

**PROPUESTA DE HERRAMIENTA SOFTWARE DE APOYO EN  
EL APRENDIZAJE DE LA AUDITORÍA DE SISTEMAS DE  
INFORMACIÓN BASADO EN EL ESTUDIO DE CASOS**

**ALEXANDER CALDERON ROJAS**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS  
BUCARAMANGA**

**2004**

**PROPUESTA DE HERRAMIENTA SOFTWARE DE APOYO EN  
EL APRENDIZAJE DE LA AUDITORÍA DE SISTEMAS DE  
INFORMACIÓN BASADO EN EL ESTUDIO DE CASOS**

**ALEXANDER CALDERON ROJAS**

**Trabajo de grado para optar el título de  
Ingeniero de Sistemas**

**Director**

**JORGE HERRERA CASTILLO**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS  
BUCARAMANGA**

**2004**

## DEDICATORIA

**A** *Mis padres quienes con su apoyo han iluminado permanentemente mi vida con su amor, ejemplo y confianza. A mi familia, y todos aquellos que me han colaborado y compartido las labores que he realizado.*

ALEXANDER CALDERON ROJAS

## **AGRADECIMIENTOS**

Estoy seguro que el camino que un día empecé a recorrer no está terminado, por eso agradezco a todas aquellas personas que han me han brindado su apoyo, confianza y ánimo para continuar hacia adelante.

A mi familia que con su paciencia y colaboración hizo posible el desarrollo de este proyecto.

Al Ing. Jorge Herrera Castillo que con su apoyo, aportes y experiencia profesional fue parte fundamental en el desarrollo de este proyecto.

Al Grupo de investigación de Sistemas y Tecnologías de Información – STI que permitió la búsqueda de nuevos retos y acceso a todos los materiales bibliográficos.

A AUDISIS LTDA – Firma especializada en controles, seguridad y auditoría de sistemas, por su colaboración al brindar material de ayuda.

A mis compañeros de Universidad que han compartido sus conocimientos y han apoyado mis ideas permitiendo que esta etapa sea una experiencia feliz e inolvidable, a todos ellos les digo “gracias”.

# CONTENIDO

	pág.
INTRODUCCIÓN	1
PARTE I - FUNDAMENTOS	2
CAPITULO 1. ASPECTOS GENERALES DE LA AUDITORÍA	3
1.1 CONCEPTO DE AUDITORÍA	3
1.2 PRINCIPALES TAREAS Y BENEFICIOS QUE PROPORCIONA LA AUDITORÍA	4
1.3 CLASES DE AUDITORÍA	5
1.3.1 Por la procedencia del auditor	5
1.3.1.1 Auditoría Interna	5
1.3.1.2 Auditoría Externa	6
1.3.2 Por los objetos auditables	6
1.3.2.1 Auditoría financiera	6
1.3.2.2 Auditoría operacional	6
1.3.2.3 Auditoría de sistemas o auditoría informática	6
1.3.2.4 Auditoría física	7
1.3.2.5 Auditoría de comunicación y redes	7
1.3.2.6 Otros tipos de auditoría	7
1.4 TECNICAS EMPLEADAS EN AUDITORÍA	7

1.4.1.1 Cuestionarios	7
1.4.1.2 Entrevistas	8
1.4.1.3 Checklist	8
1.4.1.4 Observación	9
1.4.1.5 Trazas y/o huellas	9
1.4.1.6 Confirmación	9
1.4.1.7 Análisis	9
1.4.1.8 Técnicas asistidas por computador	9
1.4.1.9 Otras técnicas	10
1.5 METODOLOGÍAS DE AUDITORÍA	10
1.5.1 Tipos de metodologías	11
1.5.1.1 Metodologías cuantitativas	11
1.5.1.2 Metodologías Cualitativas	11
1.5.2 Estándares de auditoría	11
1.5.2.1 COBIT	12
1.5.2.2 COSO	12
1.5.2.3 AICPA (SAS)	13
1.5.2.4 IFAC (NIA)	13
1.5.2.5 SAC	13
1.5.2.6 ISO	14
1.6 EL INFORME DE AUDITORÍA	14
1.6.1 La evidencia	14
1.6.2 Los papeles de trabajo	15
1.6.3 Pruebas de audioria	15
1.6.3.1 Pruebas de Cumplimiento	15
1.6.3.2 Pruebas Sustantivas	16
1.6.3.3 Riesgos y controles	16

1.6.3.4 El informe	17
1.7 RESUMEN	19
CAPITULO 2. AUDITORÍA DE SISTEMAS DE INFORMACION Y SU ENTORNO	20
2.1 SISTEMAS DE INFORMACION Y TECNOLOGIAS DE INFORMACION EN EL ENTORNO ORGANIZACIONAL	20
2.2 RELACION DE LA AUDITORÍA DE SISTEMAS DE INFORMACION CON EL ENTORNO ORGANIZACIONAL	21
2.3 DEFINICION DE LA AUDITORÍA DE SISTEMAS DE INFORMACION	22
2.4 ESTRUCTURA ORGANIZACIONAL Y FUNCIONES DE LA AUDITORÍA DE SISTEMAS DE INFORMACION	23
2.4.1 Ubicación jerarquica de la funcion	23
2.5 METODOLOGIA PARA EL DESARROLLO E IMPLANTACION DE LA AUDITORÍA DE SISTEMAS DE INFORMACION	23
2.5.1 Flujo de trabajo: planificacion y estudio	24
2.5.1.1 Investigación Preliminar	25
2.5.1.2 Estudio de información Relevante	26
2.5.1.3 Estudio de los sistemas actuales	27
2.5.1.4 Revisión Detallada	27
2.5.1.5 Definición y Organización	28
2.5.1.6 Diseño del Modelo de Auditoría	28
2.5.1.7 Definición del plan	29
2.5.1.8 Revisión y Aprobación	29

2.5.2 Flujo de trabajo: exámen y evaluación	30
2.5.2.1 Revisión formal por áreas o procesos	30
2.5.2.2 Realización de pruebas y obtención de evidencias	30
2.5.2.3 Documentación de hallazgos y preparación de informes	31
2.5.3 Flujo de trabajo: informe – diagnóstico	31
2.5.3.1 Seguimiento de las observaciones de auditoría	31
2.6 PERSPECTIVAS DE LA AUDITORÍA DE SISTEMAS DE INFORMACIÓN	32
2.6.1 En el campo educativo	32
2.6.2 En el campo práctico aplicado	32
2.7 RESUMEN	32
CAPITULO 3. EL ESTUDIO DE CASOS COMO TECNICA DIDÁCTICA	33
3.1 EL ESTUDIO DE CASOS COMO ESTRATEGIA DIDÁCTICA	33
3.2 DEFINICIÓN Y CARACTERÍSTICAS DE LA TÉCNICA	34
3.2.1 Modelo centrado en el análisis de casos	36
3.2.2 Modelo centrado en enseñar a aplicar principios y normas	36
3.2.3 Modelo centrado en el entrenamiento	36
3.2.3.1 Casos centrados en el estudio de descriptores	37
3.2.3.2 Casos de resolución de problemas	37
3.2.3.3 Casos centrados en la simulación	38
3.3 SELECCIÓN Y ELABORACIÓN DE LOS CASOS	39
3.3.1 Búsqueda y selección de los casos	39
3.3.1.1 Entrevista a un profesional con experiencia	39
3.3.1.2 Estudio de documentos conservados en archivos de la profesión	39
3.3.1.3 Escritos que se refieren a acontecimientos personales o profesionales	40

3.4 ORGANIZACIÓN DE LA TÉCNICA	40
3.4.1 Fase preliminar	41
3.4.2 Fase eclosiva	41
3.4.3 Fase de análisis	41
3.4.4 Fase de conceptualización	42
3.5 APRENDIZAJES QUE FOMENTA LA TÉCNICA DE ESTUDIO DE CASOS	42
3.6 LA EVALUACIÓN DE LA TÉCNICA DE ESTUDIO DE CASOS	43
3.7 DIFICULTADES Y BARRERAS PARA PODER EN PRACTICA LA TECNICA	44
3.8 RESUMEN	45
CAPITULO 4. APRENDIZAJE DE LA AUDITORÍA DE SISTEMAS DE INFORMACIÓN APOYADONDA SE EN LA DIDÁCTICA DEL ESTUDIO DE CASOS	47
4.1 INTRODUCCIÓN	47
4.2 DESCRIPCIÓN DEL PROBLEMA	47
4.3 DESCRIPCIÓN DE LA PROPUESTA: UNA HERRAMIENTA SOFTWARE	49
4.3.1 Dimensión de la materia	50
4.3.2 Dimensión del estudiante	50
4.3.3 Dimensión del estudio de casos	50
4.4 OBJETIVOS DEL PROYECTO	51
4.4.1 Objetivo general	51
4.4.2 Objetivos específicos	51

4.5 IMPACTO	52
PARTE II. DESARROLLO DE LA HERRAMIENTA SOFTWARE	54
CAPITULO 1. LA FASE DE INICIO	55
5.1 PLANIFICACIÓN DE LA FASE DE INICIO	55
5.2 MODELADO DEL NEGOCIO	56
5.2.1 Identificación de los procesos del Negocio	56
5.2.2 Identificación de Usuarios y elementos implicados en el proceso del negocio.	57
5.2.3 Establecer las acciones necesarias para realizar el proceso de negocio.	57
5.2.4 Construcción de un diagrama de actividades que represente el proceso de negocio	58
5.2.5 Lista de Actividades	60
5.2.6 Reglas del negocio	60
5.3 MODELADO DE REQUISITOS	61
5.3.1 Flujo de Trabajo de requisitos	61
5.3.1.1 Lista de Características o requisitos candidatos	61
5.3.1.2 Identificación de Actores	62
5.3.1.3 Casos de Uso Candidatos	62
5.3.1.4 Identificación de Riesgos Críticos	63
5.3.1.5 Contexto de la herramienta software	63
5.3.1.6 Descripción de los Casos de Uso	64
5.3.1.7 Detalle de casos de Uso	67
5.3.1.8 Modelo de casos de Uso	72
5.3.1.9 Captura de Requisitos adicionales	73

5.4 MODELADO DE ANÁLISIS Y DISEÑO	74
5.4.1 Flujo de Trabajo de Análisis	74
5.4.1.1 Identificación de Clases del análisis	74
5.4.1.2 Descripción de interacciones entre objetos del caso de Uso	80
5.4.1.3 Identificación de paquetes del análisis	80
5.4.1.4 Dependencias entre paquetes del análisis y capas	80
5.4.2 Flujo de trabajo de Diseño	81
5.4.2.1 Diseño de clases	81
5.4.2.2 Identificación y distribución de subsistemas	81
5.4.2.3 Definición de dependencias y capas entre subsistemas	82
5.5 EVALUACIÓN DE LA FASE DE INICIO	83
CAPITULO 6. FASE DE ELABORACIÓN	84
6.1 PLANIFICACIÓN DE LA FASE DE ELABORACIÓN	84
6.2 MODELO DEL NEGOCIO	85
6.2.1 Modelo de casos de uso del negocio	85
6.2.2 Descripción de casos de uso del negocio	87
6.3 MODELADO DE REQUISITOS	88
6.3.1 Flujo de trabajo de Requisitos	88
6.3.1.1 Modelo de casos de uso	89
6.3.1.2 Casos de uso detallados	90
6.3.1.3 Identificación de riesgos críticos	93
6.4 MODELO DE ANÁLISIS Y DISEÑO	94

6.4.1 Flujo de trabajo de analisis	94
6.4.1.1 Identificación de clases de análisis	94
6.4.1.2 Descripción de interacciones entre objetos del caso de uso	98
6.4.1.3 Identificación de paquetes del análisis	100
6.4.2 Flujo de trabajo de diseño	101
6.4.2.1 Diseño de la interfaz	101
6.4.2.2 Diseño de la Base de datos	103
6.4.2.3 Capa Software del sistema	109
6.4.3 Flujo de trabajo de implementación	109
6.5 EVALUACION DE LA FASE DE ELABORACION	114
CAPITULO 7. FASE DE CONSTRUCCION	115
7.1 PLANFICACION DE LA FASE DE CONTRUCCION	115
7.1.1 Gestion de identificacion	116
7.1.1.1 Implementación	116
7.1.1.2 Pruebas	120
7.1.2 Gestion de creación de casos	123
7.1.2.1 Implementación	123
7.1.2.2 Pruebas	127
7.1.3 Planeación caso de auditoría	128
7.1.3.1 Implementación	129

7.1.3.2 Pruebas	134
7.1.4 Exámen auditoría	135
7.1.4.1 Implementación	137
7.1.4.2 Pruebas	141
7.1.5 Informe de auditoría	142
7.1.5.1 Implementación	143
7.1.5.2 Pruebas	144
7.1.6 Gestión de estudio de caso	145
7.1.6.1 Implementación	147
7.1.6.2 Pruebas	149
7.2 EVALUACIÓN DE LA FASE DE CONSTRUCCIÓN	149
CONCLUSIONES	150
BIBLIOGRAFÍA	153
ANEXOS	192

## LISTA DE TABLAS

Tabla 1. Comparación entre metodologías cuantitativas y cualitativas.	11
Tabla 2. Comparación entre pruebas de cumplimiento y sustantivas	16
Tabla 3. Tareas básicas en la investigación preliminar.	26
Tabla 4. Tareas básicas en el estudio de información relevante.	26
Tabla 5. Tareas básicas en el estudio de los sistemas actuales.	27
Tabla 6. Tareas básicas en la revisión detallada.	27
Tabla 7. Tareas básicas en la Definición y organización.	28
Tabla 8. Tareas básicas en el diseño del modelo de auditoría.	28
Tabla 9. Tareas básicas del proceso de planeación	29
Tabla 10. Tareas básicas del proceso de planeación	29
Tabla 11. Casos de uso candidatos	62
Tabla 12. Identificación de riesgos críticos para la fase de inicio	63
Tabla 13. Descripción del caso de uso - Identificación de usuario con login.	64
Tabla 14. Descripción del caso de uso - Crear caso de auditoría.	65
Tabla 15. Descripción del caso de uso – Establecer parámetros.	65
Tabla 16. Descripción del caso de uso – Registro del usuario.	65
Tabla 17. Descripción del caso de uso – Establecer plan de auditoría.	66
Tabla 18. Descripción del caso de uso – Establecer hojas de trabajo.	66
Tabla 19. Descripción del caso de uso – Crear diagnóstico.	66
Tabla 20. Descripción del caso de uso – Estudiar caso	67
Tabla 21. Detalle del caso de uso Crear caso de auditoría.	68
Tabla 22. Detalle del caso de uso – Identificación de usuario.	70
Tabla 23. Detalle del caso de uso – Estudiar caso de auditoría.	72
Tabla 24. Requisitos no funcionales	73
Tabla 25. Descripción del proceso de negocio – Aprender metodologías.	87
Tabla 26. Descripción del proceso de negocio – Desarrollar caso de estudio.	87
Tabla 27. Descripción del proceso de negocio – Estudiar caso de estudio.	88

Tabla 28. Descripción del proceso de negocio – Desarrollar discusión del caso.	88
Tabla 29. Detalle del caso de uso – Establecer plan de auditoría.	91
Tabla 30. Descripción del caso de uso – Examen auditado.	92
Tabla 31. Descripción del caso de uso – crear diagnostico.	93
Tabla 32. Identificación de riesgos para la fase de elaboración.	94
Tabla 33. Entidades presentes en el submodelo entidad – relación que participan en la creación de un caso de auditoría	104
Tabla 34. Entidades presentes en el submodelo entidad – relación que participan en la creación del plan de auditoría.	105
Tabla 35. Entidades presentes en el submodelo entidad - relación que participan en la ejecución de un examen en el auditado.	106
Tabla 36. Entidades presentes en el submodelo entidad – relación que participan en la elaboración del informe de auditoría.	107
Tabla 37. Entidades presentes en el submodelo entidad – relación que participan en el estudio de un caso de auditoría.	109
Tabla 38. Iteraciones de la fase de construcción	115
Tabla 39. Escenario 1 de prueba para el ingreso a la herramienta.	120
Tabla 40. Escenario 2 de prueba para el ingreso a la herramienta.	120
Tabla 41. Escenario 1 de prueba para el registro de usuarios.	121
Tabla 42. Escenario 2 de prueba para el registro de usuarios.	122
Tabla 43. Escenario 1 de prueba para la creación de un caso de auditoría.	127
Tabla 44. Escenario 2 de prueba para la creación de un caso de auditoría	128
Tabla 45. Escenario 1 de prueba para creación de plan de auditoría.	134
Tabla 46. Escenario 2 de prueba para creación de plan de auditoría.	135
Tabla 47. Escenario de prueba 3 para creación de plan de auditoría.	135
Tabla 48. Escenario de prueba 1 para registro en hojas de trabajo.	141
Tabla 49. Escenario de prueba 2 para registro en hojas de trabajo.	142
Tabla 50. Escenario de prueba 1 para creación de informe.	144
Tabla 51. Escenario de prueba 2 para creación de informe.	144
Tabla 52. Escenario de prueba 1 para creación de notas del caso en estudio.	149
Tabla 53. Escenario 2 de prueba para creación de notas del caso en estudio.	149

## LISTA DE FIGURAS

Figura 1. Relación entre Riesgos y controles	17
Figura 2. Tipos de opinión en un informe de auditoría	18
Figura 3. Partes o Flujos de Trabajo en la auditoría	24
Figura 4. Etapas o pasos en el estudio y planeacion de la auditoría de S.I	25
Figura 5. Esquema de los modelos metodológicos del estudio de casos	46
Figura 6. Factores o dimensiones a tener en cuenta para el aprendizaje de auditoría	50
Figura 7. Esquema metodologico del proceso de auditoría	56
Figura 8. Flujo de actividades del proceso de Negocio	59
Figura 9. Contexto de la herramienta software	64
Figura 10. Diagrama inicial de casos de uso	67
Figura 11. Crear caso de auditoría.	68
Figura 12. Diagrama de estados para el caso crear nuevo caso de auditoría	68
Figura 13. Casos de uso identificación de usuario y registro	69
Figura 14. Diagrama de actividades para identificación del usuario	70
Figura 15. Casos de uso Consulta y estudio de casos de auditoría	71
Figura 16. Flujo de sucesos para el caso de uso Estudiar caso auditoría	71
Figura 17. Modelo de casos de uso	73
Figura 18. Realización del caso de uso crear caso de auditoría	74
Figura 19. Clases de análisis para crear caso de auditoría	75
Figura 20. Realización del caso de uso identificación de usuario	75
Figura 21. Clases del análisis para realizar identificación de usuario	76
Figura 22. Realización del caso de uso Estudiar caso de Auditoría	76
Figura 23. Clases de análisis para realizar Estudiar caso Auditoría	77
Figura 24. Diagrama de clases de una realización del caso de uso Crear Caso de estudio de auditoría	78

Figura 25. Diagrama de actividad para el caso de uso identificación usuario invalido	79
Figura 26. Diagrama de colaboración para el caso de uso estudiar caso auditoría	79
Figura 27. Identificación de paquetes del análisis a partir de los casos de uso	80
Figura 28. Dependencia entre capas y paquetes del análisis	81
Figura 29. La clase de diseño usuario y caso de auditoría con sus atributos y operaciones	81
Figura 30. Identificación de subsistemas a partir de paquetes del análisis existentes	82
Figura 31. Dependencias y capas de algunos de los subsistemas de la herramienta	82
Figura 32. Caso de Uso estudiar metodologías.	85
Figura 33. Desarrollar caso de estudio	86
Figura 34. Caso de uso Estudiar caso de auditoría	86
Figura 35. Modelo de casos de uso ampliado	89
Figura 36. Casos de uso para establecer plan de auditoría	90
Figura 37. Diagrama de actividades para establecer plan de auditoría	90
Figura 38. Casos de uso para realizar examen auditado	91
Figura 39. Diagrama de actividades para el caso de uso realizar examen	92
Figura 40. Caso de uso crear diagnóstico y realizar informe.	93
Figura 41. Realización de caso de uso Definir plan de auditoría.	94
Figura 42. Clases del análisis para Definir plan de auditoría.	95
Figura 43. Realización del caso de uso realizar examen auditado.	96
Figura 44. Clases del análisis para realizar examen auditado.	96
Figura 45. Realización del caso de uso crear diagnóstico.	97
Figura 46. Clases del análisis para realizar crear diagnóstico .	97
Figura 47. Diagrama de colaboración para establecer plan de auditoría.	98
Figura 48. Diagrama de colaboración para realizar examen.	99
Figura 49. Diagrama de colaboración para crear diagnóstico.	99
Figura 50. Identificación de paquetes del análisis a partir de los casos de uso	100
Figura 51. Los paquetes de servicio, datos auditado, plan auditoría, examen, informe, ubicados dentro del paquete Gestión de creación de casos	100
Figura 52. Diseño general de la interfaz de la herramienta.	102

Figura 53. Clases que maneja la aplicación y generadas por el AppWizard.	103
Figura 54. Submodelo entidad relación – Creación caso de Auditoría	105
Figura 55. Submodelo entidad relación – Plan de auditoría.	106
Figura 56. Submodelo entidad relación – Examen auditado	107
Figura 57. Submodelo entidad relación – Informe Auditoría.	108
Figura 58. Submodelo entidad relación – Estudio de caso de auditoría	108
Figura 59. Creación del esqueleto de la aplicación usando el appWizard.	110
Figura 60. Interfaz para el control de ingreso a la herramienta	118
Figura 61. Interfaz para el registro de usuarios.	120
Figura 62. Interfaz Principal de la herramienta software.	122
Figura 63. Interfaz para la creación de un caso de auditoría	123
Figura 64. Interfaz para establecimiento del plan de auditoría	129
Figura 65. Interfaz para la definición de actividades.	132
Figura 66. Interfaz para definición de recursos.	133
Figura 67. Interfaz para definición de Técnicas .	133
Figura 68. Interfaz para asignar actividades a participantes .	134
Figura 69. Interfaz para establecer examen de auditoría (hoja de trabajo).	136
Figura 70. Interfaz para selección de actividad a trabajar	136
Figura 71. Interfaz para registro de Notas de revisión del trabajo realizado	140
Figura 72 Interfaz para identificación de riesgos y controles.	141
Figura 73. Interfaz para creación de informe de auditoría.	142
Figura 74. Interfaz para estudio de caso de auditoría.	145
Figura 75. Interfaz para apertura de un caso de auditoría.	146
Figura 76. Interfaz para registro de notas del caso en estudio	147

## **LISTA DE APENDICES**

APENDICE A - El Proceso Unificado de Desarrollo de Software	156
APENDICE B - El lenguaje Unificado de Modelado	167

## **LISTA DE ANEXOS**

ANEXO A - Plantillas para descripción de procesos del negocio y casos de uso	192
ANEXO B - Manual del usuario	193

## GLOSARIO

**APRENDIZ:** Reemplaza el término “estudiante”, debido a que el que aprende no solo estudia, sino que practica, reflexiona y gracias a ello aprende a aprender.

**AUDITOR:** Profesional cuya labor implica el ejercicio de la auditoría.

**CASO DE AUDITORÍA:** Se refiere a la organización y recopilación de la información de todo el proceso llevado en la auditoría.

**CONCLUSIONES:** Son juicios del auditor, de carácter profesional, basados en las observaciones formuladas como resultado del examen.

**CONTROL:** Es la acción que se ejerce sobre una causa de riesgo con el fin de reducir su probabilidad de ocurrencia o el impacto que puede generar.

**EFFECTIVIDAD:** Es un concepto administrativo que se refiere al grado de cumplimiento de la misión y objetivos de una entidad.

**EFICIENCIA:** Virtud, facultad para lograr un efecto determinado, es decir, es el poder lograr lo planeado con los menores recursos posibles.

**HALLAZGO DE AUDITORÍA:** Concepto utilizado para describir el resultado de la comparación que se realiza entre un criterio y la situación actual encontrada durante el examen a un área, actividad, operación o circunstancias en las cuales el criterio fue aplicado.

**HOJA DE TRABAJO:** Documentos que contienen el trabajo o evidencia que respalda los hallazgos, observaciones, opiniones, conclusiones y recomendaciones del auditor.

**PRUEBA DE AUDITORÍA:** Consiste en comprobar que determinados procedimientos, normas o controles cumplen o funcionan con lo previsto y esperado.

**PLAN DE AUDITORÍA:** Corresponde a la definición y formalización del proyecto y abarca las actividades desarrolladas por el auditor: Definición de alcance, objetivos, personal y recursos necesarios.

**REVISOR:** Profesional cuya labor es apoyar la labor del auditor.

**RIESGO:** Es la probabilidad de que una amenaza llegue a suceder por vulnerabilidad.

**TECNICAS DE AUDITORÍA:** Son los instrumentos metodológicos que utiliza el auditor para abordar el examen del objeto auditado.

## RESUMEN

**TITULO:** PROPUESTA DE HERRAMIENTA SOFTWARE DE APOYO EN EL APRENDIZAJE DE LA AUDITORÍA DE SISTEMAS DE INFORMACIÓN BASADO EN EL ESTUDIO DE CASOS\*.

**AUTOR:** ALEXANDER CALDERÓN ROJAS\*\*.

**PALABRAS CLAVES:** Auditoria de sistemas de información. Didáctica del estudio de casos. Proceso Unificado de Desarrollo del Software.

## DESCRIPCIÓN

Los cambios propiciados por el desarrollo de la ciencia y la tecnología hacen que el proceso de aprendizaje y los escenarios de trabajo evolucionen, dando como resultado una permanente renovación del conocimiento. El área de auditoria de sistemas de información no es ajena a estos cambios y en búsqueda de responder a las necesidades de formación de futuros profesionales, se plantea como alternativa una propuesta en donde se logre la asimilación de conocimientos teóricos con base en el estudio de casos.

Se presenta una propuesta de herramienta software que permite la recopilación de información del proceso de auditoria agrupado en tres flujos de trabajo: la planeación, el exámen y el informe o diagnóstico, los cuales constituyen un caso

---

\* Trabajo de investigación.

\*\* Facultad de Ingenierías Físico – Mecánicas. Escuela de Ingeniería de Sistemas.

documentado. Este modelo le servirá al estudiante o profesional en formación para estudiar, evaluar y valorar el proceso de intervención llevado a cabo por un auditor.

El presente trabajo de investigación emplea el proceso unificado de desarrollo del software recorriendo las fases de inicio, elaboración y construcción, que proporcionan un producto software en su versión operativa inicial para su evaluación y prueba en el ámbito académico. Y se apoyará en la didáctica del estudio de caso para propiciar un ambiente de análisis, síntesis y evaluación de la información.

## **ABSTRACT**

**TITLE:** PROPOSAL OF SOFTWARE TOOL OF SUPPORT IN THE LEARNING OF THE AUDIT OF INFORMATION SYSTEMS BASED IN THE STUDY OF CASES\*.

**AUTHOR:** ALEXANDER CALDERÓN ROJAS\*\*.

**KEY WORDS:** Audit of information systems. Didactics of the study of cases. Unified Process of Software Development.

### **DESCRIPTION**

The changes propitiated by the development of the science and the technology make that the learning process and the work scenarios evolve, giving a permanent renovation of the knowledge as result. The area of audit of information systems is not unaware to these changes and in search of responding to the necessities of formation of professional futures; it thinks about as alternative a proposal where the assimilation of theoretical knowledge is achieved with base in the study of cases.

A proposal of software tool is presented that allows the compilation of information of the audit process contained in three work flows: the gliding, the exam and the report or diagnostic, those which constitute a documented case. This model will be good for the student or professional in formation, to study, to evaluate and to value the intervention process carried out by an auditor.

---

\* Investigation work.

\*\*Faculty of Physical - Mechanical Engineering. School of Systems Engineering.

The present investigation work uses the unified process of development of the software going through the beginning, elaboration and construction phases, which provide a software product in its initial operative version for its evaluation and test in the academic environment. And it will lean on in the didactics of the case study to propitiate an environment of analysis, synthesis and evaluation of the information.

## INTRODUCCION

La Escuela de Ingeniería de Sistemas de la Universidad Industrial de Santander afronta nuevos cambios, propiciados por el desarrollo de las tecnologías de información y comunicación que hacen que el proceso académico y los escenarios de trabajo evolucionen con el fin de responder a los desafíos del mundo actual.

Estos cambios en la academia involucran también el apoyo didáctico, facilitando contenidos que permiten al estudiante desarrollar habilidades como el análisis, síntesis, y evaluación de la información para enfrentar problemas o desafíos presentados en nuestra sociedad.

El desarrollo de este proyecto es un aporte al proceso académico, ya es una propuesta basada en la orientación que ofrece la metodología didáctica del estudio de casos para apoyar el aprendizaje de la auditoría de sistemas.

El contenido del documento esta organizado en dos partes: La parte I expone los aspectos generales del proyecto y el fundamento teórico necesario para el cumplimiento de los objetivos. La parte II expone los detalles necesarios para la construcción de la herramienta software propuesta, empleando el proceso unificado de desarrollo del software.

Finalmente se presentan las conclusiones del trabajo realizado. Así como los apéndices y anexos respectivos.

# PARTE I

## FUNDAMENTOS

En esta parte se presenta los aspectos fundamentales sobre el proyecto. Su objetivo es exponer con claridad los detalles más significativos de la investigación, estableciendo unas bases teóricas que permitan el cumplimiento de los objetivos que se desean alcanzar.

Los objetivos específicos de esta parte son:

- ✓ Exponer los aspectos generales de la auditoría.
- ✓ Enfocar el estudio sobre la auditoría de sistemas de información.
- ✓ Establecer y exponer los aspectos metodológicos del Estudio de casos como técnica didáctica.
- ✓ Formular y describir detalladamente los objetivos del proyecto.
- ✓ Establecer el marco metodológico: el aprendizaje de la auditoría de sistemas de información apoyándose en la didáctica del estudio de casos.

De esta parte se espera que se comprendan los conceptos e ideas expuestas en cada capítulo, de manera que al establecer el marco metodológico para el desarrollo del proyecto no se tengan mayores dificultades en su entendimiento.

# Capítulo 1

## ASPECTOS GENERALES DE LA AUDITORÍA

Para poder incursionar en el campo de la auditoría es necesario enunciar las bases fundamentales que permitan una comprensión global de los términos que se manejan. Por eso, este capítulo está destinado a exponer diversos conceptos de la auditoría, lo cual marcará el comienzo de un proceso de investigación que permite el desarrollo de presente trabajo de grado.

### 1.1 CONCEPTO DE AUDITORÍA

Históricamente, la auditoría (término que proviene del latín auditorius, de este proviene la palabra auditor que significa todo aquel que tiene la virtud de oír), surgió de la necesidad de revisar registros contables, ya que estos se consideraban parte primordial de la organización.

Conceptualmente la auditoría, toda y cualquier auditoría según Piattini y del Peso<sup>1</sup>, es la actividad consistente en la emisión de una opinión profesional sobre si el objeto sometido a análisis presenta adecuadamente la realidad que pretende reflejar y lo cumple las condiciones que le han sido prescritas. Con frecuencia el término se ha empleado incorrectamente y se le ha considerado como una evaluación para detectar errores y fallas. Pero indiscutiblemente la auditoría<sup>2</sup> no solo detecta errores, es un examen que se realiza para evaluar la eficiencia y eficacia del objeto (sección, organismo) sometido a estudio.

---

<sup>1</sup> Piattini y del Peso (1998:4)

<sup>2</sup> Se sugiere leer los libros listados en las notas bibliográficas debido a que solo se han seleccionado y expuesto algunos aspectos necesarios para su entendimiento.

Antes de continuar, conviene distinguir los términos: *consultoría* y *control interno* ya que a menudo tienden a confundirse con el concepto de auditoría. El término consultoría consiste en dar asesoramiento o consejo sobre lo que se ha de hacer o cómo llevar adecuadamente una determinada actividad para obtener los fines deseados. El término control interno se refiere a la actividad o acción para prevenir, corregir errores o irregularidades que pueden afectar el funcionamiento de una organización.

## **1.2 PRINCIPALES TAREAS Y BENEFICIOS QUE PROPORCIONA LA AUDITORÍA**

El proceso formal de auditoría se efectúa por solicitud de las empresas o de nuevos gobiernos establecidos y que requieren verificar las actividades realizadas con anterioridad. Así, las tareas mínimas de la auditoría serían:

- Estudiar y actualizar permanentemente las áreas susceptibles de revisión.
- Evaluación y verificación de las normas, políticas, procedimientos, controles y técnicas empleadas en las áreas requeridas por la alta dirección o responsables directos del negocio.
- Asegurar y verificar que los requerimientos de seguridad y de auditoría sean incorporados.
- La elaboración de informes sustentada en la obtención de evidencia suficientemente documentada y adecuada sobre la realidad reflejada.
- Emitir una opinión profesional, fundamentada y justificada por medio de unos procedimientos específicos tendientes a proporcionar una seguridad razonable de lo que se afirma.

Los beneficios que se obtienen dependen del tipo de auditoría que se va a desarrollar, pero se pueden destacar entre otros los siguientes:

- Examina y evalúa el control interno de la entidad, detectando oportunamente las debilidades que puedan presentarse y reportándolas, en forma coherente y clara a la dirección. En este sentido, la auditoría cumple con la función de control detectivo pero a su vez sirve de base para mejorar el sistema de control preventivo de la organización.
- Lleva a cabo un proceso de revisión de la organización con el fin de precisar: perdidas y deficiencias, mejores métodos, mejores formas de control, mejor uso de los recursos físicos y humanos.
- Asegurar la confiabilidad e integridad de la información.

### **1.3 CLASES DE AUDITORÍA**

**1.3.1 Por la procedencia del auditor.** Tomando como base la procedencia del auditor o del personal que ejecuta el trabajo, la auditoría se divide fundamentalmente en las siguientes categorías: Auditoría interna, Auditoría externa.

**1.3.1.1 Auditoría Interna.** La auditoría interna tiene la función de control al servicio de la dirección de la organización y suele ser un órgano staff. El auditor debe efectuar un estudio y evaluación de todas las actividades que cumplan los procedimientos, estándares y normas adecuados al control interno existente, que le sirvan de base para determinar el grado de confianza de los mecanismos implantados.

Como principales objetivos se pueden indicar los siguientes:

- La protección de los activos de la empresa.
- Controlar que todas las actividades se realizan cumpliendo los procedimientos y normas fijados, evaluar su bondad y asegurarse del cumplimiento de las normas legales.
- Asesorar sobre el conocimiento de las normas.

- Definir, implantar y ejecutar mecanismos y controles para comprobar el logro de los procesos efectuados.

**1.3.1.2 Auditoría Externa.** La auditoría externa es una actividad profesional, cuyo objetivo fundamental es el de examinar y evaluar una determinada realidad por personal externo al ente auditado, para emitir una opinión independiente, sobre el resultado de las operaciones y la validez técnica del sistema de control que esté operando en el área auditada.

Generalmente se contratan los servicios de auditores externos para certificar información financiera, cuando se trata de requisitos exigidos por las entidades externas, para efectos de relaciones comerciales o de contratos especiales.

**1.3.2 Por los objetos auditables.** La auditoría puede aplicarse al examen y evaluación de numerosos objetos, se puede hablar de auditoría financiera, auditoría operacional, auditoría física, auditoría de sistemas o auditoría informática, auditoría de comunicación y redes entre otras.

**1.3.2.1 Auditoría financiera.** Esta modalidad de auditoría, tiene como objeto de estudio el sistema contable y los correspondientes estados financieros, con miras a emitir opinión independiente sobre la razonabilidad financiera mostrada en los estados financieros del ente auditado.

**1.3.2.2 Auditoría operacional.** Este tipo de auditoría, tiene como objeto de estudio el proceso administrativo y las operaciones, con miras a emitir opinión sobre la habilidad de la gerencia para manejar el proceso administrativo y el grado de economicidad, eficiencia y efectividad de las operaciones del auditado.

**1.3.2.3 Auditoría de sistemas o auditoría informática.** La auditoría de sistemas o auditoría informática como es denominada en el ámbito internacional tiene como

objeto de estudio las áreas sistematizadas. Y tiene como objetivo, emitir una opinión independiente sobre la eficiencia en el uso los recursos, validez de la información, y efectividad de los controles que se presentan en el ente auditado.

**1.3.2.4 Auditoría física.** La auditoría física tiene como objeto de estudio todo objeto tangible (maquinaria, autos, plantas físicas etc.). Tiene como objetivo, emitir una opinión independiente sobre el funcionamiento de ese objeto tangible es decir asegurar que dicho objeto va a seguir prestando servicio siempre que se le necesite y de manera segura.

**1.3.2.5 Auditoría de comunicación y redes.** La auditoría de redes y comunicación como su nombre lo indica tiene como objeto de estudio las comunicaciones y las redes. Tiene como objetivo la administración, instalación operación y seguridad de las redes en un entorno de trabajo.

**1.3.2.6 Otros tipos de auditoría.** Sería pretencioso enumerar todos los tipos de auditoría posibles, ya que sobre la base del concepto de la auditoría, se daría origen a un nuevo tipo de auditoría. Por ejemplo, puede hablarse de auditoría del sector aéreo, auditoría social, auditoría económica, auditoría de la seguridad, auditoría del mantenimiento, auditoría educativa etc.

## **1.4 TECNICAS EMPLEADAS EN AUDITORÍA**

Las técnicas de auditoría son los instrumentos metodológicos que utiliza el auditor para abordar el examen del objeto auditado, durante el proceso de obtención de evidencias. Algunas de las técnicas utilizadas en auditoría son las siguientes.

**1.4.1.1 Cuestionarios.** Una auditoría se materializa recabando documentación de todo tipo. El trabajo de campo del auditor consiste en lograr toda la información necesaria para la emisión de un juicio global objetivo, siempre amparado en hechos demostrables.

Para esto, suele ser comenzar solicitando la cumplimentación de cuestionarios que se envían a personas concretas que el auditor cree adecuadas, sin que sea obligatorio que dichas personas sean las responsables oficiales de las diversas áreas a auditar.

Sobre esta base, se estudia y analiza la documentación recibida, de modo que tal análisis determine a su vez la información que deberá elaborar el propio auditor. El cruzamiento de ambos tipos de información es una de las bases fundamentales de la auditoría.

**1.4.1.2 Entrevistas.** La entrevista es una de las actividades personales más importante del auditor; en ellas, se recoge más información, y mejor matizada, que la proporcionada por medios propios puramente técnicos o por las respuestas escritas a cuestionarios.

La entrevista entre auditor y auditado se basa fundamentalmente en el recurso de interrogación para averiguar sobre la diversidad de hechos, datos y situaciones que el auditor necesita conocer durante el proceso, tales como políticas generales de la empresa, procedimientos, contingencias más relevantes, explicaciones sobre variaciones presentadas y en general sobre investigaciones verbales que el auditor considere oportunas.

**1.4.1.3 Checklist.** Las listas de chequeo o checklist son un conjunto de preguntas reelaboradas en función de los escenarios auditados. El auditor deberá aplicar la checklist de modo que el auditado responda clara y estrictamente. Se deberá interrumpir lo menos posible a éste, y solamente en los casos en que las respuestas se aparten sustancialmente de la pregunta. En algunas ocasiones, se hará necesario invitar a aquél a que exponga con mayor amplitud un tema concreto, y en cualquier caso, se deberá evitar absolutamente la presión sobre el mismo.

Los cuestionarios o checklist responden fundamentalmente a dos tipos de filosofía de calificación o evaluación:

- Checklist de rango: Contiene preguntas que el auditor debe puntuar dentro de un rango preestablecido (por ejemplo, de 1 a 5, siendo 1 la respuesta más negativa y el 5 el valor más positivo).
- Checklist Binaria: Es la constituida por preguntas con respuesta única y excluyente: Sí o No. Aritméricamente, equivalen a 1(unos) o 0(cero), respectivamente.

**1.4.1.4 Observación.** La observación consiste en presenciar personalmente la forma cómo determinados procedimientos operativos y de control son llevados a cabo en ente auditado. Esta técnica se aplica por ejemplo para observar el levantamiento físico de un inventario, las condiciones en que se encuentra la maquinaria y equipos empleados, el grado de utilización de la planta instalada, el nivel de segregación de funciones y la distribución lógica del trabajo, entre otros.

**1.4.1.5 Trazas y/o huellas.** Las traza y/o huellas se utilizan para comprobar exactamente la ejecución de las funciones previstas y no otras. Con frecuencia esta técnica es utilizada en ambientes informatizados donde se requiere rastrear los caminos que siguen los datos a través de un programa; esta información del rastreo es almacenada en logs, que vendría a ser un historial que informa que fue cambiando y cómo fue cambiando.

**1.4.1.6 Confirmación.** Esta técnica permite comprobar por escrito, la información proveniente de terceros evitando ambigüedades y falsedades en los hechos.

**1.4.1.7 Análisis.** El análisis es una técnica que permite examinar y determinar la razonabilidad entre los procedimientos y los aspectos jurídicos – normativos establecidos.

**1.4.1.8 Técnicas asistidas por computador.** En el presente, el incremento del uso de computadoras y sus aplicaciones ha hecho que el soporte documental sea reemplazado gradualmente por los documentos electrónicos, de tal manera que el rastro de auditoría tradicional ha desaparecido.

Afortunadamente la propia TI que inciden en los procedimientos que el auditor ha de aplicar proporciona paralelamente medios de ejecutarlos de forma eficiente, veloz y segura.

Las técnicas asistidas por computador (CAATS) ponen a disposición del auditor diversas herramientas (Tratamiento de textos, Hojas de calculo, Generadores de papeles de trabajo, flowcharting y otras especializadas) que no solo viabilizan los nuevos procedimientos sino que mejoran sustancialmente su aplicación.

**1.4.1.9 Otras técnicas.** El auditor profesional y experto es aquel que muchas veces combina y utiliza nuevas técnicas en función del escenario auditado permitiendo un mejor trabajo. Algunas técnicas generalmente aceptadas son:

- Análisis estructurado
- Simulación
- Documentación y modelación de datos y procesos
- Cálculos
- Conciliación
- Otras

## **1.5 METODOLOGIAS DE AUDITORÍA**

Una metodología es el fruto del nivel profesional de cada uno y su visión de cómo conseguir un mejor resultado, por ello es una herramienta de trabajo indispensable a la hora de iniciar cualquier labor.

Una metodología consta de etapas (fases o módulos) formalmente estructurados y utiliza el enfoque metodológico deductivo, esto es que aborda el examen y evaluación de determinada realidad, mediante el conocimiento global del objeto auditable, para luego iniciar un proceso de desagregación de sus elementos hasta aproximarse o llegar al origen de los hechos.

**1.5.1 Tipos de metodologías.** Todas las metodologías existentes desarrolladas y utilizadas en la auditoría y el control se pueden agrupar en dos grandes familias, las metodologías cuantitativas y las metodologías cualitativas.

**1.5.1.1 Metodologías cuantitativas.** Están diseñadas para producir una lista de riesgos que pueden compararse entre sí con facilidad por tener asignados unos valores numéricos.

**1.5.1.2 Metodologías Cualitativas.** Basadas en el criterio y raciocinio humano capaz de definir un proceso de trabajo, para seleccionar basándose en la experiencia acumulada. Están basadas en métodos estadísticos y lógica borrosa.

La tendencia de uso en la realidad es la mezcla de ambas. En la Tabla 1 se observa un cuadro comparativo de estas metodologías.

*Tabla 1. Comparación entre metodologías cuantitativas y cualitativas.*

METODOLOGIA	VENTAJAS	DESVENTAJAS
Cuantitativa	Enfoca pensamientos mediante el uso de números. Facilita la comparación de vulnerabilidades muy distintas. Proporciona una cifra "justificante" para contramedida.	Estimación de probabilidad depende de estadísticas inexistentes. Estimación de perdidas potenciales solo si son valores cuantificables. Metodologías estándar. Difíciles de mantener o modificar. Dependencia de un profesional.
Cualitativa	Enfoque lo amplio que se desee. Plan de trabajo flexible y reactivo Se concentra en la concentración de eventos Incluye factores intangibles Identificación de eventos reales más claros al no tener que aplicarles probabilidades más complejas de calcular	Depende fuertemente de la habilidad y calidad del personal involucrado Puede excluir riesgos significantes desconocidos (depende de la capacidad profesional para usar las técnicas) Dependencia de un profesional

Fuente: Piattini y del Peso(1998:52).

**1.5.2 Estándares de auditoría.** El trabajo de los auditores, sean internos o externos, se rigen en general por estándares desarrollados por una cantidad de

organizaciones profesionales, cada una de las cuales busca asegurarse de la calidad de auditoría que se realiza.

Algunas de las metodologías más importantes para los profesionales de la contabilidad y la auditoría son:

- ISACA (COBIT)
- COSO
- AICPA (SAS)
- IFAC (NIA)
- SAC
- ISO

**1.5.2.1 COBIT.** COBIT ha sido desarrollado como un estándar generalmente aplicable y aceptado para las buenas prácticas de seguridad y control en Tecnología de Información (TI). COBIT se fundamenta en los Objetivos de Control existentes de la Information Systems Audit and Control Foundation (ISACF), mejorados a partir de estándares internacionales técnicos, profesionales, regulatorios y específicos para la industria, tanto existentes como en surgimiento. Los Objetivos de Control resultantes han sido desarrollados para su aplicación en sistemas de información en toda la empresa.

**1.5.2.2 COSO.** The Committee of Sponsoring Organizations of the Treadway Commission's Internal Control - Integrated Framework (COSO).

Publicado en 1992 hace recomendaciones a los contables de gestión de cómo evaluar, informar e implementar sistemas de control, teniendo como objetivo de control la efectividad y eficiencia de las operaciones, la información financiera y el cumplimiento de las regulaciones que explica en los componentes del ambiente de control, valoración de riesgos, actividades de control, información y comunicación, y el monitoreo.

**1.5.2.3 AICPA (SAS).** The American Institute of Certified Public Accountants' Consideration of the Internal Control Structure in a Financial Statement Audit (SAS 55), que ha sido modificado por el (SAS 78), 1995.

Da una guía a los auditores externos sobre el impacto del control interno en la planificación y desarrollo de una auditoría de estados financieros de las empresas, presentado como objetivos de control la información financiera, la efectividad y eficiencia de las operaciones y el cumplimiento de regulaciones, que desarrolla en los componentes de ambiente de control, valoración de riesgo, actividades de control, información, comunicación y monitoreo.

**1.5.2.4 IFAC (NIA).** La Federación Internacional de Contables IFAC emitió las Normas Internacionales de Auditoría NIA 15, 16 y 20 en 1991.

IFAC muestra en la NIA 15 (Auditoría en Entornos Informatizados) una referencia de controles para procesamiento electrónico de datos y la necesidad de estos cuando estamos en ambientes donde los instrumentos tradicionales del papel y demás pistas de auditoría no son visibles para los contables en el momento de realizar su trabajo.

La NIA 16 (Técnicas de Auditoría Asistida por Computador) describe técnicas y procedimientos de auditoría que se pueden hacer en entornos informatizados con ayuda de los computadores y otras tecnologías.

