombre de la Institución

Nombre del papel → | Nombre del usuario (que abarca apellidos y nombres del usuario)

Nombre del papel → Descripción del papel

Extrayendo la nueva relación:

Usuario (primer apellido del usuario, segundo apellido del usuario, primer nombre del usuario, segundo nombre del usuario).

Como ya se tenía una relación anterior *Usuario* se le agregan los atributos: *cargo del usuario*, *tipo de usuario*, *identificador* y *contraseña* a esta nueva relación, para tener un solo esquema *Usuario*:

Usuario (primer apellido del usuario, segundo apellido del usuario, primer nombre del usuario, segundo nombre del usuario, cargo del usuario, tipo de usuario, identificador, contraseña).

Determinando claves primarias para las nuevas relaciones encontradas:

Para la relación *Usuario* se tienen las claves candidatas: *identificador* y *contraseña*. Sin embargo, debido a que estos atributos se almacenarán como cadenas de caracteres se prefiere hacer uso de un atributo numérico artificial. Dicho atributo se denominará *id_usu*. La relación queda:

Usuario (*id_usu*, primer apellido del usuario, segundo apellido del usuario, primer nombre del usuario, segundo nombre del usuario, cargo del usuario, tipo de usuario, identificador, contraseña).

Se combinan además en uno solo los atributos *primer apellido del usuario* y *segundo apellido del usuario* en uno solo *ape_usu*. Se hace otro tanto con *primer nombre del usuario* y *segundo nombre del usuario* para obtener el atributo *nom_usu*. Se cambian además los nombres de los atributos *cargo del usuario* a *cargo_usu*, *identificador* a *login*, *contraseña* a *passwd* y *tipo de usuario* a *id_rol*. La relación *Usuario* se escribe ahora así:

<i>Usuario</i> (<i>id_usu</i> , <i>ape_usu</i> , <i>nom_usu</i> , <i>cargo_usu</i> , <i>login</i> , <i>passwd</i> , <i>id_rol</i>)
--

En la relación *Papel de trabajo* se crea también un atributo artificial *id_papel* para utilizarlo como clave principal, ya que las claves candidatas *Nombre del papel* y *descripción del papel* son cadenas de caracteres con tamaño relativamente extenso como para cumplir el papel de llaves primarias. Se combinan los atributos *día de creación*, *mes de creación* y *año de creación* en *fecha de creación*; *día de actualización*, *mes de actualización* y *año de actualización* en *fecha de actualización*. Finalmente, se cambian los nombres de los atributos: *Nombre del papel* a *nom_papel*, *fecha de creación* a *fecha_creac*, *fecha de actualización* a *fecha_actua* y

descripción del papel a descrip. Con todo lo anterior, la relación *Papel de trabajo* se escribe así:

<i>Papel_de_trabajo</i> (id_papel, nom_papel, fech_creac, fech_actua, descrip)
--

Las dependencias funcionales para la relación *Institución* son:

Nombre de la institución → Tipo de institución

Nombre de la institución → Dirección

Nombre de la institución → Teléfono

Nombre de la institución → | Nombre del contacto

Extrayendo una nueva relación *Contacto* y agregándole los atributos de interés:

Contacto (primer apellido del contacto, segundo apellido del contacto, primer nombre del contacto, segundo nombre del contacto, teléfono del contacto, correo electrónico del contacto).

Determinando clave primaria para la nueva relación encontrada:

Para la relación *Contacto* se tienen como claves candidatas: *teléfono del contacto* y *correo electrónico del contacto*. Sin embargo, por ser éstos atributos que se almacenarán en memoria como cadenas de caracteres, se prefirió crear un atributo artificial *id_contact*. La relación queda:

Contacto (id_contact, primer apellido del contacto, segundo apellido del contacto, primer nombre del contacto, segundo nombre del contacto, teléfono del contacto, correo electrónico del contacto).

Como se hizo con la relación *Usuario*, se agrupan los atributos *primer apellido del contacto* y *segundo apellido del contacto* en *ape_contact* y los atributos *primer nombre del contacto* y *segundo nombre del contacto* en *nom_contac*. Se cambian además los nombres de los atributos: *teléfono del contacto* a *tel_contact* y *correo electrónico del contacto* a *ema_contact*. La relación *Contacto* se escribe ahora así:

Contacto (id_contact, ape_contact, nom_contact, tel_contact, ema_contact)

En cuanto a la relación *Institución* se le agrega un atributo artificial *id_insti*, debido a que la clave candidata *Nombre de la institución* se almacenará como cadena de caracteres relativamente extensa como para ser llave primaria. Seguidamente se modifican los nombres de los atributos: *Nombre de la institución* a *nom_insti*, *tipo de institución* a *tip_insti*, *dirección* a *dir_insti* y *teléfono* a *tel_insti*. Finalmente, se le agrega un campo *estado* que indicará si la institución tiene calidad de activa o inactiva. De momento, la relación *Institución* queda como sigue:

Institución (id_insti, nom_insti, tip_insti, tel_insti, dir_insti, estado)

Las dependencias funcionales para la relación *Cédula* son:

Nombre de la cédula → | Pregunta de la cédula

Nombre de la cédula → | Respuesta de la cedula

Extrayendo las nuevas relaciones y añadiendo atributos de interés:

Pregunta (texto de la pregunta)

Respuesta (texto de la respuesta)

Determinando claves primarias: En la primera relación, el atributo texto de la pregunta es clave candidata; pero su tipo de datos (texto / memo) es demasiado grande, por lo que se descarta como llave primaria. En la segunda relación no existen claves candidatas. Por tanto, se crea un atributo artificial para cada una, quedando, por el momento, las relaciones así:

Pregunta (id_preg, texto de la pregunta)

Respuesta (id_resp, texto de la respuesta)

En cuanto a la relación *Cédula*, esta será una de las relaciones llamadas *de catálogo*, es decir, que tendrán datos por defecto y que no podrán ser actualizadas

desde la capa de aplicación. Su utilidad será brindar datos a las entidades que se interrelacionen con ellas. Dichas relaciones son:

- *Categoría*: Contendrá los nombres de las categorías de preguntas en las cuales se dividen las cédulas de auditoría.
- *Cedula*: Almacena los nombres de las siete cédulas de auditoría que el sistema maneja. (Ver **especificación del problema, apartado 2.2.1**)
- *Tipo_institucion*: Contiene los cuatro tipos de entidades con los que el programa trabajará. (Ver **especificación del problema, apartado 2.2.1**).
- *Rol*: Almacena los tres tipos de roles de usuario que se han considerado en el problema. (Ver **especificación del problema, apartado 2.2.1**).

<i>Categoría</i> (id_categoria, nom_categoria)
--

<i>Cedula</i> (id_ced, nom_ced)

<i>Tipo_institucion</i> (id_tipoinsti, nom_insti)

<i>Rol</i> (id_rol, tipo_rol)

Las cuatro relaciones anteriores no surgieron como producto de un análisis de dependencias funcionales; pero se consideraron debido a su utilidad práctica tanto en el diseño de la base como para la codificación de la capa de aplicación.

Se contempla ahora la última parte de la especificación del problema. La creación de plantillas de preguntas personalizadas.

Se determina una nueva entidad: *Plantilla* de la cual interesan los atributos: Nombre de la plantilla, qué usuario crea dicha plantilla y un identificador único para dicha plantilla. Este identificador será clave primaria de la relación, la cual se declara así:

Plantilla (id_plantilla, nom_plantilla, id_usu)

Existen dos documentos necesarios antes de comenzar una auditoría:

La cédula de planeación que permite ordenar los pasos que se llevarán a cabo antes, durante y después de la investigación:

La nota de requerimientos, que el auditor envía a una persona responsable en la empresa, y que representa el conjunto de documentos en los cuales se apoyará la investigación.

La cédula de planeación de cada papel de trabajo consta de un listado ordenado de contenidos divididos en procedimientos, los cuales a su vez, se separan en subprocedimientos. Podemos ver aquí una interrelación en cascada uno a varios entre cuatro relaciones diferentes: papel de trabajo, contenido de la planeación, procedimientos y subprocedimientos.

Dichas relaciones y sus atributos de importancia de cada una, se listan a continuación:

contenido_plani (id_contenido, id_papel, orden)

procedimiento_plani (id_procedimiento, id_contenido, descripción, orden)

subprocedimiento_plani (id_subprocedimiento, id_procedimiento, orden)

Respecto a la nota de requerimientos, se dice que a cada papel de trabajo en auditoría que vaya a realizarse, le corresponden normalmente, varios requerimientos, de ahí una relación de uno a varios, teniendo la siguiente relación para una entidad requerimiento.

requerimiento (id_requer, nombre_requer, descrip_requer, entrega_requer, fecha_recibido, id_papel)

Referente a los informes que se elaboran al finalizar una auditoría, tenemos tres: El acta de auditoría, el informe y la nota de hallazgos.

Del informe de hallazgos interesan los siguientes atributos:

Encabezado del informe.

Áreas o aspectos a evaluar

Condición, criterio, causa y efecto de cada área

Revisores y elaboradores del informe

De los anteriores atributos se dice que:

Cada papel de trabajo tiene aparejado un encabezado de informe.

Cada papel de trabajo puede tener varios revisores de informe.

Cada papel de trabajo puede tener varios elaboradores de informe.

Cada pregunta de un papel de trabajo contiene un área asociada, así como también una condición, un criterio, una causa y un efecto.

De esa forma, condición, criterio, causa y efecto quedan como atributos de la entidad `preg_papel`; el encabezado será un atributo de la entidad `papel_de_trabajo` y los revisores y elaboradores del informe serán dos entidades separadas con interrelación uno a varios con `papel_de_trabajo`.

Del acta de auditoría interesan los siguientes atributos:

Encabezado del acta

Áreas a evaluar

Observación y comentario de cada área

Firmantes del acta

De los anteriores atributos se dice que:

Cada papel de trabajo tiene un encabezado de acta

Cada pregunta de un papel de trabajo contiene un área asociada, así como también una observación y un comentario

Cada papel de trabajo puede tener varios firmantes de acta. Un firmante puede aparecer en el acta de varios papeles de trabajo.

De esa forma, encabezado de acta queda como atributo de papel de trabajo mientras que área, observación y comentario son atributos de `preg_papel`. Finalmente, los firmantes de acta constituirán una entidad aparte en interrelación varios a varios con `papel_de_trabajo`

De la nota de auditoría interesan los siguientes atributos:

Encabezado y pie de la nota

Asunto, introducción y conclusión de la nota

Áreas de auditoría evaluadas, junto con su condición, criterio, causa y efecto.

De los anteriores atributos se dice que:

A cada papel de trabajo le corresponde un encabezado, pie, asunto, introducción y conclusión de nota de auditoría.

Cada pregunta de un papel de trabajo contiene un área asociada, así como también una condición, criterio, causa y efecto.

De esa forma, el encabezado, pie, asunto, introducción y conclusión de la nota serán atributos de la entidad *papel_de_trabajo*; mientras que área, condición, criterio, causa y efecto son atributos de la entidad *preg_papel*.

Otra capacidad del sistema será la de registrar en orden cronológico, las actividades de un usuarios específico dentro del sistema. A este registro se le llamará *bitácora* y está conformado por varios tipos de actividades a las cuales el usuario accede cada vez que usa el sistema (por ejemplo, abrir un papel de trabajo, crear una plantilla o ingresar un nuevo usuario al sistema). Se tiene dos entidades nuevas: *bitácora*, relacionada de varios a uno con la entidad *usuario* y *actividad*, relacionada de uno a varios con *bitácora*.

Con esto termina la aplicación de la segunda forma normal. Se han obtenido nuevas relaciones; sin embargo, no se han definido los enlaces entre entidades interrelacionadas. A continuación se trabajan dichas uniones recordando simultáneamente las condiciones que ya se habían establecido en la sección **2.2.2.2 Identificación de los conjuntos de interrelaciones**, y otras que han surgido debido a las nuevas relaciones creadas. La tabla 5 hace referencia a cada una de estas condiciones numeradas y resume todas las interrelaciones de la base.

1. Una institución puede tener varios papeles de trabajo. Un papel de trabajo pertenece a una sola institución.
2. Una institución puede tener varios contactos. Un contacto representa a una sola institución.
3. Una institución pertenece a un solo tipo de entidad. Un tipo de entidad puede albergar a varias instituciones.
4. Un papel de trabajo contiene varias preguntas. Una pregunta puede presentarse en varios papeles de trabajo.
5. Un papel de trabajo puede ser creado por un solo usuario del sistema; pero puede ser consultado o modificado por más de un usuario (dependiendo de los privilegios que el rol de usuario le conceda). Dichos usuarios a su vez puede crear varios papeles de trabajo en el sistema.
6. Una pregunta de auditoría puede ser parte de una sola categoría en específico, mientras que cada categoría contiene dentro de sí varias preguntas de auditoría.
7. Una categoría puede estar incluida en una sola cédula. Una cédula puede dividirse en varias categorías.
8. Una pregunta de auditoría puede aplicar a más de un tipo de institución. Un tipo de institución pueden tener varias preguntas de auditoría.
9. Un usuario del sistema puede poseer un solo rol (aunque puede tener varios roles si posee más de un login). A un rol en específico (de los tres que existen) pueden pertenecer varios usuarios.
10. Una pregunta de auditoría puede ser parte de varias plantillas de usuario. A su vez, una plantilla puede contener más de una pregunta del listado general.
11. Un papel de trabajo puede estar basado en una sola plantilla. Una plantilla puede alimentar de preguntas a varios papeles de auditoría.
12. Un usuario puede tener varias plantillas creadas, mientras que varias plantillas pueden ser propiedad de un solo usuario.
13. Cada usuario tiene varias entradas de su bitácora. Mientras que varias entradas de la bitácora corresponden a un usuario en específico.
14. Una actividad realizada por un usuario puede asimismo estar en las bitácoras de varios usuarios.

15. Una persona que funge como revisor de informes de auditoría de papeles de trabajo, puede tener más de un informe del cual haya sido revisor.
16. Una persona que se desempeña como elaborador de informes de auditoría de papeles de trabajo, puede tener más de un informe realizado por él.
17. Un firmante de actas de auditoría puede tener varias actas firmadas. Asimismo un acta generalmente tiene más de un firmante.
18. Cada papel de trabajo tiene varios requerimientos. Cada requerimiento pertenece a un papel de trabajo específico.
19. Cada papel de trabajo contiene varias entradas de contenido en la cédula de planeación.
20. Cada contenido en la cédula de planeación se divide en varios procedimientos.
21. Cada procedimiento en la cédula de planeación se divide en varios subprocedimiento.

Número de condición	Tipo de interrelación	Relación origen ¹³	Relación destino ¹⁴	Relación de enlace	Atributos de enlace (Claves foráneas)
1	Uno a varios	Institución	Papel_de_trabajo		Id_insti
2	Uno a varios	Institución	Contacto		Id_insti
3	Uno a varios	Tipo_institucion	Institución		Id_tipoinsti
4	Varios a varios	Papel_de_trabajo, Pregunta	Preg_papel	Preg_papel	Id_papel, id_preg
5	Varios a varios	Papel_de_trabajo, Usuario	Usu_papel	Usu_papel	Id_papel, id_usu
6	Uno a varios	Categoría	Pregunta		id_categoria
7	Uno a varios	Cedula	Categoría		id_ced
8	Varios a varios	Pregunta, Tipo_institucion	Preg_tipo	Preg_tipo	Id_tipoinsti, id_preg
9	Uno a varios	Rol	Usuario		Id_rol
10	Varios a	Pregunta, Plantilla	Preg_plantilla	Preg_plantilla	Id_preg,

¹³ La relación origen corresponde a la parte “uno” en una interrelación “uno a varios” y a las relaciones enlazadas por una tabla de enlace en una interrelación “varios a varios”.

¹⁴ La relación destino corresponde a la parte “varios” en una interrelación “uno a varios” y a la tabla de enlace en una interrelación “varios a varios”.

	varios				id_plantilla
11	Uno a varios	Plantilla	Papel_de_trabajo		Id_plantilla
12	Uno a varios	Usuario	Plantilla		Id_usu
13	Uno a varios	Usuario	Bitácora		Id_usu
14	Uno a varios	Actividad	Bitácora		Id_actividad
15	Uno a varios	Revisor_informe	Papel_de_trabajo		Id_revisor
16	Uno a varios	Elaborador_informe	Papel_de_trabajo		Id_elaborador
17	Varios a varios	Papel_de_trabajo	Firmante_acta	Firmante_papel	Id_papel, id_firmante
18	Uno a varios	Papel_de_trabajo	Requerimientos		Id_papel
19	Uno a varios	Papel_de_trabajo	Contenido_plani		Id_papel
20	Uno a varios	Contenido_plani	Procedimiento_plani		Id_contenido
21	Uno a varios	Procedimiento_plani	Subprocedimiento_plani		Id_procedimiento

Tabla 5: Interrelaciones en la Base de Datos del Sistema

Como puede observarse de la tabla anterior, las interrelaciones varios a varios implican la creación de una entidad adicional de enlace. Estas entidades sólo contienen como atributos las llaves primarias de las tablas que enlazan.

De la relación enlace *Preg_papel* se puede sacar un provecho adicional: Como en esta entidad se almacenan las preguntas por cada papel de trabajo, se pueden registrar aquí mismo las respuestas que el usuario (rol operador u administrador) dé para dichas preguntas. Estas respuestas en el sistema se reducen a tres: “sí”, “no” y “no aplica”. Además interesa agregar un atributo más de observaciones opcionales para cada pregunta. Al atributo respuesta se le llamará *resp* y al de observaciones, *obser*. Esto elimina la entidad *Respuesta* que se había obtenido en la página 65.

Finalmente, con todo lo considerado en este análisis, se recopilan las relaciones del sistema en orden alfabético. Los atributos clave principal aparecen subrayados y los atributos clave foránea se denotan con cursiva.

Actividad (id_actividad, descripción)

Bitacora (id_entrada, *id_usu*, *id_actividad*, fecha, hora)

Categoria (id_categoria, nom_categoria, id_ced)

Cedula (id_ced, nom_ced)

Contacto (id_contact, ape_contact, nom_contact, tel_contact, ema_contact, id_insti)

Contenido_plani (id_contenido, id_papel, orden)

Elaborador_informe (id_elaborador, nombre_elaborador)

Firmante_acta (id_firmante, ape_firmante, nom_firmante, cargo_firmante)

Firmante_papel (id_papel, id_firmante)

Institucion (id_insti, nom_insti, id_tipoinsti, dir_insti, tel_insti, estado)

Papel_de_trabajo (id_papel, nom_papel, fech_creac, fech_actua, id_insti, descrip, id_plantilla, estado, enca_acta, fec_infor, id_elaborador, id_revisor, asunto_nota, enca_nota, intro_nota, conclu_nota, pie_nota, destinatario_requer, fecha_inicio_audi, fecha_fin_audi)

Plantilla (id_plantilla, nom_plantilla, id_usu)

Preg_papel (id_preg, id_papel, resp, obser)

Preg_plantilla (id_preg, id_plantilla)

Preg_tipo (id_tipoinsti, id_preg)

Pregunta (id_preg, tex_preg, id_categoria)

Procedimiento_plani (<u>id_procedimiento</u> , <u>id_contenido</u> , descripción, orden)
Requerimiento (<u>id_requer</u> , nombre_requer, descrip_requer, entrega_requer, fecha_recibido, <u>id_papel</u>)
Revisor_informe (<u>id_revisor</u> , nom_revisor)
Rol (<u>id_rol</u> , tipo_rol)
Subprocedimiento_plani (<u>id_subprocedimiento</u> , <u>id_procedimiento</u> , orden)
Tipo_institucion (<u>id_tipoinsti</u> , nom_insti)
Usu_papel (<u>id_usu</u> , <u>id_papel</u>)
Usuario (<u>id_usu</u> , ape_usu, nom_usu, cargo_usu, login, passw, <u>id_rol</u>)

2.2.4 Segunda aplicación del modelo Entidad – Relación

2.2.4.1 Tercer diagrama Entidad – Relación

Las relaciones finales obtenidas en el apartado inmediato anterior, se diagraman en la figura 23. En ella se pueden notar algunos cambios de notación con respecto a los dos diagramas Entidad – Relación anteriormente vistos, los cuales se explican a continuación:

- Se omiten, por cuestiones de espacio, las elipses que representan los atributos de cada entidad. Para una referencia completa de atributos, consultar el **Diccionario de Datos**.
- Cada símbolo de interrelación (los rombos) se numeran de acuerdo a la relación que representan, referenciada en la *tabla 5* y se leen de acuerdo a la siguiente guía:

Símbolo	Frase verbal de la relación
1	Una <i>institución</i> puede tener varios <i>papeles de trabajo</i>
2	Una <i>institución</i> puede tener varios <i>contactos</i>
3	Un <i>tipo de institución</i> puede clasificar varias <i>instituciones</i> .
4	Una <i>pregunta</i> puede estar incluida en varios <i>papeles de trabajo</i> .
5	Un <i>papel de trabajo</i> contiene varias <i>preguntas</i>
6	Una <i>categoría</i> clasifica a varias <i>preguntas</i>
7	Varios <i>usuarios</i> poseen varios <i>papeles de trabajo</i>
8	Varias <i>preguntas</i> pueden aplicar a varios <i>tipos de instituciones</i>
9	Una <i>cédula de auditoría</i> puede tener varias <i>categorías</i> de <i>preguntas</i> .
10	Varios <i>usuarios</i> clasifican dentro de un <i>rol</i>
11	Varias <i>preguntas</i> están incluidas dentro de varias <i>plantillas</i>
12	Un <i>usuario</i> puede poseer varias <i>plantillas</i> creadas
13	Una <i>plantilla</i> aplica a varios <i>papeles de trabajo</i>
14	Una <i>actividad</i> puede estar incluida en varias entradas de la <i>bitácora</i>
15	Un <i>usuario</i> puede tener varias entradas de la <i>bitácora</i> .

16	Un <i>revisor de informe</i> puede revisar varios informes de <i>papeles de trabajo</i> .
17	Un <i>elaborador de informe</i> puede elaborar varios informes de <i>papeles de trabajo</i> .
18	Varios <i>firmantes de acta</i> pueden firmar varias actas de <i>papeles de trabajo</i> .
19	Un <i>papel de trabajo</i> tiene varios <i>requerimientos</i> en su planeación.
20	Un <i>papel de trabajo</i> tiene varios <i>contenidos</i> en su planeación.
21	Un <i>contenido</i> de planeación se divide en varios <i>procedimientos</i>
22	Un <i>procedimiento</i> de planeación se divide en varios <i>subprocedimientos</i>

Tabla 6: Referencia de interrelaciones en tercer diagrama Entidad - Relación

- Las entidades, representadas por rectángulos, se referencian de acuerdo a la siguiente tabla:

Símbolo	Nombre de la Entidad
1	Subprocedimiento_plani
2	Procedimiento_plani
3	Contenido_plani
4	Requerimiento
5	Firmante_acta
6	Revisor_informe
7	Elaborador_informe

8	Papel_de_trabajo
9	Institución
10	Contacto
11	Tipo_institucion
12	Pregunta
13	Preg_papel
14	Categoría
15	Actividad
16	Bitácora
17	Cedula
18	Plantilla
19	Usuario
20	Rol

Tabla 7: Referencia de entidades en tercer diagrama Entidad - Relación

La figura 23 se encuentra en la página siguiente

2.2.4.2 Tercer Diagrama Entidad - Relación

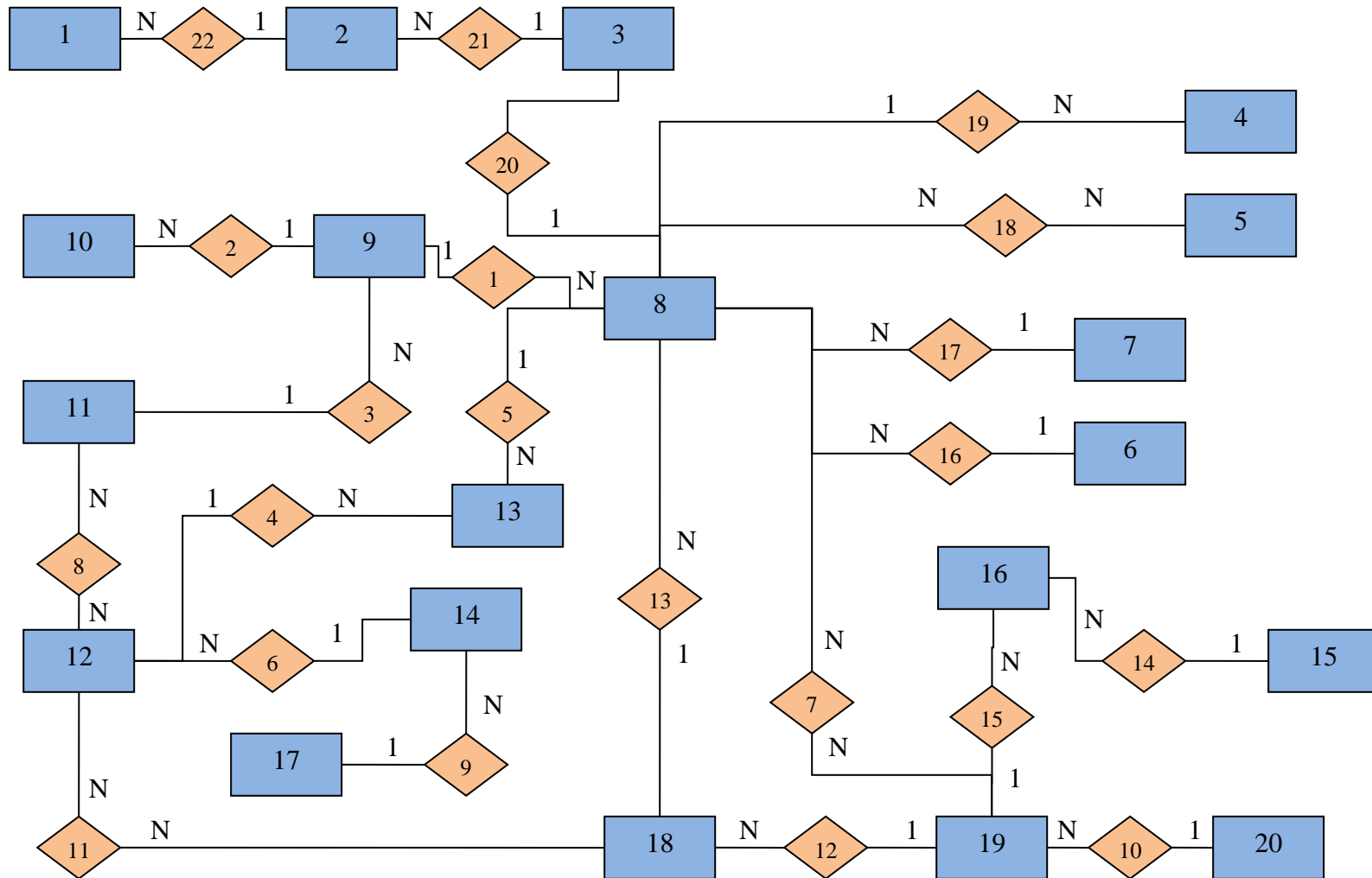


Figura 23: Tercer Diagrama Entidad - Relación

2.2.4.3 Base de datos – Diagrama Esquemático

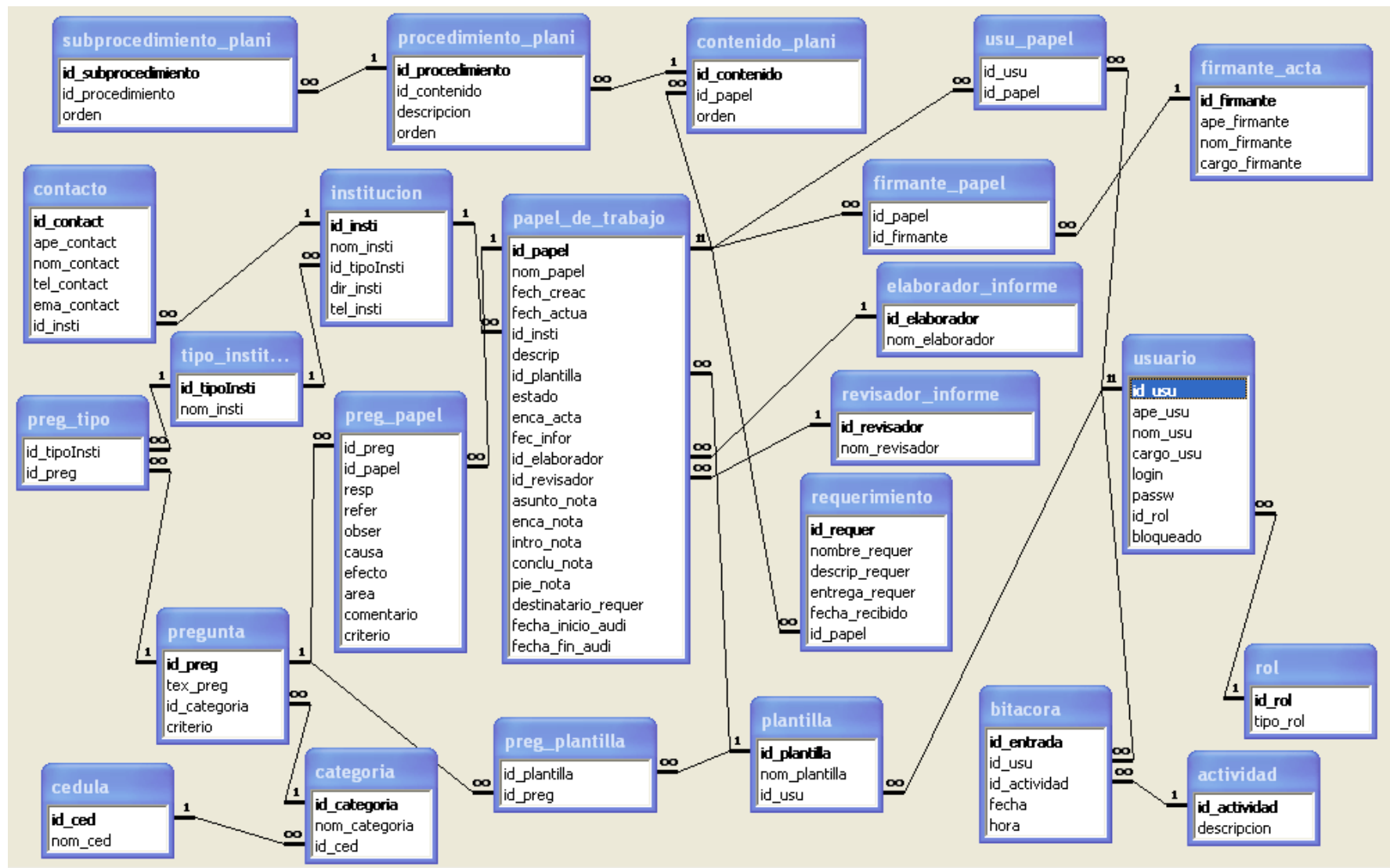


Figura 24: Diagrama Esquemático de la Base de Datos

2.3 CONSTRUCCIÓN DE LA BASE DE DATOS

2.3.1 Diccionario de Datos

En este apartado se hace una descripción de cada tabla de la base de datos. Asimismo, se detallan los campos de las entidades a ocupar en la base de datos. Por cada campo se especifican los siguientes puntos:

- Nombre del campo
- Tipo de dato del campo
- Tamaño del campo
- Número de decimales para los campos numéricos
- Descripción del campo

Software de Apoyo para Auditorías de Sistemas

Los datos se listan por tablas ordenadas alfabéticamente

ACTIVIDAD.MYI Contiene las diversas clasificaciones de preguntas pertenecientes a las cédulas de auditoría.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_ACTIVIDAD	INT	11	0	Número identificador de una actividad de bitácora específica.
2	DESCRIPCION	VARCHAR	100		Breve texto descriptivo de la actividad

BITACORA.MYI Contiene las diversas clasificaciones de preguntas pertenecientes a las cédulas de auditoría.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_ENTRADA	BIGINT	3	0	Número único identificador de la entrada de la bitácora.

2	ID_USU	INT	11	0	El identificador del usuario al cual se refiere la entrada de la bitácora
3	ID_ACTIVIDAD	INT	20	0	Número identificador del tipo de actividad registrada en esa entrada de bitácora.
4	FECHA	DATE			Fecha en la cual se registró dicha entrada de bitácora.
5	HORA	TIME			Hora en la cual se registró dicha entrada de la bitácora

CATEGORIA.MYI Contiene las diversas clasificaciones de preguntas pertenecientes a las cédulas de auditoría.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_CATEGORIA	TINYINT	3	0	Correlativo de categorías de preguntas.
2	NOM_CATEGORIA	VARCHAR	30		Almacena el nombre de la categoría
3	ID_CED	TINYINT	3	0	Identificador de la cédula a la que pertenece la categoría.

CEDULA.MYI. Contiene los nombres de las cédulas de auditoría que manejará el sistema.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_CED	TINYINT	3	0	Correlativo de cédulas.
2	NOM_CED	ENUM			Nombre de la cédula

CONTACTO.MYI. Guarda la información referente a los contactos que se tendrán en cada una de las entidades.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_CONTACT	INT	10	0	Correlativo de contactos
2	APE_CONTACT	VARCHAR	30		Registra los apellidos de los contactos.
3	NOM_CONTACT	VARCHAR	30		Registra los nombres de los contactos.
4	TEL_CONTACT	VARCHAR	9		Guarda el teléfono de los contactos
5	EMA_CONTACT	VARCHAR	30		Guarda el correo electrónico de los contactos.
6	ID_INSTI	VARCHAR	5		Código de la institución a la cual pertenece el contacto.

