UNIVERSIDAD TECNICA DEL NORTE

FACULTAD DE INGENIERIA EN CIENCIAS APLICADAS

ESCUELA DE INGENIERIA EN SISTEMAS COMPUTACIONALES

Proyecto previo aprobación para la obtención del título de

INGENIERO EN SISTEMAS COMPUTACIONALES

PROYECTO:

"SABDO: SISTEMA DE AUDITORIA PARA BASES DE DATOS ORACLE"

Autores: Ondyna Lilián Fierro Montenegro

Juan Carlos García Pinchao

Director: Ing. Jorge Caraguay

Ibarra – Ecuador

Abril 2009

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR

El presente proyecto de tesis es una herramienta útil y eficaz para realizar la auditoría de la base de datos Oracle de la Universidad Técnica del Norte.

Se diseñó acorde a la metodología RUP y utilizando las herramientas existentes en la Dirección de Informática de la institución, cumpliendo con el objetivo de dar la información requerida para una auditoría de sistema y de datos.

El uso de este proyecto será exclusivamente para la Universidad Técnica del Norte, por tanto nosotros: Lilián Fierro Montenegro y Juan Carlos García Pinchao cedemos los derechos de operación, manipulación del sistema y códigos fuente. El sistema permanecerá a cargo del personal de desarrollo del Departamento de Informática de la institución.

Los autores anulamos cualquier compromiso de soporte técnico y mantenimiento una vez culminado el proyecto.

Los Autores

DEDICATORIA

A mis padres y mis hijas:

Porque la vida continúa y las oportunidades se presentan

para que surja lo mejor de nosotros y así llegar a cumplir nuestros sueños.

Lilián Fierro Montenegro.

A mi madre:

Tu amor y dedicación han sido el incentivo para culminar un proceso

que ha sido tu mayor anhelo y orgullo.

Juan Carlos García.

AGRADECIMIENTO

Hace muchos años emprendimos un plan de vida, estudiar para obtener un título universitario que nos permitiera alcanzar nuestros anhelos, al terminar los estudios comenzamos una nueva vida, pero siempre existió un detalle inconcluso. A través del tiempo, tuvimos el apoyo de familiares, maestros y amigos, que esperaban el momento en que culmináramos el proyecto que nos permitiese obtener el título de Ingenieros en Sistemas. Ahora que este sueño se ha materializado nos permitimos dar un profundo agradecimiento a todas estas personas que no podemos nombrarlas una a una pero están en nuestro pensamiento y corazón.

Los Autores.

TABLA DE CONTENIDOS

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR ..... **I**

DEDICATORIA ..... **II**

AGRADECIMIENTO ..... **III**

RESUMEN ..... **VIII**

SUMARY ..... **VIII**

INTRODUCCIÓN **2**

BENEFICIOS **3**

CAPITULO I **6**

1.1 INTRODUCCIÓN 6
1.2 POSICIONAMIENTO 7
1.3 DESCRIPCIÓN DE STAKEHOLDERS (PARTICIPANTES EN EL PROYECTO) Y USUARIOS 9
1.4 DESCRIPCIÓN GLOBAL DEL PRODUCTO 11
1.5 DESCRIPCIÓN GLOBAL DEL PRODUCTO 13
1.6 RESTRICCIONES 14
1.7 OTROS REQUISITOS DEL PRODUCTO 15

CAPÍTULO II **17**

2 PLAN DE DESARROLLO DE SOFTWARE **17**

2.1 INTRODUCCIÓN 17
2.2 VISTA GENERAL DEL PROYECTO 19
2.3 ORGANIZACIÓN DEL PROYECTO 25
2.4 GESTIÓN DEL PROCESO 27
2.5 SEGUIMIENTO Y CONTROL DEL PROYECTO 33

CAPÍTULO III **35**

3 METODOLOGÍAS PARA LA AUDITORÍA DE BASE DE DATOS. **35**

3.1 INTRODUCCIÓN 35
3.2 OPORTUNIDAD DEL NEGOCIO 37
3.3 POSICIONAMIENTO DEL NEGOCIO 44

CAPÍTULO IV **51**

4 ESPECIFICACIÓN DE CASOS DE USO **51**

4.1 CASO DE USO: INICIALIZAR AUDITORÍA DE BDD ORACLE 51
4.2 CASO DE USO: INGRESO AL SISTEMA 52
4.3 CASO DE USO: CONFIGURACIÓN DE LA AUDITORÍA DE SISTEMA. 54
4.4 CASO DE USO: MONITOREO DE LA AUDITORÍA DE SISTEMA 56
4.5 CASO DE USO: CONFIGURACIÓN DE LA AUDITORÍA DE DATOS 58
4.6 CASO DE USO: MONITOREO DE DATOS 59

CAPÍTULO V **62**

5 ESPECIFICACIÓN DE CASOS DE PRUEBAS **62**

5.1 CASO DE PRUEBA: INICIALIZAR AUDITORÍA DE BDD ORACLE 62
5.2 CASO DE PRUEBA: CONFIGURAR AUDITORÍA DE SISTEMA. 63
5.3 CASO DE PRUEBA: MONITOREAR LA AUDITORÍA DE SISTEMA. 65
5.4 CASO DE PRUEBA: CONFIGURAR AUDITORÍA DE DATOS. 68

5.5 CASO DE PRUEBA: MONITOREAR LA AUDITORÍA DE DATOS. 69

CAPÍTULO VI 73

6 LISTA DE RIESGOS 73

6.1 INTRODUCCIÓN 73

CAPÍTULO VII 80

7 CONCLUSIONES Y RECOMENDACIONES 80

7.1 CONCLUSIONES 80
7.2 RECOMENDACIONES 81

CAPÍTULO VIII 83

8 GLOSARIO 83

8.1 INTRODUCCIÓN 83
8.2 ORGANIZACIÓN DEL GLOSARIO 83

CAPÍTULO IX 87

ANEXO A: DICCIONARIO DE DATOS 89

A.1 DBA_AUDIT_EXISTS 89
A.2 DBA_AUDIT_OBJECT 92
A.3 DBA_AUDIT_SESSION 95
A.4 DBA_AUDIT_OBJECT 97
A.5 DBA_AUDIT_TRAIL 100
A.6 SENTENCIAS DE CREACIÓN DE TABLAS Y VISTAS. 104

ANEXO B: GUIA DE PROGRAMACIÓN 119

B.1 DOCUMENTACIÓN Y COMENTARIOS EN EL CÓDIGO 119
B.2 PALABRAS RESERVADAS DEL LENGUAJE DE PROGRAMACIÓN 119

ANEXO C: PROTOTIPO DE INTERFAZ DE USUARIO 121

C.1 INTRODUCCION 121
C.2 ARCHIVOS DE CONFIGURACIÓN 122
C.3 DISEÑO DE LA PLANTILLA ESTÁNDAR. 127
C.4 FUNCIONES Y PROCEDIMIENTOS PARA LA EJECUCIÓN DE LOS PROCESOS BÁSICOS DE LA PLANTILLA ESTÁNDAR 128

ANEXO D: MANUAL DE INSTALACIÓN 144

D.1 INSTALACIÓN DE HERRAMIENTAS SOBRE LINUX 144
D.2 ORACLE 10G DATABASE SERVER 144
D.3 ORACLE 10G DEVELOPER SUITE 147
D.4 ORACLE 10G APPLICATION SERVER 149

ANEXO E: MANUAL DE USUARIO 154

E.1 INGRESO AL SISTEMA 154
E.2 MANEJO DEL SISTEMA SABDO 156

INDICE DE FIGURAS

Ilustración 0-1 Fases del Proyecto 30
Ilustración 0-2 Entorno de base de datos. 38
Ilustración 0-3 Auditoría de BDD 51
Ilustración 0-4 C.U. Ingreso al sistema 52
Ilustración 0-5 C.U. Configuración de auditoría de sistema 54
Ilustración 0-6 C.U. Monitoreo de auditoría de sistema 56
Ilustración 0-7 C.U. Configuración de auditoría de datos 58
Ilustración 0-8 C.U. Configuración de auditoría de datos 59
Ilustración 0-9 Menú de seguridad y Auditoría 127
Ilustración 0-10 Plantilla de formulario 127
Ilustración 0-11 Ingreso al sistema 154
Ilustración 0-12 Pantalla principal 155
Ilustración 0-13 Menu de Seguridad y Auditoría 155
Ilustración 0-14 Menú Principal SABDO 156
Ilustración 0-15 Menú Auditoría de Sistema 157
Ilustración 0-16 Menú Configuración de Sesiones 158
Ilustración 0-17 Opciones de Conexión 159
Ilustración 0-18 Configuración de Acciones 159
Ilustración 0-19 Menú Configuración de Acciones 160
Ilustración 0-20 Configuración de Objetos 161
Ilustración 0-21 Menú Configuración de Objetos 161
Ilustración 0-22 Monitoreo de Sesiones 162
Ilustración 0-23 Ventana de Monitoreo de Sesiones 1 163
Ilustración 0-24 Visualización de Monitoreo de Sesiones 2 164
Ilustración 0-25 Visualización de Monitoreo de Sesiones 3 164
Ilustración 0-26 Visualización de Monitoreo de Sesiones 4 165
Ilustración 0-27 Monitoreo de Objetos/Acciones 165
Ilustración 0-28 Monitoreo de Objetos Acciones 1 166
Ilustración 0-29 Monitoreo de Objetos Acciones 2 167
Ilustración 0-30 Monitoreo de Objetos Acciones 3 168
Ilustración 0-31 Monitoreo de Objetos Acciones 4 168
Ilustración 0-32 Monitoreo de Objetos Acciones 5 169
Ilustración 0-33 Monitoreo de Objetos Acciones 6 169
Ilustración 0-34 Monitoreo de Objetos Acciones 7 170
Ilustración 0-35 Menú Auditoría de Datos 171
Ilustración 0-36 Configuración de Tablas 1 para Auditoría de Datos 171
Ilustración 0-37 Configuración de Tablas 2 para Auditoría de Datos 172
Ilustración 0-38 Monitoreo de Datos 1 173
Ilustración 0-39 Monitoreo de Datos 2 173
Ilustración 0-40 Monitoreo de Datos 3 174
Ilustración 0-41 Monitoreo de Datos 4 174
Ilustración 0-42 Monitoreo de Datos 5 175
Ilustración 0-43 Monitoreo de Datos 6 175
Ilustración 0-44 Monitoreo de Datos 7 176
Ilustración 0-45 Menú de Reportes 176

INDICE DE TABLAS

Tabla 1 Resumen de Stakeholders 10
Tabla 2 Resumen de usuarios 11
Tabla 3 Beneficios del Sistema 12
Tabla 4 Costo y Precio 13
Tabla 5 Atributos de Características 16
Tabla 6 Fases del proyecto 27
Tabla 7 Actividades fase 1 31
Tabla 8 Actividades Fase 2 32
Tabla 9 Parámetros Audit Exists 91
Tabla 10 Parámetros Audit Object 94
Tabla 11 Parámetros Audit Session 96
Tabla 12 Parámetros Audit Statement 99
Tabla 13 Parámetros Audit Trail 103

RESUMEN

Las nuevas tecnologías y sistemas cada vez más complejos requieren interfaces gráficas y amigables al usuario.

Una base de datos maneja gran cantidad de datos de una empresa o institución, estos son de gran importancia para asegurar su desempeño y competitividad. La auditoría de la base de datos permite aumentar el nivel de seguridad, dando a conocer irregularidades en su manejo.

El sistema de Auditoría de Base de Datos Oracle – SABDO, es una interfaz gráfica amigable, fácil de configurar para cubrir los requerimientos, de información de auditoría, del usuario. Permite realizar el rastreo de intrusos, cambios no autorizados en los datos y objetos de la base de datos, visualizando su ubicación, nombre de usuario, fecha, hora, etc.

SABDO ha sido integrado al sistema académico de la Universidad Técnica del Norte con el objetivo de colaborar en la tarea de mantener la seguridad de la información contenida en su base de datos.

SUMARY

The new technology and systems every time to many complex require interface graph and friendly for the user.

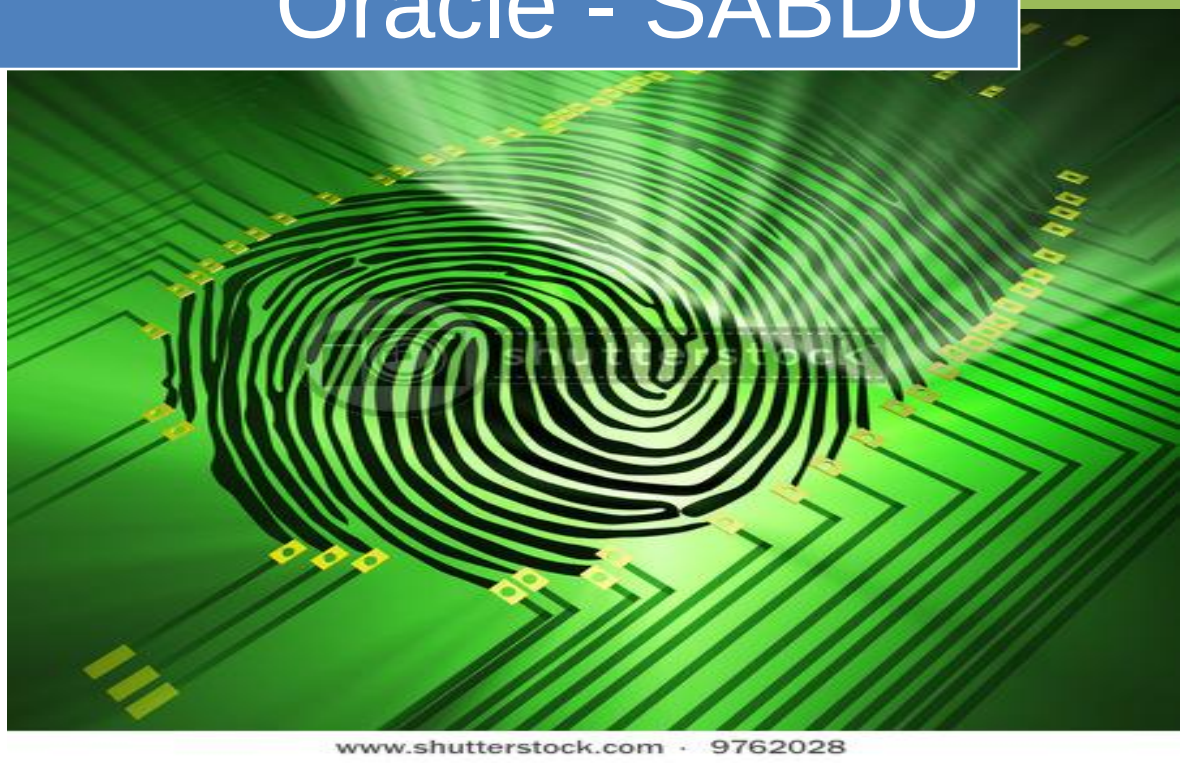
One Database operate a great quantity of data of an company or institution, this is of grand importance for assure his performance and competitiveness. The audit of the database permit increase the security level, giving at know irregularity in hers management.

The Audit of Oracle Database System - AODS, is one interface graph and friendly, easy of shape for cover the requeriments, of information of the audit, of the user.

AODS has been integrated of the academic system of the Technical University of the North with the object of collaborate in the task of maintain the security of the information contained in his database.

INTRODUCCION

Auditoría para base de datos Oracle - SABDO



INTRODUCCIÓN

La mayoría de Empresas que almacenan su información en bases de datos Oracle no disponen de herramientas de software que les proporcionen la posibilidad de detectar si se han realizado accesos autorizados pero no adecuados a sus datos, quienes lo han hecho y que información se ha comprometido.

La necesidad de las empresas de manejar grandes volúmenes de información de una manera rápida y segura han obligado a los Gerentes a recurrir a soluciones informáticas muy costosas, pero necesarias. Los sistemas implementados con Bases de Datos Oracle disponen de diversas herramientas que permiten el manejo eficaz de los datos importantes de una empresa, pero cuando se trata de auditoría de datos las herramientas degradan su rendimiento.

Al mirar esta perspectiva, los gerentes generales, gerentes de sistemas, DBAs, Desarrolladores de Sistemas consideran largo y costoso el tiempo para realizar la tarea de Auditoría de Datos que permita mantener un control rápido y adecuado de los datos de su empresa. A pesar de haberla realizado no se puede conocer sobre una falla al momento en que esto sucede.

El sistema SABDO está orientado a Gerentes generales, gerentes administrativos, DBAs y Desarrolladores de Sistemas; permite realizar una auditoría de los datos de una manera eficiente y con un resultado rápido en la entrega de información acerca de los accesos y modificaciones realizadas por los usuarios. Además realiza notificaciones en línea vía e-mail sobre operaciones críticas realizadas sobre la base de datos.

Para el desarrollo del proyecto se utilizó metodología RUP (Rational Unified Process), de lo cual obtuvimos los siguientes artefactos que los presentamos en el siguiente orden:

1. Visión
2. Plan de Desarrollo del Software
3. Caso de Negocio
4. Lista de Riesgos
5. Glosario
6. Modelo de Casos de Uso
7. Prototipo de Interfaz de Usuario
8. Diccionario de datos
9. Guía de programación
10. Modelo de Implementación
11. Material de Apoyo al Usuario Final

BENEFICIOS

Luego de poner en producción el sistema SABDO llegamos a las siguientes conclusiones:

1. Mejora la seguridad del sistema y asegura la responsabilidad del sistema. Captura el acceso regular y "back-door" a los sistemas revisados de la base de datos.
2. La administración centralizada facilita la revisión de sistemas múltiples de bases de datos.
3. La interfaz gráfica acorta la curva de aprendizaje por su facilidad de uso.
4. Proporciona informes analíticos que reducen grandes cantidades de datos a resúmenes comprensivos que permiten identificar fácilmente varias violaciones de la seguridad de la base de datos.
5. Proporciona los informes analíticos que ayudan a identificar qué procesos y usuarios acaparan los recursos del sistema.
6. Proporciona detalles del rastro de intervención que son inasequibles por parte de utilidades nativas de la base de datos.

7. Proporciona capacidad de generar alarmas en tiempo real vía email a las personas claves, cuando los cambios ocurren en datos sensibles.
8. Libera al DBA de la necesidad de crear y manejar disparadores de base de datos para la revisión de modificaciones en datos.
9. Apoya la flexibilidad para revisión de configuraciones, permitiendo al personal de la seguridad elegir tipos específicos de operaciones de la base de datos y de cambios de los datos que se deban supervisar y registrar en el rastro de intervención.
10. Proporciona a nivel de sistema de manera transparente la revisión de modificaciones de datos sin permitir ningún cambio sobre los mismos.

FASE DE INICIO

Auditoría para base de datos Oracle - SABDO

VISIÓN

PLAN DE DESARROLLO DE SOFTWARE

CASO DE NEGOCIO

CAPITULO I

1 VISIÓN DEL PROYECTO

1.1 INTRODUCCIÓN

La gran difusión de los sistemas de Gestión de Bases de Datos (SGBD) y la importancia de los datos como un recurso fundamental de la institución ha despertado el interés de los temas referentes a su control interno y auditoría.

La auditoría informática se aplica de dos formas distintas:

- Se auditan las principales áreas del departamento de informática: explotación, dirección, metodología de desarrollo, sistema operativo, telecomunicaciones, bases de datos, etc.
- Se auditan las aplicaciones (Desarrolladas internamente, subcontratadas o adquiridas) que funcionan en la empresa.

La información confidencial de los clientes, usuarios, partners y empleados de una institución no le pertenece, las instituciones solo tienen una función fiduciaria sobre esos datos. Por tanto son responsables del manejo interno y externo de los datos que almacenan.

1.1.1 ALCANCE.

El documento Visión se ocupa de ofrecer una visión global de las necesidades básicas para realizar una auditoría de datos (BDD), en nuestro caso del Sistema de Auditoría de Bases de Datos en Oracle (SABDO). Dicho sistema será desarrollado por la señora Lilián Fierro y el señor Juan Carlos García.

El sistema permitirá al Gerente, Auditor, Administrador de Base de Datos, controlar todo lo relativo a los procesos correspondientes a la auditoría de bases de datos, reportando automáticamente de irregularidades detectadas. Además, también permitirá a los Desarrolladores de sistemas controlar los cambios que se realicen debido a nuevos requerimientos.

1.2 POSICIONAMIENTO

1.2.1 OPORTUNIDAD DE NEGOCIO

Siendo el manejo de los datos tan importante, nos concentraremos en la auditoría de bases de datos. Oracle es un sistema de gestión de base de datos relacional RDBMS (Relational Data Base Management System), fabricado por Oracle Corporation. La misma se ha mantenido como líder en el manejo y seguridad de los datos.

Al ser un manejador de base de datos grande, Oracle tiene una opción de auditoría, sin embargo hemos observado que ésta puede resultar complicada de manejar y requiere un gran conocimiento de su estructura para poder realizar un reporte de los accesos y operaciones realizadas en ella, por tanto existe la necesidad de elaborar un sistema amigable que se encargue de la auditoría de las bases de datos en Oracle, debido a la complejidad que actualmente presenta para realizar una auditoría.

El propósito de éste documento es recoger, analizar y definir las necesidades de alto nivel y las características del Sistema de Auditoría de Bases de Datos en Oracle (SABDO). El documento se centra en la funcionalidad requerida por los participantes en el proyecto y los usuarios finales.

Esta funcionalidad se basa principalmente en el manejo de la información que contiene una Base de Datos en Oracle, esto comprende: la auditoría de conexiones,

auditoría de acciones, auditoría de objetos y auditoría de datos. Además de una notificación a los responsables de la base de datos en caso de un suceso que atente la integridad de la BDD.

Los detalles de cómo el sistema cubre los requerimientos se pueden observar en la especificación de los casos de uso y otros documentos adicionales.

Este sistema permitirá a la institución automatizar el control de las actividades relacionadas con la auditoría de la base de datos, lo cual supondrá un reporte rápido y sencillo de los sucesos ocurridos a la base de datos, gracias a interfaces gráficas y amigables.

El sistema también permite obtener un reporte a través del Web, de forma automática sin necesidad de que alguien lo envíe.

1.2.2 SENTENCIA QUE DEFINE EL PROBLEMA

El problema radica en que la institución no dispone de herramientas de software que le permitan detectar si se han realizado accesos autorizados pero no adecuados a sus datos, quiénes lo han hecho y qué información se ha comprometido.

El problema afecta a los Administradores, Auditores, Gerentes de Sistemas, Administradores de Bases de Datos (DBA) Desarrolladores de Aplicaciones Informáticas de la institución.

El impacto asociado es el de almacenar y organizar información referente a la Auditoría de Bases de Datos Oracle, para que estos datos sean accesibles de manera oportuna y eficaz desde lugares físicamente remotos a las instalaciones de la institución. Notificación que se realizará de modo automatizado con infraestructura Web.

1.2.3 SENTENCIA QUE DEFINE LA POSICIÓN DEL PRODUCTO

El nombre del producto desarrollado en el presente proyecto es SABDO (Sistema de Auditoría de Base de Datos Oracle). El producto llamado SABDO puede ser útil para el trabajo de administradores, Auditores, Gerentes de Sistemas, Administradores de Bases de Datos (DBA), Desarrolladores de aplicaciones informáticas de la institución.

SABDO almacena la información necesaria para la auditoría de la base de datos de la institución. Permite automatizar los diferentes procesos que implica la auditoría de la base de datos Oracle mediante una interfaz gráfica sencilla y amigable. Además proporciona un acceso rápido y actualizado de la información desde cualquier punto que tenga acceso a la base de datos.

1.3 DESCRIPCIÓN DE STAKEHOLDERS (PARTICIPANTES EN EL PROYECTO) Y USUARIOS

Para proveer de una forma efectiva productos y servicios que se ajusten a las necesidades de los usuarios, es necesario identificar e involucrar a todos los participantes en el proyecto como parte del proceso de modelado de requerimientos. También es necesario identificar a los usuarios del sistema y asegurarse de que el conjunto de participantes en el proyecto los representa adecuadamente.

Esta sección muestra un perfil de los participantes y de los usuarios involucrados en el proyecto, así como los problemas más importantes que éstos perciben para enfocar la solución propuesta hacia ellos. No describe sus requisitos específicos ya que éstos se capturan mediante otro artefacto. En lugar de esto proporciona la justificación de por qué estos requisitos son necesarios.

1.3.1 RESUMEN DE STAKEHOLDERS

Los stakeholders son personas que obtendrán los beneficios del sistema propuesto pero que no participarán directamente en su desarrollo. A continuación en la tabla 1 se resumen los datos de los stakeholders.

Nombre	**Descripción**	**Responsabilidades**
Dr. Antonio Posso Salgado	Rector de la Universidad Técnica del Norte.	Máxima autoridad ejecutiva de la Institución. Dar cumplimiento a la ley de Educación Superior y estatutos de la UTN. Canaliza la asignación de recursos académicos y financieros para la ejecución del proyecto.
Dr. Luis Gómez	Auditor Interno	Inspeccionar y auditar las operaciones de la institución.
Sr. Juan Carlos García	Jefe de proyectos	Lidera los proyectos informáticos.
Ing. Evelin Enríquez	Ingeniera de software	Desarrollo de proyectos informáticos.
Ing. Luis Aguilar	Ingeniero de software	Desarrollo de proyectos informáticos.

Tabla 1 Resumen de Stakeholders

1.3.2 RESUMEN DE USUARIOS

Los usuarios son las personas que proporcionan los requerimientos necesarios para desarrollar el proyecto de acuerdo a las necesidades de la institución, los podemos ver en la tabla 2.

Nombre	Descripción	Stakeholder
Juan Carlos García	Lidera los proyectos informáticos. Realiza actividades de control y seguimiento del proyecto	Jefe de proyectos
Dr. Luis Gómez	Inspeccionar y auditar las operaciones de la institución.	Auditor Interno
Ing. Iván Chiles	Mantener la integridad y correcto funcionamiento de los sistemas y Bases de Datos.	Administrador de la BDD Oracle.

Tabla 2 Resumen de usuarios

1.3.3 ENTORNO DE USUARIO

Debido a las características específicas del sistema de auditoría propuesto, deberá ser utilizado únicamente por usuarios calificados y autorizados para realizar auditoría a la base de datos Oracle, para el presente caso el Administrador de la base de datos de la Institución.

Los usuarios ingresarán al sistema a través de un navegador de internet (Browser), luego entrarán a la parte de aplicación diseñada para realizar la auditoría de la institución.

Los informes serán generados con Oracle Reports, y los formularios serán generados por Oracle Forms.

1.4 DESCRIPCIÓN GLOBAL DEL PRODUCTO

1.4.1 PERSPECTIVA DEL PRODUCTO

El producto a desarrollar es un sistema de Auditoría de Bases de Datos, con la intención de facilitar el control de los accesos autorizados pero no adecuados a los datos, quiénes lo han hecho y qué información se ha comprometido.

Las áreas a tratar por el sistema son: Auditoría de conexiones, auditoría de acciones, auditoría de objetos y auditoría de datos, notificaciones en tiempo real.

1.4.2 RESUMEN DE CARACTERÍSTICAS

A continuación en la tabla 3 se mostrará un listado con los beneficios que obtendrá el cliente a partir del producto:

Beneficio del cliente	**Características que lo apoyan**
Mejora la seguridad del sistema y asegura la responsabilidad del sistema.	Captura el acceso regular y "back-door" a los sistemas revisados de la base de datos.
Facilita la revisión de sistemas múltiples de bases de datos.	La administración centralizada.
Acorta la curva de aprendizaje por su facilidad de uso.	La interfaz gráfica.
Reducir grandes cantidades de datos a resúmenes comprensivos que permiten identificar fácilmente varias violaciones de la seguridad de la base de datos.	Proporcionar los informes analíticos.
Ayudar a identificar qué procesos y usuarios acaparan los recursos del sistema.	Proporcionar los informes analíticos.
Proporciona detalles del rastro de intervención que son inasequibles por parte de utilidades nativas de la base de datos.	Triggers de Base de Datos
Proporciona capacidad de generar alarmas en tiempo real a las personas claves, cuando los cambios ocurren en datos sensibles.	Vía email.
Libera al DBA de la necesidad de crear y manejar disparadores de base de datos para la revisión de modificaciones en datos	Generación automática de triggers.
Permitir al personal de la seguridad elegir tipos específicos de operaciones de la base de datos y de cambios de los datos que se deban supervisar y registrar en el rastro de intervención.	Flexibilidad para revisión de configuraciones.
Proporciona a nivel de sistema de manera transparente la revisión de modificaciones de datos sin permitir ningún cambio sobre los mismos.	Manejo directo de tablas del catálogo de la Base de Datos.

Tabla 3 Beneficios del Sistema

1.4.3 COSTO Y PRECIO

Detalle		USD	Real (USD)
Hardware	Equipos de Computación	1500	0.00
	Servidor de Aplicación Web	5000	0.00
	Servidor de Base de Datos	5000	0.00
	Equipo con Web Browser	700	0.00
Software	Oracle Standard One 10g (1 licencia por procesador)	5000	0.00
	Oracle Developer Suite Release 10g	5000	0.00
	Oracle Application Server 10g, Oracle Forms, Report Server (1 licencia por procesador)	20000	0.00
Capacitación a los Desarrollares	Cursos y Libros, Asesoramiento	800	800
Proyecto	Papelería y Suministros de Oficina	500	500
Subtotal	(Parcial)	43500	1300
5% Imprevistos		240	115
Total		43740	1415

Tabla 4 Costo y Precio

1.5 DESCRIPCIÓN GLOBAL DEL PRODUCTO

El Sistema de Auditoría de Bases de Datos mantendrá un registro de las acciones realizadas sobre la base de datos, haciendo referencia a nombre de objetos modificados, fecha de modificación, usuario que ha realizado la acción, en fin los datos más relevantes para poder llevar a cabo el seguimiento de las acciones realizadas. Se mantendrá un control en tiempo real de las actividades peligrosas realizadas en la base de datos para que sean notificadas inmediatamente a la persona responsable, además las vistas de la auditoría podrán mantenerse actualizadas de acuerdo a los requerimientos del usuario.

1.5.1 AUDITORÍA DE CONEXIONES:

La auditoría de conexiones permitirá controlar las conexiones a la base de datos tanto las aceptadas como las rechazadas, manteniendo el detalle de los datos de los usuarios que

las realizaron.

1.5.2 AUDITORÍA DE ACCIONES:

La auditoría de acciones comprenderá de un registro detallado de las instrucciones utilizadas, que permiten el manejo de los datos contenidos en los objetos pertenecientes a la base de datos Oracle entre estos mencionaremos los siguientes: Create… Alter… Drop… Rebuild… Compile… Select… Insert… Update… Delete… etc; que crean, borran, alteran, insertan datos, tablas, índices, triggers, etc.

1.5.3 AUDITORÍA DE OBJETOS:

La auditoría de objetos será un registro de los objetos de bases de datos como son: tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers propios de la base de datos a los que un usuario ha accesado.CSW3

1.5.4 AUDITORÍA DE DATOS:

La auditoría de datos representa la información detallada de los datos que han sido modificados, adicionados, borrados manteniendo un registro de los valores anteriores y actuales de los datos.

1.6 RESTRICCIONES

Serán definidas por el cliente, de acuerdo a las políticas de auditoría que se manejen en la institución.

1.7 OTROS REQUISITOS DEL PRODUCTO

1.7.1 REQUISITOS DE SISTEMA

- Navegador
- Java Runtime Environment 1.4
- Visor de pdf.

1.7.2 REQUISITOS DE DESEMPEÑO

Una de las principales preocupaciones relacionada con la auditoría de base de datos es su impacto en el desempeño de las aplicaciones entre estas tenemos:

- Impacto en el tiempo de respuesta de las aplicaciones.
- Impacto sobre la utilización de los manejadores de base de datos.
- Impacto sobre el espacio en los servidores.

1.7.3 REQUISITOS DE ENTORNO

- Servidor de Base de Datos Oracle 10g R2
- Servidor de Aplicación Web, Application Server 10g

1.7.4 ATRIBUTOS DE CARACTERÍSTICAS

A continuación en la tabla 5 podemos apreciar las características del sistema en relación con el beneficio y esfuerzo requeridos.

Número y nombre de la característica	**Estado**	**Beneficio**	**Esfuerzo**	**Asignación**
5.1 Auditoría de conexiones.	Propuesta: Sí Aprobada: Sí Incorporada: No	Útil	Bajo	Ninguna
5.2 Auditoría de Acciones	Propuesta: Sí Aprobada: Sí Incorporada: No	Importante	Medio	Ninguna
5.3 Auditoría de Objetos	Propuesta: Sí Aprobada: Sí Incorporada: No	Importante	Medio	Ninguna
5.4 Auditoría de Datos	Propuesta: Sí Aprobada: Sí Incorporada: No	Crítica	Alto	Ninguna

Tabla 5 Atributos de Características

CAPÍTULO II

2 PLAN DE DESARROLLO DE SOFTWARE

2.1 INTRODUCCIÓN

Este Plan de Desarrollo de Software utilizará la metodología RUP para permitir un desarrollo apropiado del Sistema de Auditoría de Bases de Datos Basado en Oracle (SABDO). Mismo que podrá ser comercializado a empresas o instituciones interesadas en mantener una auditoría adecuada de sus sistemas basados en Oracle.

El enfoque de desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de RUP.

2.1.1 PROPÓSITO.

El plan que presentamos pretende dar una visión clara de las actividades y personas involucradas en el desarrollo del proyecto SABDO, incluyendo los alcances funcionales del mismo. Será usado por los desarrolladores de software como una guía de trabajo.

2.1.2 ALCANCE.

El Plan de Desarrollo de Software describe el plan global usado para el desarrollo del "Sistema de Auditoría de Bases de Datos en Oracle (SABDO)". El detalle de las iteraciones individuales se describe en los planes de cada iteración, documentos que se aportan en forma separada. Durante el proceso de desarrollo en el artefacto "Visión" se definen las características del producto a desarrollar, lo cual constituye la base para la

planificación de las iteraciones. Para la versión 0.1 del Plan de Desarrollo del Software, nos hemos basado en la definición de requisitos obtenidos de la experiencia en la implantación de sistemas que utilizan bases de datos Oracle, para hacer una estimación aproximada, una vez comenzado el proyecto y durante la fase de Inicio se generará la primera versión del artefacto "Visión", el cual se utilizará para refinar este documento. Posteriormente, el avance del proyecto y el seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas.

2.1.3 RESUMEN.

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

Vista General del Proyecto — proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.

Organización del Proyecto — describe la estructura organizacional del equipo de desarrollo.

Gestión del Proceso — explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.

Planes y Guías de aplicación — proporciona una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.

2.2 VISTA GENERAL DEL PROYECTO

2.2.1 PROPÓSITO, ALCANCE Y OBJETIVOS

El propósito del presente proyecto es el de desarrollar software de calidad para auditoría y seguridad de base de datos orientados a solucionar las necesidades de las instituciones, que permita **Auditar** la manipulación de información por parte de los usuarios.

El objetivo central del proyecto es proporcionar a Auditores, DBA y Desarrolladores de Sistemas una herramienta que permita detectar si se han realizado accesos autorizados pero no adecuados a sus datos, quiénes lo han hecho y qué información se ha comprometido; de una manera sencilla y rápida.

El proyecto SABDO comprende:

- Auditoría de conexiones.
- Auditoría de acciones.
- Auditoría de objetos.
- Auditoría de datos.

La Auditoría de Bases de Datos tiene como misión el registro de las acciones realizadas sobre la Base de Datos, haciendo referencia a nombre de objetos modificados, fecha de modificación, usuario que ha realizado la acción , en fin los datos más relevantes para poder llevar a cabo el seguimiento de las acciones realizadas. Los objetos referidos anteriormente son: tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers propios de la Base de Datos.

Las acciones que mencionamos son todas aquellas instrucciones que permiten el manejo de los datos contenidos en los objetos pertenecientes a la base de datos Oracle entre estos mencionaremos los siguientes:

Create… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

Alter… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

Drop… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

Rebuild… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

Compile… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

Select… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

Insert… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

Update… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

Delete… (Tablas, vistas, índices, constraints, sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers).

El proyecto debe proporcionar una propuesta para el desarrollo de todos los módulos implicados. Los módulos identificados son los siguientes:

A. MÓDULO DE CONFIGURACIÓN.

- Configuración de auditoría de conexiones.
- Configuración de auditoría de acciones.
- Configuración de auditoría de objetos.
- Configuración de auditoría de datos.

B. MÓDULO DE AUDITORÍA.

- Auditoría de conexiones.
- Auditoría de acciones.
- Auditoría de objetos.
- Auditoría de datos.

C. MÓDULO DE REPORTES.

- Reporte de conexiones.
- Reporte de acciones.
- Reporte de objetos.
- Reporte de datos.

2.2.2 SUPOSICIONES Y RESTRICCIONES

Las suposiciones y restricciones respecto del sistema, y que se derivan directamente de las entrevistas con el stakeholder de la institución son:

- El proyecto SABDO está financiado por la Universidad Técnica del Norte;
- Los módulos serán probados en la Institución para su depuración y mejoramiento.
- El sistema será diseñado sobre plataforma WEB y cumplirá con los estándares de calidad vigentes para desarrollo de software. Esto se conseguirá cumpliendo con el estándar PMI (Project Manager International, Certificación Internacional para el Manejo de Proyectos) para dirección de proyectos, metodología RUP para el proceso de ingeniería de software y herramientas Oracle 10g para la construcción de las aplicaciones.
- El proyecto será aplicable solo a Bases de Datos Oracle.
- Funcionará solamente para entornos Web.

Como es natural, la lista de suposiciones y restricciones se incrementará durante el desarrollo del proyecto, particularmente una vez establecido el artefacto "Visión".