La NIA 20 nos presenta los efectos de un entorno informatizado en la evaluación de sistemas de información contables. Junto con las demás normas dan una guía al auditor de los controles en general a tener en cuenta en un ambiente informatizado y en las aplicaciones que procesan la información, así como técnicas de auditoría asistidas por computador y su importancia.

**1.5.2.5 SAC.** The Institute of Internal Auditors Research Foundation's Systems Auditability and Control (SAC). Realizado en 1991 y revisado posteriormente. Ofrece una guía de estándares y controles para los auditores internos en el área de

auditoría de sistemas de información y tecnología. Tiene como objetivos de control la efectividad y eficiencia de las operaciones, la integridad de la información financiera y el cumplimiento de normas y regulaciones que explica en el ambiente de control, sistemas manuales y automatizados y procedimientos de control.

**1.5.2.6 ISO.** ISO es la Organización Internacional para la Estandarización. Está conformada por más de 140 institutos nacionales para la estandarización de grandes y pequeños países, industrializados y en desarrollo, en toda la región del mundo. ISO emitió las normas de auditoría ISO 9000, ISO 14000.

## **1.6 EL INFORME DE AUDITORÍA**

Antes de profundizar sobre el informe se presentará una serie de aspectos previos para tenerlos muy presentes al redactar el informe de auditoría, esto es, la comunicación del auditor al cliente, formal y quizá, solemne, tanto del alcance de la auditoría (objetivos, periodo de cobertura, naturaleza y extensión del trabajo realizado) como los resultados y conclusiones.

**1.6.1 La evidencia.** La obtención de evidencias se hace mediante la aplicación de las técnicas de auditoría y del modelo de pruebas diseñado especialmente para el efecto.

Las evidencias de auditoría deberán precisar una plena y ordenada información que permita sustentar rigurosamente, las recomendaciones dadas al objeto auditado.

La evidencia tiene una serie de calificativos:

- Evidencia relevante, que tiene una relación lógica con los objetivos de la auditoría.
- Evidencia fiable, que es válida y objetiva, aunque con nivel de confianza.
- Evidencia suficiente, que es de tipo cuantitativo para soportar la opinión profesional del auditor.

- Evidencia adecuada, que es de tipo cualitativo para afectar a las conclusiones del auditor.

**1.6.2 Los papeles de trabajo.** El trabajo de auditoría, en cualquiera de sus modalidades requiere de la preparación de papeles de trabajo, para registrar en ellos las evidencias halladas en el desarrollo del proceso de auditoría.

El diseño de papeles de trabajo, depende del alcance, los objetivos y los tipos de pruebas, pero también juega la creatividad del auditor para diseñar y organizar los papeles de trabajo, de manera que presenten en forma detallada y clara la información (evidencia comprobatoria suficiente y competente) y los resultados de los análisis y conclusiones del trabajo de auditoría. En general para diseñar papeles de trabajo, debe procurarse que contenga por los menos lo siguiente:

- Nombre o identificación del papel de trabajo, periodo auditado, fecha de terminación.
- Descripción de la tarea realizada, identificación completa de la fuente de información.
- Métodos de verificación empleados o técnicas de auditoría.
- Nombres de las personas que aprueban y supervisan el trabajo.
- Conclusiones sobre el examen practicado.

### **1.6.3 Pruebas de auditoría.**

**1.6.3.1 Pruebas de Cumplimiento.** Las pruebas de cumplimiento son aquellas dirigidas a verificar la aplicación efectiva del control interno, de las disposiciones legales, reglamentarias, políticas, etc. y la operatividad y confiabilidad de los procedimientos establecidos. La pruebas de cumplimiento deben diseñarse, en función a las circunstancias particulares de cada objeto auditado.

**1.6.3.2 Pruebas Sustantivas.** La característica esencial de la prueba sustantiva es que esta diseñada para llegar a una conclusión con respecto a la validez de la información y los datos, en aquellos casos en los cuales no existen controles adecuados. En la tabla 2 se resume estos dos tipos de prueba.

*Tabla 2. Comparación entre pruebas de cumplimiento y sustantivas*

	De cumplimiento	Sustantivas
Sinónimo	De control interno	De validez
Objetivo	Funcionamiento de los procedimientos internos del ente.	Validar la información y los datos.
Objeto Auditado	Gestión del ente.	Sistemas de información

**1.6.3.3 Riesgos y controles.** Entre las tareas del auditor esta la elaboración de matrices de riesgo y detección de los controles, cuyo objetivo principal es detectar las áreas o componentes de mayor peligro y que requieren una revisión formal y oportuna.

El riesgo, se puede definir como la probabilidad de que una amenaza llegue a suceder por una vulnerabilidad y el control, es la acción que se ejerce sobre una causa de riesgo con el fin de reducir su probabilidad de ocurrencia o el impacto que pueda generar su ocurrencia (Vea figura 1).

Los riesgos en auditoría pueden clasificarse de la siguiente manera:

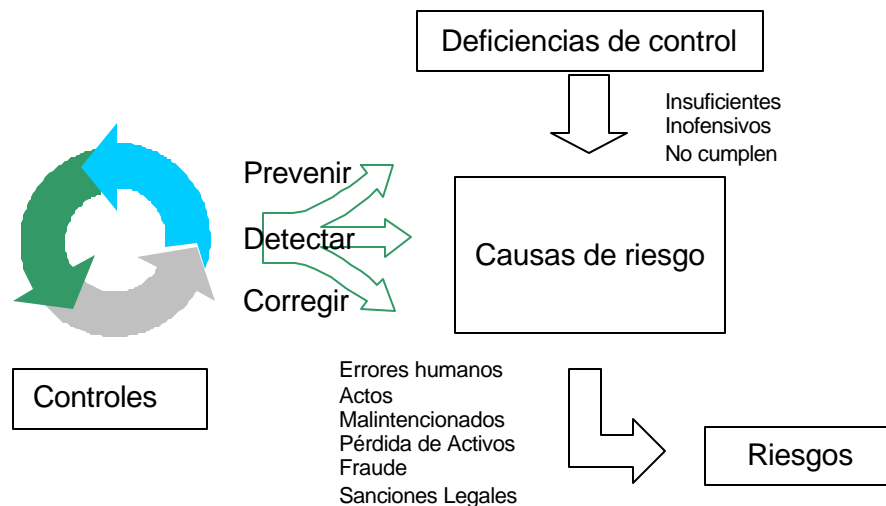
- Riesgo inherente: Es el riesgo innato o de origen que posee todo objeto de auditoría por el simple hecho de su existencia.
- Riesgo de control: Este riesgo estará dado por la ineficiencia de los procedimientos de control del ente auditado.

Los controles actúan sobre las causas del riesgo de tres maneras, mutuamente excluyentes:

1. Como control Preventivo: Para evitar la ocurrencia de la causa del riesgo. Ejemplo "letrado Use casco en su labor".

2. Como control Detectivo: Para detectar, registrar e informar la ocurrencia de la causa (actuar como alarma que se dispara cuando detecta la causa)
3. Como control Correctivo: Obligan a tomar acción correctiva para resolver el problema detectado por los controles detectivos.

Figura 1. Relación entre Riesgos y controles.



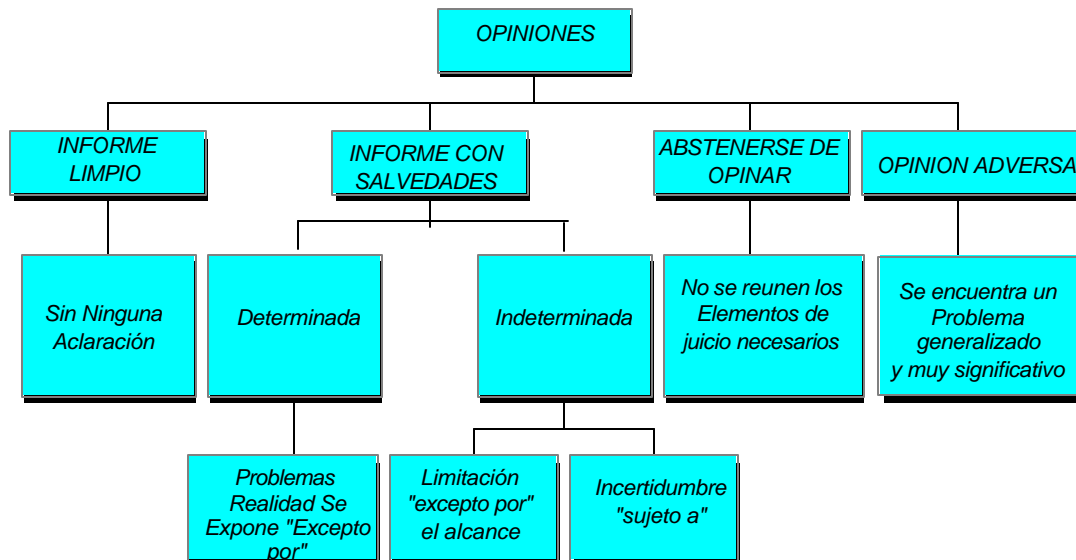
**1.6.3.4 El informe.** Dependiendo del tipo de auditoría, el informe se referirá a una u otra materia, indicando si hay o no salvedades y haciendo las sugerencias que estime convenientes para mejorar. En su opinión, el auditor dirá si el área auditada (objeto de auditoría) se encuentra conforme a los contenidos teóricos - técnicos vigentes y sobre el grado de confiabilidad del sistema de control interno.

Este informe debe contener uno de los siguientes tipos de opinión: favorable o sin salvedades, con salvedades, desfavorable o adversa y denegada

- Opinión favorable. La opinión calificada como favorable, sin salvedades o limpia, deberá manifestarse de forma clara y precisa, y es el resultado de un trabajo realizado sin limitaciones de alcance y sin incertidumbre, de acuerdo con la normativa legal y profesional.

- Opinión con salvedades. Se reitera lo dicho en la opinión favorable al respecto de las salvedades cuando sean significativas en relación con los objetivos de auditoría, describiéndose con precisión la naturaleza y razones.
- Opinión desfavorable. La opinión desfavorable o adversa es aplicable en el caso de: identificación de irregularidades, incumplimiento de la normativa legal y profesional, que afecten significativamente a los objetivos de la auditoría estipulados.
- Opinión denegada. La denegación de la opinión puede tener su origen en: las limitaciones al alcance de auditoría, incertidumbres significativas de un modo que impidan al auditor formarse una opinión, irregularidades, el incumplimiento de la normativa legal y profesional.

Figura 2. Tipos de opinión en un informe de auditoría.



## **1.7 RESUMEN**

En el presente capítulo se presentó, de forma general los aspectos teóricos más importantes de la auditoría. Se expuso la auditoría como una actividad que implica el examen y la revisión de un objeto sometido a estudio. Así, una auditoría no es una actividad fácil de realizar y requiere del ejercicio de un juicio profesional, sólido, y maduro, en algunos casos requiere del apoyo de otros profesionales, en calidad de asesores o expertos para juzgar los procedimientos que deben seguirse y estimar los resultados obtenidos.

## Capítulo 2

# AUDITORÍA DE SISTEMAS DE INFORMACION Y SU ENTORNO

Tras haber introducido los aspectos previos de la auditoría en general, se pasará a enfocar un estudio mayor sobre la auditoría de sistemas de información, con lo cual se busca clarificar la introducción de este concepto en las organizaciones.

A lo largo del presente capítulo se expondrán los conceptos relacionados con la auditoría de sistemas de información y el esquema metodológico utilizado para su desarrollo.

### 2.1 SISTEMAS DE INFORMACION Y TECNOLOGIAS DE INFORMACION EN EL ENTORNO ORGANIZACIONAL

La idea de sistema de información(SI)<sup>3</sup> como un conjunto de componentes y procesos que, operando sobre una colección de datos estructurada de acuerdo con las necesidades de una empresa, recopila, elabora y distribuye la información necesaria para desempeñar las funciones y procesos de negocio, permite obtener una visión de lo que entendemos por sistema de información en una organización.

Esta visión se obtiene observando en las empresas la información<sup>4</sup> que fluye tanto internamente como externamente y donde hoy, la introducción de las tecnologías de información(TI) como estrategia competitiva, ha impulsado un verdadero cambio en la cultura organizacional, haciéndose imprescindible para la implementación de los sistemas de información.

---

<sup>3</sup> Andreu, Ricart y Valor (1996: 13)

<sup>4</sup> Ideas expuestas en Cornella(1994)

El hecho de que las tecnologías de información jueguen un papel importante para la implementación de los sistemas de información implica darse cuenta de la importancia de la información,<sup>5</sup> del cambio estructural y de aprendizaje que se requiere para aprovechar su potencial. En otras palabras las empresas deben usar TI para digerir mejor la información que obtienen, generan y manejan, con el fin de mejorar su habilidad y capacidad de competitividad frente a los demás.

## **2.2 RELACION DE LA AUDITORÍA DE SISTEMAS DE INFORMACION CON EL ENTORNO ORGANIZACIONAL**

Si bien el desarrollo tecnológico y la introducción de las TI ofrecen ventajas en cuanto a optimizar el trabajo, y como consecuencia el funcionamiento de la organización, también genera una enorme dependencia creciente de sus elementos, con lo cual se requiere un entendimiento<sup>6</sup> sobre su empleo para proporcionar una dirección efectiva y controles adecuados.

De igual forma, las organizaciones deben cumplir con los requerimientos de calidad y seguridad tanto para su información como para sus activos garantizando de esta manera un empleo apropiado de los recursos disponibles.

En este contexto, se encuentra inmerso el concepto de auditoría de sistemas de información, cuya orientación principal es brindar a la dirección, información objetiva e independiente sobre: el grado de cumplimiento de los controles (políticas y procedimientos), detección de riesgos donde existan debilidades significativas de control y recomendaciones para realizar acciones correctivas, etc.

---

<sup>5</sup> Investigación realizada por Olave (2000) en su tesis de grado “Propuesta de un modelo de evaluación de la administración de la información en las organizaciones empresariales”.

<sup>6</sup> Cobit (1998)

De hecho, el escenario en el que se encuentra la auditoría de sistemas de información ha permitido que esta se especialice<sup>7</sup> y requiera de personal calificado, metodologías, procesos y herramientas adecuadas para el desempeño de una mejor labor.

### **2.3 DEFINICION DE LA AUDITORÍA DE SISTEMAS DE INFORMACION**

Después de analizar el concepto de auditoría en general, los diferentes tipos de auditoría, así como su relación con el entorno organizacional, conviene clarificar el término auditoría de sistemas de información (S.I) o auditoría informática como es conocida en el ámbito internacional.

Esta es la definición de Isaca<sup>8</sup> (Information Systems Audit and Control Association):

La Auditoría de sistemas de información, se define como cualquier auditoría que abarca la revisión o evaluación de todos los aspectos (o cualquier porción de ellos) de los sistemas automáticos de procesamiento de la información, incluyendo los procedimientos no automáticos relacionados con ellos y las interfaces correspondientes.

Echenique(2001) la define:

Auditoría informática es la revisión y evaluación de los controles, sistemas y procedimientos de la informática; de los equipos de computo, su utilización, eficiencia y seguridad; de la organización que participa en el procesamiento de la información, a fin de que por medio del señalamiento de cursos alternativos se logre una utilización más eficiente, confiable y segura de la información que servirá para una adecuada toma de decisiones.

Pinilla(1997) y otros autores coinciden en hacer énfasis en la revisión y evaluación de un sistema informatizado para determinar si salvaguarda los activos, lleva eficazmente los fines de la organización y utiliza eficientemente los recursos.

---

<sup>7</sup> Hernandez (2000,3-6)

<sup>8</sup> [www.isaca.org](http://www.isaca.org)

Así, la introducción de este concepto debe entenderse como un proceso de exploración, comprensión y evaluación de la realidad que pretende reflejar los sistemas de información arrojando opiniones y conclusiones que conlleven a determinar cursos alternativos de acción. Por ejemplo: la administración eficiente de T.I, cambios en la estructura y cultura organizacional, Cambios en políticas y reglas, creación y mejoramiento de procesos etc.

## **2.4 ESTRUCTURA ORGANIZACIONAL Y FUNCIONES DE LA AUDITORÍA DE SISTEMAS DE INFORMACION**

**2.4.1 Ubicación jerárquica de la función.** La alta dirección de cualquier organización tiene que estar consciente de que la función de auditoría se debe ejercer con el criterio básico de independencia personal jerárquica, es decir, el desempeño de las actividades en el proceso de evaluación y control no debe verse afectado por aspectos emocionales ni de autoridad emanados de los responsables e involucrados en el momento de la auditoría.

Según lo expresado se recomienda ubicar la función de auditoría de sistemas en un nivel organizacional (La ubicación deseable es subordinada jerárquicamente a una dirección o subdirección, ya sea administrativa o de informática) que le asegure la independencia y soporte requerido de la alta dirección, con la finalidad de contar con una entidad confiable y eficiente.

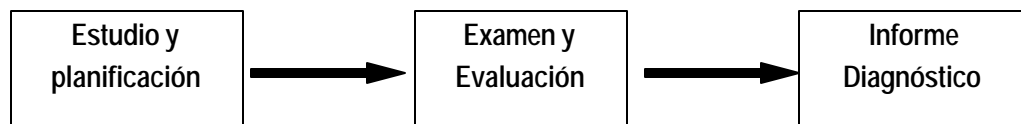
## **2.5 METODOLOGIA PARA EL DESARROLLO E IMPLANTACION DE LA AUDITORÍA DE SISTEMAS DE INFORMACION**

La auditoría de sistemas de información debe respaldarse por un proceso formal que asegure un previo entendimiento para cada uno de los responsables de llevar a la práctica dicho proceso en la organización. Al igual que otras funciones en el

negocio la auditoría de sistemas de información efectúa tareas y actividades mediante una metodología. Según Hernández (2000:63) “No es recomendable fomentar la dependencia en el desempeño de esta importante función solo con base en la experiencia, habilidades, criterios y conocimientos sin una referencia metodológica”.

Los programas de trabajo de auditoría de sistemas de información al igual que cualquier auditoría deberá incluir tres partes: Estudio y planificación, Examen y evaluación e informe – diagnóstico.

*Figura 3. Partes o Flujos de Trabajo en la auditoría*

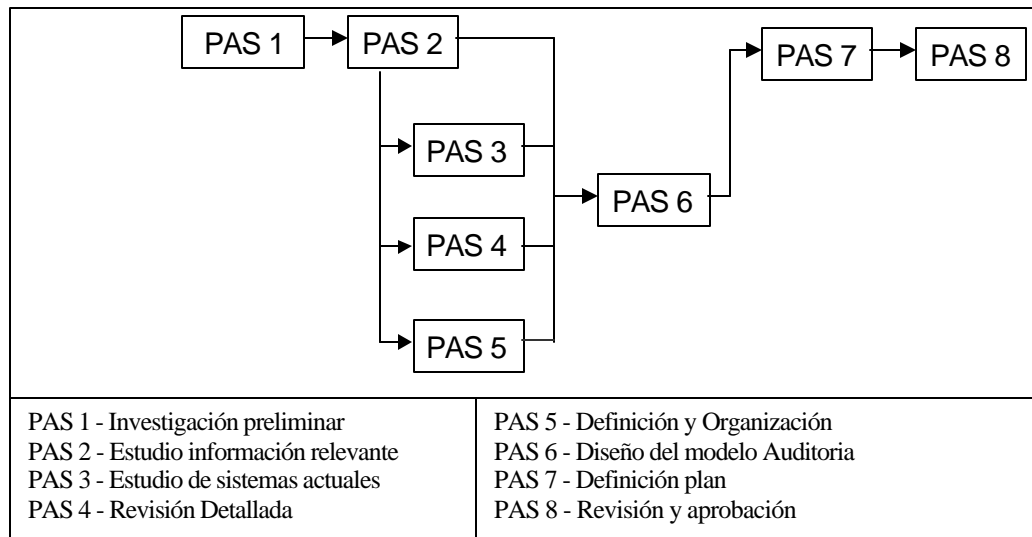


La ejecución del proceso de auditoría de sistemas de información implica el desarrollo de varias etapas enmarcadas en los tres flujos de trabajo o partes citadas.

**2.5.1 Flujo de trabajo: planificación y estudio.** El objetivo de la planeación es determinar las tareas de auditoría, de tal manera que aborden los objetivos pertinentes y se cumplan con las normas y estándares aplicables de auditoría profesional. Para realizar una adecuada planeación de la auditoría de sistemas de información hay que seguir una serie de pasos o etapas (véase figura 4) que permitan dimensionar el tamaño y características del área dentro del organismo a auditar, sus sistemas, organización y equipo. Con ello se puede determinar el número y características del personal de auditoría, herramientas necesarias, el

tiempo, y costo, así como definir alcances de la auditoría para, poder elaborar el contrato<sup>9</sup> de servicios.

Figura 4. Etapas o pasos en el estudio y planeación de la auditoría de S.I



Es importante aclarar que este proceso de planeación depende en gran medida del diagnostico previo que lleve a cabo el auditor sobre la situación del auditado. Además, el proceso de elaboración o actualización del plan de auditoría depende de las prioridades del negocio o necesidades externas que tenga dicho proceso en cada organización.

**2.5.1.1 Investigación Preliminar.** La investigación preliminar determina el arranque formal del proceso o programa de auditoría y su objetivo principal es el obtener la información necesaria para que el auditor pueda tomar la decisión de cómo proceder en la auditoría.

<sup>9</sup> Echenique(2001, 47-54) muestra un ejemplo de propuesta de servicios de auditoría informática

En esta etapa se debe recopilar la máxima información útil en el tiempo más corto posible, se obtienen objetivos, responsables y alcance.

Algunas de las tareas básicas de esta etapa se pueden ver en la tabla 3.

*Tabla 3. Tareas básicas en la investigación preliminar.*

Actividad	Entrada	Técnicas	Participantes	Salida
Análisis de la necesidad del programa de auditoría	Solicitud formal del programa de auditoría	Sesiones de trabajo	Dirección de la organización	Descripción general del plan de auditoría
Identificación del alcance	Descripción general del programa de auditoría	Sesiones de trabajo y encuestas	Dirección de la organización	Descripción general del plan de auditoría
Recolección de información	Aspectos como: Antecedentes, Marco legal, organización, operaciones	Sesiones de trabajo y encuestas	Dirección de la organización y Auditor	Descripción general del plan de auditoría

**2.5.1.2 Estudio de información Relevante.** El objetivo principal de esta fase es analizar la información de interés para el correcto desarrollo del plan de auditoría. Véase algunas de las tareas en la tabla 4.

*Tabla 4. Tareas básicas en el estudio de información relevante.*

Actividad	Entrada	Técnicas	Participantes	Salida
Selección y valoración de antecedentes	Información relevante Descripción general de procesos Catálogo de objetivos Catálogo de usuarios	Sesiones de trabajo	Equipo de Auditoría Usuarios expertos	Valoración de antecedentes
Análisis de antecedentes	Información relevante Valoración de antecedentes	Catalogación	Equipo de Auditoría	Catalogo de requisitos

**2.5.1.3 Estudio de los sistemas actuales.** Esta etapa obtiene una valoración de la situación actual de los sistemas de información actuales. En la tabla 5 se resumen algunas de las actividades de esta etapa.

*Tabla 5. Tareas básicas en el estudio de los sistemas actuales.*

Actividad	Entrada	Técnicas	Participantes	Salida
Alcance y objetivos del estudio de los S.I. Actuales	Catálogo de objetivos Descripción general de procesos Catálogo usuarios Equipos de trabajo	Catalogación Sesiones de trabajo	Audidores S.I. Usuarios expertos	Catálogo de objetivos Resultado de sesiones de trabajo
Análisis de los S.I. Actuales	Descripción general de procesos Catálogo de objetivos Resultado de sesiones de trabajo	Sesiones de trabajo Diagramas de representación	Audidores S.I. Usuarios expertos Equipo desarrollo Equipo soporte técnico Responsable de mantenimiento	Descripción general de S.I. actuales
Valoración S.I. Actuales	Catálogo de objetivos Descripción general S.I. Actuales	Sesiones de trabajo Análisis de riesgos	Audidores S.I.	Valoración de la situación actual

**2.5.1.4 Revisión Detallada.** El objetivo principal de esta etapa es la obtención de la información necesaria para tener un profundo entendimiento de los controles usados. Consúltese la tabla 6 para ver las tareas relacionadas.

*Tabla 6. Tareas básicas en la revisión detallada.*

Actividad	Entrada	Técnicas	Participantes	Salida
Estudio de procesos de la organización	Descripción de procesos Catálogo de objetivos Equipos de trabajo	Modelado de procesos Uso de Estándares Sesiones de trabajo	Audidores S.I. Usuarios expertos	Catálogo de objetivos Resultado de sesiones de trabajo
Análisis de las necesidades de información	Modelo de procesos	Pruebas Encuestas Uso de Estándares	Audidores S.I. Usuarios expertos Equipo desarrollo Equipo soporte técnico	Descripción general de S.I. actuales

			Responsable de mantenimiento	
--	--	--	------------------------------	--

**2.5.1.5 Definición y Organización.** Esta etapa permite clarificar y organizar la información obtenida en las etapas anteriores. En la Tabla 7 se listan algunas de las tareas de esta etapa.

*Tabla 7. Tareas básicas en la Definición y organización.*

Actividad	Entrada	Técnicas	Participantes	Salida
Especificación del ámbito y alcance	Descripción general del programa de auditoría	Catalogación	Comité de dirección Jefe de proyecto Auditores S.I.	Descripción general de procesos Catálogo de objetivos
Organización del programa de auditoría	Estructura organizativa Descripción general PA Descripción procesos Catálogo de objetivos	Catalogación Sesiones trabajo	Auditores S.I. Jefe de proyecto	Catálogo usuarios Equipos de trabajo
Definición del plan de trabajo	Equipos de trabajo Descripción general PA Catálogo de objetivos Descripción procesos	Planificación Estimación	Auditores S.I. Jefe de proyecto	Plan de trabajo
Comunicación del plan de trabajo	Plan de trabajo Catálogo de usuarios		Comité de dirección Auditores S.I. Jefe de proyecto	Plan de trabajo aceptado

**2.5.1.6 Diseño del Modelo de Auditoría.** El Diseño de Modelo de Auditoría Identifica y define los procedimientos que van a dar soporte a los procesos afectados por el programa de Auditoría. Véase en la Tabla 8 algunas de las tareas realizadas.

*Tabla 8. Tareas básicas en el diseño del modelo de auditoría.*

Actividad	Entrada	Técnicas	Participantes	Salida
Diseño y planificación de pruebas	Modelo de información Valoración de la situación actual	Pruebas	Auditores S.I. Usuarios expertos	Diagnóstico de la situación actual
Dimensionar el tamaño y características del área a auditar	Diagnóstico situación actual Modelo de información Modelo de procesos	Pruebas	Auditores S.I. Usuarios expertos	Diseño de controles

**2.5.1.7 Definición del plan.** En esta etapa se elabora y detalla el plan de auditoría: definición de actividades, infraestructura tecnológica, calendario y recursos para implementar el proceso de auditoría. La tabla 9 resume alguna de las tareas realizadas.

Tabla 9. Tareas básicas del proceso de planeación

Actividad	Entrada	Técnicas	Participantes	Salida
Definición de actividades a realizar	Catálogo de objetivos Descripción general del PA Modelo de información Modelo de sistemas de información Infraestructura tecnológica	Impacto en la organización Análisis coste/beneficio Sesiones de trabajo	Equipo de consultoría Usuarios expertos	Plan de seguimiento
Elaboración del plan de seguimiento	Plan de seguimiento		Equipo de Auditoría	Plan de seguimiento

**2.5.1.8 Revisión y Aprobación.** En este punto, se somete el plan a la última revisión y aprobación de la dirección. Es necesario destacar que el plan de trabajo debe ser lo suficientemente flexible en algunas circunstancias que así lo aconsejen, para impedir situaciones que no permitan el desarrollo del programa de auditoría. Algunas de las tareas de esta etapa se muestran en la tabla 10.

Tabla 10. Tareas básicas del proceso de planeación

Actividad	Entrada	Técnicas	Participantes	Salida
Convocatoria de la presentación	Catálogo de objetivos Modelo de información Modelo de auditoría Arquitectura tecnológica Plan de seguimiento	Presentación	Equipo de Auditoría	Plan de presentación
Evaluación y mejora de la propuesta	Catálogo de objetivos Modelo de información Modelo de sistemas de información Arquitectura tecnológica	Sesiones de trabajo	Equipo de Auditoría Comité de dirección Usuarios expertos	Resultado de las sesiones Presentación Modelo de información

	Plan de seguimiento			Modelo de sistemas de información infraestructura tecnológica Plan de seguimiento
Aprobación del Plan de Trabajo	Documentación generada	Sesiones de trabajo	Comité de dirección Jefe de proyecto	Aprobación formal Plan de comunicación

**2.5.2 Flujo de trabajo: examen y evaluación.** En este flujo de trabajo, el auditor comienza a ejecutar sus tareas de acuerdo con el plan aprobado, donde se deberá obtener, analizar, interpretar y documentar la información para apoyar los resultados de la auditoría.

La ejecución del trabajo además, deberá ser supervisado - revisado para proporcionar certeza de que los objetivos de auditoría están siendo alcanzados y que se satisfagan las normas y estándares aplicables de auditoría.

Básicamente, deben cumplirse las siguientes etapas: Revisión formal por áreas o procesos, Realización de pruebas y obtención de evidencias, Documentación de hallazgos y preparación de informes.

**2.5.2.1 Revisión formal por áreas o procesos.** Para efectos de facilitar el desarrollo del proceso de auditoría, se debe precisar con claridad las áreas o procesos del negocio que enfoquen la misión del auditor a mejorar o examinar alguna función específica. En la revisión formal por áreas o procesos se deben alcanzar los objetivos formulados en el plan de auditoría así como recopilar la evidencia suficiente y competente, que constituya un respaldo a los resultados de la auditoría.

**2.5.2.2 Realización de pruebas y obtención de evidencias.** El objetivo de la realización de pruebas es recopilar evidencias de auditoría adecuados sobre el funcionamiento eficaz o ineficaz de las actividades y los controles identificados. En esta etapa la recopilación del material de evidencia es un paso clave en el proceso de auditoría, por ello el auditor debe hacer uso de los papeles de trabajo, las

técnicas y conocimientos de auditoría para reunir información suficiente, confiable, pertinente y útil a fin de que se alcancen los objetivos de auditoría.

**2.5.2.3 Documentación de hallazgos y preparación de informes.** En esta etapa, los auditores, con base en las evidencias documentan los hallazgos (surgen cuando al comparar el criterio “debe ser” con la condición “lo que actualmente existe o se encuentra funcionando” se observan diferencias) y preparan informes preliminares del trabajo realizado con el fin de tener los suficientes elementos de juicio.

**2.5.3 Flujo de trabajo: informe – diagnóstico.** Los informes de auditoría son el producto final del trabajo del auditor, y son utilizados para indicar las observaciones, hallazgos, conclusiones, la opinión sobre lo adecuado o inadecuado de los controles o procedimientos revisados durante el proceso de auditoría y las recomendaciones a los interesados.

Generalmente, la estructura o contenido de un informe es el siguiente:

- Introducción al informe, donde se expresara los objetivos de la auditoría, el período o alcance cubierto por la misma, y una expresión general sobre la naturaleza o extensión de los procedimientos de auditoría realizados.
- Observaciones detalladas y recomendaciones de auditoría.
- Respuestas de la gerencia a las observaciones con respecto a las acciones correctivas.
- Conclusión global del auditor expresando una opinión sobre los controles y procedimientos revisados.

**2.5.3.1 Seguimiento de las observaciones de auditoría.** El trabajo de auditoría es un proceso continuo. El auditor de sistemas de información deberá solicitar y evaluar la información apropiada con respecto a hallazgos, conclusiones y recomendaciones relevantes anteriores para determinar si se han implementado las acciones apropiadas de manera oportuna.

## **2.6 PERSPECTIVAS DE LA AUDITORÍA DE SISTEMAS DE INFORMACION**

**2.6.1 En el campo educativo.** Naturalmente, incursionar en la investigación del campo de la auditoría de sistemas de información no es sencillo, requiere tiempo, recursos, y formación para el trabajo interdisciplinario.

En la universidad Industrial de Santander y en especial en la escuela de sistemas, el marco conceptual, metodológico e instrumental en que se mueve el conocimiento y las aplicaciones de auditoría actuales se encuentra en una etapa de exploración, por ello surge esta investigación, que además, requiere redefinir el modelo académico vigente para poder responder con las nuevas exigencias del entorno.

**2.6.2 En el campo practico aplicado.** El desarrollo de las tecnologías de información y en general del entorno organizacional, esta requiriendo cada vez mas, avanzados sistemas de control interno y de auditoría que permitan anticiparse a la incertidumbre provocada por situaciones irregulares y el mal uso de las T.I.

## **2.7 RESUMEN**

En este capitulo se ha expuesto la estructura, conceptos y la metodología de auditoría de sistemas de información a través de flujos de trabajo, sacrificando una explicación práctica que requiere el uso de las técnicas de auditoría enunciadas con anterioridad.

Se encontró que la auditoría de sistemas de información es un área de estudio nueva que requiere una continua investigación en nuestro entorno académico.

## Capítulo 3

### EL ESTUDIO DE CASOS COMO TECNICA DIDACTICA

En el proceso de aprendizaje, la representación de una situación de la realidad como base para la reflexión y conocimiento ha sido utilizada desde tiempos remotos, el planteamiento de un caso es siempre una oportunidad de aprendizaje significativo y trascendente en la medida en que quienes participan en su análisis toman conciencia exacta y justa de la situación; después, una conceptualización y finalmente, una búsqueda de las soluciones eficaces.

En este capítulo se explican las características principales de la técnica de estudio de casos. La participación en este tipo de técnica desarrolla habilidades tales como el análisis, síntesis, y evaluación de la información. Posibilita también el desarrollo del pensamiento crítico, el trabajo en equipo y la toma de decisiones, además de otras actitudes y valores como la innovación y la creatividad.

#### 3.1 EL ESTUDIO DE CASOS COMO ESTRATEGIA DIDACTICA

Según Mucchielli (1970), la necesidad de responder a la exigencia de formación es la que origina el “método de los casos”.

La técnica del caso tiene ya una larga historia en la enseñanza. Si se considera a la palabra “caso” en su sentido amplio, se puede afirmar que en la educación siempre se ha utilizado en forma de ejemplo o problema práctico. La casuística, por ejemplo, típica de la filosofía escolástica medieval, no es sino la aplicación del caso para resolver problemas morales o religiosos, pero sin entrar en el análisis de la situación social o psicológica previa.

En el sentido más estricto, el caso se comienza a utilizar en la Universidad de Harvard, en los programas de Derecho y de legislación reservados para estudiantes avanzados y en la preparación de directivos de empresa hacia 1914. Bajo el nombre de "Case System" se pretendía que los alumnos del área de leyes buscaran la solución a una situación real concreta y la defendieran. Pero es hacia 1935 cuando el método cristaliza en su estructura definitiva y se extiende, como metodología docente, a otros campos.

Se perfecciona, además, con la asimilación del "role-playing" y del sociodrama que son otras dos técnicas de enseñanza las cuales, en pocas palabras, consisten en representar o dramatizar una situación problemática concreta de la vida real.

A partir de estas experiencias, ha sido ampliamente desarrollada en la formación de profesionales en el campo del derecho, la administración de empresas y organización, medicina y ciencias políticas, entre otros.

### **3.2 DEFINICION Y CARACTERISTICAS DE LA TECNICA**

La técnica de estudio de casos, consiste precisamente en proporcionar una serie de casos que representen situaciones problemáticas reales diversas para que mediante un análisis detallado de sus elementos y de la interacción que se produce entre ellos y su contexto se pueda llegar, mediante un proceso de síntesis, a la búsqueda del significado y la toma de decisión que se requiere en la situación estudiada.

Específicamente, un caso es una relación escrita que describe una situación ocurrida en la vida de una persona, familia, grupo o empresa. Su aplicación como estrategia o técnica de aprendizaje, entrena a los alumnos en la elaboración de soluciones válidas para los posibles problemas de carácter complejo que se presenten en la realidad futura.

El caso no proporciona soluciones sino datos concretos para reflexionar, analizar y discutir en grupo las posibles salidas que se pueden encontrar a cierto problema. No ofrece las soluciones al estudiante, sino que le entrena para generarlas. Le lleva a pensar y a contrastar sus conclusiones con las conclusiones de otros, a aceptarlas y expresar las propias sugerencias, de esta manera le entrena en el trabajo colaborativo y en la toma de decisiones en equipo. Al llevar al estudiante a la generación de alternativas de solución, le permite desarrollar la habilidad creativa, la capacidad de innovación y representa un recurso para conectar la teoría a la práctica real.

El método de los casos puede ponerse en acción en todos los campos en que se trata de combinar de modo eficiente la adquisición de conocimientos teóricos y el desarrollo de una experiencia profesional útil.

Las narraciones presentadas como estudio de caso, dentro de la perspectiva didáctica, deben cumplir una serie de condiciones entre las que destacan las propuestas por Mucchielli (1970):

- Autenticidad: Ser una situación concreta, basada en la realidad.
- Urgencia de la Situación: Ser una situación problemática que provoca un diagnóstico o una decisión.
- Orientación pedagógica: Ser una situación que puede proporcionar información y formación en un dominio del conocimiento o de la acción.
- Totalidad: Ser una situación "total"; es decir que incluye toda la información necesaria y todos los hechos disponibles.

Dentro del enfoque del estudio de casos como estrategia didáctica Martínez y Musitu (1995:22-31), mencionan que se pueden considerar en principio tres modelos que se diferencian en razón de los propósitos metodológicos que específicamente se pretenden en cada uno.

**3.2.1 Modelo centrado en el análisis de casos.** Este modelo pretende el conocimiento y la comprensión de los procesos de diagnóstico e intervención llevados a cabo, así como de los recursos utilizados, las técnicas empleadas y los resultados obtenidos a través de los programas de intervención propuestos. A través de este modelo, básicamente se pretende que los estudiantes, y/o profesionales en formación, conozcan, analicen y valoren los procesos de intervención elaborados por expertos en la resolución de casos concretos. Complementariamente, se puede estudiar soluciones alternativas a la tomada en la situación objeto de estudio.

**3.2.2 Modelo centrado en enseñar a aplicar principios y normas.** Este segundo modelo pretende enseñar a aplicar principios y normas legales establecidos a casos particulares, de forma que los estudiantes se ejerciten en la selección y aplicación de los principios adecuados a cada situación. Se busca desarrollar un pensamiento deductivo, a través de la atención preferente a la norma, a las referencias objetivas y se pretende que se encuentre la respuesta correcta a la situación planteada. Este es el modelo desarrollado preferentemente en el campo del derecho.

**3.2.3 Modelo centrado en el entrenamiento.** Finalmente, el tercer modelo busca el entrenamiento en la resolución de situaciones que si bien requieren la consideración de un marco teórico y la aplicación de sus prescripciones prácticas a la resolución de determinados problemas, exigen que se atienda la singularidad y complejidad de contextos específicos. Se subraya igualmente el respeto a la subjetividad personal y la necesidad de atender a las interacciones que se producen en el escenario que está siendo objeto de estudio. En consecuencia, en las situaciones presentadas (dinámicas, sujetas a cambios) no se da "la respuesta correcta", exigen al profesor estar abierto a soluciones diversas. Dentro de este modelo se pueden considerar diversos subtipos, establecidos en función de la finalidad didáctica específica que se pretenda en cada situación y, consecuentemente, de las capacidades que se ejerciten. Se hará referencia a tres tipos: Casos centrados en el estudio de descriptores, casos de resolución de problemas, casos centrados en la simulación.

**3.2.3.1 Casos centrados en el estudio de descriptores.** En estos casos se propone como objetivo específico que los participantes se ejerciten en el análisis, identificación y descripción de los puntos clave constitutivos de una situación dada y tengan la posibilidad de debatir y reflexionar junto a otros, las distintas perspectivas desde las que puede ser abordado un determinado hecho o situación. Finalmente, pretenden la reflexión y el estudio sobre los principales temas teórico-prácticos que se derivan de la situación estudiada. No se pretende, pues, llegar al estudio y al planteamiento de soluciones, se centran en aspectos meramente descriptivos. Este tipo de casos, que tiene entidad propia en cuanto análisis descriptivo, constituye el punto de partida de los subtipos que se considerarán más adelante.

En el estudio de este tipo de casos es necesario Mucchielli (1970), centrar el análisis en dos grupos de variables:

1. Descubrir los hechos clave, tanto estáticos como dinámicos que definen la situación. Para ello se precisa considerar:
  - Hechos significativos referidos a las personas implicadas.
  - Hechos relacionados con aspectos y variables contextuales.
  - Hechos vinculados a las relaciones interpersonales.
2. Descubrir las relaciones significativas que se dan entre los distintos hechos:
  - Determinación de los elementos significativos de la situación: Consideración estática. Identificación de los momentos y tiempos decisivos de la situación: consideración dinámica.

**3.2.3.2 Casos de resolución de problemas.** El objetivo específico de este tipo de casos se centra en la toma de decisiones que requiere la solución de problemas planteados en la situación que se somete a revisión.

Las situaciones problemáticas han de ser identificadas previamente, seleccionadas y jerarquizadas en razón de su importancia o de su urgencia en el contexto en el que tienen lugar.

Dentro de este tipo de casos, se pueden considerar, en función de la finalidad específica pretendida, dos subgrupos:

- Casos centrados en el análisis crítico de toma de decisiones: Esta propuesta metodológica pretende específicamente que los participantes emitan un juicio crítico sobre las decisiones tomadas por otro individuo o grupo para la solución de determinados problemas. En este supuesto, la narración debe presentar de manera minuciosa el proceso seguido en la situación descrita explicitando la secuencia de actividades y estrategias empleadas en la solución del problema que se intenta analizar.
- Casos centrados en generar propuestas de toma de decisiones: este grupo de casos pretende el entrenamiento de los participantes en el estudio de situaciones que requieren la resolución de problemas, de manera que se impliquen en el proceso de toma de decisiones que, desde la opinión de los individuos y/o grupo, sea el más adecuado en la situación estudiada. Este tipo de casos suele ser la estrategia más utilizada didácticamente, ya que, como fase previa, incluye el estudio descriptivo de la situación en donde se define el problema al que se intenta dar solución.

**3.2.3.3 Casos centrados en la simulación.** En este tipo de casos no sólo se pretende que los sujetos estudien el relato, analicen las variables que caracterizan el ambiente en que se desarrolla la situación, identifiquen los problemas y propongan soluciones examinando imparcial y objetivamente los hechos y acontecimientos narrados, sino que específicamente se busca que los participantes se coloquen dentro de la situación, se involucren y participen activamente en el desarrollo del caso y tomen parte en la dramatización de la situación, representando el papel de los personajes que participan en el relato.

### **3.3 SELECCIÓN Y ELABORACION DE LOS CASOS**

El aspecto más importante para la elaboración de un caso es su selección de acuerdo a los propósitos metodológicos que se pretende, de tal manera que se pueda cumplir con los objetivos y contenidos del curso seleccionado.

**3.3.1 Búsqueda y selección de los casos.** Según Mucchielli (1970:156-157), para recopilar datos de base que servirán mas tarde para redactar los caso, se dispone de tres medios:

**3.3.1.1 Entrevista a un profesional con experiencia.** Se invita a profesionales veteranos a contar situaciones dramáticas o difíciles vividas durante su carrera. Se escogerán evidentemente profesionales del campo de conocimiento y de acción que interese, correspondiente a fines pedagógicos y se les explicará el objetivo a conseguir. Durante estas intervenciones, se tendrá cuidado:

- De dejar en principio al sujeto invitado a contar todo lo que recuerde del caso, sin seleccionar nada y haciéndole hablar con libertad.
- Escoger a continuación el o los casos que correspondan mejor a los criterios dados.
- Volver en una nueva entrevista sobre el (o los) caso seleccionado, y obtener por medio de preguntas, todos lo datos necesarios para la comprensión de la situación total.
- Conservar las palabras típicas o las referencias profesionales clave.
- Modificar los nombres y hallazgos demasiado precisos que revelarían la identidad de los protagonistas.

**3.3.1.2 Estudio de documentos conservados en archivos de la profesión.**

- Incidentes técnicos que pongan en juego a las personas.
- Faltas que suponen sanciones.

- Litigios, procesos o dramas.
- Problemas de reorganización.

### ***3.3.1.3 Escritos que se refieren a acontecimientos personales o profesionales.***

- Cartas que expongan una situación problemática y una decisión personal.
- Relatos autobiográficos e históricos.
- Confesiones.
- Observación detallada de una situación por uno de los actores principales o por varios de los testimonios convergentes.

## **3.4 ORGANIZACIÓN DE LA TECNICA**

Respecto al proceso que se requiere en el estudio de casos, la mayoría de los autores están de acuerdo en la propuesta de que la secuencia de operaciones que se han de llevar a cabo constituyan un proceso muy estructurado en la forma y con gran flexibilidad respecto a los contenidos. Básicamente se centran en tres etapas:

1. Familiarización con el tema, los personajes y actor principal. Requiere la lectura y formación de una impresión global y general sobre el caso.
2. Análisis detenido del relato que debe finalizar con formulación de los problemas, la naturaleza de las decisiones a tomar y la identificación de los elementos principales que influyen sobre las acciones que se puedan emprender.
3. Preparación de las recomendaciones operativas para dar solución al caso estudiado. Las recomendaciones usualmente contemplan que se comience con una fase de reflexión individual, seguida una puesta en común de las reflexiones individuales y un análisis y propuesta de acción realizada en equipo; finalmente se suele proponer un debate general centrado tanto en los descriptores del caso, como en la propuesta de soluciones y en la consideración de los temas, teorías y prescripciones a las que el caso remita.

Colbert y Desberg (1996) plantean las siguientes fases para el estudio de un caso: Fase preliminar, Fase eclosiva, Fase de análisis, Fase de conceptualización.

**3.4.1 Fase preliminar.** Presentación del caso a los participantes, proyección de la película, audición de la cinta o lectura del caso escrito.

**3.4.2 Fase eclosiva.** "Explosión" de opiniones, impresiones, juicios, posibles alternativas, etc., por parte de los participantes. Cada uno reacciona a la situación, tal como la percibe subjetivamente. Si cada cual se puede expresar libremente, se llega a continuación a un cierto relajamiento de las tensiones del comienzo y desemboca, finalmente, en el descubrimiento de la incompatibilidad de puntos de vista.

Bien llevada, esta fase revela a cada uno lo siguiente:

- Su subjetividad.
- La posibilidad de que existan otras opiniones o tomas de posición tan valiosas como las propias.
- Hasta qué punto los diagnósticos emitidos son proyecciones de la propia persona, más que análisis objetivos de la situación real.

**3.4.3 Fase de análisis.** Se impone una vuelta a los hechos y a la información disponible, para salir de la subjetividad. La búsqueda en común del sentido de los acontecimientos permite a los participantes acrecentar su conciencia de la situación analizada. Se redescubre la realidad y se integran aspectos informativos que, por determinados prejuicios, se habían orillado. La única prueba de objetividad es el consenso del grupo en las significaciones. En esta fase es preciso llegar hasta la determinación de aquellos hechos que son significativos para interpretar la

estructura dinámica de la situación. Se concluye esta fase cuando se ha conseguido una síntesis aceptada por todos los miembros del grupo.