CONTENIDO_PLANI.MYI Contiene las diversas clasificaciones de preguntas pertenecientes a las cédulas de auditoría.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_CONTENIDO	INT	11	0	Número identificador del campo contenido de la cédula de planeación.
2	ID_PAPEL	BIGINT	20	0	Identificador del papel de trabajo al cual pertenece dicho contenido de planeación
3	ORDEN	INT	11	0	Número de posición que le corresponde al contenido dentro de la cédula de planeación.

4	TEXTO	VARCHAR	50		Breve texto que distinguirá al contenido dentro de la cédula de planeación.
---	-------	---------	----	--	---

ELABORADOR_INFORME.MYI Contiene las diversas clasificaciones de preguntas pertenecientes a las cédulas de auditoría.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_ELABORADOR	INT	11	0	Número identificador de la persona que elabora el informe.
2	NOM_ELABORADOR	VARCHAR	30		Nombre de la persona que elabora el informe.

FIRMANTE_ACTA.MYI Contiene las diversas clasificaciones de preguntas pertenecientes a las cédulas de auditoría.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_FIRMANTE	INT	11	0	Número identificador del firmante del acta.
2	NOM_FIRMANTE	VARCHAR	30		Nombre de la persona que firma el informe.
3	CARGO_FIRMANTE	VARCHAR	20	0	Posición laboral del firmante.

FIRMANTE_PAPEL_ACTA.MYI Contiene las diversas clasificaciones de preguntas pertenecientes a las cédulas de auditoría.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_PAPEL	BIGINT	20	0	Número de identificador del papel a la cual pertenece un acta.

2	ID_FIRMANTE	INT	11	0	Número identificador de la persona que firma el acta de auditoría.
---	-------------	-----	----	---	--

INSTITUCION. MYI. Esta tabla es la que registra cada una de las entidades fiscalizadas.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_INSTI	VARCHAR	5		Código de la institución. Los dos primeros caracteres representan el tipo de entidad. Los tres caracteres siguientes forman un correlativo.
2	NOM_INSTI	VARCHAR	40		Registra el nombre de las entidades.
3	ID_TIPOINSTI	TINYINT	3	0	Identifica el tipo de entidad a la que pertenece.
4	DIR_INSTI	VARCHAR	50		Guarda la dirección de cada una de las entidades.
5	TEL_INSTI	VARCHAR	9		Guarda el teléfono de las entidades.
6	ESTADO	TINYINT	4	0	Indica el estado de una entidad, los cuales serán dos: activa e inactiva.

PAPEL_DE_TRABAJO. MYI. Registra todos los papeles de trabajo elaborados por los auditores de sistemas.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_PAPEL	BIGINT	20		Correlativo de papeles de trabajo.
2	NOM_PAPEL	VARCHAR	30		Registra los nombres de todos los papeles de trabajo

3	FECH_CREAC	DATE			Registra la fecha de creación de los papeles de trabajo.
4	FECH_ACTUA	DATE			Registra la fecha de modificación o último acceso a los papeles de trabajo
5	ID_INSTI	VARCHAR	5		Código de las entidades
6	DESCRIP	TEXT			Descripción de cada papel de trabajo.
7	ID_PLANTILLA	BIGINT	20		Identificador de la plantilla de preguntas en la que se basa el papel.
8	ESTADO	TINYINT	4	0	Clasifica al papel como cerrado o abierto.
9	ENCA_ACTA	TEXT			Texto que encabeza el acta de resultados del papel.
10	ENCA_INFOR	TEXT			Texto que encabeza el acta de informe de auditoría del papel de trabajo.
11	FEC_INFOR	DATE			Fecha de elaboración del informe de auditoría del papel de trabajo.

12	ID_ELABORADOR	INT	11	0	Número identificador de la persona que elabora el informe de auditoría.
13	ID_REVISADOR	INT	11	0	Número identificador de la persona que revisa el informe de auditoría
14	ASUNTO_NOTA	TEXT			Asunto de la nota de auditoría.
15	ENCA_NOTA	TEXT			Encabezado de la nota de auditoría.
16	INTRO_NOTA	TEXT			Introducción de la nota de auditoría.
17	CONCLU_NOTA	TEXT			Conclusión de la nota de auditoría.
18	PIE_NOTA	TEXT			Pie de la nota de auditoría.
19	DESTINATARIO_REQUER	VARCHAR	30		Nombre del destinatario al cual va dirigida la nota de requerimientos.
20	FECHA_INCIO_AUDI	DATE			Fecha de inicio de la auditoría.
21	FECHA_FIN_AUDI	DATE			Fecha de fin de auditoría.

PLANTILLA. MYI. Sirve para personalizar las preguntas que contendrán las cédulas de auditoría en cada papel de trabajo.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_PLANTILLA	BIGINT	20	0	Correlativo de plantillas.
2	NOM_PLANTILLA	VARCHAR	30		Almacena el nombre de la plantilla personalizada.
3	ID_USU	INT	10	0	Almacena el id del usuario propietario de la plantilla.

PREG_PAPEL. MYI. Esta tabla especifica cuáles de todas las preguntas del sistema se encuentra en cada papel, asimismo, registra las respuestas y observaciones dadas a dichas preguntas.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_PREG	BIGINT	20	0	Código que identifica de manera única a cada pregunta.
2	ID_PAPEL	BIGINT	20	0	Código que identifica de manera única a cada papel de trabajo.
3	RESP	ENUM			Registra las respuestas de una pregunta específica en un papel de trabajo determinado.
4	REFER	TEXT			Identificara a que papel de trabajo está referenciada cada pregunta.
4	OBSER	TEXT			Registra las observaciones de una pregunta específica en un papel de trabajo determinado.
5	CAUSA	TEXT			Almacena la información que describe lo que produce la situación que se detalla en el apartado de CONDICIÓN en el informe de hallazgos.

6	EFEECTO	TEXT			Registra los datos que describen las consecuencias de lo detallado en el apartado de CONDICIÓN en el informe de hallazgos.
7	AREA	TEXT			Registra la Identificación del área a la que corresponden los hallazgos encontrados.
8	COMENTARIO	TEXT			Registra la observación del personal con el que se está discutiendo el acta.

PREG_PLANTILLA.MYI. Sirve como enlace entre la tabla de preguntas y la tabla de plantillas. Especifica a qué plantilla pertenece una determinada pregunta.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_PREG	BIGINT	20	0	Almacena el id de la pregunta almacenada en la plantilla.
2	ID_PLANTILLA	BIGINT	20	0	Almacena el id de la plantilla personalizada.

PREG_TIPO.MYI Sirve como enlace entre la tabla de preguntas y la tabla de tipos de instituciones. Especifica a qué tipo de entidad aplica una determinada pregunta de auditoría.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_TIPOINSTI	TINYINT	3	0	Identifica de manera única a cada tipo de institución.
2	ID_PREG	VARCHAR	20		Identifica a una pregunta en específico del sistema.

PREGUNTA. MYI. Sirve para almacenar los enunciados de las preguntas del sistema y especificar a qué cedula pertenece cada una.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_PREG	BIGINT	20	0	Correlativo de preguntas.
2	TEX_PREG	TEXT			Enunciado de la pregunta.
3	ID_CED	TINYINT	3	0	Código de la cédula a la que pertenece la pregunta.
4	CRITERIO	TEXT			En este campo se almacenará la información referente a la base legal de cada hallazgo.

PROCEDIMIENTO_PLANI. MYI. Sirve para almacenar los enunciados de las preguntas del sistema y especificar a qué cedula pertenece cada una.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_PROCEDIMIENTO	INT	11	0	Número identificador del procedimiento de la planeación.
2	ID_CONTENIDO	INT	11	0	Número identificador del contenido al cual pertenece el procedimiento.
3	DESCRIPCION	TEXT			Breve descripción del procedimiento.
4	ORDEN	TEXT			Número de la posición que ocupará el procedimiento en la cédula de planeación.
5	TEXTO	TEXT			Breve texto que distinguirá al subprocedimiento dentro de la cédula de planeación.

REVISADOR_INFORME. MYI. Sirve para almacenar los enunciados de las preguntas del sistema y especificar a qué cedula pertenece cada una.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_REVISADOR	INT	11	0	Número identificador de la persona que revisa el informe de auditoría.
2	NOM_REVISADOR	VARCHAR	30		Nombre de la persona que revisa el informe de auditoría.

ROL. MYI. Almacena los roles de usuario que existirán en el sistema.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_ROL	TINYINT	5	0	Correlativo de roles.
2	TIPO_ROL	ENUM			Nombre del rol de usuario.

SUBPROCEDIMIENTO_PLANI. MYI. Sirve para almacenar los enunciados de las preguntas del sistema y especificar a qué cedula pertenece cada una.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_SUBPROCEDIMIENTO	INT	11	0	Número identificador del subprocedimiento de la planeación.
2	ID_PROCEDIMIENTO	INT	11	0	Número identificador del procedimiento al cual pertenece el subprocedimiento.
3	ORDEN	INT	11	0	Número de posición que ocupará el subprocedimiento dentro de la cédula de planeación.

4	TEXTO	VARCHAR	50		Breve texto que se colocará en la cédula de planeación para este subprocedimiento.
---	-------	---------	----	--	--

TIPO_INSTITUCION.MYI. Es una tabla catálogo de los tipos de institución permitidas por el sistema.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_TIPOINSTI	TINYINT	3	0	Correlativo de tipos de instituciones.
2	NOM_TIPO	VARCHAR	20		Almacena el nombre del tipo de institución.

USU_PAPEL. MYI. En esta tabla se registran los papeles de trabajo clasificados por el usuario que los creó.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_USU	INT	10	0	Identifica a cada usuario de forma única.
2	ID_PAPEL	BIGINT	20	0	Código que representa a un papel de trabajo específico.

USUARIO. MYI. Contiene información sobre cada usuario registrado en el sistema.

No	Campo	Tipo	Ancho	Dec	Descripción
1	ID_USU	INT	11	0	Correlativo de usuarios.
2	APE_USU	VARCHAR	30		Almacena el apellido del usuario.
3	NOM_USU	VARCHAR	30		Almacena el nombre del usuario.
4	CARGO_USU	VARCHAR	20		Registra el cargo que posee el usuario
5	LOGIN	VARCHAR	12		Es la identificación de cada usuario para con el sistema.

6	PASSW	VARCHAR	12		Almacena la clave secreta de cada usuario.
7	ID_ROL	TINYINT	3	0	Identifica el rol al cual pertenece el usuario.
8	BLOQUEADO	TINYINT	11	0	Es el indicador para saber si el usuario está autorizado para ingresar al sistema.

2.3.2 Elaboración de la Base de Datos en MySQL

2.3.2.1 Nombre de la Base de Datos

La base de datos del software llevará el nombre de: *bdsas* (Base de Datos del Software para Auditoría de Sistemas).

2.3.2.2 Especificación de las tablas

Las tablas se han creado haciendo uso de la sintaxis del comando *CREATE TABLE* del SQL estándar.

Para tener mayor información sobre los campos que contienen cada una de las tablas se puede verificar el **diccionario de datos** en este mismo documento.

2.3.2.3 Creación de interrelaciones

Para la construcción de los enlaces relacionales entre las tablas, se ha hecho uso del comando *FOREIGN KEY* y *REFERENCE* de MySQL. Para obtener mayor información sobre las relaciones entre tablas, revisar en la sección **tercer diagrama de entidad relación**.

2.3.2.4 Guión de Creación de la Base de Datos

El listado MySQL encargado de crear tablas, relaciones e insertar los primeros datos es el siguiente:

Archivo: **guion.sql**

```
--  
-- Base de datos: `bdsas`  
--  
-----  
--  
-- Estructura de tabla para la tabla `actividad`  
--  
  
CREATE TABLE `actividad` (  
  `id_actividad` int(11) NOT NULL default '0',  
  `descripcion` varchar(100) NOT NULL default '',  
  PRIMARY KEY (`id_actividad`),  
  UNIQUE KEY `descripcion` (`descripcion`)  
) TYPE=InnoDB;  
  
--  
-- Estructura de tabla para la tabla `bitacora`  
--  
  
CREATE TABLE `bitacora` (  
  `id_entrada` bigint(20) NOT NULL default '0',  
  `id_usu` int(11) unsigned NOT NULL default '0',  
  `id_actividad` int(20) NOT NULL default '0',  
  `fecha` date NOT NULL default '0000-00-00',  
  `hora` time NOT NULL default '00:00:00',  
  PRIMARY KEY (`id_entrada`),  
  KEY `id_actividad` (`id_actividad`),  
  KEY `id_usu` (`id_usu`)  
) TYPE=InnoDB;
```

```

--
-- Estructura de tabla para la tabla `categoria`
--

CREATE TABLE `categoria` (
  `id_categoria` tinyint(3) unsigned NOT NULL default '0',
  `nom_categoria` varchar(30) NOT NULL default '',
  `id_ced` tinyint(3) unsigned NOT NULL default '0',
  PRIMARY KEY (`id_categoria`),
  UNIQUE KEY `nom_categoria` (`nom_categoria`),
  KEY `id_ced` (`id_ced`)
) TYPE=InnoDB;

--

-- Volcar la base de datos para la tabla `categoria`
--

INSERT INTO `categoria` VALUES (0, 'GeneralPlani', 0);
INSERT INTO `categoria` VALUES (1, 'GeneralHard', 1);
INSERT INTO `categoria` VALUES (2, 'GeneralSoft', 2);
INSERT INTO `categoria` VALUES (3, 'GeneralPoli', 3);
INSERT INTO `categoria` VALUES (4, 'Base de Datos', 4);
INSERT INTO `categoria` VALUES (5, 'Contraseñas', 4);
INSERT INTO `categoria` VALUES (6, 'Seguridad física', 4);
INSERT INTO `categoria` VALUES (7, 'Seguridad General', 4);

```

```

--
-- Estructura de tabla para la tabla `cedula`
--

CREATE TABLE `cedula` (
  `id_ced` tinyint(3) unsigned NOT NULL default '0',
  `nom_ced` enum('cplan','ch','cs','cpp','cse','cha','cm') NOT
NULL default 'cplan',
  PRIMARY KEY (`id_ced`),
  UNIQUE KEY `nom_ced` (`nom_ced`)
) TYPE=InnoDB;

--
-- Volcar la base de datos para la tabla `cedula`
--

INSERT INTO `cedula` VALUES (0, 'cplan');
INSERT INTO `cedula` VALUES (1, 'ch');
INSERT INTO `cedula` VALUES (2, 'cs');
INSERT INTO `cedula` VALUES (3, 'cpp');
INSERT INTO `cedula` VALUES (4, 'cse');
INSERT INTO `cedula` VALUES (5, 'cha');
INSERT INTO `cedula` VALUES (6, 'cm');

--
-- Estructura de tabla para la tabla `contacto`
--

CREATE TABLE `contacto` (
  `id_contact` int(10) unsigned NOT NULL default '0',
  `ape_contact` varchar(30) NOT NULL default '',
  `nom_contact` varchar(30) NOT NULL default '',

```

```

`tel_contact` varchar(9) NOT NULL default '',
`ema_contact` varchar(30) default NULL,
`id_insti` varchar(5) NOT NULL default '',
PRIMARY KEY (`id_contact`),
UNIQUE KEY `tel_contact` (`tel_contact`),
UNIQUE KEY `ema_contact` (`ema_contact`),
KEY `id_insti` (`id_insti`)
) TYPE=InnoDB;

--
-- Estructura de tabla para la tabla `contenido_plani`
--

CREATE TABLE `contenido_plani` (
  `id_contenido` int(11) NOT NULL default '0',
  `id_papel` bigint(20) unsigned NOT NULL default '0',
  `orden` int(11) NOT NULL default '0',
  `texto` varchar(50) NOT NULL default '',
  PRIMARY KEY (`id_contenido`),
  KEY `id_papel` (`id_papel`)
) TYPE=InnoDB;

--
-- Estructura de tabla para la tabla `elaborador_informe`
--

CREATE TABLE `elaborador_informe` (
  `id_elaborador` int(11) NOT NULL default '0',
  `nom_elaborador` varchar(30) NOT NULL default '',
  PRIMARY KEY (`id_elaborador`)
) TYPE=InnoDB;

```

```

--
-- Estructura de tabla para la tabla `firmante_acta`
--

CREATE TABLE `firmante_acta` (
  `id_firmante` int(11) NOT NULL default '0',
  `nom_firmante` varchar(30) NOT NULL default '',
  `cargo_firmante` varchar(20) NOT NULL default '',
  PRIMARY KEY (`id_firmante`)
) TYPE=InnoDB;

--
-- Estructura de tabla para la tabla `firmante_papel_acta`
--

CREATE TABLE `firmante_papel_acta` (
  `id_papel` bigint(20) unsigned NOT NULL default '0',
  `id_firmante` int(11) NOT NULL default '0',
  KEY `id_papel` (`id_papel`),
  KEY `id_firmante` (`id_firmante`)
) TYPE=InnoDB;

--
-- Estructura de tabla para la tabla `institucion`
--

CREATE TABLE `institucion` (
  `id_insti` varchar(5) NOT NULL default '',
  `nom_insti` varchar(40) NOT NULL default '',
  `id_tipoinsti` tinyint(3) unsigned NOT NULL default '0',
  `dir_insti` varchar(50) NOT NULL default '',
  `tel_insti` varchar(9) NOT NULL default '',

```

```

`estado` tinyint(4) NOT NULL default '1',
PRIMARY KEY (`id_insti`),
UNIQUE KEY `nom_insti` (`nom_insti`),
UNIQUE KEY `dir_insti` (`dir_insti`),
UNIQUE KEY `tel_insti` (`tel_insti`),
KEY `id_tipoinsti` (`id_tipoinsti`)
) TYPE=InnoDB;

--
-- Estructura de tabla para la tabla `papel_de_trabajo`
--

CREATE TABLE `papel_de_trabajo` (
  `id_papel` bigint(20) unsigned NOT NULL default '0',
  `nom_papel` varchar(30) NOT NULL default '',
  `fech_creac` date NOT NULL default '0000-00-00',
  `fech_actua` date NOT NULL default '0000-00-00',
  `id_insti` varchar(5) NOT NULL default '',
  `descrip` text,
  `id_plantilla` bigint(20) unsigned NOT NULL default '0',
  `estado` tinyint(4) NOT NULL default '1',
  `enca_acta` text NOT NULL,
  `enca_infor` text NOT NULL,
  `fec_infor` date NOT NULL default '0000-00-00',
  `id_elaborador` int(11) default '0',
  `id_revisor` int(11) default '0',
  `asunto_nota` text NOT NULL,
  `enca_nota` text NOT NULL,
  `intro_nota` text NOT NULL,
  `conclu_nota` text NOT NULL,
  `pie_nota` text NOT NULL,
  `destinatario_requer` varchar(30) NOT NULL default ''
)

```

```

`cargo_requer` varchar(30) NOT NULL default '',
`fecha_inicio_audi` date NOT NULL default '0000-00-00',
`fecha_fin_audi` date NOT NULL default '0000-00-00',
PRIMARY KEY (`id_papel`),
UNIQUE KEY `nom_papel` (`nom_papel`),
KEY `id_insti` (`id_insti`),
KEY `id_plantilla` (`id_plantilla`),
KEY `id_elaborador` (`id_elaborador`),
KEY `id_revisor` (`id_revisor`)
) TYPE=InnoDB;

--
-- Estructura de tabla para la tabla `plantilla`
--

CREATE TABLE `plantilla` (
  `id_plantilla` bigint(20) unsigned NOT NULL default '0',
  `nom_plantilla` varchar(30) NOT NULL default '',
  `id_usu` int(10) unsigned NOT NULL default '0',
  PRIMARY KEY (`id_plantilla`),
  UNIQUE KEY `nom_plantilla` (`nom_plantilla`),
  KEY `id_usu` (`id_usu`)
) TYPE=InnoDB;

--
-- Volcar la base de datos para la tabla `plantilla`
--

INSERT INTO `plantilla` VALUES (0, 'GeneralAlmacenes', 0);
INSERT INTO `plantilla` VALUES (1, 'GeneralBolsa', 0);
INSERT INTO `plantilla` VALUES (2, 'GeneralCasas', 0);
INSERT INTO `plantilla` VALUES (3, 'GeneralCedeval', 0);

```

```

--
-- Estructura de tabla para la tabla `preg_papel`
--

CREATE TABLE `preg_papel` (
  `id_preg` bigint(20) unsigned NOT NULL default '0',
  `id_papel` bigint(20) unsigned NOT NULL default '0',
  `resp` enum('si','no','na','in') default NULL,
  `refer` text NOT NULL,
  `obser` text,
  `causa` text NOT NULL,
  `efecto` text NOT NULL,
  `area` text,
  `comentario` text,
  `criterio` text NOT NULL,
  KEY `id_preg` (`id_preg`),
  KEY `id_papel` (`id_papel`)
) TYPE=InnoDB;

```

```

--
-- Estructura de tabla para la tabla `preg_plantilla`
--

```

```

CREATE TABLE `preg_plantilla` (
  `id_preg` bigint(20) unsigned NOT NULL default '0',
  `id_plantilla` bigint(20) unsigned default NULL,
  KEY `id_preg` (`id_preg`),
  KEY `id_plantilla` (`id_plantilla`)
) TYPE=InnoDB;

```

```

--
-- Estructura de tabla para la tabla `preg_tipo`
--

CREATE TABLE `preg_tipo` (
  `id_tipoinsti` tinyint(3) unsigned NOT NULL default '0',
  `id_preg` bigint(20) unsigned NOT NULL default '0',
  KEY `id_tipoinsti` (`id_tipoinsti`),
  KEY `id_preg` (`id_preg`)
) TYPE=InnoDB;

```

```

--
-- Estructura de tabla para la tabla `pregunta`
--

CREATE TABLE `pregunta` (
  `id_preg` bigint(20) unsigned NOT NULL default '0',
  `tex_preg` text NOT NULL,
  `id_categoria` tinyint(3) unsigned NOT NULL default '0',
  PRIMARY KEY (`id_preg`),
  KEY `id_categoria` (`id_categoria`)
) TYPE=InnoDB;

```

```

--
-- Estructura de tabla para la tabla `procedimiento_plani`
--

CREATE TABLE `procedimiento_plani` (
  `id_procedimiento` int(11) NOT NULL default '0',
  `id_contenido` int(11) NOT NULL default '0',
  `descripcion` text NOT NULL,
  `orden` int(11) NOT NULL default '0',

```

```

    `texto` varchar(50) NOT NULL default '',
    PRIMARY KEY (`id_procedimiento`),
    KEY `id_contenido` (`id_contenido`)
) TYPE=InnoDB;

--
-- Estructura de tabla para la tabla `requerimiento`
--

CREATE TABLE `requerimiento` (
  `id_requer` bigint(20) NOT NULL default '0',
  `nombre_requer` varchar(50) NOT NULL default '',
  `descrip_requer` varchar(100) NOT NULL default '',
  `entrega_requer` varchar(30) NOT NULL default '',
  `fecha_recibido` date NOT NULL default '0000-00-00',
  `id_papel` bigint(20) unsigned NOT NULL default '0',
  PRIMARY KEY (`id_requer`),
  KEY `id_papel` (`id_papel`)
) TYPE=InnoDB;

--
-- Estructura de tabla para la tabla `revisor_informe`
--

CREATE TABLE `revisor_informe` (
  `id_revisor` int(11) NOT NULL default '0',
  `nom_revisor` varchar(30) NOT NULL default '',
  PRIMARY KEY (`id_revisor`)
) TYPE=InnoDB;

```

```

--
-- Estructura de tabla para la tabla `rol`
--

CREATE TABLE `rol` (
  `id_rol` tinyint(3) unsigned NOT NULL default '0',
  `tipo_rol` enum('admin','ope','consul') NOT NULL default
'admin',
  PRIMARY KEY (`id_rol`),
  UNIQUE KEY `tipo_rol` (`tipo_rol`)
) TYPE=InnoDB;

--
-- Volcar la base de datos para la tabla `rol`
--

INSERT INTO `rol` VALUES (0, 'admin');
INSERT INTO `rol` VALUES (2, 'ope');
INSERT INTO `rol` VALUES (1, 'consul');

--
-- Estructura de tabla para la tabla `subprocedimiento_plani`
--

CREATE TABLE `subprocedimiento_plani` (
  `id_subprocedimiento` int(11) NOT NULL default '0',
  `id_procedimiento` int(11) NOT NULL default '0',
  `orden` int(11) NOT NULL default '0',
  `texto` varchar(50) NOT NULL default '',
  PRIMARY KEY (`id_subprocedimiento`),
  KEY `id_procedimiento` (`id_procedimiento`)
) TYPE=InnoDB;

```

```

--
-- Estructura de tabla para la tabla `tipo_institucion`
--

CREATE TABLE `tipo_institucion` (
  `id_tipoinsti` tinyint(3) unsigned NOT NULL default '0',
  `nom_tipo` varchar(20) NOT NULL default '',
  PRIMARY KEY (`id_tipoinsti`),
  UNIQUE KEY `nom_tipo` (`nom_tipo`)
) TYPE=InnoDB;

--
-- Volcar la base de datos para la tabla `tipo_institucion`
--

INSERT INTO `tipo_institucion` VALUES (0, 'Almacen');
INSERT INTO `tipo_institucion` VALUES (1, 'Bolsa de Valores');
INSERT INTO `tipo_institucion` VALUES (2, 'Casa Corredora');
INSERT INTO `tipo_institucion` VALUES (3, 'CEDEVAL');

--
-- Estructura de tabla para la tabla `usu_papel`
--

CREATE TABLE `usu_papel` (
  `id_usu` int(10) unsigned NOT NULL default '0',
  `id_papel` bigint(20) unsigned NOT NULL default '0',
  KEY `id_usu` (`id_usu`),
  KEY `id_papel` (`id_papel`)
) TYPE=InnoDB;

```

```

--
-- Estructura de tabla para la tabla `usuario`
--

CREATE TABLE `usuario` (
  `id_usu` int(10) unsigned NOT NULL default '0',
  `ape_usu` varchar(30) NOT NULL default '',
  `nom_usu` varchar(30) NOT NULL default '',
  `cargo_usu` varchar(20) NOT NULL default '',
  `login` varchar(12) NOT NULL default '',
  `passw` varchar(12) NOT NULL default '',
  `id_rol` tinyint(3) unsigned NOT NULL default '0',
  `bloqueado` tinyint(11) NOT NULL default '0',
  PRIMARY KEY (`id_usu`),
  UNIQUE KEY `login` (`login`),
  UNIQUE KEY `passw` (`passw`),
  KEY `id_rol` (`id_rol`)
) TYPE=InnoDB;

--
-- Volcar la base de datos para la tabla `usuario`
--

INSERT INTO `usuario` VALUES (0, 'General', 'General',
'ninguno', 'sa', 'sap', 0, 0);

--
-- Filtros para las tablas descargadas (dump)
--
-- Filtros para la tabla `bitacora`
--

```

```

ALTER TABLE `bitacora`
  ADD CONSTRAINT `bitacora_ibfk_1` FOREIGN KEY (`id_actividad`)
REFERENCES `actividad` (`id_actividad`) ON DELETE CASCADE ON
UPDATE CASCADE,
  ADD CONSTRAINT `bitacora_ibfk_2` FOREIGN KEY (`id_usu`)
REFERENCES `usuario` (`id_usu`) ON DELETE CASCADE ON UPDATE
CASCADE;

--
-- Filtros para la tabla `categoria`
--
ALTER TABLE `categoria`
  ADD CONSTRAINT `categoria_ibfk_1` FOREIGN KEY (`id_ced`)
REFERENCES `cedula` (`id_ced`) ON DELETE CASCADE ON UPDATE
CASCADE;

--
-- Filtros para la tabla `contacto`
--
ALTER TABLE `contacto`
  ADD CONSTRAINT `contacto_ibfk_1` FOREIGN KEY (`id_insti`)
REFERENCES `institucion` (`id_insti`) ON DELETE CASCADE ON
UPDATE CASCADE;

--
-- Filtros para la tabla `contenido_plani`
--
ALTER TABLE `contenido_plani`
  ADD CONSTRAINT `contenido_plani_ibfk_1` FOREIGN KEY
(`id_papel`) REFERENCES `papel_de_trabajo` (`id_papel`) ON
DELETE CASCADE ON UPDATE CASCADE;

```

```

--
-- Filtros para la tabla `firmante_papel_acta`
--
ALTER TABLE `firmante_papel_acta`
  ADD CONSTRAINT `firmante_papel_acta_ibfk_1` FOREIGN KEY
(`id_firmante`) REFERENCES `firmante_acta` (`id_firmante`) ON
DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `firmante_papel_acta_ibfk_2` FOREIGN KEY
(`id_papel`) REFERENCES `papel_de_trabajo` (`id_papel`) ON
DELETE CASCADE ON UPDATE CASCADE;

--
-- Filtros para la tabla `institucion`
--
ALTER TABLE `institucion`
  ADD CONSTRAINT `institucion_ibfk_1` FOREIGN KEY
(`id_tipoinsti`) REFERENCES `tipo_institucion` (`id_tipoinsti`)
ON UPDATE CASCADE;

--
-- Filtros para la tabla `papel_de_trabajo`
--
ALTER TABLE `papel_de_trabajo`
  ADD CONSTRAINT `papel_de_trabajo_ibfk_10` FOREIGN KEY
(`id_revisor`) REFERENCES `revisor_informe`
(`id_revisor`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `papel_de_trabajo_ibfk_11` FOREIGN KEY
(`id_insti`) REFERENCES `institucion` (`id_insti`) ON UPDATE
CASCADE,
  ADD CONSTRAINT `papel_de_trabajo_ibfk_12` FOREIGN KEY
(`id_plantilla`) REFERENCES `plantilla` (`id_plantilla`) ON
UPDATE CASCADE,

```

```

    ADD CONSTRAINT `papel_de_trabajo_ibfk_9` FOREIGN KEY
(`id_elaborador`) REFERENCES `elaborador_informe`
(`id_elaborador`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Filtros para la tabla `plantilla`
--
ALTER TABLE `plantilla`
    ADD CONSTRAINT `plantilla_ibfk_1` FOREIGN KEY (`id_usu`)
REFERENCES `usuario` (`id_usu`) ON DELETE CASCADE ON UPDATE
CASCADE;

--
-- Filtros para la tabla `preg_papel`
--
ALTER TABLE `preg_papel`
    ADD CONSTRAINT `preg_papel_ibfk_1` FOREIGN KEY (`id_preg`)
REFERENCES `pregunta` (`id_preg`) ON UPDATE CASCADE,
    ADD CONSTRAINT `preg_papel_ibfk_2` FOREIGN KEY (`id_papel`)
REFERENCES `papel_de_trabajo` (`id_papel`) ON UPDATE CASCADE;

--
-- Filtros para la tabla `preg_plantilla`
--
ALTER TABLE `preg_plantilla`
    ADD CONSTRAINT `preg_plantilla_ibfk_1` FOREIGN KEY
(`id_preg`) REFERENCES `pregunta` (`id_preg`) ON DELETE CASCADE
ON UPDATE CASCADE,
    ADD CONSTRAINT `preg_plantilla_ibfk_2` FOREIGN KEY
(`id_plantilla`) REFERENCES `plantilla` (`id_plantilla`) ON
DELETE CASCADE ON UPDATE CASCADE;

```

```

--
-- Filtros para la tabla `preg_tipo`
--
ALTER TABLE `preg_tipo`
  ADD CONSTRAINT `preg_tipo_ibfk_1` FOREIGN KEY
(`id_tipoinsti`) REFERENCES `tipo_institucion` (`id_tipoinsti`)
ON UPDATE CASCADE,
  ADD CONSTRAINT `preg_tipo_ibfk_2` FOREIGN KEY (`id_preg`)
REFERENCES `pregunta` (`id_preg`) ON UPDATE CASCADE;

--
-- Filtros para la tabla `pregunta`
--
ALTER TABLE `pregunta`
  ADD CONSTRAINT `pregunta_ibfk_1` FOREIGN KEY (`id_categoria`)
REFERENCES `categoria` (`id_categoria`) ON UPDATE CASCADE;

--
-- Filtros para la tabla `procedimiento_plani`
--
ALTER TABLE `procedimiento_plani`
  ADD CONSTRAINT `procedimiento_plani_ibfk_1` FOREIGN KEY
(`id_contenido`) REFERENCES `contenido_plani` (`id_contenido`)
ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Filtros para la tabla `requerimiento`
--
ALTER TABLE `requerimiento`
  ADD CONSTRAINT `requerimiento_ibfk_1` FOREIGN KEY
(`id_papel`) REFERENCES `papel_de_trabajo` (`id_papel`) ON
DELETE CASCADE ON UPDATE CASCADE;

```

```

--
-- Filtros para la tabla `subprocedimiento_plani`
--

ALTER TABLE `subprocedimiento_plani`
  ADD CONSTRAINT `subprocedimiento_plani_ibfk_1` FOREIGN KEY
(`id_procedimiento`) REFERENCES `procedimiento_plani`
(`id_procedimiento`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Filtros para la tabla `usu_papel`
--

ALTER TABLE `usu_papel`
  ADD CONSTRAINT `usu_papel_ibfk_1` FOREIGN KEY (`id_usu`)
REFERENCES `usuario` (`id_usu`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `usu_papel_ibfk_2` FOREIGN KEY (`id_papel`)
REFERENCES `papel_de_trabajo` (`id_papel`) ON DELETE CASCADE ON
UPDATE CASCADE;

--
-- Filtros para la tabla `usuario`
--

ALTER TABLE `usuario`
  ADD CONSTRAINT `usuario_ibfk_1` FOREIGN KEY (`id_rol`)
REFERENCES `rol` (`id_rol`) ON DELETE CASCADE ON UPDATE
CASCADE;

```

CAPITULO III
DISEÑO
Y
CONSTRUCCION
DEL SISTEMA

3.1 MODELADO DE LA APLICACIÓN

3.1.1 Especificación del funcionamiento de la aplicación

La aplicación a desarrollar facilitará el trabajo del auditor de sistemas, apoyándolo en el manejo de los papeles de trabajo y en la generación de reportes durante la realización de la auditoría.