2.2.3 ENTREGABLES DEL PROYECTO

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto.

Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de perfección y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

A. VISIÓN

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

B. PLAN DE DESARROLLO DEL SOFTWARE

Es el presente documento.

C. CASO DE NEGOCIO

Son las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas etc.). Permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito.

D. LISTA DE RIESGOS

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

E. GLOSARIO

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada.

F. MODELO DE CASOS DE USO

El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Se representa mediante Diagramas de Casos de Uso.

G. PROTOTIPO DE INTERFAZ DE USUARIO

Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.

H. DICCIONARIO DE DATOS

Un **diccionario de datos** es un conjunto de metadatos que contiene las características lógicas de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización. Estos diccionarios se desarrollan durante el análisis de flujo de datos y ayuda a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño del proyecto.

I. GUÍA DE PROGRAMACIÓN

Debido a las diferentes nomenclaturas que se utilizan en la elaboración de proyectos se requiere este documento que define los estándares que regirán para todo el proyecto.

J. MODELO DE IMPLEMENTACIÓN

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento).

K. MATERIAL DE APOYO AL USUARIO FINAL

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Guías del Usuario, Guías de Operación, Guías de Mantenimiento y Sistema de Ayuda en Línea

L. PRODUCTO

Los ficheros del producto empaquetados y almacenados en un CD con los mecanismos apropiados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de Construcción es desarrollado incremental e iterativamente, obteniéndose una nueva release al final de cada iteración.

2.2.4 EVOLUCIÓN DEL PLAN DE DESARROLLO DEL SOFTWARE

El Plan de Desarrollo del Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.

2.3 ORGANIZACIÓN DEL PROYECTO

2.3.1 PARTICIPANTES EN EL PROYECTO

De momento no se incluye el personal de la institución como Responsable del Proyecto, Comité de Control y Seguimiento, otros participantes que se estimen convenientes para proporcionar los requisitos y validar el sistema.

El personal del proyecto considerando las fases de Inicio, Elaboración, Construcción estará formado así:

Jefe de Proyecto. Labor asignada a Juan Carlos García, quien se encargará de organizar, planificar, coordinar y evaluar el desarrollo del proyecto.

Ingenieros de Software. Perfiles asignados a Juan Carlos García y Lilián Fierro, quienes realizarán labores de gestión de requisitos, gestión de configuración, documentación y diseño de datos. Encargados de las pruebas funcionales del sistema y labores de Tester.

Analistas - Programadores. Con conocimientos en el entorno de desarrollo del proyecto, con el fin de que los prototipos puedan ser lo más cercanos posibles al producto final. Este trabajo ha sido encomendado a Juan Carlos García y Lilián Fierro.

Interfaces Externas. El equipo de desarrollo interactuará activamente con los empleados de la institución para especificación y validación de los artefactos generados.

2.3.2 ROLES Y RESPONSABILIDADES

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.

Jefe de Proyecto.- El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.

Analista de Sistemas.- Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos.

Programador.- Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario.

2.4 GESTIÓN DEL PROCESO

2.4.1 ESTIMACIONES DEL PROYECTO

El presupuesto del proyecto y los recursos involucrados se adjuntan en el capítulo I: Visión, en la sección Descripción del Producto numeral 1.4.3.

2.4.2 PLAN DE LAS FASES

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La tabla 6 muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Construcción y Transición es sólo una aproximación muy preliminar)

Fase	Nro. Iteraciones	Duración
Fase de Inicio	1	3 semanas
Fase de Elaboración	1	3 semanas
Fase de Construcción	1	10 semanas
Fase de Transición	1	2 semanas

Tabla 6 Fases del proyecto

Los hitos que marcan el final de cada fase se describen a continuación:

- **Fase de Inicio**.

En esta fase desarrollará los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento periódico del Plan de Desarrollo del Proyecto. La aceptación del cliente / usuario del artefacto Visión y el Plan de Desarrollo marcan el final de esta fase.

- **Fase de Elaboración**.

En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera release de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. En nuestro caso particular, por no incluirse las fases siguientes, la revisión y entrega de todos los artefactos hasta este punto de desarrollo también se incluye como hito. La primera iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis / Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos. Ambas iteraciones tendrán una duración de una semana.

- **Fase de Construcción**

Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 iteraciones,

cada una produciendo una release a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario.

- **Fase de Transición**

En esta fase se prepararán dos releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.

2.4.3 CALENDARIO DEL PROYECTO

A continuación se presenta un calendario de las principales tareas del proyecto incluyendo sólo las fases de Inicio y Elaboración. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La figura 0-1 ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (workflow) en un momento determinado del desarrollo.

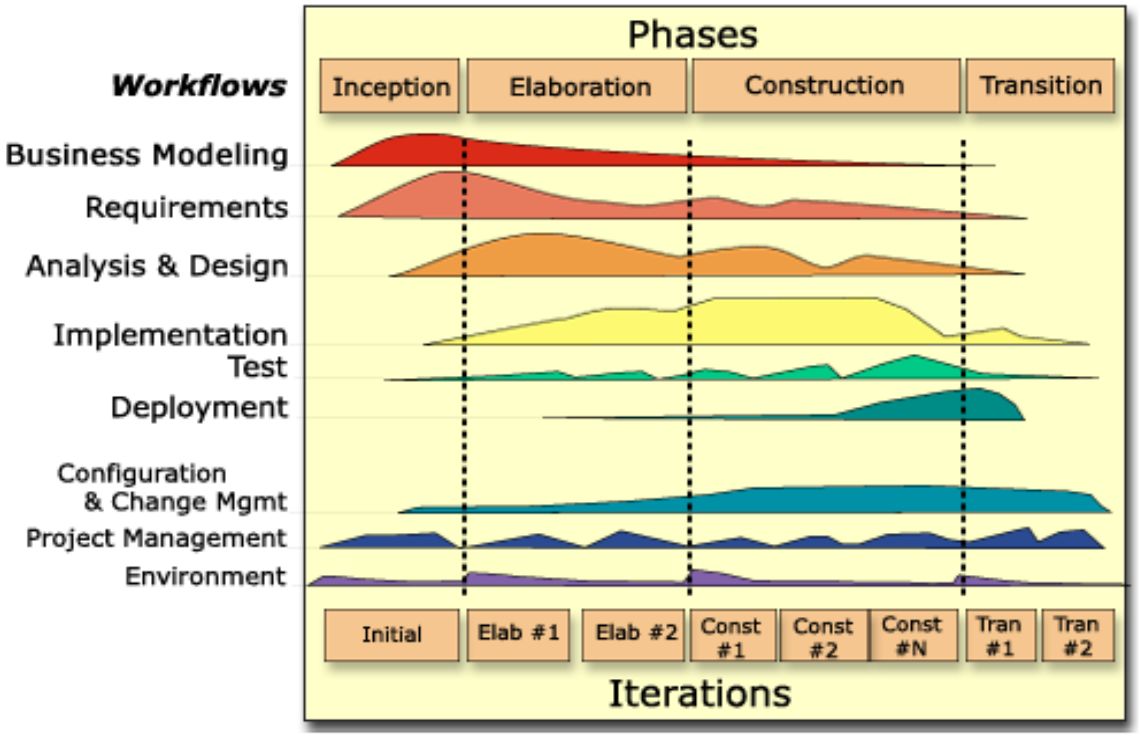


Ilustración 0-1 Fases del Proyecto

Para este proyecto se ha establecido el siguiente calendario. Ver tablas 7 y 8. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

Disciplinas / Artefactos generados o modificados durante la Fase de Inicio	**Comienzo**	**Aprobación**
Modelado del Negocio		
Casos de Negocio	Semana 1 5/12 – 19/12	Semana 3 29/12 – 02/01
Requisitos		
Glosario	Semana 1 15/12 – 19/12	Semana 3 29/12 – 02/01
Visión	Semana 2 22/12 – 26/12	Semana 3 29/12 – 02/01
Modelo de Casos de Uso	Semana 3 29/12 – 02/01	Siguiente fase
Especificaciones Adicionales	Semana 3 29/12 – 02/01	Siguiente fase
Análisis / Diseño		
Modelo de Análisis / Diseño	Semana 2 22/12 – 26/12	Siguiente fase
Modelo de Datos	Semana 2 22/12 – 26/12	Siguiente fase
Implementación		
Prototipos de Interfaces de Usuario	Semana 3 29/12 – 02/01	Siguiente fase
Modelo de Implementación	Semana 3 29/12 – 02/01	Siguiente fase
Pruebas		
Casos de Pruebas Funcionales	Semana 3 29/12 – 02/01	Siguiente fase
Despliegue		
Modelo de Despliegue	Semana 3 29/12 – 02/01	Siguiente fase
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones	Semana 1 15/12 – 19/12	Semana 3 29/12 – 02/01
Ambiente	Durante todo el proyecto	

Tabla 7 Actividades fase 1

Disciplinas / Artefactos generados o modificados durante la Fase de Elaboración	**Comienzo**	**Aprobación**
Modelado del Negocio		
Casos de Negocio	Semana 1 15/12–19/12	aprobado
Requisitos		
Glosario	Semana 1 15/12 – 19/12	aprobado
Visión	Semana 2 22/12 – 26/12	aprobado
Modelo de Casos de Uso	Semana 3 29/12 – 02/01	Semana 5 12/01 – 16/01
Análisis / Diseño		
Modelo de Análisis / Diseño	Semana 2 22/12 – 26/12	Revisar en cada iteración
Modelo de Datos	Semana 2 22/12 – 26/12	Revisar en cada iteración
Implementación		
Prototipos de Interfaces de Usuario	Semana 3 29/12 – 02/01	Revisar en cada iteración
Modelo de Implementación	Semana 3 29/12 – 02/01	Revisar en cada iteración
Pruebas		
Casos de Pruebas Funcionales	Semana 3 29/12– 02/01	Revisar en cada iteración
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 1.0 y planes de iteraciones	Semana 4 05/01– 09/01	Revisar en cada iteración
Ambiente	Durante todo el proyecto	

Tabla 8 Actividades Fase 2

2.5 SEGUIMIENTO Y CONTROL DEL PROYECTO

2.5.1 GESTIÓN DE REQUISITOS

Los requisitos del sistema son especificados en el artefacto Visión. Cada requisito tendrá una serie de atributos tales como importancia, estado, iteración donde se implementa, etc. Estos atributos permitirán realizar un efectivo seguimiento de cada requisito. Los cambios en los requisitos serán gestionados mediante una Solicitud de Cambio, las cuales serán evaluadas y distribuidas para asegurar la integridad del sistema y el correcto proceso de gestión de configuración y cambios.

2.5.2 CONTROL DE PLAZOS

El calendario del proyecto tendrá un seguimiento y evaluación semanal por el jefe de proyecto y por el Comité de Seguimiento y Control.

2.5.3 CONTROL DE CALIDAD

Los defectos detectados en las revisiones y formalizados también en una Solicitud de Cambio tendrán un seguimiento para asegurar la conformidad respecto de la solución de dichas deficiencias Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las guías de revisión y checklist (listas de verificación) incluidas en RUP.

2.5.4 GESTIÓN DE RIESGOS

A partir de la fase de Inicio se mantendrá una lista de riesgos asociados al proyecto y de las acciones establecidas como estrategia para mitigarlos o acciones de contingencia. Esta lista será evaluada al menos una vez en cada iteración.

2.5.5 GESTIÓN DE CONFIGURACIÓN

Se realizará una gestión de configuración para llevar un registro de los artefactos generados y sus versiones. También se incluirá la gestión de las Solicitudes de Cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles a todo los participantes en el proyecto. Al final de cada iteración se establecerá una baseline (un registro del estado de cada artefacto, estableciendo una versión), la cual podrá ser modificada sólo por una Solicitud de Cambio aprobada.

CAPÍTULO III

3 METODOLOGÍAS PARA LA AUDITORÍA DE BASE DE DATOS.

3.1 INTRODUCCIÓN

Como un requisito previo para la realización del proyecto propuesto debemos investigar a fondo los aspectos relevantes sobre las metodologías para la realización de una auditoría de base de datos, los requerimientos de los usuarios son de gran importancia y conjugados con el conocimiento teórico – práctico que ha sido publicado podremos obtener el éxito esperado.

3.1.1 PROPÓSITO.

El propósito de éste capítulo es el de establecer las condiciones actuales del manejo de los datos de las instituciones, las políticas definidas para la realización de una auditoría del negocio.

3.1.2 ALCANCE.

Se analizará un plan de auditoría de sistemas de gestión de bases de datos, SGBD que contenga lineamientos generales útiles para un mejor control y manejo de los activos informáticos (información confidencial de la empresa, datos de proveedores, etc.).

3.1.3 DEFINICIONES, CONCEPTOS Y ASPECTOS IMPORTANTES DE UN PLAN DE AUDITORÍA DE SISTEMAS DE GESTIÓN DE BASES DE DATOS.

AUDITORIA INFORMÁTICA:

La auditoría informática se aplica de dos formas distintas:

- Se auditan las principales áreas del departamento de informática: explotación, dirección, metodologías de desarrollo, sistema operativo, telecomunicaciones, bases de datos, etc.
- Se auditan las aplicaciones desarrolladas internamente, subcontratadas o adquiridas que funcionan en la institución.

DATOS: INFORMACIÓN Y ORGANIZACIÓN

Los datos convertidos en información a través de bases de datos y procesos de negocios representan el negocio. Sin ellos éste no existe.

Las organizaciones deben tomar medidas mucho más allá de asegurar sus datos. Deben monitorearse perfectamente a fin de conocer quién o qué les hizo exactamente qué, cuándo y cómo.

Las organizaciones deben mitigar los riesgos asociados a la pérdida de datos y a la fuga de información:

- Se requiere establecer mecanismos de auditoría de base de datos.
- La memoria institucional reside en bases de datos y debe ser cultivada y protegida.

AUDITORIA DE BASES DE DATOS:

Es el proceso que permite medir, asegurar, demostrar, monitorear y registrar los accesos a la información almacenada en las bases de datos incluyendo la capacidad de determinar:

- Quién accede a los datos
- Cuándo se accedió a los datos.
- Desde qué tipo de dispositivo/aplicación.
- Desde qué ubicación en la red.
- Cuál fue la sentencia SQL ejecutada.
- Cuál fue el efecto del acceso a la base de datos.

Es uno de los procesos fundamentales para apoyar la responsabilidad delegada a TI por la organización frente a las regulaciones y su entorno de negocios o actividad.

ABREVIACIONES Y ACRÓNIMOS.

SGBD: Sistemas de Gestión de Bases de Datos.

TI: Tecnología de la Información.

BD: Base de Datos.

L4G: Lenguajes de cuarta generación.

SO: Sistema Operativo.

ISO: Organización de estándares internacionales.

3.2 OPORTUNIDAD DEL NEGOCIO

Cuando el auditor se encuentra en el sistema en explotación, deberá estudiar el SGBD y su entorno. El problema de las bases de datos es que su entorno cada vez es más

complejo y no puede limitarse solo al propio SGBD. En la figura 1 se muestra un posible entorno de bases de datos en el que aparecen los elementos más usados.

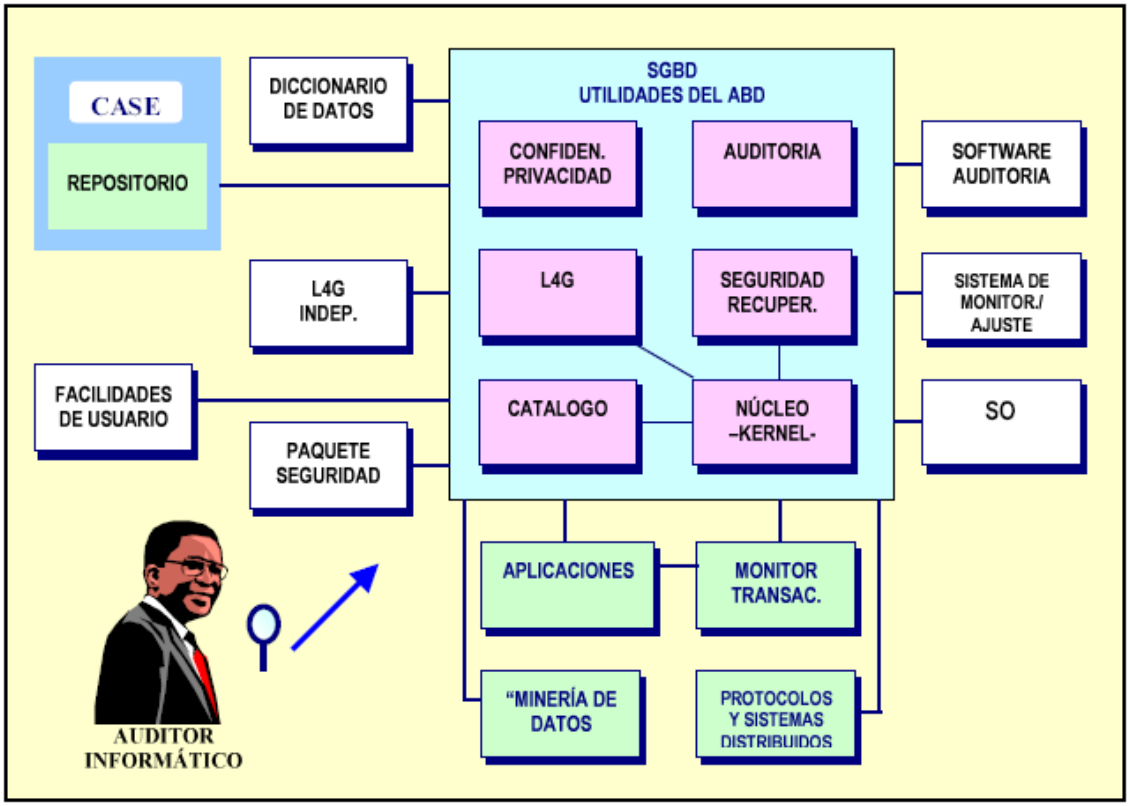


Ilustración 0-2 Entorno de base de datos.

3.2.1 SISTEMAS DE GESTIÓN DE BASES DE DATOS (SGBD).

Entre los componentes del SGBD podemos destacar el núcleo (kernel), el catálogo (componente fundamental para asegurar la seguridad de la base de datos), las utilidades para el administrador de la base de datos (entre la que se pueden encontrar algunas para crear usuario, conceder privilegios y resolver otras cuestiones relativas a la confidencialidad); las que se encargan de la recuperación de la BD: rearranque, copias de respaldo, ficheros diarios (log), etc. Y algunas funciones de auditoría, así como los lenguajes de la cuarta generación (L4G) que incorpora el propio SGBD.

En cuanto a las funciones de auditoría que ofrece el propio sistema, prácticamente todos los productos del mercado permiten registrar ciertas operaciones realizadas sobre la base de datos de un fichero (o en un conjunto de tablas) de pistas de auditoría (audit trail). El propio modelo de referencia de gestión de datos -ISO (1993)- considera las pistas de auditoría como un elemento esencial de un SGBD, señalando que "*el requisito para la auditoría es que la causa y el efecto de todos los cambios de la base de datos sean verificables".*

El auditor deberá revisar, por lo tanto, la utilización de todas las herramientas que ofrece el propio SGBD y las políticas y procedimientos que sobre su utilización haya definido el administrador, para valorar si son suficientes o si deben ser mejorados.

3.2.2 SOFTWARE DE AUDITORÍA.

Son paquetes que pueden emplearse para facilitar la labor del auditor, en cuanto a la extracción de datos de la base, el seguimiento de las transacciones, datos de prueba, etc. Hay también productos muy interesantes que permiten cuadrar datos de diferentes entornos permitiendo realizar una verdadera "auditoría del dato".

3.2.3 SISTEMA DE MONITORIZACIÓN Y AJUSTE (TUNNING)

Este tipo de sistema complementan las facilidades ofrecidas por el propio SGBD, ofreciendo mayor información para optimizar el sistema, llegando a ser en determinadas ocasiones verdaderos sistemas expertos que proporcionan la estructura óptima de la base de datos y de ciertos parámetros del SGBD y del SO.

La optimización de la base de datos, como ya hemos señalado, es fundamental, puesto que se actúa en un entorno concurrente puede degradarse fácilmente el nivel de servicio que haya podido establecerse con los usuarios.

3.2.4 SISTEMA OPERATIVO (SO)

El SO es una pieza clave del entorno, puesto que el SGBD se apoyará, en mayor o menor medida (según se trate de un SGBD dependiente o independiente) en los servicios que le ofrezca; eso en cuanto a control de memoria, gestión de áreas de almacenamiento intermedio, manejo de errores, control de confidencialidad, mecanismo de interbloqueo, etc. Desafortunadamente, el auditor informático tiene serias dificultades para controlar de manera rigurosa la interfaz entre el SGBD y el SO, debido a que, en parte, constituye información reservada de los fabricantes de los productos, además de requerir unos conocimientos excepcionales que entran en el campo de la técnica de sistemas.

3.2.5 MONITOR DE TRANSACCIONES

Algunos autores lo incluyen dentro del propio SGBD, pero actualmente, puede considerarse un elemento más del entorno con responsabilidades de confidencialidad y rendimiento.

3.2.6 PROTOCOLOS Y SISTEMAS DISTRIBUIDOS

Cada vez más se está accediendo a las bases de datos a través de redes, con lo que el riesgo de violación de la confidencialidad e integridad se acentúa. También las bases de datos distribuidas pueden presentar graves riesgos de seguridad.

Se establece cinco objetivos de control a la hora de revisar la distribución de datos.

1) El sistema de proceso distribuido debe tener una función de administración de datos centralizada que establezca estándares generales para la distribución de datos a través de las aplicaciones.
2) Deben establecerse unas funciones de administración de datos y de base de datos fuertes, para que puedan controlar la distribución de datos.
3) Deben existir pistas de auditoría para todas las actividades realizadas por las aplicaciones contra sus propias bases de datos y otras compartidas.

4) Deben existir controles software para prevenir interferencias de actualización sobre las bases de datos en sistemas distribuidos.
5) Deben realizarse las consideraciones adecuadas de costes y beneficios en el diseño de entornos distribuidos.

Respecto a este último punto, es importante destacar como, por ejemplo, muy pocas empresas han considerado rentable implementar bases de datos "realmente" distribuidas; siendo más económico y usual actualizar bases de datos distribuidas mediante trasferencia de ficheros y procesos por lotes (batch), que hacerlo en línea.

3.2.7 PAQUETES DE SEGURIDAD

Existen en el mercado varios productos que permiten la implantación efectiva de una política de seguridad, puesto que centralizan el control de accesos, la definición de privilegios, perfiles de usuarios, etc. Un grave inconveniente de este tipo de software es que a veces no se encuentra bien integrado con el SGBD, pudiendo resultar poco útil su implantación si los usuarios pueden "saltarse" los controles a través del propio SGBD.

3.2.8 DICCIONARIO DE DATOS

Este tipo de sistemas, que empezaron a implantarse en los años setenta, también juegan un papel primordial en el entorno de los SGBD en cuanto a la integración de componentes y al cumplimiento de la seguridad de los datos.

Los propios diccionarios se pueden auditar de manera análoga de las bases de datos (puesto que son bases de "metadatos"), las diferencias entre unos y otros , residen principalmente en que un fallo en una base de datos puede atentar contra la integridad de los datos y producir un mayor riesgo financiero, mientras que un fallo en un diccionario o repositorios, suele llevar consigo una pérdida de integridad de los procesos; siendo más peligrosos los fallos en los diccionarios puesto que pueden introducir errores de forma repetitiva a lo largo del tiempo, que son más difíciles de detectar.

3.2.9 LENGUAJES DE CUARTA GENERACIÓN (L4G) INDEPENDIENTES

Además de las herramientas que ofrezca el propio SGBD, al auditor se puede encontrar con una amplia gama de generadores de aplicaciones, de formas, de informes, etc., que actúan sobre la base de datos y que, por tanto, también son un elemento importante a considerar en el entorno del SGBD.

Se ofrecen varios objetivos de control para los L4G, entre los que destacan los siguientes:

- El L4G debe ser capaz de operar en el entorno de proceso de datos con controles adecuados.
- Las aplicaciones desarrolladas con L4G deben seguir los mismos procedimientos de autorización y petición de los proyectos de desarrollo convencional.
- Las aplicaciones desarrolladas con L4G deben sacar ventaja de las características incluidas en los mismos.

En efecto, uno de los peligros más graves de los L4G es que no se apliquen controles con el mismo rigor que a los programas desarrollados con lenguaje de tercera generación. Esto puede deberse, en parte, a una inadecuada interfaz entre el L4G y el paquete de seguridad y a la falta de código fuente en el sentido tradicional, que hacen más difícil de esta manera el control de cambios en las aplicaciones.

Otros problemas asociados a los L4G y con lo que nos encontramos frecuentemente, pueden ser su ineficacia y elevado consumo de recursos, las limitaciones que, en ocasiones, imponen al programador, los cambios que pueden suponer en la metodología de desarrollo, etc. Respecto a este último punto, muchos L4G se utilizan en la actualidad para desarrollar prototipos que facilitan a los usuarios la exposición de sus necesidades. "*El prototipo de una aplicación desarrollada con L4G deben proporcionar suficiente detalle para*

reemplazar los documentos escritos asociados a los procedimientos convencionales de la metodología de desarrollo de sistema".

El auditor deberá estudiar los controles disponibles en los L4G utilizados en la empresa, analizando con atención si permiten construir procedimientos de control y auditoria dentro de las aplicaciones y, en caso negativo, recomendar su construcción utilizando lenguajes de tercera generación.

3.2.10 FACILIDADES DE USUARIOS

Con la aparición de interfaces gráficas fáciles de usar (con menús, ratón, ventanas, etc.) se ha desarrollado toda una serie de herramientas que permiten al usuario final acceder a los datos sin tener que conocer la sintaxis de los lenguajes del SGBD. El auditor deberá investigar las medidas de seguridad que ofrecen estas herramientas y bajo qué condiciones han sido instaladas; las herramientas de este tipo deberían proteger al usuario de sus propios "errores".

Las aplicaciones desarrolladas empleando facilidades de usuario deben seguir los mismos sólidos principios de control y tratamiento de errores que el resto; Se destaca también otros dos importantes objetivos de control:

- La documentación de las aplicaciones desarrolladas por los usuarios finales debe ser suficiente para que tanto sus usuarios principales como cualquier otro puedan operar y mantenerlas.
- Los cambios de estas aplicaciones requieren la aprobación de la dirección y deben documentarse de forma completa.

En este apartado podemos incluir también las diferentes facilidades que ofrecen algunos SGBD que permiten su conexión con paquetes ofimáticos (por ejemplo, hojas de cálculo), que pueden acceder a la base de datos e incluso actualizarla. En este caso el auditor debe prestar especial atención a los procedimientos de carga y descarga

(uploading/downloading) de datos de la base a/desde los paquetes ofimáticos; comprobando, por ejemplo, si se puede actualizar la base de datos desde cualquiera de éstos o si la descarga se realiza con datos correctamente actualizados ("descarga de los datos correctos en el momento correcto").

3.2.11 HERRAMIENTAS DE "MINERÍA DE DATOS"

En los últimos años ha explosionado el fenómeno de los almacenes de datos **datawarehouses** y las herramientas para la explotación o "minería" de datos (datamining). Estas herramientas ofrecen soporte a la toma de decisiones sobre datos de calidad integrados en el almacén de datos. Debiéndose controlar la política de refresco y carga de los datos en el almacén a partir de las bases de datos operacionales existentes, así como la existencia de mecanismos de retroalimentación (feedback) que modifican las bases de datos operacionales a partir de los datos del almacén.

3.2.12 APLICACIONES

El auditor deberá controlar que las aplicaciones no atentan contra la integridad de los datos de base.

3.3 POSICIONAMIENTO DEL NEGOCIO

La seguridad de los datos y estructuras de las bases de datos es muy importante en un ambiente de producción, incluso de desarrollo para garantizar la disponibilidad y confiabilidad de la información. Por esto se hace necesario configurar un esquema de seguridad para garantizar estos aspectos, valiéndose de las potencialidades que proveen el hardware y el software de los servidores y motores de base de datos.

Los esfuerzos en seguridad de base de datos normalmente están orientados a:

- Impedir el acceso externo
- Impedir el acceso interno a usuarios no autorizados

- Autorizar el acceso sólo a los usuarios autorizados

Con la auditoría de BD se busca:

- Monitorear y registrar el uso de los datos por los usuarios autorizados o no.
- Mantener trazas de uso y del acceso a bases de datos
- Permitir investigaciones forenses (en caso de fraudes)
- Generar alertas en tiempo real.

3.3.1 OBJETIVOS GENERALES DE LA AUDITORÍA DE BASES DE DATOS

Disponer de mecanismos que permitan tener trazas de auditoría completas y automáticas relacionadas con el acceso a la base de datos incluyendo la capacidad de generar alertas con el objetivo de:

- Apoyar el cumplimiento regulatorio
- Mitigar los riesgos asociados con el manejo inadecuado de los datos
- Satisfacer los requerimientos de los auditores
- Evitar acciones criminales
- Evitar multas por incumplimiento

Para conseguir todo esto se deben tener en cuenta cuatro aspectos claves:

1. **No se debe comprometer el desempeño de la base de datos:** Debe soportar diferentes esquemas de auditoría, tomar en cuenta el tamaño de la base de datos a auditar y los posibles SLA establecidos.
2. **Realizar la segregación de funciones:** El sistema de auditoría de base de datos no puede ser administrado por los DBA del área de TI

3. **Proveer valor a la operación del negocio:** Debe proveer de información para auditoría y seguridad, para apoyar la toma de decisiones de la institución, para mejorar el desempeño de la organización.
4. **La auditoría debe ser completa y extensiva:** Estandarizar los reportes y reglas de auditoría.

3.3.2 CARACTERÍSTICAS PRINCIPALES DE UN SISTEMA DE AUDITORÍA DE BASE DE DATOS

El sistema de auditoría de base de datos debe cumplir con ciertas características como las siguientes:

3.3.2.1 SISTEMA CONFIABLE E INTEGRAL

Se deben poder garantizar esquemas de auditoría contínua todos los 365 días del año, es decir:

- El volumen de la información relacionada con las trazas de auditoría puede ser muchas veces más grande que el tamaño de la base de datos a auditar.
- Se debe poder controlar el acceso y la modificación a las trazas almacenadas.
- Se debe poder proteger la información almacenada en las trazas de auditoría

Se deben poder auditar el acceso a la base de datos desde todas las posibles ***capas de acceso:***

- **Aplicación / Front End:** La capa de aplicación es utilizada por usuarios calificados y posiblemente por usuarios maliciosos. Se pueden explotar las debilidades de las aplicaciones. Por ejemplo con la inyección SQL.
- **Servidor de Aplicación / Web Server:** Los usuarios con privilegios pueden tener acceso directo utilizando funciones del servidor de aplicaciones. Usuarios

maliciosos pueden aprovechar las debilidades de los servidores de aplicaciones. Pueden existir caballos de Troya listos para actuar en el código de las aplicaciones.

- **Manejador de BDD:** Pueden realizarse accesos no autorizados ODBC a través de la Red; Accesos no autorizados de usuarios con privilegios. Vulnerabilidades conocidas de los manejadores de base de datos.
- **Sistema Operativo:** Tenemos los accesos directos a los archivos en el sistema operativo.

3.3.2.2 CAPAZ DE CONSOLIDAR LAS TRAZAS DE AUDITORÍA

Las trazas de auditoría deben tener:

- Quién realizó la operación
- Desde donde se realizó la operación (Dirección IP / Host o Aplicación)
- Cuándo se realizó la operación
- Qué se hizo durante la operación (información antes y después)
- Por qué se hizo la operación (contexto sobre el que se realiza la operación)

3.3.2.3 REGLAS DE AUDITORÍA BASADAS EN NECESIDADES ESPECÍFICAS

Las trazas de auditoría pueden llegar a tener tamaños inmanejables (cientos de terabytes de información).

Cada proceso de auditoría requiere su propia definición de reglas de auditoría se deben utilizar:

- Qué método de auditoría
- Qué Servidor /manejador (Base de Datos / tablas)
- Qué usuarios (Objetos, Horas, Acciones)
- Qué acciones tomar

Tipos de eventos a auditar:

- Eventos tipo DDL (Data Definition Lenguaje): asociados con la creación de usuarios, roles, tablas, etc.
- Eventos tipo DML (Data Manipulation Lenguaje): insert, update, delete.
- Eventos tipo Select: consultas de información
- Recompilaciones de Scripts
- Auditoría de los usuarios con privilegios de acceso.

3.3.2.4 NO SE DEBE AFECTAR EL DESEMPEÑO DE LA BASE DE DATOS.

Una de las principales preocupaciones relacionada con la auditoría de base de datos es su impacto en el desempeño de las aplicaciones. Es decir:

- Impacto en el tiempo de respuesta de las aplicaciones
- Impacto sobre la utilización de los manejadores de base de datos
- Impacto sobre el espacio en los servidores

3.3.2.5 CAPAZ DE GENERAR NOTIFICACIONES EN TIEMPO REAL

Para lograr esquemas más efectivos de auditoría es necesario poder generar notificaciones en tiempo real. Mediante Email o Sistemas de monitoreo central.

3.3.2.6 CAPACIDAD PARA RETENER TRAZAS POR LARGOS PERÍODOS DE TIEMPO

Como el número de trazas puede ser muy grande se necesitan esquemas de archivos y preservación, como son:

- Compresión de la información
- Movimiento a cintas
- Rutinas de eliminación

3.3.2.7FLEXIBILIDAD PARA CREAR REPORTES

Es necesario crear reportes que permitan:

- Cumplir con los objetivos de los auditores
- Cumplir con los objetivos de las otras involucradas en el proceso
- Consolidar información
- Preparar reportes estadísticos
- Generar reportes que apoyen la toma de decisión corporativa
- Generar reportes que aporten valor a la actividad de la organización (La auditoría nos ofrece una nueva perspectiva de los datos)

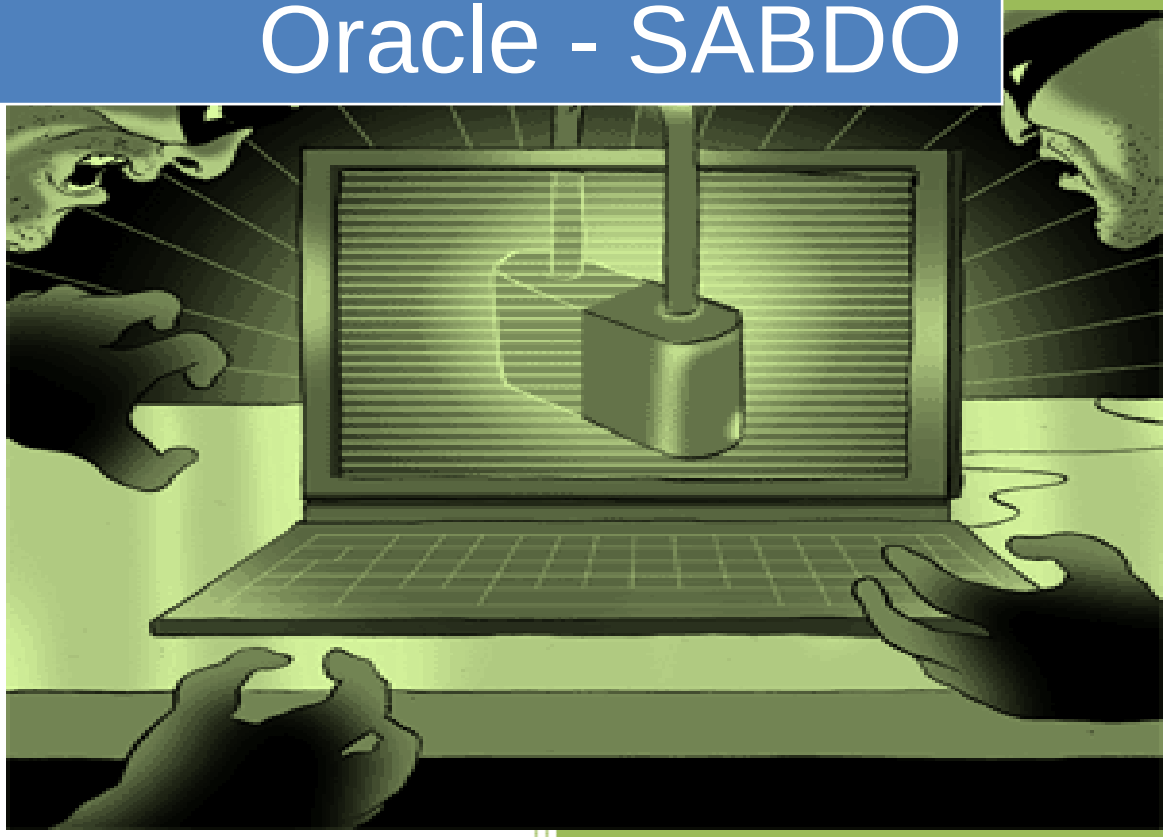
3.3.2.8 ADMINISTRABLE Y ESCALABLE EN EL TIEMPO

Debe realizarse una planificación de la Auditoría de base de datos que permita:

1. Identificar la base de datos de la institución
2. Clasificar los niveles de riesgo de los datos en la base de datos
3. Analizar los permisos de acceso
4. Analizar los controles existentes de acceso a la base de datos
5. Establecer los modelos de auditoría de BD a utilizar
6. Establecer las pruebas a realizar para la BD, aplicación y/o usuario
7. Alcance de la Auditoría (Selección de: tablas, usuarios, horario a auditar)
8. Tipos de Acciones(Solo registro de eventos, solo generación de alertas, Generación y registro de alertas)
9. Reportes a producir (Contenido y Frecuencia)

FASE DE ELABORACION

Auditoría para base de datos Oracle - SABDO



ESPECIFICACIÓN DE CASOS DE USO

CAPÍTULO IV

4 ESPECIFICACIÓN DE CASOS DE USO

4.1 CASO DE USO: INICIALIZAR AUDITORÍA DE BDD ORACLE.

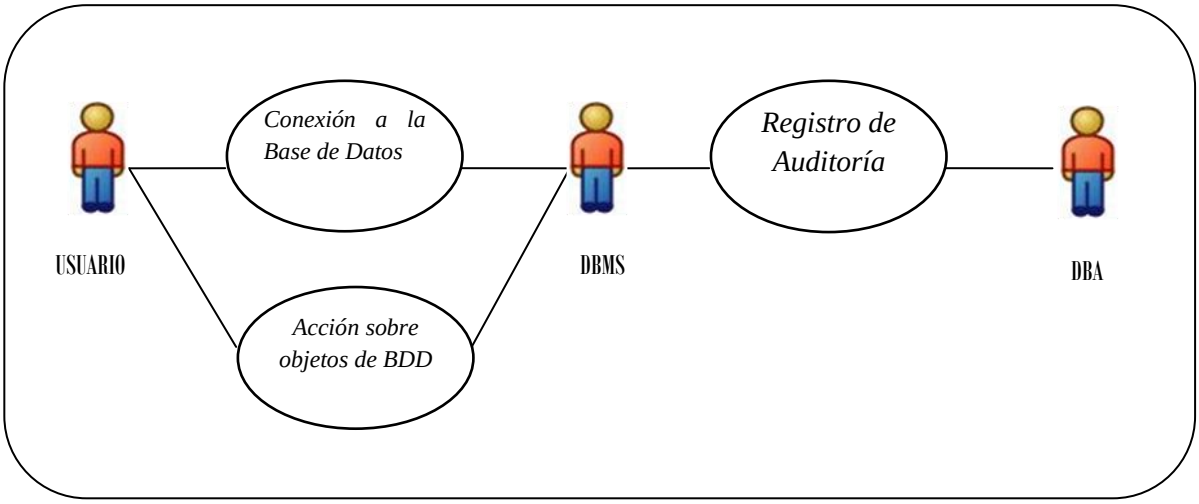


Ilustración 0-3 Auditoría de BDD

4.1.1 DESCRIPCIÓN BREVE

Describiremos las actividades preliminares para la utilización del Sistema de Auditoría de Base de Datos en Oracle, los usuarios de este sistema deben habilitar la opción de auditoría de Base de Datos mediante un parámetro de inicialización Audit Trail.