**3.4.4 Fase de conceptualización.** Es la formulación de conceptos operativos o de principios concretos de acción, aplicables en el caso actual y que permiten ser utilizados en una situación parecida. Dicho de otro modo, se trata de gestar principios pragmáticos de acción que sean válidos para una transferencia. Como en la fase anterior, la única garantía de validez y objetividad es el consenso del grupo.

Por lo anterior, el análisis de un caso concreto, aun en su singularidad, es un camino seguro hacia las leyes generales del tema considerado en él. Principalmente por lo siguiente:

- Análisis no es buscar causas recónditas en el subconsciente. Es simplemente relacionar los datos actuales de una situación, captar su configuración y evolución, encontrar la significación de cada uno por la posición que ocupa en el campo situacional global.
- Conceptualización es algo pedagógicamente esencial después del análisis. Significa que es preciso formular expresamente los conceptos clave que se deducen del caso. Pero se trata de una "conceptualización operativa": las ideas generales extraídas del caso no son leyes abstractas, sino certezas de conducta que se deben adquirir. Servirán para afrontar directamente situaciones similares en la vida real.

### **3.5 APRENDIZAJES QUE FOMENTA LA TÉCNICA DE ESTUDIO DE CASOS**

Esta técnica de trabajo tiene un notable interés en aquellas áreas que requieren un entrenamiento para la formación teórico-práctica de los estudiantes. De acuerdo con varios autores como Alfonso López (1997) y otros profesores con vasta experiencia en esta área, los estudiantes pueden desarrollar, a través del uso de esta técnica:

- Habilidades cognitivas como pensamiento crítico, análisis, síntesis, evaluación.
- Aprendizaje de conceptos y aplicación de aquéllos aprendidos previamente, tanto de manera sistemática como por la experiencia propia.
- La habilidad para trabajar en grupo y la interacción con otros estudiantes, así como la actitud de cooperación, el intercambio y la flexibilidad, lo cual constituye una preparación eficaz para las relaciones humanas.
- El acercamiento con la realidad, la comprensión de fenómenos y hechos sociales, familiarizarse con las necesidades del entorno y sensibilizarse ante la diversidad de contextos y diferencias personales, el mejoramiento en las actitudes para afrontar problemas humanos.
- El desbloqueo de actitudes inseguras o temerosas.
- La disposición a la escucha comprensiva.
- El entrenamiento dinámico de la autoexpresión, la comunicación, la aceptación, la reflexión y la integración.
- La motivación por el aprendizaje, ya que los alumnos por lo general encuentran el trabajo de estudio de casos más interesante que las lecciones magistrales y la lectura de libros de texto.
- Los procesos de toma de decisiones.

### **3.6 LA EVALUACION DE LA TECNICA DE ESTUDIO DE CASOS**

A los estudiantes se les dice que no hay respuestas correctas o incorrectas al analizar y resolver casos. También se les menciona que no traten de convencer al maestro de ejecutar los cursos de acción o soluciones que ellos proponen. Aquí el trabajo del profesor consiste en ayudar a los estudiantes a identificar y desarrollar contextos basados en prácticas específicas partiendo de teorías o modelos.

Cuando los estudiantes analizan un caso, definen los problemas, clarifican dudas, ponderan las alternativas y escogen un curso de acción. Estas habilidades integran la reflexión crítica. Se requiere que los estudiantes utilicen conocimiento práctico y

teórico para analizar minuciosamente y reestructurar un caso. El responder a las preguntas sobre un caso hace que el proceso de pensamiento de los estudiantes sea explícito.

Para evaluar el progreso de los estudiantes, es útil dividir las habilidades de un análisis de casos en tres partes: identificación de los hechos, identificación del problema y solución del mismo, por lo tanto, la evaluación del estudiante se establece en la medida en que haga explícitas sus preguntas, su proceso de información y sus soluciones. Además se puede señalar lo siguiente:

- Los alumnos deben tener claros objetivos de aprendizaje que se desean lograr al discutir el caso correspondiente.
- El profesor debe indicar qué aspectos se tomarán en cuenta para la evaluación:
  - Elaboración de alguna actividad previa a la discusión del caso.
  - Entrega de alguna tarea previa al inicio del análisis del caso (resumen, reporte, cuadro sinóptico, mapa conceptual, etc.).
  - Participación de los alumnos en la discusión (intervenciones, planteamiento de dudas, aporte de información, motivación a los compañeros para participar).
  - Actividades posteriores a la discusión del caso (tarea, resumen, consulta, conclusión individual o de equipo, etc.).
- Es recomendable para el profesor elaborar un formato que le permita registrar la evaluación a lo largo del proceso.

### **3.7 DIFICULTADES Y BARRERAS PARA PODER EN PRÁCTICA LA TÉCNICA**

No se trata simplemente de un método de instrucción o de capacitación operativa, es un método de formación en profundidad, que genera actitudes favorables, conduce al análisis metódico y entrena para una correcta toma de decisiones.

A continuación se presentan algunos puntos que pueden representar dificultades al implementar el método de casos:

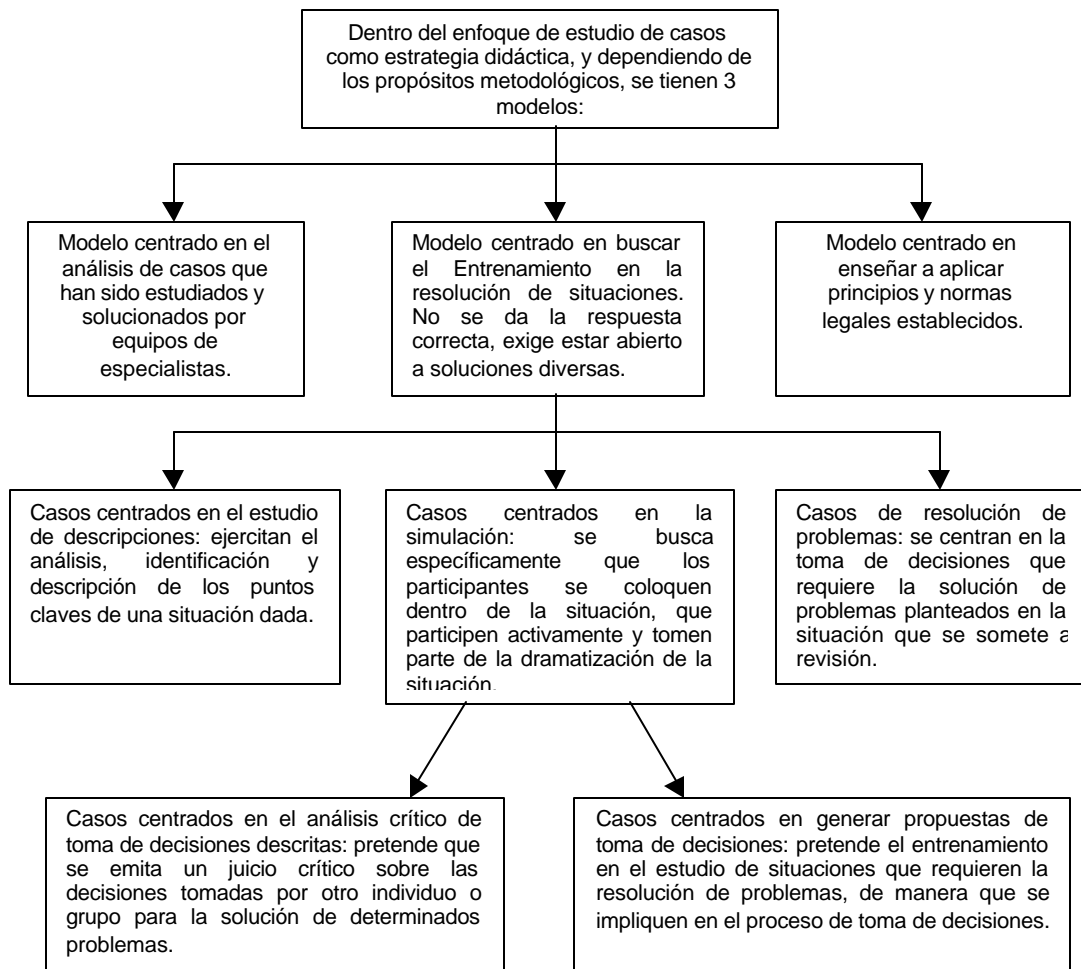
- La inexistencia de respuestas correctas pueden ser difíciles de aceptar para algunos alumnos.
- Por no ser una estrategia tradicional es difícil comunicar a la comunidad lo que los estudiantes están haciendo y aprendiendo.
- Los grupos son de alumnos son numerosos y por tanto se puede perder el control del grupo.
- Los estudiantes, sobre todo los más jóvenes, se pueden perder en la tarea de resolver el caso y olvidar sus propósitos de aprendizaje.
- Diseñar una evaluación válida puede parecer complejo y difícil en el método de casos.
- Es difícil hacer que encajen las estrategias de evaluación con las metas de aprendizaje.
- El tiempo de discusión del caso debe ser administrado adecuadamente, de lo contrario se puede perder la atención del grupo.
- El caso debe versar sobre una temática que incluya a la mayor parte del grupo, de otro modo algunos alumnos pueden perder interés por su participación en la discusión del caso.
- Los alumnos pueden percibir poca relación del caso con los contenidos de aprendizaje del curso.
- Se puede perder el orden cuando los alumnos se identifican y apasionan por una posición en torno al caso.

### **3.8 RESUMEN**

En este capítulo se abordó la metodología didáctica del estudio de casos como una alternativa para la formación de futuros profesionales (estudiantes, cursillistas) en diferentes áreas de conocimiento. Se presentó tres modelos que se diferencian en razón de los propósitos metodológicos (ver figura 5) y los elementos necesarios

para su análisis y empleo. Igualmente, se mencionó los aprendizajes que fomenta y algunas dificultades para poner en práctica la técnica.

Figura 5. Esquema de los modelos metodológicos del estudio de casos



Fuente: <http://www.sistema.itesm.mx/va/dide/inf-doc/estrategias/>

## Capítulo 4

# APRENDIZAJE DE LA AUDITORÍA DE SISTEMAS DE INFORMACION APOYADONDA SE EN LA DIDACTICA DEL ESTUDIO DE CASOS

### 4.1 INTRODUCCION

A menudo la manera en que hemos aprendido a pensar e interactuar con los demás ha ido evolucionando en la medida en que reconocemos la necesidad de vernos introspectivamente de una forma nueva. La educación esta cambiando y que cada vez involucrará las tecnologías de información con el apoyo de modelos didácticos que faciliten la apropiación de contenidos (contenidos que enseñen como, los grupos de trabajo en el desarrollo tecnológico hacen para resolver problemas o desafíos que se les presenta) para operar en la realidad.

Estos cambios en la educación tienen que ver con un apoyo didáctico, como también de una mejor definición de los objetos de estudio que exigen la necesidad de aplicar los conocimientos en la practica.

Por ello el camino que se ha empezado a recorrer al realizar esta investigación en el área de auditoría de sistemas debe involucrar estos puntos de vista para encontrar nuevos entendimientos que abrirán la forma de emprender nuevas alternativas de acción con un alto apalancamiento en los problemas de interés.

### 4.2 DESCRIPCION DEL PROBLEMA

Los sistemas informáticos se han convertido en un poderoso medio para cubrir las áreas del negocio en todos los niveles, impulsando a la organización hacia la sociedad del conocimiento, que sin lugar a dudas, exige un cambio radical en las

formas de operar para poder adaptarse al mercado con mucha rapidez. Así, las empresas que han podido afrontar estos cambios, viven en un ambiente informatizado, donde es absolutamente necesaria la integración empresarial a través de las telecomunicaciones y un sin número de componentes de tecnología, que entre otras cosas son evaluados continuamente para asegurar la utilización correcta de los recursos y así cumplir con los objetivos de la organización.

Lamentablemente, muchas de las medidas que han tomado las organizaciones conducen a crear complicaciones en sus procesos, lo que ha generado un descontento por el desconocimiento de las situaciones a las que se ven enfrentadas y que pueden suscitar una incertidumbre con respecto al manejo de la información producida en ese momento.

Estas situaciones incrementan las expectativas relacionadas con la informática y la tecnología, obligando a las organizaciones a disponer de políticas, procedimientos y controles que aseguren la calidad en la información para así brindar ventajas competitivas frente a sus similares, pero que pueden originar costos elevados y desventajas si no son bien administradas.

Sin lugar a dudas, la información y el conocimiento tienen un impacto impresionante sobre la organización, por ello las empresas han adoptado una nueva forma de examinar sus procesos: a través de la auditoría de sistemas o informática; área que se ha convertido en un proceso dispendioso, donde el profesional de auditoría recopila gran cantidad de información que analiza, detalla y evalúa, para poder emitir una opinión con respecto al área auditada (diagnostico de la situación).

Con el interés puesto en esta área, se ha identificado una deficiencia en el nivel de formación<sup>10</sup> caracterizada por la desvinculación del conocimiento teórico con el

---

<sup>10</sup> Se puede corroborar al analizar los escasos trabajos de grado de auditoría y su formación a través de la Materia de Auditoría ofrecida como electiva en la escuela de sistemas de la Universidad Industrial de Santander (UIS).

práctico y la insuficiencia de material bibliográfico en español con suficiente claridad aplicable a nuestro entorno.

Cabe resaltar que de ninguna manera sé esta siendo pesimista, puesto que la escuela de sistemas viene afrontando cambios estructurales de fondo que sin lugar a dudas nos pone en una etapa de exploración e investigación en esta y otras áreas importantes.

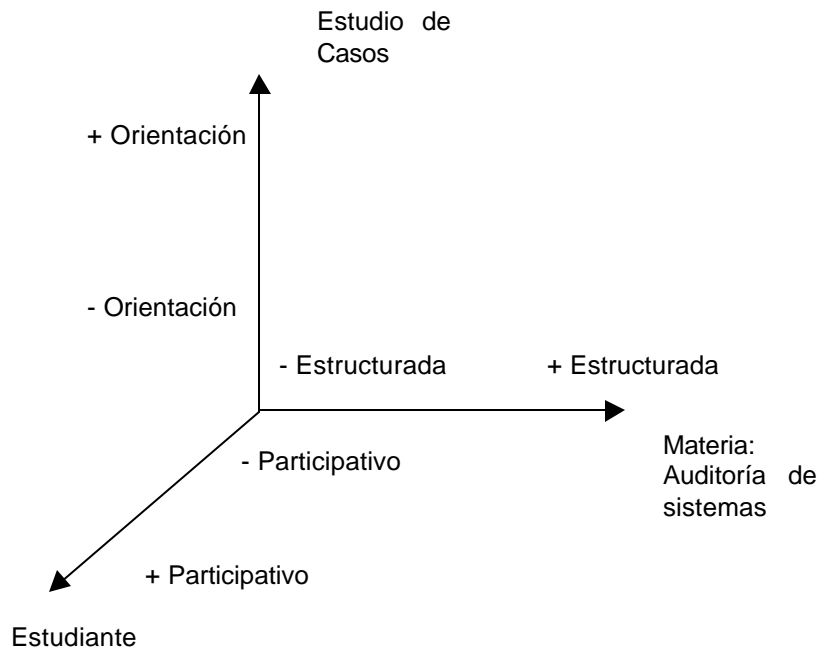
### **4.3 DESCRIPCION DE LA PROPUESTA: UNA HERRAMIENTA SOFTWARE**

La herramienta software elaborada en este proyecto, pretende servirse de la orientación didáctica del modelo centrado en el análisis de casos (capítulo 3) que ofrece la metodología del Estudio de casos, y considerando la esquematización expuesta en el capítulo 2 sobre las partes o flujos de trabajo en la auditoría.

La conveniencia en el uso de la orientación metodológica del estudio de casos para la construcción de una herramienta software aplicable al aprendizaje de la auditoría de sistemas puede analizarse desde tres factores o dimensiones: Estudiante, Materia, Estudio de casos.

Las variables expresadas en cada dimensión (Orientación, Estructuración, Participación) son algunos aspectos que se pueden analizar en cada dimensión. ¿El estudio de casos da la suficiente orientación sobre un contenido de auditoría? ¿Los contenidos de la materia deben estar rígidamente estructurados? ¿Hasta que punto el estudiante es mas participativo?

Figura 6. Factores o dimensiones a tener en cuenta para el aprendizaje de auditoría



**4.3.1 Dimensión de la materia.** Al pretender orientar el estudio de la materia de auditoría mediante metodología del Estudio de casos se obtiene el potencial adecuado para asociar y adecuar los contenidos presentados como fundamento teórico, permitiendo un análisis y una conceptualización en el estudio de casos, dejando de lado la rigidez con la cual se estructura la materia.

**4.3.2 Dimensión del estudiante.** El Estudio de casos afecta directamente las habilidades del aprendiz ya que posibilita la comunicación entre participantes del proceso de aprendizaje abriendo nuevas formas de pensamiento crítico que permiten aumentar su nivel de participación.

**4.3.3 Dimensión del estudio de casos.** El éxito de la metodología del estudio de casos radica en propiciar ambientes en los cuales en el estudio de un caso creado por un auditor se logre una asimilación de conocimientos teóricos y que al

presentarse una situación práctica se logre identificar los problemas con mayor facilidad para así llegar a un determinado diagnóstico, si es el caso.

Cuando se habla de propiciar ambientes nos referimos a un espacio académico para el trabajo individual o grupal. Y se habla de Individual, en el momento en que el aprendiz es capaz de identificar los aspectos concretos de determinada realidad o caso expuesto, y grupal, cuando es capaz de interactuar con las demás formando redes de colaboración en donde cada uno de los aportes permiten un análisis detallado y una aclaración de los conceptos y procedimientos utilizados.

## **4.4 OBJETIVOS DEL PROYECTO**

### **4.4.1 Objetivo general.**

Elaborar una propuesta de herramienta software de apoyo en el aprendizaje de la Auditoría de Sistemas de información, dirigida por planes, centrado en pruebas y organizado por casos; siguiendo la metodología del Proceso Unificado de Desarrollo de Software.

### **4.4.2 Objetivos específicos.**

- ▶ Elaborar una propuesta de herramienta software apoyándose en la metodología didáctica del estudio de casos y que permita:
  - ✓ Recopilar información del proceso de auditoría para su organización en casos de estudio.
  - ✓ Identificar, evaluar y estudiar los procesos que realizan los auditores en el manejo de la información.
  - ✓ Apoyar y fortalecer el aprendizaje de la materia de auditoría ofrecida como electiva en la escuela de Ingeniería de sistemas.

- ▶ Elaborar una herramienta software siguiendo la metodología del proceso unificado de software y con las siguientes características:
  - ✓ Permitir la creación y selección de un caso de auditoría de acuerdo a las características de la empresa auditada.
  - ✓ Permitir la definición de perfiles y parámetros de manejo de la información en principiantes, colaboradores y auditores.
  - ✓ Proporcionar los mecanismos necesarios para realizar una planeación del proceso de auditoría.
  - ✓ Capturar y vincular toda la información que el auditor considere importante para el proceso.
  - ✓ Llevar una estructura de seguimiento a las actividades realizadas durante el proceso de auditoría (En Hojas de trabajo), y la verificación de los procesos efectuados.
  - ✓ Llevar un registro de pruebas a efectuar durante el proceso, estas pruebas deben estar perfectamente descritas, además de los resultados que podrían obtenerse.
  - ✓ Permitir la generación e impresión de reportes necesarios para realizar una conclusión final (diagnóstico).

## **4.5 IMPACTO**

El desarrollo de este proyecto brindará la posibilidad de aprovechar el estudio de casos para la formación de estudiantes apoyando y reestructurando los cursos actuales auditoría de sistemas. En la auditoría influye de manera considerable la experiencia, es decir, mientras más auditorías se hayan elaborado, se va adquiriendo conocimiento o se va obteniendo mayor dominio en su desarrollo, por esta razón la necesidad de implementar la metodología del estudio de casos ya que esta va a permitir el desarrollo de cursos de auditoría en un nivel formativo.

También permitirá a los estudiantes desempeñar roles que permitan una comprensión de los procesos de auditoría en un ambiente académico y práctico, creando los mecanismos de reflexión (Incremento y desarrollo de sus capacidades de análisis) y autoevaluación en su proceso de aprendizaje.

Para el futuro ingeniero de sistemas es de vital importancia, debido a éste que debe poseer conocimiento sobre los procesos de auditoría de sistemas ya sea que pretenda desempeñarse en el área o que en algún momento, necesite apoyar algún procedimiento de auditoría sobre el sitio en la cual labora.

Además el impacto que traerá consigo la metodología del Estudio de casos en los procesos pedagógicos permitirá un estudio continuo, eficiente y completo para su aplicación en las diversas áreas y procesos realizados por los seres humanos.

## PARTE II

### DESARROLLO DE LA HERRAMIENTA SOFTWARE

En esta parte se expondrán los detalles necesarios para la construcción de la herramienta software propuesta empleando El proceso unificado de desarrollo de software; sin embargo nuestro interés no es describir los conceptos y características del proceso de modelado del software a través de este desarrollo; para ello se ha desarrollado el apéndice A y B como apoyo en la resolución de cualquier duda presentada.

Los objetivos específicos de esta parte son:

- ✓ Definir el ámbito y los límites de la herramienta software propuesta.
- ✓ Determinar la viabilidad del proyecto al identificar los riesgos que lo pueden afectar.
- ✓ Guiar al desarrollador en la implementación del proyecto.

En esta parte se mostrará el desarrollo del proyecto enmarcado dentro de las 3 primeras fases de proceso unificado y recorriendo cada flujo de trabajo fundamental: requisitos, análisis, diseño, implementación y prueba, dejando fuera la fase de transición para un proyecto de grado futuro, dadas las implicaciones que esta fase tiene en el tamaño y duración total y a la no disponibilidad inmediata para su implantación.

# Capítulo 5

## LA FASE DE INICIO

Esta fase del Proceso Unificado del Software establece el punto de partida para el desarrollo de la herramienta propuesta. En ella, se desarrolla un análisis de la situación objeto de estudio, de donde se pueden identificar varias actividades de alto nivel, las cuales va a permitir plantear un primer esquema de la arquitectura.

Se ha considerado dentro de esta fase, la planificación, descripción y reunión de los criterios de evaluación siguiendo los flujos de trabajo fundamentales propuestos en el Proceso de desarrollo del Software.

El trabajo realizado durante esta fase permitió definir algunos de los objetivos del plan de proyecto, delimitar el alcance de la herramienta propuesta y mitigar algunos riesgos críticos para anticipar algunas de las dificultades que surjan en fases posteriores.

Para evitar describir nuevamente y entrar en detalles del Proceso Unificado de Desarrollo del Software se le recomienda al lector revisar el apéndice A donde encontrará algunas pautas que pueden resolver ambigüedades respecto a este.

### 5.1 PLANIFICACIÓN DE LA FASE DE INICIO

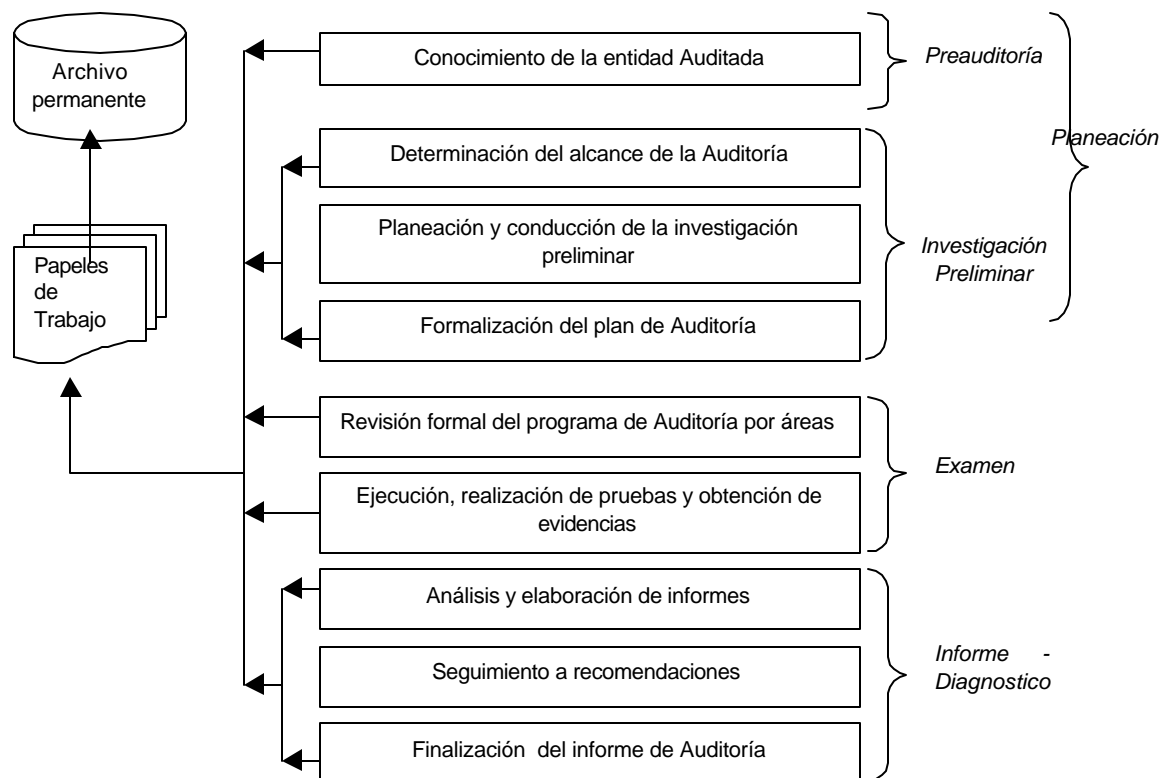
Esta fase se realizará en una sola iteración. En ella se empezará por establecer un modelado del negocio para identificar las características principales con que contará la herramienta software propuesta. En el desarrollo de esta fase se contemplarán los flujos de trabajo de requisitos, análisis y diseño. Las actividades correspondientes a los flujos de implementación y pruebas se postergarán en fases posteriores.

## 5.2 MODELADO DEL NEGOCIO

**5.2.1 Identificación de los procesos del Negocio.** Como situación objeto de estudio se ha tomado el aprendizaje de la Auditoría de Sistemas de información a través de casos de estudio, el cual en adelante se le hará referencia como "El Negocio".

El primer paso para el modelado del negocio es la identificación de sus procesos. En este punto simplemente se listarán algunos procesos observados para luego abordarlos uno a uno.

Figura 7. Esquema metodológico del proceso de auditoría



Cabe aclarar que existen diferentes metodologías que se pueden encontrar en el mundo de la auditoría y que respaldan el proceso formal del negocio. Por ello, se ha recurrido a la elección de un esquema metodológico<sup>11</sup>(ver figura 7); del cual se han extraído un conjunto de actividades y tareas que aseguren un previo entendimiento.

**5.2.2 Identificación de Usuarios y elementos implicados en el proceso del negocio.** El proceso del negocio arranca cuando es posible organizar la información del proceso de auditoría en casos de estudio, esta información es proporcionada por Revisores y Especialistas en el área (Auditor). Dicho caso de auditoría debe estar estructurado de acuerdo al esquema metodológico presentado (figura 7) y suministra la información necesaria para el conocimiento y comprensión de los procesos de diagnóstico llevados a cabo.

Por otro lado, se encuentra el aprendiz de auditoría quien solicita un caso de estudio para llevar a cabo un proceso de estudio y análisis a fin de obtener un proceso de aprendizaje, apoyándose en la metodología didáctica del estudio de casos.

Por lo tanto, los agentes implicados en el proceso del negocio son:

- ✓ Auditor.
- ✓ Revisor.
- ✓ Principiante de auditoría (aprendiz).

**5.2.3 Establecer las acciones necesarias para realizar el proceso de negocio.**

Se puede abordar este apartado tratando de describir en forma informal la interacción entre roles para que se cumpla el proceso de negocio con éxito.

Un ejemplo de interacción entre roles podría ser:

---

<sup>11</sup> En el capítulo 2 se presentan los detalles respecto al esquema metodológico presentado.

*El principiante de auditoría solicita un Caso de Auditoría para estudio. El auditor y el revisor proporcionan los casos de auditoría para estudio. Este ultimo decide el nivel de detalle que desea mostrar del caso. El principiante decide tomar el caso de estudio y de acuerdo al nivel de detalle lo analiza (estudio).*

En la secuencia de acciones anterior se pueden identificar las acciones que realiza cada agente. A continuación se mostrara una lista de los agentes y las acciones que realizan cada uno de ellos.

### **Aprendiz de auditoría**

- ✓ Solicitar caso de estudio (Búsqueda).
- ✓ Estudiar caso de auditoría.

### **Auditor**

- ✓ Crear el caso de auditoría.
- ✓ Recopilar la información necesaria almacenar el caso de estudio
- ✓ Organizar la información del caso.
- ✓ Establecer el nivel de detalle que desea mostrar del caso.

### **Revisor**

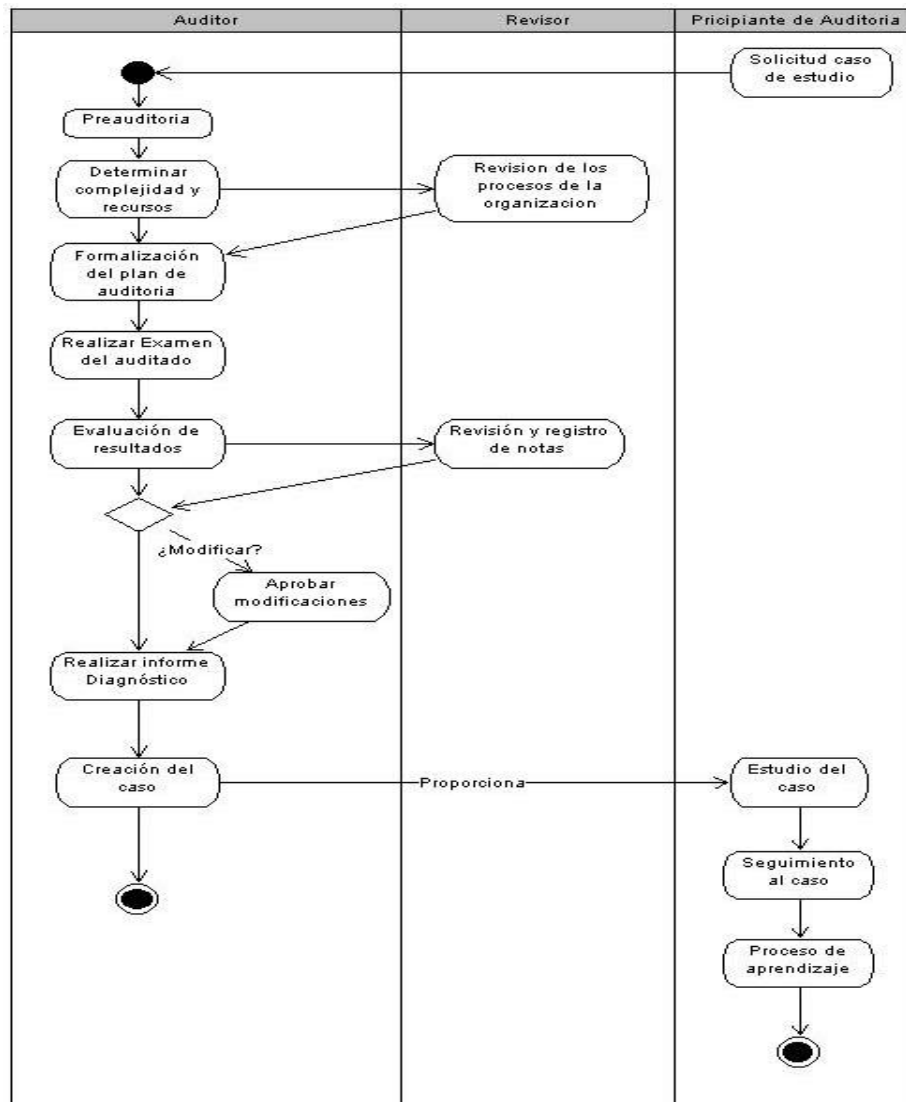
- ✓ Proporciona cada etapa del proceso de auditoría revisado.

**5.2.4 Construcción de un diagrama de actividades que represente el proceso de negocio.** Haciendo uso de los elementos de los diagramas de actividades de UML se tratará de mostrar de forma más rigurosa y ordenada el flujo de actividades del proceso de negocio.

En el siguiente diagrama se mostrará algunas acciones del proceso del negocio con el fin de obtener un producto final: El caso de auditoría que le servirá de estudio al aprendiz o principiante de auditoría. Sin embargo se aclara que la obtención de un caso no esta determinada por condiciones ni limitaciones fijadas por del Aprendiz.

Las características del caso están determinadas por el auditor quien es el especialista en el área. Así, observando el diagrama se mostrarán solo algunas actividades para luego abordarlas con mayor detalle en la siguiente fase.

Figura 8. Flujo de actividades del proceso de Negocio



**5.2.5 Lista de Actividades.** La lista de actividades que se realizan en el proceso de negocio son las siguientes:

- ✓ Solicitar caso de estudio
- ✓ Estudiar caso.
- ✓ Recopilación de información y organización del caso.
- ✓ Mostrar los flujos de trabajo en el caso (Planeación- Evaluación - Informe - Diagnóstico).
- ✓ Crear caso de estudio.
- ✓ Revisar proceso efectuado.
- ✓ Proporcionar caso de estudio.
- ✓ Recopilación de evidencias - pruebas

Es interesante dar una lista de actividades porque, en general, cada una de esas actividades estará asociada a un caso de uso. También se debe dar una especificación de las actividades. La especificación de las actividades ayuda a comprender y a detectar ambigüedades en los requerimientos en una fase temprana del desarrollo.

**5.2.6 Reglas del negocio.** Se pueden considerar las reglas de negocio como una serie de restricciones a la hora de realizar una determinada actividad.

En el proceso de negocio se definirán las siguientes reglas:

- ✓ Crear una herramienta software que apoye el aprendizaje de la auditoría de sistemas de información apoyándose en la didáctica del estudio de casos y esquema metodológico de auditoría.
- ✓ Cuando se realice un caso de estudio, la información suministrada por el auditor es para uso académico, por consiguiente el auditor decide el nivel de detalle que desea mostrar del caso.
- ✓ Todo caso de auditoría debe conservar el mismo esquema de trabajo definido en la figura 7.

- ✓ Un caso proporcionado por el auditor debe tener fecha de apertura y cierre. No deben existir casos que estén abiertos.
- ✓ Se considera el hecho de realizar un caso de auditoría a partir de un caso existente.
- ✓ Un caso de auditoría proporcionado al principiante no se puede modificar.
- ✓ El acceso a la información del caso por parte del principiante implica la utilización de la metodología didáctica del estudio de casos y el conocimiento conceptual de Auditoría.

### **5.3 MODELADO DE REQUISITOS**

Una vez realizado el modelado del negocio, se pasará a realizar una lista con las características de la herramienta software propuesta en el objetivo general del actual proyecto y que servirá de apoyo al proceso de negocio.

#### **5.3.1 Flujo de Trabajo de requisitos.**

##### ***5.3.1.1 Lista de Características o requisitos candidatos.***

- ✓ La herramienta software debe trabajar en modo monousuario.
- ✓ Permitir la definición de personal y esquema organizativo para el manejo de información. (Personal participante, áreas, Hojas de trabajo).
- ✓ Capturar la información característica del auditado.
- ✓ Estructurar la forma en que se organizará la información.
- ✓ Debe incluir un proceso de autenticación para usuarios auditores y principiantes de auditoría.
- ✓ Proporcionar los mecanismos necesarios para realizar una planeación del proceso de auditoría.
- ✓ Llevar una estructura de seguimiento a las actividades realizadas durante el proceso de auditoría (Hojas de trabajo), y anotaciones por parte de revisores.

- ✓ La herramienta debe capturar y vincular toda la información que el auditor considere importante para el proceso.
- ✓ Permite seleccionar un caso de auditoría de acuerdo a las características de la empresa auditada.
- ✓ Debe llevarse un historial de actividades, técnicas empleadas, participantes, documentos anexos, comentarios, pruebas realizadas, evidencias obtenidas durante todo el proceso de auditoría.
- ✓ La herramienta debe permitir estudiar Casos de auditoría realizados con anterioridad, los cuales pueden servir de base a la realización de nuevos casos.
- ✓ Debe permitir la generación e impresión de reportes de la conclusión final (diagnóstico).
- ✓ La herramienta debe llevar un registro de pruebas a efectuar durante el proceso, estas pruebas deben estar perfectamente descritas, además de los resultados que podrían obtenerse.
- ✓ Realizar un proceso de cierre del caso de auditoría que asegure su finalización. Para lo cual no se admitirá mas información. En este caso, solo se permitirá el acceso para consultas.

### **5.3.1.2 Identificación de Actores**

- ✓ Auditor
- ✓ Revisor
- ✓ Aprendiz
- ✓ Tutor

### **5.3.1.3 Casos de Uso Candidatos**

*Tabla 11. Casos de uso candidatos*

CASO DE USO	ACTORES	PRIORIDAD
Identificación del usuario con login.	Principiante, Auditor	Importante
Crear caso de Auditoría.	Auditor	Crítico
Establecer parametrización de acceso a información	Auditor	Importante
Registro del usuario.	Auditor, Revisor, Aprendiz	Importante
Consulta de información.	Aprendiz, Auditor	Secundario

Establecer plan.	Auditor	Importante
Registrar pruebas.	Auditor	Importante
Registro de información en Hojas de trabajo.	Auditor, Revisores	Importante
Seleccionar caso de auditoría.	Aprendiz	Secundario
Estudiar caso.	Aprendiz	Importante
Crear Diagnostico.	Auditor	Critico
Consultar Resultados.	Aprendiz, Auditor	Importante

**5.3.1.4 Identificación de Riesgos Críticos.** La siguiente tabla resume los riesgos críticos que se han identificado en esta fase, además se incluye las contingencias necesarias en caso de que se manifieste el riesgo.

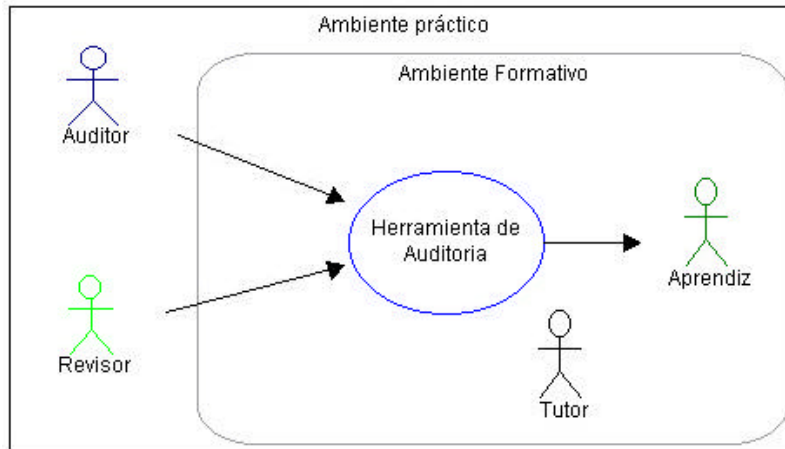
*Tabla 12. Identificación de riesgos críticos para la fase de inicio*

Descripción	Prioridad	Impacto	Responsable	Contingencia
Retardos en el Tiempo estimado	Critico	Se ve afectado el desarrollo total del proyecto	Director y Autor	Recurrir a mecanismos de evaluación y seguimiento periódico
Daño y destrucción de archivos y equipo	Critico	Detenimiento total del proyecto	Autor	Realizar copias de seguridad periódicas
Desventajas con el manejo de la herramienta de programación.	Rutinario	Retardo en el desarrollo del software	Autor	Apoyo en tutores y especialistas del lenguaje
Aceleración en la estimación de casos de uso	Significativo	Redefinición de los objetivos del proyecto	Director y Autor	Delimitar el alcance – el ámbito - de la herramienta

**5.3.1.5 Contexto de la herramienta software.** Con la herramienta software y la metodología didáctica del estudio de casos se pretende apoyar el proceso de negocio como estrategia para lograr un aprendizaje en un ambiente formativo.

En la figura 9 se han identificado dos ambientes de trabajo. Un ambiente práctico en el que participan el auditor y el revisor en la construcción de casos de estudio, y que van a estar organizados en una herramienta software. El segundo ambiente corresponde al formativo en el cual utilizando la metodología didáctica del estudio de casos y la herramienta software se pretende lograr un aprendizaje.

Figura 9. Contexto de la herramienta software.



**5.3.1.6 Descripción de los Casos de Uso.** Para la descripción de los casos de uso se utilizará la plantilla del anexo A<sup>12</sup>, con el fin de documentar cada uno.

Tabla 13. Descripción del caso de uso - Identificación de usuario con login.

Caso de uso: Identificación del usuario con login
Objetivo: Realizar la identificación del rol desempeñado en la herramienta.
Actores: Auditor, Revisor, Principiante.
Precondiciones:
Pasos: 1. A: Indicar el perfil de usuario (Auditor - Revisor - Principiante) Nombre de usuario y password de acceso. 2. H: Comprobar que la existencia de ese usuario en el rol. 3. H: Permitir el registro de un nuevo usuario.
Variaciones: 2.a. Existe un usuario para ese rol: 2.a.1. Indicar error. 2.a.2. Finalizar cdu (caso de uso)
Extensiones:
Cuestiones:

<sup>12</sup> Para mas información véase el apéndice A– El proceso unificado de desarrollo de software

**Tabla 14. Descripción del caso de uso - Crear caso de auditoría.**

<b>Caso de uso: Crear caso de auditoría</b>
Objetivo: Permite al auditor, definir un caso de auditoría, datos del auditado – información preliminar. Con lo cual se establece la apertura y cierre del proceso.
Actores: Auditor.
Precondiciones: Identificarse como auditor.
Pasos: 1. A: Indicar el perfil, nombre de usuario y password de acceso. 2. A: Definir estructura de organización y características del auditado. 3. A: Asignar revisores al proceso. 4. H: Permite el almacenamiento de datos 5. H: Valida los datos suministrados
Variaciones: 3.a. Existen revisores para el caso. 3.a.1. Indicar error. 3.a.2. Registrar nuevos Revisores 3.a.3. Finalizar caso de uso
Extensiones: Crear caso a partir de auditado existente.
Cuestiones:

**Tabla 15. Descripción del caso de uso – Establecer parámetros.**

<b>Caso de uso: Establecer parametrización de acceso a información</b>
Objetivo: Permitir el acceso a los datos del caso que será proporcionado por el auditor.
Actores: Auditor
Precondiciones: Haber creado caso de estudio.
Pasos: 1. H: Muestra las opciones que permiten limitar el acceso a la información. 2. A: El auditor establece limitantes en el acceso a la información.
Variaciones:
Extensiones:
Cuestiones:

**Tabla 16. Descripción del caso de uso – Registro del usuario.**

<b>Caso de uso: Registro de información.</b>
Objetivo: Permitir la inclusión de datos del usuario.
Actores: Auditor, Revisor, Aprendiz.
Precondiciones:
Pasos: 1. A: El usuario ingresa los datos correspondientes a su registro para poder acceder a la herramienta. 2. H: Almacena los datos en su base de datos.
Variaciones:
Extensiones: Modificar datos del registro.
Cuestiones:

**Tabla 17. Descripción del caso de uso – Establecer plan de auditoría.**

<b>Caso de uso: Establecer Plan de auditoría</b>
Objetivo: Permitir definir el modelo de plan a seguir
Actores: Auditor, Revisor
Precondiciones: Haber creado caso de estudio.
Pasos: A: El auditor usa el modelo de planeación definido. A: Define las actividades a desarrollar en la auditoría en forma de pasos. A: Crea y asigna participantes para cada actividad definida. A: Establece recursos necesarios para la elaboración del proceso.
Variaciones:
Extensiones: Cargar un modelo de planeación existente.
Cuestiones: ¿Puede el actor revisor modificar este modelo?

**Tabla 18. Descripción del caso de uso – Establecer hojas de trabajo.**

<b>Caso de uso: Establecer hojas de trabajo</b>
Objetivo: Permitir definición de hojas de trabajo para el desarrollo del proceso de auditoría.
Actores: Auditor, Revisor
Precondiciones: Haber creado caso de estudio y plan de trabajo.
Pasos: 1. A: El auditor define objetivos, alcance y desarrollo de actividades 2. A: Define las técnicas para el desarrollo de las pruebas 3. H: Permite Almacenar la descripción de pruebas y las técnicas para su desarrollo.
Variaciones:
Extensiones: Obtención de evidencias
Cuestiones:

**Tabla 19. Descripción del caso de uso – Crear diagnóstico.**

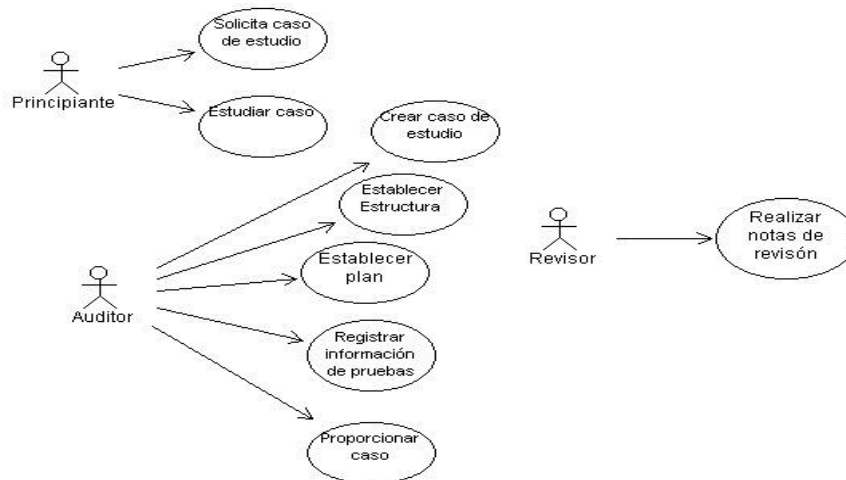
<b>Caso de uso: Crear diagnóstico</b>
Objetivo: Plasmar los resultados del proceso de auditoría, la opinión y recomendaciones
Actores: Auditor
Precondiciones: Haber recopilado datos de indagación y resultados de pruebas.
Pasos: 1. A: Analiza los datos recopilados y los resultados de las pruebas para construir un informe. 2. A: Reúne comentarios y evidencias 3. A: Establece un seguimiento para próximas auditorías. 3. H: Muestra resultados del trabajo y registro del diagnóstico dado.
Variaciones:
Extensiones:
Cuestiones:

Tabla 20. Descripción del caso de uso – Estudiar caso

Caso de uso: Estudiar Caso
Objetivo: El Aprendiz podrá analizar el proceso efectuado por el auditor para comprender sus resultados
Actores: Aprendiz
Precondiciones: El caso debe estar terminado.
Pasos: 1. A: El Principiante se identifica en la herramienta. 2. A: El principiante selecciona el caso para estudio. 3. A: Analiza y verifica los procesos efectuados. 3. H: Genera los resultados la selección del caso. Estructura de auditoría (Planeación, Obtención de evidencias, informes y recomendaciones)
Variaciones:
Extensiones:
Cuestiones:

El proceso que se ha seguido hasta el momento permite construir un diagrama inicial de casos de uso.