El usuario ingresará al sistema a través de una pantalla de autenticación. En esta pantalla se presentarán dos cuadros de texto:

- El primero, para la identificación del usuario.
- El segundo, para el ingreso de la contraseña.

Una vez ingresada esta información, el sistema verificará la validez del usuario. Si no es un usuario válido, presentará un mensaje de alerta indicando dicha condición. Este proceso se repetirá un máximo de tres veces, de lo contrario, la aplicación se cerrará y bloqueará al usuario. En caso de validar correctamente al usuario ingresado, el sistema mostrará la pantalla del menú principal de trabajo. Existirán tres tipos de usuarios:

Administrador: Que tendrá derechos de mantenimiento de usuarios e instituciones, además de crear y consultar papeles de trabajo.

Consulta: Este usuario tendrá la capacidad de revisar papeles de trabajo del sistema.

Operador: Este usuario podrá crear y consultar papeles de trabajo previamente creados.

En caso de validar correctamente al usuario ingresado, el sistema mostrará la pantalla del menú principal de trabajo. Dicho menú tendrá tres opciones principales:

- *Papeles de Trabajo*: Desde aquí se tendrá acceso a la creación de nuevos papeles de trabajo, a la apertura o cierre de papeles ya existentes y a la creación de nuevas plantillas de preguntas para el usuario en turno.

- *Mantenimiento*: El rol de administrador será el único que accederá a esta opción, donde podrá trabajar con las tablas de usuarios, instituciones y preguntas.
- *Salir*: Esta opción direccionará a una página de salida desde la cual el usuario optará por entrar de nuevo al sistema

Cuando se cree un nuevo papel o cuando se abra uno ya existente se trabajará en una pantalla con un menú adicional, en el cual se tendrán los nombres de todas las cédulas de auditoría. Cuando el usuario haga clic en el nombre de la cédula se presentarán las preguntas correspondientes para que sean contestadas. Se tendrá acceso a una opción para guardar los cambios que haga en cada papel.

Las cédulas de trabajo que manejará el sistema son las siguientes:

- Cédula sumaria de planeación.
- Cédula sumaria de hardware.
- Cédula sumaria de software.
- Cédula sumaria de políticas y procedimientos.
- Cédula sumaria de seguridad.
- Cédula sumaria de hallazgos.
- Cédula sumaria de marcas.

La cédula sumaria de planeación contiene preguntas relativas a la investigación previa a la realización de la auditoría.

La cédula sumaria de hardware se refiere a los aspectos físicos de los sistemas informáticos.

La cédula sumaria de software contiene aspectos relacionados a los sistemas de información que poseen las entidades fiscalizadas.

La cédula sumaria de políticas y procedimientos se refiere a la verificación de los métodos operativos adoptados por las entidades que son auditadas

La cédula sumaria de seguridad se refiere a la evaluación de los estándares de seguridad que poseen las entidades fiscalizadas.

La cédula sumaria de hallazgos contiene aquellos informes que contienen los resultados de la auditoría realizada.

La cédula sumaria de marcas describe la nomenclatura utilizada en las cédulas anteriores.

Las preguntas contenidas en cada cédula se extraerán de la base de datos diseñada en el capítulo II del presente documento.

En cada pantalla se presentará el nombre del usuario que está utilizando el sistema en el momento y el tipo de rol que le corresponde.

El usuario administrador podrá: agregar, modificar y eliminar usuarios; agregar y modificar instituciones; agregar, modificar y eliminar contactos.

En cada papel de trabajo el usuario operador podrá: contestar las preguntas y guardar los cambios que se hagan.

El usuario operador podrá agregar plantillas de preguntas a su “haber”.

Los datos que se inserten o modifiquen de los usuarios, instituciones, contactos y plantillas fueron determinados en el diseño de la base de datos del capítulo II de este documento.

3.1.2 Determinación de clases

Dentro de la especificación anterior, encontramos sustantivos significativos, los cuales corresponden a cada una de las tablas del sistema. Dichas tablas serán

objeto de diversas consultas tales como: extracción, inserción, modificación y eliminación de registros. Tenemos pues, que para cada una de las tablas que podrán ser mantenidas de alguna forma por el usuario, formaremos una clase diferente. Asimismo, los atributos de dichas tablas serán otros tantos para las clases y las operaciones de mantenimiento se definirán como métodos de dichas clases. Aquí se llamarán *métodos de mantenimiento*.

Lo anterior se resume en la tabla siguiente¹⁵:

Entidades : Clases	Campos : Atributos	Métodos de Mantenimiento
usuario : usuario	id_usu : numero ape_usu : apellidos nom_usu : nombre cargo_uso : cargo login : login passw : pass id_rol : rol	agregar eliminar modificar
institución : institución	id_insti : id nom_insti : nombre tip_insti : tipo dir_insti : direccion tel_insti : telefono	agregar modificar
contacto : contacto	id_contact : id ape_contact : apellido nom_contact : nombre tel_contact : telefono ema_contact : correo id_insti : insti	agregar modificar eliminar
pregunta : pregunta	id_preg : id tex_preg : texto id_ced : cedula tip_insti : insti	agregar
plantilla : plantilla	id_plantilla : idPlantilla nom_plantilla : nombrePlantilla id_usu : idUsuario	Agregar

Tabla 8: Clases del sistema, atributos y métodos de mantenimiento

¹⁵ En las clases Institución, Pregunta y Plantilla, no se han agregado métodos de eliminación debido a las relaciones entre tablas, las cuales podrían eliminar en cascada registros importantes de otras tablas. Por ejemplo, si un papel de trabajo utiliza una plantilla determinada y esta última se elimina, también se eliminan los datos guardados de dicho papel.

Además de los procedimientos usados indistintamente para el mantenimiento de las entidades, son necesarios otros que auxilien en las *restricciones de los campos*, como por ejemplo, la propiedad de valores únicos en ciertos campos tales como el login y el password de usuario o los teléfonos y correos electrónicos que no pueden contener valores repetidos. Estos métodos serán invocados dentro de la misma clase para asegurarse que los métodos de mantenimiento funcionen acorde al diseño de la base de datos y recibirán el nombre de *métodos de validación*.

Clase	Método de validación
Usuario	generar_numero verificar_login verificar_password
Institución	generar_id verificar_telefono verificar_id verificar_nombre verificar_bolsa verificar_cedeval
Contacto	generar_numero verificar_telefono verificar_correo verificar_id
Plantilla	generar_numero

Tabla 9: Clases del sistema y métodos de validación

Finalmente, se tienen aquellos métodos que permitirán devolver al código cliente los datos encapsulados en los objetos para su tratamiento respectivo. Dichos métodos serán llamados *métodos de resultados*.

Clase	Método de resultado
Usuario	cargar_datos recuperar_datos
Institución	cargar_datos recuperar_datos
Contacto	cargar_datos recuperar_datos

Tabla 10: Clases del sistema y métodos de resultado

Completamos así la lista de métodos para cada clase. En el programa fuente, dichas clases estarán codificadas en el archivo *clases.php* que será referenciado por aquellas páginas del sitio que necesiten instanciar sus clases.

En el desarrollo del código PHP habrán varias ocasiones en que se necesitará conectarse con la base de datos MySQL. Además se harán frecuentemente, consultas de selección. Para manejar ambos casos se hará uso de clases. La clase *Conexión* se hará cargo de manejar las propiedades del enlace con la base de datos y la clase *Selección* manejará las consultas de extracción de datos que se hagan a la base. Los atributos y métodos de ambas clases se verán en el apartado **3.1.3.2**

Diagramas de clases.

3.1.3 Diagramación UML

Debido a que el software desarrollado hace uso de objetos, será de utilidad la documentación de los procesos utilizando el lenguaje gráfico UML. Los diagramas utilizados son: Casos de uso, clases, secuencia, actividades y colaboración

3.1.3.1 Diagramas de casos de uso

En este apartado se representarán las actividades a las que los usuarios podrán acceder dentro del sistema, dependiendo del rol que les haya sido asignado (ver **especificación del funcionamiento en el apartado 3.1.1**).

Diagrama 1.

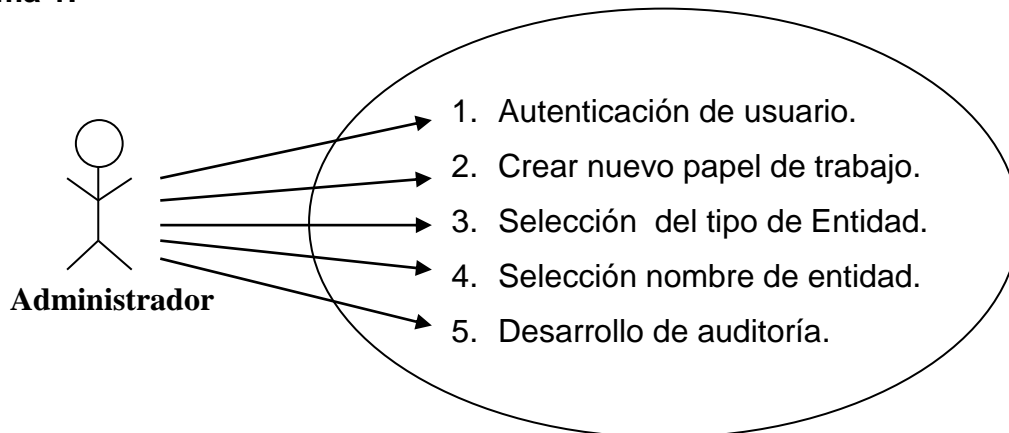


Figura 25: Diagrama 1 Casos de Uso

Usuario: Administrador **Actividad:** Crear nuevo papel de trabajo

Diagrama 2.

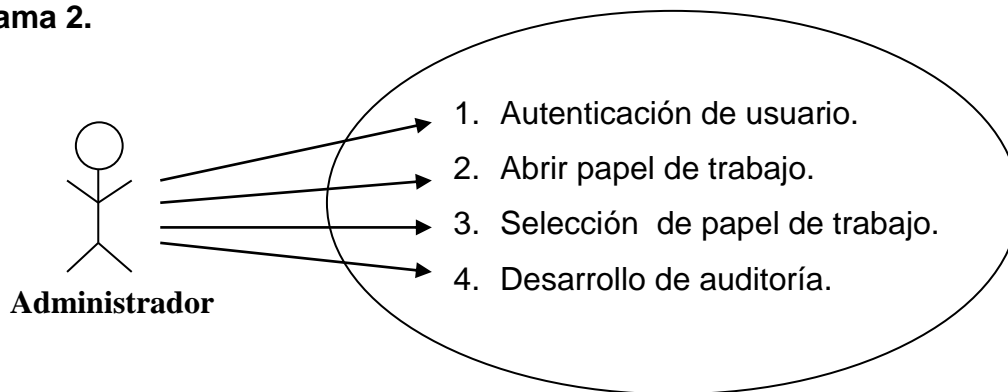


Figura 26: Diagrama 2 Casos de Uso

Usuario: Administrador **Actividad:** Abrir papel de trabajo

Diagrama 3.

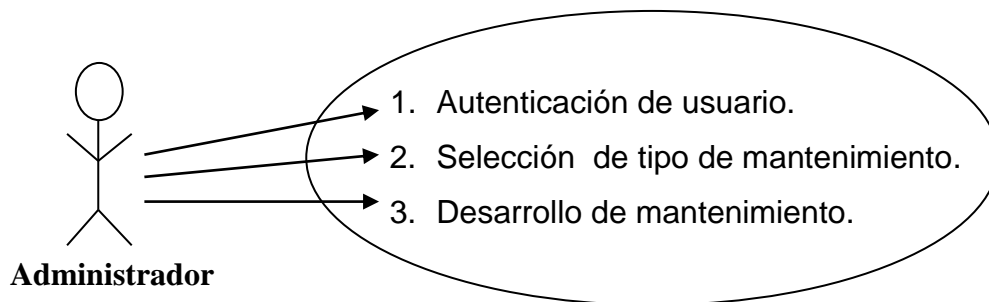


Figura 27: Diagrama 3 Casos de Uso

Usuario: Administrador **Actividad:** Mantenimiento de la base de datos

Diagrama 4.

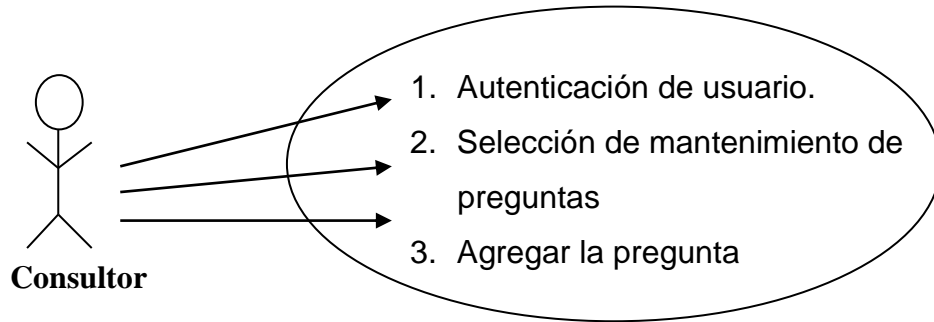


Figura 28: Diagrama 4 Casos de Uso

Usuario: Administrador

Actividad: Agregar una pregunta nueva.

Diagrama 5.

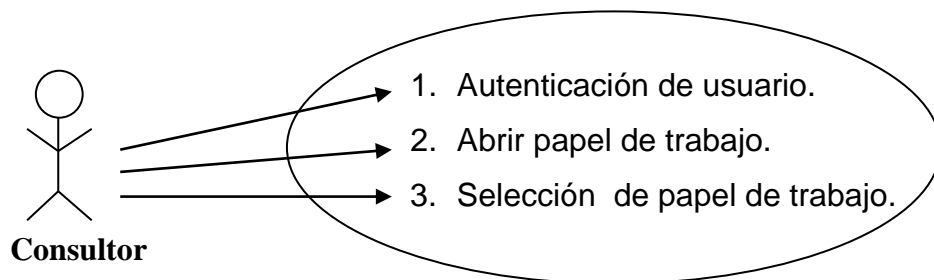


Figura 29: Diagrama 5 Casos de Uso

Usuario: Consultor

Actividad: Consultar un papel de trabajo existente

Diagrama 6.

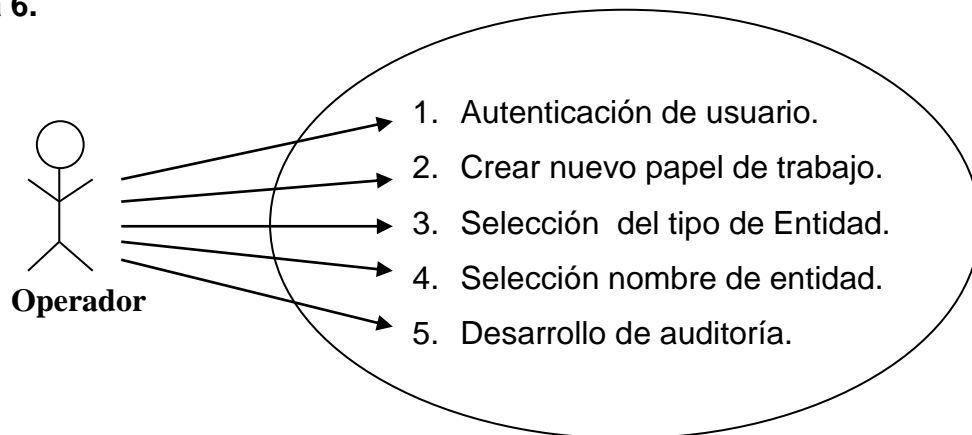


Figura 30: Diagrama 6 Casos de Uso

Usuario: Operador

Actividad: Crear un nuevo papel de trabajo

Diagrama 7.

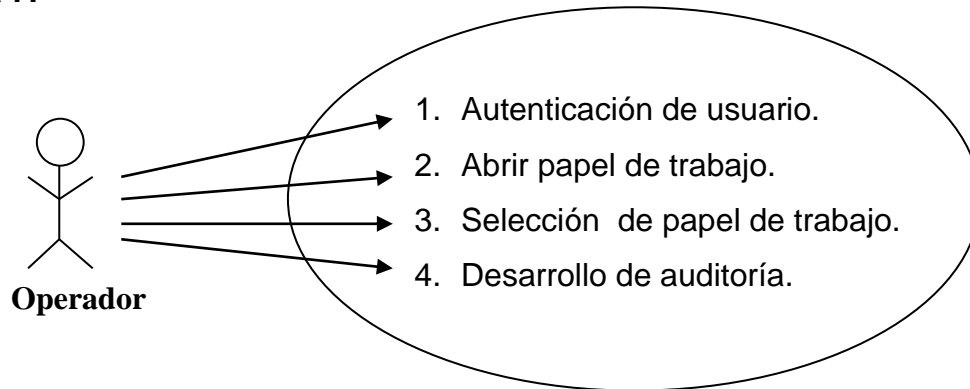


Figura 31: Diagrama 7 Casos de Uso

Usuario: Operador

Actividad: Abrir papel de trabajo

3.1.3.2 Diagramas de Clases

Los diagramas de clases muestran el encapsulamiento de datos y métodos dentro de las clases, así como también los argumentos de los métodos.

Diagrama 1: Clase Conexión

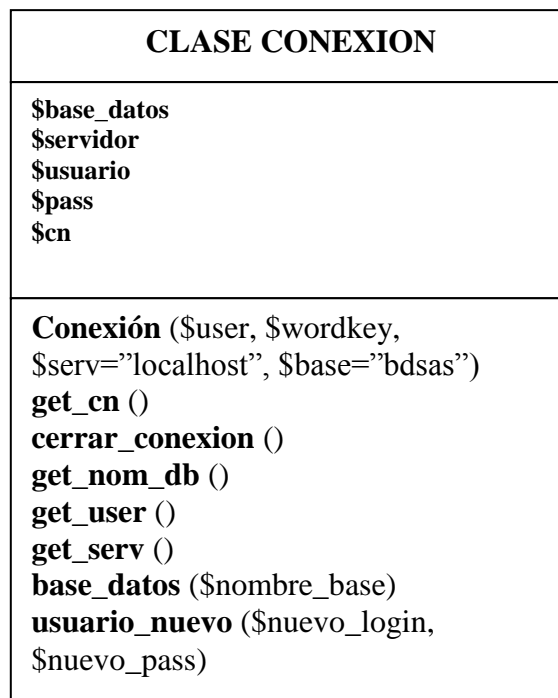


Figura 32: Diagrama 1 de Clase

Diagrama 2: Clase Selección

CLASE SELECCION
\$puntero \$num_filas \$num_columns
extraer (\$cadena_seleccion, \$con) get_pointer () get_numero_filas () get_numero_columnas () imprimir_datos ()

Figura 33: Diagrama 2 de Clase

Diagrama 3: Clase Usuario

CLASE USUARIO
\$numero \$nombre \$apellidos \$cargo \$login \$pass \$rol \$existente
generar_numero (\$con) verificar_password (\$palabra, \$con, \$opcion, \$id="") verificar_login (\$palabra, \$con, \$opcion, \$id="") verificar_administrador (\$palabra, \$con, \$opcion, \$id="") agregar (\$con, \$nom, \$ape, \$car, \$log, \$pas, \$ro) eliminar (\$con) modificar (\$con, \$nuevo_nom, \$nuevo_ape, \$nuevo_car, \$nuevo_log, \$nuevo_pas, \$nuevo_rol) cargar_datos (\$con, \$log) recuperar_datos ()

Figura 34: Diagrama 3 de Clase

Diagrama 4: Clase Institución

CLASE INSTITUCION
\$id \$nombre \$tipo \$direccion \$telefono \$estado
generar_id (\$nom, \$con, \$clasi) verificar_telefono (\$palabra, \$con, \$opcion, \$id="") verificar_id (\$palabra, \$con, \$opcion, \$id="") verificar_nombre (\$palabra, \$con, \$opcion, \$id="") verificar_bolsa (\$con, \$opcion, \$id="") verificar_cedeval (\$con, \$opcion, \$id="") agregar (\$con, \$nom, \$tip, \$dir, \$tel) modificar (\$con, \$nuevo_nom, \$nuevo_tipo, \$nueva_dir, \$nuevo_tel) cargar_datos (\$con, \$id) recuperar_datos ()

Figura 35: Diagrama 4 de Clase

Diagrama 5: Clase Preg_Plantilla

CLASE PREG_PLANTILLA
\$id \$idPregunta \$idPlantilla
generar_id (\$con) agregar (\$idPreg, \$idPlanti, \$con)

Figura 36: Diagrama 5 de Clase

Diagrama 6: Clase Contacto

CLASE CONTACTO
\$id \$apellido \$nombre \$teléfono \$correo \$insti \$existente
generar_numero (\$con) verificar_telefono (\$palabra, \$con, \$opcion, \$id="") verificar_correo (\$palabra, \$con, \$opcion, \$id="") verificar_id (\$palabra, \$con, \$opcion, \$id="") agregar (\$con, \$nom, \$ape, \$tel, \$ema, \$ins) eliminar (\$con) modificar (\$con, \$nuevo_nom, \$nuevo_ape, \$nuevo_tel, \$nuevo_ema, \$nuevo_ins) cargar_datos (\$con, \$id) recuperar_datos ()

Figura 37: Diagrama 6 de Clase

Diagrama 7: Clase Pregunta

CLASE PREGUNTA
\$id \$texto \$cedula \$tipInsti
generar_id (\$con) agregar (\$tex, \$ced, \$tipoInstitucion, \$con)

Figura 38: Diagrama 7 de Clase

Diagrama 8: Clase Preg_tipo

CLASE PREG_TIPO
\$idTipoIns \$idPregunta
agregar (\$idInsti, \$idPreg, \$con)

Figura 39: Diagrama 8 de Clase

Diagrama 9: Clase PapelTrabajo

CLASE PAPELTRABAJO
\$idPapel \$nombrePapel \$fechaCreacion \$fechaActualizacion \$idInstitucion \$descripcionPapel
generar_id (\$con) verficar_nombre (\$palabra, \$con, \$opcion, \$id="") verificar_id (\$palabra, \$con, \$opcion, \$id="") agregar (\$nom, \$fecCrea, \$fecActua, \$insti, \$descrip, \$con) modificar (\$con, \$nuevo_nom, \$nueva_descrip, \$nueva_fecActua="") cargar_datos (\$id,\$con)

Figura 40: Diagrama 9 de Clase

Diagrama 10: Clase Preg_papel

CLASE PREG_PAPEL
\$idPregunta \$idPapel \$respuesta \$observación
agregar (\$con, \$numeroPregunta, \$numeroPapel, \$respues, \$observa="") modificar (\$con, \$numeroPregunta, \$numeroPapel, \$nuevarespues, \$nuevaobserva="")

Figura 41: Diagrama 10 de Clase

Diagrama 11: Clase Plantilla

CLASE PLANTILLA
\$idPlantilla \$nombrePlantilla \$idUserario
generar_numero (\$con) agregar (\$nomPlanti,\$usuario,\$con)

Figura 42: Diagrama 11 de Clase

3.1.3.3 Diagramas de Secuencia

Los diagramas de secuencia describen la sucesión de operaciones que se dan cuando los usuarios o clases interactúan con el sistema. En otras palabras, muestra como se desarrollan en el tiempo las invocaciones a los métodos y las respuestas a los mismos.