4.1.2 FLUJO BÁSICO DE EVENTOS

1. Utilizar Oracle Enterprise Manager (OEM) o ingresar directamente con la dirección http://localhost:1158/em o en lugar de local host poner la dirección IP donde se encuentra instalado el motor de la base de datos.
2. En la pantalla de login utilizar el usuario SYSTEM y conectar como normal.

3. Ingresar en la pestaña de Administración para configurar los parámetros de inicialización, escoger SPFile con la categoría de Seguridad y Auditoría.
4. Marcar el parámetro Audit Trail, en la columna de valor cambiar la opción actual a TRUE, finalmente aplicar a la base de datos.
5. Cerrar aplicación.

4.1.3 FLUJOS ALTERNATIVOS

En el punto uno puede reemplazarse el uso de OEM por Toad, Sql plus, etc que permitan acceder a la base de datos para habilitar la opción de auditoría de base de datos Oracle.

4.1.4 PRECONDICIONES

- Tener un usuario con los privilegios necesarios para configurar la BDD Oracle.

4.1.5 POSTCONDICIONES

- En el punto 5 se debe reiniciar la BDD Oracle para garantizar que los cambios realizados se actualicen.

4.2 CASO DE USO: INGRESO AL SISTEMA

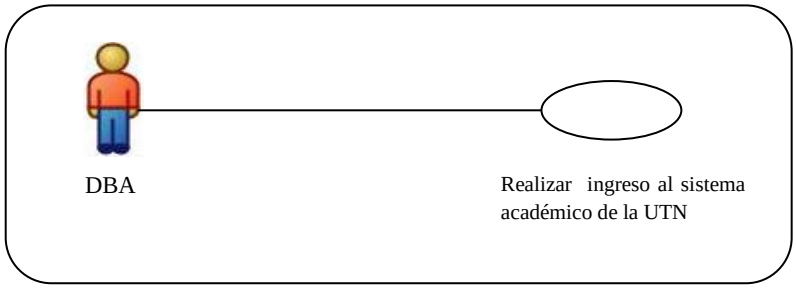


Ilustración 0-4 C.U. Ingreso al sistema

4.2.1 DESCRIPCIÓN BREVE

Describiremos las actividades preliminares para la utilización del Sistema de Auditoría de Base de Datos en Oracle, los usuarios de este sistema deben ingresar al sistema académico de la Universidad Técnica del Norte y luego al subsistema de Auditoría de BDD para Oracle que se encuentra integrado en el mismo.

4.2.2 FLUJO BÁSICO DE EVENTOS

1. Para poder trabajar es necesario abrir un browser o navegador de internet como son: Internet Explorer, Firefox, Netscape, etc. Una vez abierto debemos ubicarnos en la barra de direcciones e ingresar la siguiente dirección URL : http://172.20.1.159:7777/forms/frmservlet?config=utn
2. En la ventana que aparecerá, se ingresa el nombre del usuario (utndb), contraseña y la base de datos (servidor).
3. Escoger la suite de Seguridad y Auditoría.
4. En el menú visualizado escoger la opción de Auditoría.

4.2.3 FLUJOS ALTERNATIVOS

En el punto uno puede escogerse el navegador de internet que prefiera el usuario.

4.2.4 PRECONDICIONES

- Haber habilitado el parámetro de inicialización de la auditoría de BDD Oracle.

4.3 CASO DE USO: CONFIGURACIÓN DE LA AUDITORÍA DE SISTEMA.

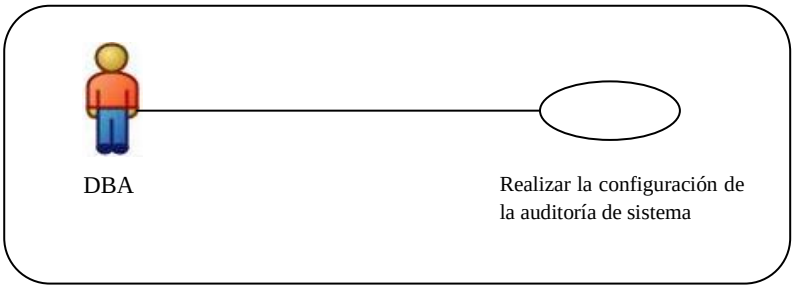


Ilustración 0-5 C.U. Configuración de auditoría de sistema

4.3.1 DESCRIPCIÓN BREVE

Los usuarios de este sistema seleccionan la opción de auditoría de sistema para poder conocer las conexiones exitosas o no que se efectuaron en la base de datos, deberán ingresar a la opción de configuración de la auditoría antes de realizar el monitoreo de la información requerida.

4.3.2 FLUJO BÁSICO DE EVENTOS

1. En la ventana principal de la auditoría, el usuario seleccionará la pestaña de auditoría de sistema, la cual permite realizar la configuración y el monitoreo.
2. Ingresar en la opción de configuración, la cual consta de: configuración de sesiones, acciones, objetos. Si se escogió: sesiones ir al paso 3, acciones ir al paso 5, objetos ir al paso 8.
3. La opción de configuración de sesiones permite elegir el tipo de acciones de la operación CONNECT (conectar) que se realiza en la BDD, estas pueden ser: conexiones exitosas, conexiones no exitosas y auditar siempre (todas las conexiones exitosas o no).

4. Una vez realizada la configuración tenemos dos opciones: auditar y no auditar, al escoger la primera inicia la auditoría de conexiones, la segunda detiene la auditoría de las conexiones a la BDD. Ir al paso 11.
5. La configuración de acciones permite seleccionar el tipo de objetos (tablas, constrains, índices, vistas, triggers, procedimientos, etc) o privilegios (crear sesiones, SYSDBA, SYSTEM, SYS) de la BDD
6. A continuación elegir las acciones que pueden realizarse sobre los objetos y privilegios del paso anterior indicando que serán o no parte de la auditoría.
7. Al finalizar la configuración presionar el botón: PROCESAR para realizar la auditoría de las acciones. Ir al paso 11.
8. La opción de configuración de objetos permite seleccionar: nombre de usuario, tipos de objetos (procedimientos, tablas, vistas, triggers, etc.) y los objetos que forman parte de la base de datos actual (nombre de cada una de las tablas, vistas, procedimientos, etc.)
9. Configurar las acciones que se pueden realizar sobre los objetos, éstas son: Alter (Alter (alterar), Audit (auditar), Compile (compilar), Delete (borrar), Grant (otorgar privilegio), Index (índice), Insert (insertar), Lock (bloquear), Rename (renombrar), Select (seleccionar), Update (actualizar), Reference (referencias), Execute (ejecutar) Create (crear), Read (leer), Write (escribir).
10. Una vez finalizada la configuración presionar el botón: PROCESAR.
11. Utilizar el botón de salida para volver al menú principal.

4.3.3 FLUJOS ALTERNATIVOS

- En el punto 7 y 10 se deshabilita la auditoría desconfigurando las acciones que se encuentran marcadas para auditoría y PROCESAR nuevamente.

- En los puntos 3, 5 y 8 puede ir al punto 11 para salir sin realizar configuración en el caso de cancelar la configuración de auditoría de sistema.

4.3.4 PRECONDICIONES

- Haber ingresado al sistema académico de la Universidad Técnica del Norte.

4.3.5 POSTCONDICIONES

- Luego de configurar la auditoría de sistema debe realizar el monitoreo para poder ver la información resultado de la configuración de auditoría de sistemas.

4.4 CASO DE USO: MONITOREO DE LA AUDITORÍA DE SISTEMA

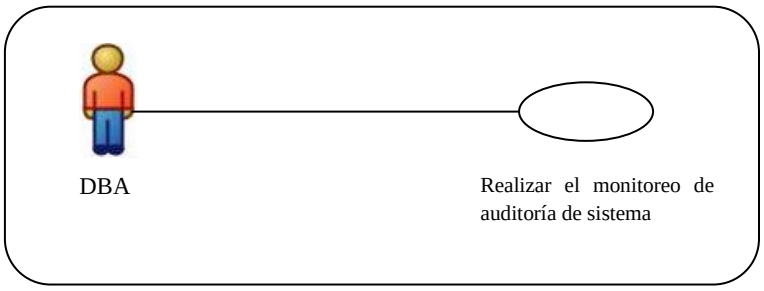


Ilustración 0-6 C.U. Monitoreo de auditoría de sistema

4.4.1 DESCRIPCIÓN BREVE

Una vez finalizada la configuración de auditoría de sistema procedemos a la visualización de la información requerida por el auditor mediante el monitoreo de sistema.

4.4.2 FLUJO BÁSICO DE EVENTOS

1. En la ventana principal de la auditoría, el usuario seleccionará la pestaña de auditoría de sistema, la cual permite realizar la configuración y el monitoreo.

2. Ingresar en la opción de monitoreo, tenemos dos elecciones posibles: monitoreo de sesiones ir al paso 3 o monitoreo de objetos (las acciones que se realizaron sobre estos) ir al paso 5.
3. El monitoreo de sesiones visualizará la siguiente información básica: Nombre de usuario, código del terminal, Nombre del Usuario de Sistema Operativo, Estado de conexión. A continuación se ubicará un menú navegable con la información adicional requerida en la configuración. Ir al paso 5.
4. El monitoreo de objetos y acciones visualizará la siguiente información básica: Cuenta de usuario de sistema operativo asociado al usuario de BDD cuyas acciones fueron auditadas, Nombre del usuario cuyas acciones fueron auditadas, Nombre de la máquina anfitrión del cliente, Identificador del terminal del usuario. A continuación se ubicará un menú navegable con la información adicional requerida en la configuración.
5. Utilizar la opción salir para regresar al menú principal.

4.4.3 PRECONDICIONES

- Haber ingresado al sistema académico de la Universidad Técnica del norte.
- Haber realizado la configuración de sesiones y de objetos.

4.5 CASO DE USO: CONFIGURACIÓN DE LA AUDITORÍA DE DATOS

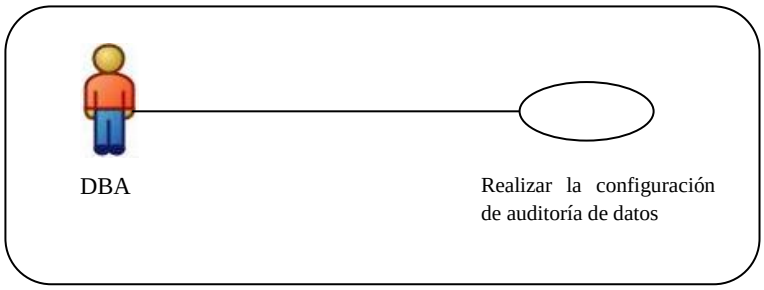


Ilustración 0-7 C.U. Configuración de auditoría de datos

4.5.1 DESCRIPCIÓN BREVE

Los usuarios de este sistema seleccionan la opción de auditoría de datos para obtener el detalle sobre los datos que fueron modificados, adicionados, borrados de la base de datos, deberán ingresar a la opción de configuración de la auditoría antes de realizar el monitoreo de la información requerida.

4.5.2 FLUJO BÁSICO DE EVENTOS

1. En la ventana principal de la auditoría, el usuario seleccionará la pestaña de auditoría de datos, la cual permite realizar la configuración y el monitoreo.
2. Ingresar en la opción de configuración que consta del usuario sus tablas y columnas. Además las acciones básicas: inserción, eliminación y actualización que pueden auditarse.
3. Elegir el nombre del usuario.
4. Escoger las acciones básicas que pueden realizarse sobre las tablas pertenecientes a este usuario.
5. Escoger las columnas pertenecientes a las tablas anteriores que serán auditadas.

6. Finalmente tenemos dos opciones auditar (inicia la auditoría) y no auditar (detiene la auditoría).
7. Utilizar la opción salir para regresar al menú principal.

4.5.3 PRECONDICIONES

- Haber ingresado al sistema académico de la Universidad Técnica del Norte.

4.5.4 POSTCONDICIONES

- Luego de configurar la auditoría de datos debe realizar el monitoreo para poder ver la información resultado de la configuración de auditoría de dato.

4.6 CASO DE USO: MONITOREO DE DATOS

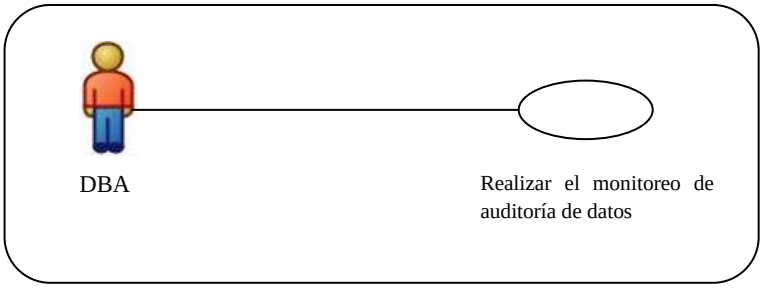


Ilustración 0-8 C.U. Configuración de auditoría de datos

4.6.1 DESCRIPCIÓN BREVE

Una vez finalizada la configuración de auditoría de datos procedemos a la visualización de la información requerida por el auditor mediante el monitoreo de datos.

4.6.2 FLUJO BÁSICO DE EVENTOS

1. En la ventana principal de la auditoría, el usuario seleccionará la pestaña de auditoría de datos, la cual mostrará las opciones configuración y monitoreo.

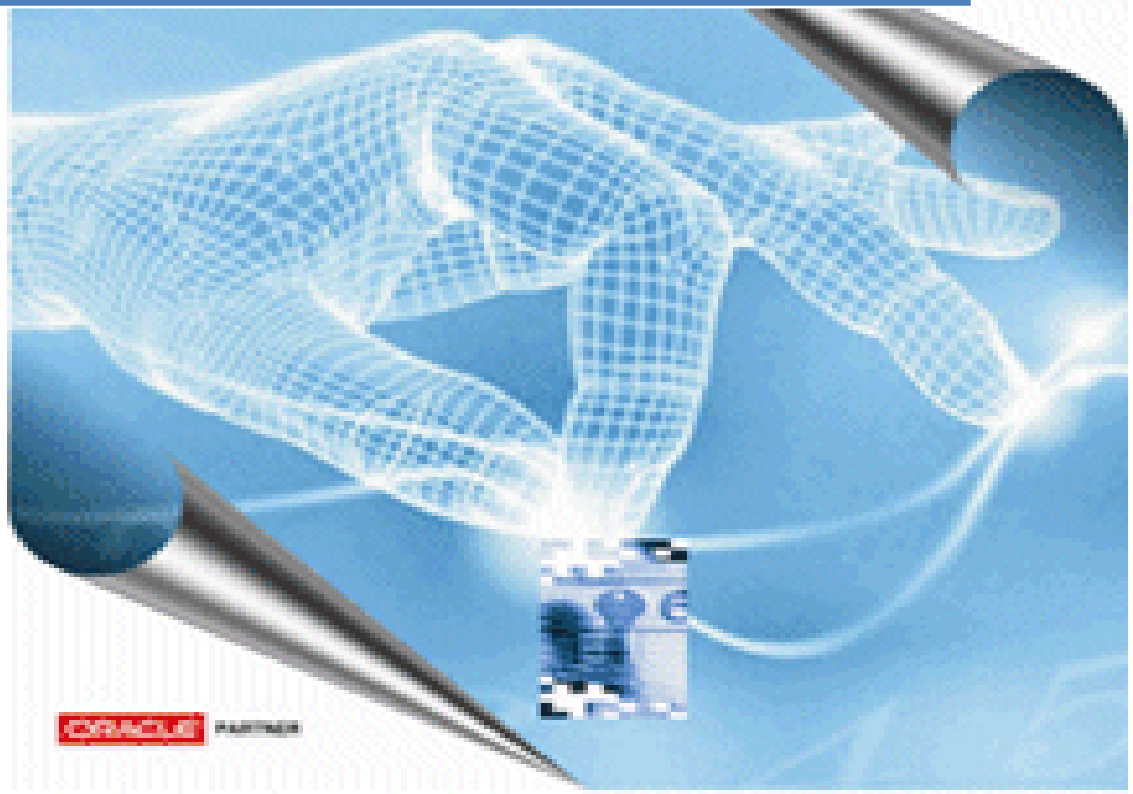
2. Ingresar en la opción de monitoreo de datos, desplegará la información solicitada en la configuración de datos,
3. Los datos básicos que se visualizarán son: el nombre de usuario y los nombres de las tablas que se han auditado y fueron escogidas en la configuración.
4. Se presentará la información resultado de la auditoría: el usuario, terminal, usuario de sistema operativo, fecha /hora, tipo de acción y tipo de valor (anterior y actual). A continuación se anexará un menú navegable con la información adicional solicitada en la configuración de datos.
5. Utilizar el botón de salida para volver al menú principal.

4.6.3 PRECONDICIONES

- Haber ingresado al sistema académico de la Universidad Técnica del Norte.
- Haber realizado la configuración de datos.

FASE DE PRUEBAS

Auditoría para base de datos Oracle - SABDO



ESPECIFICACIÓN DE CASOS DE PRUEBA

CAPÍTULO V

5 ESPECIFICACIÓN DE CASOS DE PRUEBAS

5.1 CASO DE PRUEBA: INICIALIZAR AUDITORÍA DE BDD ORACLE.

5.1.1 DESCRIPCIÓN BREVE

La única prueba que se puede realizar a este caso de uso es comprobar que el usuario realice la habilitación del parámetro audit trail.

5.1.2 COMPROBAR LA CONFIGURACIÓN DE LA BDD

Utilizando el usuario "system" y sus privilegios ingresar al OEM, accediendo a su funcionalidad y consultar si el parámetro audit trail ya consta inicializado en la base de datos, caso contrario ingresar los datos pertinentes, y se reiniciará la BDD.

5.1.3 CONDICIONES DE EJECUCIÓN

Las condiciones de ejecución del caso de prueba son que el usuario o DBA ha dado de alta la instancia y la base de datos. Se utilizará la herramienta Oracle Enterprise Manager OEM.

5.1.4 ENTRADA

- Introducimos 'system' en el campo usuario
- Introducimos '******' en el campo contraseña
- Escoger opción normal de la ventana desplegable
- Pulsamos entrar o el botón "conectar".
- Aparece la interfaz del OEM, donde pulsaremos la pestaña "Administración" y

seleccionamos la opción "Todos los parámetros de inicialización".

- Si el parámetro audit trail no se encuentra inicializado en la vista que desplegada, hacemos click en la pestaña "Spfile", buscar parámetro audit trail , marcarlo y en la columna "valor" escoger opción "true"
- Utilizamos el botón "aplicar".
- Volvemos a la pestaña "inicio" para reiniciar la BDD y salir del OEM.

5.1.5 RESULTADO ESPERADO

El sistema SABDO mostrará sus opciones de auditoría y las realizará sin enviar mensaje de error.

5.1.6 EVALUACIÓN DE LA PRUEBA

Prueba superada con éxito

5.2 CASO DE PRUEBA: CONFIGURAR AUDITORÍA DE SISTEMA.

5.2.1 DESCRIPCIÓN BREVE

La única prueba que se puede realizar a este caso de uso es hacer una configuración de auditoría de sistema para su posterior monitoreo.

5.2.2 COMPROBAR LA MANIPULACIÓN DE DATOS

Ingresar en el sistema como usuario "utndb", accediendo a su funcionalidad, escoger las opciones: auditoría de sistema y configuración, se ingresa los datos de la nueva configuración, y el sistema guardara la información.

5.2.3 CONDICIONES DE EJECUCIÓN

Las condiciones de ejecución del caso de prueba son que el usuario 'utndb' ingresó a configuración de auditoría de sistema. Las pruebas serán:

1. Configuración de Sesiones
2. Configuración de Acciones
3. Configuración de Objetos

5.2.4 ENTRADA

- En la ventana de Auditoría escoger la opción configuración de sistema
- Aparecen tres opciones: Sesiones, Acciones y Objetos

1. Configuración de Sesiones:
    - Hacer click en configuración de Sesiones
    - Se verá la operación connect, en las opciones de conexión escogemos "always" (todas las conexiones exitosas o no)
    - Pulsamos el botón "auditar" de la aplicación para iniciar la auditoría.
2. Configuración de Acciones:
    - Hacer click en configuración de Acciones
    - Ingresar objeto "procedure"
    - Del grupo de acciones sobre procedimiento desplegadas marcar para su auditoría: "Alter any procedure", "Debug any procedure" y "execute any procedure"
    - Pulsamos el botón "procesar" de la aplicación para iniciar la auditoría.
3. Configuración de Objetos:
    - Hacer click en configuración de Objetos
    - Ingresar usuario "utndb".

- En tipos de objetos ingresar “procedure”
- Hacer click en “cargar_foto”, es uno de los nombres de objetos visualizados en la ventana.
- En las acciones sobre los objetos que pueden ser auditadas marcar: ”Alter”, "Audit”, ”Grant”, ”Rename”, ”Execute”, ”Create”.
- Pulsamos el botón “procesar” de la aplicación para iniciar la auditoría.

- Utilizar botón de salida para volver al menú principal.

5.2.5 RESULTADO ESPERADO

En los tres casos, el sistema nos muestra las interfaces que consisten en una pantalla con diferentes pestañas para configurar las diferentes opciones de la auditoría de sistema.

5.2.6 EVALUACIÓN DE LA PRUEBA

Prueba superada con éxito

5.3 CASO DE PRUEBA: MONITOREAR LA AUDITORÍA DE SISTEMA.

5.3.1 DESCRIPCIÓN BREVE

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “monitoreo de sistema”. La única prueba que se puede realizar a este caso de uso es comprobar que la información mostrada en el monitoreo corresponda a la solicitada en la configuración de sistema.

5.3.2 COMPROBAR LA MANIPULACIÓN DE DATOS

Ingresar en el sistema como usuario "utndb", accediendo a su funcionalidad, escoger las opciones: auditoría de sistema y monitoreo. Realizar la verificación de la información.

5.3.3 CONDICIONES DE EJECUCIÓN

Las condiciones de ejecución del caso de prueba son que el usuario 'utndb' ingresó a monitoreo de auditoría de sistema. Las pruebas serán:

1. Visualizar información de monitoreo de Sesiones
2. Visualizar información de monitoreo de Objetos y Acciones

5.3.4 ENTRADA

- En la ventana de auditoría escogemos la opción de monitoreo de sistema.
- Aparecerán dos opciones: monitoreo de Sesiones y monitoreo de Objetos/Acciones

1. Monitoreo de Sesiones:
    - Hacer click en monitoreo de Sesiones
    - Se verá las operaciones tipo connect realizadas por el usuario utndb de:

      terminal: LILIAN-F8DD4AD3,

      Usuario S.O: LILIAN-F8DD4AD3\Lilian,

      Estado: conexión exitosa,

      Ingreso: 04/02/2009 20:27:04,

      Salida: 04/02/2009 20:30:52,

      Acción: Logoff,

      Userhost: GRUPO_TRABAJO,

Logoff Lread: 22724,

Logoff Pread: 5381,

Logoff Lwrite: 2495,

Logoff Dlock: 0,

Session id: 57864

Que son el resultado de la operación de conexión exitosa escogida en la configuración de auditoría. Se visualiza además las operaciones no exitosas.

2. Monitoreo de Acciones:

   - Hacer click en monitoreo de Acciones/Objetos.
   - Se verá las acciones sobre objetos realizadas por el usuario utndb de:

     terminal: LILIAN-F8DD4AD3,

     Usuario S.O: LILIAN-F8DD4AD3\Lilian,

     Userhost: GRUPO_TRABAJO,

     Timestamp: 04/02/2009,

     Owner: SYS,

     Obj Name: DBA_OBJ_AUDIT_OPT$,

     Action name: SESSION REC,

     Ses Actions: ----S----,

     Session id: 57864,

     Entryid: 35,

     Statementid: 195,

     Que son el resultado de las acciones realizadas sobre la base de datos y fue escogida en la configuración de auditoría.

- Utilizamos el botón salir para volver a la ventana principal.

5.3.5 RESULTADO ESPERADO

En los dos casos, el sistema nos muestra la información de auditoría, que consiste en una pantalla con datos básicos acompañados de un menú navegable que visualiza las diferentes opciones de la auditoría de sistema que fueron solicitadas.

5.3.6 EVALUACIÓN DE LA PRUEBA

Prueba superada con éxito

5.4 CASO DE PRUEBA: CONFIGURAR AUDITORÍA DE DATOS.

5.4.1 DESCRIPCIÓN BREVE

La única prueba que se puede realizar a este caso de uso es hacer una configuración de auditoría de datos para su posterior monitoreo.

5.4.2 COMPROBAR LA MANIPULACIÓN DE DATOS

Ingresar en el sistema como usuario "utndb", accediendo a su funcionalidad, escoger las opciones: auditoría de datos y configuración, se ingresa los datos de la nueva configuración, y el sistema guardara la información.

5.4.3 CONDICIONES DE EJECUCIÓN

Las condiciones de ejecución del caso de prueba son que el usuario 'utndb' ingresó a configuración de auditoría de datos.

5.4.4 ENTRADA

- En la ventana de Auditoría escoger la opción configuración de Datos
- Se despliega una ventana en donde ingresaremos el usuario "utndb"

- En la sección donde se despliegan los nombres de las tablas seleccionamos: "ACA_TAB_NOTAS" y marcamos las operaciones de inserción y actualización.
- En la sección donde se despliegan los nombres pertenecientes a las columnas de dicha tabla marcamos: "MATERIA_CODIGO", "MODA_ESTUD_CODIGO", "NIVEL_CODIGO", "NOTA1", "NOTA2".
- Pulsamos el botón "auditar" de la aplicación para iniciar la auditoría.

5.4.5 RESULTADO ESPERADO

El sistema nos muestra la interfaz que consiste en una pantalla con diferentes pestañas para configurar las diferentes opciones de la auditoría de datos.

5.4.6 EVALUACIÓN DE LA PRUEBA

Prueba superada con éxito

5.5 CASO DE PRUEBA: MONITOREAR LA AUDITORÍA DE DATOS.

5.5.1 DESCRIPCIÓN BREVE

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso "monitoreo de datos". La única prueba que se puede realizar a este caso de uso es comprobar que la información mostrada en el monitoreo corresponda a la solicitada en la configuración de datos.

5.5.2 COMPROBAR LA MANIPULACIÓN DE DATOS

Ingresar en el sistema como usuario "utndb", accediendo a su funcionalidad, escoger las opciones: auditoría de datos y monitoreo. Realizar la verificación de la información.

5.5.3 CONDICIONES DE EJECUCIÓN

Las condiciones de ejecución del caso de prueba son que el usuario 'utndb' ingresó a monitoreo de auditoría de datos y visualizará el contenido de las tablas, sus columnas y las operaciones que fueron realizadas en los datos.

5.5.4 ENTRADA

- En la ventana de auditoría escogemos la opción de monitoreo de datos.
- En la pestaña usuario escogemos: "utndb", aparecerá el nombre de la tabla auditada "ACA_TAB_NOTAS".
- En la sección inferior se visualizará una ventana con la siguiente información:

Usuario:	01100605052	01100605052
Usuario SO:	orass	orass
Fecha:	09/02/2009 17:12:43	09/02/2009 17:12:43
Acción:	UPDATE	UPDATE
Tipo Valor:	OLD	NEW
Aprobó:	N	S
Ciclo_acad_codigo:	1008-0309	1008-0309
Depen_codigo:	00147	00147
Docente_cedula:	1100605052	1100605052
Estudiante_cedula:	1003444989	1003444989
Inst_codigo:	001	001
Materia_codigo:	RB0708	RB0708
Nota1:	10	10
Nota2:		10

Que son el resultado de la operación de actualización escogida en la configuración de auditoría de datos. Se visualiza además las demás operaciones.

- Utilizamos el botón salir para volver a la ventana principal.

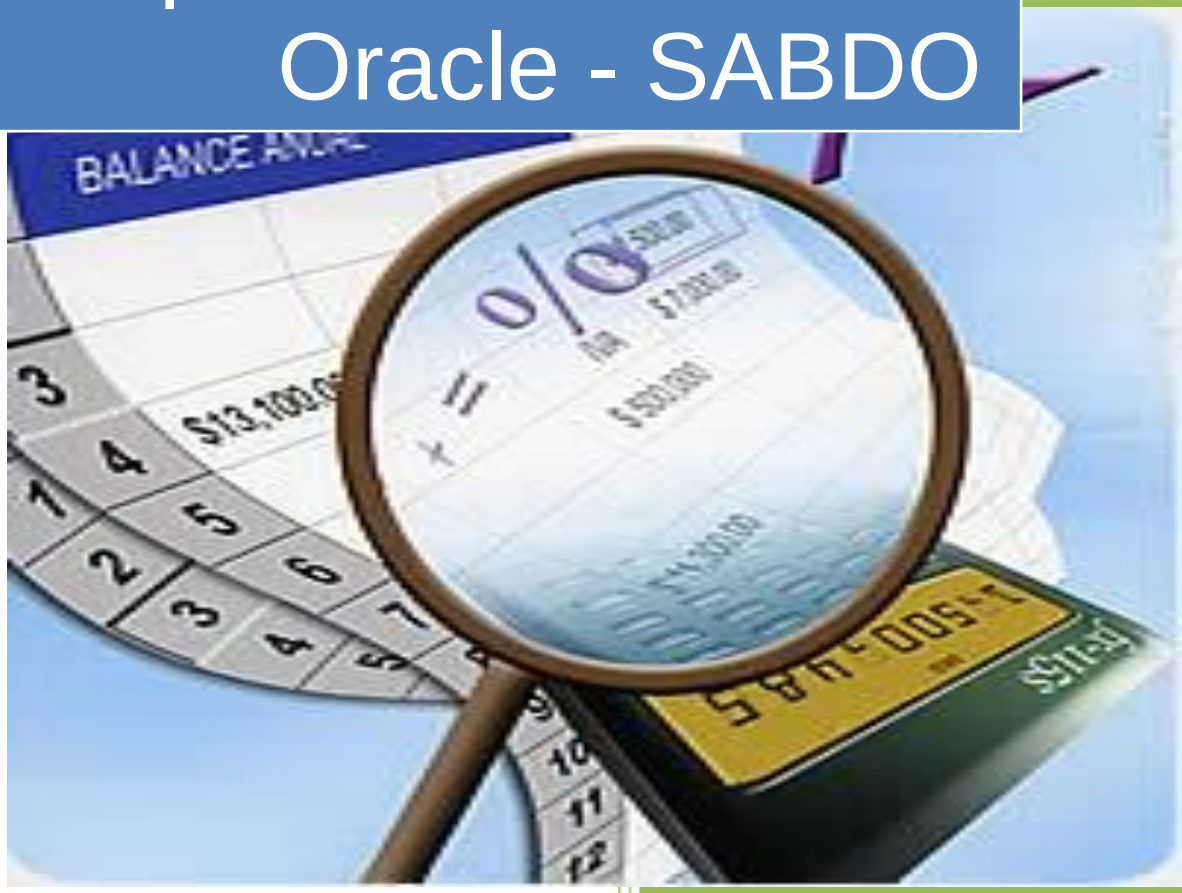
5.5.5 RESULTADO ESPERADO

El sistema nos muestra la información de auditoría, que consiste en una pantalla con datos básicos acompañados de un menú navegable que visualiza las diferentes opciones de la auditoría de datos que fueron solicitadas.

5.5.6 EVALUACIÓN DE LA PRUEBA

Prueba superada con éxito

Auditoría para base de datos Oracle - SABDO



LISTA DE RIESGOS

CAPÍTULO VI

6 LISTA DE RIESGOS

6.1 INTRODUCCIÓN

La lista de riesgos del proyecto es un compendio de acciones o razones por las cuales el proyecto puede experimentar retrasos para así poder establecer un plan de mitigación de riesgos, podrá ser modificada de acuerdo al avance del proyecto y será revisada periódicamente al menos una vez por iteración. A continuación se enumera y detalla cada uno de los riesgos encontrados y se adjuntan las respectivas recomendaciones:

6.1.1 DISEÑO INADECUADO.

El diseño del proyecto no es adecuado a la realidad de los procesos existentes.

Medidas de mitigación:

- Realizar un análisis profundo del problema con los integrantes del proyecto
- Tener reuniones constantes para analizar profundamente cada módulo asignado a cada integrante

6.1.2 DESMOTIVACIÓN DEL PERSONAL

Los miembros del proyecto no se encuentran debidamente motivados.

Medidas de mitigación:

- Realizar reuniones informales con el objetivo de unir al grupo.

- Realizar un seguimiento adecuado del trabajo realizado por cada miembro del proyecto.
- Evaluar el desempeño de los integrantes del proyecto.

6.1.3 FALTA DE CONFIGURACIÓN OPTIMA DE LA BASE DE DATOS ORACLE

La configuración de la Base de Datos no es la óptima, el desempeño de la misma depende de una buena distribución de los espacios de memoria, velocidad de respuesta, etc.

Medidas de mitigación:

- Hacer evaluaciones constantes del desempeño de la BDD.
- Hacer conocer al administrador de la BDD de la institución, sobre el desempeño de la BDD.

6.1.4 MOTIVACIÓN ECONÓMICA

Retrasos en la asignación de recursos económicos al grupo de desarrollo debido a trámites burocráticos.

Medida de mitigación:

- El jefe de proyecto realizará el seguimiento adecuado de los recursos económicos del proyecto.

6.1.5 FALTA DE POLÍTICAS Y REGLAMENTOS

Algunas actividades que se realizan en la institución no están normadas adecuadamente

Medidas de mitigación:

- Realizar un estudio adecuado de los reglamentos y políticas que rigen el normal desenvolvimiento de cada entidad
- Realizar un análisis profundo de los procesos en la BDD.
- En caso de existir procesos que se realizan y que no estén debidamente normadas, el grupo de desarrollo realizará la propuesta adecuada a las autoridades para que estas irregularidades se solucionen y que el proyecto se enmarque dentro de las normas adecuadas

6.1.6 FALTAS EXCESIVAS DE LOS MIEMBROS DEL PROYECTO

Los miembros del proyecto no se comprometen, y tienen muchas faltas al lugar de trabajo.

Medida de mitigación:

- Seguimiento constante de las actividades encomendadas a cada miembro del proyecto.

6.1.7 ADQUISICIÓN DE SERVIDORES Y EQUIPOS DE COMUNICACIONES RETRASADA

La adquisición de servidores y equipos de comunicación no se concreta.

Medidas de mitigación:

- El jefe de proyecto será el encargado de realizar el seguimiento y gestión para la adquisición del hardware necesario para el desarrollo e implementación del proyecto.
- En caso de faltar el hardware necesario, se improvisara configurando e implementando adecuadamente el hardware existente en el departamento.

6.1.8 FALTA DE PLANIFICACIÓN DEL PROYECTO

No se realiza la respectiva planificación del proyecto a desarrollarse.

Medidas de mitigación:

- El Jefe de proyecto realizará la planificación adecuada para el desarrollo del proyecto.
- Se realizará el seguimiento adecuado a cada miembro para que cada meta se cumpla en los tiempos previstos

6.1.9 LAS AUTORIDADES DESCONOCEN EL PROCESO DE INGENIERÍA DE SOFTWARE

Existe poco o ningún conocimiento por parte de las autoridades de la institución sobre la auditoría de la Base de Datos.