Figura 10. Diagrama inicial de casos de uso



**5.3.1.7 Detalle de casos de Uso.** Dada la importancia del caso de uso Crear caso de auditoría, se ha seleccionado para mostrarlo en detalle.

Figura 11. Crear caso de auditoría.

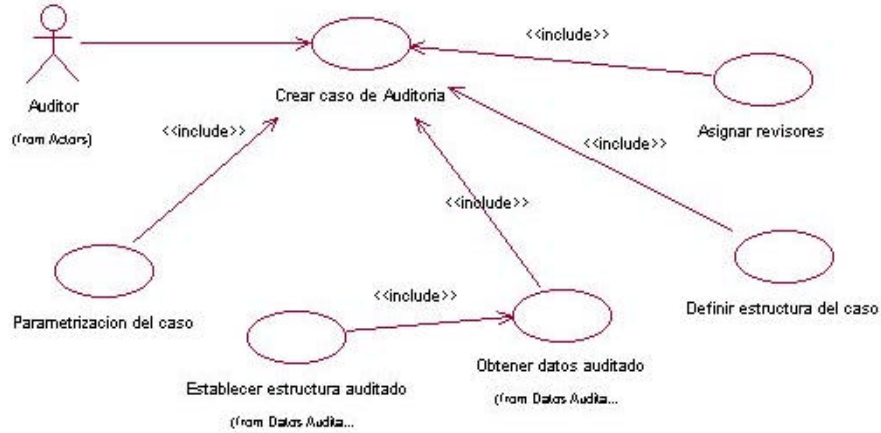


Figura 12. Diagrama de estados para el caso crear nuevo caso de auditoría

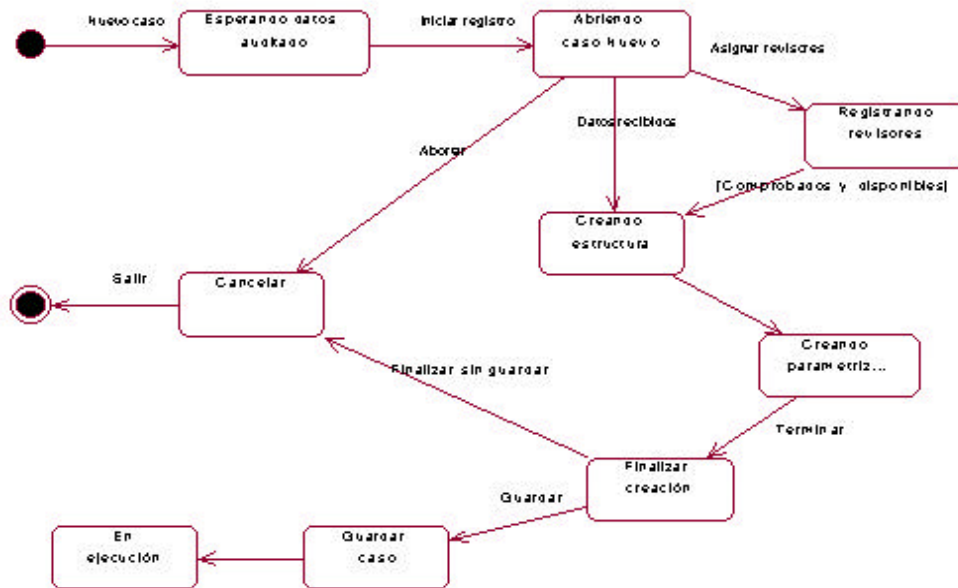


Tabla 21. Detalle del caso de uso Crear caso de auditoría.

Caso de Uso: Crear caso de Auditoría	
Precondición	El auditor se ha identificado en la herramienta
Descripción o Flujo de Sucesos	Un auditor inicia la apertura de un nuevo caso de auditoría al ingresar los datos del auditado. El auditor crea la estructura del caso, parámetros y asigna revisores. Una vez creada la estructura y parámetros del caso finaliza la creación. Se guarda el caso. La instancia del caso permanece en ejecución.
Caminos Alternativos	En el paso 1, el auditor puede decidir abortar la creación del caso. En el paso 3, puede escogerse no guardar el caso, con lo cual se cancela la creación del caso.
Poscondiciones	Ha sido creada una estructura para almacenar datos del proceso de auditoría. El estado del caso permanece en ejecución.
Requisitos especiales	Un caso de estudio debe estar determinado por una fecha de cierre, que es estimada.

Para el caso de uso *Identificación de usuario*:

Figura 13. Casos de uso identificación de usuario y registro

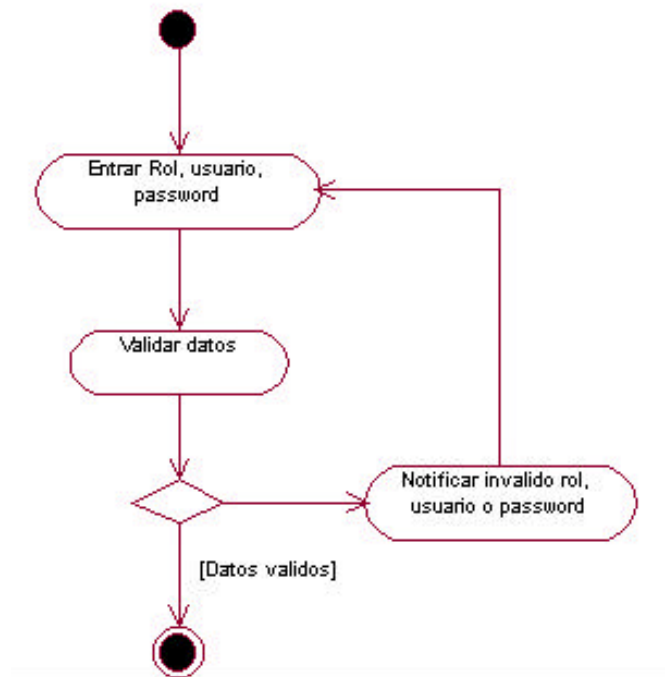
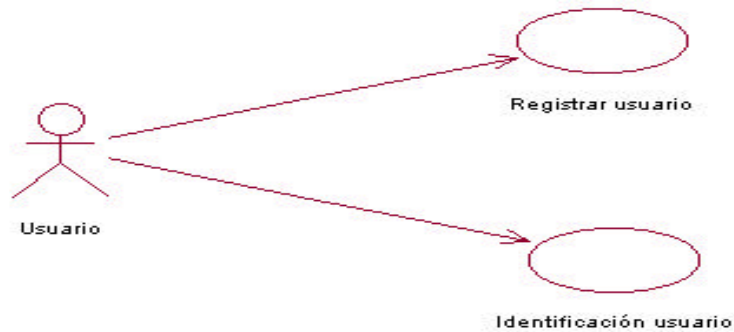


Figura 14. Diagrama de actividades para identificación del usuario

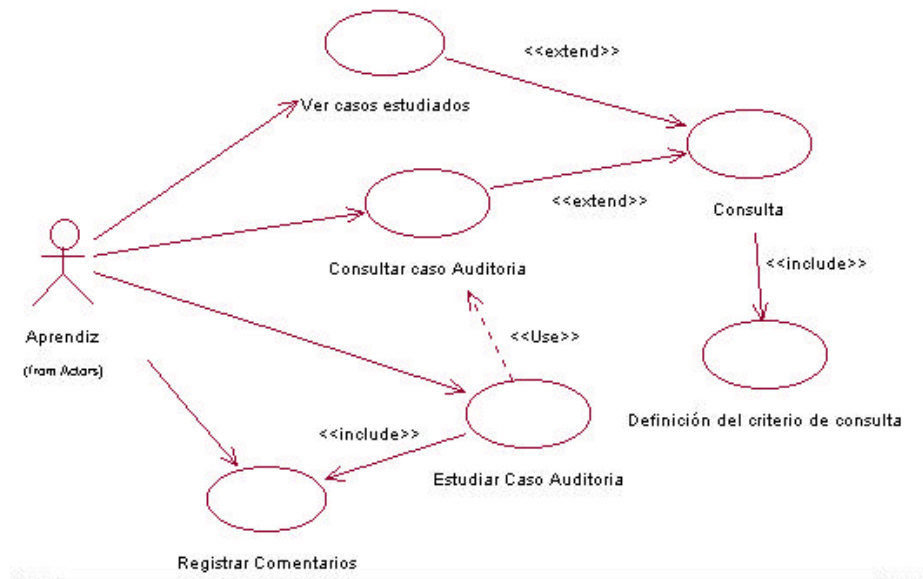


En la tabla 22 se muestran los detalles para el ingreso a la herramienta.

Tabla 22. Detalle del caso de uso – Identificación de usuario.

Caso de Uso: Identificación de Usuario	
Precondición	Para ingresar a la herramienta, el usuario debe estar registrado
Descripción o Flujo de Sucesos	<p>Ingresar rol, usuario, password: El usuario que desee hacer uso de la herramienta, se identifica en uno de los tres roles: aprendiz, revisor o auditor; proporcionando además el nombre de usuario y password.</p> <p>Validar datos: Una vez ingresado el rol, usuario y password serán validados para permitir su acceso.</p> <p>Permitir acceso: Al realizar la respectiva verificación de rol, usuario y password, el usuario habrá ingresado a la herramienta y podrá comenzar a realizar lo que esté dentro de su rol.</p> <p>La instancia del caso de uso finaliza.</p>
Caminos Alternativos	En el paso 3, si los datos no son validos regresa al paso 1.
Poscondiciones	<p>La instancia del caso de uso finaliza :</p> <p>Cuando los datos del usuario son validos, permitiéndole el acceso a la herramienta.</p>

Figura 15. Casos de uso Consulta y estudio de casos de auditoría



La figura 16 muestra un diagrama de estados para el caso de uso Estudiar caso Auditoría.

Figura 16. Flujo de sucesos para el caso de uso Estudiar caso auditoría

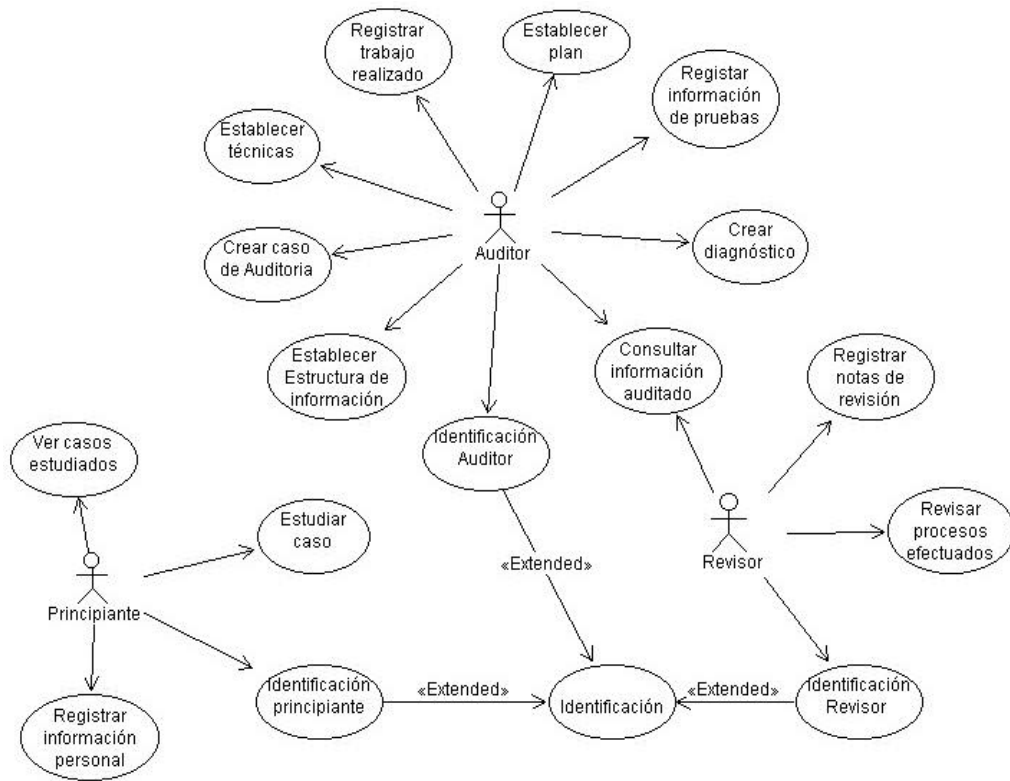


**Tabla 23. Detalle del caso de uso – Estudiar caso de auditoría.**

Caso de Uso: Estudiar caso de auditoría	
Precondición	El usuario debe identificarse como Aprendiz.
Descripción o Flujo de Sucesos	<p>Abrir casos de estudio: Un principiante ingresa a la herramienta con lo cual podrá abrir un caso de auditoría para acceder a la información en modo de lectura.</p> <p>Mostrar casos a estudiar: Una vez abierto el caso, se mostrará un listado de casos a estudiar.</p> <p>Seleccionar caso: Del listado de casos se selecciona un caso a estudiar este caso esta descrito mediante un título.</p> <p>Verificar estado de estudio: Al realizar la respectiva verificación del estado del caso y detectar que no ha sido revisado, se habrá ingresado a estudiar el caso</p> <p>Activar caso de estudio. Una vez ingresado al caso se activará para ver su estructura de información.</p> <p>Mostrar detalles del caso. Una vez activado el caso se podrá estudiar.</p> <p>Cambiar de estado. Al finalizar el estudio se cambia de estado el caso de estudio.</p> <p>La instancia del caso de uso finaliza.</p>
Caminos Alternativos	- En el paso 4, si el estado del caso es revisado se regresará al paso 2.
Poscondiciones	<p>La instancia del caso de uso finaliza:</p> <p>Cuando el aprendiz cambia el estado del caso de estudio a revisado.</p> <p>Si el aprendiz decide cancelar el estudio de caso.</p>

**5.3.1.8 Modelo de casos de Uso.** En esta fase se han seleccionado los casos de uso más representativos y otros adicionales, con el fin de proporcionar una entrada fundamental para el análisis y el diseño.

Figura 17. Modelo de casos de uso.



**5.3.1.9 Captura de Requisitos adicionales.** La tabla 24 contiene un listado de requisitos no funcionales que tienen un alto impacto para el desarrollo de la herramienta.

Tabla 24. Requisitos no funcionales

Tipo de Requisito	Descripción
Requisitos de Interfaz	Permite la interacción con los usuarios de forma fácil y útil.
Requisitos de Plataforma hardware	PC Mínimo Intel 486.
Restricciones de plataforma software	La herramienta debe ejecutarse sobre el sistema operativo Windows 98
Restricciones de Implementación	Para la Implementación se utiliza el lenguaje de programación Visual C++ Utilizar un modelo de base de datos relacional.
Otros requisitos adicionales	Diseñar y construir una base de datos relacional. Desarrollar una base de datos sobre Access y conectarla a la aplicación.

## 5.4 MODELADO DE ANALISIS Y DISEÑO

### 5.4.1 Flujo de Trabajo de Análisis.

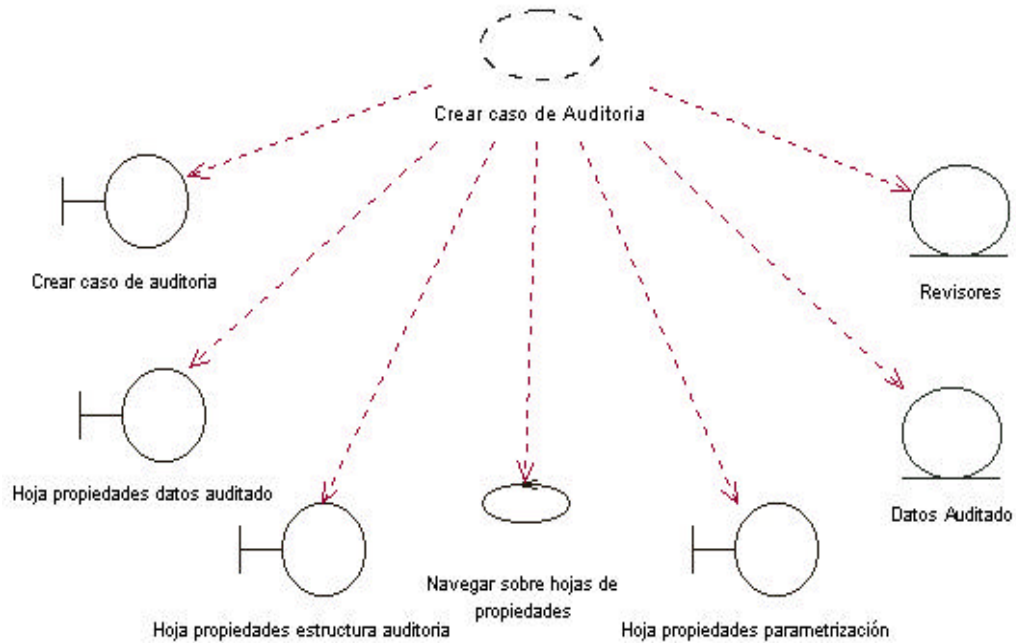
#### 5.4.1.1 Identificación de Clases del análisis

Figura 18. Realización del caso de uso crear caso de auditoría



Se han identificado cuatro clases de interfaz: Crear caso de auditoría, Hoja propiedades datos auditado, Hoja propiedades estructura auditoría, Hoja propiedades parametrización que permiten la apertura de un caso de auditoría, la solicitud de datos del auditado y la parametrización del caso. La clase control *navegar sobre hoja de propiedades* es la encargada de establecer los límites en los datos que se van a mostrar en las hojas de propiedades.

Figura 19. Clases de análisis para crear caso de auditoría



Las clases entidad *Datos Auditado* y *Revisores* se utilizan para representar los datos del auditado y los revisores que van a hacer parte del caso.

Figura 20. Realización del caso de uso identificación de usuario

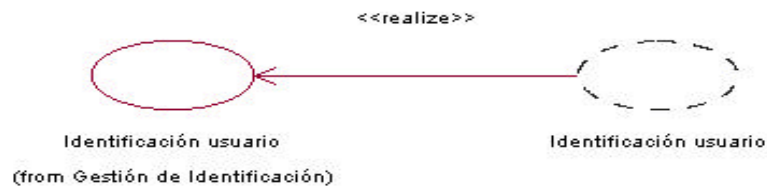
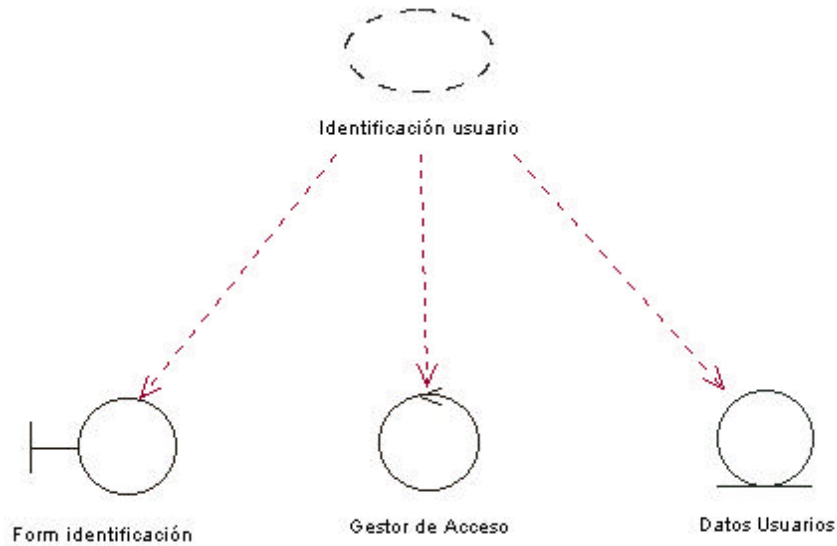


Figura 21. Clases del análisis para realizar identificación de usuario

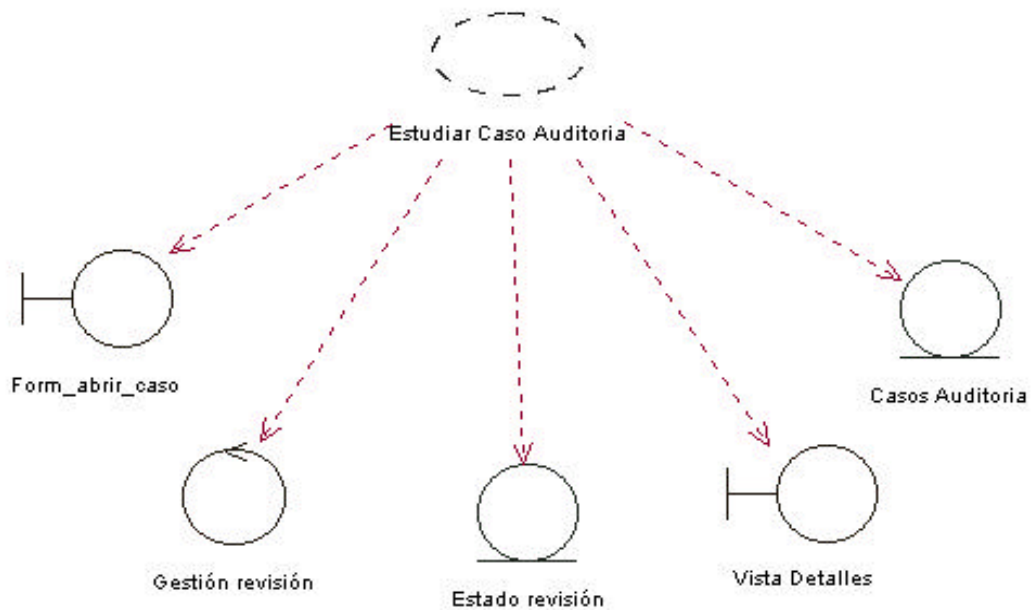


Para este caso de uso se han identificado una clase interfaz *Form identificación* que permite la identificación del usuario, una clase *Gestor de acceso* que permite la comprobación del usuario, una clase entidad *datos usuario* que se utiliza para representar los datos de los usuarios registrados.

Figura 22. Realización del caso de uso Estudiar caso de Auditoría



Figura 23. Clases de análisis para realizar Estudiar caso Auditoría

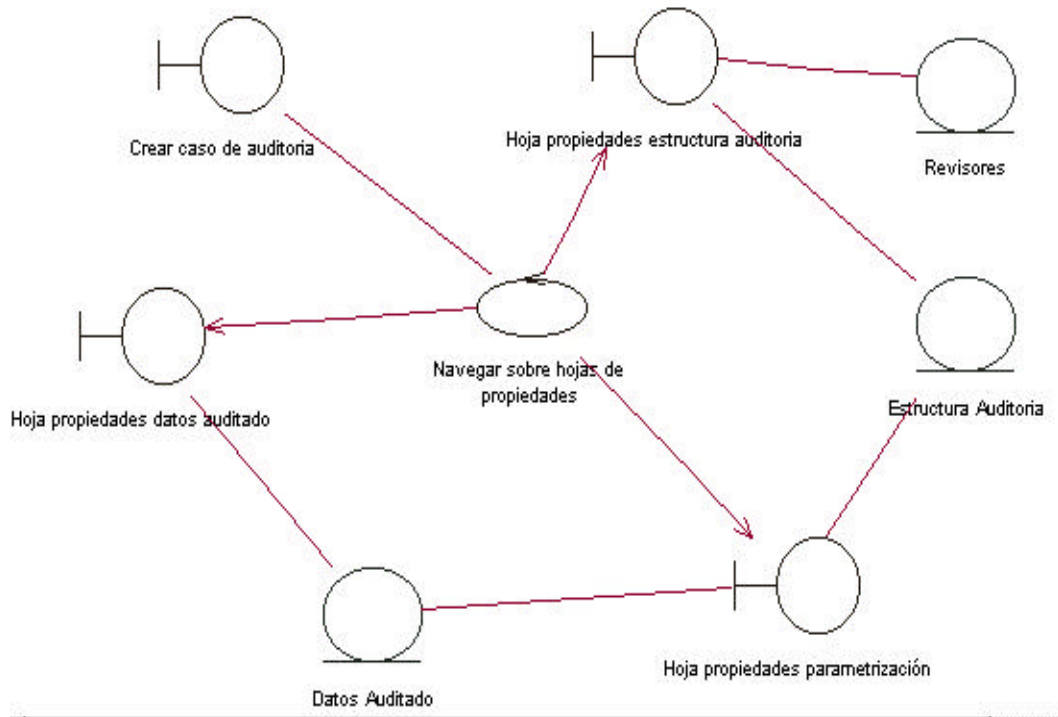


Para el caso de uso Estudiar caso Auditoría se han identificado la clase interfaz *form\_abrir\_caso* que permite la apertura de un caso para su estudio, La interfaz *vista detalles* que permite desplegar los detalles del caso abierto, la clase control *gestión de revisión* que permite el control de los estados de revisión del caso en estudio y finalmente, dos clases entidad *casos auditoría*, *estado revisión* que permiten representar los datos de caso de auditoría y revisión.

#### **5.4.1.2 Descripción de interacciones entre objetos del caso de Uso**

El auditor a través de la Interfaz *Crear caso de auditoría* solicita los datos necesarios para la apertura de un nuevo caso de estudio invocando los revisores que hacen parte del proceso y parametrizando estos datos con el fin de crear un caso de estudio.

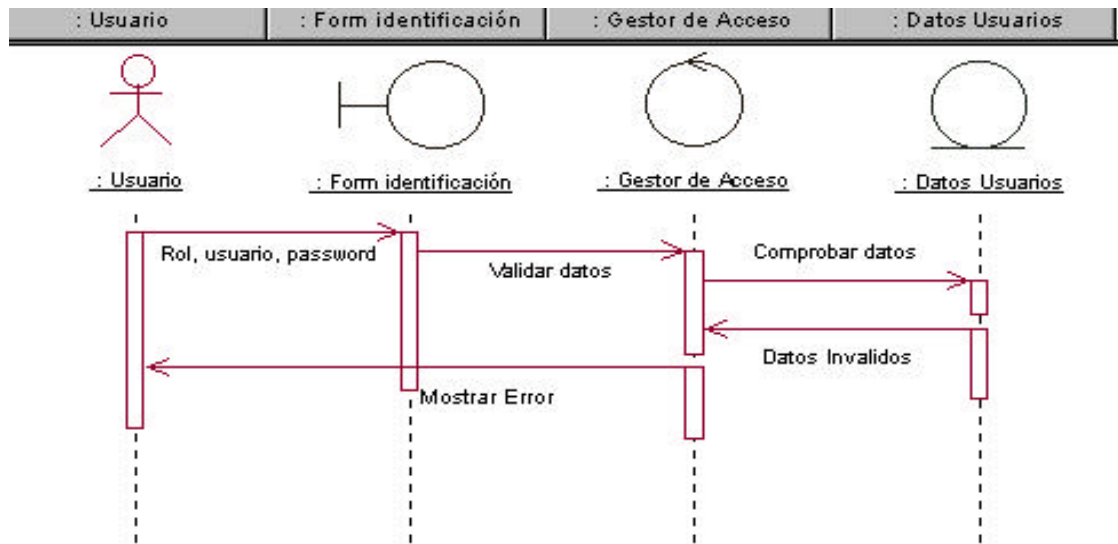
Figura 24. Diagrama de clases de una realización del caso de uso Crear Caso de estudio de auditoría



En este momento se ha creado un caso de estudio que permanecerá activo hasta que se le indique el cierre definitivo.

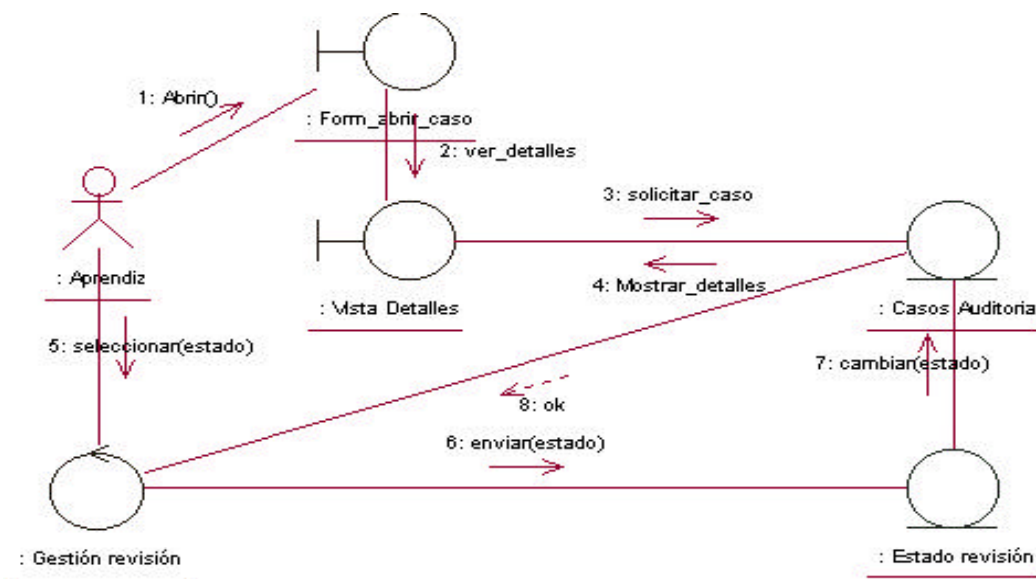
Para el caso de uso *identificación de usuario* se ha realizado un diagrama de actividad para el escenario en el cual, el usuario se identifica en forma errónea y la herramienta software responde mediante la visualización de un mensaje de error. Los detalles los puede observar en la figura 25.

Figura 25. Diagrama de actividad para el caso de uso identificación usuario invalido



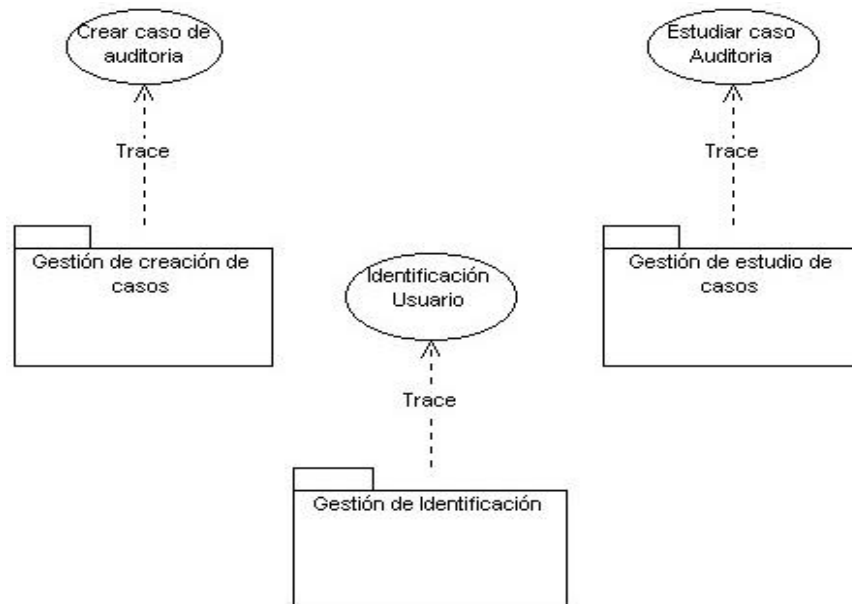
Para el caso de uso *Estudiar caso Auditoría* se mostrará un diagrama de colaboración.

Figura 26. Diagrama de colaboración para el caso de uso estudiar caso auditoría



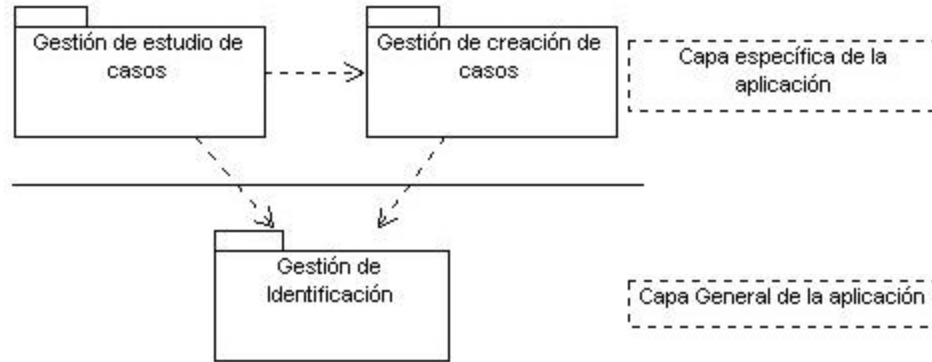
**5.4.1.3 Identificación de paquetes del análisis.** Se ha identificado el paquete Gestión de creación de casos de estudio que se ajusta a la necesidad del usuario Auditor de crear el caso de auditoría; el paquete Gestión de estudio de casos se ajusta a la necesidad del usuario aprendiz permitiéndole el estudio de casos creados por el auditor; el paquete Gestión identificación se adapta como requisito para la identificación de los diferentes usuarios de la herramienta.

Figura 27. Identificación de paquetes del análisis a partir de los casos de uso



**5.4.1.4 Dependencias entre paquetes del análisis y capas.** El paquete de Gestión de identificación contiene varias clases, como usuario, que son utilizadas por clases en otros paquetes. La figura 28 muestra como los paquetes Gestión de creación de casos y Gestión de estudio de casos dependen de Gestión de identificación.

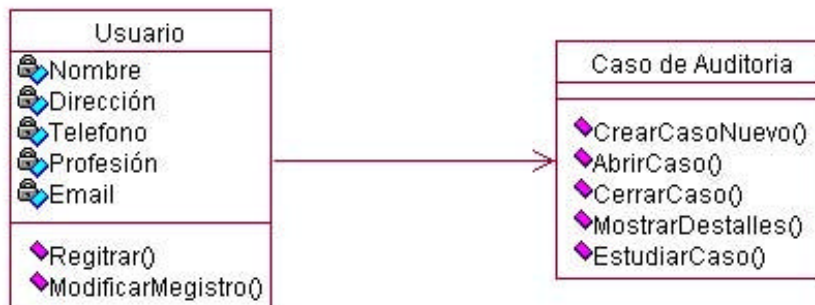
Figura 28. Dependencia entre capas y paquetes del análisis



#### 5.4.2 Flujo de trabajo de Diseño.

**5.4.2.1 Diseño de clases.** La figura 29 ilustra la clase de diseño Usuario y Caso auditoría tal como se elabora en el diseño.

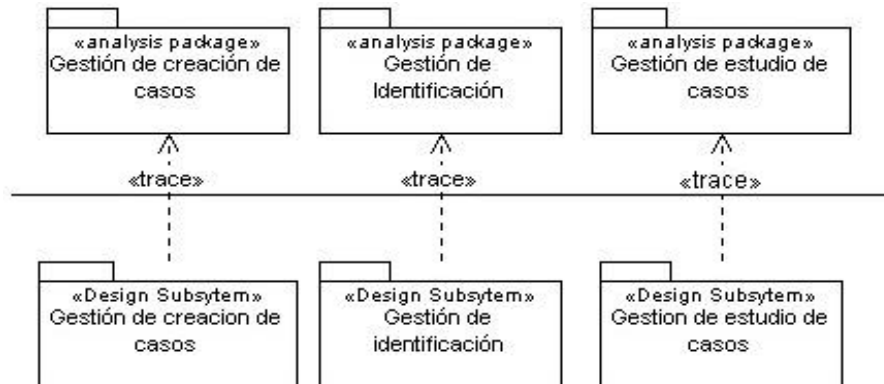
Figura 29. La clase de diseño usuario y caso de auditoría con sus atributos y operaciones



**5.4.2.2 Identificación y distribución de subsistemas.** Los paquetes Gestión de creación de casos, Gestión de estudio de casos y Gestión de identificación del

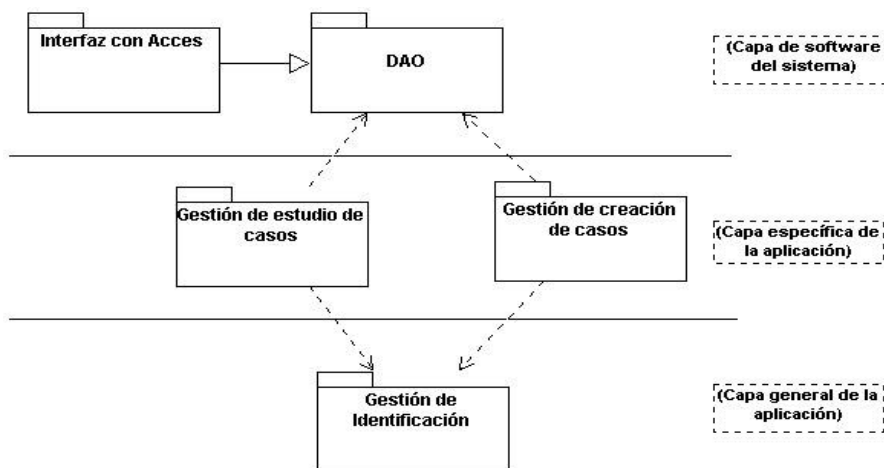
modelo de análisis se utilizan para identificar los correspondientes subsistemas del modelo de diseño.

Figura 30. Identificación de subsistemas a partir de paquetes del análisis existentes



**5.4.2.3 Definición de dependencias y capas entre subsistemas.** En la figura 31 se muestran los subsistemas y algunas de las dependencias iniciales para la herramienta software propuesta. Obsérvese que se ha tomado la capa de software del sistema para realizar conexión con la base de datos a través de DAO.

Figura 31. Dependencias y capas de algunos de los subsistemas de la herramienta



## **5.5 EVALUACION DE LA FASE DE INICIO**

Como resultado de la fase de inicio, se consigue la obtención de una arquitectura candidata para la herramienta software propuesta, esta arquitectura se ha esbozado con un análisis inicial del negocio, una lista de características de la herramienta software, la identificación y análisis de los casos de uso críticos, la identificación de algunos riesgos y por supuesto la delimitación de la situación objeto de estudio.

La conclusión más importante de esta fase es haber logrado un cierto grado de confianza, que permite el desarrollo de una herramienta software deseable para apoyar el aprendizaje del proceso de auditoría. La próxima fase se realizará en una iteración donde se debe abarcar la mayor parte de los requisitos pendientes.

## Capítulo 6

### FASE DE ELABORACION

Tras haber formulado la arquitectura inicial (Arquitectura candidata) en la fase de inicio, se entrará en detalles para abarcar el modelado del negocio, y los flujos de trabajo correspondientes a esta fase con el fin establecer una base de la arquitectura sólida.

Se tomarán en cuenta los resultados de la fase de inicio para recopilar la mayor parte de los requisitos que aún quedaron pendientes, formulando requisitos funcionales como casos de uso, recopilando y controlando los riesgos críticos que queden, hasta el punto de estimar su impacto en el análisis del negocio.

Al concluir esta fase, se evaluarán los resultados obtenidos para comprobar que, efectivamente se van a cumplir los objetivos propuestos para el actual proyecto y determinar si la línea base representa una arquitectura capaz de prevenir fallos en la comprensión y organización de los elementos más significativos de la herramienta software propuesta.

#### 6.1 PLANIFICACIÓN DE LA FASE DE ELABORACIÓN

Para el desarrollo de esta fase se ha establecido una iteración, en la cual se realiza la descripción del negocio, la arquitectura de algunos modelos para esbozar la arquitectura: casos de uso, análisis, diseño e implementación hasta completar la línea base de la arquitectura. Al concluir esta iteración se evaluará lo que se ha logrado frente a los criterios establecidos en la fase de inicio.

Así, al final de esta fase se habrá abordado alrededor del 80 por ciento de los casos de uso, los riesgos que interfieran en el desarrollo del actual proyecto y la información necesaria para planificar la fase de construcción.

## 6.2 MODELO DEL NEGOCIO

El modelado del negocio comienza identificando los objetivos principales de este. Los objetivos principales del negocio son los siguientes:

- 1 Apoyar el proceso de aprendizaje de la auditoría de sistemas apoyándose en la metodología didáctica del estudio de casos.
- 2 Permitir la recopilación, organización de Casos de auditoría por parte de especialistas en el área (Auditor) para su posterior estudio.
- 3 Utilizar una metodología que permita la comprensión y descripción de los procesos realizados durante el desarrollo de la auditoría.
- 4 Crear una herramienta software que apoye el proceso de aprendizaje de la auditoría basada en las metodologías anteriores.

Básicamente, estos objetivos reúnen un punto de vista clave para establecer y profundizar sobre el tema de la auditoría en un ambiente formativo.

### 6.2.1 Modelo de Casos de uso del Negocio.

Figura 32. Caso de Uso estudiar metodologías.

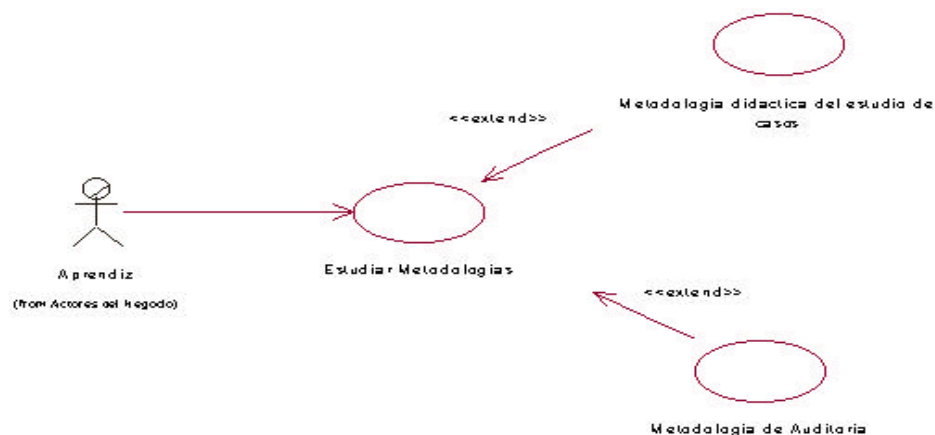


Figura 33. Desarrollar caso de estudio

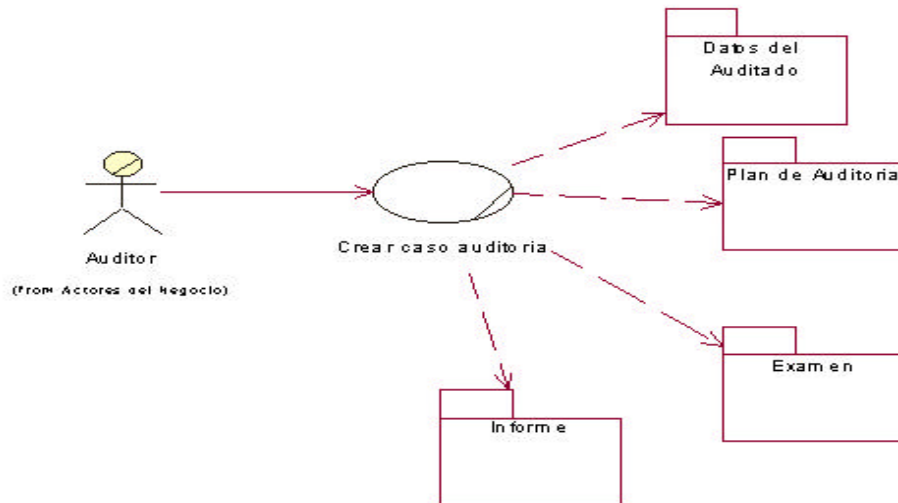
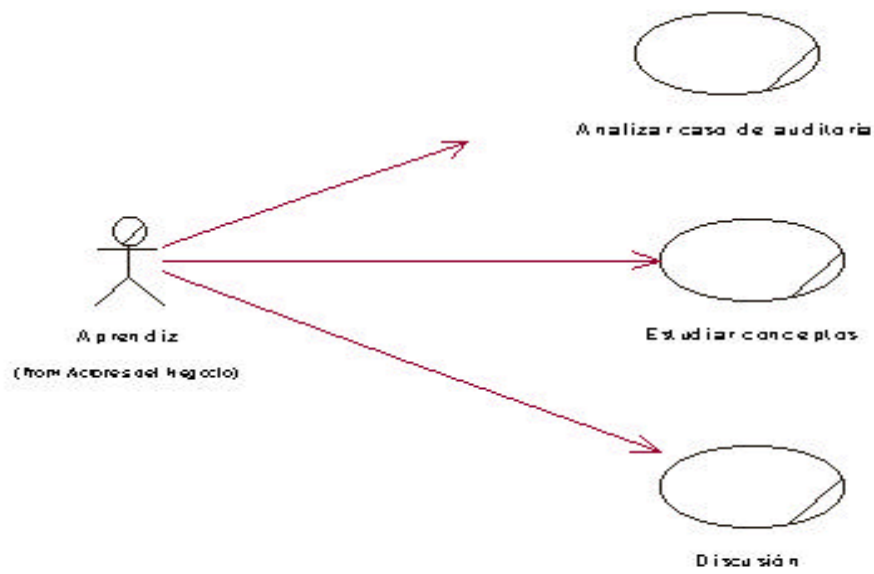


Figura 34. Caso de uso Estudiar caso de auditoría



## 6.2.2 Descripción de casos de uso del negocio.

Para la descripción de los procesos del negocio se utilizará la plantilla descrita en el anexo A.

*Tabla 25. Descripción del proceso de negocio – Aprender metodologías.*

Proceso de Negocio: Aprender Metodologías	
Objetivo	Que el aprendiz comprenda las metodologías usadas para apoyar el proceso de aprendizaje (Metodología didáctica del estudio de casos, Metodología de auditoría).
Descripción	El Tutor explica al aprendiz los siguientes aspectos: -Cuál es el propósito la Metodología didáctica del estudio de casos. - La metodología usada para el desarrollo de auditorías.
Prioridad	Importante
Riesgos	Rutinario
Posibilidades	Este proceso puede ser soportado a través de la creación de manuales y archivos de ayuda.
Tiempo de Ejecución	...

*Tabla 26. Descripción del proceso de negocio – Desarrollar caso de estudio.*

Proceso de Negocio: Desarrollar caso de estudio	
Objetivo	Recopilar y organizar la información característica de los Casos de auditoría por parte de especialistas en el área (Auditor) para su posterior estudio.
Descripción	El auditor a través de los flujos de trabajo organiza el caso de estudio. El auditor a través de la metodología recopila la mayor parte de la información resultado del proceso de auditoría. El revisor adiciona conclusiones a sus observaciones y verifica procesos efectuados
Prioridad	Crítico
Riesgos	Rutinario
Posibilidades	Este proceso puede ser soportado a través del registro de datos en la base de datos de la herramienta software propuesta.
Tiempo de Ejecución	...

*Tabla 27. Descripción del proceso de negocio – Estudiar caso de estudio.*

Proceso de Negocio: Estudiar caso de estudio	
Objetivo	Analizar el caso de auditoría Lograr una conceptualización con los detalles presentados en el caso. Utilizar la metodología didáctica del estudio de casos para lograr los objetivos que se persiguen con ella.
Descripción	El aprendiz toma el caso proporcionado por el auditor con el fin de realizar un análisis y comprensión de los procesos efectuados en el desarrollo de una auditoría.
Prioridad	Importante
Riesgos	Rutinario
Posibilidades	Este proceso puede ser soportado a través de la visualización de los flujos de trabajo y los datos almacenados en la herramienta software propuesta.
Tiempo de Ejecución	...