Diagrama 1: Creación de un nuevo papel de trabajo

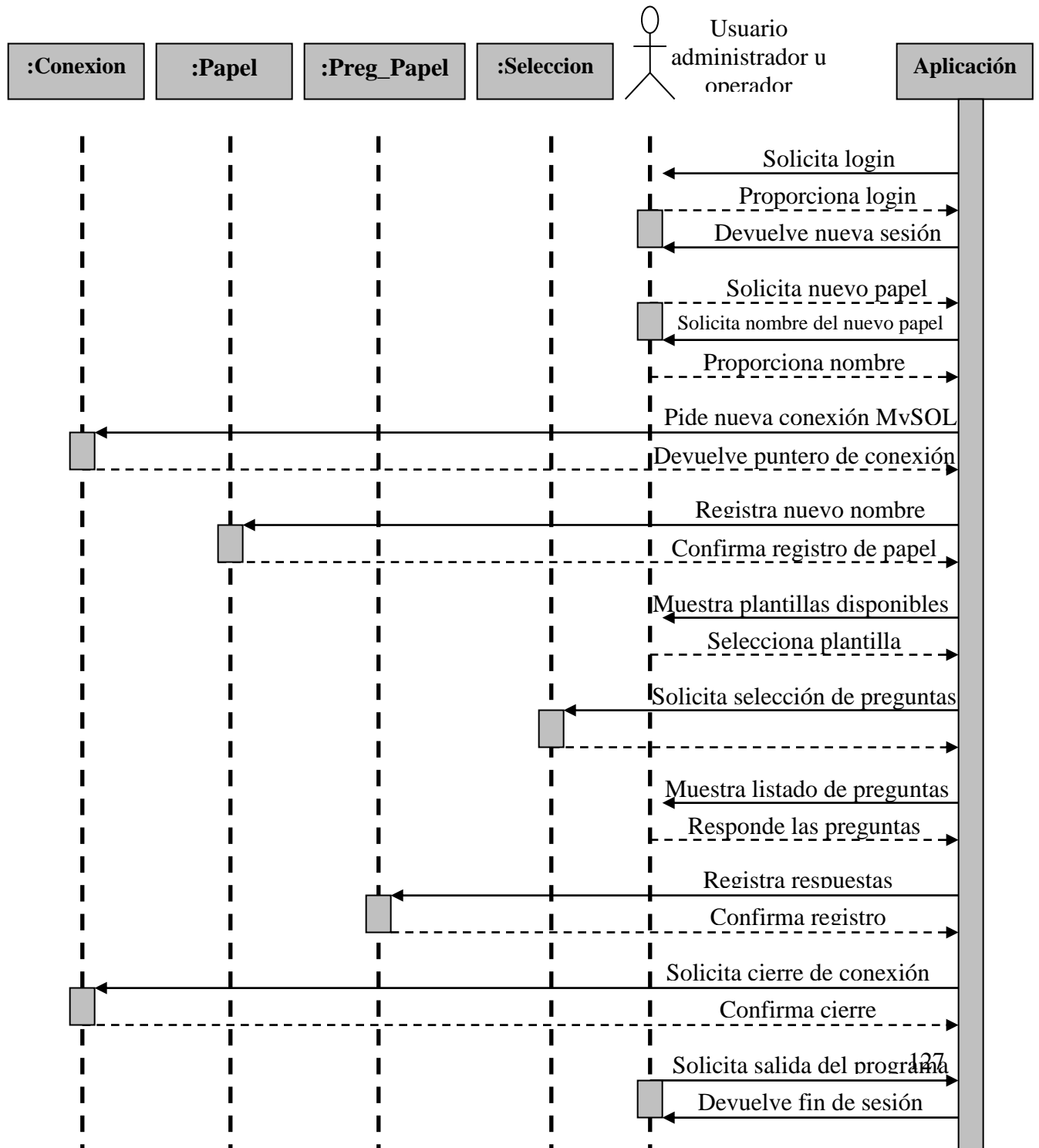


Figura 43: Diagrama 1 de Secuencia

Diagrama 2: Consultar un papel de trabajo existente

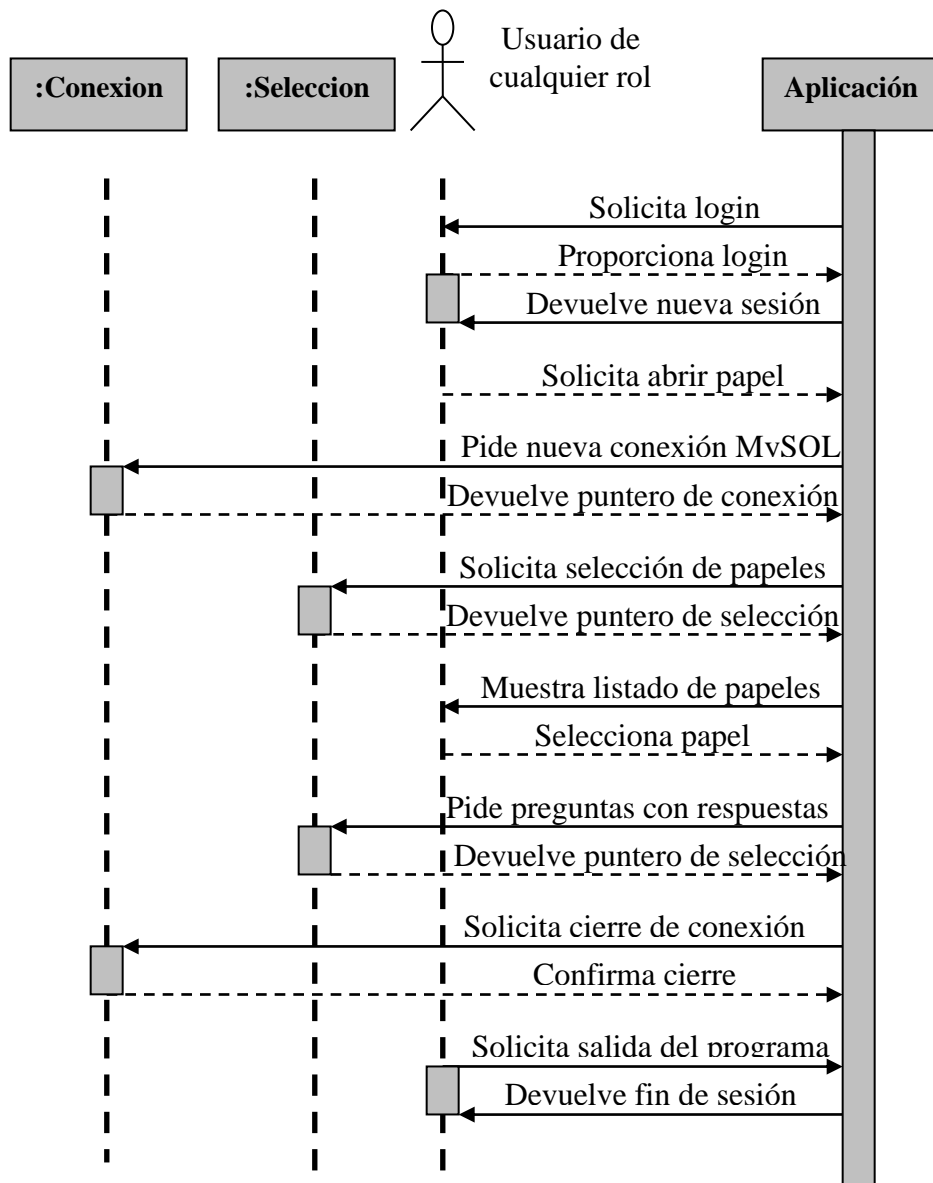


Figura 44: Diagrama 2 de Secuencia

Diagrama 3: Registrar un nuevo usuario al sistema

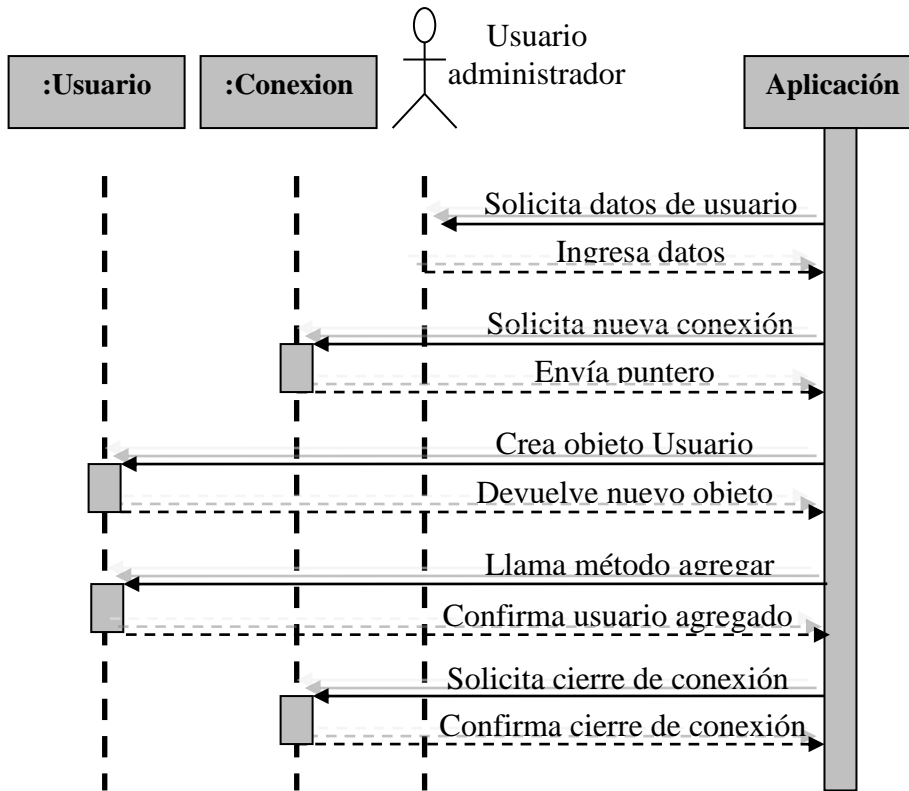


Figura 45: Diagrama 3 de Secuencia

Diagrama 4: Registrar una nueva institución al sistema

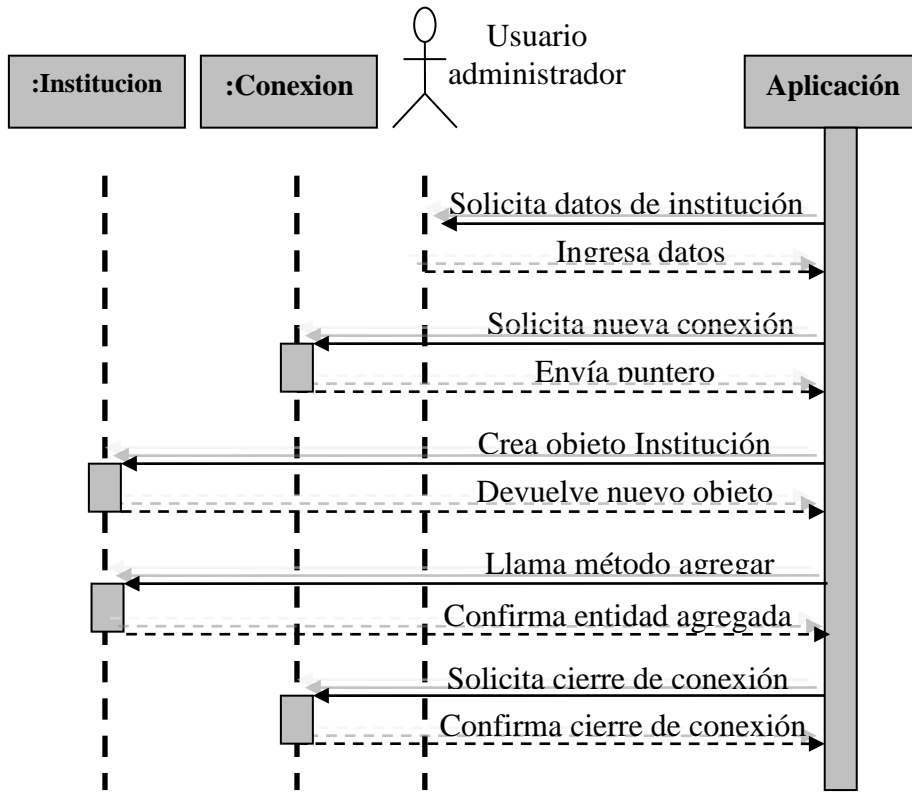


Figura 46: Diagrama 4 de Secuencia

Diagrama 5: Registrar un nuevo contacto al sistema

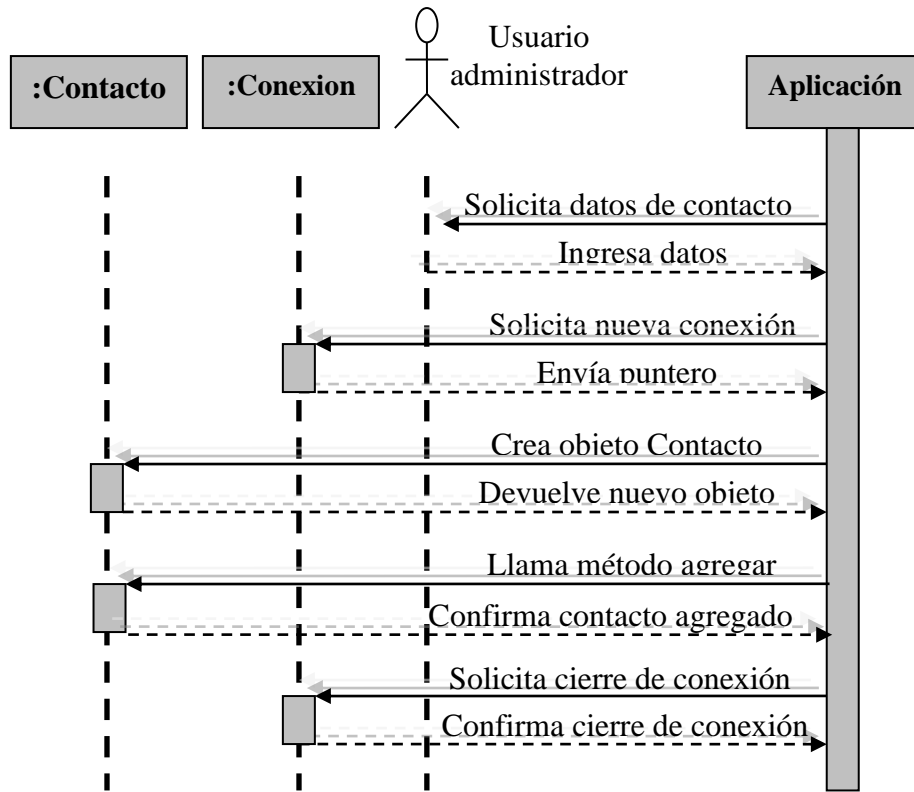


Figura 47: Diagrama 5 de Secuencia

Diagrama 6: Registrar una nueva pregunta al sistema

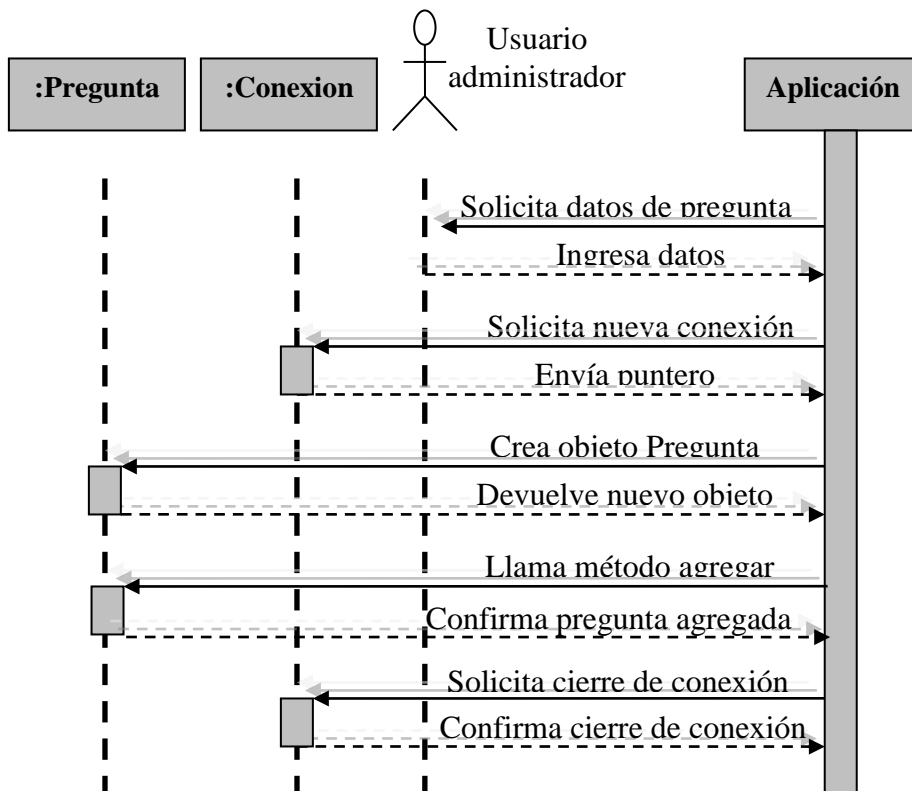


Figura 48: Diagrama 6 de Secuencia

Diagrama 7: Modificar un usuario ya registrado en el sistema

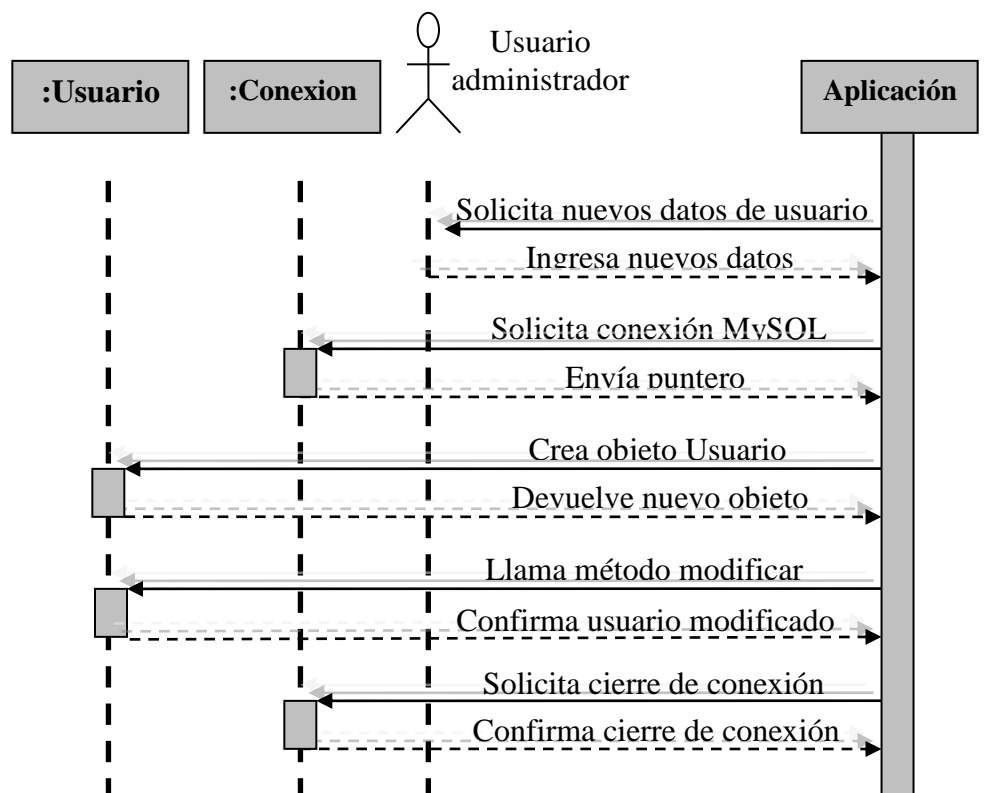


Figura 49: Diagrama 7 de Secuencia

Diagrama 8: Modificar un contacto ya registrado en el sistema

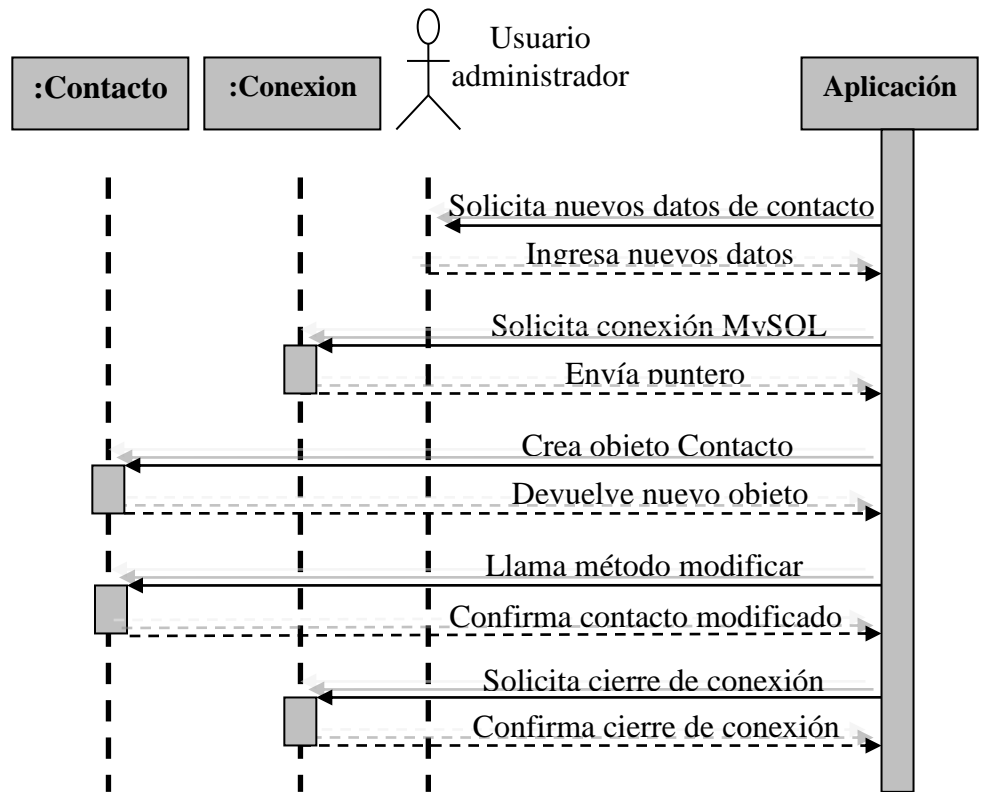


Figura 50: Diagrama 8 de Secuencia

Diagrama 9: Modificar una institución ya registrada en el sistema

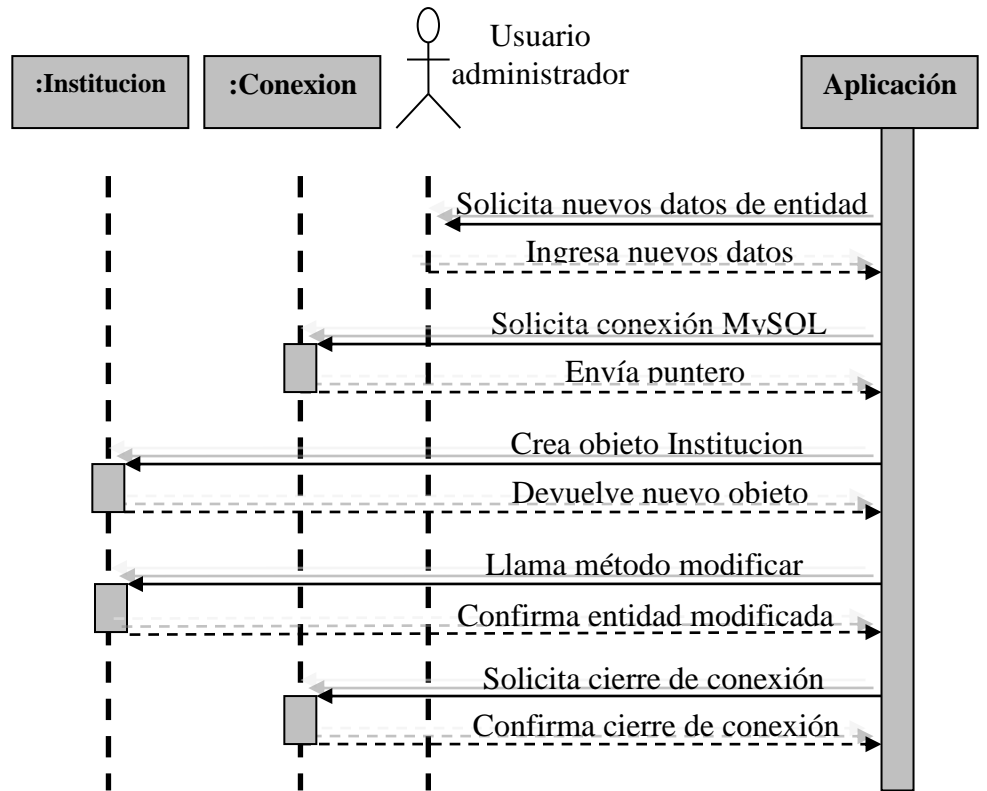


Figura 51: Diagrama 9 de Secuencia

Diagrama 10: Eliminar un usuario ya registrado en el sistema

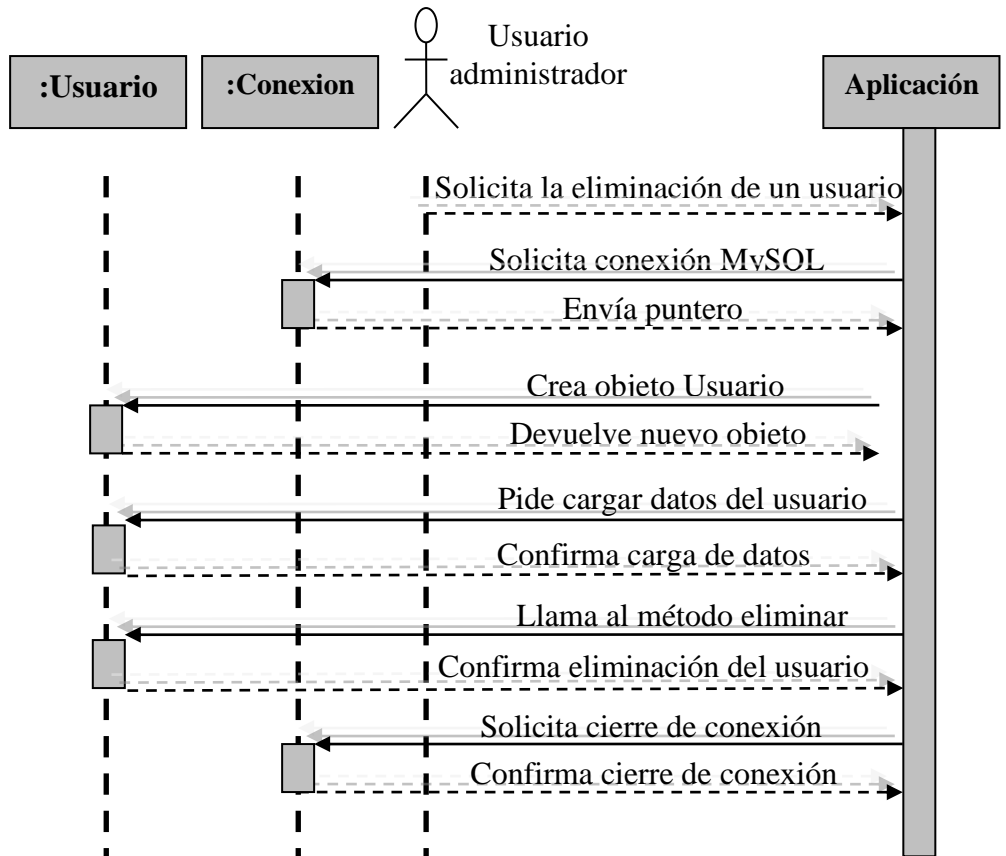


Figura 52: Diagrama 10 de Secuencia

Diagrama 11: Eliminar un contacto ya registrado en el sistema

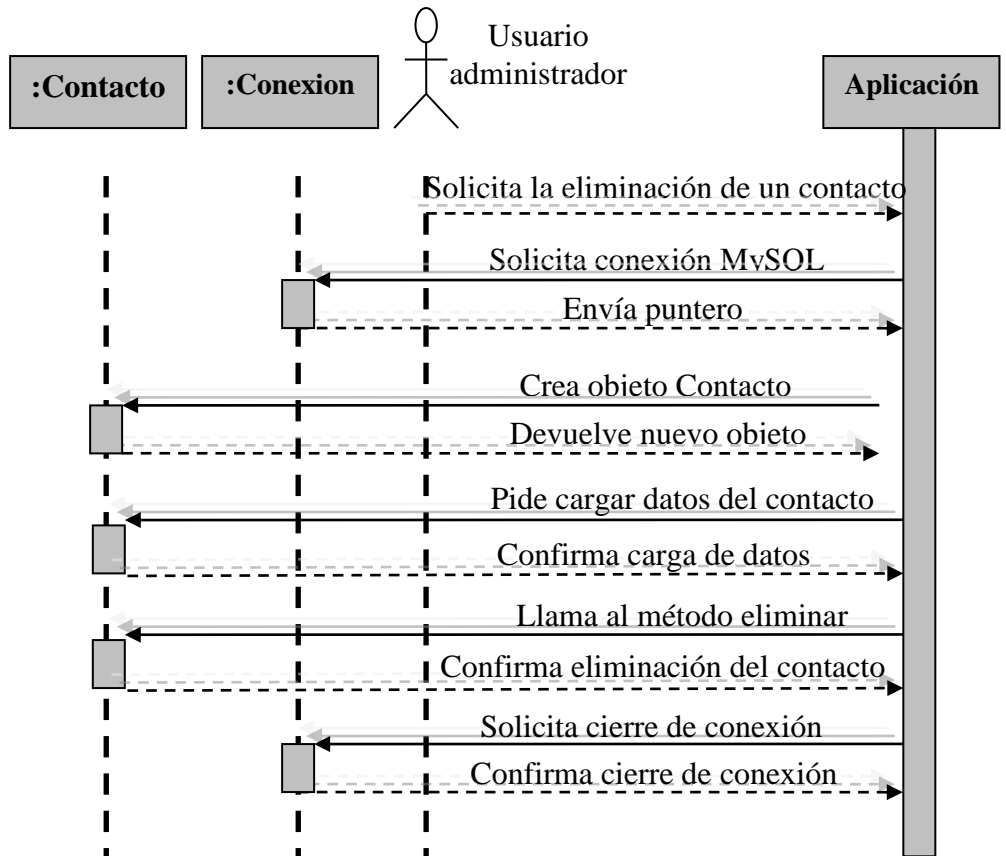


Figura 53: Diagrama 11 de Secuencia

3.1.3.4 Diagramas de Actividad

Los diagramas de actividad representan el flujo de procedimientos que se da cuando un usuario lleva a cabo una función específica de la aplicación, tales como dar de alta un usuario, eliminar un contacto.

Diagrama 1: Diagrama para la actividad de mantenimiento

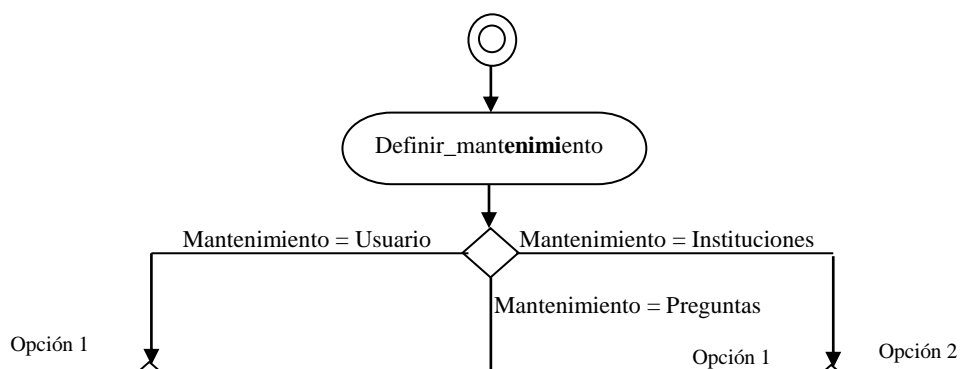
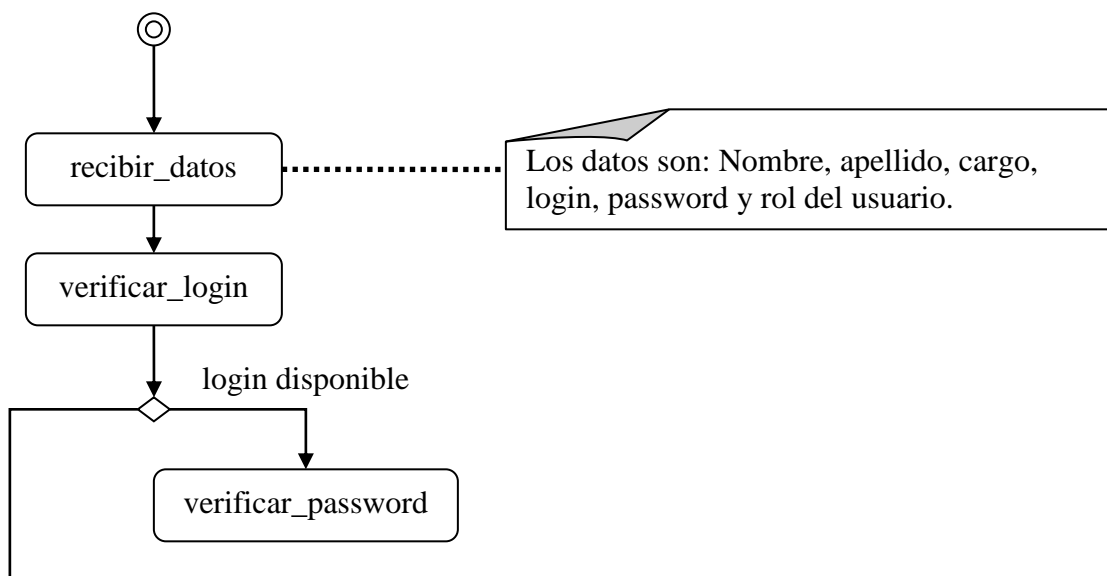


Diagrama 2: Diagrama para la actividad de agregar usuarios



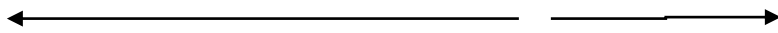
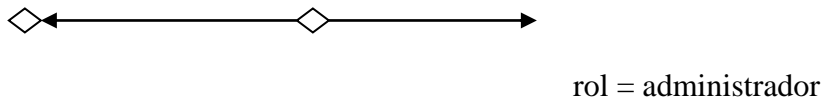
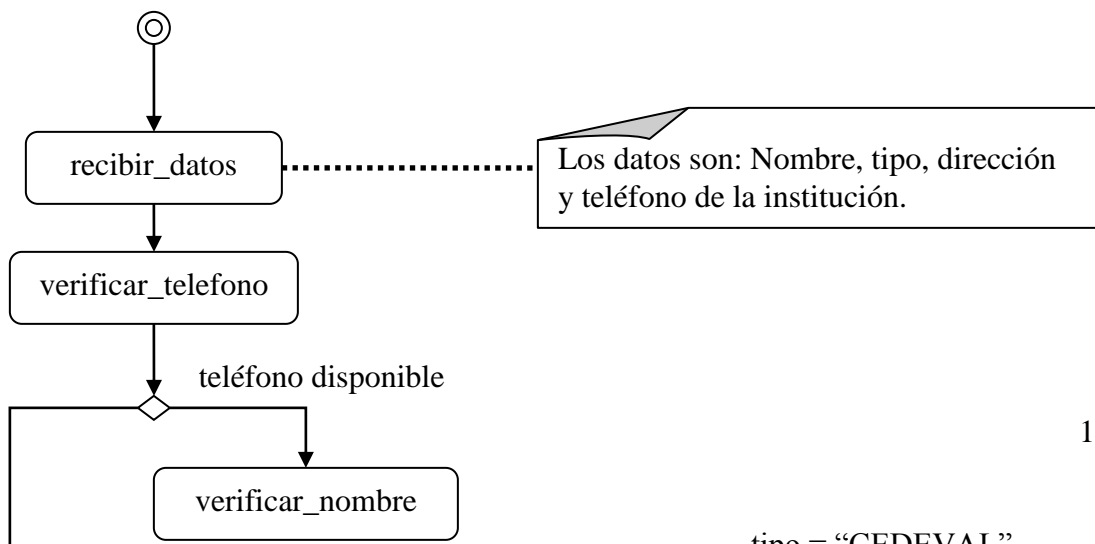


Figura 55: Diagrama 2 de Actividad



Diagrama 3: Diagrama para la actividad de agregar instituciones



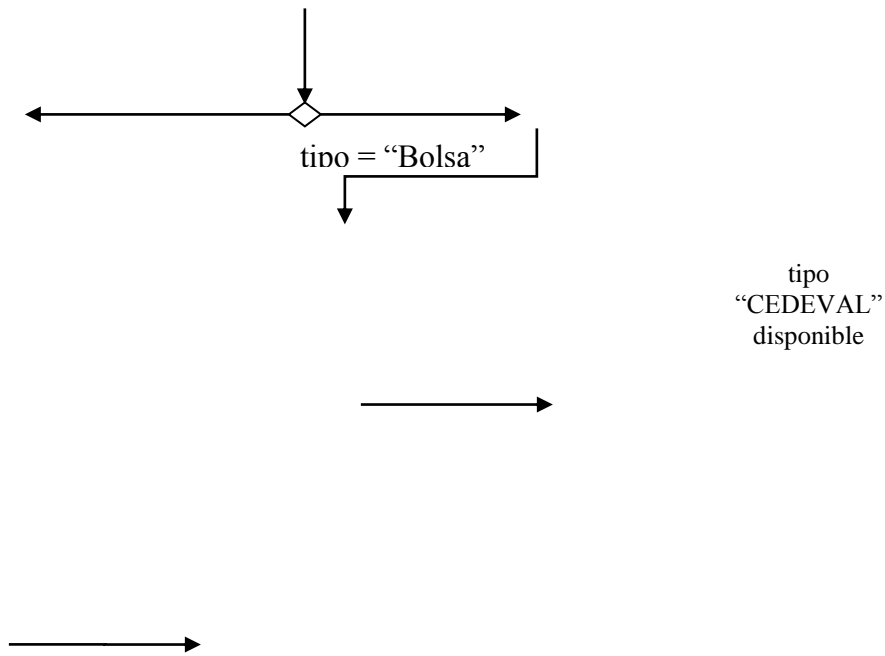
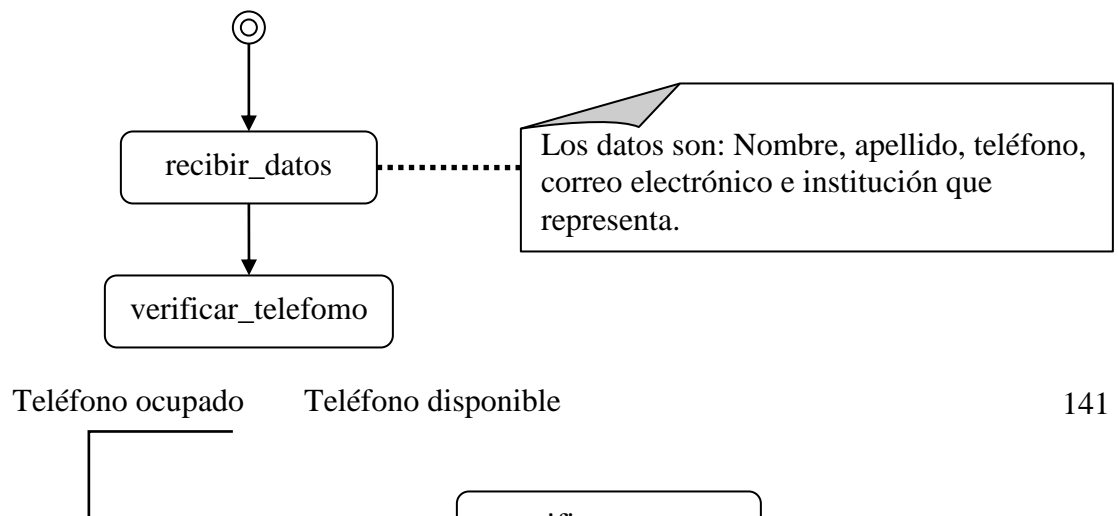


Figura 56: Diagrama 3 de Actividad

Diagrama 4: Diagrama para la actividad de agregar contactos



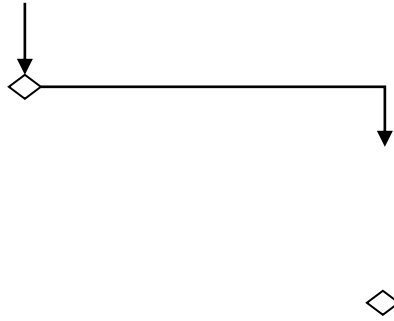
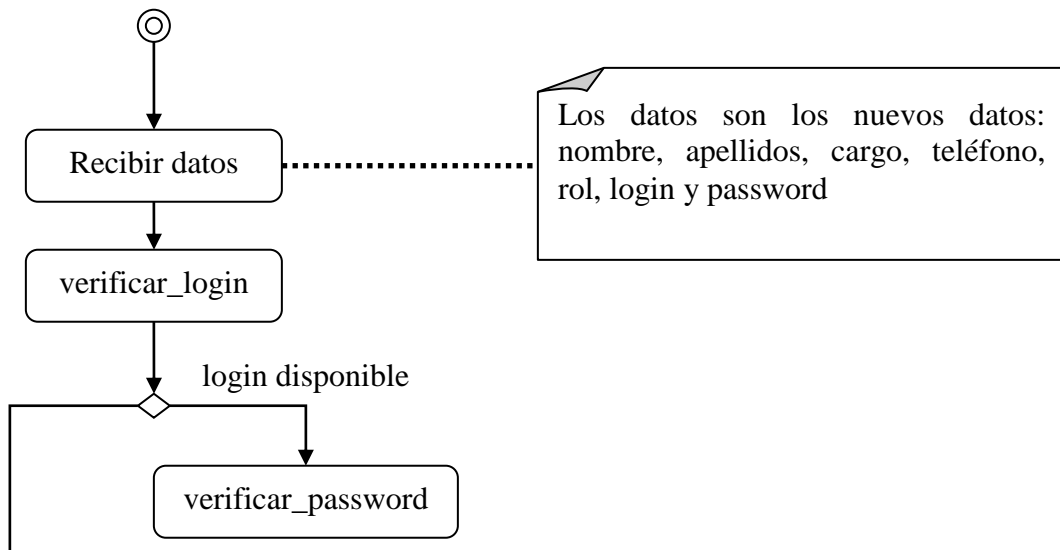