Medida de mitigación:

- Programar reuniones con las autoridades de institución, haciéndoles conocer la importancia del proyecto y el esfuerzo requerido para su implementación.
- Presentar simulaciones periódicas a las autoridades de la institución, las mismas que les dará una clara idea de la importancia del proyecto.

6.1.10 FALTA DE PARTICIPACIÓN DEL USUARIO PRIORIZACIÓN INADECUADA DE REQUERIMIENTOS

El usuario final no participa de una forma activa en el desarrollo del proyecto.

Medidas de mitigación:

- Planificar reuniones periódicas con los usuarios finales en momentos que no interrumpan su actividad diaria.
- Hacerles saber a los usuarios finales del sistema que cualquier aporte por parte de ellos es importante para el desarrollo del proyecto, con esto logramos la participación activa del usuario.

6.1.11 GESTIÓN INSUFICIENTE DE RIEGOS

No se realiza un análisis y seguimiento adecuado los riesgos presentes en el proyecto.

Medida de mitigación:

- Se realizará un análisis de riesgos adecuado lo más real posible, el mismo que permitirá saber la factibilidad del proyecto en cuanto a tiempo de desarrollo y costos.

6.1.12 PLANIFICACIÓN EXCESIVAMENTE OPTIMISTA

No se delimita adecuadamente el alcance del software a desarrollarse en el proyecto.

Medida de mitigación:

- Realizar el análisis adecuado y delimitar de una manera real el alcance que va a tener el software a desarrollarse

6.1.13 SÍNDROME DE LA HERRAMIENTA PANACEA

Las herramientas de desarrollo no cumplen con los requerimientos del grupo de trabajo.

Medidas de mitigación:

- Realizar un análisis profundo de las herramientas que se van a utilizar.

- Probar las herramientas en situaciones críticas de los procesos, donde verdaderamente se conozca del potencial de estas.

6.1.14 EXPECTATIVAS IRREALES

Creación de falsas expectativas en las autoridades de la institución.

Medidas de mitigación:

- Delimitar adecuadamente el alcance del proyecto.
- Planificar adecuadamente los tiempos de desarrollo.
- No crear falsas expectativas en los desarrolladores del proyecto, para evitar frustración en el caso de que no se lograr las metas previstas.

6.1.15 INCOMPATIBILIDAD DEL SOFTWARE CLIENTE

Incompatibilidad con la Base de Datos y la configuración de clientes finales.

Medida de mitigación:

- Instalar, configurar y realizar todas las pruebas necesarias en el software cliente.

Auditoría para base de datos Oracle - SABDO

CONCLUSIONES Y RECOMENDACIONES

CAPÍTULO VII

7 CONCLUSIONES Y RECOMENDACIONES

7.1 CONCLUSIONES

La complejidad de la tecnología de bases de datos y su entorno crecen aceleradamente por tanto su auditoría y control requieren de personal capacitado, para ello tenemos auditores informáticos, sin embargo el gran número de componentes de un entorno de SGBD y sus interfaces complican su trabajo, justificando la utilización de una herramienta versátil que permita facilitar su labor. SABDO fue realizado tomando estas consideraciones, convirtiendo los requerimientos en facilidades para el usuario.

La auditoría informática es un campo que aún no ha sido reconocido por su importancia, pero que con el pasar del tiempo y de desagradables experiencias va encontrando su lugar, acompañado de la respectiva inversión que las empresas e instituciones deben realizar.

Herramientas de Auditoría de Base de Datos para Oracle en el mercado actual pueden llegar a costar demasiado y se encuentran fuera de nuestro alcance, por tanto el desarrollo de SABDO representa una inversión y ahorro para la universidad.

7.2 RECOMENDACIONES

Para que SABDO pueda ser explotado al máximo podemos recomendar que se establezca un plan de seguridad y auditoría de base de datos, que contenga los lineamientos para el ingreso y acceso, una buena afinación de los recursos de la BDD, buen manejo de triggers, etc.

Configurar un esquema de seguridad valiéndose de las potencialidades que proveen el hardware y software de los servidores y motor de base de datos.

Auditoría para base de datos Oracle - SABDO

GLOSARIO DE TERMINOS

CAPÍTULO VIII

8 GLOSARIO

8.1 INTRODUCCIÓN

Este documento recoge todos y cada uno de los términos manejados a lo largo de todo el proyecto de desarrollo de un sistema para Auditoría de Base de Datos Oracle (SABDO). Se trata de un diccionario informal de datos y definiciones de la nomenclatura que se maneja, de tal modo que se crea un estándar para todo el proyecto.

8.1.1 PROPÓSITO.

El propósito de este glosario es definir con exactitud y sin ambigüedad la terminología manejada en el proyecto de desarrollo de un sistema para la gestión de artículos deportivos. También sirve como guía de consulta para la clarificación de los puntos conflictivos o poco esclarecedores del proyecto.

8.1.2 ALCANCE.

El alcance del presente documento se extiende a todos los subsistemas definidos para la auditoría de base de datos Oracle. De tal modo que la terminología empleada se refleja con claridad en este documento.

8.2 ORGANIZACIÓN DEL GLOSARIO

El presente documento está organizado por definiciones de términos ordenados de forma ascendente según la ordenación alfabética tradicional del español.

8.2.1 DEFINICIONES:

A continuación se presentan todos los términos manejados a lo largo de todo el proyecto SABDO.

- **AUDITORIA DE ACCIONES:** La auditoría de acciones comprende un registro detallado de las instrucciones utilizadas, que permiten el manejo de los datos contenidos en los objetos pertenecientes a la base de datos Oracle entre estos mencionaremos los siguientes: Create… Alter… Drop… Rebuild… Compile… Select… Insert… Update… Delete… etc; que crean, borran, alteran, insertan datos, tablas, índices, triggers, etc.
- **AUDITORIA DE BASES DE DATOS:** Es el proceso que permite medir, asegurar, demostrar, monitorear y registrar los accesos a la información almacenada en las bases de datos. incluyendo la capacidad de determinar: Quién accede a los datos, cuándo se accedió a los datos, desde qué tipo de dispositivo/aplicación, desde qué ubicación en la red, cuál fue la sentencia SQL ejecutada, cuál fue el efecto del acceso a la base de datos.
- **AUDITORIA DE CONEXIONES:** La auditoría de conexiones permite controlar las conexiones a la base de datos tanto las aceptadas como las rechazadas, manteniendo el detalle de los datos de los usuarios que las realizaron.
- **AUDITORIA DE DATOS:** La auditoría de datos representa la información detallada de los datos que han sido modificados, adicionados, borrados manteniendo un registro de los valores anteriores y actuales de los datos.
- **AUDITORIA DE OBJETOS:** La auditoría de objetos es un registro de los objetos de bases de datos como son: tablas, vistas, índices, constraints,

sinónimos, vistas materializadas, secuencias, procedimientos, funciones, paquetes, triggers propios de la base de datos a los que un usuario ha accesado.

- **BDD:** Base de Datos.
- **ISO:** Organización de estándares internacionales.
- **L4G:** Lenguajes de cuarta generación.
- **OEM:** Oracle Enterprise Manager. Herramienta de Oracle para configuración y mantenimiento de la base de datos.
- **Oracle:** Oracle es un sistema de gestión de base de datos relacional RDBMS (Relational Data Base Management System), fabricado por Oracle Corporation. La base de datos Oracle se ha mantenido como el líder en el manejo y seguridad de los datos.
- **RUP:** Son las siglas de Rational Unified Process. Se trata de una metodología para describir el proceso de desarrollo de software.
- **SGBD:** Sistemas de Gestión de Base de Datos.
- **SO:** Sistema Operativo.
- **TI:** Tecnología de la Información.

Auditoría para base de datos Oracle - SABDO

REFERENCIAS

CAPÍTULO IX

9 REFERENCIAS

- Glosario.
- Curso de Auditoría en Tecnología de la Información. Tema: Auditoría de Bases de Datos; Tomado por: Mario G. Piatini Yelthuis, Emilio de peso; Editorial Alfaomega, 1996; Universidad Nacional José Faustino Sánchez Carrión; Facultad de Ingeniería en Sistemas; Huacho-Perú.
- La Auditoría de Bases de Datos; Grau Bonet – The Eniac Corporation; grau.bonet@eniac-corp.com ; ISACA, San Juan, Noviembre 2007.
- RUP (Rational Unified Process).
- Publicación en web tema: Seguridad y auditoria de bases de datos Oracle, GSE – 32U.00. Universidad de Pamplona – Colombia.

ANEXOS

Auditoría para base de datos Oracle - SABDO

DICCIONARIO DE DATOS

GUÍA DE PROGRAMACIÓN

PROTOTIPO DE INTERFAZ DE USUARIO

MANUAL DE USUARIO

MANUAL DE INSTALACIÓN

ANEXO A: DICCIONARIO DE DATOS

El propósito de éste documento es recoger, analizar y definir las características de las sentencias que manejan las pistas de Auditoría (Audit Trail) de Oracle.Las tablas utilizadas por el sistema de auditoría de Base de Datos Oracle se enumeran a continuación:

- DBA_AUDIT_EXISTS
- DBA_AUDIT_OBJECT
- DBA_AUDIT_SESSION
- DBA_AUDIT_STATEMENT
- DBA_AUDIT_TRAIL

A.1 DBA_AUDIT_EXISTS

A.1.1 DEFINICIÓK

DBA_AUDIT_EXISTS visualiza entradas a pistas de auditoría producidas por AUDIT EXISTS y AUDIT NOT EXISTS. Ver tabla 9.

Columna	Tipo de Dato	NULL	Descripción
OS_USERNAME	VARCHAR2(255)		Cuenta de usuario de sistema operativo asociado al usuario de base de datos cuyas acciones fueron auditadas
USERNAME	VARCHAR2(30)		Nombre (no número de identificación) del usuario cuyas acciones fueron auditadas.
USERHOST	VARCHAR2(128)		Nombre de la máquina anfitrión del cliente.
TERMINAL	VARCHAR2(255)		Identificador del terminal del usuario.
TIMESTAMP	DATE		Fecha y hora de la creación de la entrada de la pista de auditoría

Columna	**Tipo de Dato**	**NULL**	**Descripción**
			(fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en el horario de sesión de base de datos local
OWNER	VARCHAR2(30)		Intención del creador del objeto inexistente
OBJ_NAME	VARCHAR2(128)		Nombre del objeto afectado por la acción
ACTION_NAME	VARCHAR2(28)		Nombre del tipo de acción correspondiente al código numérico en la columna ACTION en DBA_AUDIT_TRAIL.
NEW_OWNER	VARCHAR2(30)		Propietario del objeto nombrado en la columna NEW_NAME
NEW_NAME	VARCHAR2(128)		Nuevo nombre de un objeto después de un RENAME (renombrar) o el nombre del objeto subyacente.
OBJ_PRIVILEGE	VARCHAR2(16)		Privilegios de objeto concedidos o revocados por una sentencia GRANT o REVOKE
SYS_PRIVILEGE	VARCHAR2(40)		Privilegios de sistema concedidos o revocados por una declaración GRANT o REVOKE
GRANTEE	VARCHAR2(30)		Nombre de la concesión especificada en una declaración GRANT o REVOKE
SESSIONID	NUMBER	NOT NULL	Identificador numérico para cada sesión Oracle
ENTRYID	NUMBER	NOT NULL	Identificador numérico para cada entrada de pista de auditoría en la sesión.
STATEMENTID	NUMBER	NOT NULL	Identificador numérico para cada sentencia ejecutada.
RETURNCODE	NUMBER	NOT NULL	Código de error Oracle generado por la acción. Algunos valores útiles: • 0 - Acción sucedida • 2004 - Violación de

Columna	Tipo de Dato	NULL	Descripción
			seguridad
CLIENT_ID	VARCHAR2(64)		Identificador de cliente en cada sesión Oracle
SESSION_CPU	NUMBER		Cantidad de tiempo de procesador usado por cada sesión Oracle
EXTENDED_TIMESTAMP	TIMESTAMP(6) WITH TIME ZONE		Fecha y hora de la creación de la entrada de la pista de auditoría (fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en UTC (Coordenada Universal de Tiempo) uso horario.
PROXY_SESSIONID	NUMBER		Número serial de Sesión proxy, si un usuario empresarial ha sido registrado en el mecanismo proxy
GLOBAL_UID	VARCHAR2(32)		Identificador global de usuario para el usuario, si el usuario está registrado como un usuario empresarial
INSTANCE_NUMBER	NUMBER		Número de instancia especificado por el parámetro de inicialización INSTANCE_NUMBER
OS_PROCESS	VARCHAR2(16)		Identificador de proceso de sistema operativo para el proceso Oracle
TRANSACTIONID	RAW(8)		Identificador de transacción de la transacción en que el objeto es accesado o modificado.
SCN	NUMBER		Número de cambio de sistema (SCN) de la consulta.
SQL_BIND	NVARCHAR2(2000)		Unir variable de datos a la consulta
SQL_TEXT	NVARCHAR2(2000)		Texto SQL de la consulta.

Tabla 9 Parámetros Audit Exists

A.2 DBA_AUDIT_OBJECT

A.2.1 DEFINICIÓN

DBA_AUDIT_OBJECT visualiza las pistas de auditoría para todos los objetos en la base de datos.

A.2.2 CONSIDERACIONES RELACIONADAS

USER_AUDIT_OBJECT visualiza las pistas de auditoría para todos los objetos accesibles al usuario actual. Ver tabla 10.

Columna	Tipo de Dato	NULL	Descripción
OS_USERNAME	VARCHAR2(255)		Cuenta de usuario de sistema operativo asociado al usuario de base de datos cuyas acciones fueron auditadas
USERNAME	VARCHAR2(30)		Nombre (no número de identificación) del usuario cuyas acciones fueron auditadas.
USERHOST	VARCHAR2(128)		Nombre de la máquina anfitrión del cliente.
TERMINAL	VARCHAR2(255)		Identificador del terminal del usuario.
TIMESTAMP	DATE		Fecha y hora de la creación de la entrada de la pista de auditoría (fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en el horario de sesión de base de datos local
OWNER	VARCHAR2(30)		Creador del objeto afectado por la acción
OBJ_NAME	VARCHAR2(128)		Nombre del objeto afectado por la acción
ACTION_NAME	VARCHAR2(28)		Nombre del tipo de acción correspondiente al código numérico en la columna ACTION en DBA_AUDIT_TRAIL.

Columna	Tipo de Dato	NULL	Descripción
NEW_OWNER	VARCHAR2(30)		Propietario del objeto nombrado en la columna NEW_NAME
NEW_NAME	VARCHAR2(128)		Nuevo nombre de un objeto después de un RENAME (renombrar) o el nombre del objeto subyacente.
SES_ACTIONS	VARCHAR2(19)		Sesión resumen (una cadena de 16 caracteres, uno por cada tipo de acción en el orden: ALTER, AUDIT, COMMENT, DELETE, GRANT, INDEX, INSERT, LOCK, RENAME, SELECT, UPDATE, REFERENCES, y EXECUTE. Posiciones 14,15 y 16 son reservadas para uso futuro. Los caracteres son: - para ninguna, S para éxito, F para fracaso, and B para los dos).
COMMENT_TEXT	VARCHAR2(4000)		Comentarios de texto en la pista de auditoría.
SESSIONID	NUMBER	NOT NULL	Identificador numérico para cada sesión Oracle
ENTRYID	NUMBER	NOT NULL	Identificador numérico para cada entrada de pista de auditoría en la sesión.
STATEMENTID	NUMBER	NOT NULL	Identificador numérico para cada sentencia ejecutada.
RETURNCODE	NUMBER	NOT NULL	Código de error Oracle generado por la acción. Algunos valores útiles: • 0 - Acción sucedida • 2004 - Violación de seguridad
PRIV_USED	VARCHAR2(40)		Privilegio de sistema usado para ejecutar la acción.
CLIENT_ID	VARCHAR2(64)		Identificador de cliente en cada sesión Oracle
SESSION_CPU	NUMBER		Cantidad de tiempo de procesador usado por cada sesión Oracle
EXTENDED_TIMESTAMP	TIMESTAMP(6) WITH TIME ZONE		Fecha y hora de la creación de la entrada de la pista de auditoría (fecha y hora de conexión de usuario para entradas creadas por AUDIT

Columna	Tipo de Dato	NULL	Descripción
			SESSION) en UTC (Coordenada Universal de Tiempo) uso horario.
PROXY_SESSIONID	NUMBER		Número serial de Sesión proxy, si un usuario empresarial ha sido registrado en el mecanismo proxy
GLOBAL_UID	VARCHAR2(32)		Identificador global de usuario para el usuario, si el usuario está registrado como un usuario empresarial
INSTANCE_NUMBER	NUMBER		Número de instancia especificado por el parámetro de inicialización INSTANCE_NUMBER
OS_PROCESS	VARCHAR2(16)		Identificador de proceso de sistema operativo para el proceso Oracle
TRANSACTIONID	RAW(8)		Identificador de transacción de la transacción en que el objeto es accesado o modificado.
SCN	NUMBER		Número de cambio de sistema (SCN) de la consulta.
SQL_BIND	NVARCHAR2(2000)		Unir variable de datos a la consulta
SQL_TEXT	NVARCHAR2(2000)		Texto SQL de la consulta.

Tabla 10 Parámetros Audit Object

A.3 DBA_AUDIT_SESSION

A.3.1 DEFINICIÓN

DBA_AUDIT_SESION visualiza todos los registros de pistas de auditoría relacionadas con CONEXION y DESCONEXION.

A.3.2 CONSIDERACIONES RELACIONADAS

USER_AUDIT_SESSION contiene registros de pistas de auditoría relacionados a conexiones y desconexiones del usuario actual. Ver tabla 11.

Columna	Tipo de dato	NULL	Descripción
OS_USERNAME	VARCHAR2(255)		Cuenta de usuario de sistema operativo asociado al usuario de base de datos cuyas acciones fueron auditadas
USERNAME	VARCHAR2(30)		Nombre (no número de identificación) del usuario cuyas acciones fueron auditadas.
USERHOST	VARCHAR2(128)		Nombre de la máquina anfitrión del cliente.
TERMINAL	VARCHAR2(255)		Identificador del terminal del usuario.
TIMESTAMP	DATE		Fecha y hora de la creación de la entrada de la pista de auditoría (fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en el horario de sesión de base de datos local
ACTION_NAME	VARCHAR2(28)		Nombre del tipo de acción correspondiente al código numérico en la columna ACTION en DBA_AUDIT_TRAIL.
LOGOFF_TIME	DATE		Fecha y hora de cierre de sesión de usuario
LOGOFF_LREAD	NUMBER		Lectura lógica de la sesión.
LOGOFF_PREAD	NUMBER		Lectura física de la sesión.

Columna	Tipo de dato	NULL	Descripción
LOGOFF_LWRITE	NUMBER		Escritura lógica de la sesión.
LOGOFF_DLOCK	VARCHAR2(40)		Puntos muertos detectados durante la sesión.
SESSIONID	NUMBER	NOT NULL	Identificador numérico para cada sesión Oracle
RETURNCODE	NUMBER	NOT NULL	Código de error Oracle generado por la acción. Algunos valores útiles: • 0 - Acción sucedida • 2004 - Violación de seguridad
CLIENT_ID	VARCHAR2(64)		Identificador de cliente en cada sesión Oracle.
SESSION_CPU	NUMBER		Cantidad de tiempo de procesador usado por cada sesión Oracle.
EXTENDED_TIMESTAMP	TIMESTAMP(6) WITH TIME ZONE		Fecha y hora de la creación de la entrada de la pista de auditoría (fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en UTC (Coordenada Universal de Tiempo) uso horario.
PROXY_SESSIONID	NUMBER		Número serial de Sesión proxy, si un usuario empresarial ha sido registrado en el mecanismo proxy.
GLOBAL_UID	VARCHAR2(32)		Identificador global de usuario para el usuario, si el usuario está registrado como un usuario empresarial.
INSTANCE_NUMBER	NUMBER		Número de instancia especificado por el parámetro de inicialización INSTANCE_NUMBER
OS_PROCESS	VARCHAR2(16)		Identificador de proceso de sistema operativo para el proceso Oracle.

Tabla 11 Parámetros Audit Session

A.4 DBA_AUDIT_OBJECT

A.4.1 DEFINICIÓN

DBA_AUDIT_STATEMENT visualiza registros de pistas de auditoría concernientes a sentencias GRANT, REVOKE, AUDIT, NOAUDIT, y declaraciones ALTER SYSTEM a través de la base de datos.

A.4.2 CONSIDERACIONES RELACIONADAS

USER_AUDIT_STATEMENT visualiza registros de pistas de auditoría para las mismas declaraciones emitidas por el actual usuario. Ver tabla 12.

Columna	Tipo de Dato	NULL	Descripción
OS_USERNAME	VARCHAR2(255)		Cuenta de usuario de sistema operativo asociado al usuario de base de datos cuyas acciones fueron auditadas
USERNAME	VARCHAR2(30)		Nombre (no número de identificación) del usuario cuyas acciones fueron auditadas.
USERHOST	VARCHAR2(128)		Nombre de la máquina anfitrión del cliente.
TERMINAL	VARCHAR2(255)		Identificador del terminal del usuario.
TIMESTAMP	DATE		Fecha y hora de la creación de la entrada de la pista de auditoría (fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en el horario de sesión de base de datos local
OWNER	VARCHAR2(30)		Creador del objeto afectado por la acción
OBJ_NAME	VARCHAR2(128)		Nombre del objeto afectado por la acción.
ACTION_NAME	VARCHAR2(28)		Nombre del tipo de acción correspondiente al código numérico en la columna ACTION en

Columna	Tipo de Dato	NULL	Descripción
			DBA_AUDIT_TRAIL.
NEW_NAME	VARCHAR2(128)		Nuevo nombre de un objeto después de un RENAME (renombrar) o el nombre del objeto subyacente.
OBJ_PRIVILEGE	VARCHAR2(16)		Privilegios de objeto concedidos o revocados por una sentencia GRANT o REVOKE
SYS_PRIVILEGE	VARCHAR2(40)		Privilegios de sistema concedidos o revocados por una declaración GRANT o REVOKE
ADMIN_OPTION	VARCHAR2(1)		Indica el rol o privilegio de sistema concedido con la opción ADMIN.
GRANTEE	VARCHAR2(30)		Nombre de la concesión especificada en una declaración GRANT o REVOKE
AUDIT_OPTION	VARCHAR2(40)		Auditando la opción colocada con la declaración AUDIT.
SES_ACTIONS	VARCHAR2(19)		Sesión resumen (una cadena de 16 caracteres, uno por cada tipo de acción en el orden: ALTER, AUDIT, COMMENT, DELETE, GRANT, INDEX, INSERT, LOCK, RENAME, SELECT, UPDATE, REFERENCES, y EXECUTE. Posiciones 14,15 y 16 son reservadas para uso futuro. Los caracteres son: - para ninguna, S para éxito, F para fracaso, and B para los dos).
COMMENT_TEXT	VARCHAR2(4000)		Comentarios de texto sobre la pista de auditoría, insertada por la aplicación.
SESSIONID	NUMBER	NOT NULL	Identificador numérico para cada sesión Oracle
ENTRYID	NUMBER	NOT NULL	Identificador numérico para cada entrada de pista de auditoría en la sesión.
STATEMENTID	NUMBER	NOT NULL	Identificador numérico para cada sentencia ejecutada.

Columna	Tipo de Dato	NULL	Descripción
RETURNCODE	NUMBER	NOT NULL	Código de error Oracle generado por la acción. Algunos valores útiles: • 0 - Acción sucedida • 2004 - Violación de seguridad
PRIV_USED	VARCHAR2(40)		Privilegio de sistema usado para ejecutar la acción.
CLIENT_ID	VARCHAR2(64)		Identificador de cliente en cada sesión Oracle
SESSION_CPU	NUMBER		Cantidad de tiempo de procesador usado por cada sesión Oracle
EXTENDED_TIMESTAMP	TIMESTAMP(6) WITH TIME ZONE		Fecha y hora de la creación de la entrada de la pista de auditoría (fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en UTC (Coordenada Universal de Tiempo) uso horario.
PROXY_SESSIONID	NUMBER		Número serial de Sesión proxy, si un usuario empresarial ha sido registrado en el mecanismo proxy
GLOBAL_UID	VARCHAR2(32)		Identificador global de usuario para el usuario, si el usuario está registrado como un usuario empresarial
INSTANCE_NUMBER	NUMBER		Número de instancia especificado por el parámetro de inicialización INSTANCE_NUMBER
OS_PROCESS	VARCHAR2(16)		Identificador de proceso de sistema operativo para el proceso Oracle
TRANSACTIONID	RAW(8)		Identificador de transacción de la transacción en que el objeto es accesado o modificado.
SCN	NUMBER		Número de cambio de sistema (SCN) de la consulta.
SQL_BIND	NVARCHAR2(2000)		Unir variable de datos a la consulta.
SQL_TEXT	NVARCHAR2(2000)		Texto SQL de la consulta.

Tabla 12 Parámetros Audit Statement

A.5 DBA_AUDIT_TRAIL

A.5.1 DEFINICIÓN

DBA_AUDIT_TRAIL visualiza todas las entradas a pistas de auditoría.

A.5.2 CONSIDERACIONES RELACIONADAS

USER_AUDIT_TRAIL visualiza todas las entradas a pistas de auditoría relacionadas al usuario actual.Ver Tabla 13.

Columna	Tipo de Dato	NULL	Descripción
OS_USERNAME	VARCHAR2(255)		Cuenta de usuario de sistema operativo asociado al usuario de base de datos cuyas acciones fueron auditadas
USERNAME	VARCHAR2(30)		Nombre (no número de identificación) del usuario cuyas acciones fueron auditadas.
USERHOST	VARCHAR2(128)		Nombre de la máquina anfitrión del cliente.
TERMINAL	VARCHAR2(255)		Identificador del terminal del usuario.
TIMESTAMP	DATE		Fecha y hora de la creación de la entrada de la pista de auditoría (fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en el horario de sesión de base de datos local
OWNER	VARCHAR2(30)		Creador del objeto afectado por la acción
OBJ_NAME	VARCHAR2(128)		Nombre del objeto afectado por la acción
ACTION	NUMBER	NOT NULL	Código numérico del tipo de acción. El nombre correspondiente al tipo de acción está en la columna ACTION_NAME.
ACTION_NAME	VARCHAR2(28)		Nombre del tipo de acción correspondiente al código numérico

Columna	Tipo de Dato	NULL	Descripción
			en la columna ACTION
NEW_OWNER	VARCHAR2(30)		Propietario del objeto referenciado en la columna NEW_NAME.
NEW_NAME	VARCHAR2(128)		Nuevo nombre de un objeto después de un RENAME (renombrar) o el nombre del objeto subyacente.
OBJ_PRIVILEGE	VARCHAR2(16)		Privilegios de objeto concedidos o revocados por una sentencia GRANT o REVOKE
SYS_PRIVILEGE	VARCHAR2(40)		Privilegios de sistema concedidos o revocados por una declaración GRANT o REVOKE
ADMIN_OPTION	VARCHAR2(1)		Indica el rol o privilegio de sistema concedido con la opción ADMIN.
GRANTEE	VARCHAR2(30)		Nombre de la concesión especificada en una declaración GRANT o REVOKE
AUDIT_OPTION	VARCHAR2(40)		Auditando la opción colocada con la declaración AUDIT
SES_ACTIONS	VARCHAR2(19)		Sesión resumen (una cadena de 16 caracteres, uno por cada tipo de acción en el orden: ALTER, AUDIT, COMMENT, DELETE, GRANT, INDEX, INSERT, LOCK, RENAME, SELECT, UPDATE, REFERENCES, y EXECUTE. Posiciones 14,15 y 16 son reservadas para uso futuro. Los caracteres son: • - - Ninguna • S - Éxito • F - Fracaso • B – Ambos
LOGOFF_TIME	DATE		Fecha y hora de cierre de sesión de usuario
LOGOFF_LREAD	NUMBER		Lectura lógica de la sesión
LOGOFF_PREAD	NUMBER		Lectura física de la sesión
LOGOFF_LWRITE	NUMBER		Escritura lógica de la sesión.

Columna	Tipo de Dato	NULL	Descripción
LOGOFF_DLOCK	VARCHAR2(40)		Puntos muertos detectados durante la sesión
COMMENT_TEXT	VARCHAR2(4000)		Comentarios de texto sobre la entrada de la pista de auditoría, proporcionando más información acerca de la sentencia auditada. También indica como el usuario fue autenticado, el método puede ser uno de los siguientes: • DATABASE - Autenticación fue hecha por contraseña (password) • NETWORK - Autenticación fue hecha por servicios Oracle Net o la opción Seguridad Avanzada. • PROXY – Cliente fue autenticado por otro usuario; el nombre del usuario proxy sigue el método tipo.
SESSIONID	NUMBER	NOT NULL	Identificador numérico para cada sesión Oracle
ENTRYID	NUMBER	NOT NULL	Identificador numérico para cada entrada de pista de auditoría en la sesión.
STATEMENTID	NUMBER	NOT NULL	Identificador numérico para cada sentencia ejecutada.
RETURNCODE	NUMBER	NOT NULL	Código de error Oracle generado por la acción. Algunos valores útiles: • 0 - Acción sucedida • 2004 - Violación de seguridad
PRIV_USED	VARCHAR2(40)		Privilegio de sistema usado para ejecutar la acción.
CLIENT_ID	VARCHAR2(64)		Identificador de cliente en cada sesión Oracle
SESSION_CPU	NUMBER		Cantidad de tiempo de procesador usado por cada sesión Oracle
EXTENDED_TIMESTAMP	TIMESTAMP(6)		Fecha y hora de la creación de la entrada de la pista de auditoría

Columna	Tipo de Dato	NULL	Descripción
	WITH TIME ZONE		(fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION) en UTC (Coordenada Universal de Tiempo) uso horario.
PROXY_SESSIONID	NUMBER		Número serial de Sesión proxy, si un usuario empresarial ha sido registrado en el mecanismo proxy
GLOBAL_UID	VARCHAR2(32)		Identificador global de usuario para el usuario, si el usuario está registrado como un usuario empresarial
INSTANCE_NUMBER	NUMBER		Número de instancia especificado por el parámetro de inicialización INSTANCE_NUMBER
OS_PROCESS	VARCHAR2(16)		Identificador de proceso de sistema operativo para el proceso Oracle
TRANSACTIONID	RAW(8)		Identificador de transacción de la transacción en que el objeto es accesado o modificado.
SCN	NUMBER		Número de cambio de sistema (SCN) de la consulta.
SQL_BIND	NVARCHAR2(2000)		Unir variable de datos a la consulta
SQL_TEXT	NVARCHAR2(2000)		Texto SQL de la consulta.