*Tabla 28. Descripción del proceso de negocio – Desarrollar discusión del caso.*

Proceso de Negocio: Desarrollar discusión del caso	
Objetivo	Determinar con exactitud el grado de comprensión y la capacidad de diagnóstico de la situación. Observar la capacidad de síntesis sobre la situación frente a otros aprendices en un ambiente de participación.
Descripción	El aprendiz analiza la información suministrada en el caso y se analiza el porqué del diagnóstico dado.
Prioridad	Importante
Riesgos	Rutinario
Posibilidades	Se soporta mediante el registro de anotaciones al caso hechas por el aprendiz
Tiempo de Ejecución	...

## 6.3 MODELADO DE REQUISITOS

### 6.3.1 Flujo de trabajo de Requisitos.



### 6.3.1.2 Casos de uso detallados.

Figura 36. Casos de uso para establecer plan de auditoría

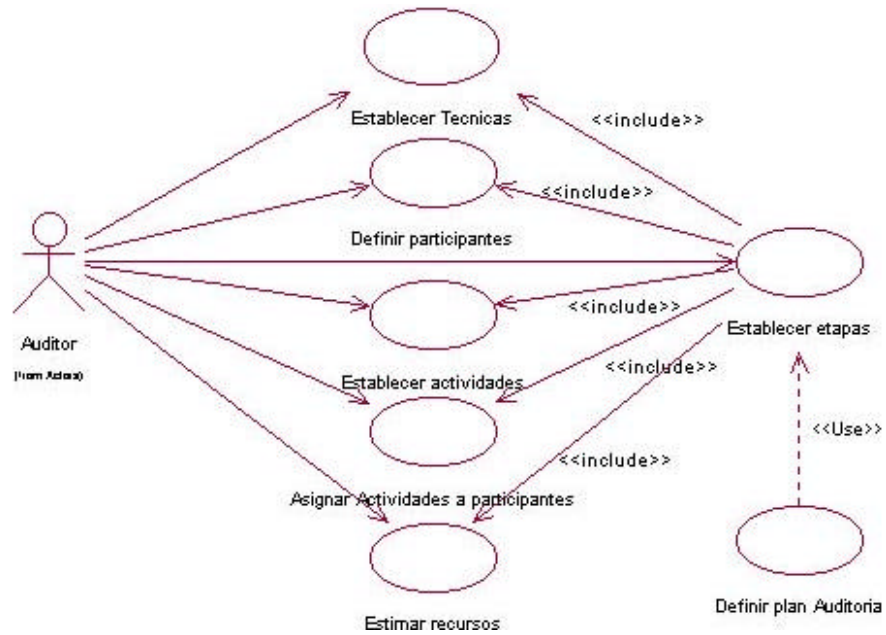


Figura 37. Diagrama de actividades para establecer plan de auditoría

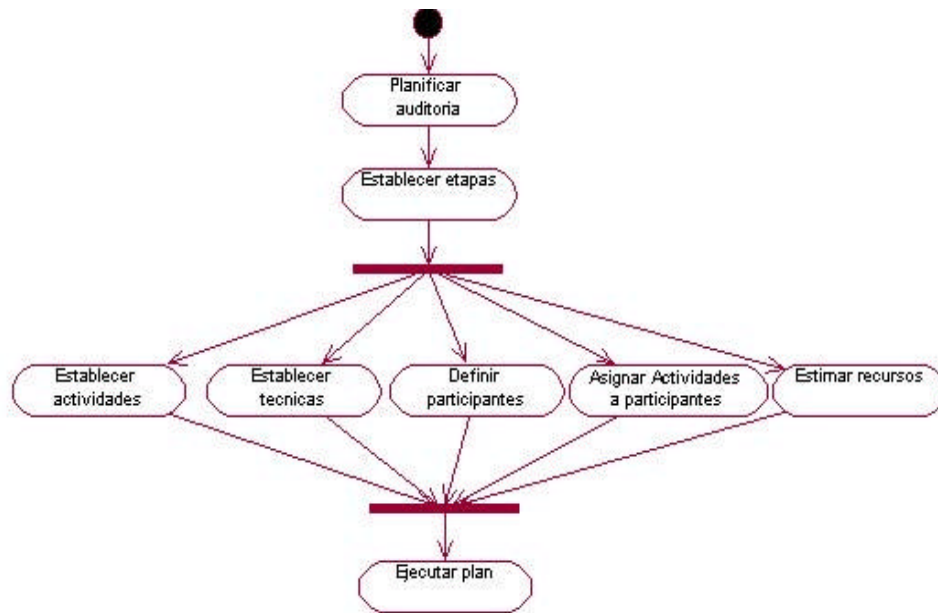


Tabla 29. Detalle del caso de uso – Establecer plan de auditoría.

Caso de uso Establecer plan de auditoría	
Precondición	Para utilizar este caso se debe haber creado un caso de estudio de auditoría
Descripción o Flujo de Sucesos	<p>Establecer etapas: El auditor establece las etapas que se van a seguir en el proceso de auditoría según la metodología adoptada.</p> <p>Solicitar actividades: Por cada etapa definida se establecen las actividades correspondientes.</p> <p>Solicitar técnicas: Una vez definidas las actividades en cada etapa se establecen las técnicas necesarias para realizar la actividad.</p> <p>Definir participantes. Se establecen el personal involucrado en el desarrollo de la auditoría.</p> <p>Estimar recursos. Para cada etapa definida se estiman los recursos necesarios.</p> <p>Asignar actividades. Se establecen las actividades para cada participante.</p> <p>La instancia del caso de uso finaliza.</p>
Camino Alternativos	
Poscondiciones	La instancia del caso de uso finaliza: Cuando se han definido las etapas, actividades, técnicas y personal participante del plan de auditoría.

Figura 38. Casos de uso para realizar examen auditado

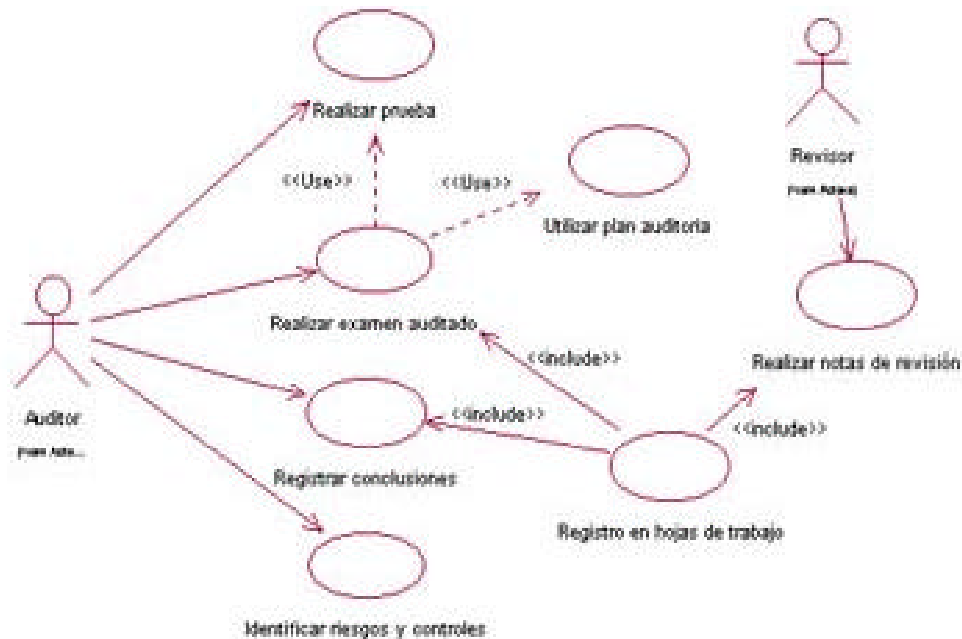


Figura 39. Diagrama de actividades para el caso de uso realizar examen

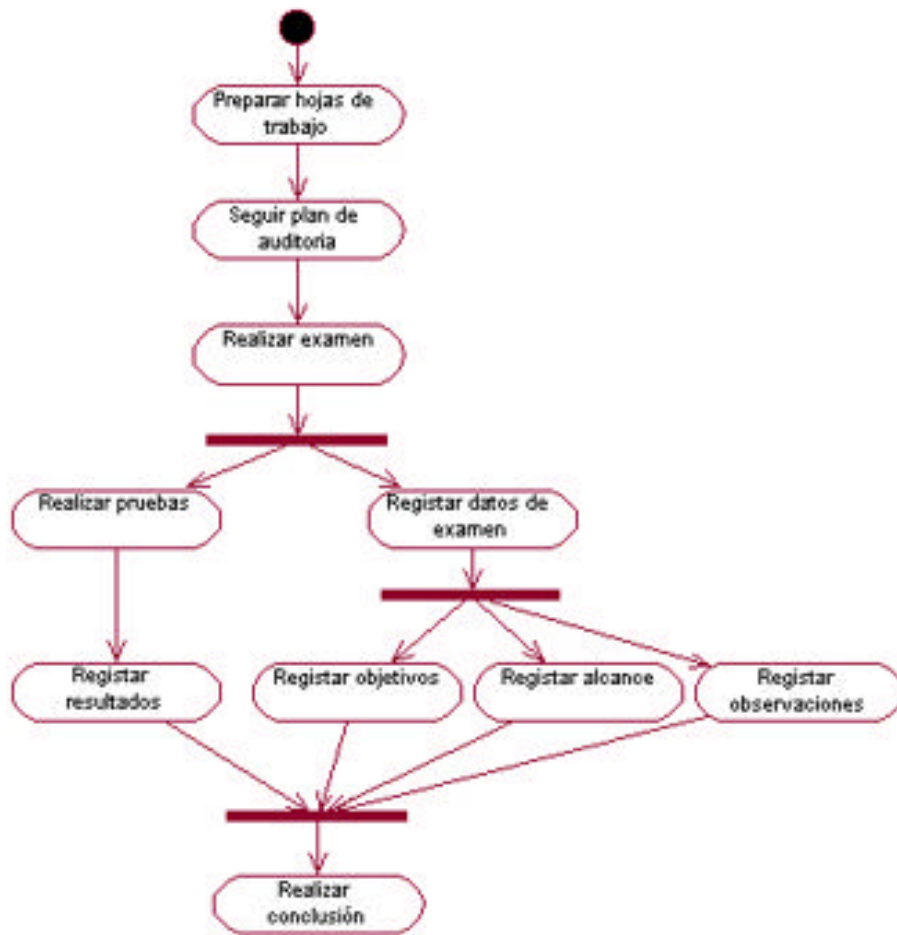


Tabla 30. Descripción del caso de uso – Examen auditado.

Caso de uso realizar examen auditado	
Precondición	Para la realización del examen se requiere haber definido el plan de auditoría
Descripción o Flujo de Sucesos	<p>Se debe:</p> <p>Preparar hojas de trabajo: En las cuales se indica el nombre de la hoja de trabajo, fecha de inicio/terminación, identificación de la fuente de información.</p> <p>Seguir plan de auditoría: Este paso implica el seguimiento de las actividades y técnicas para ejecutar el trabajo.</p> <p>Realizar examen: Se debe registrar la labor a realizar, observaciones, conclusiones, pruebas e identificación de riesgos y controles.</p>

	Registrar conclusiones: Finalmente se elabora conclusiones con base al trabajo realizado.
Caminos Alternativos	
Poscondiciones	La instancia del caso de uso finaliza cuando se termina de desarrollar la(s) actividad(es) del plan de auditoría en cada hoja de trabajo.

Figura 40. Caso de uso crear diagnóstico y realizar informe.

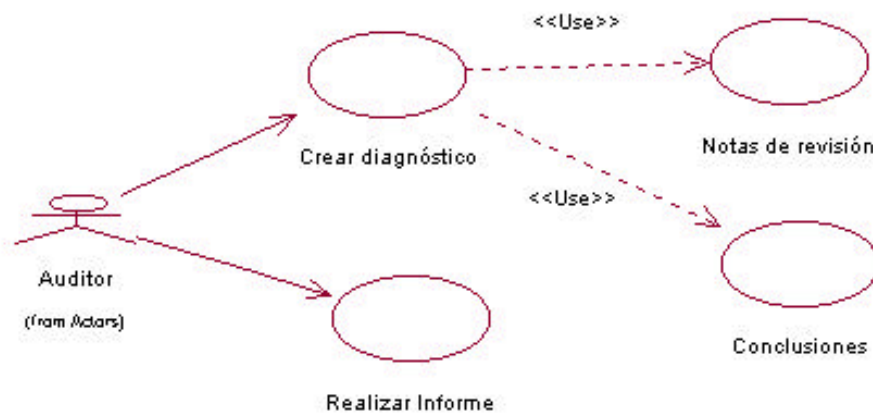


Tabla 31. Descripción del caso de uso – crear diagnóstico.

Caso de uso crear diagnóstico	
Precondición	Para realizar este caso se debe haber terminado el examen al auditado
Descripción o Flujo de Sucesos	Para crear diagnóstico el auditor hace uso de las conclusiones y notas de revisión plasmadas en las hojas de trabajo.  Además, estos datos permiten realizar un informe de la labor efectuada durante todo el proceso.
Caminos Alternativos	
Poscondiciones	La instancia del caso de uso finaliza cuando se finaliza el informe y diagnóstico del auditado. Procediendo a finalizar el caso.

**6.3.1.3 Identificación de riesgos críticos.** En este punto se identificarán algunos riesgos que pueden afectar el desarrollo del proyecto.

*Tabla 32. Identificación de riesgos para la fase de elaboración.*

Descripción	Prioridad	Impacto	Responsable	Contingencia
Una descripción incompleta de los requerimientos	Significativo	Retardo en la fase de construcción	Autor	Detallar y documentar con detenimiento
Identificación incorrecta de herramientas de desarrollo	Critico	Detenimiento de la fase de construcción.	Director y Autor	Estudio y pruebas de las herramientas de desarrollo más adecuadas
Estimación errónea del tiempo de desarrollo del proyecto	Rutinario	Retardo en el desarrollo del proyecto	Director y Autor	Establecer puntos de evaluación periódica

## 6.4 MODELO DE ANALISIS Y DISEÑO

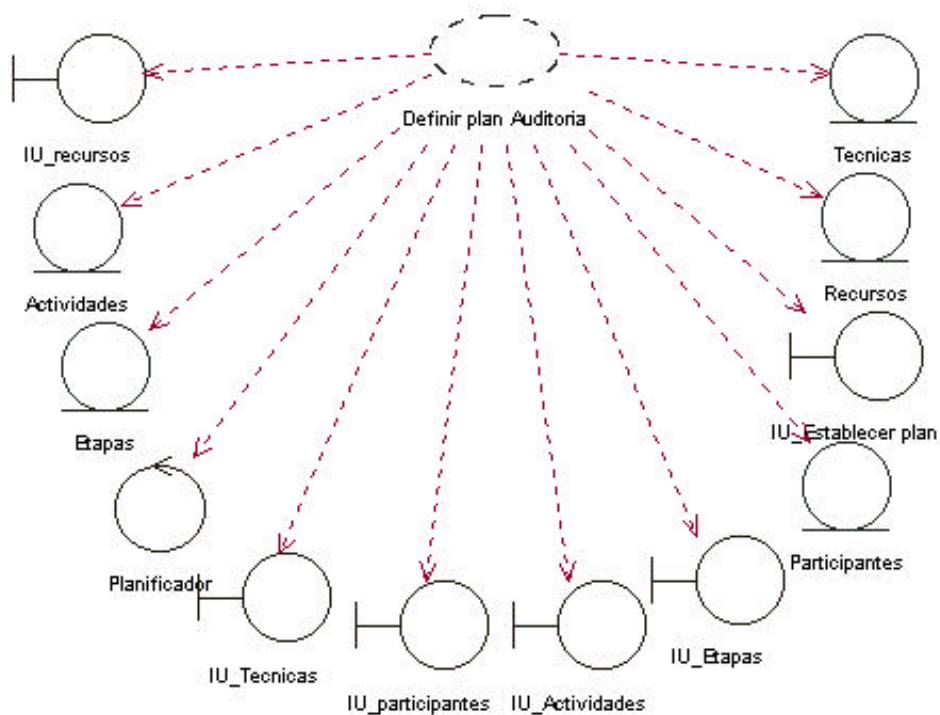
### 6.4.1 Flujo de trabajo de análisis.

#### 6.4.1.1 Identificación de clases de análisis

*Figura 41. Realización de caso de uso Definir plan de auditoría.*



Figura 42. Clases del análisis para Definir plan de auditoría.



Para el caso de uso establecer plan de auditoría se han identificado las clases de interfaz: *IU\_etapas* que permite establecer las etapas que se van a realizar en el plan de auditoría, *IU\_actividades* que permite establecer las actividades que se van a desarrollar por cada etapa, *IU\_técnicas* que permite establecer las técnicas necesarias para la realización de cada actividad, *IU\_participantes* en donde se definen los participantes en procesos de desarrollo del plan, *IU\_recursos* que permite definir los recursos empleados para el desarrollo del plan. Se han identificado las clases de entidad: *técnicas*, *Etapas*, *actividades*, *participantes*, *recursos* que modelan los datos que intervienen en el desarrollo del plan de auditoría. Además se han identificado las clases de control *planificador* que coordina los objetos mencionados.

Figura 43. Realización del caso de uso realizar examen auditado.

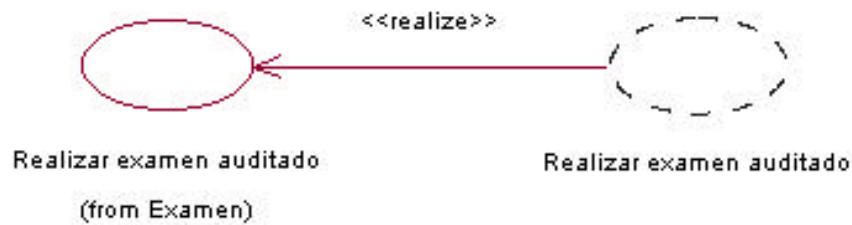
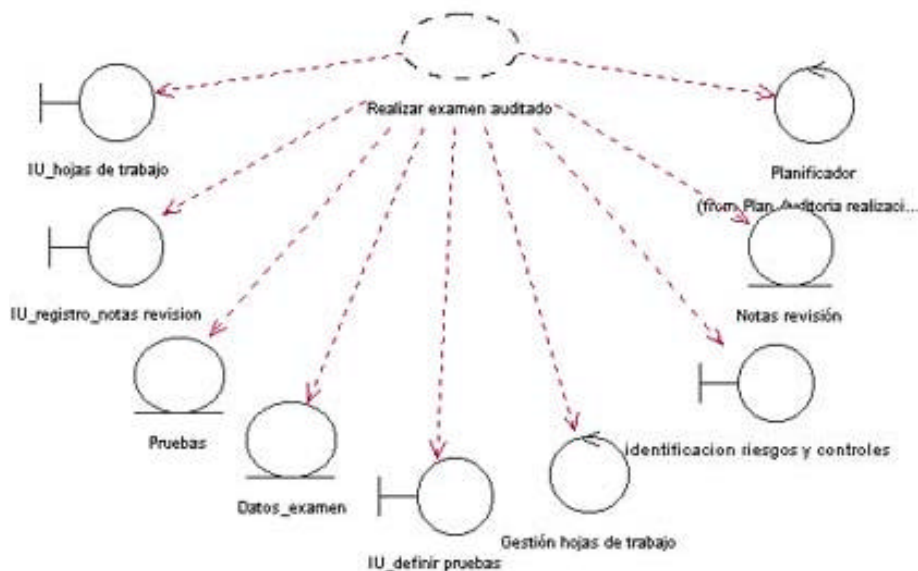


Figura 44. Clases del análisis para realizar examen auditado.

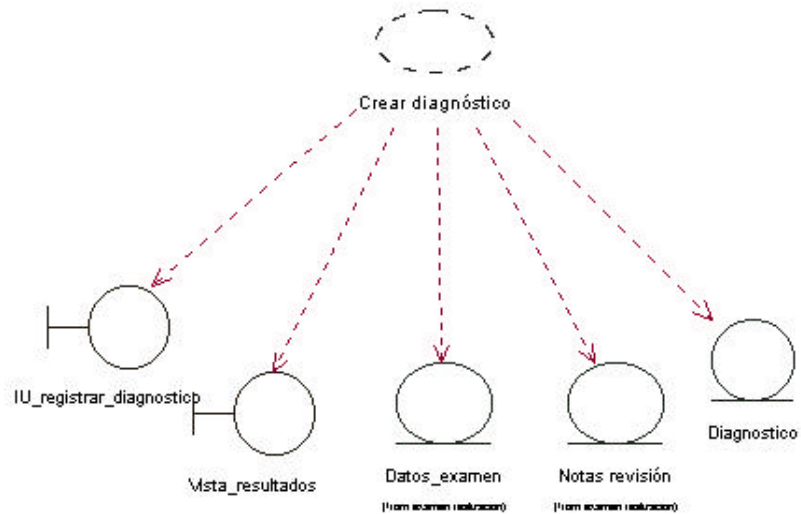


Para la realización del caso de uso *Realizar examen auditado* se han identificado la clase de interfaz *IU\_hojas de trabajo* que permite almacenar los datos del trabajo realizado, *IU\_registro\_notas revision* que permite gestionar las notas de revisión del revisor, *IU\_definir pruebas* permite diligenciar las pruebas a realizar y los resultados obtenidos, finalmente la interfaz para identificar riegos y controles.. Se identificaron las clases entidad *Datos examen*, *Notas revisión*, *pruebas*, que modelan los datos recopilados en el examen. Se identificó la clase de control *gestión hojas de trabajo* que coordina la creación de las hojas de trabajo y la clase de control *planificador* que permite organizar el trabajo de acuerdo al plan desarrollado.

Figura 45. Realización del caso de uso crear diagnóstico.



Figura 46. Clases del análisis para realizar crear diagnóstico .



Para la realización del caso de uso *crear diagnóstico* se ha identificado una clase interfaz *IU\_registrar\_diagnostico* que permite el registro del diagnóstico efectuado por el auditor, la clase interfaz *vista\_resultados* que permite ver los resultados del trabajo efectuado basándose en las clases entidad *datos\_exámen* y *notas revision* que modelan los datos efectuados en el examen.



Figura 48. Diagrama de colaboración para realizar examen.

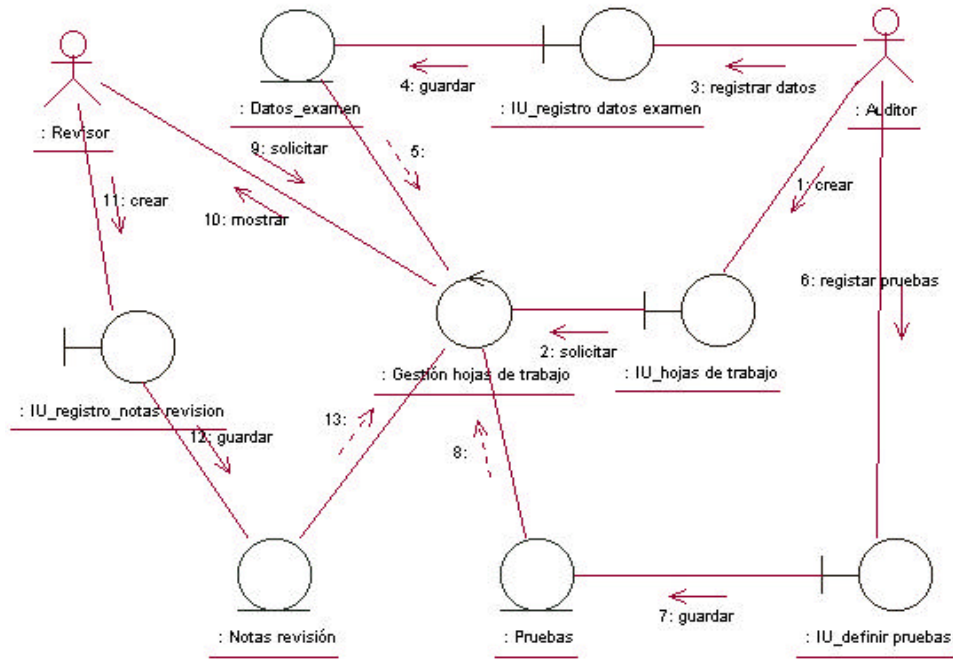
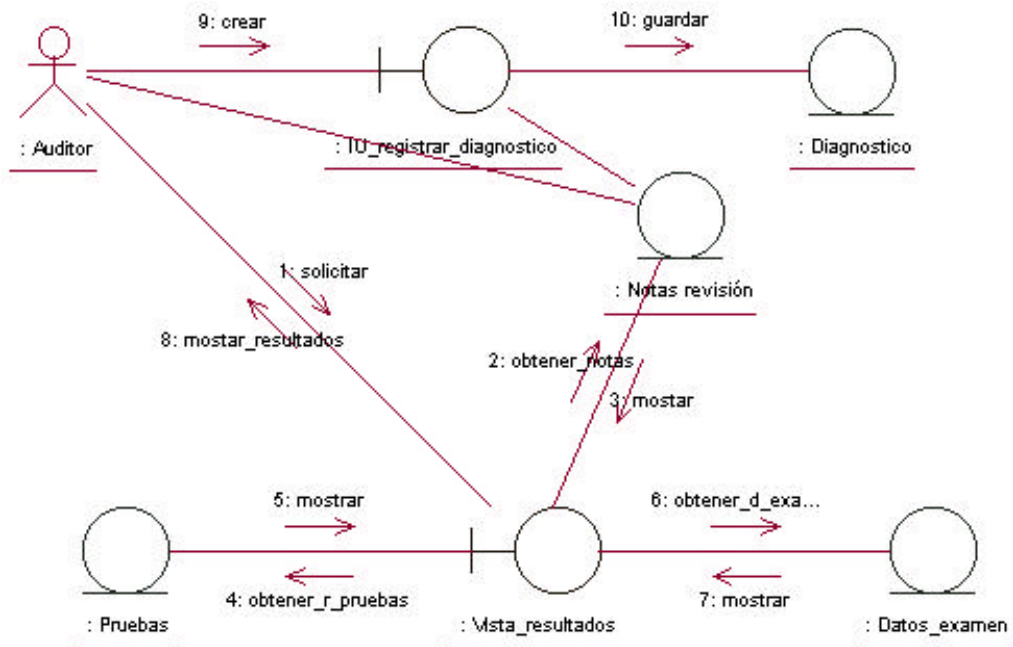


Figura 49. Diagrama de colaboración para crear diagnóstico.



### 6.4.1.3 Identificación de paquetes del análisis

Figura 50. Identificación de paquetes del análisis a partir de los casos de uso

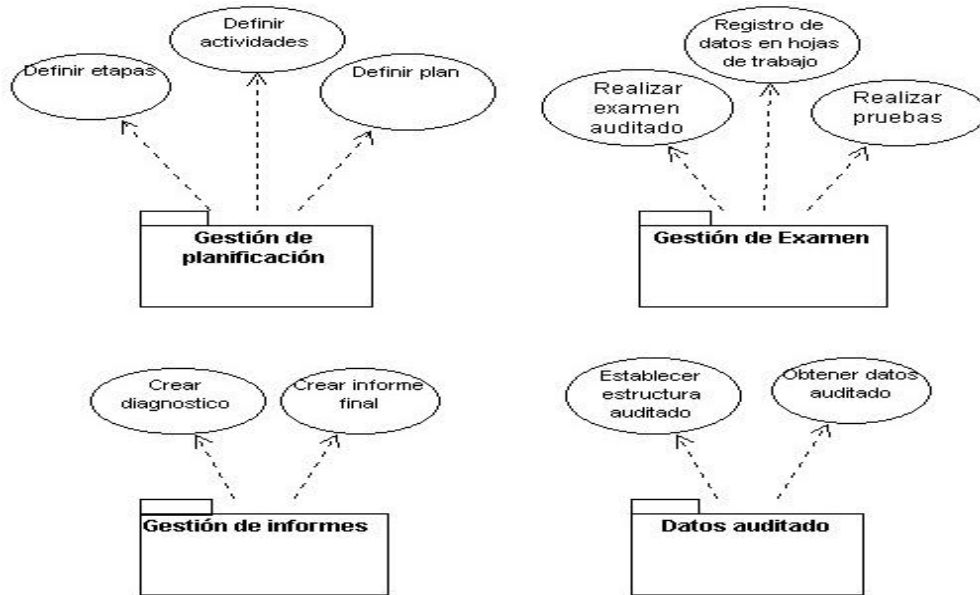
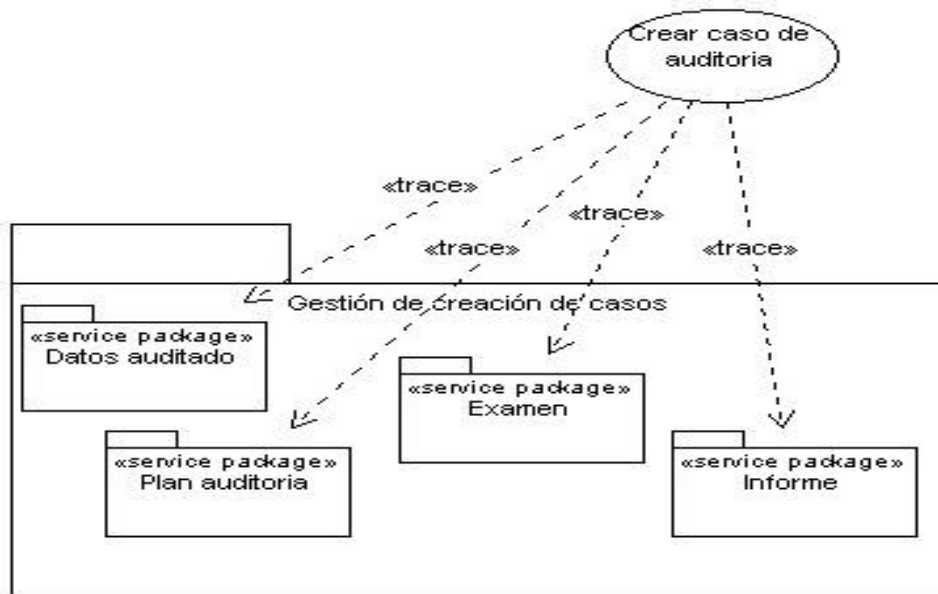


Figura 51. Los paquetes de servicio, datos auditado, plan auditoría, examen, informe, ubicados dentro del paquete Gestión de creación de casos



Cada uno de los paquetes identificados durante esta fase y la fase de inicio serán considerados en la fase de construcción permitiendo una mayor organización para la implementación de los casos de uso.

## **6.4.2 Flujo de trabajo de diseño.**

### **6.4.2.1 Diseño de la interfaz.**

Para el desarrollo de la interfaz se siguieron los siguientes lineamientos:

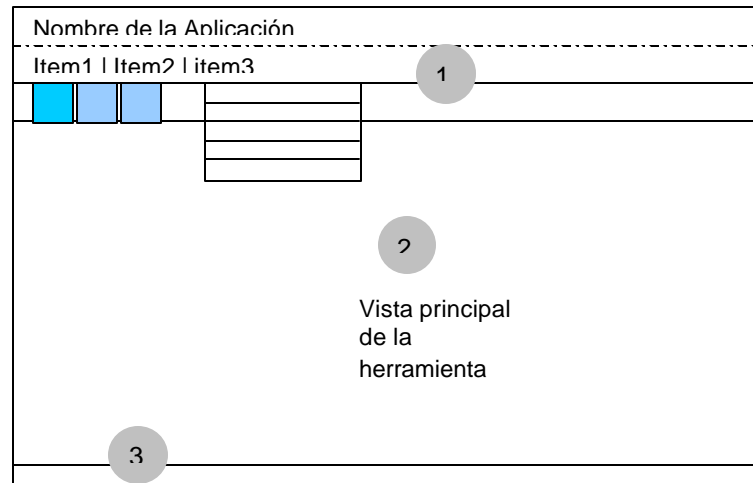
- **Coherencia:** Todas las pantallas de la aplicación deben ser coherentes con el contenido que se maneje. Ante acciones comunes del usuario se debe responder de manera adecuada presentando mensajes de diálogos claros y coherentes en idioma español.
- **Eficiencia y sencillez de uso:** La tarea es construir una interfaz amigable y fácil de usar siguiendo los lineamientos del usuario, por eso se debe tener en cuenta el orden y la forma lógica de las opciones (menús, submenús) para el correcto funcionamiento de la aplicación.
- **Navegación entre pantallas:** La navegación entre pantallas se debe realizar a través de vistas y diálogos modales evitando al máximo el sobre poblamiento de pantallas y la desviación de la atención del usuario hacia otros lugares.

Utilizando la biblioteca MFC de visual C++ se puede construir una aplicación SDI (Single Document Interface) en la cual se puede tener una ventana marco con su vista, para presentar el documento que se tiene abierto, y mantenerlo único para cada ejemplar de la aplicación.

La descripción de la interfaz se muestra en la figura 52 donde se identifican tres partes o secciones: En la sección superior (1) se encuentra la barra de menús, barra de título, barra de herramientas las cuales permiten un acceso a las opciones según el rol desempeñado. En la sección central (2) se encuentra la vista principal de la

herramienta donde se muestran los informes y generalmente es la vista contenedora de varios cuadros

Figura 52. Diseño general de la interfaz de la herramienta.

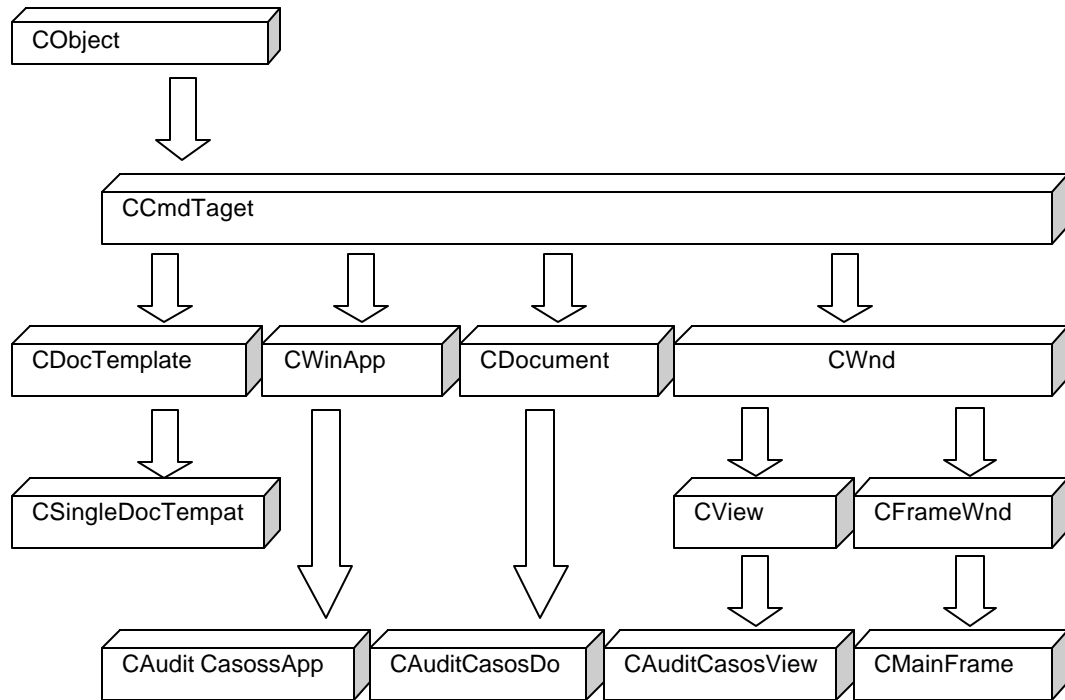


de diálogos; por último en la sección inferior (3) se muestra la barra de estado que permite mostrar información de ayuda sobre opciones e indicadores de estado.

La estructura de clases de la aplicación con interfaz de un solo documento se basa en cuatro clases que maneja la estructura de la aplicación. La herencia de estas clases se muestra en la figura 53.

La clase CAuditCasosApp, derivada de CWinApp que gestiona la iniciación de la aplicación, y es responsable de mantener la asociación entre las clases de documento, vista y marco. También recibe mensajes de Windows y los despacha a la ventana blanco correspondiente. Según se ha indicado anteriormente, con una aplicación de interfaz de un solo documento solo se llegará a crear un único ejemplar de la clase CAuditCasosDoc derivada de CDocument.

Figura 53. Clases que maneja la aplicación y generadas por el AppWizard.



El papel de la clase CAuditCasosDoc es el de contenedor de los datos de la aplicación. La clase CAuditCasosView es responsable de crear la representación visual de los datos del documento para el usuario y también debe capacitar al usuario para interactuar con los datos. El papel de la clase CMainFrame es proporcionar una ventana para el uso de la aplicación. Esta derivada de CFrameWnd, que es el envoltorio de una ventana sencilla que también maneja la creación, iniciación y destrucción de barras de herramientas y la barra de estado.

**6.4.2.2 Diseño de la Base de datos.** La herramienta software que se propone tiene objetos persistentes, lo cual implica el uso de tecnologías de bases de datos. Se empleará una base de datos relacional<sup>13</sup> ya que cada una de las clases del

<sup>13</sup> Véase Apéndice B. Diseño de bases de datos relacionales - una extensión informal de UML.

análisis tienen una estrecha relación con cada una de las entidades del modelo entidad-relación. Además, Visual C++ proporciona varias formas de trabajo con bases de datos y utilizando la biblioteca de clases MFC, se recurrirá a las clases DAO (Data Acces Objects – Objetos de acceso a datos).

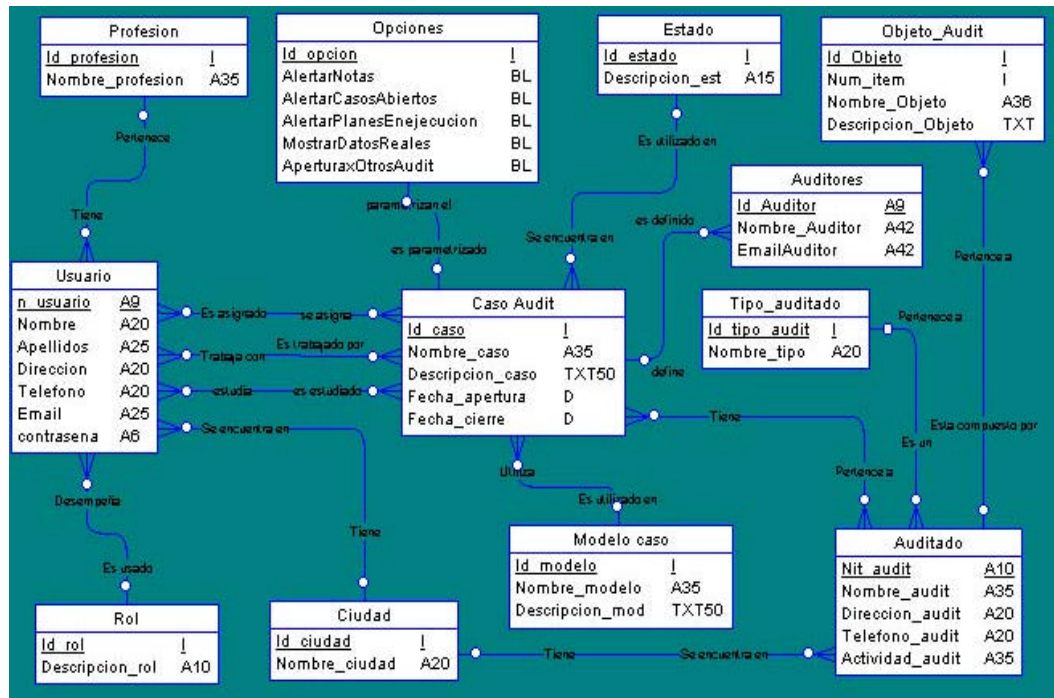
Para mostrar el modelo entidad-relación se ha recurrido a utilizar submodelos con el fin de abarcar los detalles necesarios en su comprensión.

En la figura 54 se muestra un submodelo con las entidades que participan en la creación de un caso de auditoría; además en la tabla 33 se presenta un resumen general de las entidades presentes en el diagrama.

*Tabla 33. Entidades presentes en el submodelo entidad – relación que participan en la creación de un caso de auditoría*

TABLA	INFORMACION
Usuario	Registro de usuarios de la herramienta.
Rol	Registro de los perfiles del usuario
Profesión	Registro de profesiones del usuario.
Ciudad	Registro de ciudades.
Auditado	Registro de con detalles del auditado.
Tipo_Auditado	Registro del rasgo característico del auditado.
Objeto_Audit	Registro de la estructura interna (elementos) del auditado
Caso_Audit	Registro de casos de estudio.
Modelo_caso	Registro característico del caso.
Opciones	Registro con opciones del caso.
Estado	Registro de posibles estados del caso.
Auditores	Registro de Auditores participantes en el caso.

Figura 54. Submodelo entidad relación – Creación caso de Auditoría



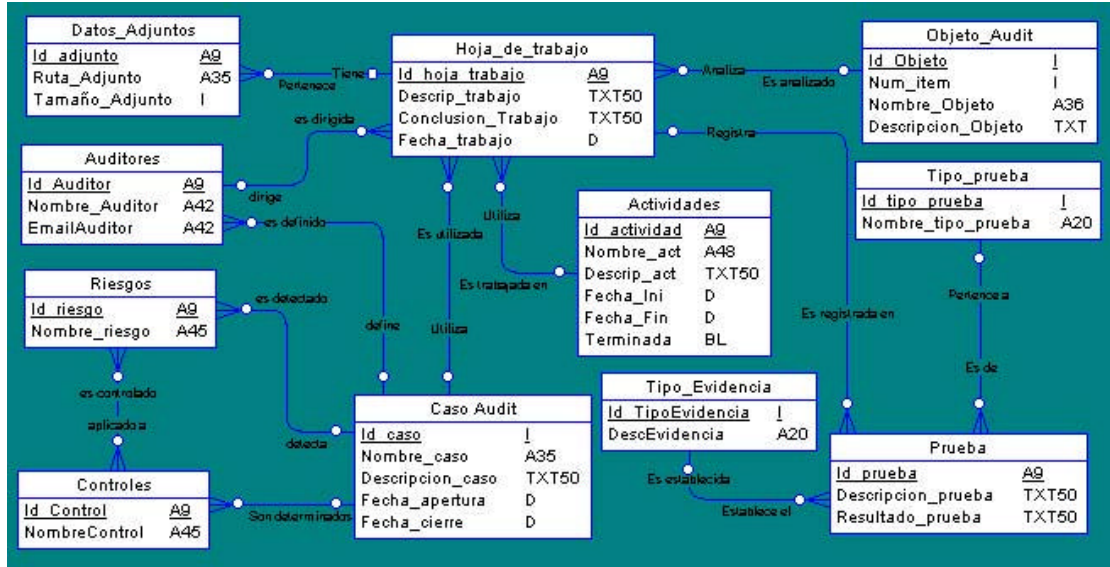
En la figura 55 se mostrará las entidades que participan en la creación del plan del auditoría y la tabla 34 muestra los detalles de cada entidad.

Tabla 34. Entidades presentes en el submodelo entidad – relación que participan en la creación del plan de auditoría.

TABLA	INFORMACION
Planificador	Registro del plan de auditoría.
Objetivos	Registro de objetivos del plan de auditoría.
Alcances	Registro de Alcances de trabajo de auditoría.
Actividades	Registro de actividades a realizar.
Técnicas	Registro de técnicas usadas para la realización de cada actividad.
Etapas	Registro de Fases en las que se divide el trabajo de auditoría.
Participantes	Registro de personal que participa en el desarrollo del plan de auditoría.
Recursos	Registro de recursos empleados en el plan de auditoría.
Estado	Registro de estado en que se encuentra el plan de auditoría.



Figura 56. Submodelo entidad relación – Examen auditado

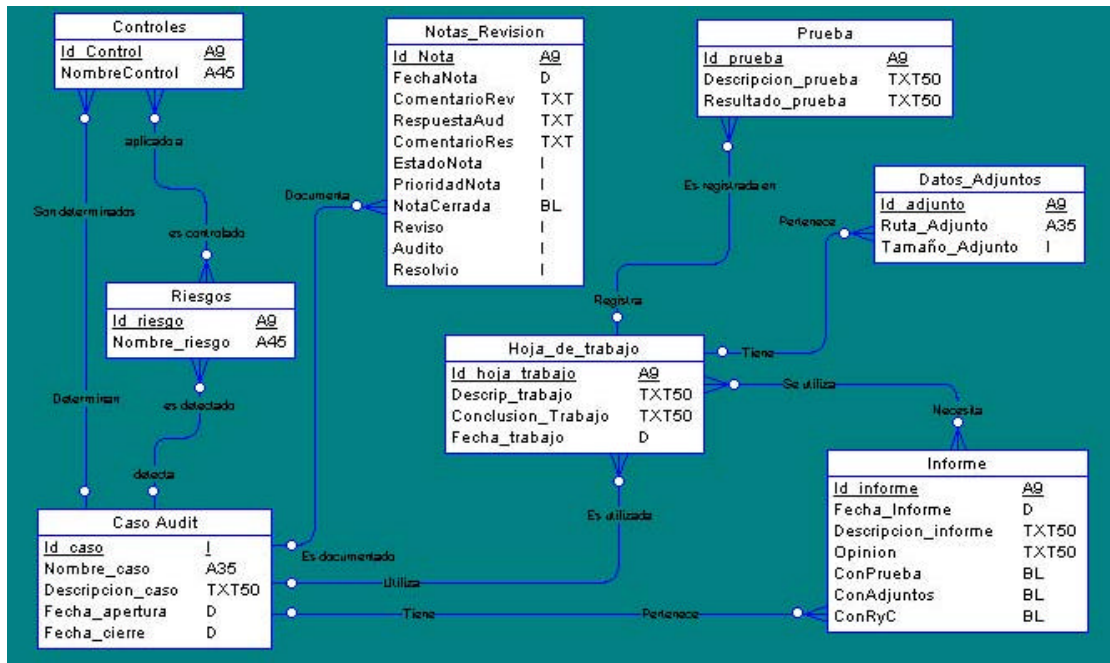


Siguiendo el mismo orden se mostrará las entidades responsables en la creación del informe de auditoría, la figura 57 muestra estos detalles y en la tabla 36 solo describe las entidades informe y notas\_revisión ya que las demás se han descrito en los anteriores submodelos.

Tabla 36. Entidades presentes en el submodelo entidad – relación que participan en la elaboración del informe de auditoría.