Figura 57: Diagrama 4 de Actividad

Diagrama 5: Diagrama para la actividad de modificar contactos



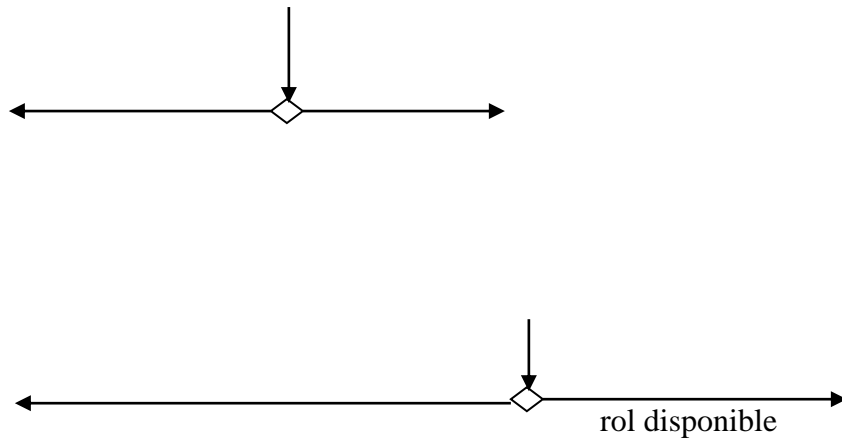
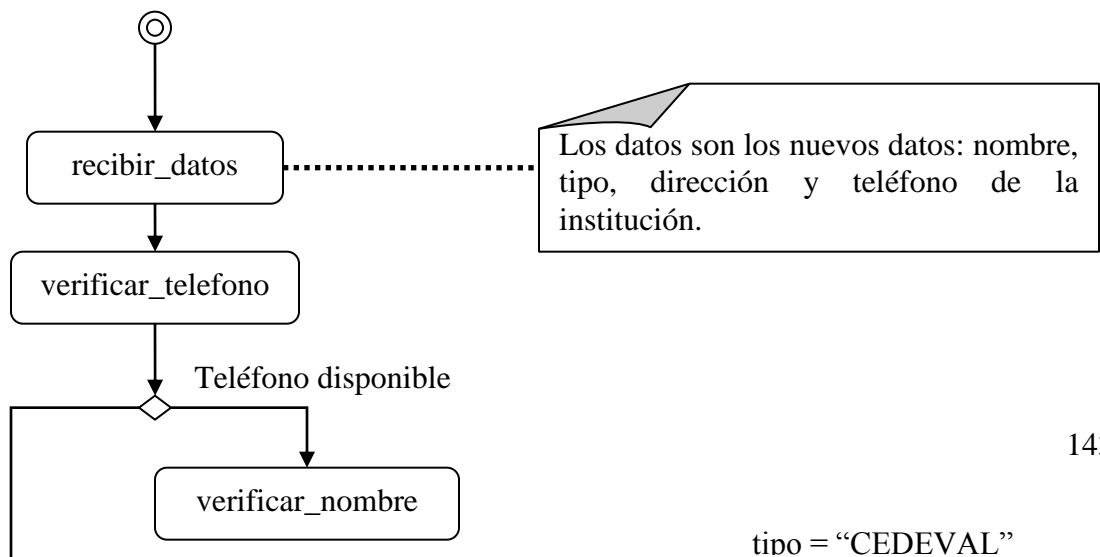


Figura 58: Diagrama 5 de Actividad

Diagrama 6: Diagrama para la actividad de modificar instituciones



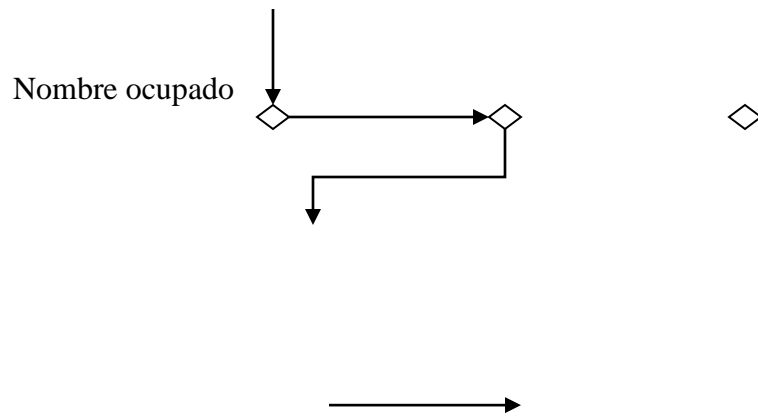


Figura 59: Diagrama 6 de Actividad

Diagrama 7: Diagrama para la actividad de modificar usuarios

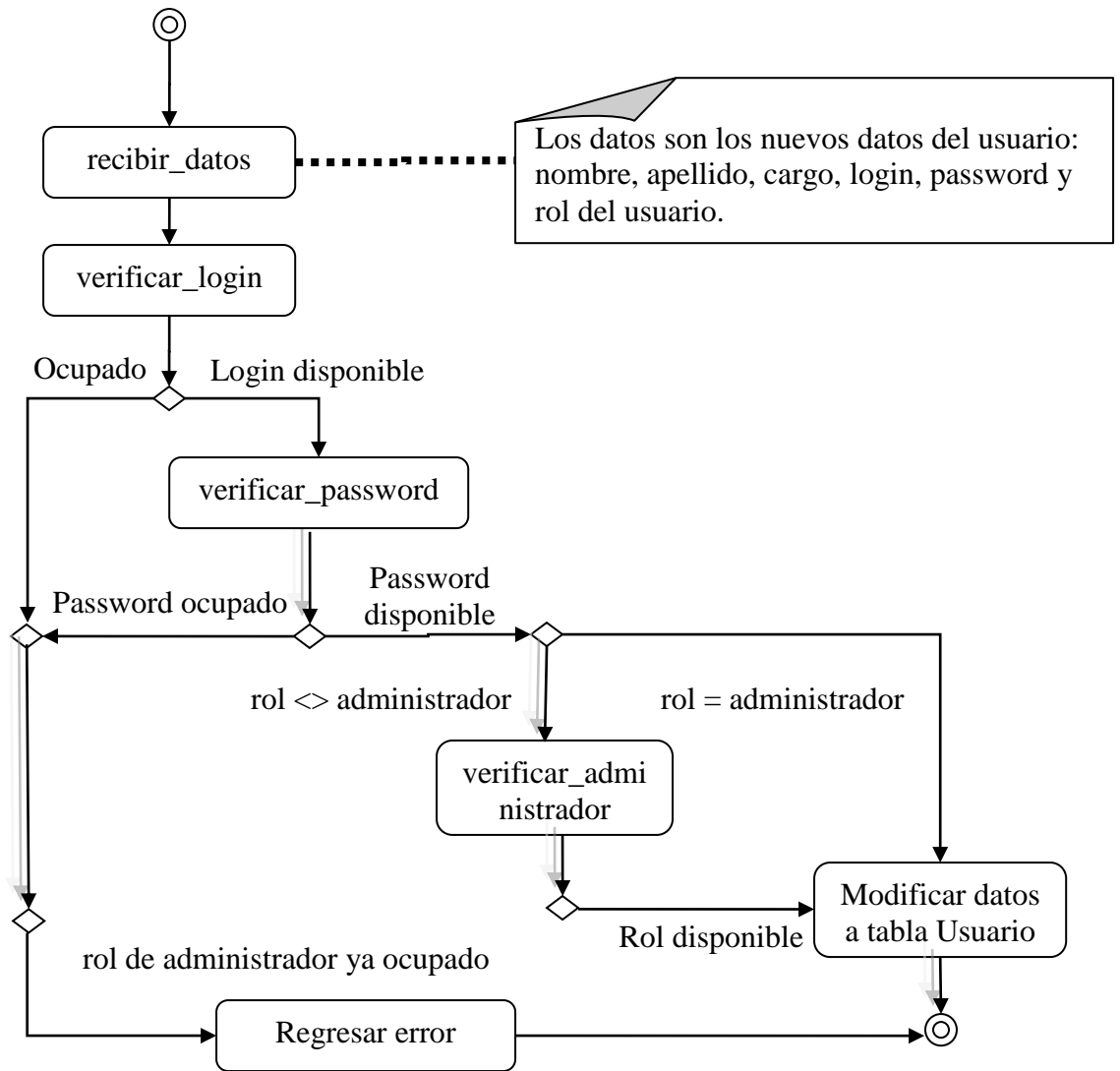


Figura 60: Diagrama 7 de Actividad

Diagrama 8: Diagrama para la actividad de eliminar usuarios

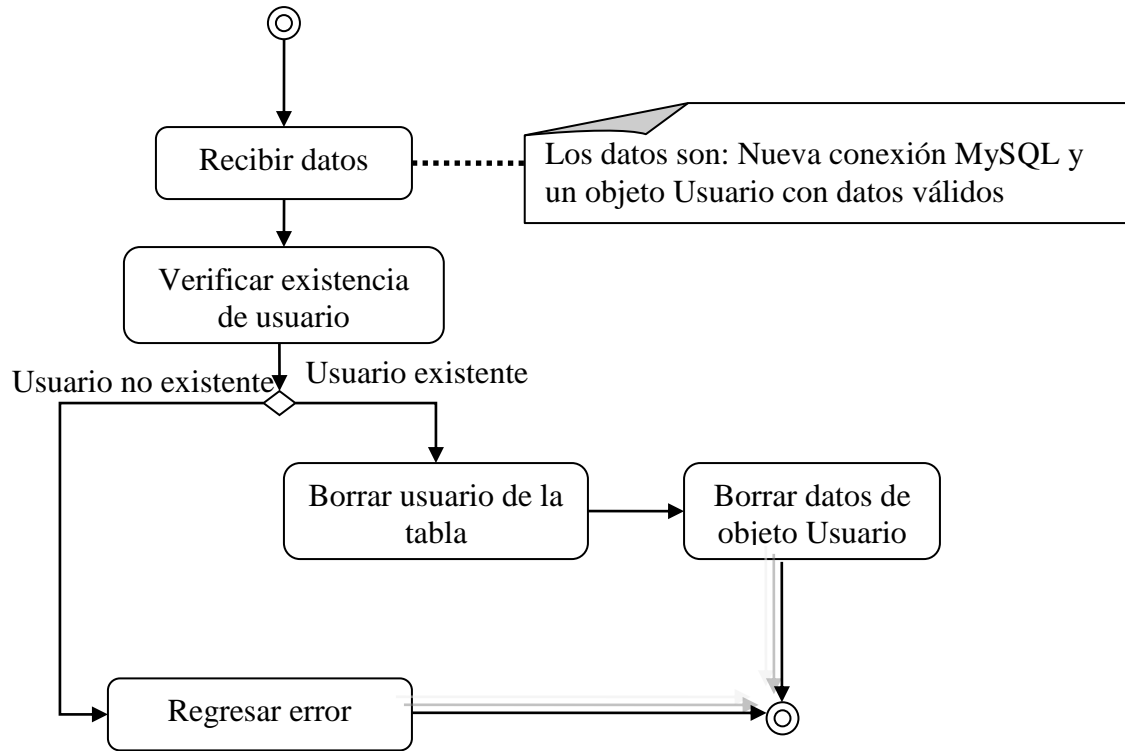
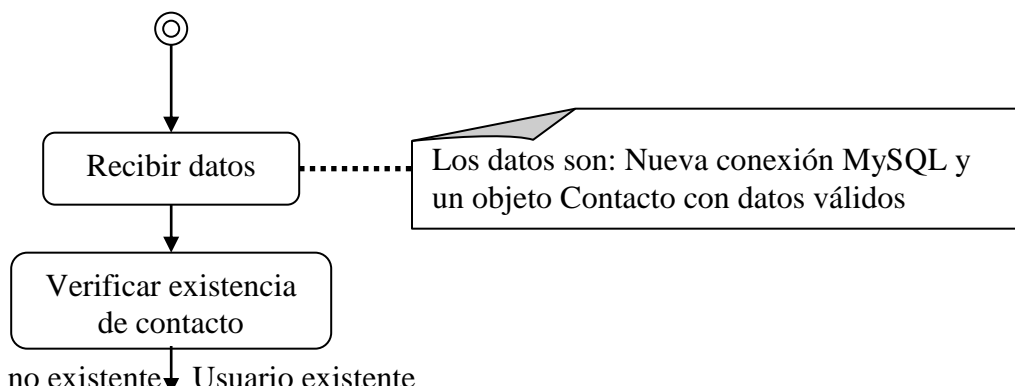


Figura 61: Diagrama 8 de Actividad

Diagrama 9: Diagrama para la actividad de eliminar contactos



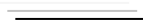


Figura 62: Diagrama 9 de Actividad

3.1.3.5 Diagramas de Colaboración

Los diagramas de colaboración esquematizan el uso que una clase hace de otra. Este uso puede darse en forma de invocación de métodos de una clase a otra. En los diagramas de colaboración, la clase que llama apunta una flecha a la clase invocada. Sobre la flecha se especifican cuáles son los métodos requeridos por la clase llamadora.

Diagrama 1: Colaboración para la clase Usuario

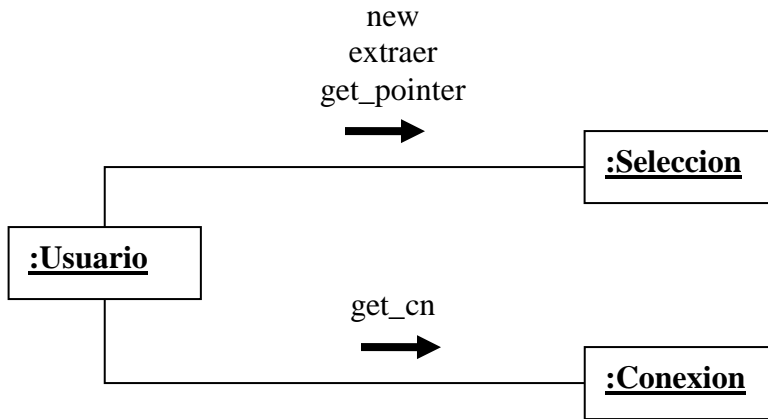


Figura 63: Diagrama 1 de Colaboración

Diagrama 2: Colaboración para la clase Institucion

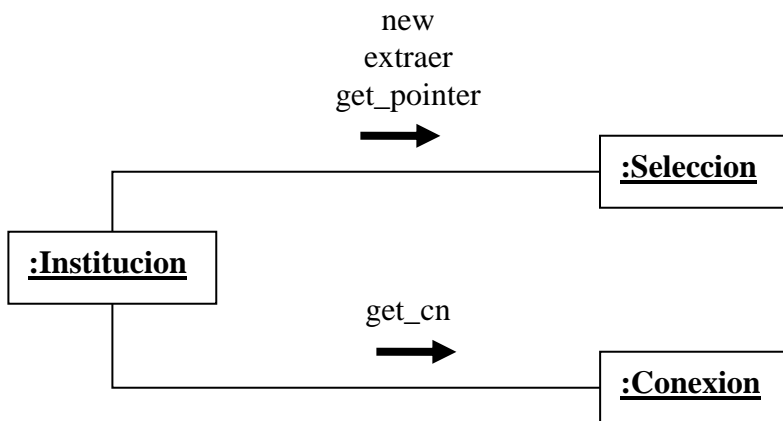


Figura 64: Diagrama 2 de Colaboración

Diagrama 3: Colaboración para la clase Contacto

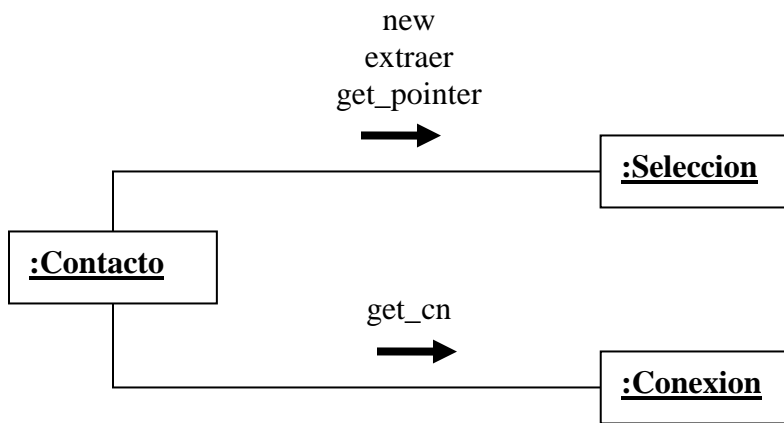


Figura 65: Diagrama 3 de Colaboración

Diagrama 4: Colaboración para la clase Pregunta

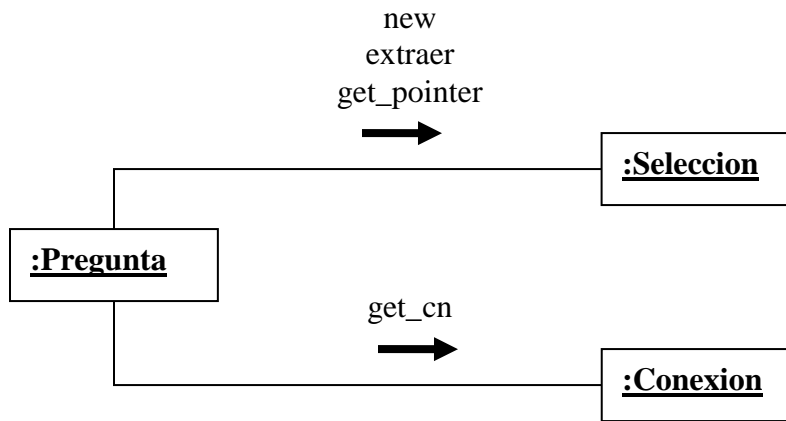


Figura 66: Diagrama 4 de Colaboración

Diagrama 5: Colaboración para la clase PapelTrabajo

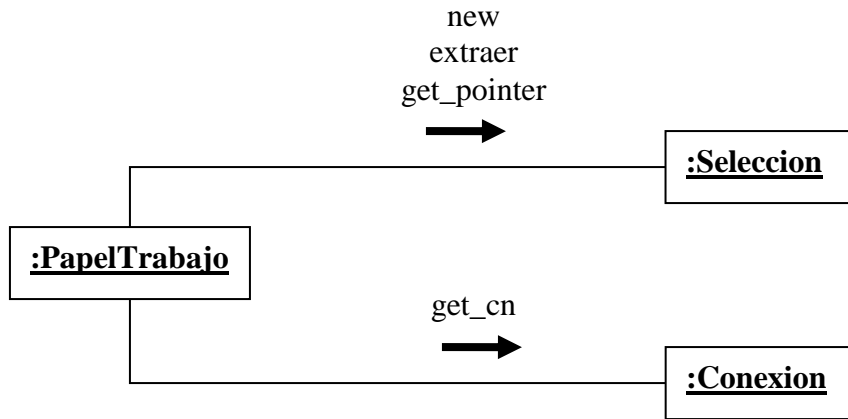


Figura 67: Diagrama 5 de Colaboración

Diagrama 6: Colaboración para la clase Preg_papel

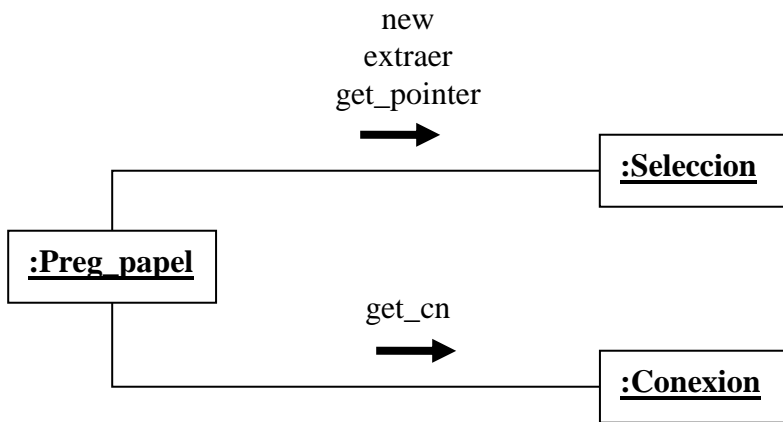


Figura 68: Diagrama 6 de Colaboración

Diagrama 7: Colaboración para la clase Seleccion

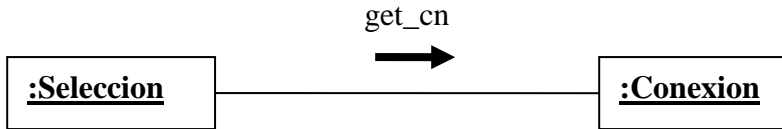


Figura 69: Diagrama 7 de Colaboración

Diagrama 8: Colaboración para la clase Preg_tipo

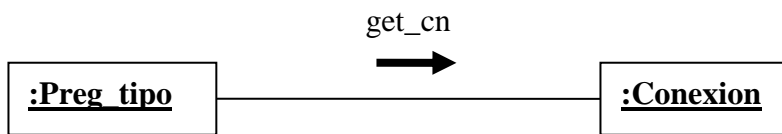


Figura 70: Diagrama 8 de Colaboración

Diagrama 9: Colaboración para la clase Preg_plantilla

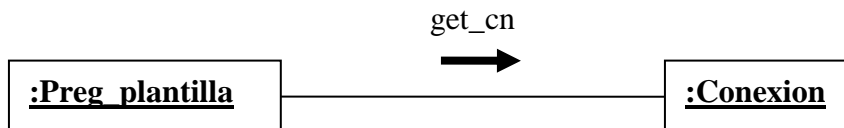


Figura 71: Diagrama 9 de Colaboración

3.2 CONSTRUCCIÓN DE LA APLICACIÓN

3.2.1 Raíz del sitio

El sistema contará con un sitio de entrada denominada *página raíz*. Esta página lleva por nombre *index.php* y está colocada en la carpeta *sassv* que contiene la aplicación. Cada vez que se ingresa a la dirección *http://localhost/sassv* se carga la página raíz.

La función de esta página es redireccionar el sitio a la versión más adecuada a la resolución de pantalla 1024 x 768 u 800 x 600.

3.2.2 Página de autenticación

Después de pasar por la página raíz, se carga la *página de autenticación*, la cual consta de dos cuadros de texto para recibir el login y el password del usuario. La página redirecciona a la página *menuPrincipal.php* donde se muestra el menú de la aplicación

3.2.3 Estilos del sitio y recursos gráficos

La apariencia del sitio en cuanto a colores, márgenes, tipografía se manejan en el archivo *estilos.css* que está en la carpeta *estilos*. En este archivo se usan etiquetas de Hojas de Estilo en Cascada. Este archivo es llamado en cada página del sitio que requiera de estilos con la siguiente sentencia de inclusión:

```
<link rel="stylesheet" href="estilos/estilos1.css" />
```

Todas las imágenes ocupadas en el sitio se almacenan en la carpeta *imágenes*.

3.2.4 Manejo de sesiones

En el sistema se hace uso de la capacidad de PHP para manejar sesiones. Esto con el fin de evitar accesos al software sin autenticación previa. Para el uso de dichas sesiones se incluyen al comienzo de cada página PHP los comandos siguientes:

- *session_start()*: Para dar inicio a una nueva sesión e incluir la página actual en una sesión existente.
- *sesión_register()*: Para crear una nueva variable de sesión.
- *sesión_name()*: Para obtener el nombre de la sesión actual.
- *sesión_id()*: Para obtener la cadena identificadora de la sesión actual.
- *sesión_is_registered()*: Para comprobar si una variable de sesión ya ha sido registrada como tal.
- *sesión_destroy()*: Para dar por finalizada una sesión y salir del sistema.

Además, en cada hipervínculo de las páginas del sistema y en cada formulario se pasan a las páginas destino el nombre y el identificador de sesión. De esa forma cada página de la aplicación queda enlazada en una sola sesión.

3.2.5 Organización del contenido

Las páginas que trabajarán propiamente con la información de la base de datos, su mantenimiento, la creación de papeles, etc. Se organizan en dos directorios: La carpeta *mantenimiento* y la carpeta *papeles*. En la primera se listan las páginas encargadas de dar mantenimiento a la base de datos y la segunda contienen el código de mantenimiento de los papeles de trabajo de las diferentes instituciones. Tanto el directorio *mantenimiento* como el de *papeles* son subcarpetas de *páginas*.

3.2.6 Creación e inclusión de clases

Para hacer uso de las clases diseñadas para la aplicación, cada página del sitio que necesitará instanciar dichas clases incluyen una línea que invoca el archivo *clases.php*, el cual está en la carpeta *clases* del sitio. En este archivo se definen una por una todas las clases que se diseñaron en el apartado **3.1.2 Determinación de clases**. La línea en cuestión es la siguiente:

```
require("clases/clases.php");
```

3.2.7 Manejo de la programación del lado del cliente

Para incluir porciones de código JavaScript que realicen actividades tales como el cambio de color en los menús y el redireccionamiento automático, se incluyen dos archivos llamados *scripts1.js* y *scripts2.js* en cada página que utilicen dichos códigos. Este archivo de scripts, que se encuentra en la carpeta *scripts*, incluye el código de las funciones JavaScript y es llamado en cada página a través de la siguiente línea:

```
<script language="javascript" src="scripts/scripts1.js"></script>
```

3.2.8 Estructura del sitio

La figura 71 muestra el diagrama de árbol de los directorios que conforman la aplicación. El listado de las páginas que contiene cada carpeta así como el papel que desempeña cada una en el funcionamiento global del software se explicará en el *manual de mantenimiento* de la aplicación.

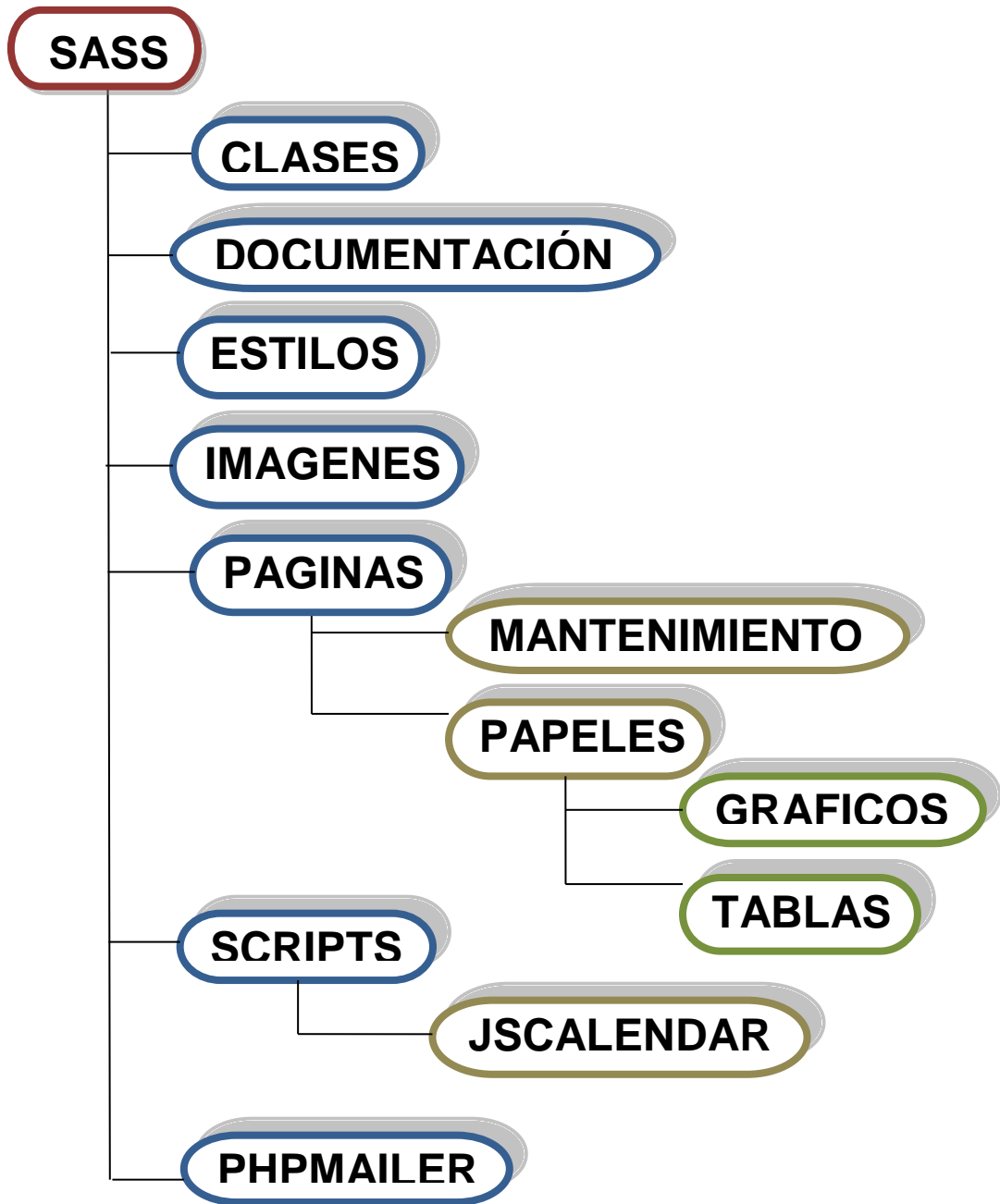


Figura 72: Estructura de Directorios de la Aplicación Web

CAPITULO IV

DOCUMENTACION

DEL

SISTEMA

Manual de Usuario



SAS-SV

Software de Auditoría de Sistemas para la Superintendencia de Valores



San Salvador, Noviembre de 2007

Sumario de Contenido

1. Ingreso al sistema
2. Funcionamiento del sistema.
 - 2.1 Menú Principal
 - 2.2 Opciones del menú de papeles de trabajo
 - 2.2.1 Proceso para la creación de un nuevo papel de trabajo
 - 2.2.2 Cédulas de auditoría de sistemas
 - 2.2.2.1 Cédula de planeación
 - 2.2.2.2 Cédula sumaria de hardware
 - 2.2.2.3 Cédula sumaria de software
 - 2.2.2.4 Cédula sumaria de políticas y procedimientos
 - 2.2.2.5 Cédula sumaria de seguridad
 - 2.2.2.6 Cédula de hallazgos
 - 2.2.2.7 Cédula de marcas
 - 2.2.3 Abrir papel
 - 2.2.4 Cerrar papel
 - 2.2.5 Mis plantillas
 - 2.2.5.1 Proceso para crear una nueva plantilla
 - 2.3 Opciones del menú de mantenimiento
 - 2.3.1 Mantenimiento de usuarios
 - 2.3.2 Mantenimiento de instituciones
 - 2.3.3 Mantenimiento de preguntas
 - 2.3.3.1 Proceso para el ingreso de preguntas
 - 2.3.4 Copia de seguridad
 - 2.3.4.1 Proceso para realizar un respaldo
 - 2.3.4.2 Proceso para recuperación de información
 - 2.3.5 Estadísticas
 - 2.4 Salida del sistema
- 3 Generación de reportes
 - 3.1 Proceso para generar reportes

1. INGRESO AL SISTEMA



SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

UNIVERSIDAD DON BOSCO
VALORES INDEPENDENCIA RESPETO

Superintendencia de Valores

UNIVERSIDAD DON BOSCO
VALORES INDEPENDENCIA RESPETO

Login:

Password:

Ingresar

Imagen 1: Validación de Usuario y Clave

En esta pantalla es donde el usuario se autentica para poder ingresar al menú principal del sistema, introduciendo su password con un máximo de tres intentos, en caso que estos sean fallidos el usuario se bloquea automáticamente y solo el administrador del sistema tendrá la opción de activarlo nuevamente.

2. FUNCIONAMIENTO DEL SISTEMA

2.1 MENÚ PRINCIPAL



Imagen 2: Menú principal del SAS-SV

Este es el menú principal del sistema, las opciones de este se describen de la siguiente forma:

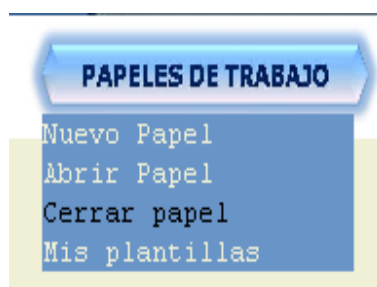


Imagen 3: Submenú de la opción Papeles de trabajo

En este menú del sistema es donde el usuario podrá crear, modificar y consultar los papeles de trabajo, esta compuesto por cuatro opciones las cuales son: crear los papeles de trabajo, abrir papel, cerrar un papel y mis plantillas.

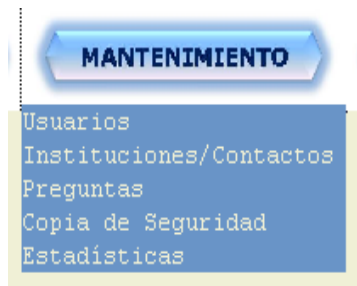


Imagen 4: Submenú de la opción Mantenimiento

Este menú es donde el administrador del sistema podrá darle mantenimiento a la base de datos de la aplicación. Con esta opción el usuario podrá adicionar una nueva institución, un nuevo usuario o nuevas preguntas para las plantillas que contendrá el sistema.