Tabla 13 Parámetros Audit Trail

A.6 SENTENCIAS DE CREACIÓN DE TABLAS Y VISTAS.

```
CREATE TABLE AUD$
(
  SESSIONID        NUMBER                         NOT NULL,
  ENTRYID          NUMBER                         NOT NULL,
  STATEMENT        NUMBER                         NOT NULL,
  TIMESTAMP#       DATE,
  USERID           VARCHAR2(30 BYTE),
  USERHOST         VARCHAR2(128 BYTE),
  TERMINAL         VARCHAR2(255 BYTE),
  ACTION#          NUMBER                         NOT NULL,
  RETURNCODE       NUMBER                         NOT NULL,
  OBJ$CREATOR      VARCHAR2(30 BYTE),
  OBJ$NAME         VARCHAR2(128 BYTE),
  AUTH$PRIVILEGES  VARCHAR2(16 BYTE),
  AUTH$GRANTEE     VARCHAR2(30 BYTE),
  NEW$OWNER        VARCHAR2(30 BYTE),
  NEW$NAME         VARCHAR2(128 BYTE),
  SES$ACTIONS      VARCHAR2(19 BYTE),
  SES$TID          NUMBER,
  LOGOFF$LREAD     NUMBER,
  LOGOFF$PREAD     NUMBER,
  LOGOFF$LWRITE    NUMBER,
  LOGOFF$DEAD      NUMBER,
  LOGOFF$TIME      DATE,
  COMMENT$TEXT     VARCHAR2(4000 BYTE),
  CLIENTID         VARCHAR2(64 BYTE),
  SPARE1           VARCHAR2(255 BYTE),
  SPARE2           NUMBER,
  OBJ$LABEL        RAW(255),
  SES$LABEL        RAW(255),
  PRIV$USED        NUMBER,
  SESSIONCPU       NUMBER,
  NTIMESTAMP#      TIMESTAMP(6),
  PROXY$SID        NUMBER,
  USER$GUID        VARCHAR2(32 BYTE),
  INSTANCE#        NUMBER,
  PROCESS#         VARCHAR2(16 BYTE),
  XID              RAW(8),
  AUDITID          VARCHAR2(64 BYTE),
  SCN              NUMBER,
  DBID             NUMBER,
  SQLBIND          CLOB,
  SQLTEXT          CLOB
)
TABLESPACE SYSTEM
PCTUSED    40
PCTFREE    10
```

```
INITRANS   1
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
LOGGING
NOCOMPRESS
LOB (SQLBIND) STORE AS
      ( TABLESPACE  SYSTEM
        ENABLE      STORAGE IN ROW
        CHUNK       8192
        PCTVERSION  0
        NOCACHE
        STORAGE    (
                    INITIAL          64K
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    FREELISTS        1
                    FREELIST GROUPS  1
                    BUFFER_POOL      DEFAULT
                   )
      )
  LOB (SQLTEXT) STORE AS
      ( TABLESPACE  SYSTEM
        ENABLE      STORAGE IN ROW
        CHUNK       8192
        PCTVERSION  0
        NOCACHE
        STORAGE    (
                    INITIAL          64K
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    FREELISTS        1
                    FREELIST GROUPS  1
                    BUFFER_POOL      DEFAULT
                   )
      )
NOCACHE
NOPARALLEL
MONITORING;
```

```
CREATE INDEX I_AUD1 ON AUD$
(SESSIONID, SES$TID)
LOGGING
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;


GRANT DELETE ON  AUD$ TO DELETE_CATALOG_ROLE;


CREATE TABLE AUDIT_ACTIONS
(
  ACTION  NUMBER                                NOT NULL,
  NAME    VARCHAR2(28 BYTE)                     NOT NULL
)
TABLESPACE SYSTEM
PCTUSED    40
PCTFREE    10
INITRANS   1
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
LOGGING
NOCOMPRESS
NOCACHE
NOPARALLEL
MONITORING;

COMMENT ON TABLE AUDIT_ACTIONS IS 'Description table for audit trail
action type codes.  Maps action type numbers to action type names';
```

```
COMMENT ON COLUMN AUDIT_ACTIONS.ACTION IS 'Numeric audit trail action
type code';

COMMENT ON COLUMN AUDIT_ACTIONS.NAME IS 'Name of the type of audit trail
action';


CREATE UNIQUE INDEX I_AUDIT_ACTIONS ON AUDIT_ACTIONS
(ACTION, NAME)
LOGGING
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;


CREATE PUBLIC SYNONYM AUDIT_ACTIONS FOR AUDIT_ACTIONS;


GRANT SELECT ON  AUDIT_ACTIONS TO PUBLIC;


CREATE TABLE OBJ$
(
  OBJ#          NUMBER                          NOT NULL,
  DATAOBJ#      NUMBER,
  OWNER#        NUMBER                          NOT NULL,
  NAME          VARCHAR2(30 BYTE)               NOT NULL,
  NAMESPACE     NUMBER                          NOT NULL,
  SUBNAME       VARCHAR2(30 BYTE),
  TYPE#         NUMBER                          NOT NULL,
  CTIME         DATE                            NOT NULL,
  MTIME         DATE                            NOT NULL,
  STIME         DATE                            NOT NULL,
  STATUS        NUMBER                          NOT NULL,
  REMOTEOWNER   VARCHAR2(30 BYTE),
  LINKNAME      VARCHAR2(128 BYTE),
  FLAGS         NUMBER,
  OID$          RAW(16),
  SPARE1        NUMBER,
```

```
  SPARE2        NUMBER,
  SPARE3        NUMBER,
  SPARE4        VARCHAR2(1000 BYTE),
  SPARE5        VARCHAR2(1000 BYTE),
  SPARE6        DATE
)
TABLESPACE SYSTEM
PCTUSED    40
PCTFREE    10
INITRANS   1
MAXTRANS   255
STORAGE    (
            INITIAL          16K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
LOGGING
NOCOMPRESS
NOCACHE
NOPARALLEL
MONITORING;


CREATE UNIQUE INDEX I_OBJ1 ON OBJ$
(OBJ#)
LOGGING
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;


CREATE UNIQUE INDEX I_OBJ2 ON OBJ$
(OWNER#, NAME, NAMESPACE, REMOTEOWNER, LINKNAME,
SUBNAME)
LOGGING
```

```
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          16K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;

CREATE INDEX I_OBJ3 ON OBJ$
(OID$)
LOGGING
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;


GRANT SELECT ON  OBJ$ TO CTXSYS WITH GRANT OPTION;

GRANT SELECT ON  OBJ$ TO OLAPSYS WITH GRANT OPTION;


CREATE TABLE STMT_AUDIT_OPTION_MAP
(
  OPTION#    NUMBER                              NOT NULL,
  NAME       VARCHAR2(40 BYTE)                   NOT NULL,
  PROPERTY   NUMBER                              NOT NULL
)
TABLESPACE SYSTEM
PCTUSED    40
PCTFREE    10
INITRANS   1
MAXTRANS   255
```

```
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
LOGGING
NOCOMPRESS
NOCACHE
NOPARALLEL
MONITORING;

COMMENT ON TABLE STMT_AUDIT_OPTION_MAP IS 'Description table for
auditing option type codes.  Maps auditing option type numbers to type
names';

COMMENT ON COLUMN STMT_AUDIT_OPTION_MAP.OPTION# IS 'Numeric auditing
option type code';

COMMENT ON COLUMN STMT_AUDIT_OPTION_MAP.NAME IS 'Name of the type of
auditing option';

COMMENT ON COLUMN STMT_AUDIT_OPTION_MAP.PROPERTY IS 'Property flag of
auditing option';


CREATE UNIQUE INDEX I_STMT_AUDIT_OPTION_MAP ON STMT_AUDIT_OPTION_MAP
(OPTION#, NAME)
LOGGING
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;

CREATE PUBLIC SYNONYM STMT_AUDIT_OPTION_MAP FOR STMT_AUDIT_OPTION_MAP;

GRANT SELECT ON  STMT_AUDIT_OPTION_MAP TO PUBLIC;
```

```
CREATE TABLE SYSTEM_PRIVILEGE_MAP
(
  PRIVILEGE  NUMBER                                 NOT NULL,
  NAME       VARCHAR2(40 BYTE)                      NOT NULL,
  PROPERTY   NUMBER                                 NOT NULL
)
TABLESPACE SYSTEM
PCTUSED    40
PCTFREE    10
INITRANS   1
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
LOGGING
NOCOMPRESS
NOCACHE
NOPARALLEL
MONITORING;

COMMENT ON TABLE SYSTEM_PRIVILEGE_MAP IS 'Description table for
privilege type codes.  Maps privilege  type numbers to type names';
COMMENT ON COLUMN SYSTEM_PRIVILEGE_MAP.PRIVILEGE IS 'Numeric privilege
type code';
COMMENT ON COLUMN SYSTEM_PRIVILEGE_MAP.NAME IS 'Name of the type of
privilege';
COMMENT ON COLUMN SYSTEM_PRIVILEGE_MAP.PROPERTY IS 'Property flag of
privilege like not export this privilege, etc';

CREATE UNIQUE INDEX I_SYSTEM_PRIVILEGE_MAP ON SYSTEM_PRIVILEGE_MAP
(PRIVILEGE, NAME)
LOGGING
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
```

```
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;
CREATE PUBLIC SYNONYM SYSTEM_PRIVILEGE_MAP FOR SYSTEM_PRIVILEGE_MAP;
GRANT SELECT ON  SYSTEM_PRIVILEGE_MAP TO PUBLIC WITH GRANT OPTION;


CREATE TABLE TAB$
(
  OBJ#          NUMBER                          NOT NULL,
  DATAOBJ#      NUMBER,
  TS#           NUMBER                          NOT NULL,
  FILE#         NUMBER                          NOT NULL,
  BLOCK#        NUMBER                          NOT NULL,
  BOBJ#         NUMBER,
  TAB#          NUMBER,
  COLS          NUMBER                          NOT NULL,
  CLUCOLS       NUMBER,
  PCTFREE$      NUMBER                          NOT NULL,
  PCTUSED$      NUMBER                          NOT NULL,
  INITRANS      NUMBER                          NOT NULL,
  MAXTRANS      NUMBER                          NOT NULL,
  FLAGS         NUMBER                          NOT NULL,
  AUDIT$        VARCHAR2(38 BYTE)               NOT NULL,
  ROWCNT        NUMBER,
  BLKCNT        NUMBER,
  EMPCNT        NUMBER,
  AVGSPC        NUMBER,
  CHNCNT        NUMBER,
  AVGRLN        NUMBER,
  AVGSPC_FLB    NUMBER,
  FLBCNT        NUMBER,
  ANALYZETIME   DATE,
  SAMPLESIZE    NUMBER,
  DEGREE        NUMBER,
  INSTANCES     NUMBER,
  INTCOLS       NUMBER                          NOT NULL,
  KERNELCOLS    NUMBER                          NOT NULL,
  PROPERTY      NUMBER                          NOT NULL,
  TRIGFLAG      NUMBER,
  SPARE1        NUMBER,
  SPARE2        NUMBER,
  SPARE3        NUMBER,
  SPARE4        VARCHAR2(1000 BYTE),
  SPARE5        VARCHAR2(1000 BYTE),
  SPARE6        DATE
)
CLUSTER C_OBJ#(OBJ#)
NOCOMPRESS ;
```

```
CREATE INDEX I_TAB1 ON TAB$
(BOBJ#)
LOGGING
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;
GRANT SELECT ON  TAB$ TO CTXSYS WITH GRANT OPTION;

CREATE TABLE USER$
(
  USER#          NUMBER                          NOT NULL,
  NAME           VARCHAR2(30 BYTE)               NOT NULL,
  TYPE#          NUMBER                          NOT NULL,
  PASSWORD       VARCHAR2(30 BYTE),
  DATATS#        NUMBER                          NOT NULL,
  TEMPTS#        NUMBER                          NOT NULL,
  CTIME          DATE                            NOT NULL,
  PTIME          DATE,
  EXPTIME        DATE,
  LTIME          DATE,
  RESOURCE$      NUMBER                          NOT NULL,
  AUDIT$         VARCHAR2(38 BYTE),
  DEFROLE        NUMBER                          NOT NULL,
  DEFGRP#        NUMBER,
  DEFGRP_SEQ#    NUMBER,
  ASTATUS           NUMBER                                    DEFAULT  0
NOT NULL,
  LCOUNT             NUMBER                                   DEFAULT  0
NOT NULL,
  DEFSCHCLASS    VARCHAR2(30 BYTE),
  EXT_USERNAME   VARCHAR2(4000 BYTE),
  SPARE1         NUMBER,
  SPARE2         NUMBER,
  SPARE3         NUMBER,
  SPARE4         VARCHAR2(1000 BYTE),
  SPARE5         VARCHAR2(1000 BYTE),
  SPARE6         DATE
)
CLUSTER C_USER#(USER#)
```

```
NOCOMPRESS ;


CREATE UNIQUE INDEX I_USER1 ON USER$
(NAME)
LOGGING
TABLESPACE SYSTEM
PCTFREE    10
INITRANS   2
MAXTRANS   255
STORAGE    (
            INITIAL          64K
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      0
            FREELISTS        1
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
NOPARALLEL;

GRANT SELECT ON  USER$ TO CTXSYS WITH GRANT OPTION;
GRANT SELECT ON  USER$ TO OLAPSYS WITH GRANT OPTION;
GRANT SELECT ON  USER$ TO XDB;


CREATE OR REPLACE VIEW DBA_AUDIT_EXISTS
(OS_USERNAME, USERNAME, USERHOST, TERMINAL, TIMESTAMP,
 OWNER, OBJ_NAME, ACTION_NAME, NEW_OWNER, NEW_NAME,
 OBJ_PRIVILEGE, SYS_PRIVILEGE, GRANTEE, SESSIONID, ENTRYID,
 STATEMENTID, RETURNCODE, CLIENT_ID, ECONTEXT_ID, SESSION_CPU,
 EXTENDED_TIMESTAMP,    PROXY_SESSIONID,    GLOBAL_UID,    INSTANCE_NUMBER,
OS_PROCESS,
 TRANSACTIONID, SCN, SQL_BIND, SQL_TEXT)
AS
select os_username, username, userhost, terminal, timestamp,
         owner, obj_name,
         action_name,
         new_owner,
         new_name,
         obj_privilege, sys_privilege, grantee,
         sessionid, entryid, statementid, returncode, client_id,
         econtext_id, session_cpu,
         extended_timestamp,         proxy_sessionid,         global_uid,
instance_number,
         os_process, transactionid, scn, sql_bind, sql_text
  from dba_audit_trail
  where returncode in
  (942, 943, 959, 1418, 1432, 1434, 1435, 1534, 1917, 1918, 1919, 2019,
   2024, 2289,
```

```
   4042, 4043, 4080, 1, 951, 955, 957, 1430, 1433, 1452, 1471, 1535,
1543,
   1758, 1920, 1921, 1922, 2239, 2264, 2266, 2273, 2292, 2297, 2378,
2379,
   2382, 4081, 12006, 12325)

/


CREATE OR REPLACE VIEW DBA_AUDIT_OBJECT
(OS_USERNAME, USERNAME, USERHOST, TERMINAL, TIMESTAMP,
 OWNER, OBJ_NAME, ACTION_NAME, NEW_OWNER, NEW_NAME,
 SES_ACTIONS, COMMENT_TEXT, SESSIONID, ENTRYID, STATEMENTID,
 RETURNCODE, PRIV_USED, CLIENT_ID, ECONTEXT_ID, SESSION_CPU,
 EXTENDED_TIMESTAMP,    PROXY_SESSIONID,    GLOBAL_UID,    INSTANCE_NUMBER,
OS_PROCESS,
 TRANSACTIONID, SCN, SQL_BIND, SQL_TEXT)
AS
select OS_USERNAME, USERNAME, USERHOST, TERMINAL, TIMESTAMP,
       OWNER, OBJ_NAME, ACTION_NAME, NEW_OWNER, NEW_NAME,
       SES_ACTIONS, COMMENT_TEXT, SESSIONID, ENTRYID, STATEMENTID,
       RETURNCODE, PRIV_USED, CLIENT_ID, ECONTEXT_ID, SESSION_CPU,
       EXTENDED_TIMESTAMP, PROXY_SESSIONID, GLOBAL_UID, INSTANCE_NUMBER,
       OS_PROCESS, TRANSACTIONID, SCN, SQL_BIND, SQL_TEXT
from dba_audit_trail
where (action between 1 and 16)
   or (action between 19 and 29)
   or (action between 32 and 41)
   or (action = 43)
   or (action between 51 and 99)
   or (action = 103)
   or (action between 110 and 113)
   or (action between 116 and 121)
   or (action between 123 and 128)
   or (action between 160 and 162)
/
CREATE OR REPLACE VIEW DBA_AUDIT_SESSION
(OS_USERNAME, USERNAME, USERHOST, TERMINAL, TIMESTAMP,
 ACTION_NAME, LOGOFF_TIME, LOGOFF_LREAD, LOGOFF_PREAD, LOGOFF_LWRITE,
 LOGOFF_DLOCK, SESSIONID, RETURNCODE, CLIENT_ID, SESSION_CPU,
 EXTENDED_TIMESTAMP,    PROXY_SESSIONID,    GLOBAL_UID,    INSTANCE_NUMBER,
OS_PROCESS)
AS
select   os_username,      username,   userhost,   terminal,   timestamp,
action_name,
       logoff_time,     logoff_lread,     logoff_pread,     logoff_lwrite,
logoff_dlock,
       sessionid,        returncode,        client_id,        session_cpu,
extended_timestamp,
       proxy_sessionid, global_uid, instance_number, os_process
```

```
from dba_audit_trail
where action between 100 and 102
/


CREATE OR REPLACE VIEW DBA_AUDIT_STATEMENT
(OS_USERNAME, USERNAME, USERHOST, TERMINAL, TIMESTAMP,
 OWNER, OBJ_NAME, ACTION_NAME, NEW_NAME, OBJ_PRIVILEGE,
 SYS_PRIVILEGE, ADMIN_OPTION, GRANTEE, AUDIT_OPTION, SES_ACTIONS,
 COMMENT_TEXT, SESSIONID, ENTRYID, STATEMENTID, RETURNCODE,
 PRIV_USED, CLIENT_ID, ECONTEXT_ID, SESSION_CPU, EXTENDED_TIMESTAMP,
 PROXY_SESSIONID,      GLOBAL_UID,      INSTANCE_NUMBER,      OS_PROCESS,
TRANSACTIONID,
 SCN, SQL_BIND, SQL_TEXT)
AS
select OS_USERNAME, USERNAME, USERHOST, TERMINAL, TIMESTAMP,
       OWNER, OBJ_NAME, ACTION_NAME, NEW_NAME,
       OBJ_PRIVILEGE,      SYS_PRIVILEGE,      ADMIN_OPTION,      GRANTEE,
AUDIT_OPTION,
       SES_ACTIONS, COMMENT_TEXT,  SESSIONID, ENTRYID, STATEMENTID,
       RETURNCODE, PRIV_USED, CLIENT_ID, ECONTEXT_ID, SESSION_CPU,
       EXTENDED_TIMESTAMP, PROXY_SESSIONID, GLOBAL_UID, INSTANCE_NUMBER,
       OS_PROCESS, TRANSACTIONID, SCN, SQL_BIND, SQL_TEXT
from dba_audit_trail
where action in (        17 /* GRANT OBJECT  */,
                         18 /* REVOKE OBJECT */,
                         30 /* AUDIT OBJECT */,
                         31 /* NOAUDIT OBJECT */,
                         49 /* ALTER SYSTEM */,
                        104 /* SYSTEM AUDIT */,
                        105 /* SYSTEM NOAUDIT */,
                        106 /* AUDIT DEFAULT */,
                        107 /* NOAUDIT DEFAULT */,
                        108 /* SYSTEM GRANT */,
                        109 /* SYSTEM REVOKE */,
                        114 /* GRANT ROLE */,
                        115 /* REVOKE ROLE */ )
/


CREATE OR REPLACE VIEW DBA_AUDIT_TRAIL
(OS_USERNAME, USERNAME, USERHOST, TERMINAL, TIMESTAMP,
 OWNER, OBJ_NAME, ACTION, ACTION_NAME, NEW_OWNER,
 NEW_NAME, OBJ_PRIVILEGE, SYS_PRIVILEGE, ADMIN_OPTION, GRANTEE,
 AUDIT_OPTION, SES_ACTIONS, LOGOFF_TIME, LOGOFF_LREAD, LOGOFF_PREAD,
 LOGOFF_LWRITE, LOGOFF_DLOCK, COMMENT_TEXT, SESSIONID, ENTRYID,
 STATEMENTID, RETURNCODE, PRIV_USED, CLIENT_ID, ECONTEXT_ID,
```

```
 SESSION_CPU,     EXTENDED_TIMESTAMP,     PROXY_SESSIONID,     GLOBAL_UID,
INSTANCE_NUMBER,
 OS_PROCESS, TRANSACTIONID, SCN, SQL_BIND, SQL_TEXT)
AS
select spare1            /* OS_USERNAME */,
       userid            /* USERNAME */,
       userhost          /* USERHOST */,
       terminal          /* TERMINAL */,
       cast (            /* TIMESTAMP */
           (from_tz(ntimestamp#,'00:00') at local) as date),
       obj$creator       /* OWNER */,
       obj$name          /* OBJECT_NAME */,
       aud.action#       /* ACTION */,
       act.name          /* ACTION_NAME */,
       new$owner         /* NEW_OWNER */,
       new$name          /* NEW_NAME */,
       decode(aud.action#,
              108 /* grant  sys_priv */, null,
              109 /* revoke sys_priv */, null,
              114 /* grant  role */, null,
              115 /* revoke role */, null,
              auth$privileges)
                         /* OBJ_PRIVILEGE */,
       decode(aud.action#,
              108 /* grant  sys_priv */, spm.name,
              109 /* revoke sys_priv */, spm.name,
              null)
                         /* SYS_PRIVILEGE */,
       decode(aud.action#,
              108 /* grant  sys_priv */, substr(auth$privileges,1,1),
              109 /* revoke sys_priv */, substr(auth$privileges,1,1),
              114 /* grant  role */, substr(auth$privileges,1,1),
              115 /* revoke role */, substr(auth$privileges,1,1),
              null)
                         /* ADMIN_OPTION */,
       auth$grantee      /* GRANTEE */,
       decode(aud.action#,
              104 /* audit   */, aom.name,
              105 /* noaudit */, aom.name,
              null)
                         /* AUDIT_OPTION  */,
       ses$actions       /* SES_ACTIONS   */,
       logoff$time       /* LOGOFF_TIME   */,
       logoff$lread      /* LOGOFF_LREAD  */,
       logoff$pread      /* LOGOFF_PREAD  */,
       logoff$lwrite     /* LOGOFF_LWRITE */,
       decode(aud.action#,
              104 /* audit   */, null,
              105 /* noaudit */, null,
              108 /* grant  sys_priv */, null,
```

```
              109 /* revoke sys_priv */, null,
              114 /* grant  role */, null,
              115 /* revoke role */, null,
              aud.logoff$dead)
                          /* LOGOFF_DLOCK */,
       comment$text       /* COMMENT_TEXT */,
       sessionid          /* SESSIONID */,
       entryid            /* ENTRYID */,
       statement          /* STATEMENTID */,
       returncode         /* RETURNCODE */,
       spx.name           /* PRIVILEGE */,
       clientid           /* CLIENT_ID */,
       auditid            /* ECONTEXT_ID */,
       sessioncpu         /* SESSION_CPU */,
       from_tz(ntimestamp#,'00:00') at local,
                                  /* EXTENDED_TIMESTAMP */
       proxy$sid                     /* PROXY_SESSIONID */,
       user$guid                          /* GLOBAL_UID */,
       instance#                     /* INSTANCE_NUMBER */,
       process#                           /* OS_PROCESS */,
       xid                             /* TRANSACTIONID */,
       scn                                      /* SCN */,
       to_nchar(substr(sqlbind,1,2000))       /* SQL_BIND */,
       to_nchar(substr(sqltext,1,2000))       /* SQL_TEXT */
from sys.aud$ aud, system_privilege_map spm, system_privilege_map spx,
     STMT_AUDIT_OPTION_MAP aom, audit_actions act
where    aud.action#     = act.action    (+)
  and - aud.logoff$dead = spm.privilege (+)
  and    aud.logoff$dead = aom.option#   (+)
  and - aud.priv$used   = spx.privilege (+)
/
```

ANEXO B: GUIA DE PROGRAMACIÓN

B.1 DOCUMENTACIÓN Y COMENTARIOS EN EL CÓDIGO

Todo bloque de código tendrá como encabezado las siguientes líneas:

/*
Creado por: Ing. Juan Carlos García
Fecha de creación: 24/09/2007
Última modificación: 24/09/2007
Descripción del bloque: Una breve descripción sobre el bloque de código siguiente.
Descripción de Variables: Una breve descripción de las variables utilizadas y su utilización en l bloque de código
*/

B.2 PALABRAS RESERVADAS DEL LENGUAJE DE PROGRAMACIÓN

Todas las palabras reservadas que forman parte del lenguaje serán escritas en MAYUSCULA.

Ejemplo:
/*
Creado por: Ing. Juan Carlos García
Fecha de creación: 24/09/2007
Última modificación: 24/09/2007
Descripción del bloque: Este bloque permite sumar los subtotales del campo total.
Descripción de Variables:
lnumPrueba Variable que almacena la suma de los totales

```
*/
PACKAGE BODY CALCULOS IS
    PROCEDURE SUBTOTAL IS
            lnumPrueba NUMERIC;
      BEGIN
             lnumPrueba:=454545;
      END;
    END;
```

ANEXO C: PROTOTIPO DE INTERFAZ DE USUARIO

C.1 INTRODUCCION

Toda obra de diseño conlleva la elección de los elementos básicos que la van a formar. Una composición gráfica está destinada a representar un medio de comunicación entre personas y ordenadores, este es el caso de una interfaz de usuario.

Es decir el diseño gráfico aplicado a la construcción de interfaces Web, para conseguir un medio de interacción entre los usuarios y el conjunto de páginas de un sitio Web y las aplicaciones que corren por debajo de ellas.

C.1.1 PROPÓSITO.

Dar a conocer a los interesados la plantilla que regirá las aplicaciones que se desean implementar en la institución, así como también los archivos de configuración, el mismo que servirá de base para las aplicaciones futuras.

C.1.2 DESCRIPCIÓN.

Este documento presenta al interesado los siguientes aspectos:

- Archivos y configuraciones necesarias para la personalización de interfaces gráficas.
- Diseño de la plantilla estándar.
- Funciones y procedimientos para la ejecución de los procesos básicos de la plantilla estándar.

Utilizando la plataforma Oracle 10g como servidor de base de datos, OC4J como servidor de aplicaciones y como IDE de programación Developer Forms con lenguaje de programación PL/SQL.

C.2 ARCHIVOS DE CONFIGURACIÓN

C.2.1 VISUALIZACIÓN DE ICONOS

El sistema también permite obtener un reporte a través del Web, de forma automática sin necesidad de que alguien lo envíe. Para visualizarlos en tiempo de ejecución haremos lo siguiente:

- Editamos el archivo **orion-web.xml** localizado en **ORA-HOME/j2ee/ DevSuite/Application-deployments/forms/formsweb** y añadimos el directorio virtual donde se va encontrar los iconos:

  <virtual-directory virtual-path="/icons" real-path="C:MyAplicacion/iconos" />

- Le indicamos ahora al servicio que extensión van a tener y en que directorio virtual se encuentran. Editamos el archivo **Registry.dat** que está en la ruta **ORA-HOME/forms/java/oracle/forms/registry** y añadimos o modificamos las siguientes líneas: default.icons.iconpath=icons/default.icons.iconextension=jpg

Si estamos trabajando con Developer Forms en tiempo de diseño, podemos observar que los botones icónicos aparecen en blanco aunque hayamos introducido la ruta correcta de donde se encuentran. La forma de implantarlos es la siguiente:

- Los nombres de los archivos icónicos no deben tener el path ni la extensión, únicamente el nombre.

- Editamos el registro de Windows y en **HKEY_LOCAL_MACHINE/Software/Oracle/HOME0** creamos la variable **UI_ICON_EXTENSION** con valor jpg ya que estamos utilizando los iconos con esta extensión. Lógicamente debemos indicar el path de los iconos en la clave **UI_ICON** (esta clave normalmente ya esta creada, si no es así debemos crearla).

Con esto tendríamos configurada la visualización de íconos.

C.2.2 PERSONALIZACIÓN DE LA PÁGINA PRINCIPAL DE LA APLICACIÓN. (CONFIGURACIÓN DEL ARCHIVO FORMSWEB.CFG).

El archivo formsweb.cfg se encuentra ubicado en el siguiente directorio:

toolsOracle\oracle\produc\10.2.0\db_2\forms90\server\

En este archivo se definen los valores de parámetro usados por el FormsServlet (f90servlet). Cualquiera de ellos se puede eliminar o modificar en las secciones de configuración nombradas.

A continuación se presenta un listado de los parámetros más importantes para la personalización de la página principal.

pageTitle

Nombre del título de la página. Ejemplo.

```
# HTML page title
pageTitle=Aplicaciones UTN
```

width

Especifica el ancho del applet del formulario, en pixeles. Por defecto es 650.Ejemplo.

Forms applet parameter

width=980

height

Especifica el alto del applet del formulario, en pixeles. Por defecto es 500.Ejemplo.

Forms applet parameter

height=590

separateFrame

Se determina si el applet aparece dentro de una ventana separada. Valores legales: Verdad o falso. Ejemplo.

Forms applet parameter

separateFrame=false

splashScreen

Especifica el archivo .GIF que debe aparecer antes de que aparezca el applet. Fijar a NO para no aparecer. Dejar vacío para utilizar la imagen por defecto.

Para fijar el parámetro incluir el nombre del archivo (por ejemplo, myfile.gif) o la trayectoria virtual y nombre del archivo (por ejemplo, imágenes/myfile.gif). Ejemplo.

Forms applet parameter

splashScreen=utn2.gif

background

Especifica el archivo .GIF que debe aparecer en el fondo. Fijar a NO para ningún fondo. Dejar vacío para utilizar el fondo por defecto.

Forms applet parameter

background=utn1.gif

lookAndFeel

Para modificar la apariencia de la aplicación, los valores que puede tomar son:

- generic: Apariencia típica de Windows
- oracle: Apariencia por defecto definida por Oracle.

Ejemplo:

Forms applet parameter

lookAndFeel=oracle

colorScheme

El valor del parámetro lookAndFeel es oracle en colorScheme se puede definir el siguiente conjunto de colores:

- teal
- red
- titanium
- blue
- khaki

- olive
- purple

Ejemplo:

```
# Forms applet parameter

colorScheme=blue
```

Logo

Especifica el archivo .GIF que debe aparecer en la barra de menú de las formas. Fijar a NO para ninguna insignia. Dejar vacío para utilizar la insignia de Oracle por defecto. Ejemplo.

```
# Forms applet parameter

logo=utn.gif
```

C.3 DISEÑO DE LA PLANTILLA ESTÁNDAR.

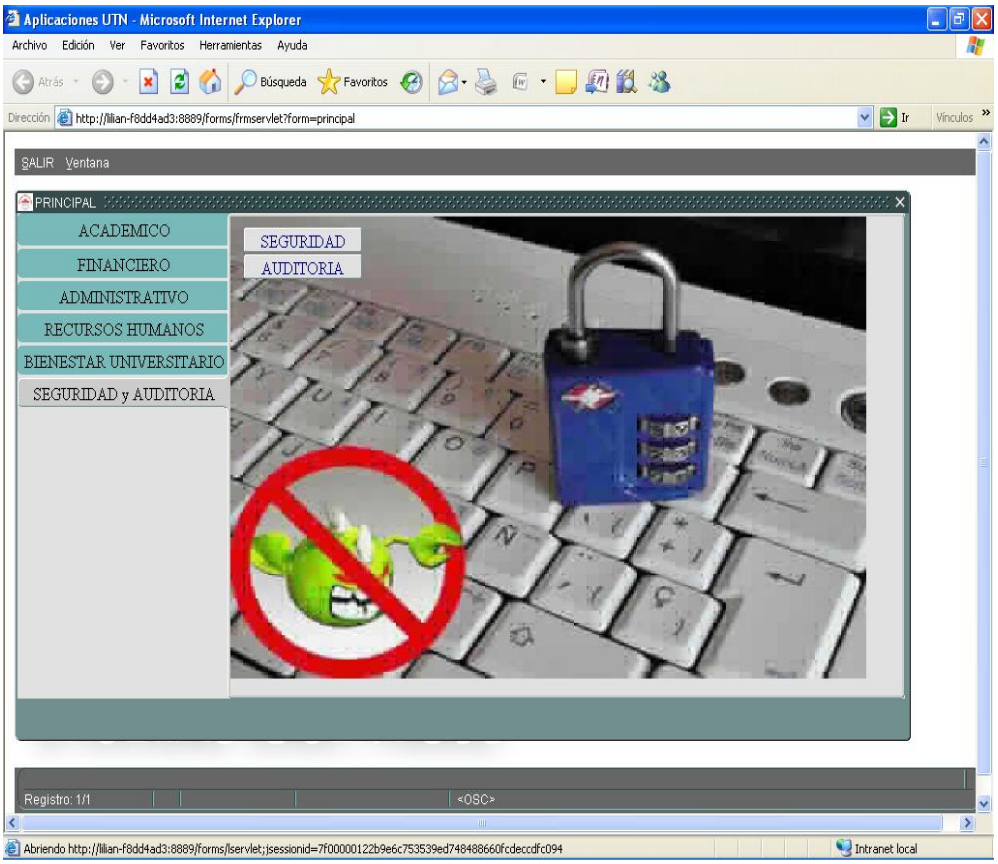


Ilustración 0-9 Menú de seguridad y Auditoría

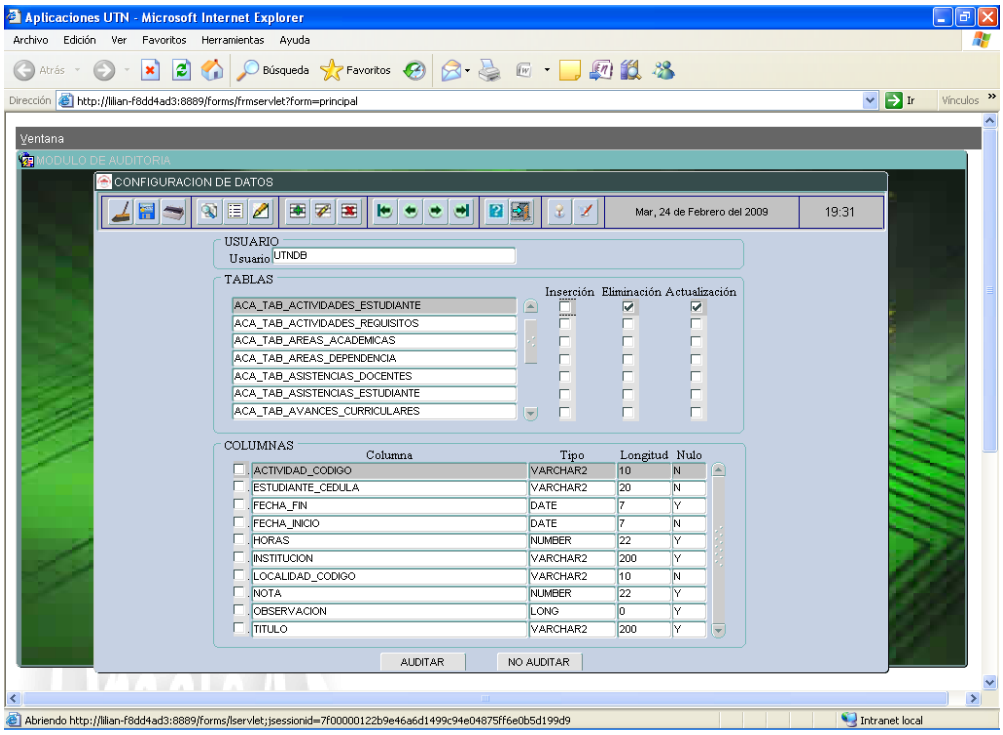


Ilustración 0-10 Plantilla de formulario

C.4 FUNCIONES Y PROCEDIMIENTOS PARA LA EJECUCIÓN DE LOS PROCESOS BÁSICOS DE LA PLANTILLA ESTÁNDAR.

FUNCTION FUN_ALERTA_2BOTONES

```
/*
Esta función permite establecer una alerta con 2 botones personalizada, y retorna 1, 2 o 0.
Descripción de Variables:
 pvarc2NombreAlerta  Este parámetro recibe el nombre para la alerta, cuyos valores pueden ser.
                     ALE_ATENCION
                     ALE_INFORMACION
                     ALE_ERROR
 pvarc2MensajeAlerta  Este parámetro recibe el mensaje para la alerta
 pvarc2Boton1Alerta    Este parámetro recibe el nombre del boton1
 pvarc2Boton2Alerta    Este parámetro recibe el nombre del boton2
 pvarc2TituloAlerta     Este parámetro recibe el titulo de la alerta
 lnumbBanderaBoton   Esta variable obtiene el valor que retorna la alerta
*/
FUNCTION FUN_ALERTA_2BOTONES
(
  pvarc2NombreAlerta  VARCHAR2,
  pvarc2TituloAlerta  VARCHAR2,
```

```
  pvarc2MensajeAlerta VARCHAR2,
  pvarc2Boton1Alerta  VARCHAR2,
  pvarc2Boton2Alerta  VARCHAR2
)
RETURN NUMBER
IS
  lnumbBanderaBoton NUMBER;
BEGIN
SET_ALERT_PROPERTY (pvarc2NombreAlerta, ALERT_MESSAGE_TEXT, pvarc2MensajeAlerta);
 SET_ALERT_PROPERTY (pvarc2NombreAlerta, TITLE, pvarc2TituloAlerta);
       SET_ALERT_BUTTON_PROPERTY (pvarc2NombreAlerta, ALERT_BUTTON1, LABEL,
       pvarc2Boton1Alerta);
 SET_ALERT_BUTTON_PROPERTY (pvarc2NombreAlerta, ALERT_BUTTON2, LABEL, pvarc2Boton2Alerta);
 lnumbBanderaBoton := SHOW_ALERT (pvarc2NombreAlerta);
 IF lnumbBanderaBoton = ALERT_BUTTON1 THEN
    RETURN 1;
 ELSIF lnumbBanderaBoton = ALERT_BUTTON2 THEN
    RETURN 2;
 ELSE
    RETURN 0;
 END IF;
END;
*********************************************************************************
```

FUNCTION FUN_OBTENER_FECHA_LARGA

/*Esta función permite obtener la fecha actual en el siguiente formato (01 DE ENERO DEL 2007) recibiendo como parámetro la fecha actual del sistema.

Descripción de Variables:

lvarc2FechaLarga Variable en la que se va concatenando la fecha larga.

lvarc2Mes Variable que almacena el número de mes.

lvarc2Anio Variable que almacena el año.

*/

```
FUNCTION FUN_OBTENER_FECHA_LARGA
(
  pdatFechaCorta DATE
)
RETURN VARCHAR2
IS
    lvarc2FechaLarga VARCHAR2 (100)           ;
  lvarc2Mes VARCHAR2(2);
  lvarc2Año VARCHAR2(4);
BEGIN
  lvarc2FechaLarga := TO_CHAR (pdatFechaCorta, 'Dy') || ', ';
  lvarc2FechaLarga := lvarc2FechaLarga || TO_CHAR (pdatFechaCorta,
                    'DD') || ' DE ';
  lvarc2Mes := TO_CHAR (pdatFechaCorta, 'MM');
  IF lvarc2Mes = '01' THEN
```

```
  lvarc2FechaLarga := lvarc2FechaLarga || 'ENERO ';
ELSIF lvarc2Mes = '02' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'FEBRERO ';
ELSIF lvarc2Mes = '03' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'MARZO ';
ELSIF lvarc2Mes = '04' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'ABRIL ';
ELSIF lvarc2Mes = '05' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'MAYO ';
ELSIF lvarc2Mes = '06' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'JUNIO ';
ELSIF lvarc2Mes = '07' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'JULIO ';
ELSIF lvarc2Mes = '08' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'AGOSTO ';
ELSIF lvarc2Mes = '09' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'SEPTIEMBRE ';
ELSIF lvarc2Mes = '10' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'OCTUBRE ';
ELSIF lvarc2Mes = '11' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'NOVIEMBRE ';
ELSIF lvarc2Mes = '12' THEN
  lvarc2FechaLarga := lvarc2FechaLarga || 'DICIEMBRE ';
        END IF;
```

```
  lvarc2Año := TO_CHAR (pdatFechaCorta, 'YYYY');
  IF substr (lvarc2Año, 1, 1) = '2' THEN
    lvarc2FechaLarga := lvarc2FechaLarga || 'DEL ' || lvarc2Año;
  ELSE
    lvarc2FechaLarga := lvarc2FechaLarga || 'DE ' || lvarc2Año;
  END IF;
  RETURN lvarc2FechaLarga;
END;
```

****

FUNCTION FUN_OBTENER_HORA_ACTUAL

/*

Esta función permite obtener la hora actual en el siguiente formato 21:30.

Descripción de Variables:

lvarc2Hora Variable que almacenan las horas, en este caso en formato de 24 horas.

lvarc2Minuto Variable que almacenan los minutos.

lvarc2HoraActual Variable que almacena la hora tal como se va a mostrar.

*/

```
FUNCTION FUN_OBTENER_HORA_ACTUAL
RETURN VARCHAR2
IS
  lvarc2Hora VARCHAR2 (4);
  lvarc2Minuto VARCHAR2 (2);
```

```
  lvarc2HoraActual VARCHAR2(10) ;
BEGIN
  lvarc2Hora := TO_CHAR (SYSDATE, 'HH24');
  lvarc2HoraActual := lvarc2HoraActual || '  ' || lvarc2Hora;
  lvarc2Minuto := TO_CHAR (SYSDATE, 'MI');
  lvarc2HoraActual := lvarc2HoraActual || ':' || lvarc2Minuto;
  RETURN lvarc2HoraActual;
END;
```

****

PROCEDURE PRO_ACCIONES_TOOLBAR

/*

Este procedimiento permite determinar que botón ha sido seleccionado de la barra y le asigna una acción.

Descripción de Variables:

lvarc2NombreElemento Esta variable sirve para recuperar el nombre del elemento seleccionado en la barra.

lvarc2NombreBloqueElemento Esta variable sirve para recuperar el nombre del bloque y el elemento seleccionado en la barra.

lnumbBanderaAlerta number Esta variable sirve para obtener el valor retornado de la alerta.

*/

```
PROCEDURE PRO_ACCIONES_TOOLBAR
IS
lvarc2NombreElemento VARCHAR2(30);
```

```
lvarc2NombreBloqueElemento VARCHAR2(60);
lnumbBanderaAlerta NUMBER;
 BEGIN
lvarc2NombreBloqueElemento := NAME_IN('SYSTEM.TRIGGER_ITEM');
lvarc2NombreElemento := SUBSTR(lvarc2NombreBloqueElemento, INSTR(
                    lvarc2NombreBloqueElemento, '.') + 1);
IF(lvarc2NombreElemento = 'CMD_GUARDAR') THEN
lnumbBanderaAlerta := FUN_ALERTA_2BOTONES('ALE_INFORMACION'
                    ,'Atención UTN','Desea Guardar Los Cambios','Sí','No');
IF(lnumbBanderaAlerta = 1) THEN
   DO_KEY('COMMIT_FORM');
END IF;
 ELSIF(lvarc2NombreElemento = 'CMD_IMPRIMIR')THEN
      DO_KEY('PRINT');
 ELSIF (lvarc2NombreElemento = 'CMD_LIMPIAR_FORMA') THEN
DO_KEY('CLEAR_FORM');
              :BLOQ_TOOLBAR.TXT_MOSTRAR_FECHA                                    :=
               FUN_OBTENER_FECHA_LARGA(SYSDATE);
       :BLOQ_TOOLBAR.TXT_MOSTRAR_HORA := FUN_OBTENER_HORA_ACTUAL();

 :BLOQ_TOOLBAR.TXT_MOSTRAR_USUARIO:=get_application_property(USERNAME);
  ELSIF (lvarc2NombreElemento = 'CMD_BUSCAR')   THEN
    IF (name_in('SYSTEM.MODE') != 'ENTER-QUERY') THEN
          DO_KEY('ENTER_QUERY');
    ELSE
```

```
        DO_KEY('EXECUTE_QUERY');
   END IF;
  elsif (lvarc2NombreElemento = 'CMD_INSERTAR_REGISTRO') THEN
   CREATE_RECORD;
  elsif (lvarc2NombreElemento = 'CMD_BORRAR_REGISTRO') THEN
   lnumbBanderaAlerta := FUN_ALERTA_2BOTONES('ALE_ATENCION','Atención
UTN','Desea Eliminar El Cliente','Aceptar','Cancelar');
   IF(lnumbBanderaAlerta = 1) THEN
        DELETE_RECORD;
   END IF;
  elsif (lvarc2NombreElemento = 'CMD_LIMPIAR_REGISTRO') THEN
        CLEAR_RECORD;
  elsif (lvarc2NombreElemento = 'CMD_PRIMER_REGISTRO') THEN
        FIRST_RECORD;
  elsif (lvarc2NombreElemento = 'CMD_SIGUIENTE_REGISTRO') THEN
        NEXT_RECORD;
  elsif (lvarc2NombreElemento = 'CMD_ANTERIOR_REGISTRO') THEN
        PREVIOUS_RECORD;
  elsif (lvarc2NombreElemento = 'CMD_ULTIMO_REGISTRO') THEN
        LAST_RECORD;
  elsif (lvarc2NombreElemento = 'CMD_LISTAR') THEN
        DO_KEY('LIST_VALUES');
  elsif (lvarc2NombreElemento = 'CMD_EDITAR')       THEN
        DO_KEY('EDIT_FIELD');
  elsif (lvarc2NombreElemento = 'CMD_AYUDA') THEN
```

```
        show_keys;
  elsif (lvarc2NombreElemento = 'CMD_SALIR') THEN
    lnumbBanderaAlerta := FUN_ALERTA_2BOTONES('ALE_ATENCION','Atención
                UTN','Desea Salir De La Aplicación','Sí','No');
    IF(lnumbBanderaAlerta = 1) THEN
      DO_KEY ('exit_form');
    END IF;
  END IF;
END;
```

****

PROCEDURE PRO_INFORMACION_OBJETO

/*

Este procedimiento permite obtener información de un objeto al pasar el mouse sobre él.