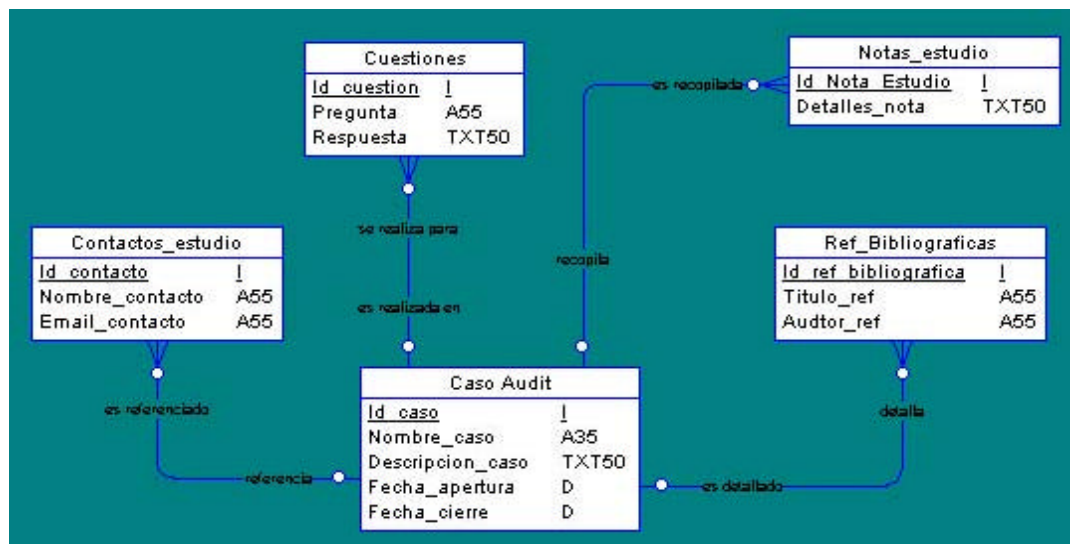
TABLA	INFORMACION
Informe	Registro de la opinión o conclusión final dada por el auditor.
Notas_revisión	Registro de notas de revisión de cada actividad realizada.

Figura 57. Submodelo entidad relación – Informe Auditoría.



Finalmente, en la figura 58 se mostrará las entidades presentes para el estudio de un caso de auditoría. Y una vez mas en la tabla 37 se describen las entidades presentes en el submodelo.

Figura 58. Submodelo entidad relación – Estudio de caso de auditoría



*Tabla 37. Entidades presentes en el submodelo entidad – relación que participan en el estudio de un caso de auditoría.*

TABLA	INFORMACION
Notas_estudio	Registro de notas sobre el caso estudiado.
Cuestiones	Preguntas y respuestas para el caso de estudio.
Contactos_estudio	Registro de contactos que pueden dar mas detalles o información sobre un caso de auditoría.
Ref_Bibliográficas	Registro de material para conceptos teóricos sobre el área de auditoría.

**6.4.2.3 Capa Software del sistema.** La Herramienta software propuesta se desarrolla bajo el sistema operativo Windows, sistema muy popular y caracterizado por los ambientes de manejo gráfico, la facilidad entre conmutación entre programas, la configuración de programas, el compartir datos, y la organización de sus programas y datos para un uso eficaz.

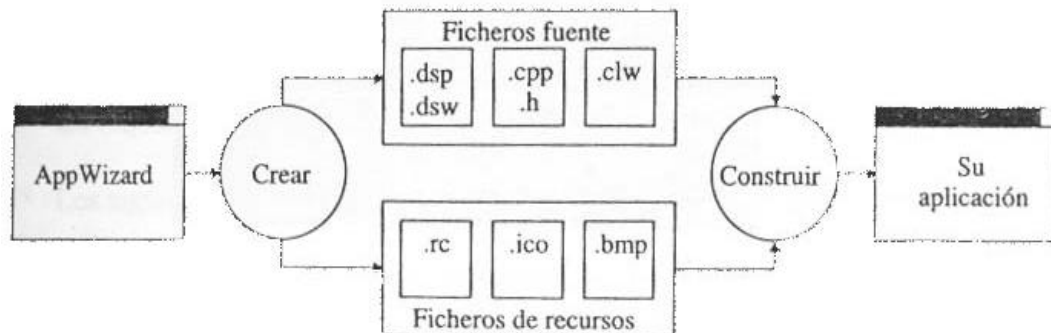
La herramienta se desarrolla en el lenguaje Visual C++ 6.0 utilizando la biblioteca de clases MFC (Microsoft Foundation Class) que facilita enormemente el desarrollo de aplicaciones. Visual C++ es un entorno de desarrollo diseñado especialmente para crear aplicaciones gráficas orientada a objetos, además proporciona varias formas de trabajo con base de datos.

Se utilizará Access como sistema de gestión de base de datos relacional (SGBDR) ya que es totalmente funcional. Proporciona la definición, manipulación y control de los datos que se necesitan para gestionar grandes cantidades de los mismos.

### **6.4.3 Flujo de trabajo de implementación.**

Según lo descrito en el apartado anterior se instaló el compilador Visual C++ 6.0 configurando además la versión de Microsoft Access 97. Y utilizando el AppWizard de visual C++ se creó una aplicación SDI base para el desarrollo de la herramienta, que será el esqueleto de la aplicación.

Figura 59. Creación del esqueleto de la aplicación usando el AppWizard.



Clase CAuditCasosApp creada con el AppWizard:

```

class CAuditCasosApp : public CWinApp
{
public:
    CAuditCasosApp();
// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CAuditCasosApp)
    public:
    virtual BOOL InitInstance();
    //}}AFX_VIRTUAL
// Implementation
    //{{AFX_MSG(CAuditCasosApp)
    afx_msg void OnAppAbout();
    // NOTE - the ClassWizard will add and remove member functions
    here.
    // DO NOT EDIT what you see in these blocks of generated code !
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
  
```

Clase CAuditCasosView creada por el AppWizard

```
class CAuditCasosView : public CView
{
public: // create from serialization only
    CAuditCasosView();
    DECLARE_DYNCREATE(CAuditCasosView)

// Attributes
public:
    CAuditCasosDoc* GetDocument();

// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
   //{{AFX_VIRTUAL(CAuditCasosView)
public:
    virtual void OnDraw(CDC* pDC); // overridden to draw this view
    virtual BOOL PreCreateWindow(CREATESTRUCT& cs);
    virtual void OnInitialUpdate();
protected:
    virtual BOOL OnPreparePrinting(CPrintInfo* pInfo);
    virtual void OnBeginPrinting(CDC* pDC, CPrintInfo* pInfo);
    virtual void OnEndPrinting(CDC* pDC, CPrintInfo* pInfo);
    //}}AFX_VIRTUAL

// Implementation
public:

    virtual ~CAuditCasosView();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif
};
```

```
#endif
```

```
protected:
```

```
// Generated message map functions
```

```
protected:
```

```
   //{{AFX_MSG(CAuditCasosView)
```

```
    /}}AFX_MSG
```

```
    DECLARE_MESSAGE_MAP()
```

```
};
```

La Clase CMainFrame creada por el AppWizard:

```
class CMainFrame : public CPersistentFrame
```

```
{
```

```
protected: // create from serialization only
```

```
    CMainFrame();
```

```
    DECLARE_DYNCREATE(CMainFrame)
```

```
// Attributes
```

```
public:
```

```
// Operations
```

```
public:
```

```
// Overrides
```

```
    // ClassWizard generated virtual function overrides
```

```
    {{{AFX_VIRTUAL(CMainFrame)
```

```
    virtual BOOL PreCreateWindow(CREATESTRUCT& cs);
```

```
    /}}AFX_VIRTUAL
```

```
// Implementation
```

```
public:
```

```
    virtual ~CMainFrame();
```

```

#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif

protected: // control bar embedded members
    CStatusBar m_wndStatusBar;
    CToolBar m_wndToolBar;

// Generated message map functions
protected:
   //{{AFX_MSG(CMainFrame)
    afx_msg int OnCreate(LPCREATESTRUCT lpCreateStruct);
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};

```

La clase CAuditCasosDoc creada con el AppWizard:

```

class CAuditCasosDoc : public CDocument
{
protected: // create from serialization only
    CAuditCasosDoc();
    DECLARE_DYNCREATE(CAuditCasosDoc)

// Attributes
public:

// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
   //{{AFX_VIRTUAL(CAuditCasosDoc)
public:

```

```

        virtual BOOL OnNewDocument();
        virtual void Serialize(CArchive& ar);
        //}}AFX_VIRTUAL
// Implementation
public:
    virtual ~CAuditCasosDoc();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif
protected:
// Generated message map functions
protected:
   //{{AFX_MSG(CAuditCasosDoc)
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};

```

## 6.5 EVALUACION DE LA FASE DE ELABORACION

Durante esta fase, se identificaron la mayoría de los casos de uso y se analizaron los más significativos de tal manera que se pudo establecer una línea base de la arquitectura que específicamente establece el esqueleto y la visión general de la herramienta software propuesta.

Dado que se identificaron algunos paquetes del análisis a través de los casos de uso más significativos, la fase de construcción pretenderá elaborar los casos de cada paquete y junto con el modelo entidad - relación se logrará un desarrollo sin mayores complicaciones.

# Capítulo 7

## FASE DE CONSTRUCCION

El propósito de esta fase es dejar listo el producto software en su versión operativa inicial a partir de la línea base de la arquitectura establecida en las fases anteriores. Para realizar esta labor se detallan algunos casos de uso y los escenarios restantes, modificando si es necesario la descripción de la arquitectura establecida para luego integrar todo el sistema y probarlo.

El énfasis se pone en completar las realizaciones de casos de uso y clases necesarios, implementándolos como componentes y probándolos tanto en forma individual como en construcciones de modo que se complete cada uno de los paquetes identificados; haciendo que el desarrollo de la herramienta sea más manejable.

En esta fase además se construirá la guía de usuario que permitirá orientar al usuario en el manejo y comprensión de la herramienta software. (Véase anexo B).

### 7.1 PLANIFICACION DE LA FASE DE CONTRUCCION

Un desarrollo iterativo, guiado por casos de uso y centrado en la arquitectura, construye el software mediante incrementos. Así se obtiene una versión específica ejecutable del sistema. Por ello las iteraciones que se van a realizar están organizadas de la siguiente manera:

*Tabla 38. Iteraciones de la fase de construcción*

ITERACION	PAQUETE/SUBSISTEMA	TIEMPO ESTIMADO
1	Gestión identificación	15 días
2	Gestión de creación caso	20 días

3	Planeación auditoría	15 días
4	Examen auditoría	15 días
5	Informe auditoría	15 días
6	Gestión de estudio caso	15 días
Total		95 días

La duración de esta fase es de 95 días realizando 6 iteraciones. La iteración 1 permite desarrollar el paquete de gestión de identificación de usuario y las iteraciones 2,3,4,5 permiten desarrollar el paquete gestión de creación de casos; finalmente la iteración 6 permite desarrollar el paquete de gestión de estudio de casos.

**7.1.1 Gestión de identificación.** Tal como se describió en las etapas anteriores el paquete está organizado teniendo en cuenta los casos de uso identificación de usuario y registro. Además, se mostrarán otros aspectos importantes como es el seguimiento del control de ingreso a través de una barra de progreso que indica el número de intentos al ingresar a la herramienta; teniendo en cuenta que después de tres intentos se obligue al usuario a registrarse.

**7.1.1.1 Implementación.** La herramienta software propuesta valida el rol, usuario y contraseña de acceso permitiendo al usuario el ingreso a la herramienta. Si el usuario no se encuentra registrado se podría elegir realizar el registro; si por el contrario el usuario intenta ingresar sin realizar su registro se notifica un mensaje de error. En el siguiente código se muestran las variables y operaciones de la clase Ingreso.

```
class CDlgIngreso : public CDialog
{
// Construction
public:
    CString const GetUsuario() const;
    int rol;
```

```

        int const get_Rol() const;
        void Inhabilitarcajas();
        void Habilitarcajas();
        CDlgIngreso(CWnd* pParent = NULL); // standard constructor
// Dialog Data
        {{{AFX_DATA(CDlgIngreso)
        enum { IDD = IDD_INGRESO };
        CProgressCtrl m_nProgreso_ingr;
        CComboBox m_nRol;
        CString m_nClave;
        CString m_nUsuario;
        }}}AFX_DATA
// Overrides
        // ClassWizard generated virtual function overrides
        {{{AFX_VIRTUAL(CDlgIngreso)
        protected:
        virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
        }}}AFX_VIRTUAL
// Implementation
protected:
        // Generated message map functions
        {{{AFX_MSG(CDlgIngreso)
        virtual void OnOK();
        afx_msg void OnClose();
        virtual void OnCancel();
        virtual BOOL OnInitDialog();
        afx_msg void OnRegistrar();
        }}}AFX_MSG
        DECLARE_MESSAGE_MAP()
};

```

La figura 60 muestra el dialogo para el ingreso de datos proporcionados por el usuario, y que es implementado en el código anterior.

Figura 60. Interfaz para el control de ingreso a la herramienta



Para el registro del usuario se ha considerado el siguiente código con sus variables y procedimientos:

```
class CDlgRegistro : public CDialog
{
// Construction
public:
    void CargarTbACombo(CDaoRecordset *pRs, int ctrol);
    BOOL ComprobarUsuario();
    ~CDlgRegistro();
    int CamposLlenos;
    VerificarCampos();
    CDaoRecordset* m_pRecordsetC;
    CDaoRecordset* m_pRsetProfesion;
    CDlgIngreso *m_pDlgIngreso;
    CDaoRecordset* m_pRecordset;
    CDaoDatabase* m_pBdcasos;
    int m_ncampos;
    void CajasABdatos();
    CDlgRegistro(CWnd* pParent = NULL); // standard constructor
// Dialog Data
//{{AFX_DATA(CDlgRegistro)
enum { IDD = IDD_REGISTRO };
```

```

        CComboBox m_sProfesion;
        CComboBox m_Rol;
        CComboBox m_ctrCiudad;
        CString    m_sApellidos;
        CString    m_Clave;
        CString    m_sComentario;
        CString    m_sConClave;
        CString    m_sDireccion;
        CString    m_sEmail;
        CString    m_sNombre;
        CString    m_sUsuario;
        CString    m_sTelefono;
    //}}AFX_DATA
// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CDlgRegistro)
protected:
    virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
    //}}AFX_VIRTUAL
// Implementation
protected:
    // Generated message map functions
    //{{AFX_MSG(CDlgRegistro)
afx_msg void OnNuevo();
afx_msg void OnSelchangeRolADesempenar();
virtual BOOL OnInitDialog();
afx_msg void OnGrabar();
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};

```

Y la interfaz de registro para interactuar con el usuario de la siguiente manera:

Figura 61. Interfaz para el registro de usuarios.

### 7.1.1.2 Pruebas

Se han establecido dos escenarios de prueba para el control de ingreso:

Tabla 39. Escenario 1 de prueba para el ingreso a la herramienta.

Escenario 1: Datos Inválidos		
Datos de entrada		
Datos necesarios para validarse en la herramienta		
Nombre	Valor	Observación
Rol de usuario	Auditor	Valido/Invalido
Nombre de usuario	alec	Valido/Invalido
Contraseña	xxxxx	Invalido
Resultado: Datos de salida		
Mensaje de error de Datos inválidos y progreso en la barra. En caso de reintento por tercera vez, se inhabilita el botón ingresar		

Tabla 40. Escenario 2 de prueba para el ingreso a la herramienta.

Escenario 2: Datos validos		
Datos de entrada		
Datos necesarios para validarse en la herramienta		
Nombre	Valor	Observación
Rol de usuario	Auditor	Valido
Nombre de usuario	alec	Valido
Contraseña	xxxxx	Valido
Resultado: Datos de salida		
Muestra la interfaz de vista documento.		

Para el registro del usuario se ha considerado también dos escenarios:

*Tabla 41. Escenario 1 de prueba para el registro de usuarios.*

Escenario 1: Datos Inválidos		
Datos de entrada		
Datos necesarios para el registro del usuario		
Nombre	Valor	Observación
Nombre		Valido/Invalido
Apellidos		Valido/Invalido
Dirección		Valido/Invalido
Ciudad		Valido/Invalido
Teléfono		Valido/Invalido
Profesión		Valido/Invalido
Email		Valido/Invalido
Rol		Valido/Invalido
Usuario		Valido/Invalido
Contraseña		Valido/Invalido
Confirmar contraseña		Valido/Invalido
Resultado: Datos de salida		
Mensaje de error de Datos inválidos para cada campo de entrada.		

Figura 62. Interfaz Principal de la herramienta software.

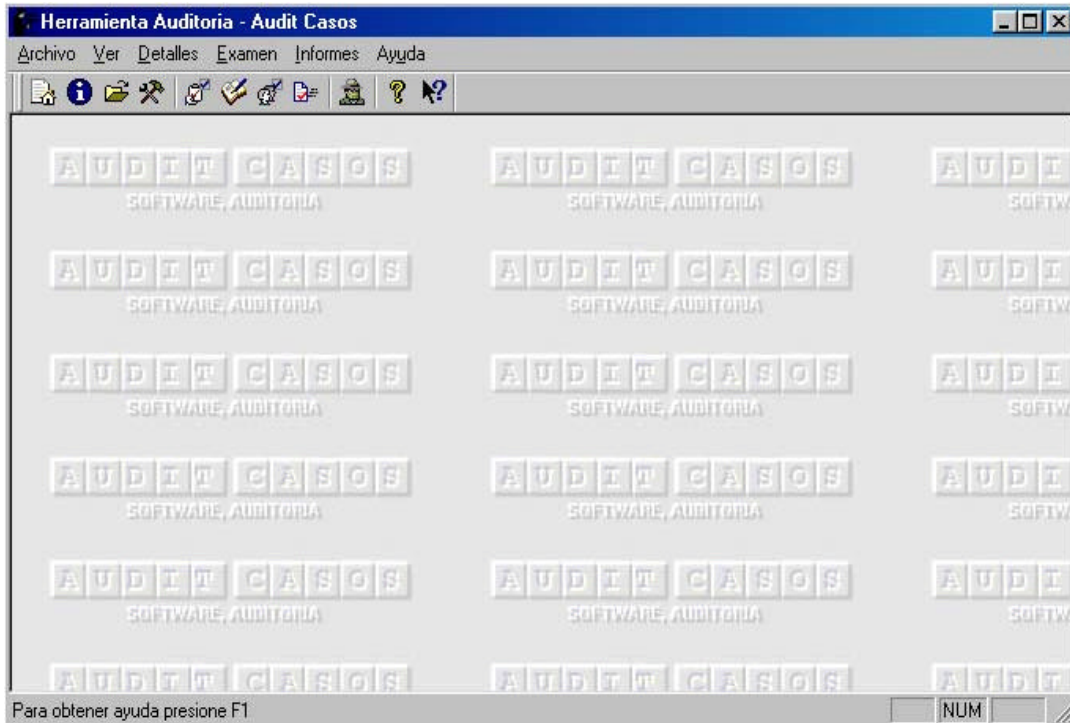


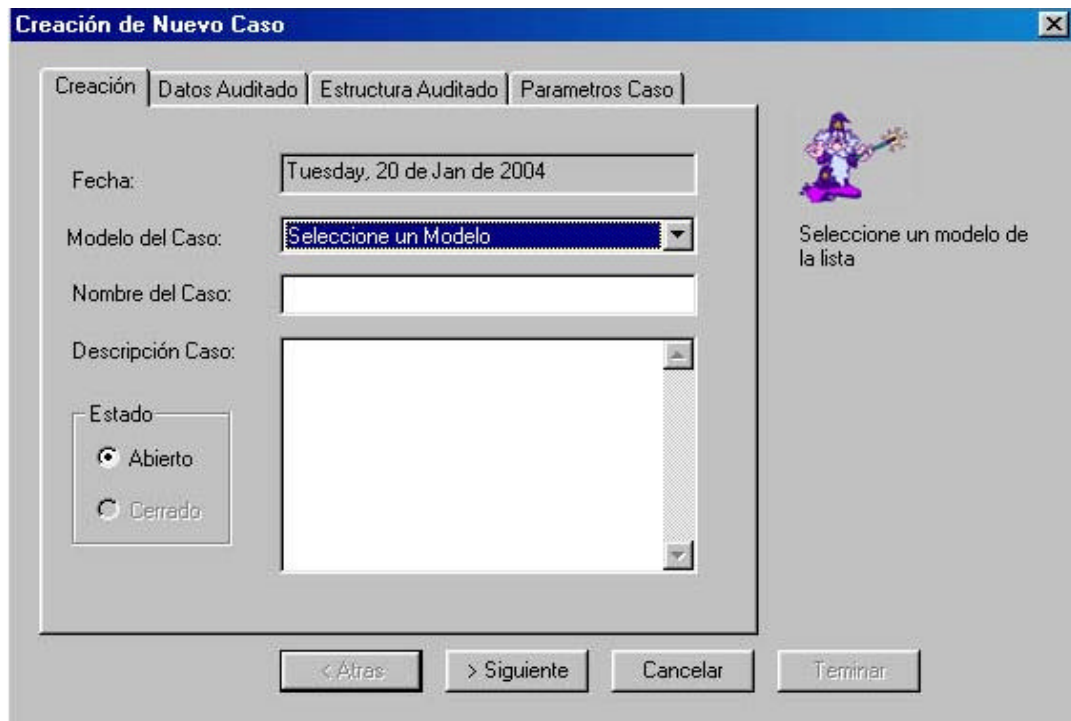
Tabla 42. Escenario 2 de prueba para el registro de usuarios.

Escenario 2: Datos validos		
Datos de entrada		
Datos necesarios para el registro del usuario		
Nombre	Valor	Observación
Nombre	José Antonio	
Apellidos	Pérez	
Dirección	Cra 27 no 34-15	
Ciudad	Bucaramanga	
Teléfono	6354875	
Profesión	Estudiante	
Email	Josea@hotmail.com	
Rol	Auditor	
Usuario	josea	
Contraseña	xxxx	
Confirmar contraseña	xxxx	
Resultado: Datos de salida		
Mensaje de registro adicionado		

### 7.1.2 Gestión de creación de casos.

Este paquete considera los casos de uso identificados, además de paquetes de servicio: datos auditado, plan de auditoría, examen auditoría e informe de auditoría. La figura 63 muestra la interfaz para la creación de un caso de auditoría.

Figura 63. Interfaz para la creación de un caso de auditoría



**7.1.2.1 Implementación.** En su implementación se ha considerado el siguiente código con sus variables y operaciones:

```
class CDlgCrearCaso : public CDialog
{
// Construction
public:
    void guardaropciones();
```

```

CString const GetUbicacionAuditado() const;
void CargarABDAuditor();
void ObtenerUsuarioBD();
CString EmailUsuario, NombreUsuario, ApellidoUsuario;
void GuardarCasodelUsuario();
CString usuario;
void Set_Usuario(CString &value);
CString const GetNitAuditado() const;
CString const GetNombreAuditado() const;
int const GetIdCaso() const;
void LimpiarCajas();
BOOL ComprobarCampLlenos();
void DesactivarBotones();
void ActivarBotones();
BOOL ComprobarNumItem(int num);
BOOL ComprobarItemDeArbol(CString item);
void CargarDeBdArbol();
void CargarArbolAbaseDatos();
int nPagMostrar;
long Id_Caso;
void MostrarControles();
BOOL ComprobarAuditado();
void CargarTbACombo(CDaoRecordset *pRs, int ctrol);
void CajasCasoABdatos();
void CajasAudABdatos();
void CajasABdatosRel();
void BasedatosACajasAudit();
CDaoDatabase * m_pBdCasos;
CDaoRecordset * m_pRsetCaso;
CDaoRecordset * m_pRsetUsuario;
CDaoRecordset * m_pRsetAuditores;
CDaoRecordset * m_pRsetModelos;
CDaoRecordset * m_pRsetTipo;

```

```

CDaoRecordset * m_pRsetCiudades;
CDaoRecordset * m_pRsetUsuarioCaso;
CDaoRecordset * m_pRsetCasoAuditado;
CDaoRecordset * m_pRsetAuditado;
CDaoRecordset * m_pRsetEstructura;
CDaoRecordset * m_pRsetConsEstr;
CDaoRecordset * m_pRsetOpciones;
int m_nCamposCaso;
CTime date;
BOOL ArbolCargado;
int m_nCamposRelacion;
int m_nCamposAuditado;
~CDlgCrearCaso();
CDlgCrearCaso(CWnd* pParent = NULL); // standard constructor
// Dialog Data
//{{AFX_DATA(CDlgCrearCaso)
enum { IDD = IDD_CREARCASO };
CButton      m_AlertaPlanes;
CButton      m_CheckAlertaNotas;
CButton      m_CheckAlertaCasoAb;
CButton      m_CheckMostrarDatos;
CButton      m_CheckApertura;
CTabCtrl     m_CtrlHojaPro;
CComboBox    m_nCiudad;
CComboBox    m_nModelo;
CComboBox    m_nTipo;
CTreeCtrl    m_EstructArbol;
int          m_Eabierto;
CString      m_sDirAuditado;
CString      m_sDescCaso;
CString      m_sSolicitud;
CString      m_sFecha;
CString      m_sNombreAudit;

```

```

        CString      m_sNombreCaso;
        CString      m_sTelefono;
        CString      m_sNit;
        CString      m_ActividadAudit;
    //}}AFX_DATA
// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CDIgCrearCaso)
    protected:
    virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
    //}}AFX_VIRTUAL
// Implementation
protected:
    // Generated message map functions
    //{{AFX_MSG(CDIgCrearCaso)
    afx_msg void OnSelchangeTabHojapro(NMHDR* pNMHDR, LRESULT*
pResult);
    afx_msg void OnSiguiente();
    afx_msg void OnAtras();
    afx_msg void OnBeginlabeleditTreeEstruct(NMHDR* pNMHDR, LRESULT*
pResult);
    afx_msg void OnBEliminar();
    afx_msg void OnBInsertar();
    afx_msg void OnSelchangedTreeEstruct(NMHDR* pNMHDR, LRESULT*
pResult);
    afx_msg void OnChangeNit();
    afx_msg void OnEndlabeleditTreeEstruct(NMHDR* pNMHDR, LRESULT*
pResult);
    afx_msg void OnMoverADer();
    afx_msg void OnMoverAlzq();
    afx_msg void OnTerminar();
    virtual BOOL OnInitDialog();
    afx_msg void OnContinuar();

```

```

afx_msg void OnSelchangeCModelo();
afx_msg void OnSetfocusNombreCaso();
afx_msg void OnSetfocusDescrCaso();
afx_msg void OnSetfocusNit();
afx_msg void OnSetfocusNombre();
afx_msg void OnSetfocusTipo();
afx_msg void OnSetfocusDireccion();
afx_msg void OnSetfocusTelefono();
afx_msg void OnSetfocusCiudad();
afx_msg void OnSetfocusDescripcion();
afx_msg void OnSetfocusCModelo();
afx_msg void OnSetfocusActividadaudit();
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
private:
    CBitmapButton m_EditEliminar;
    CBitmapButton m_EditAdd;
    CBitmapButton m_EditDmover;
    CBitmapButton m_EditIzmover;
    HTREEITEM estructura;
};

```

### 7.1.2.2 Pruebas

Se ha establecido dos escenarios de prueba

Tabla 43. Escenario 1 de prueba para la creación de un caso de auditoría.

Escenario 1: Datos validos		
Datos de entrada		
Datos necesarios para la creación de un caso		
Nombre	Valor	Observación
Modelo del caso	Centrado en el análisis	Valido
Nombre caso	Caso de exposición	Valido
Descripción caso	Aquí descripción	Valido

Nit Auditado	01245639-8	Valido
Nombre auditado	Empresa de ejemplo	Valido
Actividad auditado	Venta de suministros	Valido
Tipo auditado	Sociedad Anónima	Valido
Dirección auditado	Cra 23 no 26-14	Valido
Ciudad auditado	Bucaramanga	Valido
Teléfono auditado	6354789	Valido
Situación actual auditado	Aquí la situación actual	Valido
Resultado: Datos de salida		
Mensaje de creación de caso		

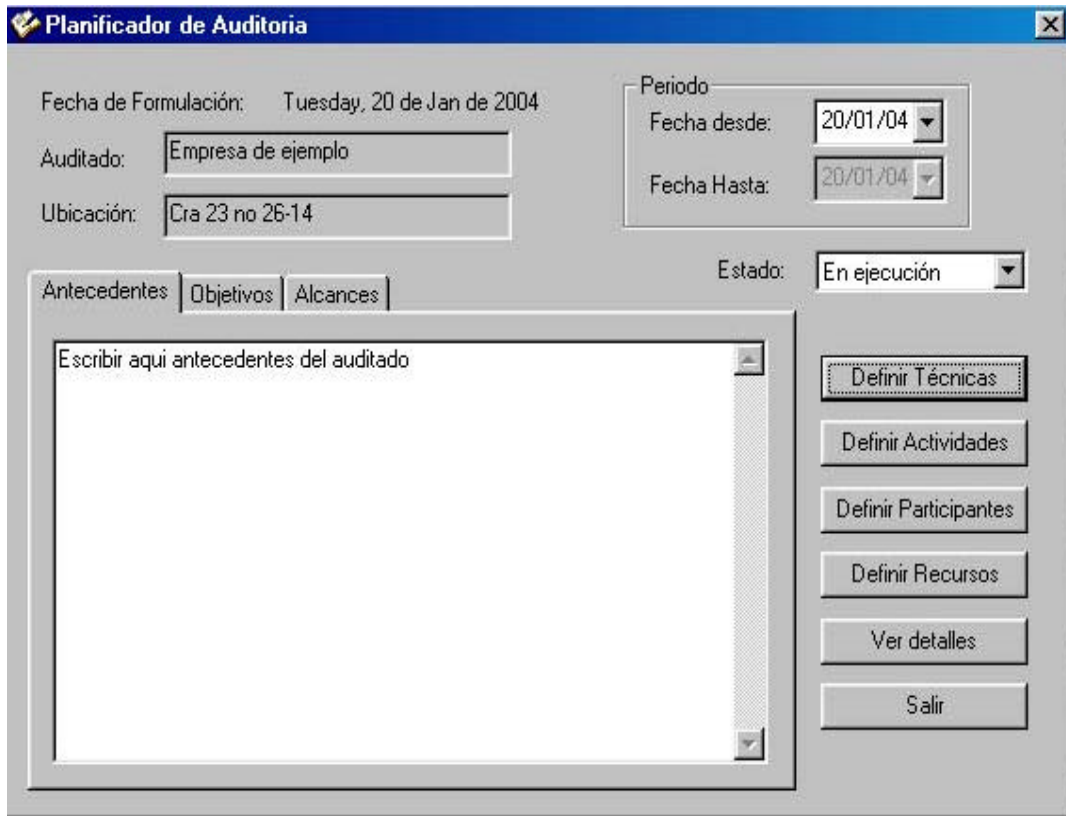
**Tabla 44. Escenario 2 de prueba para la creación de un caso de auditoría**

Escenario 2: Datos Inválidos		
Datos de entrada		
Datos necesarios para la creación de un caso		
Nombre	Valor	Observación
Modelo del caso		Valido/Invalido
Nombre caso		Valido/Invalido
Descripción caso		Valido/Invalido
Nit Auditado		Valido/Invalido
Nombre auditado		Valido/Invalido
Actividad auditado		Valido/Invalido
Tipo auditado		Valido/Invalido
Dirección auditado		Valido/Invalido
Ciudad auditado		Valido/Invalido
Teléfono auditado		Valido/Invalido
Situación actual auditado		Valido/Invalido
Resultado: Datos de salida		
Para cada una de las combinaciones de validez e invalidez se muestra un mensaje de dato invalido según sea el caso.		

### 7.1.3 Planeación caso de auditoría.

La figura 64 muestra la interfaz para el establecimiento de la planeación de un caso de auditoría, Nótese el establecimiento de actividades, técnicas, participantes, recursos.

Figura 64. Interfaz para establecimiento del plan de auditoría



### 7.1.3.1 Implementación

```
class CDlgPlanificador : public CDialog
{
// Construction
public:
    void Set_UbicacionAuditado(CString &value);
    int alcanceini;
    void ActualizarAlcances();
    int nreg;
    void MostrarAlcances();
    CString id_alcance;
    CString CrearIdAlcance();
```

```

    BOOL ComprobarAlcance(CString alcance);
    void GuardarAlcances();
    int nlnt;
    int filaini;
    void ActualizarObjetivos();
    void Mostrarobjetivos();
    CString CrearIdObjetivo();
    CString id_objet;
    BOOL ComprobarObj(CString obj);
    void GuardarObjetivos();
    int const GetIdPlan() const;
    int encontrado;
    int var;
    void BaseDatosACajas();
    int id_plan;
    void InhabilitarBotones();
    void HabilitarBotones();
    void CajasABdatos(CDaoRecordset *pRs);
    ~CDlgPlanificador();
    CDaoDatabase* m_pBdcasos;
    CDaoRecordset* m_pRsetEstado;
    CDaoRecordset* m_pRsetPlan;
    CDaoRecordset* m_pRsetplansinterminar;
    CDaoRecordset* m_pRsetObjetivos;
    CDaoRecordset* m_pRsetAlcance;
    CTime date;
    long Id_caso;
    void Set_IdCaso(long & value);
    void Set_NombreAuditado(CString& value);
    void MostrarControles();
    CDlgPlanificador(CWnd* pParent = NULL); // standard constructor
// Dialog Data
//{{AFX_DATA(CDlgPlanificador)

```

```

enum { IDD = IDD_DLG_PLANIFICADOR };
CEdit m_Antecedentes;
CDateTimeCtrl m_Fecha_desde;
CComboBox m_CtrlEstado;
CTabCtrl m_TabPlanificador;
CString m_sAuditado;
CString m_fechaHoy;
CGrid m_Objetivos;
CGrid m_Alcance;
CString m_sUbicacion;
//}}AFX_DATA
// Overrides
// ClassWizard generated virtual function overrides
//{{AFX_VIRTUAL(CDlgPlanificador)
protected:
virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
//}}AFX_VIRTUAL
// Implementation
protected:
HICON m_hIcon;
// Generated message map functions
//{{AFX_MSG(CDlgPlanificador)
virtual BOOL OnInitDialog();
afx_msg void OnSelchangeTabplanificador(NMHDR* pNMHDR, LRESULT*
pResult);
afx_msg void OnParticipantes();
afx_msg void OnActividades();
afx_msg void OnDetalles();
afx_msg void OnCrearPlan();
afx_msg void OnTecnicas();
afx_msg void OnRecursos();
virtual void OnOK();
afx_msg void OnClickGridobjetivos();

```

```

afx_msg void OnDbClickGridobjetivos();
afx_msg void OnKeyPressGridobjetivos(short FAR* KeyAscii);
afx_msg void OnClickGridalcance();
afx_msg void OnDbClickGridalcance();
afx_msg void OnKeyPressGridalcance(short FAR* KeyAscii);
DECLARE_EVENTSINK_MAP()
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
};

```

Figura 65. Interfaz para la definición de actividades.

**Definir Actividades**

Etapa:

Nombre Actividad:

Descripción:

Fecha Inicio:  Fecha Fin:

Técnica a usar:

Id Tecnica	Nombre Actividad
Tecn_1	Observación

Figura 66. Interfaz para definición de recursos.

Recursos Estimados

Etapa: Planeación

Nombre Recurso:

Costo/mes: 0

% Utilización: 0

	Nombre Recurso	Costo/mes	%Utilizacion
1	Computador	150000	50

Adicionar

Eliminar

Salir

Figura 67. Interfaz para definición de Técnicas .

Crear Técnicas

Id\_Técnica: Techn\_1

Nombre Técnica: Observación

Descripción Técnica: Consiste en presenciar personalmente la forma como determinado procesos operativos y de control se llevan a cabo

Nueva

Grabar

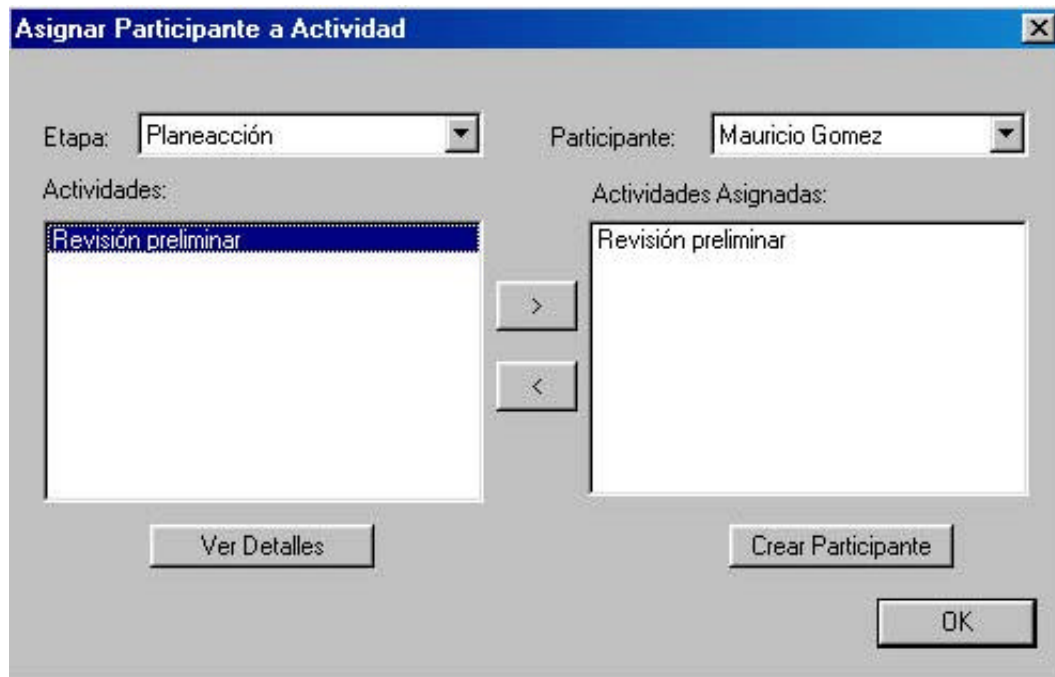
Modificar

Borrar

Salir

Inicio < Anterior Siguiete > Final

Figura 68. Interfaz para asignar actividades a participantes .



### 7.1.3.2 Pruebas

Se establecieron los siguientes escenarios de prueba

Tabla 45. Escenario 1 de prueba para creación de plan de auditoría.

Escenario 1: Datos validos		
Datos de entrada		
Datos necesarios para la creación de un plan de auditoría		
Nombre	Valor	Observación
Antecedentes	Registro de antecedentes	Valido
Objetivos	Registro de objetivos	Valido
Alcance	Registro de alcances	Valido
Resultado: Datos de salida		
Creación del plan para el caso creado y habilitación de botones de técnicas, actividades, recursos, participantes		

**Tabla 46. Escenario 2 de prueba para creación de plan de auditoría.**

Escenario 2: Datos Inválidos		
Datos de entrada		
Datos necesarios para la creación de un plan de auditoría		
Nombre	Valor	Observación
Creación caso		Invalido
Antecedentes	Registro de antecedentes	Valido
Objetivos	Registro de objetivos	Valido
Alcance	Registro de alcances	Valido
Resultado: Datos de salida		
Mensaje de imposibilidad para crear plan para caso desconocido		

**Tabla 47. Escenario de prueba 3 para creación de plan de auditoría.**

Escenario 3: Datos para actividades, técnicas, recursos, participantes		
Datos de entrada		
Datos necesarios para la creación técnicas, actividades, recursos, participantes		
Nombre	Valor	Observación
Actividades	Registro de actividades	Valido/Invalido
Técnicas	Registro de técnicas	Valido/Invalido
Participantes	Registro de participantes	Valido/Invalido
Recursos	Registro de recursos	Valido/Invalido
Resultado:		
Se muestra mensaje de creación en caso de datos validos y en caso de datos inválidos no se permite su registro.		

#### **7.1.4 Exámen auditoría.**

La figura 69 muestra la interfaz utilizada para el examen de auditoría basándose en la hoja de trabajo, nótese esta interfaz permite el registro de los detalles del trabajo realizado y las descripciones de las pruebas. La figura 70 muestra la interfaz para seleccionar la actividad a trabajar y que fue creada durante la etapa de planificación de auditoría. La figura 71y 72 muestran las interfaces correspondientes al registro de notas de revisión para el trabajo realizado y el registro de riesgos y controles identificados para el caso.

Figura 69. Interfaz para establecer examen de auditoría (hoja de trabajo).

The screenshot shows a software window titled "Hoja de Trabajo" with a menu bar containing "Archivo", "Ver", and "Herramientas". The interface is divided into several sections:

- Activity Selection:** A text field for "Actividad:" with a "Seleccionar" button.
- Auditor Information:** "Auditado por:" is set to "Alexander Calderon Roj" and "Apoyado por:" is "No definido".
- Worksheet Details:** "Hoja de Trabajo No.:" is "Ht\_2".
- Characteristics of the Worksheet:** "Estado:" is "En ejecución" and "Objeto Auditable:" is "Objeto no definido".
- Test Section:** A checkbox for "Prueba" is unchecked. Below it, "No definida" and "Sin evidencia" are selected in dropdown menus. An "Adjuntar Datos" button is present.
- Work Done and Test Description:** Two large empty text areas with scrollbars, labeled "Trabajo Realizado" and "Descripción Prueba".
- Conclusions and Recommendations:** A large empty text area with a scrollbar.
- Footer:** "Auditado:" is "Empresa de ejemplo" and "Fecha:" is "20/01/2004". Navigation buttons (back, close, forward) are on the right.

Figura 70. Interfaz para selección de actividad a trabajar

The screenshot shows a software window titled "Seleccionar Actividad" with a close button in the top right corner. The interface includes:

- Stage Selection:** "Etapa:" is set to "Planeación".
- Activity Selection:** "Actividades:" list shows "Revisión preliminar" selected.
- Details:** A text area containing the text: "Obtener la información necesaria para que el auditor pueda tomar la decisión de como proceder en la auditoria."
- Techniques to Use:** A text area containing the text: "Observación".
- Action:** A "Salir" button is located at the bottom right.

#### **7.1.4.1 Implementación**

```
class CDlgHojaT : public CDialog
{
// Construction
public:
    BOOL ValidarCampos();
    void CargarAuditores();
    void CargarRevisores();
    void Inhabilitarcajas();
    void HabilitarCajas();
    void CargarPruebasACajas();
    BOOL BuscarEnPruebas(CString Id_hoja);
    int campop;
    void CargarPruebaABD();
    COleVariant dato;
    BOOL BuscarenActividades(CString idactividad);
    int nInt;
    int campo;
    int var;
    bool pruebasalvada;
    bool salvado;
    int numreg;
    void CargarBDCajas();
    BOOL comprobarIdHt(CString Idhojat);
    CTime date;
    void GuardarHojaTrabajo(CDaoRecordset * m_pRset);
    CString Id_actividad;
    int Id_plan;
    void Set_IdPlan(int &value);
    long m_IdCaso;
    CString m_NitAuditado;
    void Set_IdCaso(long &value);
    void Set_NitAuditado(CString &value);
```

```

void Set_NombreAuditado(CString &value);
void CargarTbACombo(CDaoRecordset *pRs, int ctrol);
CDaoDatabase * m_pBdCasos;
CDaoRecordset * m_pRsetEstadoHoja;
CDaoRecordset * m_pRsetParticipantes;
CDaoRecordset * m_pRsetObjetosAudit;
CDaoRecordset * m_pRsetTipoPruebas;
CDaoRecordset * m_pRsetPrueba;
CDaoRecordset * m_pRsetEvidencia;
CDaoRecordset * m_pRsetHojaTrabajo;
CDaoRecordset * m_pRsetActividades;
CDaoRecordset * m_pRsetConsultaHoja;
CDaoRecordset * m_pRsetAuditores;
~CDlgHojaT();
void MostrarControles();
CDlgHojaT(CWnd* pParent = NULL); // standard constructor
// Dialog Data
//{{AFX_DATA(CDlgHojaT)
enum { IDD = IDD_HOJATRABAJO };
CComboBox m_CtrlIdsAuditores;
CComboBox m_CtrlAuditores;
CComboBox m_CtrlRevisores;
CComboBox m_CtrlIds;
CComboBox m_ObjetoAudit;
CComboBox m_estado;
CComboBox m_CtrlEvidencia;
CComboBox m_TipoPrueba;
CButton m_checkprueba;
CStatic m_Static_ResPrueba;
CStatic m_Static_conclusion;
CEdit m_ResultPruebaT;
CEdit m_DescTrabajoT;
CEdit m_DescPruebaT;

```

```

    CEdit m_ConclusionT;
    CTabCtrl m_TabHojaTrabajo;
    CString m_Auditado;
    CString m_actividad;
    CString m_FechaHoy;
    CString m_CodHoja;
    //}}AFX_DATA

// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CDIgHojaT)
protected:
    virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
    //}}AFX_VIRTUAL

// Implementation
protected:
    // Generated message map functions
    //{{AFX_MSG(CDIgHojaT)
    virtual void OnOK();
    virtual BOOL OnInitDialog();
    afx_msg void OnSelchangeHojatrabajo(NMHDR* pNMHDR, LRESULT*
pResult);
    afx_msg void OnSelActHoja();
    afx_msg void OnCheckprueba();
    afx_msg void OnVerInicio();
    afx_msg void OnArchivoSalir();
    afx_msg void OnVerSiguiente();
    afx_msg void OnVerAnterior();
    afx_msg void OnVerFinal();
    afx_msg void OnArchivoNuevahoja();
    afx_msg void OnArchivoBorrar();
    afx_msg void OnArchivoGuardar();
    afx_msg void OnHadelante();
    afx_msg void OnHatras();

```

```

afx_msg void OnHerramientasRiesgoscontroles();
afx_msg void OnArchivoBorrarprueba();
afx_msg void OnHerramientasNotasderevisin();
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
private:
    CBitmapButton m_HojaAdelante;
    CBitmapButton m_HojaAtras;
    CBitmapButton m_HojaCerrar;
};

```

Figura 71. Interfaz para registro de Notas de revisión del trabajo realizado

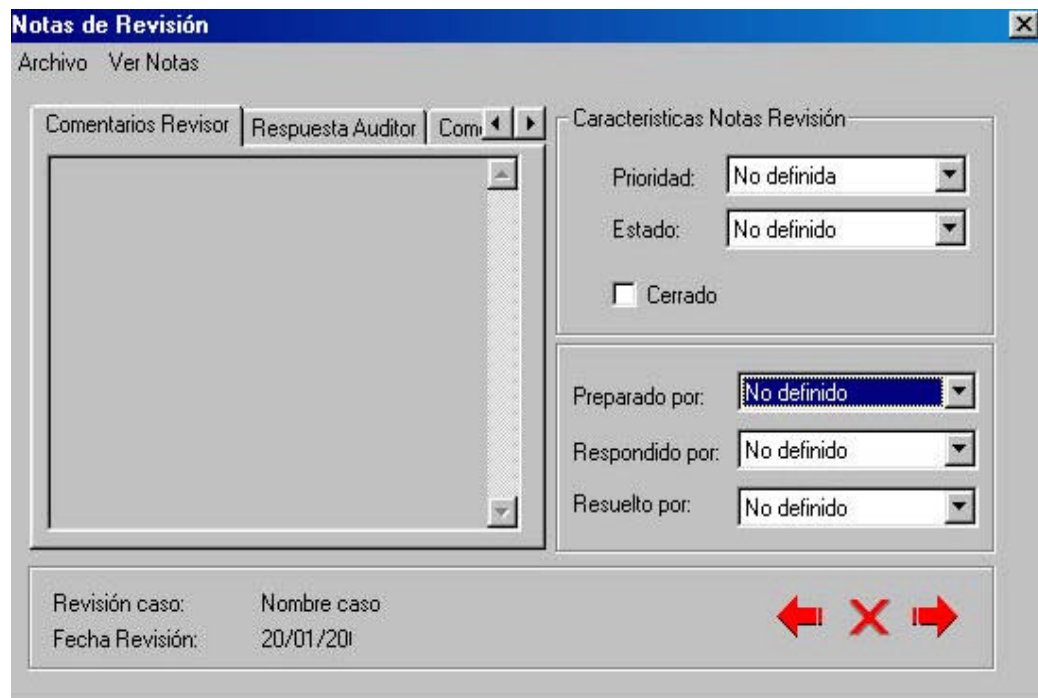
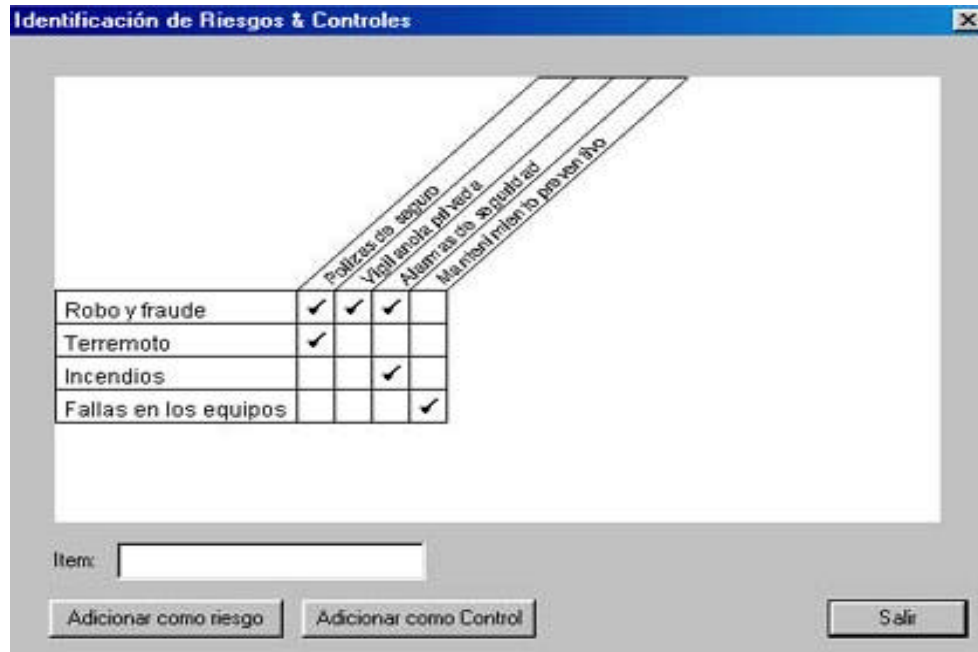


Figura 72 Interfaz para identificación de riesgos y controles.