Imagen 5: opción de ayuda del sistema

Esta opción de menú es el que contendrá el manual de usuario en línea, el cual servirá para que el usuario lo consulte en caso de tener algún inconveniente en el manejo del software.



Imagen 6: opción salida del sistema

Con esta opción el usuario podrá salir del sistema.

2.2 OPCIONES DEL MENU DE PAPELES DE TRABAJO

Nuevo Papel:

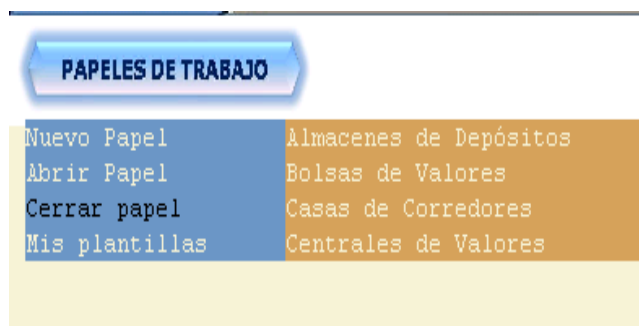


Imagen 7: Opciones del submenú de Papeles de trabajo

Con esta opción de menú el usuario creará los papeles de trabajo, eligiendo uno de los cuatro tipos de instituciones existentes.

2.2.1 PROCESO PARA CREACIÓN DE UN NUEVO PAPEL DE TRABAJO

Paso 1:



The screenshot shows the 'Nuevo Papel de Trabajo para Almacenes' screen. At the top, there is a header with the SAS-SV logo and the text 'SOFTWARE DE APOYO PARA LA AUDITORIA DE SISTEMAS SUPERINTENDENCIA DE VALORES'. Below the header, there are navigation buttons: 'PAPELES DE TRABAJO', 'MANTENIMIENTO', 'AYUDA', and 'SALIR'. The main content area has the title 'Nuevo Papel de Trabajo para Almacenes'. There are three input fields: 'Haga clic en el nombre de la Entidad:' with a dropdown menu showing 'ALDESA S A de CV' and 'ALMACONSA S A de CV'; 'Escoja una plantilla de preguntas:' with a dropdown menu showing 'GeneralAlmacenes'; and 'Escriba un nombre para el papel:' with a text input field containing 'Observaciones para este papel:'. At the bottom, there is a 'Crear papel' button. The date and time 'Fecha: 16-12-2007 Hora: 0:20:37' are displayed in the top right corner.

Imagen 8: Pantalla de creación de nuevo papel de trabajo

En esta pantalla es donde se elige la entidad a la que se le va a generar un papel de trabajo, además es aquí donde se le asigna nombre al papel de trabajo que se va a crear.

Paso 2:



Imagen 9: Pantalla de cedulas de auditoría

En esta pantalla es donde se le da mantenimiento a los papeles de trabajo que se han creado. Usando las distintas cedulas de auditoría que posee el sistema, las cuales son las siguientes: Sumaria de planeación, sumaria de hardware, sumaria software, sumaria de políticas y procedimientos, sumaria de seguridad, sumaria de hallazgos, marcas.

Otra funcionalidad de esta pantalla es que al papel abierto actualmente se le puede ver/cambiar datos y cerrar.

2.2.2 CEDULAS DE AUDITORIA DEL SISTEMA.

2.2.2.1 PLANEACION



Imagen 10: Opción de cedula de planeación

Con la opción “Planeación” del menú lateral izquierdo se pueden acceder a dos opciones:



Imagen 11: Opciones contenidas en la cedula de planeación

CEDULA DE PLANEACION

Esta opción Cédula de planeación, le permite acceder a la generación de la planificación del desarrollo de la auditoría.

La pantalla que se presenta es la siguiente:

The screenshot displays the SAS-SV software interface. At the top, it reads "SAS-SV SOFTWARE DE APOYO PARA LA AUDITORIA DE SISTEMAS SUPERINTENDENCIA DE VALORES". Below this is a navigation bar with buttons for "PAPELES DE TRABAJO", "MANTENIMIENTO", "AYUDA", and "SALIR". The date and time are shown as "Fecha: 16-12-2007" and "Hora: 0:25:31".

The main area is titled "Cédula de Planeación" and "Scotia Inversiones - Auditoria a Scotia Inversiones". It features a table with three columns: "Contenidos", "Procedimientos", and "Sub-Procedimientos".

Contenidos	Procedimientos	Sub-Procedimientos
Antecedentes de la casa	Descripción del Origen de la operat. de la Casa	Lic. Julio Lopez
Personal clave para el desarrollo de la auditoria		Ing. Martin Perez

Below the table, there is a form for adding new content, labeled "Escriba el nuevo contenido:", with an "Agregar" button and a "Cancelar" button.

On the left side, there is a vertical menu with buttons for "CEDULAS", "PLANEACION", "HARDWARE", "SOFTWARE", "POLITICAS Y PROCEDIMIENTOS", "SEGURIDAD", "HALLAZGOS", "MARCAS", "PAPEL ACTUAL", "VER / CAMBIAR DATOS", and "CERRAR PAPEL".

Imagen 12: Ingreso de datos para la cedula de planeación

Aquí se presentan los contenidos de la programación, cada contenido, tiene un conjunto de procedimientos asociados y cada procedimiento puede contener uno o más subprocedimientos.

Para ver los procedimientos de un contenido:

Haga clic en cualquiera de los apartados de la columna contenido y podrá ver sus procedimientos.

Para ver los subprocedimientos de un procedimiento:

Haga clic en el procedimiento de la tabla del cual quiere ver sus subprocedimientos.

Para ver el contenido de la planeación:

Cada contenido aparece numerado en un orden secuencial, por lo que para ver la información de clic en ellos. Dicha información aparecerá en la parte baja de la pantalla.

Para poder ingresar los procedimientos o subprocedimientos:

Haga clic en el botón de radio del texto al cual desea adicionarle información.

Luego haga clic en el botón para ingresar los datos y aparecerá el siguiente formulario.

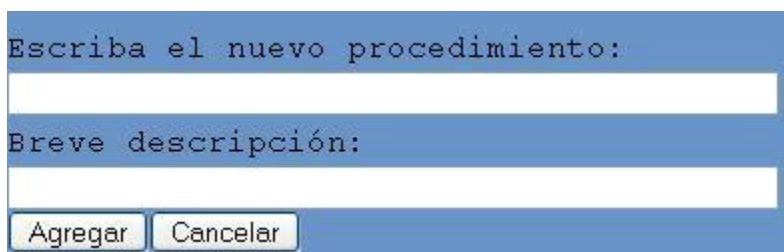
El formulario tiene un fondo azul claro. En la parte superior, el texto "Escriba el nuevo procedimiento:" está en un color gris oscuro. Debajo de este texto hay un campo de entrada de texto blanco. Luego, el texto "Breve descripción:" también en gris oscuro, precede a otro campo de entrada de texto blanco. En la parte inferior del formulario, hay dos botones rectangulares con bordes azules: "Agregar" a la izquierda y "Cancelar" a la derecha.

Imagen 13: Pantalla para adicionar datos en la cedula de planeación

Una vez ingresado el nuevo dato, haga clic en el botón agregar.

NOTA DE REQUERIMIENTOS

La opción Nota de Requerimientos, le permite mantener una ficha de datos con los requerimientos solicitados por el auditor previa realización de la auditoría o en el desarrollo de la auditoría.

Dando clic a la opción de Nota de requerimientos el sistema mostrará la siguiente pantalla donde el usuario ingresa los datos del destinatario de la nota.

SAS-SV
SOFTWARE DE APOYO PARA LA AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

PAPELES DE TRABAJO MANTENIMIENTO AYUDA SALIR

Fecha: 16-12-2007
Hora: 0:58:58

Cédula de Planeación

Scotia Inversiones - Auditoria a Scotia Inversiones

Nota de Requerimientos

CÉDULAS

- PLANEACION
- HARDWARE
- SOFTWARE
- POLITICAS Y PROCEDIMIENTOS
- SEGURIDAD
- HALLAZGOS
- MARCAS
- PAPEL ACTUAL
- VER / CAMBIAR DATOS
- CERRAR PAPEL

Nombre del destinatario:

Cargo del destinatario:

Requerimiento	Descripción	Entregado por	Fecha de Recibido	Recibido
Documentación técnica	Manual de usuario		0000-00-00	<input type="checkbox"/>
Documentación técnica	Diccionario de datos	Paricia Perez	2007-12-12	<input type="checkbox"/>

Ingresar Generar nota

Imagen 16: Pantalla de ingreso de datos para la nota de requerimientos

En la tabla de contenido aparecen sucesivamente los datos de:

Requerimiento: El nombre del recurso que se solicita.


Breve descripción del requerimiento.

El nombre de la persona que lo entrega

La fecha en que se recibió.

Para ingresar un nuevo requerimiento:

Dar clic en el botón ingresar y aparecerá un formulario como el que se muestra en la siguiente figura.



The image shows a web form with a blue background. It contains four text input fields stacked vertically. The labels for these fields are: 'Nombre del Requerimiento', 'Descripción del Requerimiento', 'Entregado por', and 'Fecha de recibido'. The first three fields are full-width, while the fourth is narrower. To the right of the 'Fecha de recibido' field is a small button with three dots. Below the input fields is a larger button labeled 'Aplicar'.

Imagen 19: Pantalla para ingresar información de la nota de requerimientos

Una vez agregado el nuevo dato, haga clic en el botón aplicar.

2.2.2.2 CEDULA SUMARIA DE HARDWARE

La funcionalidad de esta cedula es evaluar los aspectos relacionados con los equipos informáticos que posee cada entidad fiscalizada.

SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

UNIVERSIDAD DEL MERCADO
VALORES

PAPELES DE TRABAJO MANTENIMIENTO AYUDA SALIR

Fecha: 16-12-2007
Hora: 11:29:9

Cédula de Hardware

ALMACONSA S A de C V - Auditoria a Almaconsa

CEDULAS							
No.	Pregunta	Respuestas				Referencias	Observaciones
1	Solicite el inventario del Hardware que pertenece a la entidad.	<input type="radio"/> Si	<input type="radio"/> No	<input type="radio"/> Incompleto	<input type="radio"/> N/A	-	-
2	pregunta nueva	<input type="radio"/> Si	<input type="radio"/> No	<input type="radio"/> Incompleto	<input type="radio"/> N/A	-	-

Imagen 20: Cedula sumaria de hardware

2.2.2.3 CEDULA SUMARIA DE SOFTWARE

Con esta cedula se van evaluar los aspectos relacionados con todo el software instalado en las estaciones de trabajo que posee cada entidad fiscalizada.

No.	Pregunta	Respuestas	Referencias	Observaciones
1	Solicite y revise los diagramas Entidad-Relación de los sistemas de información que utilice la entidad.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A	-	-
2	Solicite y revise el manual de usuario de los sistemas de información que utilice la entidad.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A	-	-
3	Solicite y revise el diccionario de datos de los sistemas informáticos de la entidad.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A	-	-
4	Solicite y revise que la descripción de los módulos de los sistemas de información sean de conformidad al manual contable de Almacenes Generales de Depósito.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A	-	-
5	Solicite y revise el proceso para generar reportes.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A	-	-
6	Solicite acceso a la base de datos y verifique la actualización de las	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A	-	-

Imagen 21: Cedula sumaria de software

2.2.2.4 CEDULA SUMARIA DE POLITICAS Y PROCEDIMIENTOS

Con esta cedula se va a evaluar la documentación referente a los diversos procesos que utiliza cada entidad fiscalizada para la operatividad de los sistemas y su cumplimiento.

Cédula de Políticas y Procedimientos

ALMACONSA S A de C V - Auditoria a Almaconsa

No.	Pregunta	Respuestas	Referencias	Observaciones
1	Solicite y revise que exista un plan de trabajo para la implementación de los sistemas informáticos.	<input type="radio"/> Si <input type="radio"/> No <input checked="" type="radio"/> Incompleto <input type="radio"/> N/A	--	--
2	Solicite y revise los procesos de flujo de información.	<input type="radio"/> Si <input type="radio"/> No <input checked="" type="radio"/> Incompleto <input type="radio"/> N/A	--	--
3	Verifique que la ejecución de las operaciones estén registradas en la base de datos según los horarios autorizados por la Junta Directiva de la entidad.	<input type="radio"/> Si <input type="radio"/> No <input checked="" type="radio"/> Incompleto <input type="radio"/> N/A	--	--

Imagen 22: Cedula sumaria de políticas y procedimientos

2.2.2.5 CEDULA SUMARIA DE SEGURIDAD

Con esta cedula se va a evaluar los aspectos relacionados con los estandares de seguridad implementados por cada entidad fiscalizada.

SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

Fecha: 16-12-2007
Hora: 1:36:2

Cédula de Seguridad

ALMACONSA S A de C V - Auditoria a Almaconsa

No.	Pregunta	Respuestas	Referencias	Observaciones
1	Solicite el programa de seguridad de la entidad y verifique si está actualizado.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A		
2	Verifique si la entidad tiene un plan de contingencia anti-desastres.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A		
3	Verifique las políticas de administración de las cuentas de usuario de las bases de datos.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A		
4	Verifique si existen políticas de administración de contraseñas de usuario.	<input type="radio"/> Si <input type="radio"/> No <input type="radio"/> Incompleto <input type="radio"/> N/A		

Imagen 23: Cedula sumaria de seguridad

2.2.2.6 CEDULA DE HALLAZGOS

Esta cedula es la que contendrá los reportes emitidos por el software desarrollado, en el literal número tres del presente documento se detalla cada uno de estos reportes.

SAS-SV
SOFTWARE DE APOYO PARA LA AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

UNIVERSIDAD DON BOSCO
VALPARAISO, AGENCIA, 1980

PAPELES DE TRABAJO MANTENIMIENTO AYUDA SALIR

Fecha: 16-12-2007
Hora: 1:36:2

Cedula de Hallazgos

Scotia Inversiones - Auditoria a Scotia Inversiones

CÉDULAS

- PLANEACION
- HARDWARE
- SOFTWARE
- POLITICAS Y PROCEDIMIENTOS
- SEGURIDAD
- HALLAZGOS**
- MARCAS
- PAPEL ACTUAL
- VER / CAMBIAR DATOS
- CERRAR PAPEL

Generación de Reportes

Encabezado Acta [Editar...]
Firmantes del acta [Modificar...]
Elaborador del informe [Modificar...]
Revisor del informe [Modificar...]
Encabezado de nota [Editar...]
Asunto de nota [Editar...]
Introducción de nota [Editar...]
Conclusión de nota [Editar...]
Pie de nota [Editar...]

Criterio:

[Guardar]

Texto de la pregunta	Respuesta	Observación	Area auditada	Criterio	Causa	Efecto	Comentario	Incluir en el acta
Solicite el inventario del Hardware que pertenece a la entidad.	Incompleto	La información de inventario del HW no está completa.	[Editar]	[Editar]	[Editar]	[Editar]	[Editar]	<input type="checkbox"/>
Solicite y revise los diagramas Entidad-Relación de los sistemas de información que utilice la entidad.	No cumple	Diagrama E-R está desactualizado.	[Editar]	[Editar]	[Editar]	[Editar]	[Editar]	<input checked="" type="checkbox"/>
Solicite y revise que exista un plan de trabajo para la implementación de los sistemas informáticos.	No cumple	No existe documentación sobre plan de trabajo para la implementación de los sistemas.	[Editar]	[Editar]	[Editar]	[Editar]	[Editar]	<input checked="" type="checkbox"/>

[Guardar Datos] [Acta de Auditoria] [Informe de Hallazgos] [Nota de Auditoria] [Enviar por correo]

Imagen 24: Cedula de Hallazgos

2.2.2.7 CEDULA DE MARCAS

En esta cédula se hace referencia de toda la simbología utilizada en el software. Se ingresa a esta cédula haciendo clic al botón que se muestra en la figura 25.



Imagen 25: Cédula de marcas

2.2.3 ABRIR PAPEL



Imagen 25: Pantalla para abrir un papel de trabajo

Esta opción sirve para abrir un papel de trabajo existente, es decir que al seleccionar esta opción, el usuario podrá consultar cualquier papel de trabajo.

2.2.4 CERRAR PAPEL

Con esta opción el usuario podrá salirse del papel de trabajo en el que esté trabajando.



Imagen 26: Opción para cerrar un papel de trabajo

2.2.5 MIS PLANTILLAS

La imagen muestra la interfaz de usuario del software SAS-SV. En la parte superior, hay una barra de navegación con el logo de la Superintendencia de Valores a la izquierda y el logo de la Universidad del Bío-Bío a la derecha. El título principal "SAS-SV" está en rojo, seguido de "SOFTWARE DE APOYO PARA LA AUDITORIA DE SISTEMAS" y "SUPERINTENDENCIA DE VALORES" en azul. Debajo de la barra de navegación hay cuatro botones: "PAPELES DE TRABAJO", "MANTENIMIENTO", "AYUDA" y "SALIR". En la esquina superior derecha, se muestra la fecha "Fecha: 16-12-2007" y la hora "Hora: 1:57:13". El contenido principal de la pantalla muestra el título "PLANTILLAS DEL USUARIO" en rojo. Debajo de este título, se indica "Se encontraron 1 plantillas registradas para el usuario actual." y un enlace "Crear una nueva plantilla". Abajo de esto hay una tabla con dos columnas: "Número" y "Nombre".

Número	Nombre
1	Plantilla nueva

Imagen 27: Pantalla para consultar plantillas de preguntas

Esta opción es la que sirve para elegir las preguntas que contendrá el programa de auditoria a desarrollar.

2.2.5.1 PROCESO PARA CREAR UNA NUEVA PLANTILLA

PASO 1:



Imagen 28: Pantalla para creación de plantillas de preguntas

Se le da clic a la opción “**Mis plantillas**” y se entra a la pantalla que se muestra en la figura anterior que se llama “**Plantillas del usuario**”, en esta pantalla se pueden ver las plantillas creadas por el usuario actual.

Para crear una nueva plantilla el usuario da un clic a la opción “**Crear una nueva plantilla**” y luego seleccionará las preguntas que contendrá la nueva plantilla.

PASO 2:

SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

UNIVERSIDAD DEL BÍO-BÍO
VALDIVIA 1990

Fecha: 16-12-2007
Hora: 0:22:53

PAPELES DE TRABAJO MANTENIMIENTO AYUDA SALIR

Crear una nueva plantilla

Nombre para la nueva plantilla:

Seleccione las preguntas a incluir y haga clic en Grabar Plantilla

Grabar Plantilla

CÉDULA DE PLANIFICACIÓN		
Número	Texto Pregunta	Seleccionada
1	se planean las inversiones	<input type="checkbox"/>
2	se precisan los informes	<input type="checkbox"/>
3	asfdasf	<input type="checkbox"/>
4	esto es un ejemplo	<input type="checkbox"/>
5	esto es otro ejemplo	<input type="checkbox"/>
6	cosas nuevas	<input type="checkbox"/>

CÉDULA DE HARDWARE		
Número	Texto Pregunta	Seleccionada
1	Solicite el inventario del Hardware que pertenece a la entidad.	<input type="checkbox"/>
2	pregunta nueva	<input type="checkbox"/>

Imagen 29: Pantalla para almacenar una nueva plantilla

Habiendo entrado a la pantalla “**crear nueva plantilla**”, se eligen las preguntas que va a contener la plantilla, para ello se le da clic en la casilla que aparece en la parte de derecha de cada pregunta, es decir en la columna con nombre “seleccionada”.

Después de haber elegido las preguntas se le da clic a la opción **gravar plantilla** para que esta quede almacenada en la base de datos.

2.3 OPCIONES DEL MENU DE MANTENIMIENTO

2.3.1 MANTENIMIENTO DE USUARIOS



Imagen 30: Submenú de la opción de Mantenimiento



Imagen 31: Pantalla para crear un nuevo usuario al SAS - SV

Esta opción de menú sirve para crear los usuarios del sistema, opción que solamente estará activa para el administrador del sistema.

Al hacer clic en el icono de la columna **Ver Historial** se accede a la pantalla de bitacora del sistema, la cual se muestra a continuación:

Bitacora de rpineda1

Todas las entradas Fecha especifica:

Actividad realizada	Fecha	Hora
Entrada al sistema	2007-12-16	13:11:25
Entrada al sistema	2007-12-16	13:14:12
Crear nuevo papel de trabajo	2007-12-16	13:15:22
Abrir papel de trabajo	2007-12-16	13:16:12
Consultar plantillas creadas	2007-12-16	13:16:56
Agregar nueva plantilla	2007-12-16	13:17:44
Crear nuevo papel de trabajo	2007-12-16	13:18:34
Trabajar con la cedula de planificaci♦n	2007-12-16	13:19:13
Trabajar con la cedula de planificaci♦n	2007-12-16	13:19:43
Salir del sistema	2007-12-16	13:19:50
Entrada al sistema	2007-12-16	13:19:58
Abrir papel de trabajo	2007-12-16	13:20:01
Trabajar con la cedula de planificaci♦n	2007-12-16	13:20:05

Imagen 32: Pantalla para consultar la bitacora del sistema

2.3.2 MANTENIMIENTO DE INSTITUCIONES

Usuario: rpineda1
Fecha: 2/9/2007

- Usuarios
- Instituciones/Contactos**
- Preguntas

Imagen 33: Submenú de la opción Mantenimiento

Imagen 34: Pantalla para crear una nueva institución en el SAS - SV

Con esta opción del menú se ingresan los datos generales de las entidades que estarán registradas en el sistema.

2.3.3 MANTENIMIENTO DE PREGUNTAS

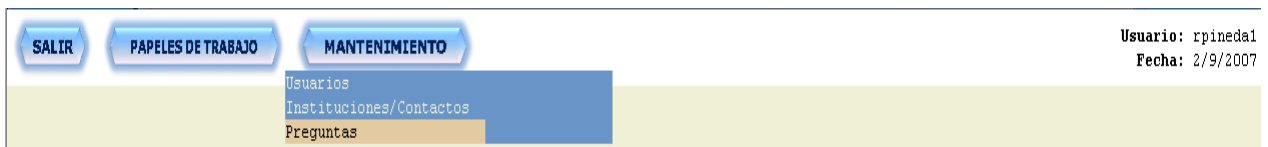


Imagen 35: Submenú de la opción Mantenimiento

Para adicionar nuevas preguntas a los programas de auditoria el administrador del sistema deberá ingresar al menú de **“mantenimiento”**, luego debe elegir la opción **“Preguntas”**, inmediatamente el sistema mostrará la pantalla que se muestra en la siguiente figura.

SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

Fecha: 16-12-2007
Hora: 0:54:56

MANTENIMIENTO DE PREGUNTAS

[Agregar una pregunta nueva](#)
CÉDULA DE PLANIFICACIÓN

Número	Texto Pregunta	Entidades que aplican			
		Almacenes	Bolsa de Valores	Casas Corredoras	CEDEVAL
1	se planean las inversiones	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	se precisan los informes	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	asfdasdf	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	esto es un ejemplo	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	esto es otro ejemplo	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	cosas nuevas	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

CÉDULA DE HARDWARE

Número	Texto Pregunta	Entidades que aplican			
		Almacenes	Bolsa de Valores	Casas Corredoras	CEDEVAL
1	Solicite el inventario del Hardware que pertenece a la entidad.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Imagen 36: Pantalla para una nueva pregunta al SAS - SV

En esta pantalla el administrador del sistema podrá verificar todas las preguntas que contiene el sistema, luego para adicionar una nueva se le debe dar clic a la opción de **“agregar una pregunta nueva”**.

2.3.3.1 PROCESO PARA EL INGRESO DE PREGUNTAS

The screenshot shows the SAS-SV software interface. At the top, there is a header with the logo of the Superintendencia de Valores (SV) on the left and the logo of the Universidad del Banco on the right. The main title in the center reads "SAS-SV SOFTWARE DE APOYO PARA LA AUDITORIA DE SISTEMAS SUPERINTENDENCIA DE VALORES". Below the header, there are four navigation buttons: "PAPELES DE TRABAJO", "MANTENIMIENTO", "AYUDA", and "SALIR". On the right side, there is a date and time display: "Fecha: 16-12-2007 Hora: 0:55:23". The main content area is titled "INGRESO DE NUEVA PREGUNTA" and contains a form with the following fields:

Texto de la Pregunta:	Cédula a la que pertenece: <input type="radio"/> Planificación <input type="radio"/> Hardware <input type="radio"/> Software <input type="radio"/> Políticas y Procedimientos <input type="radio"/> Seguridad	Entidades a la que aplica: <input type="checkbox"/> Almacenes <input type="checkbox"/> Bolsa de Valores <input type="checkbox"/> Casas Corredoras <input type="checkbox"/> CEDEVAL
	Categoría:	
<input type="button" value="Agregar pregunta"/>		

Imagen 37: Pantalla para editar el texto de la nueva pregunta

En esta pantalla el administrador del sistema podrá ingresar el texto de la nueva pregunta a registrar, para ello se deberá seguir los siguientes pasos:

PASO 1: En el cuadro “**Texto de la pregunta**”, digite el enunciado que contendrá la pregunta a ingresar.

PASO 2: En la opción “**Cedula a la que pertenece**”, seleccione el nombre de la cedula a la que pertenecerá la pregunta digitada.

PASO 3: En la opción “**entidades a la que aplica**”, seleccione la entidad a la que aplica la pregunta que se desea ingresar.

PASO 4: Después de haber realizado los pasos 1,2 y 3 se da clic en la opción “**Agregar pregunta**” para guardar la información digitada, luego el sistema mostrará un mensaje de confirmación el cual le indica al usuario que la pregunta ha sido agregada.

2.3.4 COPIA DE SEGURIDAD

Con esta opción se podrán realizar respaldos de la base de datos y recuperación de información, para ello en el menú principal se selecciona del menú de mantenimiento la opción “COPIA DE SEGURIDAD”, luego el sistema mostrará la pantalla que aparece en la siguiente figura.

The screenshot shows the main interface of the SAS-SV software. At the top, there is a header with the logo of the Superintendencia de Valores (SV) on the left and the logo of the Universidad Don Bosco (UB) on the right. The central text reads "SAS-SV SOFTWARE DE APOYO PARA LA AUDITORIA DE SISTEMAS SUPERINTENDENCIA DE VALORES". Below the header, there are four navigation buttons: "PAPELES DE TRABAJO", "MANTENIMIENTO", "AYUDA", and "SALIR". On the right side, the date and time are displayed as "Fecha: 16-12-2007" and "Hora: 0:56:47". The main content area is titled "COPIA DE SEGURIDAD DE LA BASE DE DATOS" and contains two sections: "RESPALDO" and "RECUPERACIÓN". Each section has a text input field for the file name, a "Browse..." button, and a "Respaldar" or "Recuperar" button.

Imagen 38: Pantalla para hacer una Copia de Seguridad

En esta pantalla el usuario tendrá la opción de realizar un respaldo de la base de datos o recuperar la información de un respaldo ya efectuado.

2.3.4.1 PROCESO PARA REALIZAR UN RESPALDO

PASO 1: Cree un archivo nuevo de texto en la zona de la computadora donde desea el respaldo.

PASO 2: De clic en el botón examinar del cuadro archivo de respaldo.

PASO 3: Seleccione el archivo creado en el paso 1.

NOTA: Los tres pasos anteriores pueden ser obviados, en cuyo caso se creará automáticamente el archivo respaldo.txt en la carpeta respaldos del directorio raíz de la aplicación.

PASO 4: De clic en el botón respaldar. Luego aparecerá un mensaje de confirmando la finalización del respaldo.

2.3.4.2 PROCESO PARA RECUPERACIÓN DE INFORMACIÓN

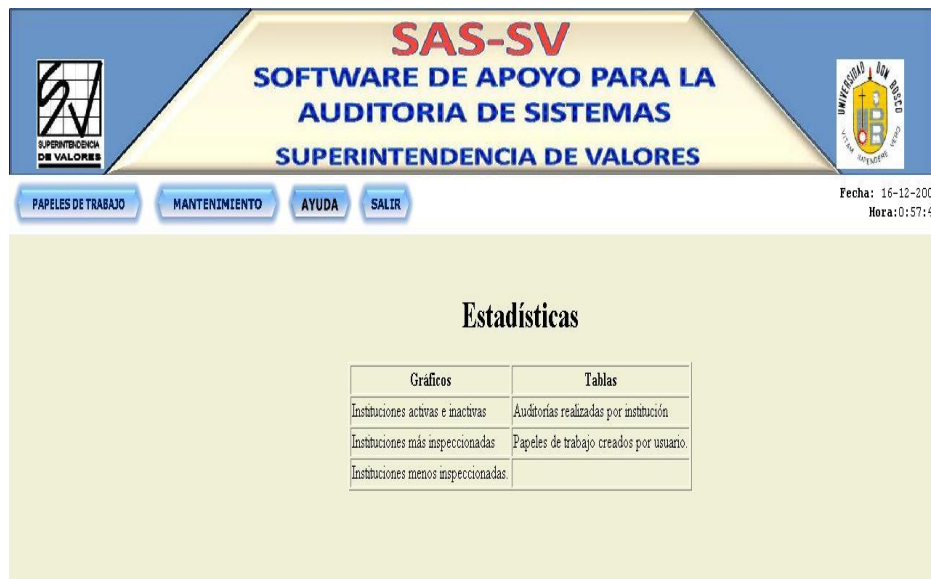
PASO 1: De clic en el botón examinar del cuadro archivo de recuperación.

PASO 2: Seleccione el archivo que contiene el respaldo de la base de datos.

PASO 3: De clic en el botón recuperar y luego aparecerá un mensaje de proceso de recuperación, al finalizar dicho proceso aparecerá otro mensaje confirmando la finalización de la recuperación de información.

2.3.5 ESTADÍSTICAS

En este menú el administrador del sistema podrá obtener datos estadísticos que le ayudaran en la toma de decisiones referente al desempeño de los auditores y al trabajo realizado a las diversas entidades.



SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

PAPELES DE TRABAJO MANTENIMIENTO AYUDA SALIR

Fecha: 16-12-2007
Hora: 0:57:49

Estadísticas

Gráficos	Tablas
Instituciones activas e inactivas	Auditorías realizadas por institución
Instituciones más inspeccionadas	Papeles de trabajo creados por usuario
Instituciones menos inspeccionadas	

Imagen 39: Pantalla para Consulta de Estadísticas

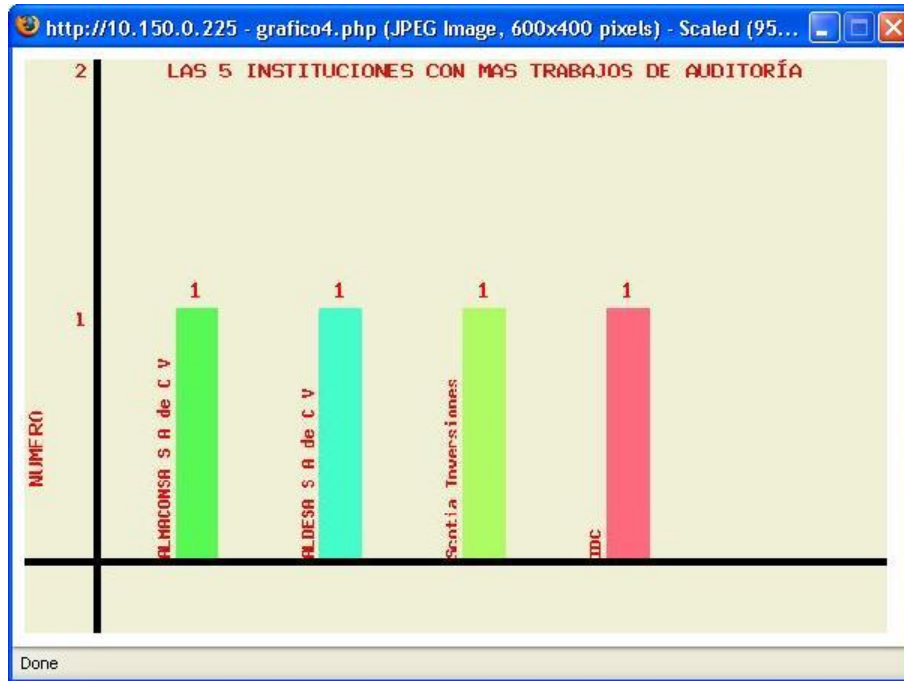


Imagen 40: Pantalla para mostrar grafico de Estadísticas

2.4 SALIDA DE SISTEMA

Con esta opción del menú el usuario podrá salirse completamente del sistema.