Descripción de Variables:

varc2 Objeto Parámetro que almacena nombre de un objeto

varc2 Informacion Parámetro que almacena la información que va a aparecer al pasar el mouse.

*/

```
PROCEDURE PRO_INFORMACION_OBJETO (pvarc2Objeto VARCHAR2, pvarc2Informacion
VARCHAR2)IS
BEGIN
        SET_ITEM_PROPERTY(pvarc2Objeto, TOOLTIP_TEXT,
                pvarc2Informacion);
```

```
        SET_ITEM_PROPERTY(pvarc2Objeto, TOOLTIP_FONT_SIZE, 800);
        SET_ITEM_PROPERTY(pvarc2Objeto, TOOLTIP_FOREGROUND_COLOR,
                'r0g50b0');
    SET_ITEM_PROPERTY(pvarc2Objeto, BACKGROUND_COLOR,
                'r180g220b180');
END;
**************************************************************************
```

PROCEDURE PRO_INFORMACION_TOOLBAR

```
/*
```

Este procedimiento permite obtener información de cada uno de los objeto de la barra de herramientas al pasar el mouse.

```
*/
PROCEDURE  PRO_INFORMACION_TOOLBAR IS
BEGIN
 :BLOQ_TOOLBAR.TXT_MOSTRAR_FECHA :=
FUN_OBTENER_FECHA_LARGA(SYSDATE);
 :BLOQ_TOOLBAR.TXT_MOSTRAR_HORA:=FUN_OBTENER_HORA_ACTUAL();
 :BLOQ_TOOLBAR.TXT_MOSTRAR_USUARIO:=get_application_property(USERNAME);

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_SALIR','Salir');
        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_AYUDA','Ayuda');
        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_BUSCAR','Buscar');
        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_IMPRIMIR','Imprimir');
```

```
        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_LIMPIAR_FORMA','Limpi
ar

                Forma');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_LISTAR','Lista');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_EDITAR','Editar');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_GUARDAR','Guardar');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_INSERTAR_REGISTRO','I
nsertar

                Registro');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_LIMPIAR_REGISTRO','Li
mpiar

               Registro');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_BORRAR_REGISTRO','Bo
rrar

                Registro');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_SIGUIENTE_REGISTRO','
Registro

                Siguiente');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_ANTERIOR_REGISTRO','
Registro

               Anteior');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_PRIMER_REGISTRO','
Primer

               Registro');

        PRO_INFORMACION_OBJETO('BLOQ_TOOLBAR.CMD_ULTIMO_REGISTRO','Ulti
mo

               Registro');
```

END;

****

PROCEDURE PRO_ALERTA

/*

Este procedimiento permite establecer una alerta personalizada.

Descripción de Variables:

pvarc2NombreAlerta Este parámetro recibe el nombre para la alerta, cuyos valores pueden ser.

ALE_ATENCION

ALE_INFORMACION

ALE_ERROR

pvarc2MensajeAlerta Este parámetro recibe el mensaje para la alerta

pvarc2TituloAlerta Este parámetro recibe el titulo de la alerta

lnumbBanderaBoton Esta variable obtiene el valor que retorna la alerta

*/

```
PROCEDURE PRO_ALERTA
(
  pvarc2NombreAlerta  VARCHAR2,
  pvarc2TituloAlerta  VARCHAR2,
  pvarc2MensajeAlerta VARCHAR2
)
```

```
IS
  lnumbBanderaBoton NUMBER;
BEGIN
  SET_ALERT_PROPERTY (pvarc2NombreAlerta, ALERT_MESSAGE_TEXT,
                pvarc2MensajeAlerta);
  SET_ALERT_PROPERTY (pvarc2NombreAlerta, TITLE, pvarc2TituloAlerta);
  lnumbBanderaBoton := SHOW_ALERT (pvarc2NombreAlerta);
END;
```

****

PROCEDURE PRO_TITULO_COLOR_VENTANA

```
/*
Este procedimiento permite poner titulo a la ventana, además se define el color y se maximiza.
Descripción de Variables:
  pvarc2NombreVentana Parámetro que recibe el nombre de la Ventana
  pvarc2TituloVentana Parámetro que recibe el título de la Ventana
*/
PROCEDURE PRO_TITULO_COLOR_VENTANA
(
  pvarc2NombreVentana VARCHAR2,
  pvarc2TituloVentana VARCHAR2
)
IS
BEGIN
```

```
  PRO_VENTANA_CENTRADA('WINDOW1');
  --SET_WINDOW_PROPERTY('WINDOW1', WINDOW_STATE, MAXIMIZE);
  SET_WINDOW_PROPERTY (pvarc2NombreVentana, TITLE,
          pvarc2TituloVentana || '   Form:(' ||
          get_application_property(CURRENT_FORM_NAME) || ')');
  SET_WINDOW_PROPERTY (pvarc2NombreVentana,
              BACKGROUND_COLOR,'r200g230b210');
END;
```

****

PROCEDURE PRO_VENTANA_CENTRADA

/*

Este procedimiento permite centrar la ventana

Descripción de Variables:

pvarc2win	Parámetro que recibe el nombre de la ventana
lwinWinId	Variable que almacena el nombre de la ventana
lnumbWinX	Variable para la posicion en x de la ventana
lnumbWinY	Variable para la posicion en y de la ventana
lnumbWinW	Variable para el ancho de la ventana
lnumbWinH	Variable para el largo de la ventana
lnumbDisplayW	Variable para el ancho de la pantalla
lnumbDisplayH	Variable para el largo de la pantalla
lnumbHeightOffsett	Variable para el largo de la ventana

```
*/

PROCEDURE PRO_VENTANA_CENTRADA ( pvarc2Win VARCHAR2 )

IS

  lwinWinId  window;

  lnumbWinX   NUMBER;

  lnumbWinY   NUMBER;

  lnumbWinW   NUMBER;

  lnumbWinH   NUMBER;

  lnumbDisplayW NUMBER;

  lnumbDisplayH NUMBER;

  lnumbHeightOffset NUMBER := 0;

BEGIN

  IF Get_Application_Property(USER_INTERFACE)='MSWINDOWS' THEN

    lnumbHeightOffset := .05; -- inches;

  END IF;

  lwinWinId := FIND_WINDOW(pvarc2Win);

  IF ID_NULL(lwinWinId) THEN

    RETURN;

  END IF;

  lnumbDisplayH :=

              TO_NUMBER(GET_APPLICATION_PROPERTY(DISPLAY_HEIGHT));

  lnumbDisplayW := TO_NUMBER(GET_APPLICATION_PROPERTY(DISPLAY_WIDTH));


  lnumbWinX := GET_WINDOW_PROPERTY(lwinWinId, X_POS);
```

```
  lnumbWinY := GET_WINDOW_PROPERTY(lwinWinId, Y_POS);
  lnumbWinW := GET_WINDOW_PROPERTY(lwinWinId, WIDTH);
  lnumbWinH := GET_WINDOW_PROPERTY(lwinWinId, HEIGHT);
  lnumbWinH := lnumbWinH+100;
  IF ( lnumbWinW >= lnumbDisplayW ) THEN
    lnumbWinX := 0;
  ELSE
    lnumbWinX := (lnumbDisplayW - lnumbWinW) / 2;
  END IF;
  IF ( lnumbWinH >= lnumbDisplayH ) THEN
    lnumbWinY := 0;
  ELSE
    lnumbWinY := (lnumbDisplayH - lnumbHeightOffset - lnumbWinH) / 2;
  END IF;
  -- Set window's new position
  SET_WINDOW_PROPERTY(lwinWinId, X_POS, lnumbWinX-20);
  SET_WINDOW_PROPERTY(lwinWinId, Y_POS, lnumbWinY-55);
  SHOW_WINDOW(lwinWinId);
END;
```

ANEXO D: MANUAL DE INSTALACIÓN

D.1 INSTALACIÓN DE HERRAMIENTAS SOBRE LINUX

Requisitos mínimos de hardware

- Procesador de más de 480 Mhz de velocidad.
- Por lo menos 512 Mb de RAM para Linux y 1Gb para Windows.
- El suficiente espacio de disco duro (por lo menos 3Gb).

D.2 ORACLE 10G DATABASE SERVER

Se recomienda instalar el motor de base de datos en S.u.S.E. Linux Enterprise Edition versión 9 o 10, incluso soporta la versión 11g sin problemas, pero es necesario incrementar el valor de RAM a por lo menos 1 Gb. Tampoco hubo problemas al instalar el motor de base de datos en Red Hat Enterprise Linux versión 3. Al instalar linux, es necesario configurar el tamaño de la partición swap a por lo menos 1 Gb.

D.2.1 PRERREQUISITOS

Deben estar instalados los siguientes paquetes en el sistema: glibc, glibc-devel, libstdc++, libstdc++-devel, gcc, gcc-c++, openmotif-libs, openmotif21-libs, pdksh, make, sysstat. Se debe revisar para cada versión de linux cuales son las versiones correctas de los paquetes, se detalla completamente en la Guía de Instalación de Oracle 10g Database Server. Para S.u.S.E. Linux se incluye además un paquete de compatibilidad que configura las variables de entorno y scripts en el sistema que es el orarun.

Se debe crear un usuario para poder realizar la instalación, si se quiere instalar en un directorio propio del sistema como /opt o /usr, se le debería dar los permisos necesarios al

directorio sobre el cual se instale el producto, aunque no existe ningún problema al instalar en un directorio personalizado que sea propietario el usuario.

Hay que realizar algunos cambios en los parámetros del sistema operativo, se los puede realizar manualmente con el comando sysctl (para obtener información de este comando se puede utilizar el comando "man sysctl" desde la línea de comandos). Los parámetros a cambiar son:

net.ipv4.ip_local_port_range=1024 65000

kernel.sem=250 32000 100 128

kernel.shmmax=2147483648

fs.file-max=65536

Estos parámetros los incluimos en el archivo /etc/sysctl.conf.

Para S.u.S.E. Linux se debe incluir en el arranque el boot.sysctl de la siguiente manera en la línea de comandos como usuario root:

chkconfig boot.sysctl //Nos debe dar como resultado boot.sysctl off

chkconfig boot.sysctl on

chkconfig boot.sysctl //Nos debe dar como resultado boot.sysctl on

Una vez configurado el boot.sysctl y el archivo /etc/sysctl.conf ejecutamos el comando como root:

sysctl -p

Y tenemos una salida de los nuevos parámetros configurados.

D.2.2 INSTALACIÓN

Desde el CD de instalación o el directorio en el que se desempaqueto los instaladores ejecutamos el script runInstaller, se ejecuta el Oracle Universal Installer, seguimos las instrucciones de acuerdo a las necesidades.

D.2.3 DESPUÉS DE INSTALAR

Se deben configurar algunas variables de entorno (en S.u.S.E. Linux en el .profile y en Red Hat el .bash_profile del usuario del sistema que se definió como administrador de oracle, no el usuario root):

ORACLE_HOME=<Directorio de Instalación de Oracle>

ORACLE_SID=<Valor de la Instancia Configurada de Oracle>

ORACLE_OWNER=<EL usuario del sistema que se definió como Administrador de Oracle>

Son las variables de entorno principales, también se pueden configurar las siguientes:

NLS_LANG=<Idioma de Oracle, verificar en el Manual de Administración de Oracle>

CLASSPATH=<Directorio de clases de Java>

LD_LIBRARY_PATH=<Directorio de librerías binarias de Oracle>

PATH=<Agregar el Path de los binarios de Oracle que es $ORACLE_HOME/bin>

D.2.4 INICIAR Y PARAR ORACLE 10G

Para iniciar se debe montar las bases de datos y luego subir el listener, complementariamente también el Enterprise Manager de la Base de Datos. Todo esto se lo hace como usuario administrador.

Subir la Base de Datos: desde línea de comandos como usuario administrador ejecutamos

sqlplus /nolog

SQL>connect / as sysdba

SQL>startup

De igual manera para terminar ejecutamos

sqlplus /nolog

SQL> connect / as sysdba

SQL> shutdown [modo de parada abort | immediate | normal | transactional]

Cuando la base no está en producción son preferibles los modos abort o immediate.

El listener inicia y para con el comando lsnrctl:

LSNRCTL>start | stop

Y el Enterprise Manager inicia o para con el comando

emctl start | stop dbconsole

D.3 ORACLE 10G DEVELOPER SUITE

La instalación no tiene ninguna complicación, se usa los mismos requerimientos que para instalar la base de datos, pero el correcto funcionamiento se da sobre Red Hat Enterprise Edition, igualmente creamos un usuario del sistema para la instalación, pero además agregamos los siguientes paquetes: compat-glibc, compat-libstdc++, compat-libstdc++-devel, compat-db, binutils, gnome-libs, setarch. Igualmente las versiones

correctas de estos paquetes se encuentran en la Guía de Instalación de Oracle Developer Suite 10g.

Desde el CD de instalación o desde los directorios donde se desempaqueto el instalador corremos el script runInstaller y seguimos las instrucciones de instalación. Luego tenemos que setear la variable de entorno ORACLE_HOME al directorio de instalación.

Para ejecutar los programas, lo hacemos desde el directorio bin donde instalamos, el forms builder es el frmbld.sh y para el reports builder el rwbuilder.sh.

Para correr las formas debemos configurar el mozilla navigator, en el directorio de instalación de mozilla (para Red Hat Enterprise Linux 3 es /usr/lib/mozilla-1.7.10) vamos al directorio de plugins y como usuario root creamos un enlace simbólico a la librería de plugins de java de jdk1.4.2_6 para ns610-gcc32, de esta manera (la librería es libjavaplugin_oji.so):

ln -s $ORACLE_HOME/jdk/jre/plugin/ns610-gcc32/libjavaplugin_oji.so

/usr/lib/mozilla-1.7.10/plugins/libjavaplugin_oji.so

El mismo procedimiento se utiliza para el mozilla firefox, suele estar instalado en /usr/lib/firefox, por lo que el enlace sería:

ln -s $ORACLE_HOME/jdk/jre/plugin/ns610-gcc32/libjavaplugin_oji.so

/usr/lib/firefox/plugins/libjavaplugin_oji.so

Se puede utilizar mozilla, mozilla firefox o netscape navigator, por lo que se realiza el enlace simbólico al directorio plugins donde se encuentre instalado el navegador.

Para iniciar la instancia del contenedor java de aplicaciones para correr las formas ejecutamos el script:

$ORACLE_HOME/j2ee/DevSuite/startinst.sh

Igualmente para detener la instancia usamos el script stopinst.sh. Ya se puede correr las forms en modo de desarrollo.

D.4 ORACLE 10G APPLICATION SERVER

De igual manera que en la instalación de los otros paquetes, ejecutamos el script runInstaller desde el disco 1, pero hay que tener otras consideraciones antes de lanzar el script de instalación. Se utilizan los mismos requisitos que para instalar la Oracle 10G Database Server, pero además debemos instalar los siguientes paquetes: db1, compat-glibc, compat-libstdc++, compat-libstdc++-devel, compat-db. Se deben desconfigurar las siguientes ENV, ORACLE_HOME, ORACLE_BASE, ORACLE_SID, de la siguiente manera:

unset ENV

unset ORACLE_HOME

unset ORACLE_BASE

unset ORACLE_SID

D.4.1 INSTALACIÓN

Una vez hechos los cambios para los prerrequisitos ya por fin se puede lanzar el instalador. Primero se debe instalar la infraestructura, que consiste en Oracle LDAP (Oracle Internet Directory), para la autenticación del acceso a las aplicaciones, además

también se debe escoger la opción Single Sign-on, que sirve para acceso a la web de las aplicaciones Oracle, todos estas opciones son parte de Oracle Identity Manager, y deben funcionar sobre una instancia especial de Oracle 10G Database (Metadatos), preparada especialmente para soportar el Oracle Internet Directory, se pueden instalar todo en un solo conjunto o utilizar una instancia ya instalada y prepararla manualmente (lo cual complica las cosas), por lo que es preferible que el instalador realice estas acciones.

Una vez que tenemos instalada la infraestructura, podemos instalar el Oracle Application Server, los contenedores OC4J (Oracle Application Server Containers for (4) J2EE) para Forms y Reports. Si queremos instalar tanto infraestructura como los contenedores en un mismo equipo tenemos que instalar cada cosa en una instancia diferente y en diferente usuario, también es necesario asignar una instancia diferente del Enterprise Manager para cada instalación, ej.: ias1, ias2.

Pueden encontrarse varios problemas al instalar tanto la infraestructura como los contenedores:

- Suele mostrarse un mensaje de que no se puede iniciar el gestor OPMN (luego se explicará en detalle para que sirve), esto se presenta cuando se están copiando los archivos en el disco duro, se debe poner continuar.

- Cuando el instalador se detiene en las configuraciones, es preferible no detener la instalación, porque se debería reiniciar todo el proceso nuevamente, desinstalando lo último y volviendo a reinstalar, sino más bien observar los archivos de logs o los mensajes que se muestran en el mismo instalador, corregir el problema y reintentar la configuración, el instalador me da la posibilidad. Los problemas frecuentes

suelen ser por incompatibilidad en las librerías o el haberse olvidado instalar algún paquete.

- También en el configurador se suele detener en el inicio de OPMN, que es el Oracle Process Manager and Notification Server, que sirve para iniciar todos los servidores del Application Server. Para corregir este problema hay que parar el OPMN y reintentar la configuración. El OPMN se ejecuta en:

  $ORACLE_HOME/opmn/bin/opmnctl <startall|stopall>

D.4.2 DESPUÉS DE INSTALAR

Después de instalar la infraestructura se deben configurar las variables de entorno de la misma manera que en Oracle 10G Database Server, en cambio en el usuario que se instalan los contenedores es suficiente con configurar la variable de entorno ORACLE_HOME.

D.4.3 INICIAR Y PARAR EL APPLICATION SERVER

Primero se requiere iniciar la infraestructura, obviamente el primer paso a seguir es subir la DB, es de la misma forma como ya se detallo anteriormente. Luego es de subir el Oracle Internet Directory

El monitor:

oidmon connect=<nombre de instancia de base de datos> <stop|start>

La instancia del Internet Directory

oidctl connect=cc server=ss instance=nn <start|stop>

donde cc=nombre de la instancia de base de datos, ss=puede ser oidldapd/oidrepld/odisrv pero en nuestro caso necesitamos iniciar el ldap y la opción sería oidldapd, nn=número de la instancia que debe ser único y es un entero. Aunque al subir la

base de datos y el monitor automáticamente se sube el Internet Directory, pero también se deben subir el resto de servicios instalados con el OPMN en:

$ORACLE_HOME/opmn/bin/opmnctl <startall|stopall>

Por último subir el Enterprise Manager:

emctl <start|stop> iasconsole

Luego toca levantar los contenedores, desde el usuario que se instaló toca subir el OPMN y el Enterprise Manager, de la misma forma descrita anteriormente.

D.4.4 NOTAS DE ÚLTIMO MOMENTO.

Todos los sistemas anteriormente descritos también fueron probados en Red Hat Enterprise Server 5.0 y 5.1, lográndose instalar pero con ciertas modificaciones.

Lo primero es modificar el archivo /etc/redhat-release, cambiar el número 5 por 4, ya que los instaladores soportan hasta Red Hat Enterprise Server 4.

Aparte de las librerías necesarias, hay que instalar las librerías xorg-x11-deprecated-libs-6.8.2-1.EL.19.i386.rpm (descargar de Internet) y forzar la librería openmotif21-2.1.30-9.RHEL3.6.i386.rpm de Red Hat Enterprise Server 3. Con estas modificaciones se puede instalar sin complicaciones.

También se probó sobre Red Hat Enterprise Server 3 y 4, resultando error en la configuración del Internet Directory del Application Server.

En Windows 2003 Server también no configuro el Internet Directory del Application Server, pero con la versión 10g 9.0.4 no hubo inconveniente. Se necesita estrictamente las precondiciones para poder instalar, caso contrario el instalador no sigue.

La opción de Discoverer del Application Server no instaló sobre ninguna versión de Linux. Hay conflictos con librerías de compatibilidad de Linux. Las versiones utilizadas son: Database Server 10g 10.2.0.1, Application Server 10g 10.1.2.02, Developer Suite 10g 10.1.2.0.2.

ANEXO E: MANUAL DE USUARIO

El presente manual ha sido elaborado para permitir el uso apropiado del Sistema de Auditoría de Bases de Datos Basado en Oracle (SABDO). Se presentan las pantallas correspondientes a cada menú, acompañado de la respectiva aclaración de sus funciones.

E.1 INGRESO AL SISTEMA

Para poder trabajar es necesario abrir un browser o navegador de internet como son: Internet Explorer, Firefox, Netscape, etc. Una vez abierto debemos ubicarnos en la barra de direcciones e ingresar la siguiente dirección URL :

http://172.20.1.159:7777/forms/frmservlet?config=utn

A continuación se visualizará la pantalla de conexión, ver fig. 11

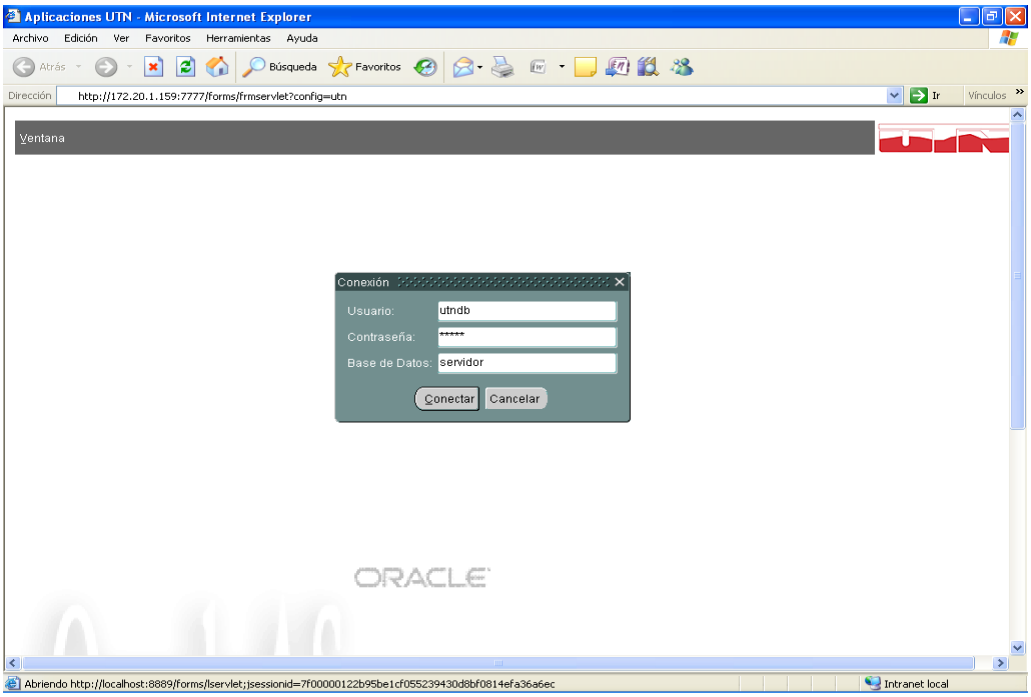


Ilustración 0-11 Ingreso al sistema

La ventana solicitará el nombre del usuario, que en nuestro caso es : **utndb**; la base de datos : **servidor** se hace click en el botón ***conectar*** para ingresar a la pantalla principal del sistema integrado de la Universidad Técnica del Norte, se muestra en la fig.12

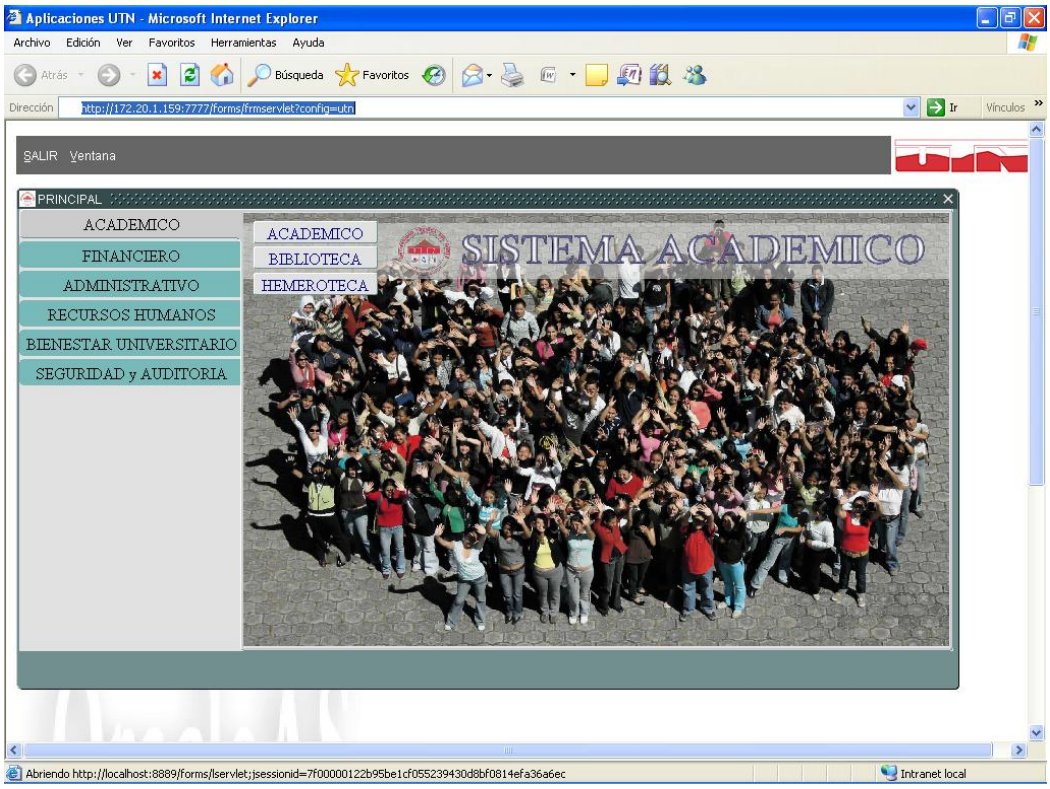


Ilustración 0-12 Pantalla principal

Debido a que el sistema SABDO ya ha sido integrado al sistema de la UTN, para utilizarlo hacemos click en la suite de ***SEGURIDAD y AUDITORIA*** . Ver fig. 13.

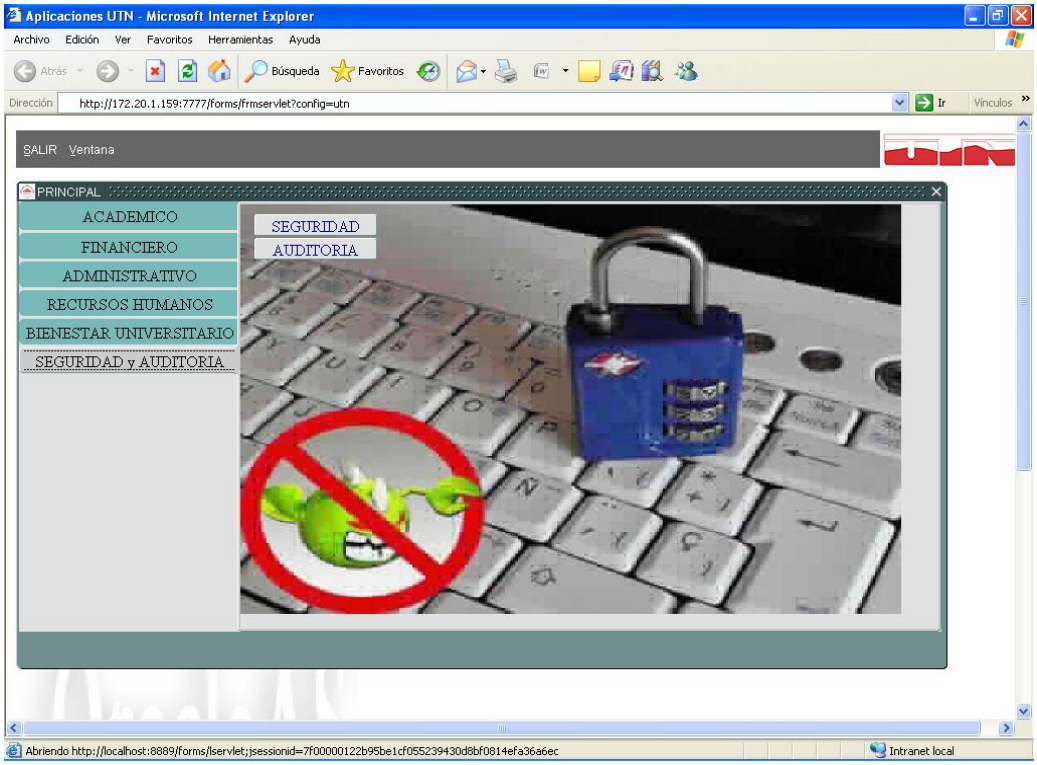


Ilustración 0-13 Menú de Seguridad y Auditoría

Como podemos apreciar aparecen dos opciones: Seguridad y Auditoría. Al hacer click en Auditoría hemos ingresado al menú principal del sistema SABDO. Ver fig. 14.

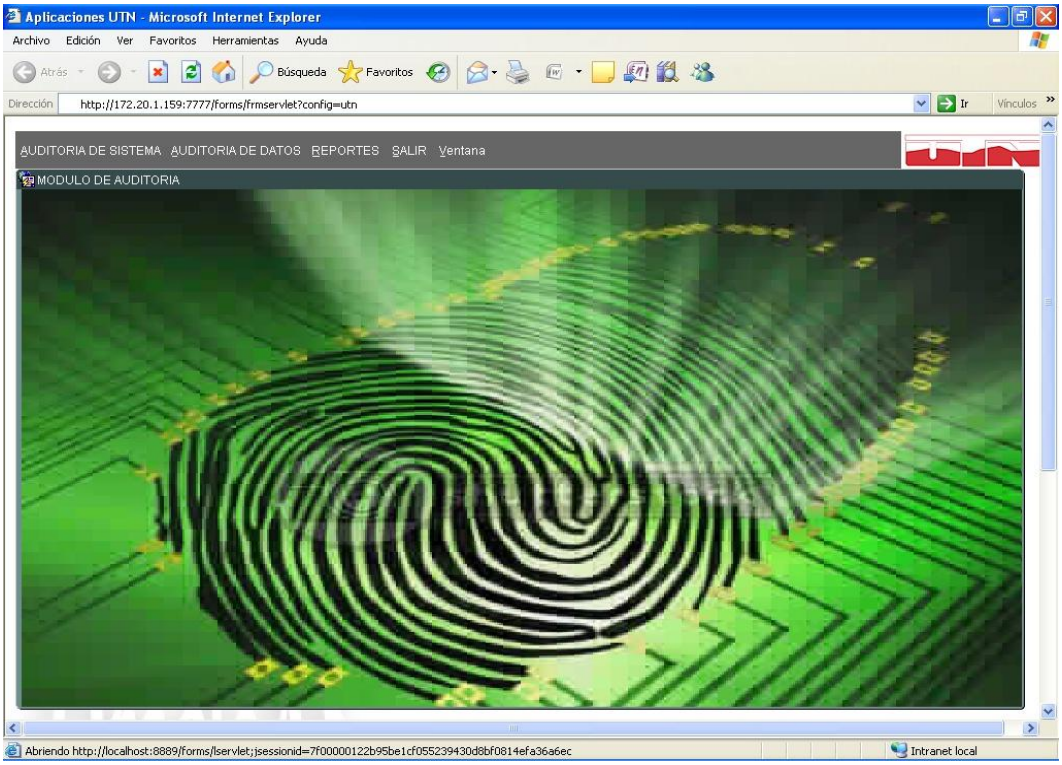


Ilustración 0-14 Menú Principal SABDO

E.2 MANEJO DEL SISTEMA SABDO

Las opciones que se presentan en el menú principal del sistema SABDO se muestran en la fig. 4 las cuales son:

1. Auditoría de Sistema
2. Auditoría de Datos
3. Reportes
4. Salir de la Aplicación

E.2.1 AUDITORÍA DE SISTEMA

La opción de Auditoría de Sistema permite auditar las sesiones, acciones y objetos pertenecientes a la base de datos, contiene el submenú (ver fig. 15) con las opciones:

- Configuración
- Monitoreo

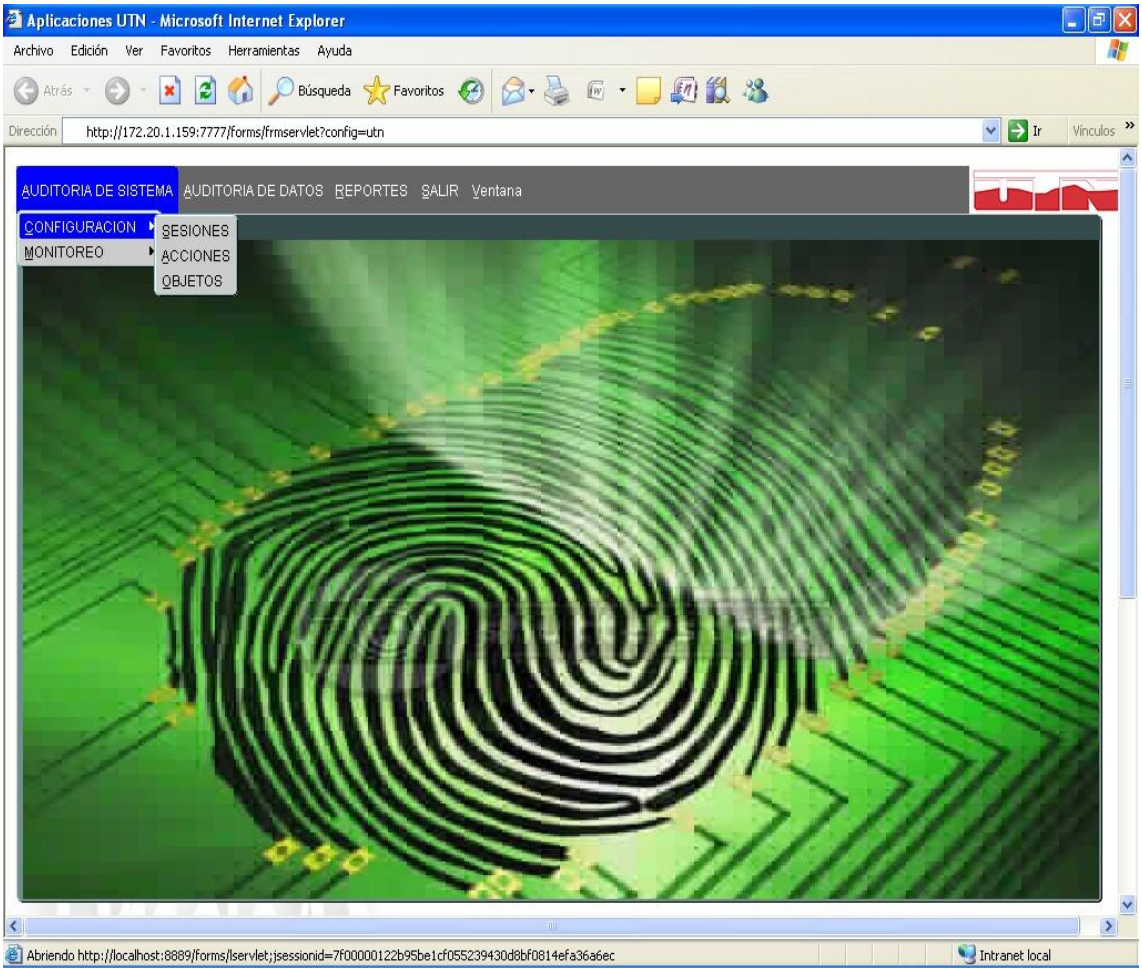


Ilustración 0-15 Menú Auditoría de Sistema

E.2.1.1CONFIGURACIÓN

La opción de configuración contiene otro submenú con las opciones: Sesiones, Acciones y Objetos.