#### 7.1.4.2 Pruebas

Se han establecido los siguientes escenarios de prueba:

Tabla 48. Escenario de prueba 1 para registro en hojas de trabajo.

Escenario 1: Datos validos para creación de hojas de trabajo		
Datos de entrada		
Datos necesarios para la creación de hojas de trabajo		
Nombre	Valor	Observación
Trabajo realizado	Descripción	Valido
Conclusiones y recomend	Descripción	Valido
Identificar riesgos y controles	Descripción	Interfaz para identificación
Notas de revisión	Descripción	Interfaz registro
Resultado:		
Seleccionada la actividad del plan se procede a realizar una descripción del trabajo realizado para la actividad. El resultado es el registro del trabajo realizado por cada actividad y adicionalmente un registro de riegos y controles identificados. Las notas de revisión se realiza de acuerdo al contenido o proceso realizado durante el registro en las hojas de trabajo.		

*Tabla 49. Escenario de prueba 2 para registro en hojas de trabajo.*

Escenario 2: Datos inválidos para creación de hojas de trabajo		
Datos de entrada		
Datos necesarios para la creación de hojas de trabajo		
Nombre	Valor	Observación
Trabajo realizado	Descripción	Valido/Invalido
Conclusiones y recomend	Descripción	Valido/Invalido
Identificar riesgos y controles	Descripción	Interfaz para identificación
Notas de revisión	Descripción	Interfaz registro
Resultado:		
Para cualquier combinación de datos validos o inválidos no se permite su registro, mostrando su mensaje correspondiente.		

### 7.1.5 Informe de auditoría

La figura 73 muestra la interfaz correspondiente para la creación de un informe de auditoría.

*Figura 73. Interfaz para creación de informe de auditoría.*

### 7.1.5.1 Implementación

```
class CDlgInforme : public CDialog
{
// Construction
public:
    void GuardarRelacion(CString hoja);
    void CargarHojaAlista();
    long IdCaso;
    CTime date;
    void Set_IdCaso(long &value);
    void GuardarInforme();
    BOOL ComprobarNumInfo(CString id_info);
    void CrearNoInforme();
    ~CDlgInforme();
    CDaoDatabase * m_pBdCasos;
    CDaoRecordset * m_pRsetInforme;
    CDaoRecordset * m_pRsetConsultaHojas;
    CDaoRecordset * m_pRsetRelacion;
    CDlgInforme(CWnd* pParent = NULL); // standard constructor

// Dialog Data
   //{{AFX_DATA(CDlgInforme)
    enum { IDD = IDD_DLGINFORME };
    CComboBox m_ListadeHojas;
    CString m_NumInfo;
    CString m_DesclInfo;
    CString m_OpinionInfo;
    BOOL m_Conprueba;
    BOOL m_ConAdjunto;
    BOOL m_ConRiesgoyC;
    int m_ConHoja;
    //}}AFX_DATA

// Overrides
    // ClassWizard generated virtual function overrides
```

```

//{{AFX_VIRTUAL(CDIgInforme)
protected:
virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
//}}AFX_VIRTUAL

// Implementation
protected:

// Generated message map functions
//{{AFX_MSG(CDIgInforme)
virtual BOOL OnInitDialog();
afx_msg void OnGuardarinfo();
afx_msg void OnRhojat();
afx_msg void OnRunahojat();
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
};

```

### 7.1.5.2 Pruebas

Se han establecido los siguientes escenarios de prueba.

*Tabla 50. Escenario de prueba 1 para creación de informe.*

Escenario 1: Datos validos		
Datos de entrada		
Datos necesarios para la creación de informe		
Nombre	Valor	Observación
Descripción informe	Descripción	Valido
Opinión	Descripción	Valido
Resultado:		
Registro de la opinión y trabajo realizado en la auditoría según hojas seleccionadas		

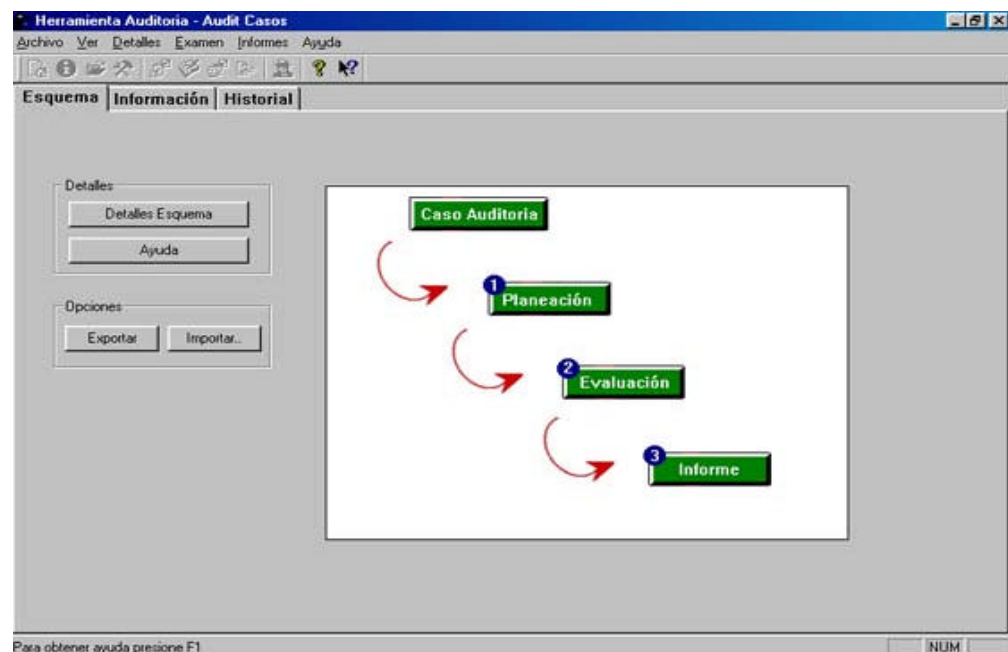
*Tabla 51. Escenario de prueba 2 para creación de informe.*

Escenario 2: Datos inválidos		
Datos de entrada		
Datos necesarios para la creación de informe		
Nombre	Valor	Observación
Caso		Invalido
Descripción informe	Descripción	Valido
Opinión	Descripción	Valido
Resultado:		
Muestra de mensaje de error para caso de auditoría desconocido		

### 7.1.6 Gestión de estudio de caso.

Para el estudio de un caso de auditoría se considera las tres fases o etapas contempladas en la auditoría; iniciando por la selección de un caso, analizando su planificación, examen e informe.

Figura 74. Interfaz para estudio de caso de auditoría.



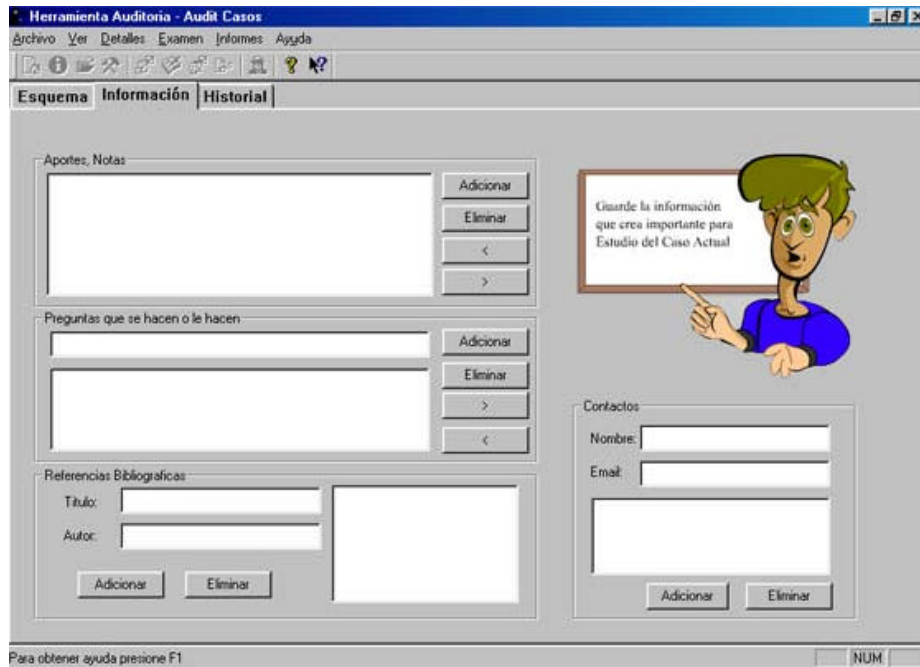
El esquema presentado en la interfaz de la figura 74 permite la selección de cada una de las fases antes mencionadas, de tal manera que al dar un simple click se muestran diálogos con la información del caso en particular. Por ejemplo para abrir un caso de auditoría se presenta la figura 75.

Figura 75. Interfaz para apertura de un caso de auditoría.



El registro de las notas de apreciación del caso llevadas por el aprendiz se muestra en la figura 75, allí se puede registrar notas, preguntas y repuestas acerca del caso, referencias bibliográficas para tener en cuenta, contactos de personas que pueden dar una opinión del caso.

Figura 76. Interfaz para registro de notas del caso en estudio



### 7.1.6.1 Implementación.

```
class CDlgAbrirCaso : public CDialog
{
// Construction
public:
    void ObtenderDetallesCaso(long codcaso);
    int const GetIdPlan() const;
    const CString GetUbicacion() const;
    const CString GetAuditado() const;
    const CString GetNitAudit() const;
    int IdPlan;
    CString NitAudit;
    CString UbicacionAudit;
    CString Auditado;
    void DatosDelCaso(long caso);
```

```

    int nInt;
    long element;
    long const GetCodCaso() const;
    int ListCods[50];
    ~CDlgAbrirCaso();
    CDaoDatabase * m_pBdCasos;
    CDaoRecordset * m_pRsetCaso;
    void PoblarControlLista();
    CDlgAbrirCaso(CWnd* pParent = NULL); // standard constructor
// Dialog Data
//{{AFX_DATA(CDlgAbrirCaso)
enum { IDD = IDD_ABRIRCASO };
CListCtrl      m_CasosSel;
CString        m_DescripcionCaso;
//}}AFX_DATA
// Overrides
// ClassWizard generated virtual function overrides
//{{AFX_VIRTUAL(CDlgAbrirCaso)
protected:
virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
//}}AFX_VIRTUAL
// Implementation
protected:
// Generated message map functions
//{{AFX_MSG(CDlgAbrirCaso)
virtual BOOL OnInitDialog();
afx_msg void OnItemchangedListadecasos(NMHDR* pNMHDR, LRESULT*
pResult);
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
};

```

**7.1.6.2 Pruebas.** Se han establecido los siguientes escenarios de prueba

*Tabla 52. Escenario de prueba 1 para creación de notas del caso en estudio.*

Escenario 1: Datos validos		
Datos de entrada		
Datos necesarios para creación de notas del caso en estudio		
Nombre	Valor	Observación
Notas, aportes	Descripción	Valido
Preguntas- respuestas	Descripción	Valido
Ref.Bibliograficas	Descripción	Valido
contactos	Descripción	Valido
Resultado:		
Registro de notas, preguntas y respuestas, ref. Bibliográficas, contactos para el caso en estudio.		

*Tabla 53. Escenario 2 de prueba para creación de notas del caso en estudio.*

Escenario 2: Datos Inválidos		
Datos de entrada		
Datos necesarios para creación de notas del caso en estudio		
Nombre	Valor	Observación
Caso de estudio	Cod_caso	Invalido
Notas, aportes	Descripción	Valido/Invalido
Preguntas- respuestas	Descripción	Valido/Invalido
Ref.Bibliograficas	Descripción	Valido/Invalido
contactos	Descripción	Valido/Invalido
Resultado:		
No Registro de notas, preguntas y respuestas, ref. Bibliográficas, contactos para el caso en estudio. Mostrando mensaje de error para caso desconocido.		

## 7.2 EVALUACION DE LA FASE DE CONSTRUCCION

Durante esta fase se logró construir la versión inicial de la herramienta software de auditoría “*Audit Casos*” a partir de la línea base de la arquitectura; trabajando con las iteraciones propuestas en la planeación.

Sobre los resultados de las pruebas y otros criterios de evaluación se logró corregir algunas deficiencias encontradas para alcanzar la capacidad operativa inicial de la herramienta.

Se deja para la etapa de transición la operatividad de la herramienta en manos de los usuarios para así detectar algunos problemas, riesgos y defectos que no se han puesto en evidencia durante las pruebas.

## CONCLUSIONES

- ❖ La orientación que ofrece la metodología didáctica del estudio de casos y la metodología de auditoría propuesta, permite establecer un marco metodológico y de trabajo, que hace posible el desarrollo de la propuesta de herramienta software de apoyo al aprendizaje de la auditoría de sistemas de información en el ambiente académico.
- ❖ El método del caso, o el estudio de casos es un método pedagógico que específicamente pretende el conocimiento y valoración de los procesos de intervención diseñados y aplicados por expertos en situaciones resueltas y el entrenamiento en el diagnóstico de situaciones concretas y es considerado en el proyecto como una perspectiva que contribuye a un mejor conocimiento de la realidad en el campo de la auditoría de sistemas de información, permitiendo a los estudiantes desempeñar roles que permitan una comprensión de los procesos, en un ambiente académico y práctico, creando los mecanismos de reflexión (Incremento y desarrollo de sus capacidades de análisis) y autoevaluación en su proceso de aprendizaje.
- ❖ El uso del Proceso Unificado de desarrollo del Software como metodología de desarrollo de la propuesta de herramienta software de auditoría “*Audit casos*” es un aspecto clave que permite la consolidación del proyecto. La importancia del proceso unificado para las futuras generaciones del proyecto radica en el establecimiento de una guía centrada en la arquitectura que es comprensible, adaptable, robusta y evolutiva.
- ❖ El desarrollo dirigido por casos de uso proporciona enormes beneficios al proyecto debido a que son un medio sistemático e intuitivo de capturar requisitos funcionales centrándose en el valor añadido por el usuario, además dirigen todo el proceso de

desarrollo debido a que la mayoría de las actividades como el análisis, diseño y prueba se llevan a partir de casos de uso.

- ❖ El desarrollo de este proyecto constituye un aporte importante para futuras investigaciones en el campo de la auditoría de sistemas de información ya que enfoca sus esfuerzos en apropiarse de los conocimientos teóricos sobre el área, abriendo expectativas con respecto los campos de acción que permiten un profundización mayor sobre el tema.

## BIBLIOGRAFÍA

ALARCON GUERRERO, Carlos A. y MESTRE PIANETA, Ernesto F. Tesis de grado: Prototipo software para asistir procesos de calidad soportado en los estándares ISO 9000. Bucaramanga: Universidad Industrial de Santander. 2000.

ANDREU, Rafael, RICART Joan E y VALOR Josep. Estrategias y Sistemas de Información. Madrid: McGraw-Hill, 1996.

BATES, Jon y TOMKINS, Tim. Descubre Visual C++ 6. Primera edición. Madrid: Prentice Hall, 1999. 687 p.

CARCAMO SEPULVEDA, José. Bases de Datos Relacionales: Un enfoque práctico en el diseño. Ediciones UIS. Bucaramanga: Universidad Industrial de Santander. 1996. 199 p.

CEBALLOS SIERRA, Francisco Javier. Visual C++ 6. Aplicaciones para Win32. Segunda Edición. Madrid: RA-MA, 2000. 752 p.

CORNELLA, Alfons. Los recursos de información: Ventaja competitiva de las empresas. España: McGraw-Hill. 1994.

ECHENIQUE GARCIA, José Antonio. Auditoría en Informática. Segunda edición. México: McGraw-Hill, 2001, 300 p.

GÓMEZ FLORES, Luis Carlos. Introducción a la Auditoría de Sistemas. Bucaramanga Universidad Industrial de Santander, 1993.

HERNÁNDEZ HERNÁNDEZ, Enrique. Auditoría en Informática: Un enfoque metodológico y práctico. Primera Edición. México: Compañía Editorial Continental, S.A de CV, 1996. 315 p.

JACOBSON, Ivar – BOOCH, Grady y RUMBAUGH James. El proceso Unificado de desarrollo de software. Primera Edición. Madrid: ADDISON WESLEY, 2000. 438 p.

JACOBSON, Ivar – BOOCH, Grady y RUMBAUGH James. El lenguaje Unificado de Modelado: Manual de referencia. Primera Edición. Madrid: ADDISON WESLEY, 2000. 526 p.

LEURO RAMIREZ, Nicolás y TRISTANCHO CELIS, William. Tesis de grado: Auditoría de Sistemas de información para el sistema de información de la Corporación Educativa ASED. Bucaramanga, 2003.

LOPEZ, Alfonso. Iniciación al análisis de casos: Una metodología activa de aprendizaje en grupos. Bilbao, España Ediciones Mensajero S.A, 1997.

MONROY VECINO, Olga Lucia. Tesis de grado: Conceptos generales de auditoría de sistemas para aplicaciones en funcionamiento. Bucaramanga: Universidad Industrial de Santander, 1995.

MORA VILLAMIZAR, Andrea Marcela. Tesis de grado: Sistema intranet de información para apoyo de la actividad académica en el programa de especialización en docencia universitaria del CEDEUIS – SIEDUIS 1.0. Bucaramanga: Universidad Industrial de Santander, 2002.

MORGAN, Gareth. Imágenes de la organización. Primera Edición. Madrid: RA-MA, 1996. 408 p.

MURRAY, Chris y PAPPAS, Chris H. Microsoft Visual C++: Manual de referencia. McGraw-Hill. 1999.

MUCCHIELLI, Roger. El método del caso. Primera edición. España. Ibérica Europea de Ediciones. S.A. 1970.

OLAVE CACERES, Yesid A. Tesis de grado: Propuesta de un modelo de evaluación de la administración de la información en las organizaciones empresariales. Bucaramanga: Universidad Industrial de Santander, 2002, 157 p.

PERRY, Greg. Aprendiendo programación orientada a objetos con turbo C++ en 21 días. Primera Edición. México: Prentice-Hall. 1995. 775 p.

PIATTINI Mario G y del Peso Emilio. Auditoría Informática: Un enfoque práctico. España: RA-MA. 1998. 609 p.

PINILLA FORERO, José Dagoberto. Auditoría Informática: Un enfoque Operacional. Segunda Edición. Bogotá: ECOE Ediciones, 1997. 252 p.

PRESSMAN, Roger. Ingeniería del software: Un enfoque práctico. Cuarta edición. España. McGraw-Hill. 1998.

SÁNCHEZ MARTÍNEZ, Amparo y OCHOA MUSITU, Gonzalo. El estudio de Casos para profesionales de la acción Social. Primera Edición. Madrid, Narcea S.A de Ediciones, 1995.

SENN, James A. Análisis y diseño de sistemas de información. Segunda Edición. México: McGraw-Hill Interamericana de México, 1992. 942 p.

VIESCAS, John L. Running Microsoft Access 97. Primera Edición. España, McGraw-Hill. 1997. 893 p.

Instituto Tecnológico y de estudios superiores de Monterrey [Online]. Mexico: Dirección de Investigación y Desarrollo Educativo. El estudio de casos como técnica didáctica. Disponible en Internet: <http://www.sistema.itesm.mx/va/dide/inf-doc/estrategias/> en formato PostScript.

## **APENDICE A**

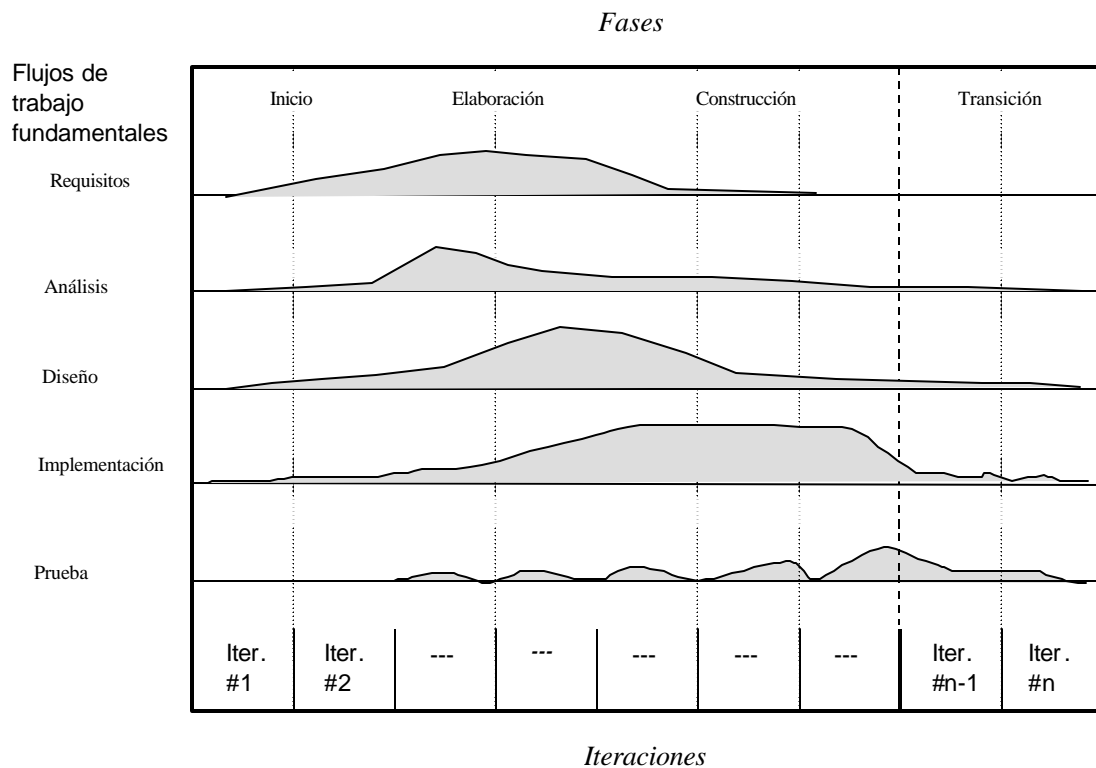
### ***EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE***

El proceso Unificado es un proceso de desarrollo de software que incluye un conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. Este proceso es un proceso de desarrollo iterativo y gradual, en el sentido de que el software no se libera de un solo gran golpe al final del proyecto, sino que, al contrario, se desarrolla y se libera por partes.

El objetivo del Proceso Unificado de Desarrollo de Software es permitir la producción de un software de la mayor calidad que satisfaga las necesidades de los usuarios finales, dentro de planificaciones y presupuestos predecibles. El Proceso Unificado captura algunas de las mejores prácticas de desarrollo de software, de una forma que es adaptable a un amplio rango de proyectos y organizaciones. En el aspecto de la gestión, el Proceso Unificado proporciona un enfoque disciplinado sobre cómo asignar tareas y responsabilidades dentro de una organización del desarrollo de software.

#### **Panorámica del Proceso**

Como ilustra la figura A1, el proceso Unificado consta de las cuatro fases siguientes: Fase de inicio, Fase de elaboración, Fase de construcción y Fase de transición.



*Figura A1. Flujos de trabajo sobre cada fase*

Dentro de cada fase hay iteraciones. Una iteración representa un ciclo de desarrollo completo, desde la captura de requisitos hasta la implementación y pruebas, que produce como resultado la entrega al cliente o la salida al mercado de un proyecto ejecutable.

Cada fase e iteración se centra en disminuir algún riesgo y concluye con un hito bien definido. La revisión de hitos es el momento adecuado para evaluar cómo se están satisfaciendo los objetivos y si el proyecto necesita ser reestructurado de alguna forma para continuar.

Así, durante la fase de inicio se establece la planificación del proyecto y se delimita su alcance. En la elaboración, se reúnen requerimientos mas detallados, se hacen análisis y diseños de alto nivel a fin de establecer una arquitectura base, y se crea un plan de construcción. En la fase de construcción, se desarrolla de forma iterativa e incremental un producto completo que está preparado para la transición hacia la comunidad de usuarios.

Incluso con este tipo de proceso iterativo, a menudo aparecen cuestiones que requieren desarrollo adicional para ajustar el sistema, corregir algunos problemas no detectados o finalizar algunas características que habían sido propuestas, ello constituye la fase de transición.

Ahora se mostrará la mayor cantidad de detalles y la información necesaria acerca del proceso de desarrollo abordado.

### **Características del proceso**

*Las actividades de desarrollo bajo el Proceso Unificado están dirigidas por los casos de uso.* El Proceso Unificado pone un gran énfasis en la construcción de sistemas basada en una amplia comprensión de cómo se utilizará el sistema que se entregue. Por tanto para construir un sistema con éxito se debe conocer lo que sus futuros usuarios necesitan y desean. Las nociones de los casos de uso y los escenarios se utilizan para guiar el flujo de proceso desde la captura de los requisitos hasta las pruebas, y para proporcionar caminos que se pueden reproducir durante el desarrollo del sistema.

*El desarrollo bajo el Proceso Unificado está centrado en la arquitectura.* El papel de la arquitectura software es parecido al papel que juega la arquitectura en la construcción de edificios. El edificio se contempla desde varios puntos de vista: estructura, servicios, conducción de calefacción, fontanería, electricidad, etc. Esto permite al constructor ver una imagen completa antes de que comience la construcción. El proceso se centra en establecer al principio una arquitectura software que guía el desarrollo del sistema. Tener una arquitectura robusta facilita el desarrollo en paralelo, minimiza la repetición de trabajos e incrementa la probabilidad de reutilización de componentes y el mantenimiento posterior del sistema. Este diseño arquitectónico sirve como una sólida base sobre la cual se puede planificar y manejar el desarrollo de software basado en componentes.

*El Proceso Unificado es un proceso iterativo e incremental.* Para los sistemas simples, parece perfectamente factible definir de forma secuencial el problema

completo, diseñar la solución completa, construir el software y, a continuación, hacer pruebas con el producto final. Sin embargo, dadas la complejidad y sofisticación que demandan los sistemas actuales, este enfoque lineal al desarrollo de sistemas no es realista. Un enfoque iterativo propone una comprensión incremental del problema a través de refinamientos sucesivos y un crecimiento incremental de una solución efectiva a través de varios ciclos. Como parte del enfoque iterativo se encuentra la flexibilidad para acomodarse a nuevos requisitos o a cambios tácticos en los objetivos del negocio. También permite que el proyecto identifique y resuelva los riesgos más bien pronto que tarde.

*Las actividades del Proceso Unificado destacan en la creación y el mantenimiento de modelos más que documentos sobre papel.* Los modelos (especialmente aquellos especificados mediante UML<sup>14</sup>) proporcionan representaciones ricas, desde el punto de vista semántico, del sistema software que se está desarrollando. Estos modelos pueden verse de muchas formas, y la información representada puede ser capturada instantáneamente y controlada electrónicamente. La razón subyacente al interés que pone el Proceso Unificado en los modelos, antes que en los documentos sobre papel, es minimizar la sobrecarga asociada con la generación y el mantenimiento de los documentos y maximizar el contenido de información relevante.

## **Flujos de trabajo fundamentales**

El Proceso Unificado consta de cinco flujos de trabajo.

1. **Requisitos** Describe la estructura y la dinámica del espacio del problema, así como el método basado en casos de uso para extraer los requisitos.
2. **Análisis** Describe las diferentes vistas arquitectónicas mediante un lenguaje cercano a los arquitectos del sistema.

---

<sup>14</sup> Véase Apéndice B El lenguaje Unificado de Modelado

3. Diseño Describe las diferentes vistas arquitectónicas mediante un lenguaje cercano a los desarrolladores del sistema.

4. Implementación Tiene en cuenta el desarrollo de software, la prueba de unidades y la integración

5. Pruebas Describe los casos de pruebas, los procedimientos y las métricas para evaluación de defectos.

Dentro de cada flujo de trabajo del proceso hay un conjunto de artefactos y actividades relacionados. Un *artefacto* es algún documento, informe o ejecutable que se produce, se manipula o se consume. Una *actividad* describe las tareas (pasos de concepción, realización y revisión) que llevan a cabo los trabajadores para crear o modificar los artefactos, junto con las técnicas y guías para ejecutar las tareas, incluyendo quizá el uso de herramientas para ayudar a automatizar algunas de ellas.

A algunos de estos flujos de trabajo del proceso se les asocian importantes conexiones entre los artefactos. Por ejemplo, el modelo de casos de uso que se genera durante la captura de requisitos es *realizado por* el modelo de diseño del proceso de análisis y diseño, es *implementado por* el modelo de diseño del proceso de implementación, y es *verificado por* el modelo de pruebas del proceso de pruebas.

### **Iteraciones en el proceso de desarrollo**

Una iteración es un miniproyecto, un recorrido más o menos completo a lo largo de todos los flujos de trabajo fundamentales, que obtiene como resultado una versión interna. La planificación de las iteraciones para cada fase depende, básicamente, de la complejidad del sistema propuesto. Un proyecto simple podría ser realizado con una sola iteración por fase, mientras que un proyecto más complicado podría requerir de más iteraciones. Por ejemplo:

Fase de inicio: Una iteración, dedicada definir el ámbito del sistema.

Fase de elaboración: Dos iteraciones, la primera para esbozar la arquitectura y la segunda para completar la línea base de la arquitectura.

Fase de construcción: Dos iteraciones, para asegurar que los incrementos resultantes funcionan satisfactoriamente.

Fase de transición: Una iteración.

Los detalles de iteraciones posteriores pueden estar limitados por el conocimiento disponible de la iteración actual y sus duraciones variarán, dependiendo del tamaño del sistema, entre una semana y tres meses por cada iteración.

## **Fases del Proceso de desarrollo**

### **Inicio.**

Antes de esta fase, se puede tener simplemente una idea interesante. La fase de inicio se trata precisamente de recopilar una declaración de objetivos, esbozar una arquitectura, desarrollar un análisis de negocio razonable, incluso delimitar un problema que se quiere resolver. Durante la fase de iniciación, se establece la planificación<sup>15</sup> del proyecto y se delimita su alcance. La planificación del proyecto incluye los criterios de éxito, la evaluación del riesgo, estimaciones de recursos que se necesitarán y un plan de fases que muestre la planificación de los hitos principales. Durante la iniciación, es frecuente crear un prototipo ejecutable que sirva para probar los conceptos.

Al final de la fase de inicio se examinan los objetivos del ciclo de vida del proyecto y se decide si proceder con el desarrollo del sistema.

Productos clave:

- ✓ Una lista de características.

---

<sup>15</sup> Véase modelo de Elaboración de planificación.

- ✓ Una primera versión del modelo del negocio<sup>16</sup> (o del dominio) que describe el contexto del sistema.
- ✓ Esbozo de los modelos que representan una primera versión del modelo de casos de uso<sup>17</sup>.
- ✓ Un primer esquema de la arquitectura candidata.
- ✓ Posiblemente un prototipo exploratorio que muestre el uso del nuevo sistema.
- ✓ Una lista inicial de riesgos y una clasificación de casos de uso.
- ✓ Los rudimentos del plan para el proyecto en su totalidad.
- ✓ Un primer borrador del análisis del negocio.

### **Elaboración.**

Los objetivos de la fase de elaboración son analizar el dominio del problema, establecer una base arquitectónica sólida, desarrollar el plan del proyecto y eliminar los elementos de más alto riesgo del proyecto. Las decisiones arquitectónicas deben tomarse con una comprensión del sistema global. Esto implica que se deben describir la mayoría de los requisitos del sistema. Para verificar la arquitectura, se implementa un sistema que demuestre las distintas posibilidades de la arquitectura y ejecute los casos de usos significativos.

Al final de la fase de elaboración se examinan el alcance y los objetivos del sistema, la elección de la arquitectura y la resolución de los riesgos más grandes, y se decide si se debe pasar a la construcción.

Productos clave:

- ✓ Preferiblemente un modelo completo del negocio (o del dominio) que describe el contexto del sistema.
- ✓ Una nueva versión de todos los modelos.
- ✓ Una línea base de la arquitectura.
- ✓ El plan de proyecto para las fases de construcción y transición.

---

<sup>16</sup> Puede apoyarse en plantillas de descripción del proceso del negocio. Véase plantilla A1.

<sup>17</sup> Utilícese plantillas que permitan la identificación de casos de uso. Véase plantilla A2.

- ✓ La lista de riesgos actualizada.
- ✓ El Análisis del negocio completo, incluido el presupuesto.

### **Construcción.**

Durante la fase de construcción, se desarrolla de forma iterativa e incremental un producto completo que está preparado para la transición hacia la comunidad de usuarios. Esto implica describir los requisitos restantes y los criterios de aceptación, refinando el diseño y completando la implementación y las pruebas del software.

Al final de la fase de construcción se decide si el software, los lugares donde se instalará y los usuarios están todos preparados para empezar a funcionar.

Productos clave:

- ✓ El plan de proyecto para la fase de transición.
- ✓ El sistema software ejecutable.
- ✓ Todos los artefactos, incluyendo los modelos del sistema.
- ✓ El análisis del negocio, que refleja la situación final de la fase.

### **Transición.**

Durante la fase de transición, el software se despliega en la comunidad de usuarios. Una vez que el sistema ha sido puesto en manos de los usuarios finales, a menudo aparecen cuestiones que requieren un desarrollo adicional para ajustar el sistema, corregir algunos problemas no detectados o finalizar algunas características que habían sido pospuestas. Esta fase comienza normalmente con una versión beta del sistema, que luego será reemplazada con el sistema de producción.

Al final de la fase de transición se decide si se han satisfecho los objetivos del ciclo de vida del proyecto, y se determina si se debería empezar otro ciclo de desarrollo. Este es también un punto en el que se asimilan las lecciones aprendidas en el



3. Modelo de casos de uso                      Establece los requisitos funcionales del sistema.
  
4. Modelo de análisis                              Establece un diseño de las ideas.
  
5. Modelo de diseño Establece el vocabulario del problema y su solución.
  
6. Modelo del proceso (opcional)      Establece los mecanismos de concurrencia y sincronización del sistema.
  
7. Modelo de despliegue      Establece la topología hardware sobre la cual se ejecutará el sistema.
  
8. Modelo de implementación      Establece las partes que se utilizarán para ensamblar y hacer disponible el sistema físico.
  
9. Modelo de pruebas              Establece las formas de validar y verificar el sistema.

Una vista es una proyección de un modelo. En el Proceso Unificado, la arquitectura de un sistema se captura en forma de cinco vistas que interactúan entre sí: la vista de diseño, la vista de procesos, la vista de despliegue, la vista de implementación y la vista de casos de uso.

### **Otros artefactos**

Los artefactos del Proceso Unificado se clasifican en artefactos de gestión y artefactos técnicos. Los artefactos técnicos del Proceso Unificado pueden dividirse en cuatro conjuntos principales.

1. Conjunto de requisitos                      Describe qué debe hacer el sistema
  
2. Conjunto de diseño                              Describe cómo se va a construir el sistema

3. Conjunto de implementación Describe el ensamblado de los componentes software

4. Conjunto de despliegue Proporciona todos los datos para la configuración entregable.

**Conjunto de requisitos.** Este conjunto agrupa toda la información que describe lo que debe hacer el sistema. Esto puede comprender un modelo de casos de uso, un modelo de requisitos no funcionales, un modelo del dominio, un modelo de análisis y otras formas de expresión de las necesidades del usuario, incluyendo pero no limitándose a maquetas, prototipos de la interfaz, restricciones legales, etcétera.

**Conjunto de diseño.** Este conjunto agrupa información que describe cómo se va a construir el sistema y captura las decisiones acerca de cómo se va a realizar, teniendo en cuenta las restricciones de tiempo, presupuesto, aplicaciones existentes, reutilización, objetivos de calidad y demás consideraciones. Esto puede implicar un modelo de diseño, un modelo de pruebas y otras formas de expresión de la naturaleza del sistema, incluyendo, pero no limitándose, a prototipos y arquitecturas ejecutables.

**Conjunto de implementación.** Este conjunto agrupa toda la información acerca de elementos software que comprende el sistema, incluyendo, pero no limitándose, a código fuente en varios lenguajes de programación, archivos de configuración, archivos de datos, componentes software, etcétera, junto con la información que describe cómo ensamblar el sistema.

**Conjunto de despliegue.** Este conjunto agrupa toda la información acerca de la forma en que se empaqueta actualmente el software, se distribuye, se instala y se ejecuta en el entorno destino.

## **APENDICE B**

### ***EL LENGUAJE UNIFICADO DE MODELADO***

El lenguaje Unificado de modelado (Unified Modeling Lenguaje, UML) es un lenguaje estándar para escribir planos de software y puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. El lenguaje de modelado es la notación (principalmente gráfica) de que se valen los métodos para expresar sus diseños.

UML está pensado principalmente para sistemas con gran cantidad de software y ha sido utilizado de forma efectiva en dominios tales como: Sistemas de información, Bancos y servicios financieros, telecomunicaciones, transporte, defensa / industria aeroespacial, comercio, electrónica médica, ámbito científico, servicios distribuidos basados en la web.

#### **Modelo conceptual de UML**

Para una comprensión de UML, se necesita adquirir un modelo conceptual del lenguaje, y esto requiere aprender tres elementos principales: los bloques básicos de construcción de UML, las reglas que dictan como se pueden combinar estos bloques básicos y algunos mecanismos comunes que se aplican a través de UML.

#### **Bloques de construcción de UML**

El vocabulario UML incluye tres clases de bloques de construcción: Elementos, Relaciones, Diagramas. Los elementos son abstracciones que hacen parte de un modelo; las relaciones ligan estos elementos entre sí y los diagramas agrupan colecciones interesantes de elementos. La figura B1 resume esta clasificación.

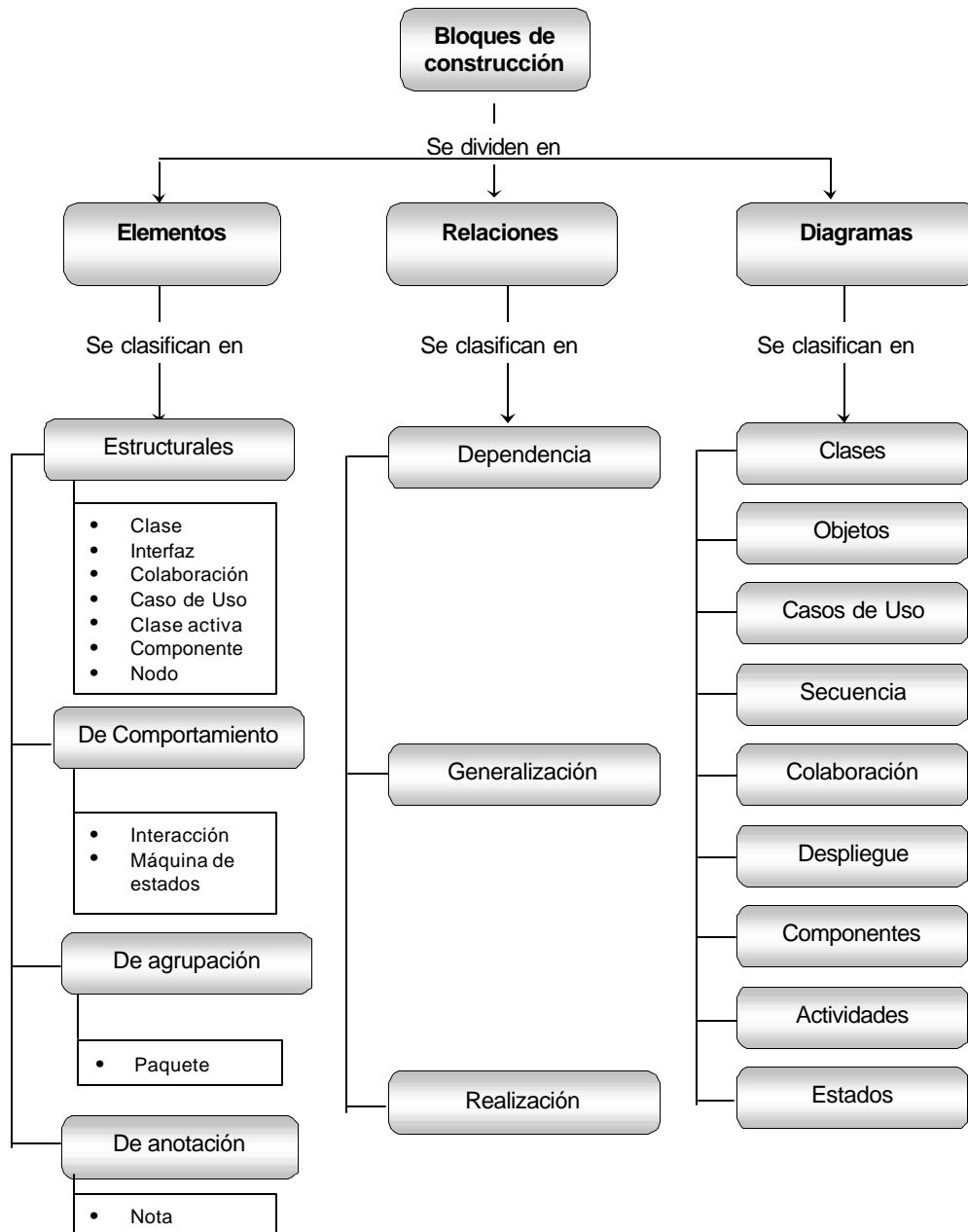


Figura B1. Bloques de Construcción

### Elementos Estructurales

Son los nombres de los modelos UML. En su mayoría son las partes estáticas de un modelo, y representan cosas que son conceptuales o materiales. La tabla B1 resume estos elementos.

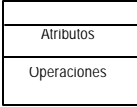

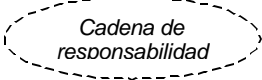
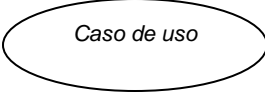
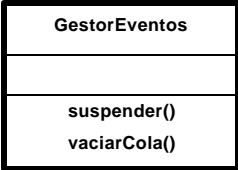
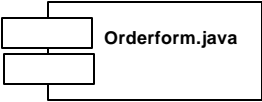
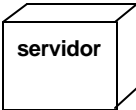
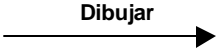

ELEMENTOS ESTRUCTURALES		
ELEMENTO	DESCRIPCION	REPRESENTACION
CLASE	Es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica. Una clase implementa una o más interfaces.	
INTERFAZ	Colección de operaciones que especifican un servicio de una clase o componente, describe el comportamiento visible externamente de ese elemento y representa el comportamiento completo de una clase o componente o sólo una parte de ese comportamiento	   ortografía
COLABORACIÓN	Define una interacción y es una sociedad de roles y otros elementos que colaboran para proporcionar un comportamiento cooperativo mayor que la suma de los comportamientos de los elementos	 <i>Cadena de responsabilidad</i>
CASO DE USO	Descripción de un conjunto de secuencias de acciones que un sistema ejecuta y que produce un resultado observable de interés para un actor particular. Se utiliza para estructurar los aspectos de comportamiento de un modelo.	 <i>Caso de uso</i>
CLASE ACTIVA	Es una clase cuyos objetos tienen uno o más procesos o hilos de ejecución y por lo tanto pueden dar origen a actividades de control. Es igual a una clase, excepto en que sus objetos representan elementos cuyo comportamiento es concurrente con otros elementos.	
COMPONENTE	Es una parte física y reemplazable de un sistema que conforma con un conjunto de interfaces y representa típicamente el empaquetamiento físico de diferentes elementos lógicos, como clases, interfaces y colaboraciones.	 Orderform.java
NODO	Es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional, que por lo general dispone de algo de memoria y, con frecuencia, capacidad de procesamiento.	 servidor

Tabla B1. Elementos estructurales

## Elementos de comportamiento


Son las partes dinámicas de los modelos UML. Estos son los verbos de un modelo, y representan comportamiento en el tiempo y el espacio. En total hay dos tipos principales de elementos de comportamiento. Semánticamente estos elementos están conectados normalmente a diversos elementos estructurales como clases, objetos y colaboraciones.

ELEMENTOS DE COMPORTAMIENTO		
ELEMENTO	DESCRIPCION	REPRESENTACION
INTERACCION	Comportamiento que comprende un conjunto de mensajes intercambiables entre un conjunto de objetos.	
MAQUINA DE ESTADOS	Comportamiento que especifica la secuencia de estados por las que pasa un objeto.	