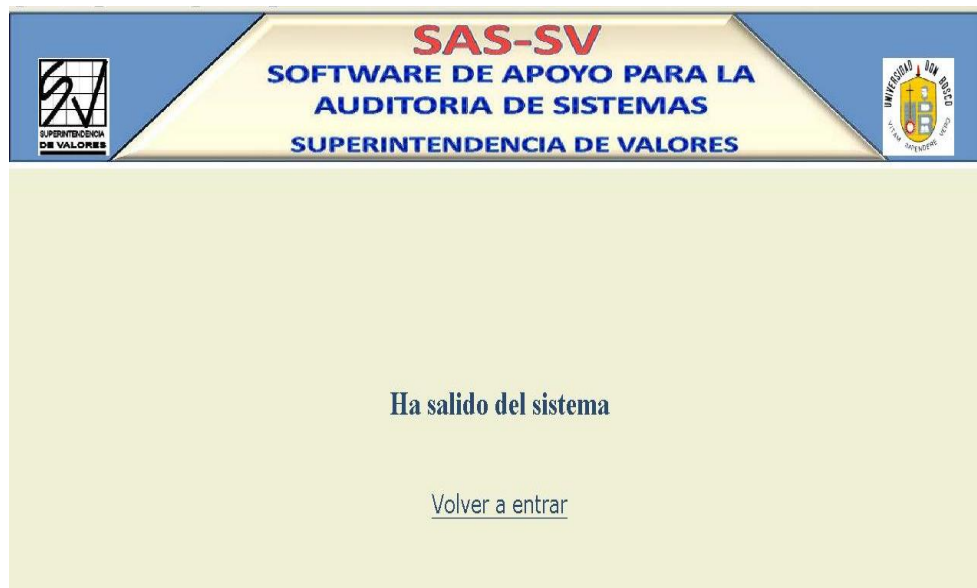


Imagen 41: Pantalla de confirmación de salida del sistema

3. GENERACIÓN DE REPORTES

Los reportes que el sistema va a generar son los siguientes: Acta de resultados de auditoría, informe de hallazgos y nota de resultados de auditoría.

3.1 PROCESO PARA GENERAR LOS REPORTES DEL SISTEMA.

Dando clic en la opción de HALLAZGOS aparecerá en la pantalla principal un cuadro como se muestra en la figura 41, donde se pueden modificar diversos aspectos específicos de cada uno de los reportes siendo estos los siguientes:

Encabezado del acta de auditoría

Firmantes del acta

Elaborador del informe

Revisor del informe

Encabezado de la nota de hallazgos

Asunto de la nota de hallazgos

Introducción de la nota de hallazgos

Conclusión de la nota

Pie de la nota

Para modificar cualquiera de estos aspectos, haga clic en el botón ubicado al lado derecho de cada uno de ellos.

SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

UNIVERSIDAD DON BOSCO
VICARIA ACADÉMICA

Fecha: 16-12-2007
Hora: 1:36:2

PAPALES DE TRABAJO MANTENIMIENTO AYUDA SALIR

Cedula de hallazgos

Scotia Inversiones - Auditoria a Scotia Inversiones

Generación de Reportes

CÉDULAS

- PLANEACION
- HARDWARE
- SOFTWARE
- POLITICAS Y PROCEDIMIENTOS
- SEGURIDAD
- HALLAZGOS
- MARCAS
- PAPEL ACTUAL**
- VER / CAMBIAR DATOS
- CERRAR PAPEL

Encabezado Acta	Editar...
Firmantes del acta	Modificar...
Elaborador del informe	Modificar...
Revisor del informe	Modificar...
Encabezado de nota	Editar...
Asunto de nota	Editar...
Introducción de nota	Editar...
Conclusión de nota	Editar...
Pie de nota	Editar...

Criterio:

Guardar

Texto de la pregunta	Respuesta	Observación	Area auditada	Criterio	Causa	Efecto	Comentario	Incluir en el acta
Solicite el inventario del Hardware que pertenece a la entidad.	Incompleto	La información de inventario del HW no está completa.	Editar	Editar	Editar	Editar	Editar	<input type="checkbox"/>
Solicite y revise los diagramas Entidad-Relación de los sistemas de información que utilice la entidad.	No cumple	Diagrama E-R está desactualizado.	Editar	Editar	Editar	Editar	Editar	<input checked="" type="checkbox"/>
Solicite y revise que exista un plan de trabajo para la implementación de los sistemas informáticos.	No cumple	No existe documentación sobre plan de trabajo para la implementación de los sistemas.	Editar	Editar	Editar	Editar	Editar	<input checked="" type="checkbox"/>

Imagen 42: Pantalla de cedula de hallazgos

Encabezado Acta	Editar...
Firmantes del acta	Modificar...
Elaborador del informe	Modificar...
Revisor del informe	Modificar...
Encabezado de nota	Editar...
Asunto de nota	Editar...
Introducción de nota	Editar...
Conclusión de nota	Editar...
Pie de nota	Editar...

Imagen 43: Pantalla de campos contenidos en la nota e informe de hallazgos

Después de dar clic al botón a la derecha de cada opción, aparecerá un formulario como se muestra en la figura 40, en el cual se podrá modificar la información respectiva.

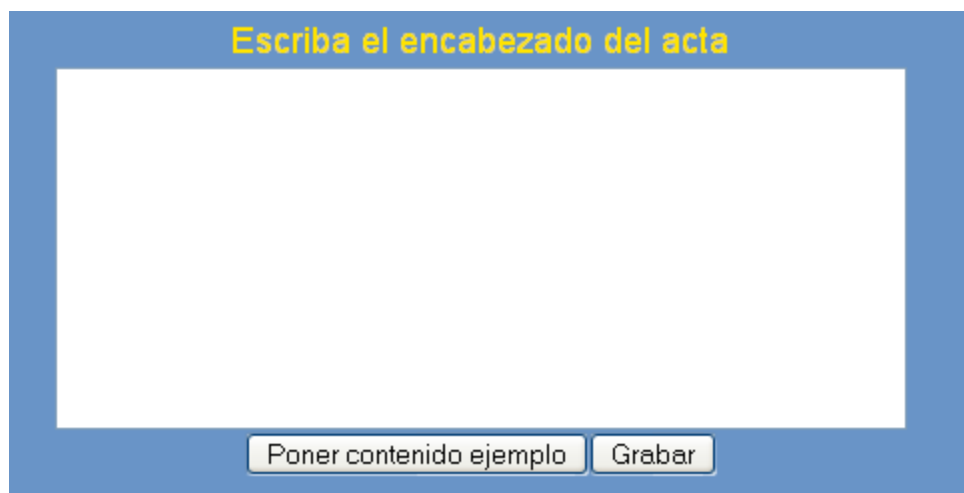


Imagen 44: Pantalla para editar los reportes del sistema SAS-SV

El botón “Poner contenido ejemplo”, carga en el área de edición un texto que puede servir como guía para escribir la información del campo.

Una vez editada la información, haga clic en el botón “Grabar”, para guardar los cambios.

Luego de rellenar cada uno de estos campos, se procede a construir la información de contenido de los reportes, esto se hace mediante la tabla que aparece a continuación, la cual reúne todos aquellos aspectos que fueron catalogados con respuestas “no cumple” o “incompleto” en los papeles de auditoría.

Texto de la pregunta	Respuesta	Observación	Area auditada	Criterio	Causa	Efecto	Comentario	Incluir en el acta
Solicite el inventario del Hardware que pertenece a la entidad.	Incompleto	La información de Inventario del HW no está completa.	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="checkbox"/>
Solicite y revise los diagramas Entidad-Relación de los sistemas de información que utilice la entidad.	No cumple	Diagrama E-R está desactualizado.	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input checked="" type="checkbox"/>
Solicite y revise que exista un plan de trabajo para la implementación de los sistemas informáticos.	No cumple	No existe documentación sobre plan de trabajo para la Implementación de los sistema.	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input type="button" value="Editar"/>	<input checked="" type="checkbox"/>

Imagen 45: Pantalla para generar el contenido de los reportes del sistema SAS-SV

Los campos de contenido a editar son los siguientes:

Área auditada: Identificación del área a la que corresponden los hallazgos encontrados.

Criterio: Base legal que se está incumpliendo.

Causa: Motivo por lo que se ha dado el hallazgo.

Efecto: Consecuencias que trae el incumplimiento del hallazgo encontrado y recomendaciones del auditor.

Comentario: observación del personal con el que se está discutiendo el acta.

NOTA: para ingresar información a cada campo haga clic al botón editar correspondiente.

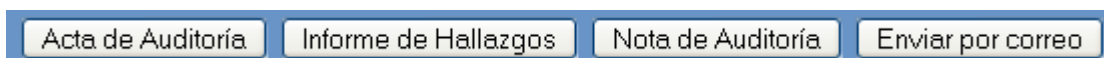


Imagen 46: Pantalla para generar los reportes del sistema SAS-SV

Los tres primeros botones generan el Acta de Auditoría, el Informe de Hallazgos y la Nota de Auditoría respectivamente. El último envía los reportes generados al correo del administrador.

Manual de Mantenimiento



SAS-SV

Software de Auditoría de Sistemas para la Superintendencia de Valores



San Salvador, Noviembre de 2007

SUMARIO DE CONTENIDO

1. Introducción
2. Estructura del software como sitio web.
Diagrama general de directorios
Matrices de contenidos
 - Directorio clases
 - Directorio estilos
 - Directorio imágenes
 - Directorio páginas
 - Directorio raíz.
 - Directorio respaldos
 - Directorio scripts
3. Manejo de sesiones
4. Manejo de envíos de datos POST
5. Documentación interna

1. INTRODUCCIÓN

El presente manual tiene por objetivo ilustrar a las personas que deseen conocer detalles del funcionamiento interno del software SAS-SV. Esta información puede servir al programador, web master o administrador, como apoyo al llevar a cabo una actualización del sistema a nivel de código o para llevar un registro de los archivos del sitio que facilite su posterior mantenimiento.

2. ESTRUCTURA DEL SOFTWARE COMO SITIO WEB

En este apartado se describirá el software, tomándolo como una aplicación de Internet, la cual, al igual que cualquier sitio para la Web, se compone de recursos de datos enlazados convenientemente. Se enumerarán dichos recursos utilizando matrices de contenidos para cada uno de los directorios.

2.1 Diagrama General de Directorios

La estructura de directorios que compone en general al sitio Web se detalla como sigue:

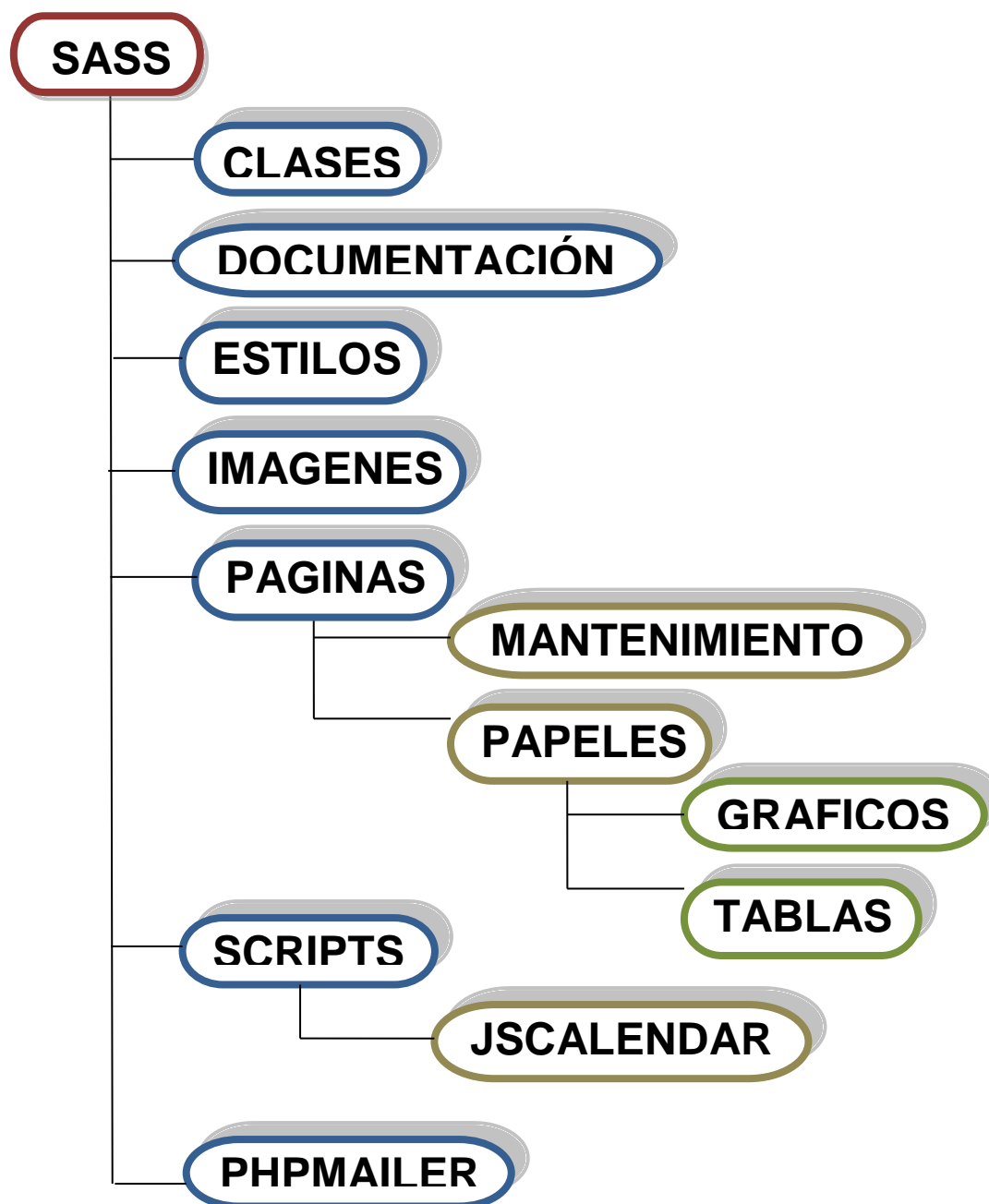


Imagen 1: Estructura de archivos de la aplicación web

2.2 Matrices de Contenidos

Las tablas que a continuación se muestran, llamadas también “matrices de contenido”, listan y describen cada uno de los archivos del sitio. Se presentan directorio por directorio según la estructura vista en el apartado 3.1 de este manual.

Directorio Raíz

Nombre del archivo	Descripción	Tipo de archivo	Formato
index.php	Página cuyo fin es redireccionar a la página de entrada al sistema <i>autenticación.php</i>	Texto	PHP
menu.php	Contiene la función que construye y personaliza la línea principal de menú, dependiendo de la página de la aplicación donde el usuario se encuentre en un determinado momento.	Texto	PHP

Directorio Clases

Nombre del archivo	Descripción	Tipo de archivo	Formato
clases.php	Archivo que contiene la definición de todas las clases PHP que se han instanciado en las páginas del sitio.	Texto	PHP

Directorio Documentación

Nombre del archivo	Descripción	Tipo de archivo	Formato
manualUsuario.php	El manual de usuario del sistema.	Texto	PHP
manualMantenimiento.php	El manual de mantenimiento del sistema.	Texto	PHP
manuallInstalacion.php	El manual de instalación del sistema.	Texto	PHP

Directorio Estilos

Nombre del archivo	Descripción	Tipo de archivo	Formato
estilos.css	Hoja de estilos que se incluye y aplica en las páginas de la aplicación visibles al usuario.	Texto	CSS

Directorio Imágenes

Nombre del archivo	Descripción	Tipo de archivo	Formato
boton1.jpg	Botón de salida de la aplicación, ubicado en el menú principal	Imagen	JPG
boton2.jpg	Botón para el trabajo con papeles de auditoría, ubicado en el menú principal.	Imagen	JPG
boton3.jpg	Botón para el acceso a las páginas de mantenimiento. Sólo visible en el rol de administrador y ubicado en el menú principal.	Imagen	JPG

botonCambios.jpg	Botón para acceder a la opción de renombrar el papel de trabajo actual o modificar su descripción. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
botonCerrar.jpg	Botón para cerrar el papel de trabajo actual y permitir a la aplicación abrir otro. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
botonHallazgos.jpg	Botón para acceder a la funcionalidad de generación de reportes. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
botonHardware.jpg	Botón para acceder a las preguntas de auditoría en el papel actual, correspondientes a la cédula de hardware. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
botonMarcas.jpg	Botón para acceder a la cédula de marcas del papel actual. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
botonPlaneacion.jpg	Botón para acceder a las preguntas de auditoría en el papel actual, correspondientes a la cédula de planeación. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG

botonPlaneación.jpg	Botón para acceder a las preguntas de auditoría en el papel actual, correspondientes a la cédula de planeación. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
botonPolíticas.jpg	Botón para acceder a las preguntas de auditoría en el papel actual, correspondientes a la cédula de políticas y procedimientos. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
botonSeguridad.jpg	Botón para acceder a las preguntas de auditoría en el papel actual, correspondientes a la cédula de seguridad. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
botonSoftware.jpg	Botón para acceder a las preguntas de auditoría en el papel actual, correspondientes a la cédula de software. Ubicado en el menú lateral izquierdo para trabajo con papeles de auditoría.	Imagen	JPG
encabezado2.jpg	Logo del sistema. Ubicado en la parte central superior de cada una de las páginas del sitio.	Imagen	JPG
iconoContactos.jpg	Icono que señala la opción de acceso al mantenimiento de contactos de una institución. Sólo visible para el rol de administrador.	Imagen	JPG

iconoModificar.jpg	Icono que señala la opción de acceso a la modificación de datos en los mantenimientos de usuario, instituciones o contactos.. Sólo visible para el rol de administrador.	Imagen	JPG
iconoPapelera.jpg	Icono que señala la opción de acceso al borrado de datos para un usuario o contacto Sólo visible para el rol de administrador.	Imagen	JPG
logoSuper.jpg	Imagen representativa de la Superintendencia de Valores de El Salvador. Ubicada en la parte izquierda superior de todas las páginas del sitio.	Imagen	JPG
logoUDB.jpg	Imagen representativa de la Universidad Don Bosco. Ubicada en la parte izquierda superior de todas las páginas del sitio.	Imagen	JPG

Directorio Scripts

Nombre del archivo	Descripción	Tipo de archivo	Formato
scripts.js	Archivo de funciones Java Script utilizadas en la mayoría de páginas del sitio. Las funciones incluidas en él están ordenadas alfabéticamente.	Texto	JS

Directorio jsCalendar

Este directorio no es nativo del software que ocupa este manual. Es una librería Java Script de libre distribución, diseñada para construir calendarios dinámicos. En el software SAS-SV se hace referencia a ella en la página *bitacoraUsuario.php* para desplegar un calendario que sirva para elegir la fecha de la cual se quiere ver la bitácora. Esta librería fue descargada en su última versión de la dirección: <http://www.dynarch.com/projects/calendar/>. En este sitio se puede además encontrar la documentación referente a este directorio.

Directorio phpMailer

En este directorio se encuentra el archivo *class.phpmailer.php* que contiene una clase espacial para el manejo del correo electrónico. Este archivo no pertenece propiamente al software, sino que se ha descargado externamente del sitio <http://phpmailer.sourceforge.net> donde se encuentra la documentación del archivo.

3. MANEJO DE SESIONES

El enlace de las páginas del sitio no se hace únicamente mediante vínculos HTML. Esto debido a que es necesario mantener determinada información: nombre del usuario y su rol, papel e institución actual así como también el nombre de la plantilla de preguntas y el tipo de menú principal que se despliega dependiendo del rol del usuario. En el software SAS-SV se ha optado por incluir estos datos una sola vez en una página (específicamente en la página *menuPrincipal.php* del directorio *paginas*) y referenciarlos varias veces en las demás páginas. Esto se logra aplicando el concepto de sesiones, el cual es un mecanismo que provee el lenguaje PHP para agrupar un grupo de páginas web dinámicas de un mismo sitio en una sola aplicación con variables comunes. Además, las sesiones son útiles para evitar la entrada de usuarios a la aplicación sin antes pasar por el proceso de autenticación.

La tabla siguiente describe cada una de las variables de sesión que se utilizan en el sitio:

Nombre de la variable	Descripción
\$modoSesion	Esta variable puede contener dos valores: El 1 indica que el software está funcionando en usuario de “emergencia”, que sólo permite añadir un usuario administrador, si es que no está añadido ya uno. El valor 2 indica que se está usando como usuario “administrador”, “consulta” u “operador”.
\$rolSesion	Esta variable contiene una cadena que representa el rol del usuario que está usando el sistema actualmente. Sus valores pueden ser “Administrador”, “Operador” o “Consulta”.
\$usuarioSesion	Contiene el nombre de usuario de la persona que ocupa el software. Dicho nombre también es conocido como su <i>login</i> .
\$idMenu	Esta variable almacena un entero que identifica el tipo de menú que se construirá dinámicamente, dependiendo del rol del usuario actual. Un valor de 1 hará un menú para administrador, un 2 para operador, un 3 para consulta y un 4 para emergencia.
\$tipoEntidadSesion	Contiene un número identificador del tipo de institución a la cual se le está trabajando un papel de auditoría. Este número es 0 para almacenes, 1 para Bolsa de Valores, 2 para una casa corredora y 3 para CEDEVAL.
\$nombrePlantillaSesion	Contiene el nombre de la plantilla de preguntas personalizada que se está aplicando a las cédulas del papel actual.
\$nombrePapelSesion	Contiene el nombre dado por el usuario al <u>papel</u> actual.

\$nombreEntidadSesion	Contiene el nombre de la institución a la cual se le está trabajando un papel de auditoría.
\$nsesion	Número identificador de la sesión PHP para mantener todas las páginas del sitio con variables comunes.
\$sesion	Nombre identificador de la sesión PHP para mantener todas las páginas del sitio con variables comunes.
\$entrada	Variable que determina si se ha entrado al sistema por medio de la página de autenticación.

4. MANEJO DE ENVIO DE DATOS POST

En la aplicación, por motivos de seguridad, se prefiere usar el envío de datos POST al GET cuando es necesario. Dicho envío se realiza por una de las dos formas siguientes:

La primera es por medio de *formularios ocultos*, los cuales se colocan al principio de la página que se ocupan. Estos formularios contienen campos tipo “hidden”, y se envían con método POST. Para disparar su evento *submit*, se hace uso de funciones Java Script que se nombran *enviarFormularioX* donde X es un número que identifica a la función. Una de estas funciones típicas se define como sigue:

```
function enviarFormularioX(valor1,valor2){
    // Se coloca el valor de los campos hidden que se enviarán:
    document.frmOculto1.campoHidden1.value = valor1;
    document.frmOculto1.campoHidden2.value = valor2;
    // Poner el valor de otros campos hidden
    // Se envía el formulario vía POST:
    document.frmOculto1.submit();
}
```

Todas las funciones de envío de formularios ocultos se listan en el archivo *scripts.js* de la carpeta *scripts*.

La segunda forma es ocupando *envío de datos asíncrono*. Esto se logra por medio del objeto Java Script XMLHttpRequest, incluido en la mayoría de navegadores populares de la actualidad. Para ejemplos de su uso y comentarios específicos, revisar el código de las funciones *objetoAjax* y *autenticar2* en el archivo *scripts.js* de la carpeta *scripts*.

5. DOCUMENTACIÓN INTERNA

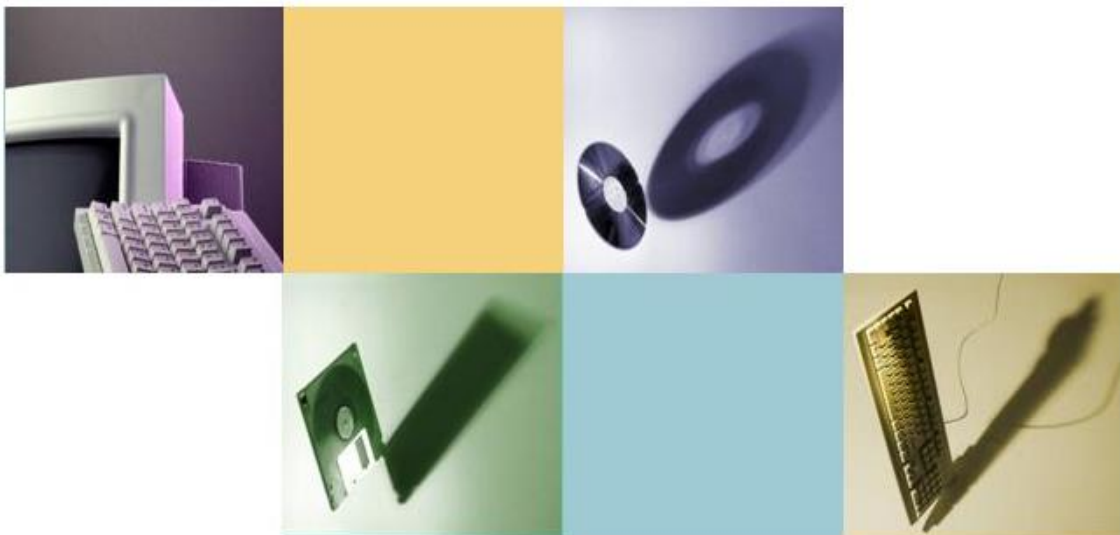
Tanto en los archivos de Java Script como en los de PHP, se comenta el código profusamente. Las líneas de comentarios se distinguen por comenzar con `//` para una sola línea, o encerradas entre `/*` y `*/` para varias líneas. Para distinguir mejor el código, se sugiere el empleo de algún editor especializado con cromado de código como el ConText, que se ocupó para el desarrollo de este software, o el OmegaEdit.

Manual de Instalación



SAS-SV

Software de Auditoría de Sistemas para la Superintendencia de Valores



San Salvador, Noviembre de 2007

SUMARIO DE CONTENIDO

1. Introducción
2. Requisitos del sistema
 - a. Software
 - b. Hardware
3. Proceso de instalación
4. Acerca del CD de instalación

1. INTRODUCCIÓN

En el presente manual se describen los pasos que el usuario debe realizar para instalar por primera vez en su estación de trabajo el sistema SAS-SV (Software de auditoría de sistemas para la Superintendencia de Valores).

2. REQUISITOS DEL SISTEMA

El sistema puede usarse en una sola máquina que funja como servidor y cliente al mismo tiempo, o con una máquina servidora y varias computadoras clientes. La tabla siguiente muestra los requisitos mínimos de ambos tipos de máquina antes de la instalación del software, la cual sólo se lleva a cabo en la computadora servidora.

Tipo de requisito	Tipo de Máquina	
	Servidor	Cliente
Hardware	300 megabytes mínimos de espacio libre en disco duro de la máquina servidora, para la instalación de los servidores y de la base de datos.	Conexión de red local, con acceso a la computadora servidora.
	Procesador Pentium	Procesador Pentium
	24 MB de RAM	16 MB de RAM
	Conexión de red local con las computadoras cliente.	
Software	Servidor Web Apache 5.0	Sistema Operativo Windows 9x, Me, NT, 2000, XP, o Windows Server 2003.

	Intérprete de lenguaje PHP versión 5	Soporte de protocolo de comunicación TCP/IP
	Servidor de Base de Datos MySQL 5.0	Navegador de Internet (Explorer 5, Firefox 1.0.7, Netscape 8.1.3 u Opera 9.21)
	Sistema Operativo Windows 9x, Me, NT, 2000, XP, o Windows Server 2003.	
	Soporte de protocolo de comunicación TCP/IP	
	Navegador de Internet (Explorer 5, Firefox 1.0.7, Netscape 8.1.3 u Opera 9.21)	

3. PROCESO DE INSTALACIÓN

El software se instala una sola vez en la máquina destinada como servidor. Una vez hecho esto, las máquinas clientes (que están conectadas en red al servidor), podrán usar el software, digitando en su navegador de Internet:

`http://direccion ip del servidor/sassv`

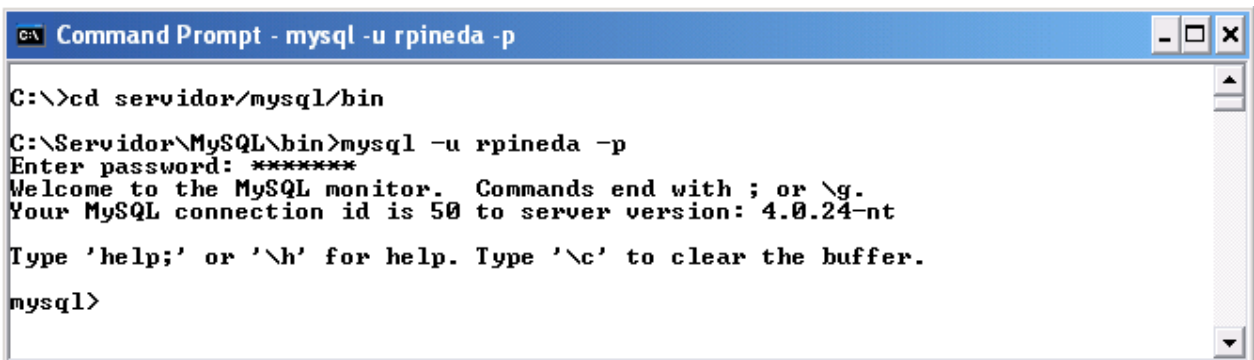
Un ejemplo es: `http://4.8.15.2/sassv`

En donde, en el ejemplo, la dirección ip de red del servidor es 4.8.15.2

Para instalar el software en el servidor, siga los siguientes pasos (se presuponen existentes en el servidor los requisitos software mencionados en el apartado 2 de este manual) :

- 1) Localice en la computadora servidor, la carpeta raíz de su Web (la localización de ésta variará dependiendo de cómo haya instalado su servidor Web).
- 2) Copie la carpeta SASSV del CD-ROM del software a dicha carpeta raíz.
- 3) Localice en la computadora servidor, la carpeta bin de su servidor de base de datos MySQL (la localización de ésta variará dependiendo de cómo haya instalado su servidor Web, aunque generalmente suele instalarse en la misma carpeta padre del servidor Web).
- 4) Copie el archivo guion.sql del CD-ROM del programa, a la carpeta bin encontrada anteriormente.

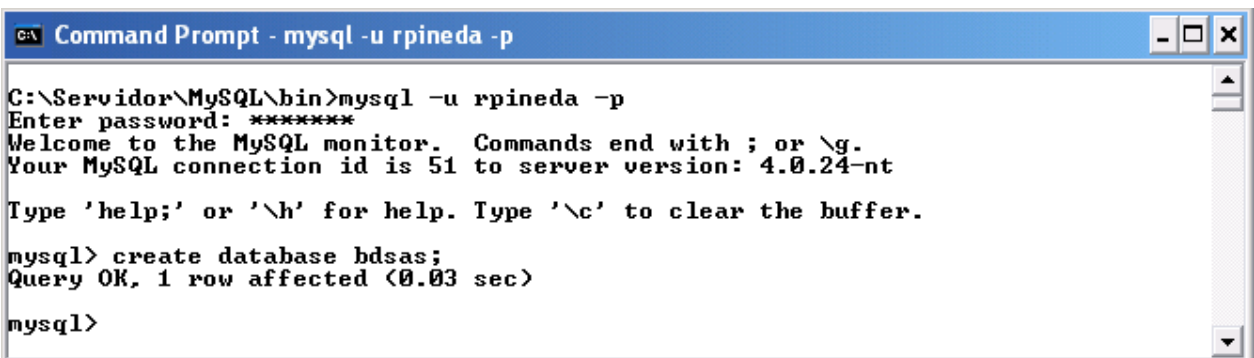
- 5) Entre al programa cliente mysql de su servidor de base de datos. Para ello puede utilizar la consola MS-DOS de Windows. Para correr dicho programa deberá ubicarse en el directorio bin de la carpeta MySQL. Ingrese al mismo utilizando un nombre de usuario y contraseñas válidas. Todo lo anterior, dependerá de cómo se haya instalado el servidor MySQL. Aquí hay un ejemplo específico para una máquina, el cual no necesariamente debe ser su caso:



```
C:\>cd servidor/mysql/bin
C:\Servidor\MySQL\bin>mysql -u rpineda -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 50 to server version: 4.0.24-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql>
```

Imagen 1: Pantalla de ingreso al programa cliente mysql

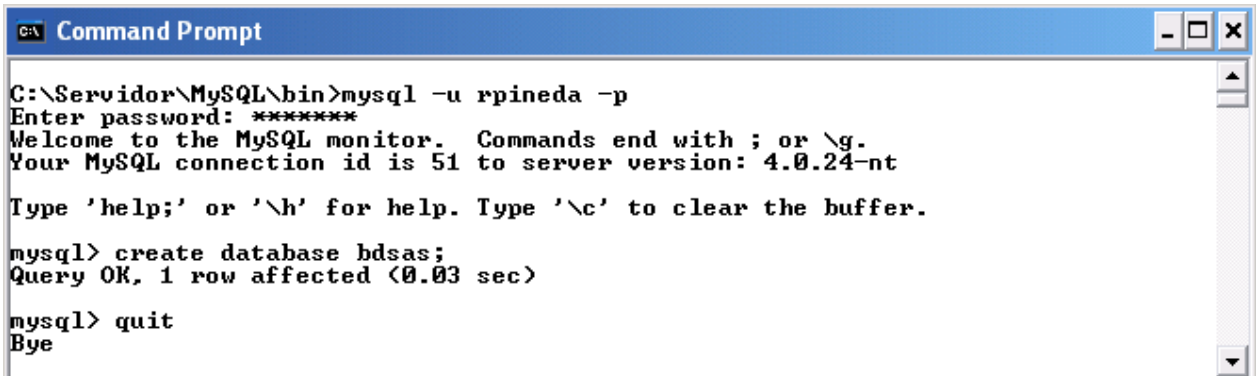
- 6) Una vez adentro, cree la base de datos bdsas:



```
C:\Servidor\MySQL\bin>mysql -u rpineda -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 51 to server version: 4.0.24-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> create database bdsas;
Query OK, 1 row affected (0.03 sec)
mysql>
```

Imagen 2: Pantalla de creación de la base de datos bdsas

7) Salga del cliente mysql:



```
C:\Servidor\MySQL\bin>mysql -u rpineda -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 51 to server version: 4.0.24-nt

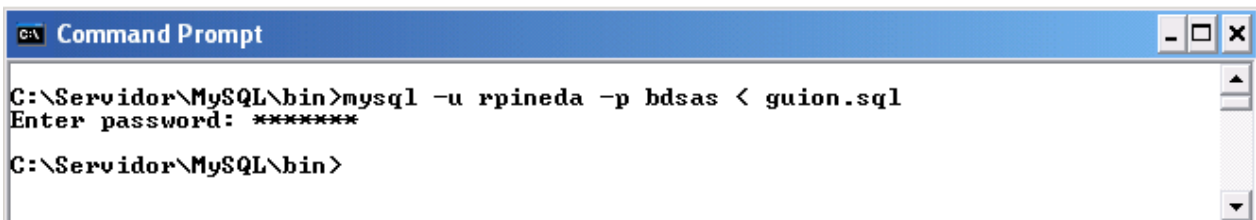
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create database bdsas;
Query OK, 1 row affected (0.03 sec)

mysql> quit
Bye
```

Imagen 3: Pantalla de salida del cliente mysql

8) Vuelva a entrar; pero esta vez, utilizando la base creada y llamando el archivo guion.sql copiado en el paso 2. La siguiente figura, muestra como hacerlo.



```
C:\Servidor\MySQL\bin>mysql -u rpineda -p bdsas < guion.sql
Enter password: *****
C:\Servidor\MySQL\bin>
```

Imagen 4: Pantalla de ingreso a la base de datos cargando el archivo guion.sql

9) Abra el archivo clases.php de la carpeta clases del directorio sassv que copió en el paso 2.