La opción configuración de Sesiones permite auditar la información de los usuarios que hayan realizado o intentado realizar y no lograron la operación **CONNECT.** Es decir permite conocer quienes se han conectado y detectar supuestos intrusos que fallaron la conexión, esta se puede apreciar en la figura 16.

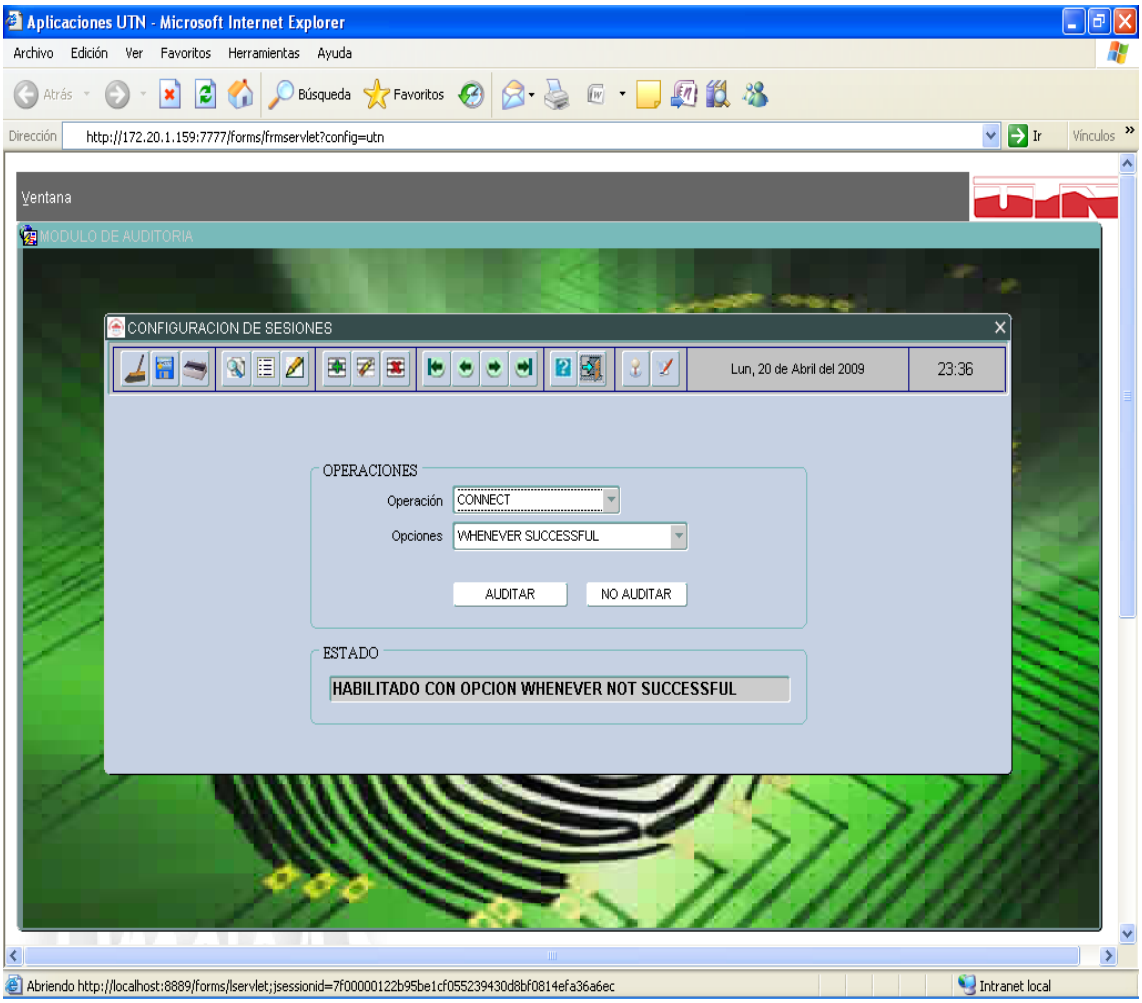


Ilustración 0-16 Menú Configuración de Sesiones

La operación para poder realizar una conexión es: **CONNECT;** Las opciones de conexión pueden ser: **WHENEVER SUCCESSFUL** (Auditar todas las conexiones exitosas), **WHENEVER NOT SUCCESSFUL** (Auditar las conexiones no exitosas, que pueden ser por el ingreso de un password incorrecto superando el límite de intentos permitidos por la BDD) y **ALWAYS** (Permite auditar siempre, sean conexiones exitosas o no).

Tenemos una barra de ***ESTADO*** que visualiza el estado actual de las sesiones. Este menú lo podemos ver en la figura 17.

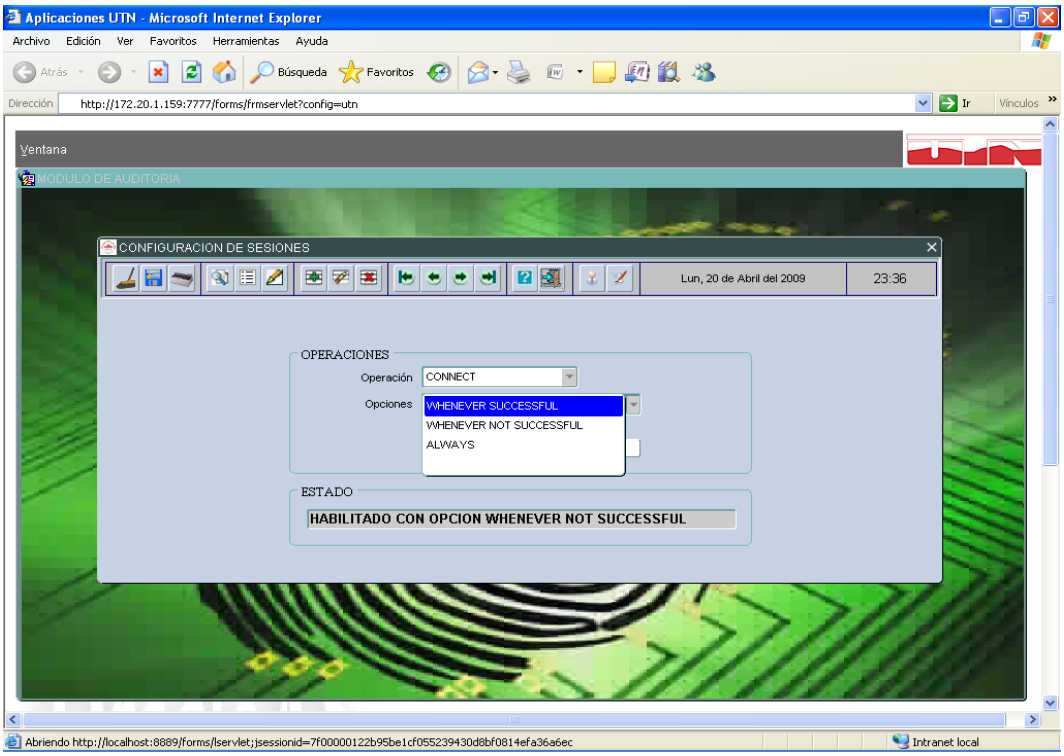


Ilustración 0-17 Opciones de Conexión

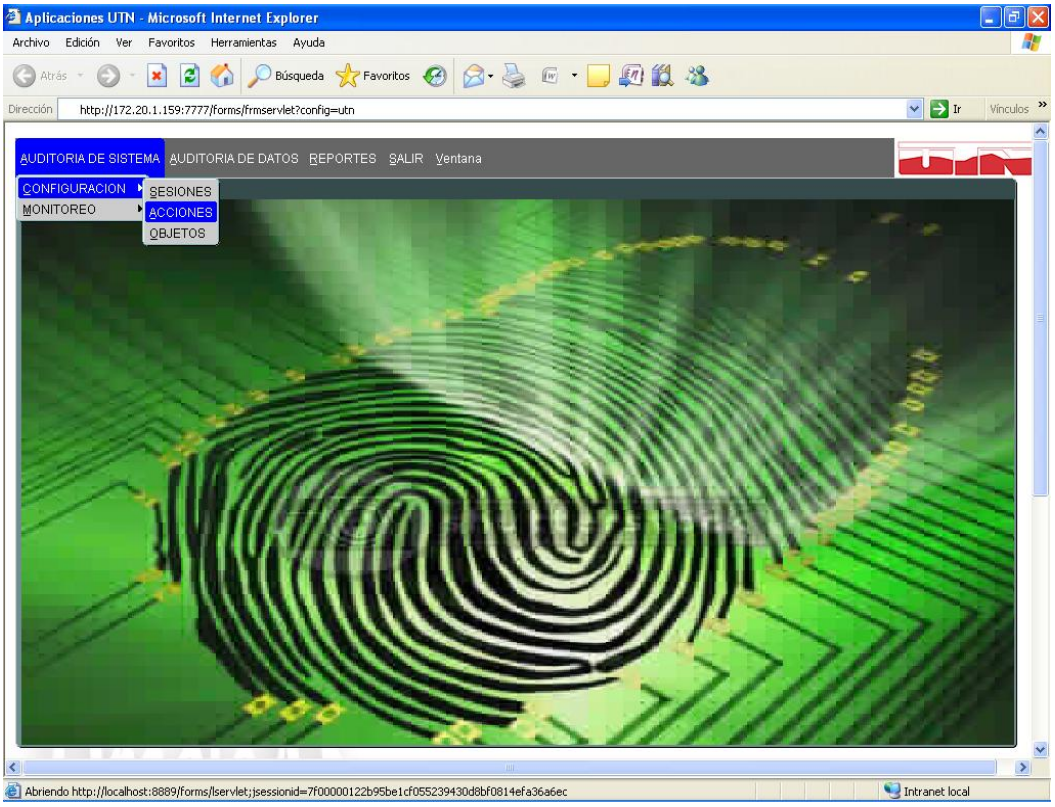


Ilustración 0-18 Configuración de Acciones

La ventana de configuración de acciones permite escoger los objetos o los privilegios sobre los cuales se realizan las diferentes acciones de base de datos (ver la figura 18). Entre los objetos de base de datos tenemos por ejemplo: Tablas, procedimientos, constraints, vistas, índices, triggers, etc. Los privilegios que se pueden escoger son: Crear sesiones, SYSDBA, SYSTEM, SYS.

Se puede visualizar las acciones que afectarán los objetos o privilegios mencionados, las acciones son: *Alter* (alterar), *Create* (crear), *Drop* (Borrar), *Execute* (ejecutar). Para escoger qué acciones se auditarán se marca con el mouse en el chek (ítem) **Auditoría** que se encuentra en el lado derecho de cada acción, al finalizar la configuración se presiona el botón **PROCESAR** para poder ejecutar dicha configuración. Ver figura 19.

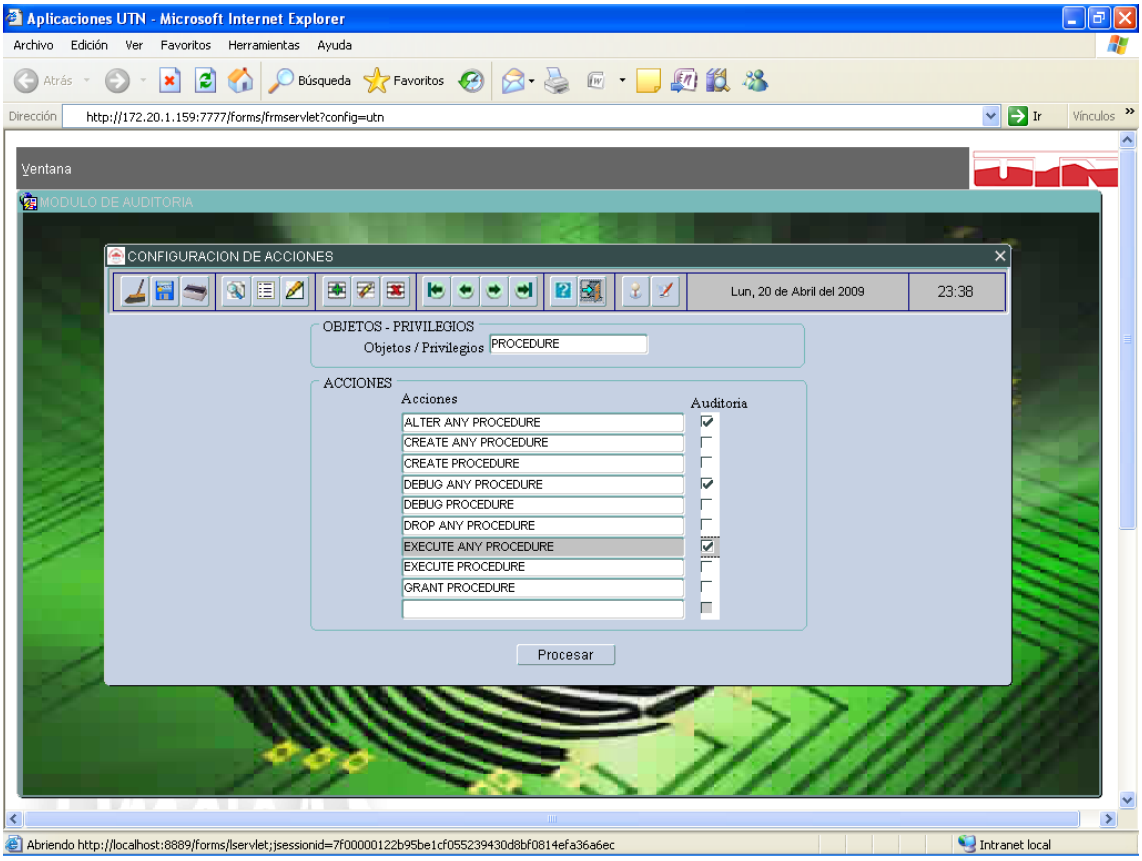


Ilustración 0-19 Menú Configuración de Acciones

La opción de configuración de Objetos puede verse en la figura 20, se realiza cuando se desea conocer las acciones de un usuario específico sobre un grupo de objetos (Tablas, triggers , vistas , constraints, etc.).

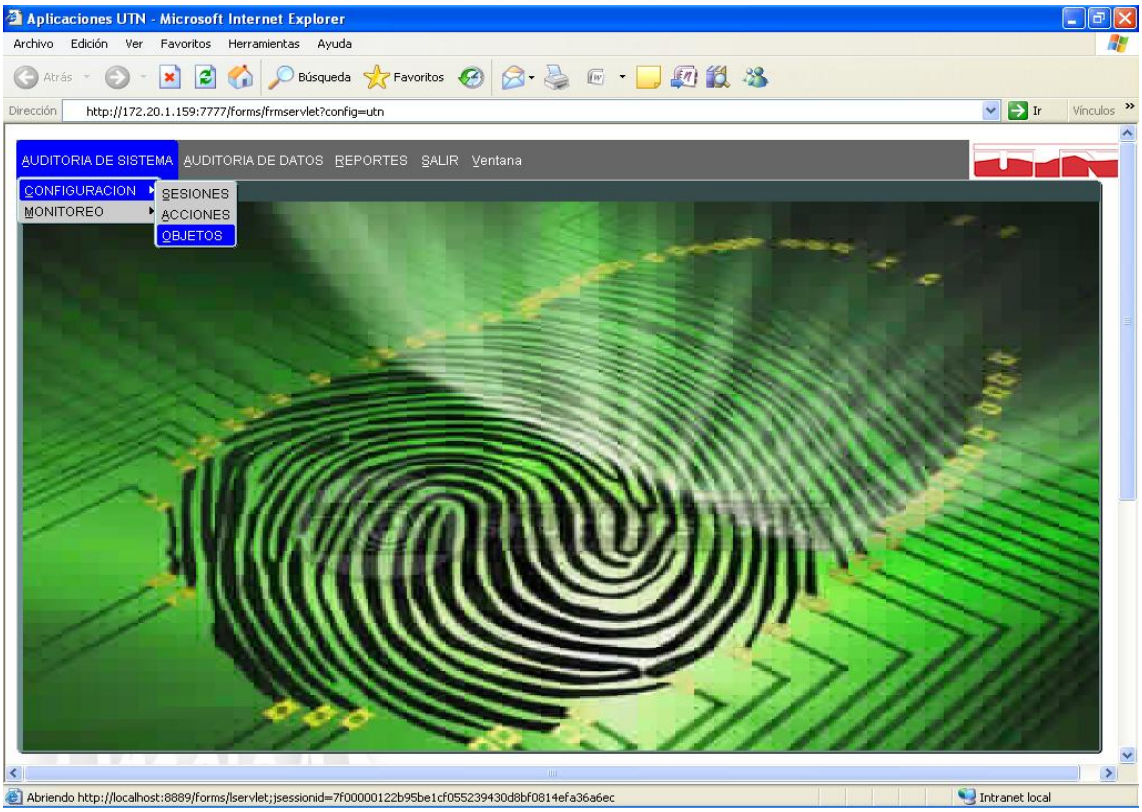


Ilustración 0-20 Configuración de Objetos

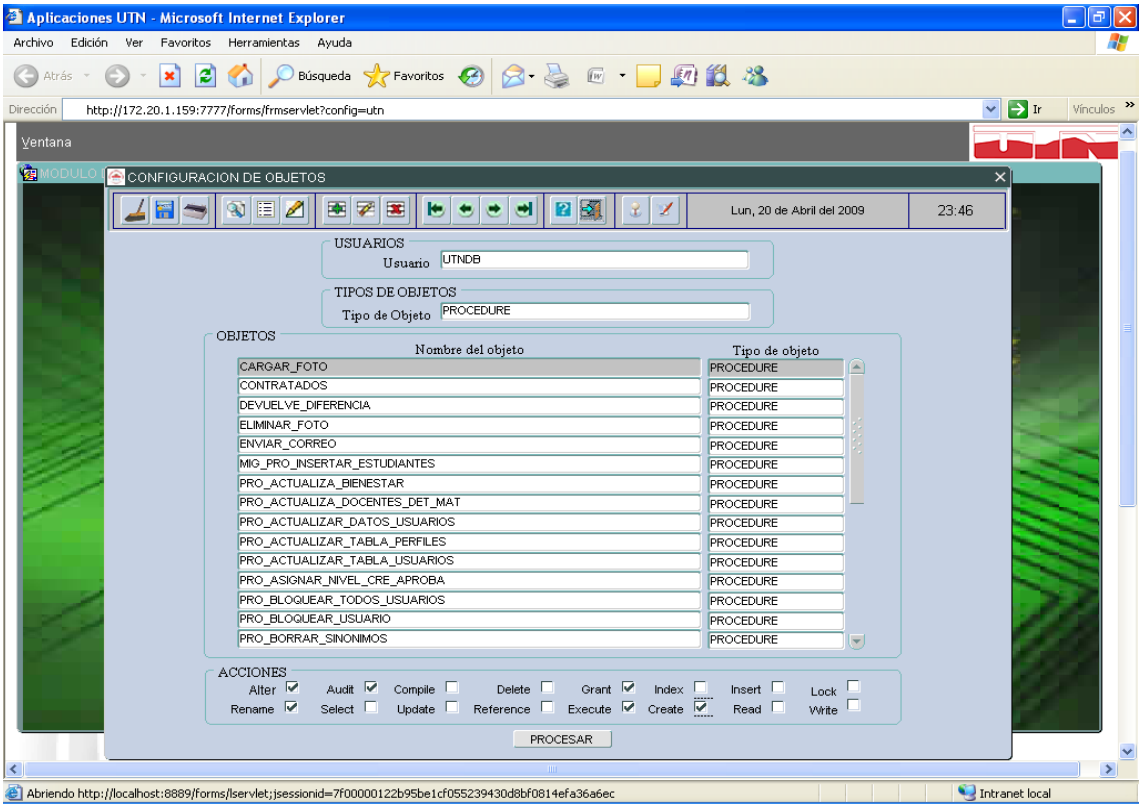


Ilustración 0-21 Menú Configuración de Objetos

En la figura 21 podemos ver que la ventana de configuración de objetos contiene los siguientes ítems: Usuarios, Tipo de Objeto, Objetos y Acciones.

En nuestro caso el usuario es utndb, el tipo de objeto específico que se desea auditar puede escogerse de una pantalla desplegable y son los que ya mencionamos anteriormente, igualmente se despliegan los objetos relacionados a este tipo. Las acciones que se pueden auditar son: Alter (alterar), Audit (auditar), Compile (compilar), Delete (borrar), Grant (otorgar privilegio), Index (índice), Insert (insertar), Lock (bloquear), Rename (renombrar), Select (seleccionar), Update (actualizar), Reference (referencias), Execute (ejecutar) Create (crear), Read (leer), Write (escribir) y se muestran en un menú de ítems que pueden ser marcados fácilmente con el mouse.

E.2.1.2MONITOREO

La opción de Auditoría de Sistema tiene además la opción de Monitoreo, la cual permite visualizar la información de auditoría de sistema de BDD Oracle que fue configurada en las opciones anteriores. El monitoreo se realiza para: Sesiones y Objetos (acciones realizadas sobre estos). Ver figura 22.

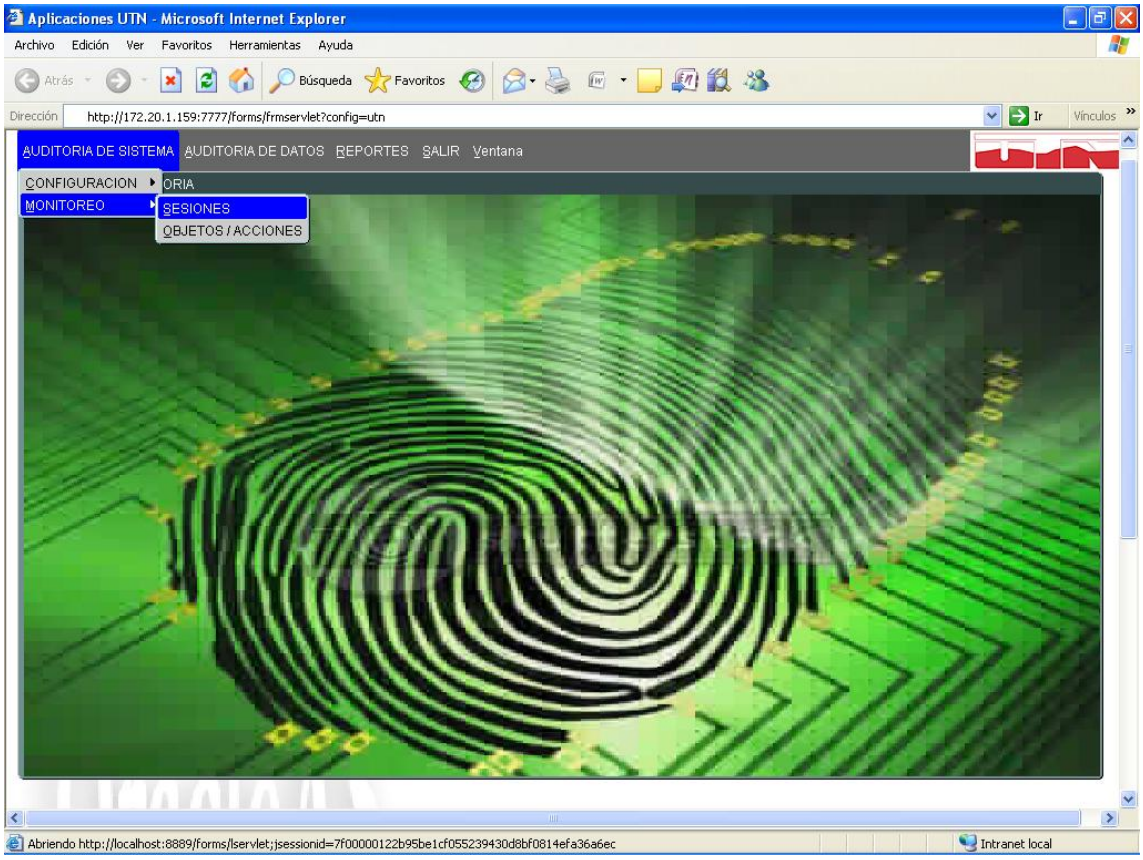


Ilustración 0-22 Monitoreo de Sesiones

Al escoger la opción de Sesiones del menú monitoreo de Sistema (fig. 22) obtenemos como resultado la siguiente información (Ver figuras: 23, 24, 25, 26) : Nombre de usuario, código del Terminal, Nombre del Usuario de Sistema Operativo, Estado de conexión, y un menú navegable que contiene información adicional como: Fecha y hora de conexión de usuario para Ingreso y salida creadas por AUDIT SESSION (ver fig. 23); Acción [contiene el nombre del tipo de acción : ACTION NAME] , Userhost [contiene el nombre de la máquina anfitrión del cliente]y Log off (ver fig. 24); Logoff Lread [es la lectura lógica de la sesión] , Logoff Pread [es la lectura física de la sesión] y Logoff Lwrite [es la escritura lógica de las sesión](Ver fig. 25); Logoff Dlock [son los puntos muertos detectados durante la sesión] y Sessionid [es el identificador numérico para cada sesión Oracle] (ver fig. 26).

La información mostrada en las pantallas será la escogida durante la configuración de la auditoría de sistemas.

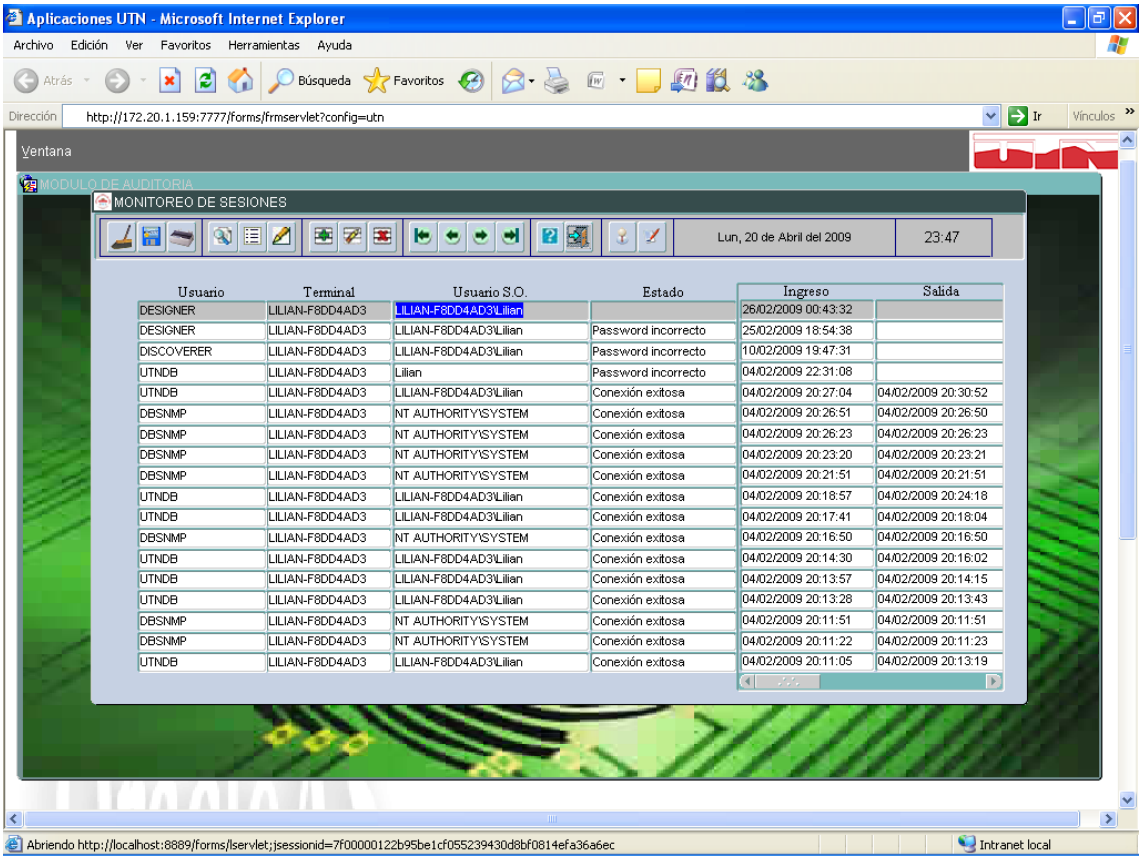


Ilustración 0-23 Ventana de Monitoreo de Sesiones 1

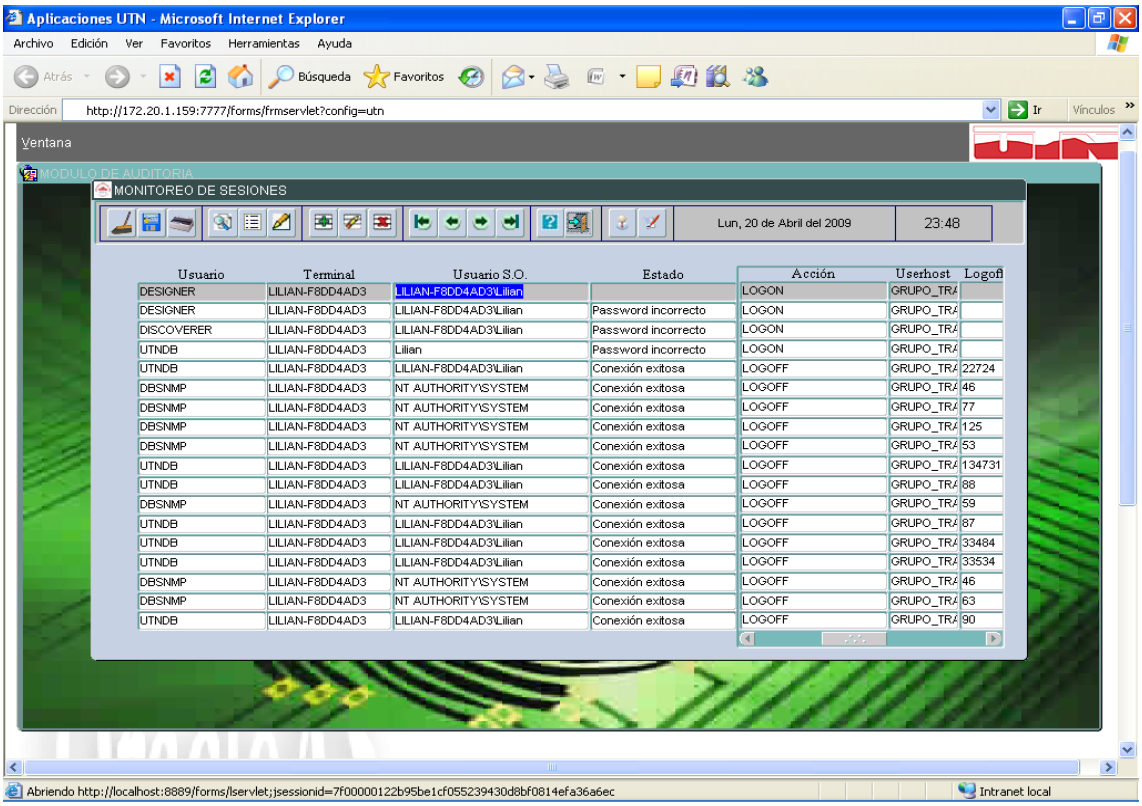


Ilustración 0-24 Visualización de Monitoreo de Sesiones 2

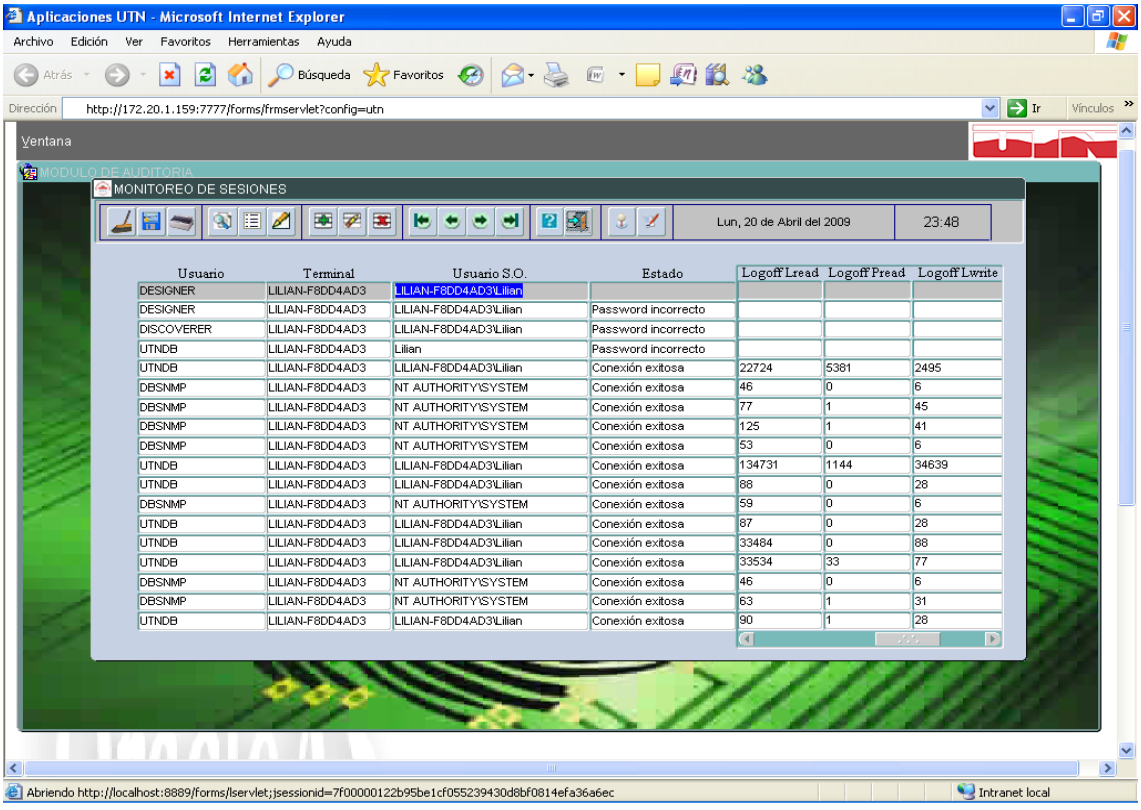


Ilustración 0-25 Visualización de Monitoreo de Sesiones 3

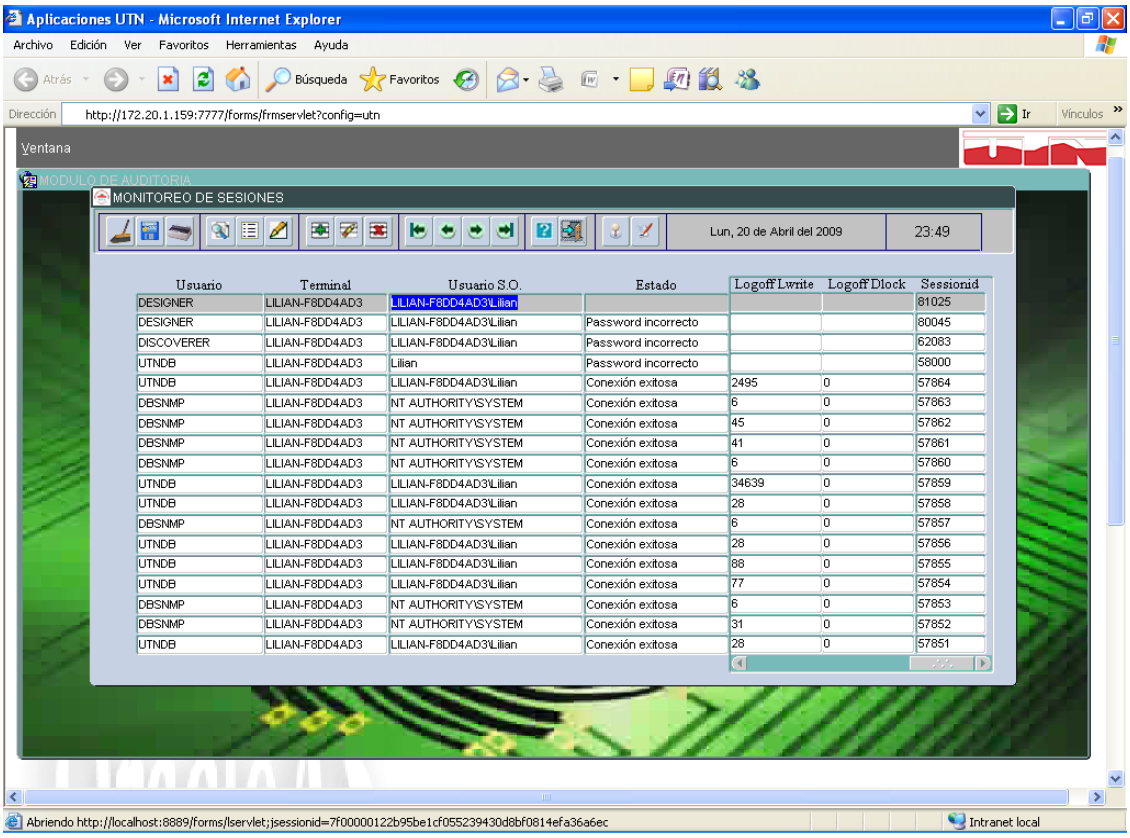


Ilustración 0-26 Visualización de Monitoreo de Sesiones 4

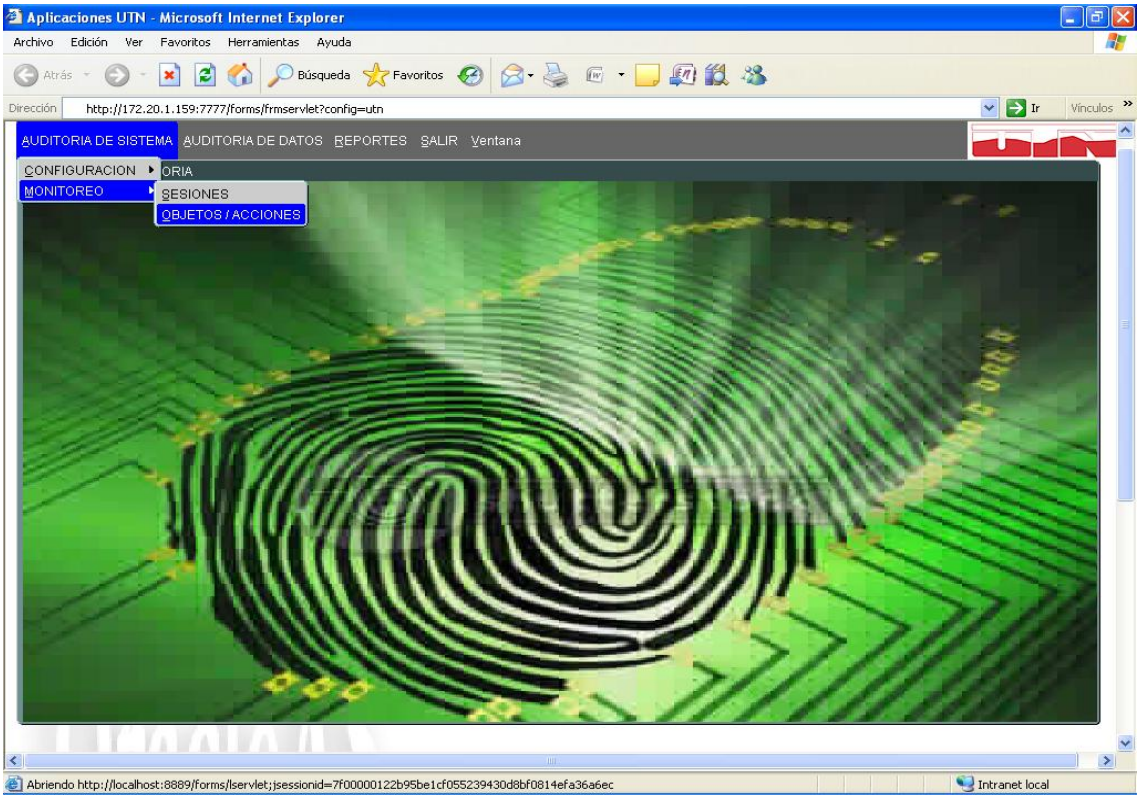


Ilustración 0-27 Monitoreo de Objetos/Acciones

La opción de monitoreo de objetos y acciones de sistema se ilustra en la figura 27. La información desplegada en el monitoreo de objetos es la escogida durante la configuración de auditoría de sistemas mencionada anteriormente. Por ejemplo tenemos: Os Username [cuenta de usuario de sistema operativo asociado al usuario de BDD cuyas acciones fueron auditadas], Username [nombre del usuario cuyas acciones fueron auditadas], Userhost [nombre de la máquina anfitrión del cliente], Terminal [Identificador del terminal del usuario], acompañado de un menú navegable que contiene datos adicionales. Ver figura 28.