*Tabla B2. Elementos de comportamiento*

## Elementos de agrupación


Son las partes organizativas de los modelos UML. Estos son las cajas en las que puede descomponerse un modelo.

ELEMENTOS DE AGRUPACIÓN		
ELEMENTO	DESCRIPCION	REPRESENTACION
PAQUETE	Ofrece un mecanismo general para la organización de los modelos/subsistemas agrupando elementos de modelado	

*Tabla B3. Elementos de agrupación*

## Elementos de anotación

Son comentarios que se pueden aplicar para describir, clarificar y hacer observaciones sobre cualquier elemento de un modelo, el principal elemento de anotación es la nota que es un símbolo para mostrar restricciones y comentarios junto a un elemento o una colección de elementos.

ELEMENTOS DE ANOTACION		
ELEMENTO	DESCRIPCION	REPRESENTACION
NOTA	Es un símbolo para mostrar restricciones y comentarios junto a un elemento o una colección de elementos	

*Tabla B4. Elementos de anotación*

## Relaciones

Son los bloques básicos de construcción para relaciones de UML. Se utilizan para escribir modelos bien formados. Existen cuatro tipos de relaciones en UML resumidas en la tabla B5.

## Diagramas en UML

Los diagramas se dibujan para visualizar un sistema desde diferentes perspectivas, y constituyen una proyección de un sistema, puede contener en teoría cualquier combinación de elementos y relaciones. En la práctica, sin embargo, sólo surge un pequeño número de combinaciones, las cuales son consistentes con las cinco vistas más útiles que comprenden la arquitectura de un sistema con gran cantidad de software. Cabe resaltar que la conveniencia de utilizar un determinado diagrama está determinada por el aporte que éste da al entendimiento del sistema. No está determinada ni por la metodología, ni por el propio lenguaje UML. En la tabla B6 se resumen los diagramas incluidos en UML.

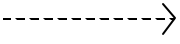
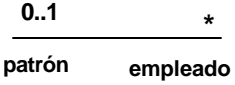
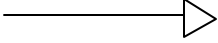
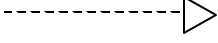
RELACIONES		
RELACION	DESCRIPCION	REPRESENTACION
DEPENDENCIA	Relación semántica entre dos elementos, en la cual un cambio a un elemento (el elemento independiente) puede afectar a la semántica del otro elemento (el elemento dependiente).	
ASOCIACIÓN	Relación estructural que describe un conjunto de enlaces, los cuales son conexiones entre objetos. Representa una relación estructural entre un todo y sus partes. A menudo incluye otros adornos como la multiplicidad y los nombres de rol.	
GENERALIZACIÓN	Relación de especialización / generalización en la cual los elementos del elemento especializado (el hijo) pueden sustituir a los objetos del elemento general (el padre). De esta forma, el hijo comparte la estructura y el comportamiento del padre. Se representa con una línea continua con una punta de flecha vacía apuntando al padre.	
REALIZACION	Relación semántica entre clasificadores, en donde un clasificador (clases, interfaces, señales, nodos, casos de uso y subsistemas) especifica un contrato que otro clasificador garantiza que cumplirá. Se pueden encontrar relaciones de generalización en dos sitios: entre interfaces y las clases y componentes que las realizan y entre los casos de uso y las colaboraciones que los realizan.	

Tabla B5. Relaciones

DIAGRAMAS	
DIAGRAMA	DESCRIPCION
De clases	Muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones.
De Objetos	Muestra un conjunto de objetos y sus relaciones.

De casos de uso	Muestra un conjunto de casos de uso y actores (Un tipo especial de clases) y sus relaciones.
De Secuencia	Representan un conjunto de elementos de un sistema que interaccionan entre ellos organizados en secuencias de tiempo.
De colaboración	Es una representación espacial de los objetos, sus enlaces y sus interacciones. Junto con los de secuencias se denominan, diagramas de interacción.
De Estados	Muestra una maquina de estados, que consta de estados, eventos y actividades.
De actividades	Es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro de un sistema.
De componentes	Muestra la organización y las dependencias entre un conjunto de componentes.
De despliegue	Muestra un conjunto de nodos y sus relaciones. Representa el despliegue de los componentes y subsistemas sobre los dispositivos físicos.

*Tabla B6. Diagramas*

**Reglas de UML y Mecanismos de comunes**

Las reglas especifican a que debe parecerse un modelo bien formado. Es decir aquel que es semánticamente auto consistente y está en armonía con todos sus modelos relacionados.

UML posee reglas semánticas para:

Nombres: Cómo llamar a los elementos, relaciones y diagramas.

Alcance : El contexto que da un significado específico a un nombre

Visibilidad: Como se pueden ver y utilizar esos nombres por otros.

Integridad : Como se relacionan apropiada y consistentemente unos elementos con otros.

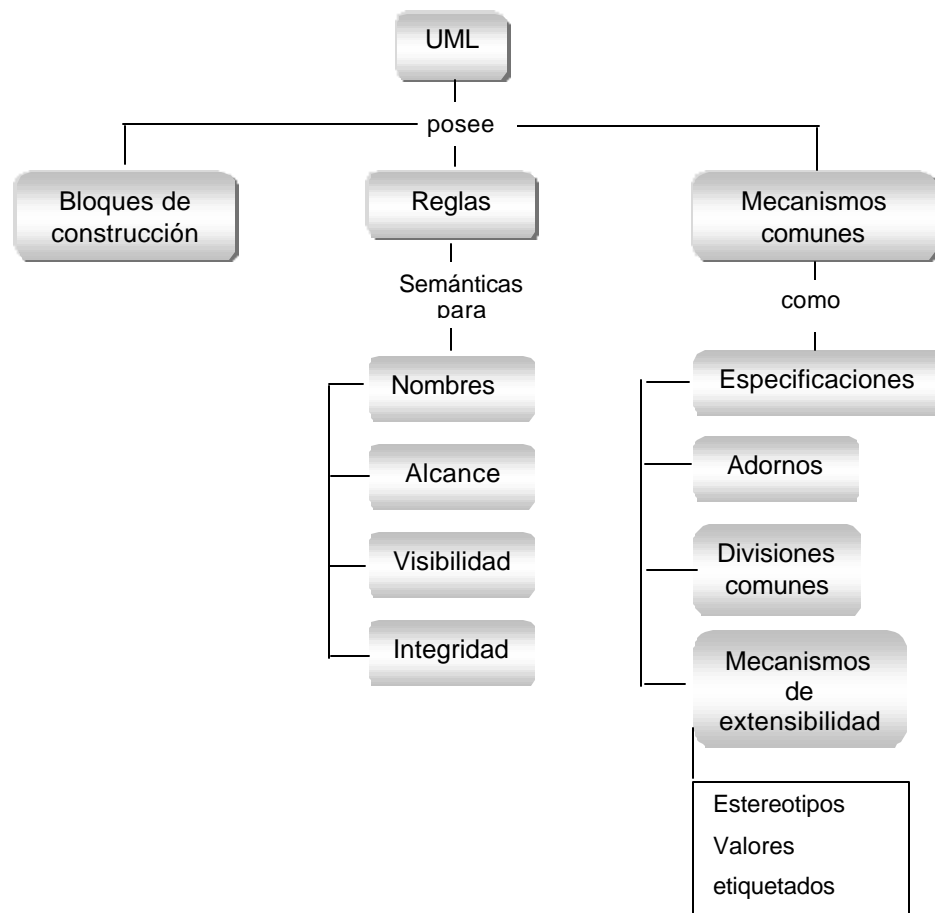


Figura B2. Reglas y mecanismos comunes

Ejecución : qué significa ejecutar o simular un modelo dinámico.

Cabe resaltar que no todos los modelos en UML cumplen con estas características pues debido a la revisión de los modelos por diferentes usuarios, y debido a las diferentes etapas es posible que se construyan además modelos:

Abreviados : ciertos elementos se ocultan para simplificar la vista.

Incompletos: Pueden estar ausentes ciertos elementos

Inconsistentes: No se garantiza la integridad del modelo.

Esto quiere decir que aunque las reglas de UML estimulan (pero no obligan) a considerar cuestiones más importantes de análisis, diseño e implementación que llevan a tales sistemas a convertirse en bien formados con el paso del tiempo.

Además en UML existen cuatro mecanismos para facilitar el entendimiento de los diagramas y se aplican consistentemente en todas las partes del lenguaje: especificaciones, adornos, divisiones comunes y mecanismos de extensibilidad (véase tabla B7.)

<b>MECANISMOS COMUNES EN UML</b>	
<b>MECANISMO</b>	<b>DESCRIPCION</b>
Especificaciones	Se utilizan para enunciar los detalles del sistema, con el fin de agregar una semántica a las especificaciones de los modelos de UML.
Adornos	Se refiere a los detalles adicionales de los gráficos, pues aunque los elementos comienzan con un símbolo básico, pueden añadirse a estos adornos específicos.
Divisiones Comunes	Hace referencia a la dicotomía existente entre los diferentes diagramas como por ejemplo la división entre clase y objeto, la separación entre interfaz e implementación (una interfaz declara un contrato, y una implementación representa una realización concreta de ese contrato), también los casos de uso y las colaboraciones que los realizan, así como operaciones y métodos que los implementan.
Mecanismos de extensibilidad	UML puede extenderse de manera controlada, para esto cuenta con tres mecanismos: estereotipos, valores etiquetados y restricciones, con el fin de configurar y extender UML a las necesidades de un proyecto, además de adaptarse a nuevas tecnologías de software, como la aparición de un lenguaje de programación distribuida más potente.

*Tabla B7. Mecanismos comunes de UML*

## **Diseño de Bases de Datos Relacionales - Una Extensión informal de UML**

### *Modelado de un esquema lógico de Base de datos*

Muchos de los sistemas que se modelen tendrán objetos persistentes, lo que significa que estos objetos podrán ser almacenados en una base de datos con el fin de poderlos recuperar posteriormente. La mayoría de las veces se empleará una base de datos relacional, una base de datos orientada a objetos o una base de datos híbrida objeto-relacional para almacenamiento persistente. UML es apropiado para modelar esquemas lógicos de bases de datos, así como bases de datos físicas.

Los diagramas de clases de UML son un superconjunto de los diagramas entidad-relación (E-R), una herramienta de modelado para el diseño lógico de bases de datos es utilizada con mucha frecuencia. Mientras los diagramas E-R clásicos se centran solo en los datos, los diagramas de clases van un paso mas allá, permitiendo el modelado del comportamiento. En la base de datos física, estas operaciones lógicas normalmente se convierten en disparadores (Triggers) o procedimientos almacenados.

Para modelar un esquema:

- ✓ Hay que identificar aquellas clases del modelado cuyo estado deben trascender el tiempo de vida de las aplicaciones.
- ✓ Hay que crear un diagrama de clases que contenga clases y marcarlas como persistentes (Un valor etiquetado estándar). Se puede definir un conjunto propio de valores etiquetados para cubrir detalles específicos de bases de datos.
- ✓ Hay que expandir detalles estructurales de estas clases. En general, esto significa especificar los detalles de sus atributos y centrar la atención en las asociaciones que estructuran estas clases y en sus cardinalidades.
- ✓ Hay que buscar patrones comunes que complican el diseño físico de bases de datos, tales como asociaciones cíclicas, asociaciones uno a uno y asociaciones n-arias. Donde sea necesario, se deben crear abstracciones intermedias para simplificar la estructura lógica.

- ✓ Hay que considerar también el comportamiento de las clases persistentes expandiendo las operaciones que sean importantes para el acceso a los datos y la integridad de los datos. En general, para proporcionar una mejor separación de intereses, las reglas del negocio relativas a la manipulación de conjuntos de estos objetos deberían encapsularse en una capa por encima de estas clases persistentes.
- ✓ Donde sea posible, hay que usar herramientas que ayuden a transformar un diseño lógico en un diseño físico.

#### *Modelado de una base de datos física*

Un esquema lógico de base de datos captura el vocabulario de los datos persistentes de un sistema, junto con la semántica de sus relaciones. Físicamente, estos elementos se almacenan en una base de datos para su posterior recuperación. La correspondencia entre un esquema lógico de bases de datos y una base de datos orientada a objetos es bastante directo, porque incluso las jerarquías de herencia complejas se pueden hacer persistentes directamente. Sin embargo la correspondencia de un esquema lógico de base de datos con una base de datos relacional no es tan simple.

#### *Correspondencia entre clases de objetos y tablas*

Toda clase corresponde con una o más tablas. (Además una tabla puede corresponderse con mas de una clase). Los objetos de una clase se pueden descomponer horizontal o verticalmente. Por ejemplo, si una clase tiene muchas instancias, y solo se hace referencia frecuente a unas cuantas, la descomposición horizontal puede mejorar la eficiencia al poner los objetos de acceso frecuente en una tabla, y los objetos restantes en otra tabla. Por supuesto, las aplicaciones no se beneficiaran de la descomposición horizontal a no ser que sepan en que tabla deben de buscar. De manera similar, si una clase tiene atributos con distintas tramas de acceso, la descomposición vertical de objetos puede servir de ayuda. La Figura B3 muestra las descomposiciones horizontal y vertical.

Partición horizontal

ID-persona	Nombre-persona	Dirección
1	Miguel Rodríguez	Casas, 14
5	Modesto Bravo	Cra 27- 42-23

ID-persona	Nombre-persona	Dirección
999	Jaime García	Calle 33-45-34

Partición Vertical

ID-persona	Nombre-persona
1	Miguel Rodríguez
5	Modesto Bravo
999	Jaime Gracia

ID-persona	Dirección
1	Casas, 14
5	Cra 27 – 42-23
999	Calle 33-45-34

*Figura B3. Descomposición vertical y horizontal de tablas*

En la Figura B4, una clase de objetos se transforma en una tabla. La clase Persona tiene los atributos nombre y dirección. El modelo de tablas enumera estos atributos, y añade la ID implícita del objeto. Añadimos detalles como parte de la formulación del modelo de tablas. Especificamos que la ID-persona no puede ser nula, puesto que es un candidato a clave. Decidimos que nombre persona no debe de ser nulo; es preciso introducir un nombre para cada persona, (nombre persona no es un candidato a clave porque puede haber dos personas que tengan el mismo nombre). El atributo dirección puede ser nulo. Asignamos un dominio a cada atributo, especificamos la clave primaria para cada tabla, y tomamos nota de los grupos de atributos a los que se acceda frecuentemente.

El código SQL crea la tabla Persona. (El estándar SQL actual admite la integridad referencial). El índice de Nombre-persona asegura una rápida recuperación para este atributo, puesto que se accede a él frecuentemente. El nivel SQL también hace corresponder los dominios con tipos de datos.

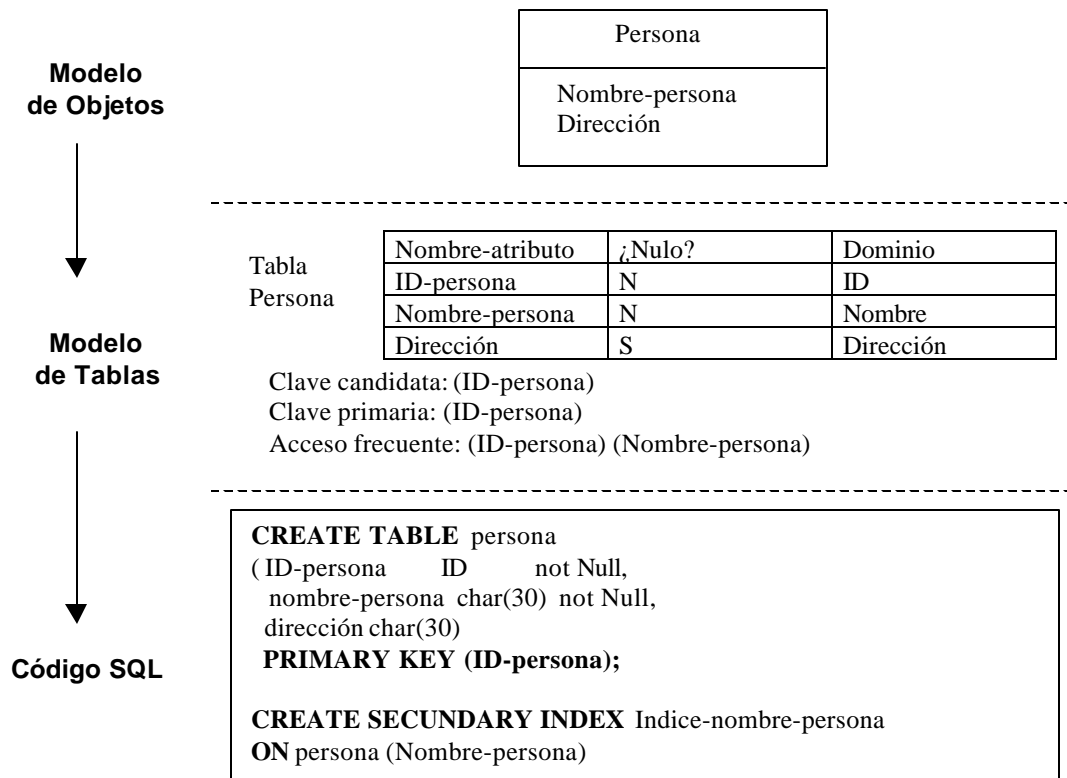


Figura B4. Correspondencia entre clase y tabla

#### Correspondencia entre asociaciones binarias y tablas

En general, una asociación puede o no corresponderse con una tabla. Depende del tipo y multiplicidad de la asociación, y de las preferencias de quien diseña la base de datos en lo tocante a extensibilidad, número de tablas, y criterios de rendimiento.

La Figura B5 muestra que una asociación muchos-a-muchos siempre se corresponde con una tabla concreta. Este esquema satisface la tercera forma normal.

Las claves primarias tanto para las clases relacionadas como para los atributos de enlace pasan a ser atributos de la tabla de asociación. Los atributos ID compañía e ID persona se combinan para formar un único candidato a clave para la tabla Posee

acciones. En general, una asociación se puede recorrer comenzando por cualquiera de las clases, así que tanto ID compañía como ID persona podrían ser de acceso frecuente. Las cláusulas de clave externa del código SQL indican que cada tupla de Posee-acciones debe de hacer referencia a una Compañía y a una Persona que hayan sido definidas en sus tablas respectivas.

Las tablas de asociación siempre dan a las claves externas procedentes de los objetos relacionados un valor no nulo; por definición, un enlace entre dos objetos requiere que ambos objetos sean conocidos. Si un par de objetos dados no posee un enlace, entonces se omite una entrada en la tabla de asociación.

Las Figuras B6, B7 y B8 muestran dos opciones para hacer corresponder con tablas una asociación uno-a-muchos. Se puede crear una tabla distinta para la asociación, o bien se puede incluir una clave externa en la tabla para la clase “muchos”.

No se muestra el código SQL para este ejemplo ni para otros, porque el código SQL es voluminoso y el modelo de tablas ya deja claro lo que se quiere decir. Las ventajas de fusionar una asociación en una clase son:

- ✓ Menos tablas
- ✓ Mayor rendimiento al tener que atravesar menos tablas

Las desventajas de fusionar una asociación en una clase son:

- ✓ Menos rigor de diseño. Las asociaciones se hacen entre objetos independientes de igual peso sintáctico. En general, parece impropio contaminar objetos con el conocimiento de otros objetos. Este aspecto está relacionado con el argumento a favor del encapsulamiento para los lenguajes orientados a objetos.
- ✓ Extensibilidad reducida. Es difícil calcular bien la multiplicidad en las primeras pasadas del diseño. Las asociaciones uno-a-uno y uno-a-muchos se pueden hacer externas. Las asociaciones muchos-a-muchos tienen que hacerse externas.

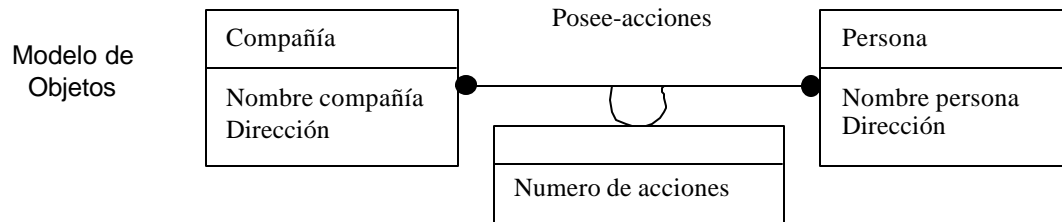


Tabla compañía (Similar a la tabla persona de la figura B4)

Tabla persona de la figura B4

Modelo de Tablas

Posee-acciones  
tabla

Nombre-atributo	¿Nulo?	Dominio
ID-compañía	N	ID
ID-persona	N	ID
Numero-de-acciones	S	Cuenta-acciones

Clave candidata: (ID-compañía, ID-persona)

Clave primaria: (ID-compañía, ID-persona)

Acceso frecuente: (ID-compañía) (ID-persona)

Código SQL

- Crear tabla e índices para compañía..

- Crear tabla e índices para persona..

**CREATE TABLA** Posee-acciones

(ID-compañía ID not Null,

ID-persona ID not Null,

Numero-de-acciones entero

**PRIMARY KEY** (ID-compañía, ID-persona)

**FOREIGN KEY** (ID-compañía) **REFERENCES** Compañía,

**FOREIGN KEY** (ID-persona) **REFERENCES** persona);

**CREATE SECONDARY INDEX** Indice-compañía-posee-acciones

**ON** posee-acciones (ID-compañía);

**CREATE SECONDARY INDEX** Indice-persona-posee-acciones

**ON** posee-acciones (ID-persona);

Figura B5. Correspondencia entre asociaciones muchos a muchos y tablas

- ✓ Mas complejidad. Una representación asimétrica de la asociación complica las búsquedas y las actualizaciones.

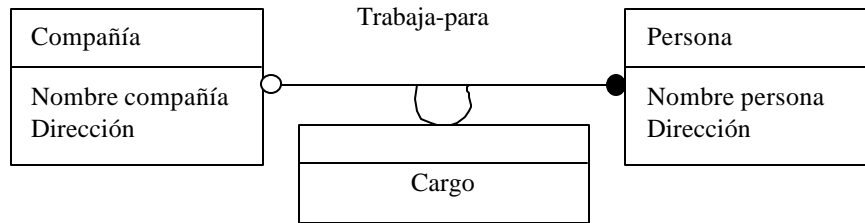


Figura B6. Modelo de objetos para una asociación uno a muchos.

Tabla compañía (similar a la tabla persona de la figura B4)  
 Tabla Persona de la figura B4

Tabla Trabaja-para

Nombre-atributo	¿Nulo?	Dominio
ID-compañía	N	ID
ID-persona	N	ID
Cargo	S	Cargo

Clave candidata: (ID-compañía)  
 Clave Primaria: (ID-persona)  
 Acceso frecuente: (ID-compañía) (ID-persona)

Figura B7. Modelo de objetos para una asociación uno a muchos - clave externa independiente

Tabla compañía (similar a la tabla persona de la figura B4)

Tabla Persona

Nombre-atributo	¿Nulo?	Dominio
ID-persona	N	ID
Nombre-persona	N	Nombre
Dirección	S	Dirección
ID-compañía	S	ID
Cargo	S	Cargo

Clave candidata: (ID-persona)  
 Clave Primaria: (ID-persona)  
 Acceso frecuente: (ID-persona) (Nombre-persona) (ID-compañía)

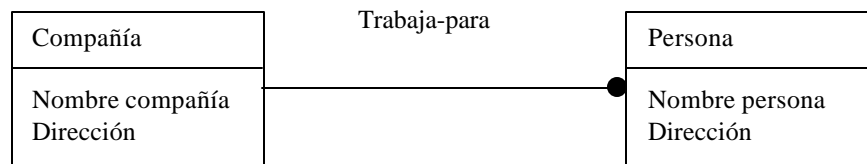
Figura B8. Modelo de objetos para una asociación uno a muchos – clave externa incluida

La decisión final de reducir o no una asociación en una clase relacionada depende de la aplicación.

También se puede reducir una asociación uno-a-uno en una tabla de objetos, o bien se puede incluso fundir y almacenar ambos objetos y la asociación en una sola tabla. Al fusionarse en una sola tabla se mejora el rendimiento, y se reduce el espacio de la base de datos a costa de menos extensibilidad, y de la posible violación de la tercera forma normal.

Obsérvese que para una asociación uno-a-muchos también se pueden plegar ambas clases y la asociación en una sola tabla. Sin embargo, esto no suele ser conveniente, y puede violar la segunda forma normal.

El símbolo de multiplicidad en forma de disco blanco, en la Figura B6, indica que no es necesario que una persona tenga un empleo. Aquí, los objetos del tipo Persona saben de los objetos del tipo Compañía, puesto que todo objeto del tipo Persona requiere un objeto del tipo Compañía. Resulta menos beneficioso tener una tabla de asociación independiente. El único cambio en el modelo de tablas para el caso de una clave externa incluida es que la ID compañía no puede ser nula.



*Figura B9. Modelo de objetos para una asociación con dependencia de existencia*  
Correspondencia entre asociaciones ternarias y tablas

La Figura B10 muestra la implementación para una asociación ternaria. Obsérvese que se muestra una tabla para cada una de las clases que participan en la asociación ternaria, incluso para una clase que pueda ser trivial, tal como Año. Aquí, si la tabla Ano se limita a emparejar ID-año con año, se podría efectuar una

optimización secundaria: descartar la tabla Ano y sustituir ID-ano por año en la tabla ternaria.

La Figura B10 no muestra dependencias de existencia. Esta certeza parece tener sentido en el ejemplo del lanzador. Un lanzador, un equipo y un año tienen una existencia independiente, mas allá de sí el lanzador juega en un equipo dado y en una temporada concreta. Para una asociación ternaria, en la cual es importante una dependencia entre una clase de objetos y la asociación ternaria, lo mejor será, posiblemente, promover la asociación ternaria a una clase de y manejar la dependencia tal como se hace en la Figura B9. La promoción de la asociación ternaria a clase le da a cada instancia ternaria una ID a la cual puede entonces hacer referencia el objeto dependiente.

Obsérvese que la tabla Persona hace alusión a ID-persona mientras que la tabla ternaria hace alusión a ID-lanzador. La tabla ternaria refleja que una persona asume el papel de lanzador. ID-lanzador es un nombre de atributo más significativa que ID-persona para la tabla ternaria. El modelo de objetos muestra que estos dos nombres hacen alusión a la misma entidad, y sería esperable que compartieran una misma vía de navegación. El nombre de rol de una clase debería de ser utilizado para los atributos que estén incluidos en una tabla de asociación. La Figura B10 muestra la forma de manejar los roles mediante código SQL. (Podríamos haber utilizado roles en algunos de nuestros ejemplos anteriores. Por ejemplo, en la Figura B5 la persona asume el papel de accionista. En las Figuras B6 y B9, la compañía es el empresario y una persona es el empleado).

Las Figuras B11 y B12 describen otra estructura mas de asociación: la cualificación. Una compañía tiene muchas personas trabajando como directores. La mayoría de los cargos de la compañía son ocupados por una persona, como puedan ser el Presidente o el Tesorero, Unos pocos cargos, tales como los miembros del consejo de dirección, pueden ser ocupados por muchas personas. Una persona puede tener muchas combinaciones de puestos en la varias empresas. Por ejemplo, una persona puede tener cargos en mas de una compañía, o puede desempeñar múltiples funciones dentro de la misma compañía.

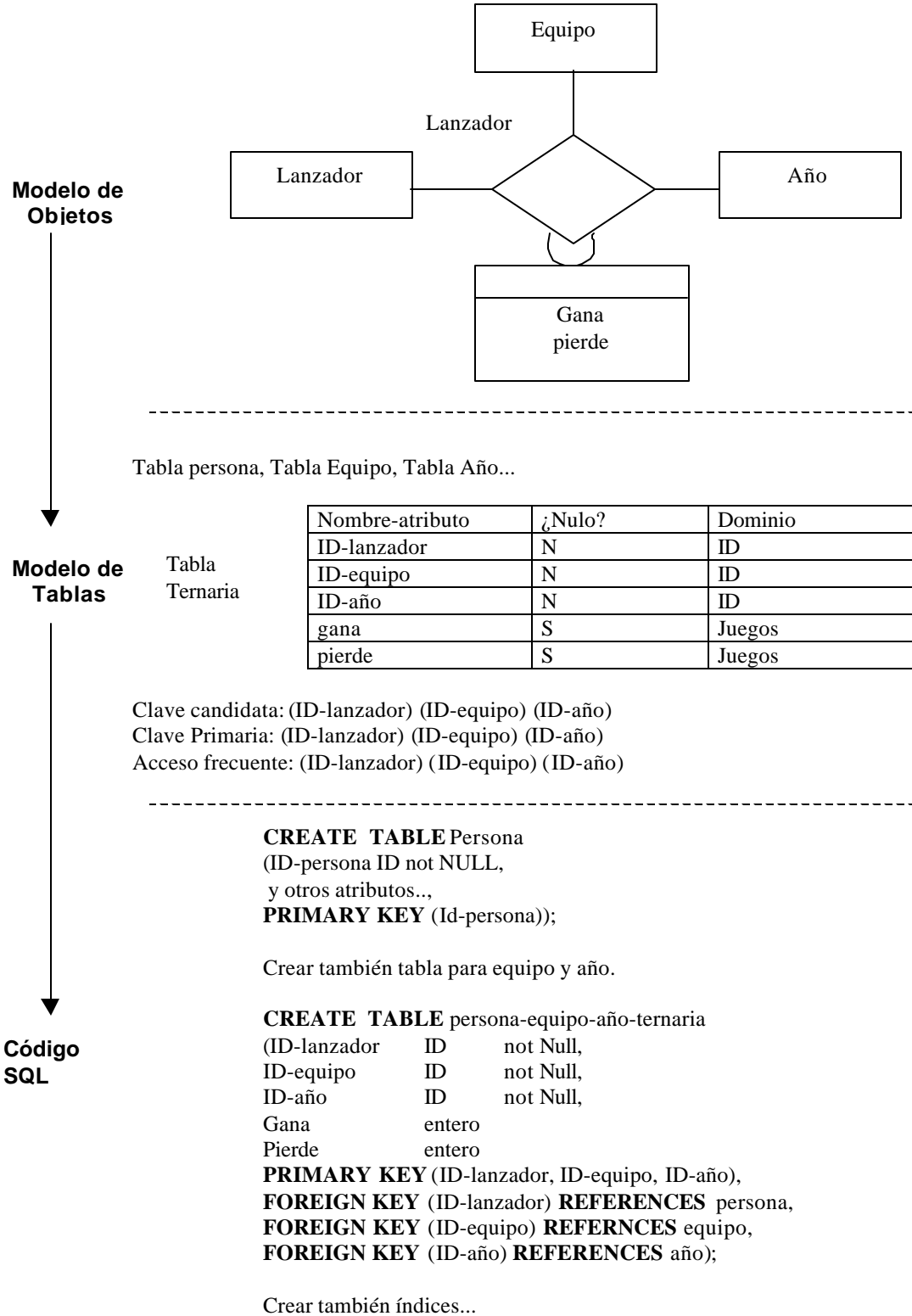


Figura B10. Correspondencia entre asociaciones ternarias y tablas

Las tablas Compañía y Persona son similares a los ejemplos anteriores. La tabla de asociación es más interesante. La tabla de asociación tiene tres atributos: la clave primaria para Compañía, la clave primaria para Persona y el calificativo cargo. Ninguno de estos atributos puede ser nulo, puesto que son todos ellos parte esencial de la asociación cualificada. Los tres atributos deben de aparecer en la clave primaria y en el candidato a clave puesto que después de la cualificación, la asociación es muchos-a-muchos.

No se han discutido explícitamente las agregaciones en ninguno de los ejemplos de bases de datos. El diseño de tablas de agregación sigue las mismas reglas de correspondencia que las asociaciones.

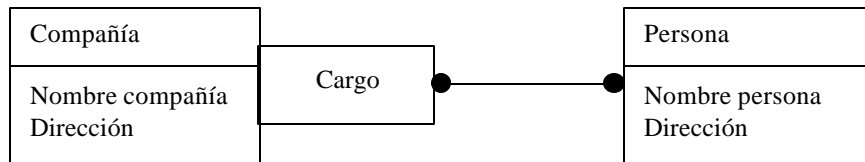


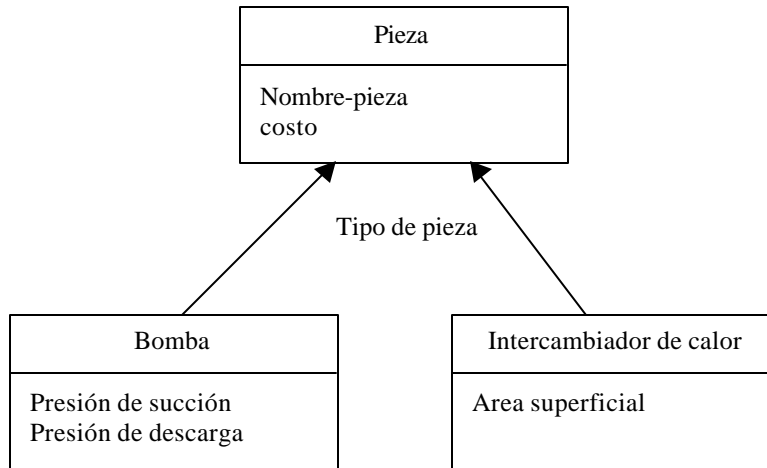
Figura B11. Modelo de objetos para una asociación cualificada.

Tabla compañía (similar a la tabla persona de la figura B4)			
Tabla Persona de la figura B4			
Tabla de asociación cualificada	Nombre-atributo	¿Nulo?	Dominio
	ID-compañía	N	ID
	ID-persona	N	ID
	Cargo	N	Nombre-Cargo
Clave candidata: (ID-compañía, ID-persona, cargo)			
Clave Primaria: (ID-compañía, ID-persona, cargo)			
Acceso frecuente: (ID-persona) (ID-compañía, cargo)			

Figura B12. Modelo de tablas para una asociación cualificada.

Correspondencia de las generalizaciones con tablas

La correspondencia entre generalizaciones y tablas se puede hacer tomando cuatro enfoques. Se Utilizará la Figura B13 como punto de partida para explorar estas estrategias. Se comenzará por describir la herencia simple:



*Figura B13. Modelo de Objetos para la generalización.*

La Figura B14 muestra el enfoque normal. Las subclases y superclases se corresponden con una tabla. La identidad del objeto a través de una generalización se mantiene mediante el uso de una ID compartida. De esta forma, el producto bomba puede tener una fila en la tabla Pieza con la ID 101, y otra fila en la tabla Bomba, también con ID 101. Este enfoque es lógicamente limpio y extensible, Sin embargo, implica muchas tablas, y la navegación de superclases a subclases puede resultar lenta. Se podrían recorrer las tablas en la forma siguiente:

1. El usuario proporciona un nombre de pieza.
2. Se busca la fila de Pieza que corresponde al nombre de pieza.
3. Se recuperan la ID pieza y el tipo pieza para esta fila.

4. Se va a la tabla de subclases indicada por tipo pieza, y se busca la fila de subclase cuya ID sea la misma que la fila de Pieza.

Tabla pieza	Nombre-atributo	¿Nulo?	Dominio
	ID-pieza	N	ID
	Nombre-pieza	N	Nombre
	Costo	S	Dinero
	Tipo -pieza	N	Tipo-equipo
Clave candidata: (ID-pieza) (Nombre-pieza)			
Clave Primaria: (ID-pieza)			
Acceso frecuente: (ID-pieza ) (Nombre-pieza)			
Tabla Bomba	Nombre-atributo	¿Nulo?	Dominio
	ID-pieza	N	ID
	Presión-succión	S	presión
	Presión-descarga	S	presión
Clave candidata: (ID-pieza)			
Clave Primaria: (ID-pieza)			
Acceso frecuente: (ID-pieza )			
Tabla Intercambiador de calor	Nombre-atributo	¿Nulo?	Dominio
	ID-pieza	N	ID
	Area-superficial	S	area
Clave candidata: (ID-pieza)			
Clave Primaria: (ID-pieza)			
Acceso frecuente: (ID-equipo)			

*Figura B14. Modelo de tablas para una generalización: tablas de superclases y subclases*

Por ejemplo, el usuario puede especificar como nombre de pieza "producto bomba". La aplicación busca en la tabla de piezas y encuentra que el producto bomba tiene la ID 101 y el tipo de pieza bomba. Entonces la aplicación examina la tabla Bomba y recupera datos adicionales para la ID 101.

La Figura B15 muestra el código SQL para la Figura B14. Obsérvese que SQL no puede imponer la descomposición indicada por el discriminador de generalización. Hay que almacenar producto bomba tanto en la tabla Bomba como en la tabla Intercambiador de calor, y SQL lo admitirá. El problema básico es que SQL tiene

poco apoyo para las restricciones de integridad. Seria necesario escribir un código especial de aplicación para hacer obligatoria la descomposición de generalización, y seria preciso utilizar los consentimientos de SQL para bloquear el acceso interactivo o el acceso a través de otros programas.

```

CREATE TABLE pieza
(ID-pieza          ID          not Null,
Nombre-pieza      Char(30)   not Null,
Costo             dinero
Tipo-pieza        Char(30)   not Null,
PRIMARY KEY (ID-pieza));

CREATE SECONDARY INDEX Indice-nombre-pieza
ON pieza (nombre-pieza);

CREATE TABLE Bomba
(ID-pieza          ID          not Null,
presion-succion   real
presion-descarga  real
PRIMARY KEY (ID-pieza),
FOREIGN KEY (ID-pieza) REFERENCES pieza);

Crear tabla para intercambiador de calor (similar a la de Bomba)

```

*Figura B15. Código SQL para la generalización – tablas de superclases y subclases.*

Las Figuras B16 y B17 son enfoques alternativos de correspondencia. Están motivados por el deseo de eliminar la navegación de superclases a subclases, y acelerar así el proceso. Sin embargo, la mejora de rendimiento tiene su costo.

La Figura B17 ilustra el enfoque de "muchas subclases". Este enfoque elimina la tabla de superclases y duplica todos los atributos de la superclase en cada tabla de subclase. Se podría utilizar este enfoque si una subclase tuviera muchos atributos, si la superclase tuviera pocos atributos, y si la aplicación supiera que subclase había que examinar. La Figura B17 mantiene la tercera forma normal, pero resulta menos satisfactoria que el enfoque estándar. No se puede hacer obligatoria la unicidad de nombres de piezas a través de las tablas de subclases, porque los RDBMS no ofrecen índices que abarquen varias tablas.

Nombre-atributo	¿Nulo?	Dominio
ID-pieza	N	ID
Nombre-pieza	N	Nombre
costo	S	dinero
Presión-succión	S	presión
Presión-descarga	S	presión

Tabla Bomba

Clave candidata: (ID-pieza) (Nombre-pieza)  
Clave Primaria: (ID-pieza)  
Acceso frecuente: (ID-pieza) (Nombre-pieza)

Nombre-atributo	¿Nulo?	Dominio
ID-pieza	N	ID
Nombre-pieza	N	nombre
costo	S	dinero
Area-superficial	S	área

Tabla Itercambiador de calor

Clave candidata: (ID-pieza) (Nombre-pieza)  
Clave Primaria: (ID-pieza)  
Acceso frecuente: (ID-equipo) (Nombre-pieza)

Figura B16. Modelo de tablas para la generalización: muchas tablas de subclase.

Nombre-atributo	¿Nulo?	Dominio
ID-pieza	N	ID
Nombre-pieza	N	Nombre
costo	S	dinero
Tipo-pieza	N	Tipo-equipo
Presion-succion	S	presión
Presion-descarga	S	presión
Area-superficial	S	área

Tabla pieza

Clave candidata: (ID-pieza) (Nombre-pieza)  
Clave Primaria: (ID-pieza)  
Acceso frecuente: (ID-pieza) (Nombre-pieza)

Figura B17. Modelo de tablas para la generalización: una tabla de superclase.

El enfoque de "una tabla de superclases", mostrado en la Figura B17, eleva todos los atributos de las subclases al nivel de la superclase. Cada registro de la tabla de superclase utiliza atributos pertinentes a una subclase; los valores de los demás atributos son nulos. La tabla de esta figura viola la tercera forma normal; la clave

primaria es ID-pieza o bien nombre-pieza, pero los valores de los atributos dependen también de tipo equipo. Este enfoque puede ser útil si solo hay dos o tres subclases con unos pocos atributos.

La mejor manera de manejar las relaciones de generalización que presentan herencia múltiple de clases disjuntas es utilizar el enfoque estándar que se muestra en la Figura B14 una tabla por superclase y una tabla por subclase. La mejor manera de manejar la herencia múltiple de clases que se solapan es utilizar una tabla para la superclase, una tabla para cada subclase, y una tabla para la relación de generalización (el cuarto enfoque para hacer corresponder las generalizaciones con tablas). La herencia múltiple se produce con poca frecuencia, así que no suele merecer la pena el esfuerzo de intentar optimizar la correspondencia.

## ANEXO A.

### **PLANTILLAS PARA DESCRIPCION DE PROCESOS DEL NEGOCIO Y CASOS DE USO**

Plantilla para descripción de procesos del negocio.

<b>Proceso del Negocio</b>	Nombre del proceso
<b>Objetivo</b>	Descripción informal de los objetivos del proceso
<b>Descripción</b>	Breve descripción de lo que se hace en el proceso
<b>Prioridad</b>	El proceso puede ser crítico, importante o secundario
<b>Riegos</b>	Nivel de conflictividad para efectuar el proceso.
<b>Posibilidades</b>	Se refiere a la manera de dar soporte al proceso de negocio mediante el software.
<b>Tiempo de ejecución</b>	Cantidad de tiempo estimada para la ejecución del proceso.
<b>Costo de ejecución</b>	Cantidad de dinero estimado para la ejecución del proceso

Plantilla para descripción de casos de uso.

<b>Caso de uso:</b> Nombre del caso de uso
<b>Objetivo:</b> Descripción informal de los objetivos del caso de uso
<b>Actores:</b> Actores que intervienen en el caso de uso: principales y secundarios
<b>Precondiciones:</b> condiciones que deben cumplirse para que pueda realizarse el caso de uso
<b>Pasos:</b> Secuencia de pasos necesarios para que el caso de uso se desarrolle con éxito. Debemos mostrar las interacciones de los actores y las acciones del sistema.
<b>Variaciones:</b> Variaciones en la secuencia de pasos.
<b>Extensiones:</b> Puntos de extensión del caso de uso.
<b>Cuestiones:</b> Cuestiones planteadas durante la especificación del caso de uso

Este documento se elaboró con el fin de ofrecer al usuario de la herramienta software “Audit casos” una guía fácil para el manejo y comprensión de sus características. El documento está integrado en la herramienta y es de fácil acceso a través del menú de ayuda.

✓ Ingreso a la herramienta

Para el ingreso a la herramienta software el usuario debe registrarse; para realizar este proceso presione el botón **Registrarse**.

Una vez registrado, en **Rol usuario**, seleccione el rol a desempeñar dentro de la herramienta, en **Usuario** digite el nombre de usuario, y en **Contraseña** digite la clave personal y presione el botón **Ingresar** o presione enter.

Si los datos son correctos se dará paso al usuario y se mostrará la interfaz correspondiente al rol desempeñado, en caso contrario se mostrará un mensaje de error. Cuando el rol desempeñado es **Auditor** o **revisor** se visualizará una vista con barra de menu principal, barra de herramienta y barra de estado. Si por el contrario el rol desempeñado es **Aprendiz** se mostrará una vista con pestañas identificadas como: Esquema, Información e historial.

✓ Menu Principal

El menú principal presenta un conjunto de menús y submenús de opciones que constituyen accesos directos, facilitando al usuario la interacción con la herramienta.



*Menú Principal de la herramienta.*

En el Menú Archivo se manejan las opciones relacionadas con la gestión de un caso de auditoría, el submenú Nuevo con sus opciones Nuevo caso y definir plan, Abrir caso, Imprimir, presentación preliminar, configurar impresora, cerrar caso.

En el menú ver se presentan las opciones ver barra de herramienta y ver barra de estado. En el menú detalles lo relacionado con los detalles del caso y del usuario actual, se muestran las opciones, detalles del caso, detalles del usuario, opciones del caso.











En el menú examen lo correspondiente al trabajo de auditoría, representado en la hoja de trabajo, identificación de riesgos y controles, revisión y observación de caso. En el menú Informe lo relacionado con la gestión de informes las opciones crear informe y ver informe. Y finalmente el menú Ayuda que corresponde a los temas de ayuda de la herramienta.

✓ Barra de herramientas

La barra de herramienta proporciona una serie de iconos con los accesos a las opciones más importantes y que se van a describir con mayor detalle.



*Barra de herramientas de aplicación*

ICONO	DESCRIPCION DEL COMANDO
	Use este icono para crear un caso de auditoría. Se mostrará el dialogo de creación de caso nuevo en donde se piden los datos de tipo de caso, datos del auditado, estructura del auditado, opciones de parametrización del caso.
	Use este icono para ver la información disponible de un caso de auditoría abierto. Se mostrará un dialogo con la información del caso y datos del auditado.
	Use este icono para abrir un caso de auditoría existente. Se mostrará un dialogo para selección de un caso de auditoría.
	Use este icono para visualizar o cambiar los parámetros establecidos del caso actual.
	Use este icono para crear una hoja de trabajo, se registrarán los datos del examen realizado al auditado.
	Este icono permite crear un plan de auditoría o abrir un plan de auditoría existente. Se mostrará un dialogo con las opciones de creación de técnicas, actividades, personal y recursos.
	Use icono para crear auditores o asignar revisores al caso actual.
	Este icono permite la identificación de riesgos y controles para el caso actual de auditoría.
	Use este icono para modificar los datos de registro del usuario actual.
	Este icono presenta información de ayuda para el manejo de comandos y opciones de la herramienta.

- ✓ Estudiar caso de auditoría.

Según lo mencionado anteriormente para estudiar un caso de auditoría el usuario debe identificarse como aprendiz. Se presentará una vista con las pestañas antes mencionadas.



*Pestañas presentadas en la vista.*

En la pestaña esquema se mostrará un gráfico con la estructura de auditoría empleada: Caso de auditoría, planeación, evaluación e informe y sobre el cual el aprendiz puede visualizar los datos más importantes del caso.

ICONO DEL ESQUEMA	DETALLES DEL COMANDO
	Use este icono para mostrar el dialogo abrir caso y seleccionar el caso de auditoría a estudiar.
	Use este icono para visualizar los datos concernientes al plan de auditoría ejecutado, se mostrará las actividades, técnicas y recursos empleados.
	Use este icono para visualizar los detalles del examen realizado para el caso de auditoría.
	Use este icono para visualizar los detalles del informe generado por el auditor.

En la hoja de información permite el registro de datos como aportes y o notas para el caso estudiado, preguntas que se hacen, referencias bibliográficas, y los contactos de personas con conocimiento sobre el caso o tema de auditoría en particular. Finalmente en la hoja Historial se presenta una lista con los últimos casos estudiados.

Si desea mas información puede remitirse a la ayuda de la herramienta software.