10) La línea 13 se lee así:

```
function Conexion($user="rpineda",$wordkey="rpineda",$serv="localhost",$base="bdsas")
```

Modifique dicha línea para que incluya el nombre de su servidor web, así como el login y password de su servidor de base de datos MySQL. Consulte la documentación de su servidor web para encontrar estos datos. Por ejemplo, si la

dirección de su servidor web es 4.8.15.7, el usuario de la base de datos es *userdb* y el password es *passdb*, la línea quedaría así:

```
function Conexion($user="userdb",$wordkey="passdb",$serv="4.8.15.7",$base="bdsas")
```

Una vez hecho lo anterior, podrá ingresar al sistema, desde cualquier explorador Web utilizando la siguiente dirección:

http:// dirección ip de la máquina servidor/sassv

Si la misma computadora servidora sirve como cliente (no será así generalmente), la dirección ip será 127.0.0.1 ó localhost.

- 11) Posteriormente, deberá registrarse un usuario administrador. Para ello, ingrese al sistema desde su navegador web, de la forma explicada anteriormente:



The image shows a web application interface for user authentication. At the top, there is a blue header bar with a yellow trapezoidal background. On the left side of the header is the logo of the Superintendencia de Valores (SV), and on the right side is the logo of Universidad Don Bosco. The central text in the header reads "SAS-SV SOFTWARE DE APOYO PARA LA AUDITORIA DE SISTEMAS SUPERINTENDENCIA DE VALORES". Below the header, the main content area is light yellow. In the center of this area, there is a login form consisting of two input fields: "Login:" and "Password:", followed by an "Ingresar" button.

Imagen 5: Pantalla de validación de Usuario y Clave

12) Una vez en esta pantalla ingrese utilizando los datos

Login: sa

Password: sap



SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

Login:

Password:

Ingresar

Imagen 6: Pantalla de ingreso de Usuario y Clave

13) Escoja el menú mantenimiento y luego la opción usuarios



SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

PAPELES DE TRABAJO MANTENIMIENTO AYUDA SALIR

Usuarios
Instituciones/Contactos
Preguntas
Copia de Seguridad
Estadísticas

Fecha: 19-11-2007
Hora: 23:52:1

Imagen 7: Pantalla de ingreso al mantenimiento de usuarios del sistema

14) Haga clic en Agregar un nuevo usuario.

SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

Fecha: 19-11-2007
Hora: 23:56:3

MANTENIMIENTO DE USUARIOS

Se encontraron 4 usuarios registrados en el sistema. [Agregar un nuevo usuario](#)

Número	Nombre	Apellido	Cargo	Login	Password	Rol	Eliminar	Modificar	Ver Historial
1	Margarita	Pineda	auditora	rpineda1	rpineda1	Administrador			
2	Juan	Perez	contador	juanjuan	juanjuan	Consulta			
3	Joaquin	Bolaños	Auditor de sistemas	joaquin1	joaquin1	Operador			
4	joaquin	bolaños	operador	joaquin2	joaquin2	Operador			

Imagen 8: Pantalla de mantenimiento de usuarios

15) Llene el formulario con los datos del administrador y haga clic en ingresar usuario.

SAS-SV
SOFTWARE DE APOYO PARA LA
AUDITORIA DE SISTEMAS
SUPERINTENDENCIA DE VALORES

Fecha: 16-12-2007
Hora: 0:35:57

Agregar Nuevo Usuario

Nombres:

Apellidos:

Cargo:

Login (8 caracteres mínimo):

Password (8 caracteres mínimo):

Confirmar password:

Rol:

Imagen 9: Pantalla para adicionar un nuevo usuario del sistema

Luego de haber llenado el formulario aparecerá un mensaje de confirmación del ingreso.

Una vez hecho esto, puede comenzar a utilizarse el sistema. El administrador podrá registrar roles de usuario y de consulta y éstos podrán utilizarlo según los privilegios que su rol les proporcione.

Para mayor explicación sobre el uso del sistema, consulte el **manual de usuario** de sas-sv.

4. ACERCA DEL CD DE LA APLICACIÓN

El CD que acompaña este trabajo contiene dos carpetas:

El directorio `sassv` incluye el software propiamente dicho. Esta carpeta a la que se refiere el paso 2 del apartado **proceso de instalación**. Aquí también se incluye una carpeta con la documentación (manuales de usuario, mantenimiento e instalación).

El directorio `trabajo` incluye el trabajo escrito en el cual se apoya el diseño y desarrollo de la aplicación.

CAPITULO 5
CONCLUSIONES
Y
RECOMENDACIONES

CONCLUSIONES

- El software diseñado en este proyecto es compatible con la tecnología informática utilizada por la Superintendencia de Valores, lo anterior fue verificado habiendo realizado una visita a las instalaciones de dicha entidad.
- Las cédulas de auditoría contenidas en el software pueden ser utilizadas por los auditores de sistemas de la Superintendencia de Valores y a la vez, se puede aprovechar la elaboración automática de reportes.
- Un buen diseño de flujo de datos y de relaciones entre entidades de base de datos, facilita la etapa de desarrollo del software.
- El ambiente WEB puede adaptarse a una interfaz gráfica de usuario acorde a sus necesidades y las del sistema elaborado.
- Las pruebas de funcionamiento son esenciales durante la mayor parte del Ciclo de Vida de Desarrollo de Sistemas de Información.
- Todo aplicativo destinado a distribución pública debe contar con el debido apoyo de documentación técnica, con el fin de facilitar su uso, mantenimiento e instalación.

RECOMENDACIONES

- Se recomienda que el software desarrollado sea implementado en un ambiente de red interna con el fin de preservar la seguridad de la información manejada y mejorar el flujo de datos.
- La Universidad debería proponer en un futuro la segunda parte de este proyecto para la formación de nuevos profesionales, la cual consiste en la adición de la parte de auditoría financiera.

ANEXOS

ANEXO 1: Formato de encuesta de investigación de hechos



**Universidad Don Bosco
Facultad de Ingeniería
Escuela de Computación**

Tema de trabajo de graduación:

Software de Apoyo para el Control de Auditorías de Sistemas con énfasis en el Área de Tecnología de Información.

Encuesta presentada por:

Rosa Margarita Pineda Jovel
Joaquín Eduardo Bolaños Amaya

Objetivo de la encuesta:

Determinar la necesidad de un software especializado que sirva de apoyo para el trabajo del auditor de sistemas.

ENCUESTA

1. Escriba su nombre (opcional):

2. Indique la empresa donde usted labora (opcional):

3. Que es para usted la auditoría de sistemas?

4. ¿Conoce cuáles son las funciones de un auditor de sistemas? Si las conoce, mencione algunas

5. Marque cuáles aspectos conoce que se evalúan cuando lleva a cabo una auditoría de sistemas.

- | | | | | | |
|---------------------------------|-----|-----------------------------|-----|-------------------------------|-----|
| Instalaciones | () | Licencias | () | Soporte técnico | () |
| Respaldo de datos | () | Contraseñas | () | Captura de datos | () |
| Validación de datos | () | Tipo de software | () | Sistemas Operativos | () |
| Acceso a personal no autorizado | () | Protección contra Desastres | () | Documentación de los sistemas | () |
| Interfaz del sistema | () | Capacitación del personal | () | Presupuesto | () |

Otros: _____

No conozco ()

6. ¿Conoce qué tipo de documentación elabora al final de una auditoría?

- Acta ()
- Informe de hallazgos ()
- Informe Gerencial ()
- Nota con hallazgos ()
- No conozco ()

7. ¿En promedio cuanto tiempo estima usted que se tarda una persona en elaborar dichos documentos, sin software especializado?

- Menos de 1 hora () Entre 1 y 2 horas () Entre 2 y 3 horas ()
Entre 3 y 4 horas () Entre 4 y 5 horas () Entre 5 y 6 horas ()
Entre 6 y 7 horas () Mas de 7 horas ()

8. ¿Si usted trabajara como auditor de sistemas estima necesario utilizar alguna clase de formularios para realizar dicha tarea?

Si () No ()

¿Porque? _____

9. ¿Cuánta experiencia tiene usando computadoras?

- Nada ()
Poca ()
Normal ()
Mucha ()

10. ¿Está usted familiarizado con el uso de Internet?

Si () No ()

11. ¿En qué forma lo utiliza?

- Consultar sus papeles de trabajo ()
Investigar información de la empresa a auditar ()
Sitio web de mi empresa ()
Contacto con las empresas fiscalizadas ()
Otro uso: _____

12. ¿Conoce usted algún software para asistir la auditoría de sistemas?

Si () No ()

13. Si la respuesta es si, seleccione cuales conoce:

AutoAudit ()

ERA ()

Cobit Advisor ()

Word ()

Excel ()

Otros: _____

14. ¿Cuál cree que es el mayor impedimento para que un auditor de sistemas use software especializado en su labor?

Alto Costo ()

Falta de experiencia en el manejo de computadoras ()

Falta de computadora ()

Otros: _____

15. ¿Qué características considera debe tener un software para que le sirva de apoyo a un usuario auditor de sistemas?

Facilidad de uso ()

Agilidad en la obtención de resultados ()

Bajo Costo ()

Interfaz amigable ()

Otros: _____

Si tiene observaciones a esta encuesta, puede hacerlas aquí:

Gracias por su colaboración.

ARTICULO TÉCNICO

BIOGRAFIAS

Rosa Margarita Pineda Jovel

Joaquín Eduardo Bolaños Amaya

Rosa Margarita Pineda Jovel, nació el 27 de Enero de 1,977 en la ciudad de Usulután departamento de Usulután.

Realizo sus estudios de educación básica y educación media en el Colegio Cristiano Prof. Justo Gonzales Carrasco de Usulután, luego el bachillerato lo efectuó en el Liceo Británico de Usulután habiéndose graduado con honores del bachillerato comercial opción computación; Sus estudios superiores los realizó en la Universidad Don Bosco en la especialidad de ingeniería en ciencias de la computación.

Laboralmente se ha destacado como administradora de sistemas y actualmente en el área de auditoría de sistemas.

Joaquín Eduardo Bolaños Amaya, nació en San Salvador el 31 de Octubre de 1982. Realizó su educación media y básica en el colegio Don Bosco de Soyapango, donde obtuvo el bachillerato de técnico vocacional comercial opción contaduría.

Realizó sus estudios superiores en la universidad Don Bosco en la especialidad de ingeniería en ciencias de la computación.

Se ha desarrollado laboralmente en el campo de la programación web donde continúa al presente año 2007.

RESUMEN

El presente artículo técnico describe de manera resumida la forma en que se realizó el trabajo de graduación “**SOFTWARE DE APOYO PARA LA AUDITORÍA DE SISTEMAS CON ÉNFASIS EN EL ÁREA TECNOLÓGICA PARA LA SUPERINTENDENCIA DE VALORES**”, realizado en el año 2007 para optar al título de ingeniero en ciencias de la computación.

El sistema SAS-SV, es un software de apoyo para el control de auditoría de sistemas con énfasis en el área de tecnología de información para la Superintendencia de Valores.

Debido a que dicha institución se dedica a la supervisión del mercado de valores, es recomendable que utilicen software especializado para la realización de auditorías de sistemas. Nuestro país está dando pasos para lograr un mayor desarrollo tecnológico, sin embargo referente a las auditorías de sistemas escasea el software especializado. Por lo cual, este proyecto será una propuesta innovadora para la realización de las auditorías en la Superintendencia de Valores. El software ayudará al manejo de papeles de trabajo en el momento en que el auditor de sistemas realice su trabajo de campo. También contendrá estadísticas para una buena toma de decisión.

OBJETIVOS DEL DESARROLLO DEL SOFTWARE.

De manera general se puede decir que el propósito del proyecto es desarrollar una aplicación informática que sistematice el proceso de auditorías de sistemas enfocadas en el área del software para la Superintendencia de Valores.

RESUMEN DEL DESARROLLO DEL PROYECTO.

En base a investigaciones realizadas se verifico que la Superintendencia de Valores actualmente no cuenta con una aplicación especializada para realizar las auditorias de sistemas, por lo que se optó por trabajar con dicha entidad para elaborar el software SAS-SV (Software de auditoria de sistemas para la Superintendencia de Valores).

A continuación se muestra breve conclusión de cada capítulo contenido en el presente proyecto.

CAPITULO I

En este capítulo se habla sobre los antecedentes del proyecto desarrollado y se menciona la importancia del tema la cual radica en sistematizar los procedimientos de auditoría de sistemas que realiza la Superintendencia de Valores.

También en este capítulo se describen los objetivos por el cual se ha realizado el proyecto.

En el marco teórico se detallan las evoluciones que ha tenido la computadora con el desarrollo de la tecnología a nivel mundial.

En cuanto al marco conceptual se da el significado de auditoría y sus diversas clasificaciones; finalmente se da a conocer el presupuesto del proyecto.

Del trabajo realizado en este capítulo se concluye que una de las áreas beneficiadas con el avance de la tecnología es la auditoría de sistemas, ya que con el apoyo de herramientas especializadas el auditor puede realizar su labor en menor tiempo y con mayor eficacia.

CAPITULO II

Es en este capítulo donde se da a conocer el proceso completo de la encuesta realizada a profesionales del área de sistemas, siendo el objetivo de esta determinar la necesidad de un software especializado que sirva de apoyo para el trabajo del auditor de sistemas.

También en esta parte es donde se muestra el diseño de la base de datos utilizando el modelo de entidad – relación, el diseño relacional y la normalización.

De lo anterior se concluye que existe poco conocimiento de la existencia de software especializados en el área de auditoría de sistemas, también se puede decir que en El Salvador no existe mucha documentación referente a esta área, por lo que el trabajo de graduación desarrollado será de mucho beneficios para aquellos interesados en esta clase de herramientas..

CAPITULO III

El tercer capítulo del proyecto de graduación explica el funcionamiento del software desarrollado, y a la vez describe los procesos que se realizan para la operatividad de los datos.

Se puede concluir de este capítulo que el software desarrollado tiene como respaldo la documentación de cada proceso, haciendo uso del diseño orientado a objetos utilizando el Lenguaje Unificado de Modelado (UML).

CAPITULO IV

Este capítulo se compone de los manuales que sirven como apoyo para comprender el software, siendo estos los siguientes: manual de usuario, de instalación y de mantenimiento. De lo anterior se concluye que el trabajo de graduación elaborado se ha documentado de manera que sirva de apoyo tanto a los programadores como a los usuarios finales.

Finalmente como agregado al proyecto se ha editado el presente artículo técnico.

EXPERIENCIAS DE APRENDIZAJE EN EL DESARROLLO DEL PROYECTO

Con la elaboración del presente proyecto se puede decir que se ha logrado conocer con mayor profundidad el funcionamiento del área de la auditoría de sistemas, además se ha logrado conocer más de cerca la labor realizada por los auditores de sistemas.

BIBLIOGRAFÍA Y FUENTES DE CONSULTA

A. Bibliografía

AMBEGAONKAR, Prakash; Kit de Recursos de Intranet; 1ª. Edición, editorial McGrawHill, Madrid 1997, 495 páginas.

Se ocupó para determinar las necesidades hardware de la computadora servidora de la aplicación web.

BELL, Douglas; PARR, Mike; Java para estudiantes, 3ª. Edición, editorial PrenticeHall, México 2003, 634 páginas.

El apéndice I de este libro contiene una descripción de tres de los diagramas UML ocupados en el diseño de este proyecto. De clases, de colaboración y de actividades.

DEITEL, Harvey M.; DEITEL Paul J.; C++ Cómo Programar; 4ª. Edición, editorial PrenticeHall, México 2003, 1320 páginas.

En los primeros cinco capítulos de este libro se trata un ejemplo práctico del diseño orientado a objetos con UML. De este apartado se basaron los diagramas de secuencia del presente documento.

ESTUPIÑÁN, Rodrigo; Papeles de Trabajo en la Auditoría Financiera, 1ª. Edición, editorial Roesga, Bogotá, 285 páginas.

Se consultaron modelos de formatos en cédulas financieras para el diseño de las cédulas de auditoría informática.

KROENKE, David M. Procesamiento de Bases de Datos; 8ª. Edición, editorial Pearson Education, México 2003.

Este libro se consultó en el capítulo 3 con motivo de averiguar la nomenclatura utilizada en los diagramas Entidad – Relación.

MICROSOFT, Diccionario de Informática e Internet, 2ª. Edición, editorial McGrawHill, España 2005, 881 páginas.

Este diccionario sirvió como apoyo en la definición de términos técnicos usados en el contenido de este documento.

RUBIO, Javier Gil; TEJEDOR CERBEL, Jorge A., Creación de sitios web con PHP 4, editorial McGrawHill, España 2001, 547 páginas.

La utilidad de este libro radica en que sirvió como referencia en la etapa de desarrollo (codificación) del proyecto. Fue de particular ayuda sus capítulos 10 y 11, “El Lenguaje SQL” y “Acceso a Bases de Datos” respectivamente.

WHITTEN, Jeffrey L.; BENTLEY, Lonie D.; BARLOW Victor M.; Análisis y Diseño de Sistemas de Información, editorial McGrawHill, Colombia, 1996, 907 páginas.

Este libro sirvió como apoyo en el desarrollo de los conceptos de Ciclo de Vida de Sistemas de Información, los cuales se tratan en el capítulo I de este trabajo.

B. Recursos de Internet¹⁶

<http://www.clikear.com/manuales/uml/>

En esta página se encuentra un manual del lenguaje gráfico UML. Abarca tanto grafismo como principios de diseño orientado a objetos. El UML incluye tanto notación básica como notación avanzada. De esta página se han consultado los conceptos básicos del lenguaje.

<http://mysql.conclase.net/curso/index.php>

El curso de MySQL contenido en esta página contiene un apartado dedicado al diseño de bases de datos relacionales. Ejemplifica los modelos conceptual y lógico de las bases de datos y la normalización. El apartado del diseño de bases de datos de este documento está basado en dicha metodología.

¹⁶ Sitios Web visitados entre Diciembre de 2006 y Mayo de 2007.

<http://www.netconsul.com/>

Sitio dedicado a las soluciones informáticas para auditoría. Sirvió como referencia para el apartado de marco experimental, donde se describen algunos de los principales software de este rubro.

<http://www.audisis.com/index.html>

Sitio Web que promueve soluciones informáticas para auditoría. Sirvió como fuente para el marco experimental.

GLOSARIO

A

Álgebra relacional: Reglas sintácticas que definen la utilización de operadores entre las relaciones de un modelo relacional de bases de datos, para especificar operaciones de manipulación y extracción de datos de las mismas relaciones.

Atributo (1): En teoría de Bases de Datos, cada una de las características que posee una entidad de bases de datos, la cual puede representarse por un dato específico.

Atributo (2): En Programación Orientada a Objetos, cada una de las características que describen a un objeto y que son representadas como variables de clase en los lenguajes de programación orientados a objetos.

Auditor: Persona capacitada para realizar auditorías en empresas u otras instituciones.

Auditoría: Es la revisión independiente de alguna o algunas actividades, funciones específicas, resultados u operaciones de una entidad administrativa, realizada por un profesional de la auditoría, con el propósito de evaluar su correcta realización y, con base en ese análisis, poder emitir una opinión autorizada sobre la razonabilidad de sus resultados y el cumplimiento de sus operaciones.

Auditoría Administrativa: Es la revisión sistemática y exhaustiva que se realiza a la actividad administrativa de una empresa, en cuanto a su organización, las relaciones entre sus integrantes y el cumplimiento de las funciones y actividades que regulan sus operaciones.

Auditoría de Sistemas: Es el proceso de revisión y evaluación, parcial o completo de los aspectos relacionados con el procesamiento automatizado de la información.

Auditoría Financiera (contable): Es la revisión sistemática, explorativa y crítica que realiza un profesional de la contabilidad a los libros y documentos contables, a los controles y registros de las operaciones financieras y a la emisión de los estados financieros de una empresa, con el fin de evaluar y opinar sobre la razonabilidad, veracidad, confiabilidad y oportunidad en la emisión de los resultados financieros obtenidos durante un período específico o un ejercicio fiscal.

Auditoría Fiscal: Se realiza con el objeto de verificar el correcto cumplimiento de las obligaciones en materia tributaria de los contribuyentes.

Auditoría Forense: Se define como una auditoría especializada en descubrir, divulgar y atestar sobre fraudes y delitos en el desarrollo de las funciones públicas, considerándose un verdadero apoyo a la tradicional auditoría gubernamental, en especial ante delitos tales como: Enriquecimiento ilícito, peculado, cohecho, soborno, desfalco, malversación de fondos, prevaricato, conflicto de intereses.

Auditoría Interna: Es la revisión que realiza un profesional de la auditoría, cuya relación de trabajo es directa y subordinada a la institución donde se aplicará la misma, con el propósito de evaluar en forma interna el desempeño y cumplimiento de las actividades, operaciones y funciones que se desarrollan en la empresa y sus áreas administrativas, así como evaluar la razonabilidad en la emisión de sus resultados financieros.

Auditoría Operacional: Es la revisión exhaustiva, sistemática y específica que se realiza a las actividades de una empresa, con el fin de evaluar su existencia, suficiencia, eficacia, eficiencia y el correcto desarrollo de sus operaciones,

cualesquiera que éstas sean, tanto en el establecimiento y cumplimiento de los métodos, técnicas y procedimientos de trabajo necesarios para el desarrollo de sus operaciones, en coordinación con los recursos disponibles, como en las normas, políticas, lineamientos y capacitación que regulan el buen funcionamiento de la empresa.

B

Base de Datos: Un archivo compuesto de registros, cada uno de los cuales contiene una serie de campos así como también un conjunto de operaciones para realizar búsquedas, ordenaciones, re combinaciones y otras funciones.

C

Cadena de caracteres: Estructura de datos compuesta de una secuencia de caracteres que normalmente representa un texto legible para las personas.

Cédula de Auditoría: Son los formularios que utiliza el auditor para realizar sus actividades y dejar constancia del trabajo efectuado.

Ciclo de Vida de Desarrollo de Software: Proceso por el cual los analistas de sistemas, los ingenieros de software, los programadores y los usuarios finales elaboran sistemas de información y aplicaciones informáticas.

Clase: En Programación Orientada a Objetos, una categoría generalizada que describe un grupo de elementos más específicos, denominados objetos, que pueden existir dentro de ella. Una clase es una herramienta descriptiva utilizada en un programa para definir un conjunto de atributos o un conjunto de métodos que caracterizan a cualquier miembro (objeto) de la clase.

Clave candidata: Es cada uno de los atributos que identifican de manera única a una entidad, sin importar si es capaz de llegar a ser clave principal de un conjunto de entidades.

Clave principal: Es una clave candidata que se elige de forma arbitraria, o bien tomando en cuenta características como: longitud de la clave, tipo de dato que la representa, facilidad para ser recordada, etc., para que cumpla el papel de identificar siempre de manera única a cada entidad, y a la vez, servir como campo enlace en las interrelaciones de conjuntos de entidades.

COBIT: Es una herramienta de gobierno de tecnologías de la información que vincula tecnología informática y prácticas de control, COBIT consolida y armoniza estándares de fuentes globales prominentes en un recurso crítico para la gerencia, los profesionales de control y los auditores. Se aplica a los sistemas de información de toda la empresa, incluyendo las computadoras personales, mini computadoras y ambientes distribuidos. Esta basado en la filosofía de que los recursos de tecnología de información necesitan ser administrados por un conjunto de procesos naturalmente agrupados para proveer la información pertinente y confiable que requiere una organización para lograr sus objetivos.

Controles: Medios a través de los cuales la gerencia de una entidad asegura que el sistema es efectivo y es manejado en armonía con eficiencia y economía, dentro del marco legal vigente.

Control de calidad (sector gubernamental): Conjunto de métodos y procedimientos implementados dentro de la Contraloría General de la República u otra entidad auditora, para obtener seguridad razonable que la auditoría llevada a cabo y el informe correspondiente cumplen con las Normas de Auditoría Gubernamental.

D

Dependencia funcional: Es una característica que se cumple en dos atributos de una entidad, siempre y cuando, para cada valor del primer atributo, le corresponde sólo un valor del segundo atributo.

Diagrama de actividades: Notación gráfica perteneciente al estándar UML la cual describe una secuencia de actividades en un programa de computación, incluyendo secuencia, decisión, ciclos y llamadas a métodos.

Diagrama de casos de uso: Notación gráfica perteneciente al estándar UML la cual se utiliza para representar las formas en que un usuario de un sistema de cómputo interactúa con el mismo.

Diagrama de clases: Notación gráfica perteneciente al estándar UML, la cual puede utilizarse para listar los atributos y métodos de una clase, haciendo énfasis a la encapsulación de los mismos. Asimismo muestra, haciendo uso de un diagrama arborizado, las relaciones de herencia entre clases hijas y superclases.

Diagrama de colaboración: Notación gráfica perteneciente al estándar UML, cuyo objeto es representar las invocaciones que un método de una clase realiza a otra clase

Diagrama de secuencia: Notación gráfica perteneciente al estándar UML, la cual grafica el orden en que se dan cada uno de las participaciones de los usuarios en el sistema de cómputo.

Diccionario de datos: Es un tipo especial de base de datos que almacena información sobre todos los campos de las tablas que conforman una base de datos de una aplicación informática. Dentro de dicha información tenemos: nombre del campo, tipo de dato, longitud, etc.

DIPA: Declaraciones Internacionales de Prácticas de Auditoría. Fue aprobada por el Comité Internacional de Prácticas de Auditoría y se emiten como suplementos a las NIAS, con la intención de ayudar al auditor a implementar dichas normas.

E

Ejecución (fase de): Fase de la auditoría de gestión focalizada, básicamente, en la obtención de evidencia suficiente y competente sobre los asuntos significativos (líneas de auditoría) aprobados en el plan de auditoría.

Entidad (1): En teoría de Base de Datos es una representación de un objeto individual concreto del mundo real. En una base de datos informatizada las entidades representan registros de una tabla de dicha base.

Entidad (2): Institución a la que se le realiza una auditoría de sistemas.

Entidad débil: En teoría de Base de Datos, una entidad que debe su permanencia en la base y su razón de ser a la existencia de otra entidad interrelacionada de la misma base. A esta última se le conoce como “entidad fuerte”.

F

Formas normales: Conjunto de reglas que se aplican al diseño de una base de datos relacional, cuyo objetivo es eliminar redundancia e inconsistencias en la base. Para más información sobre éstas características no deseables en una base de datos, consultar el apartado **2.2.3.2 Aplicación de formas normales**.

I

Interrelación: En teoría de Base de Datos, enlace entre las entidades de dos conjuntos de entidades diferentes, el cual posibilita la actualización y eliminación en cascada de los registros relacionados. Una interrelación puede darse de tres tipos: Uno a varios, uno a uno y varios a varios. De estos tres tipos, la interrelación uno a varios es la deseable en una implementación por computadora.

L

Llave primaria: Ver **clave principal**.

M

Método: En Programación Orientada a Objetos, un proceso que realiza un objeto cuando éste recibe un mensaje o invocación.

Modelo Conceptual de Base de Datos: Método de diseño de bases de datos, que consiste en buscar las entidades que describen los objetos que intervienen en el problema a modelar y las interrelaciones entre esas entidades.

Modelo Entidad – Relación de Base de Datos: Ver **Modelo Conceptual de Base de Datos**.

Modelo Lógico de Base de Datos: Ver **Modelo Relacional de Base de Datos**.

Modelo Relacional de Base de Datos: Método de diseño de bases de datos, que consiste en definir estructuras de datos para las entidades del problema (ver **relación**), operaciones para recuperar, derivar o modificar los datos (ver **álgebra relacional**) y una colección de reglas para definir la consistencia de la base (ver **normalización**).

Muestra: Es un subconjunto de la población que es tomado para un estudio de investigación. Véase también **población**.

MySQL: Es un gestor de bases de datos accesible de forma gratuita, disponible para las principales plataformas del mercado. Es muy utilizado dentro de los entornos de desarrollo de sistemas web con PHP.

N

NIAS: Normas Internacionales de Auditoría. Estas contienen los principios básicos y esenciales de la aplicación de la auditoría de los estados financieros.

Normalización: En gestión de Bases de Datos, es la aplicación de una serie formalizada de técnicas a una base de datos relacional para minimizar la inclusión

de información duplicada. La normalización simplifica la gestión de consultas y actualizaciones, aunque lo hace a cambio de crear un mayor número de tablas.

O

Objeto: En Programación Orientada a Objetos, una variable compuesta tanto por rutinas como por datos y que representa una instancia de una clase previamente codificada.

Observación: Esta referida a hechos o circunstancias significativos identificados durante el examen que pueden motivar oportunidades de mejoras. Si bien el resultado obtenido adquiere la denominación de hallazgo, para fines de presentación en el informe se convierte en observación.

P

Papeles de trabajo: Documentos que contienen la evidencia que respalda los hallazgos, observaciones, opiniones de funcionarios responsables de la entidad examinada, conclusiones y recomendaciones del auditor. Deben incluir toda la evidencia que se haya obtenido durante la auditoría.

PHP: Un lenguaje de script de código abierto utilizado con documentos HTML para ejecutar funciones interactivas del lado del servidor. PHP se ejecuta en todos los principales sistemas operativos y se utiliza principalmente con servidores web Linux y UNIX o con servidores Windows dotados de módulos de ampliación software.

Planeamiento (fase): Fase de la auditoría durante la cual el auditor se aboca a la identificación de qué examinar, cómo, cuándo y con qué recursos, así como la determinación del enfoque de la auditoría, objetivos, criterios y estrategia.

Plan de Auditoría: Tiene por propósito definir el alcance global de la auditoría de gestión, en términos de objetivos generales y objetivos específicos por áreas que serán materia de examen. Este documento incluye:

- Origen de la acción
- Objetivos de la auditoría
- Alcance de la auditoría, especificando períodos
- Áreas que serán examinadas, incluye objetivos específicos y alcance
- Criterios de auditoría a utilizarse
- Recursos de personal y especialistas que se necesitan
- Información administrativa
 - o Presupuesto de tiempo
 - o Informes a emitir y fechas de entrega
 - o Formato tentativo del informe.

Población: Total de individuos o elementos que representan determinadas características o manifestaciones que ameriten ser sujetas a un estudio de investigación.

Primera Forma Normal: Método de normalización en el cual se busca que cada campo de las tablas contenga información unívoca y no repetitiva. A esta característica se le llama atomicidad de campos.

Programa de auditoría: Documento, preparado por el auditor encargado y el supervisor encargado, donde se señala las tareas específicas que deben ser cumplidas por el equipo de auditoría para llevar a cabo el examen, así como los responsables de su ejecución y los plazos fijados para cada actividad.

Programación Orientada a Objetos (POO): Paradigma de programación en el que la aplicación se ve como una colección de objetos discretos que son colecciones autocontenidas de estructuras de datos y rutinas que interactúan con otros objetos.

R

Relación: En la Teoría de Bases de Datos relacionales, notación que distingue a una entidad, donde se destaca tanto la entidad como sus atributos. Esta notación es útil en el momento de aplicar formas normales y el álgebra relacional.

S

Segunda Forma Normal: Método de normalización que se aplica sobre una base de datos que cumple con la primera forma normal, y que separa dicha base en nuevas tablas por el procedimiento de definir dependencias funcionales, entre los campos llave primaria de las tablas y los demás atributos de la misma tabla.

T

Técnicas de auditoría: Son métodos prácticos de investigación y prueba que utiliza el auditor para obtener evidencia necesaria que fundamente su opinión.

U

UML: Acrónimo de Lenguaje Unificado de Modelado por sus siglas en inglés. Un lenguaje desarrollado por Grady Booch, Ivar Jacobson y Jim Rumbaugh, de Rational Software, que puede utilizarse para especificar, construir y documentar sistemas software o de otro tipo, como por ejemplo, modelos de negocio. La notación UML proporciona una base común para el diseño orientado a objetos, al proporcionar descripciones de conceptos de modelado, incluyendo clases de objetos, interfases y responsabilidades.