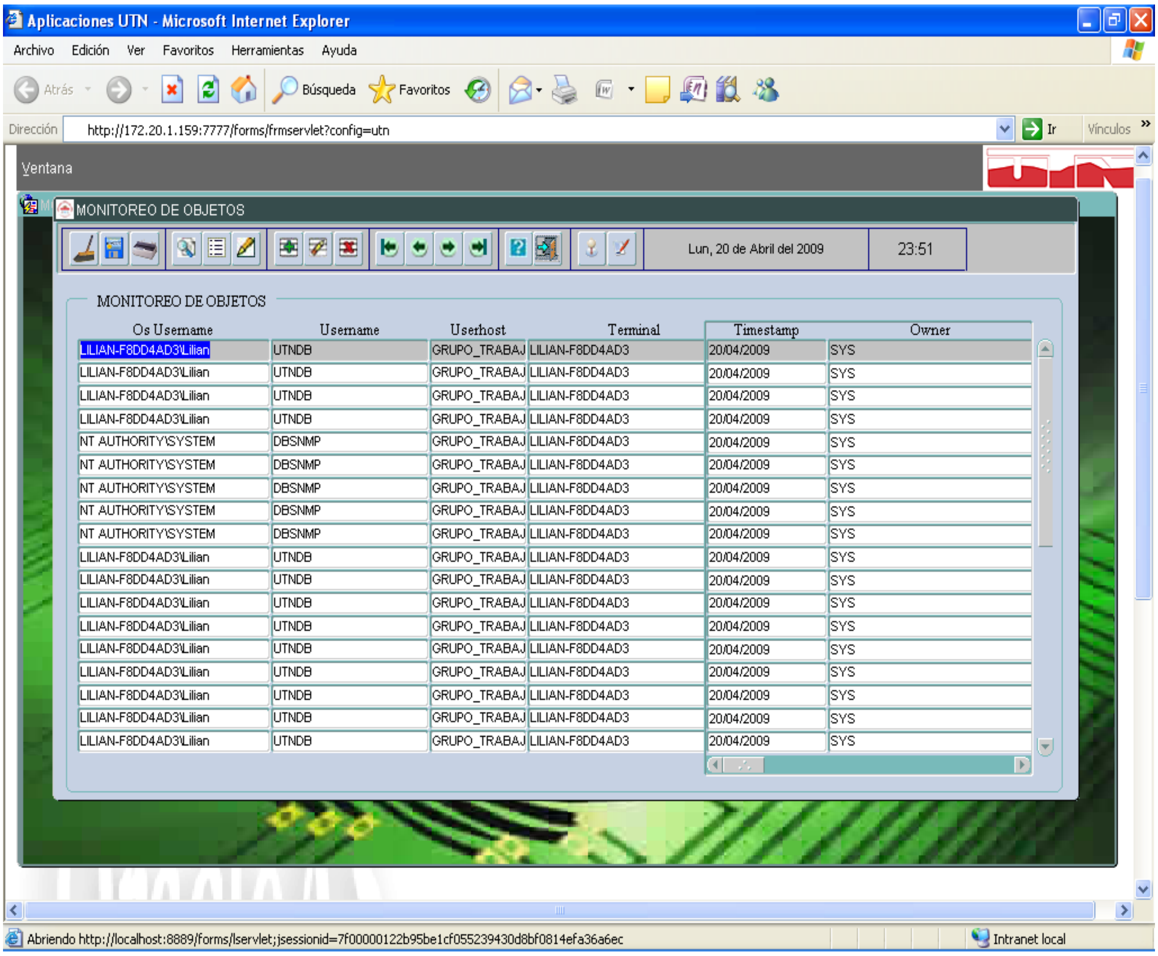


Ilustración 0-28 Monitoreo de Objetos Acciones 1

Entre los datos adicionales tenemos por ejemplo: TIMESTAMP [fecha y hora de conexión de usuario para entradas creadas por AUDIT SESSION], OWNER [es el creador del objeto afectado por la acción] (ver fig. 28), OBJNAME [nombre del objeto afectado por la acción] (ver fig. 29), ACTION NAME [Nombre del tipo de acción realizada], NEW

OWNER [Propietario del objeto referenciado en la columna NEW_NAME] (ver fig. 30), NEW NAME [Nuevo nombre de un objeto después de un RENAME], SES ACTIONS [Sesión resumen (una cadena de 16 caracteres, uno por cada tipo de acción en el orden: ALTER, AUDIT, COMMENT, DELETE, GRANT, INDEX, INSERT, LOCK, RENAME, SELECT, UPDATE, REFERENCES, y EXECUTE. Posiciones 14,15 y 16 son reservadas para uso futuro. Los caracteres son: - para ninguna, S para éxito, F para fracaso, and B para los dos)] (ver fig. 31), COMMENT TEXT [Comentarios de texto sobre la pista de auditoría, insertada por la aplicación], SESSIONID [Identificador numérico para cada sesión] (ver fig. 32), ENTRYID [Identificador numérico para cada entrada de pista de auditoría en la sesión], STATEMENTID [Identificador numérico para cada sentencia ejecutada], RETURNCODE [Código de error Oracle generado por la acción. Algunos valores útiles:0 = Acción sucedida, 2004 = Violación de seguridad](ver fig.33), PRIV USED [Privilegio de sistema usado para ejecutar la acción] (ver fig. 34).

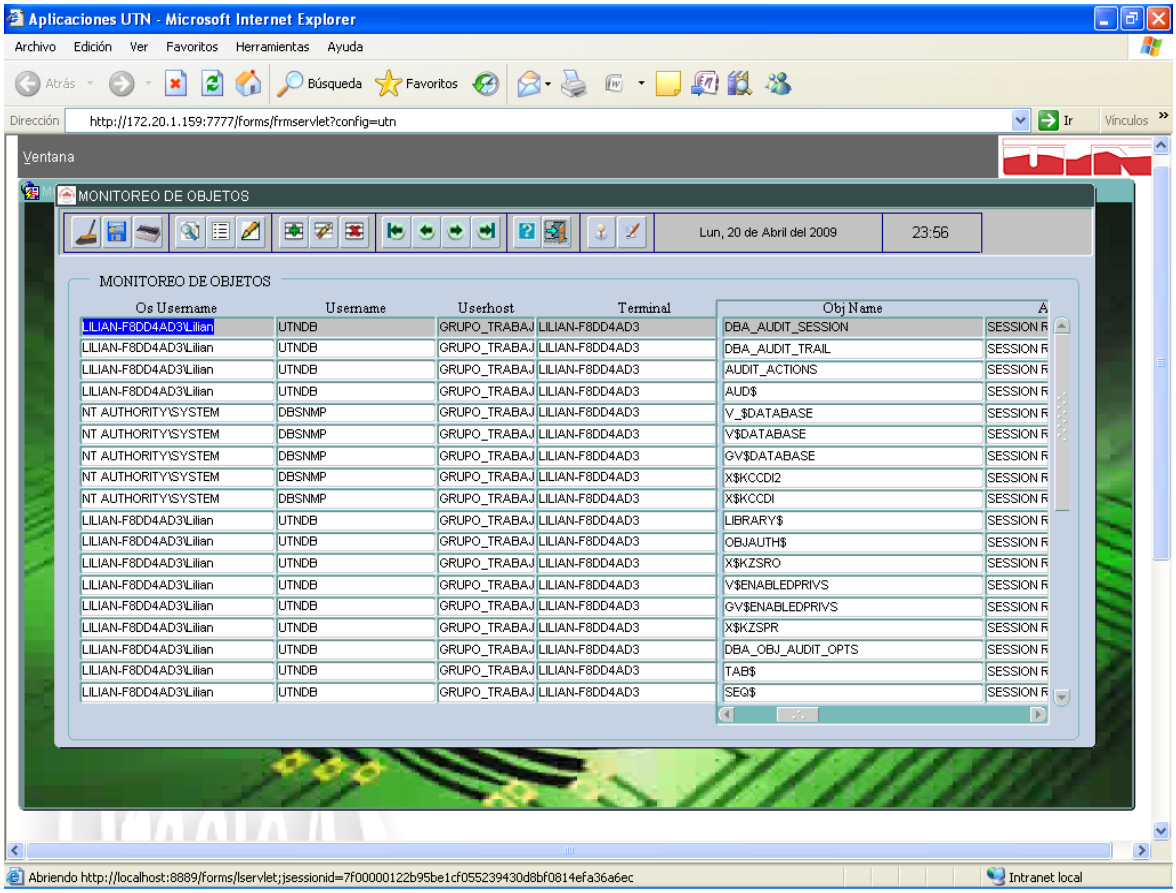


Ilustración 0-29 Monitoreo de Objetos Acciones 2

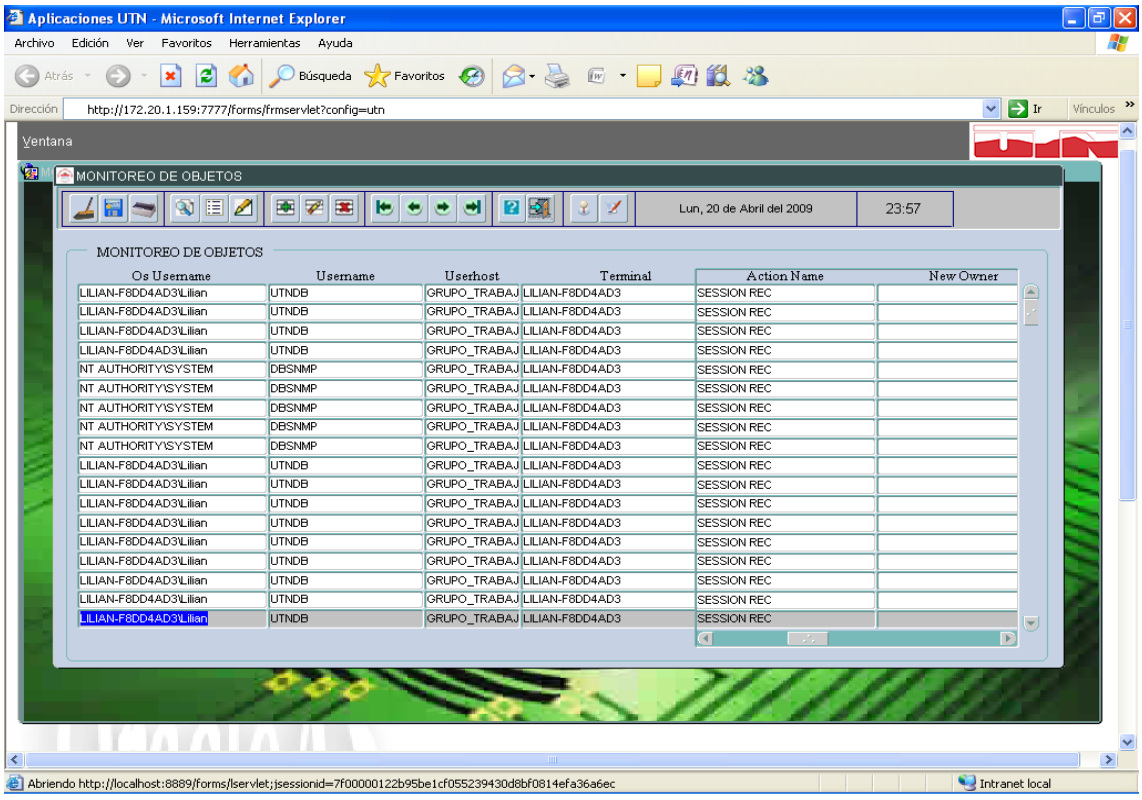


Ilustración 0-30 Monitoreo de Objetos Acciones 3

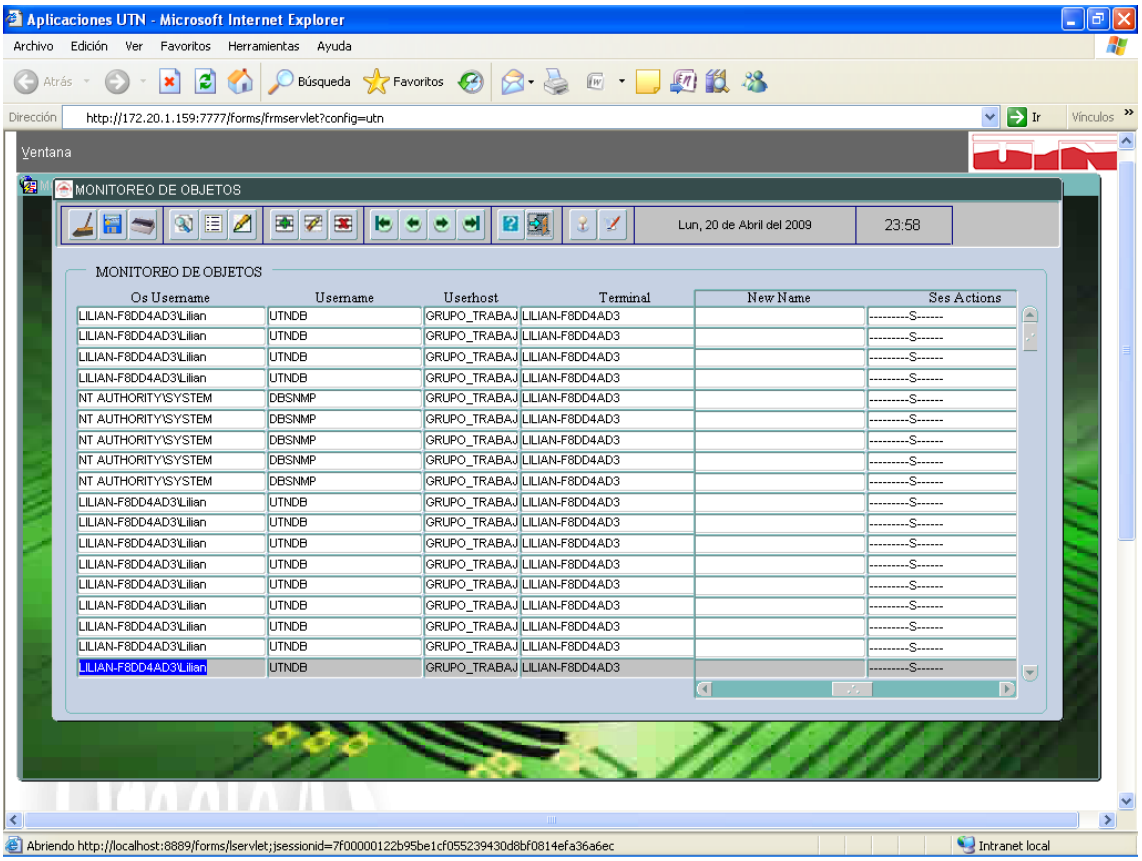


Ilustración 0-31 Monitoreo de Objetos Acciones 4

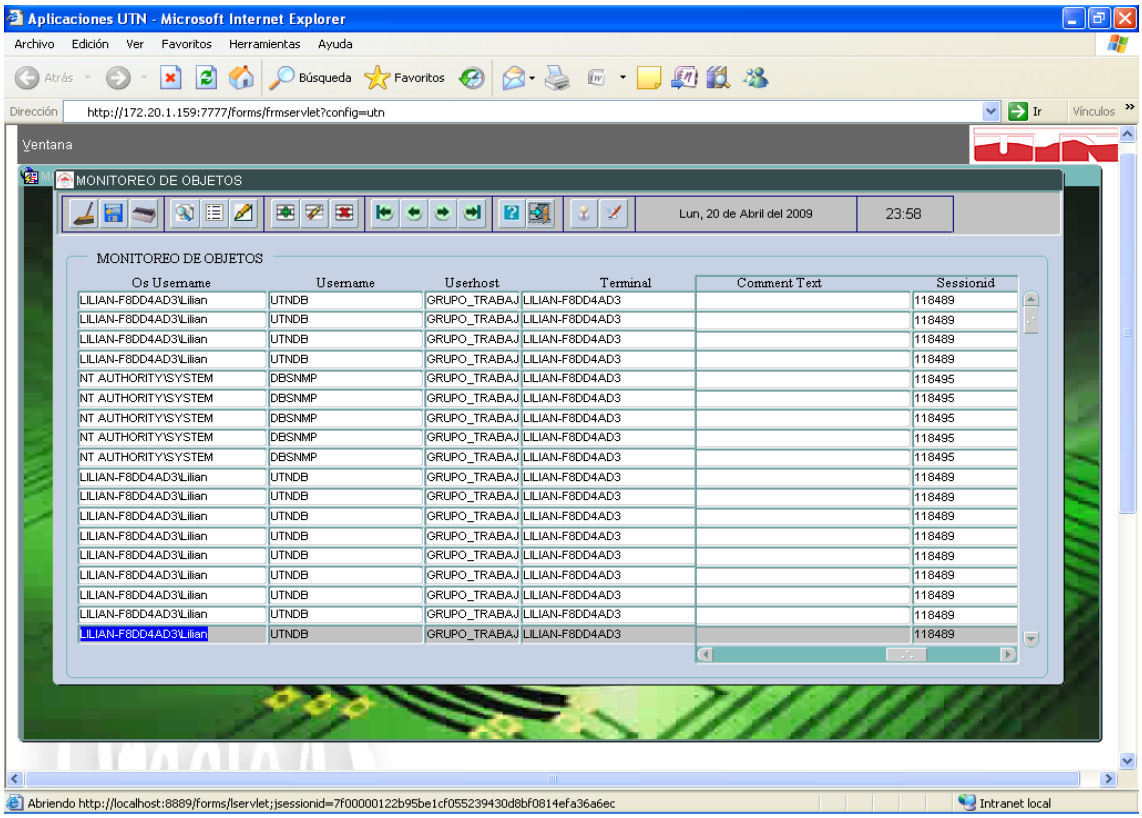


Ilustración 0-32 Monitoreo de Objetos Acciones 5

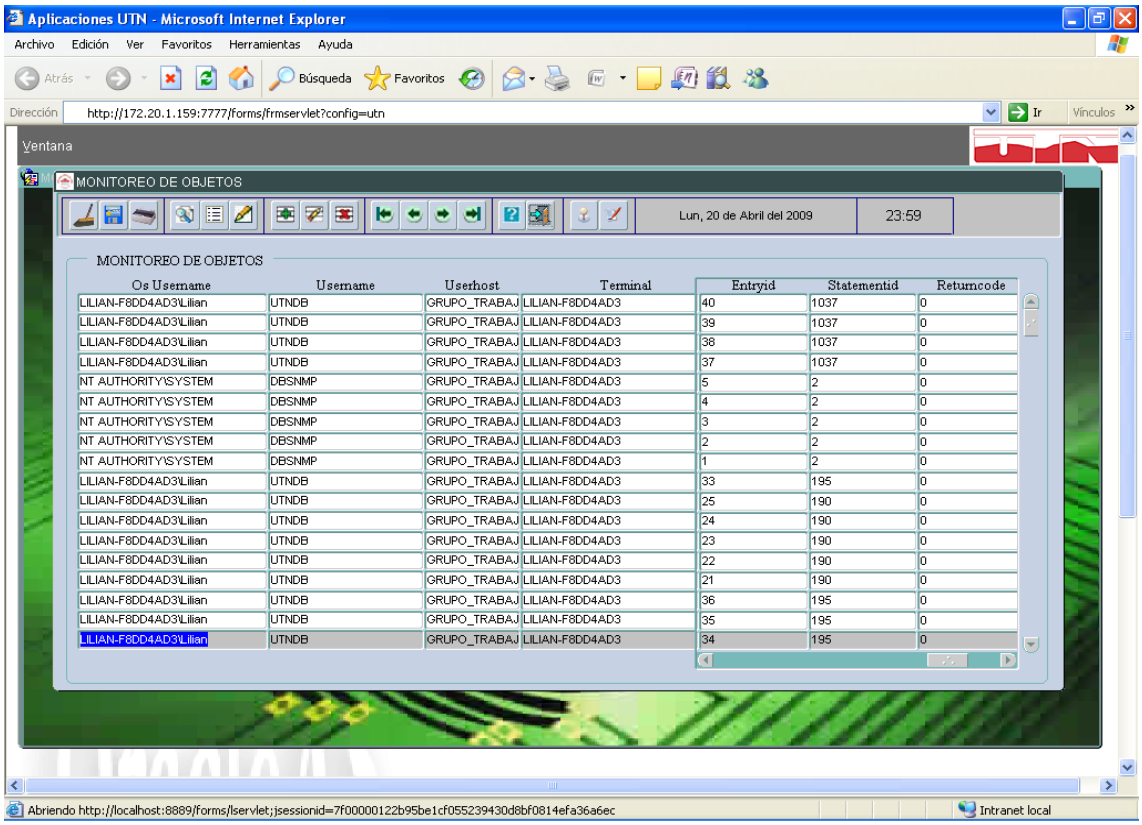


Ilustración 0-33 Monitoreo de Objetos Acciones 6

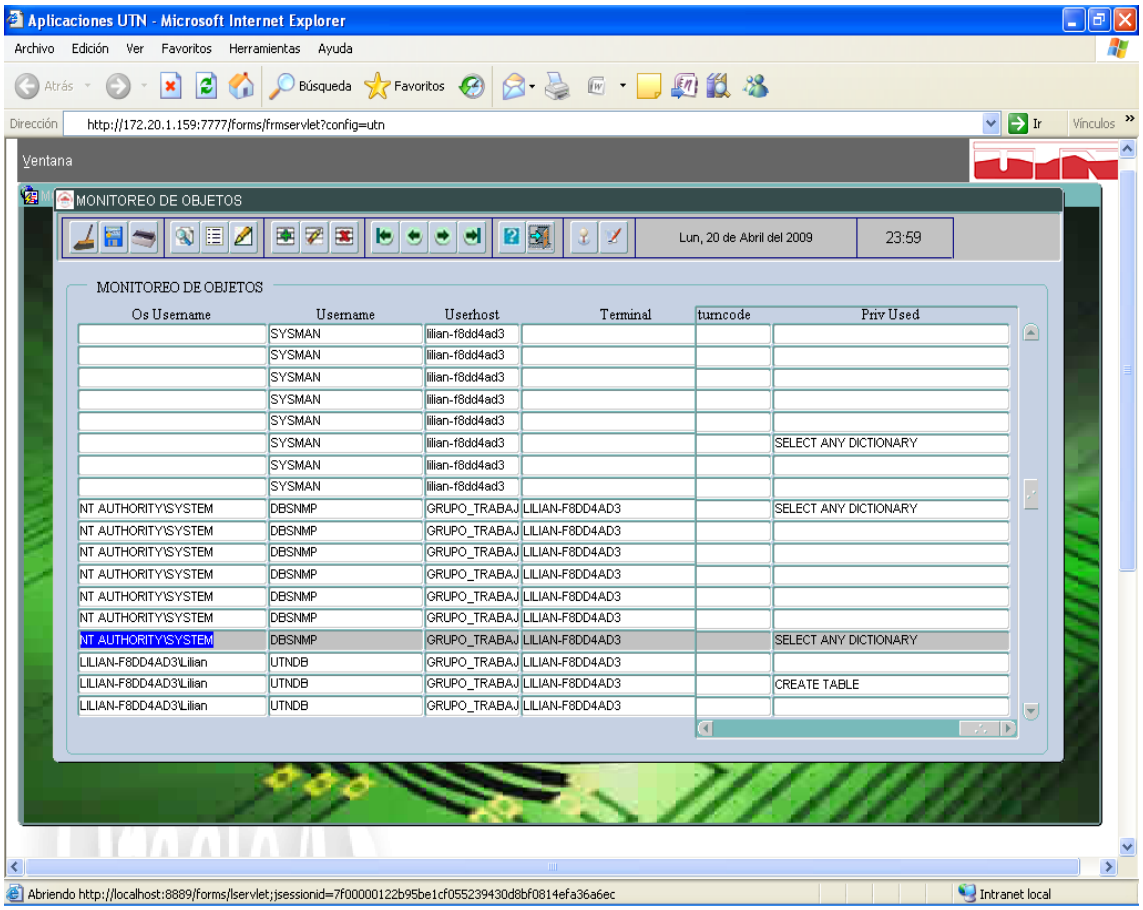


Ilustración 0-34 Monitoreo de Objetos Acciones 7

E.2.2 AUDITORÍA DE DATOS

La auditoría de datos presenta la información detallada de los datos que han sido modificados, adicionados, borrados manteniendo un registro de los valores anteriores y actuales de los datos. Esta opción contiene un submenú (ver figura 35) con las opciones:

- Configuración
- Monitoreo

E.2.2.1CONFIGURACIÓN

Permite configurar las opciones de auditoría para los datos contenidos en la base de datos, ver figuras 36 y 37. Se puede elegir: el usuario, las tablas y sus columnas, las acciones que se pueden ejecutar con los datos como son: insertar, eliminar y actualizar. Para finalmente elegir si se empieza a auditar o detener la auditoría.

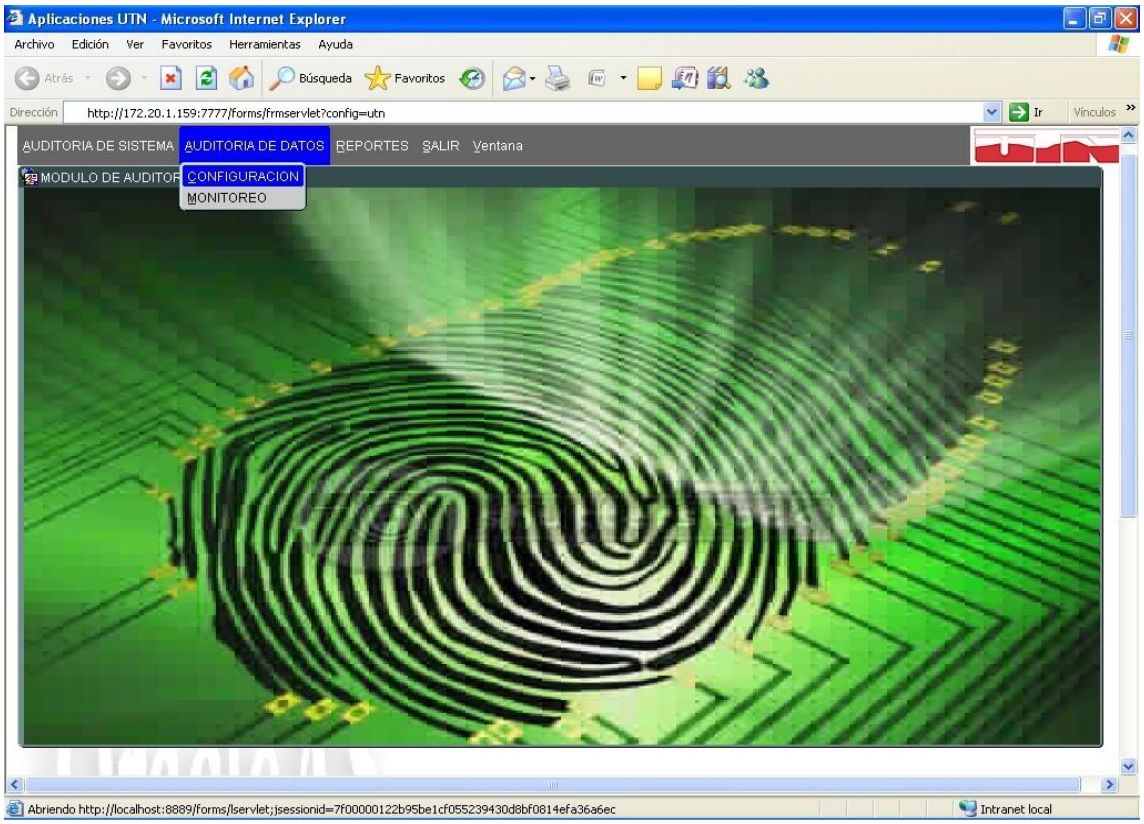


Ilustración 0-35 Menú Auditoría de Datos

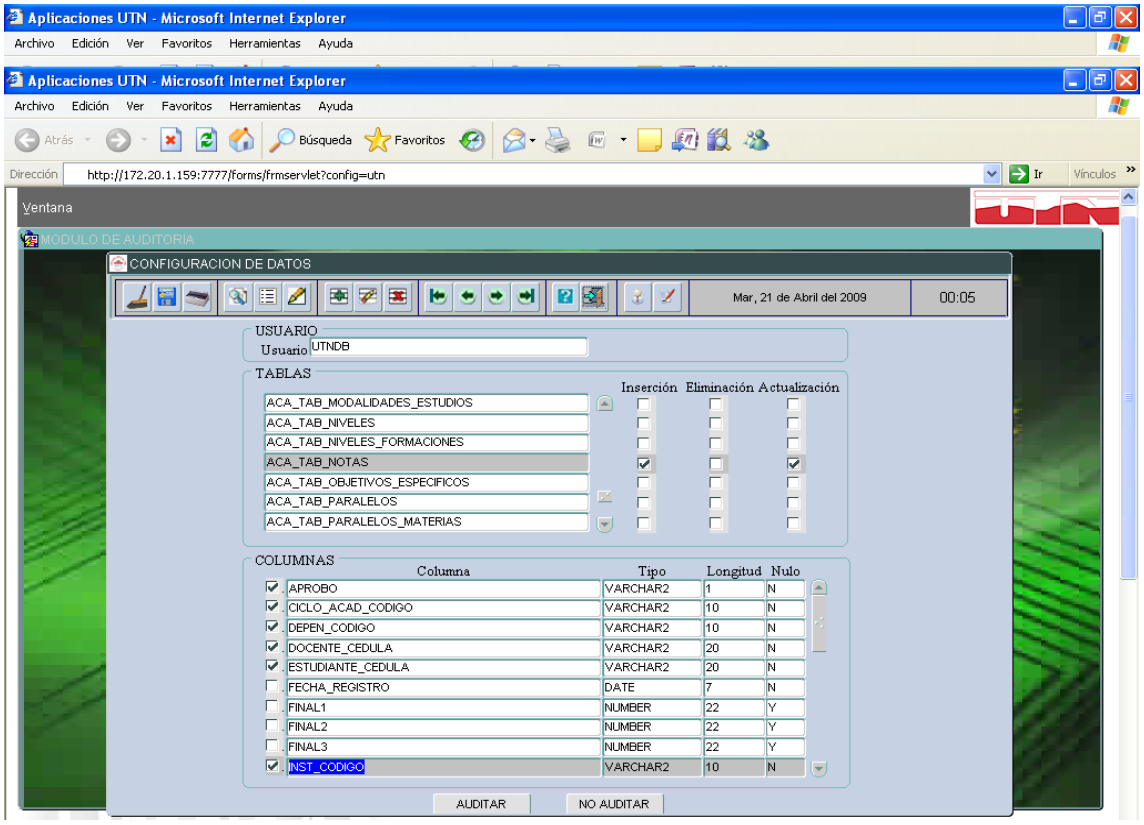


Ilustración 0-36 Configuración de Tablas 1 para Auditoría de Datos

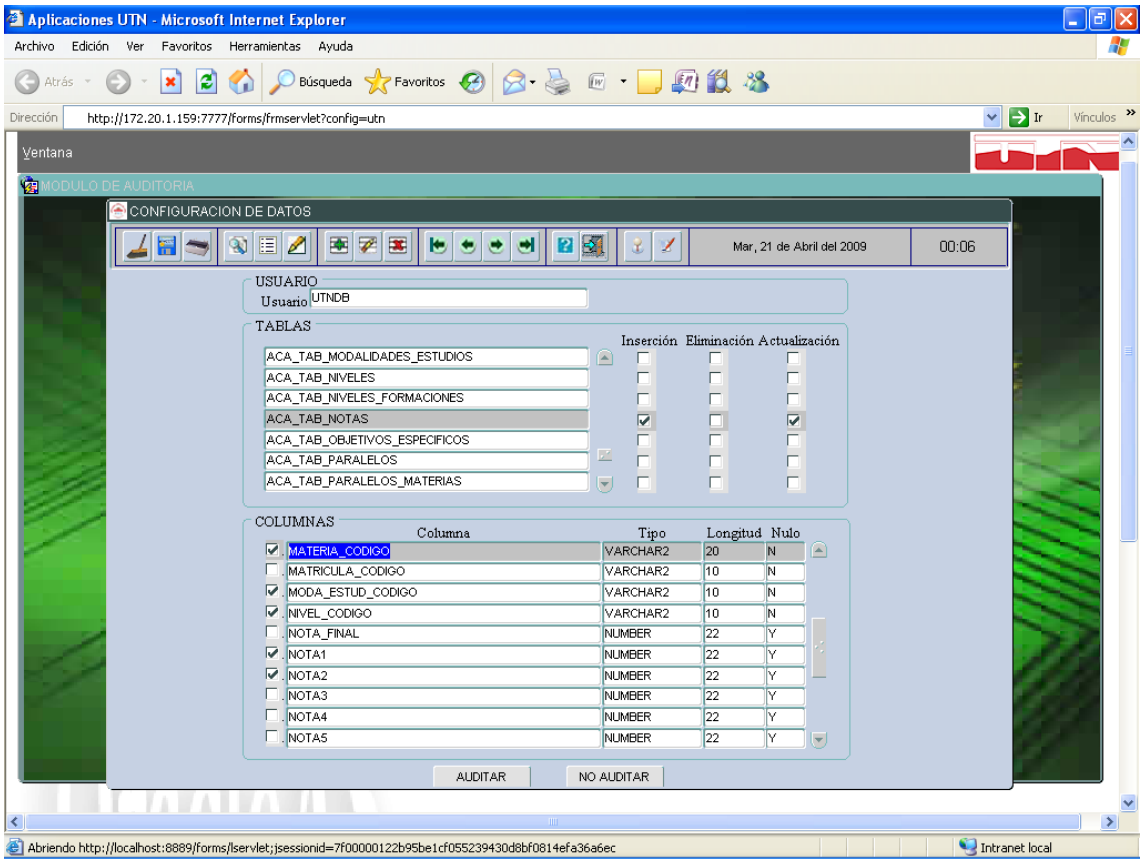


Ilustración 0-37 Configuración de Tablas 2 para Auditoría de Datos

E.2.2.2MONITOREO

El menú de monitoreo de datos se aprecia en la figura 38. La opción de monitoreo despliega la información solicitada en la configuración de la auditoría de datos, la cual varía dependiendo de las columnas escogidas, si se pasa a otra tabla, las columnas mostradas son las pertenecientes a dicha tabla. Por ejemplo en el menú de la fig. 36 se escogieron: la tabla ACA_TAB _NOTAS , las acciones de inserción (INSERT) y actualización (UPDATE) de las columnas: APROBO , CICLO_ACAD_CODIGO (fig. 38), DEPEN_CODIGO, DOCENTE_CEDULA (fig. 39), ESTUDIANTE_CEDULA, INST_CODIGO (fig. 40). En el menú de la fig. 37 se cambiaron las columnas a auditarse: MATERIA_CODIGO, MODA_ESTUD_CODIGO (ver fig. 41), NIVEL_CODIGO, NOTA1, NOTA2 (ver fig. 42)

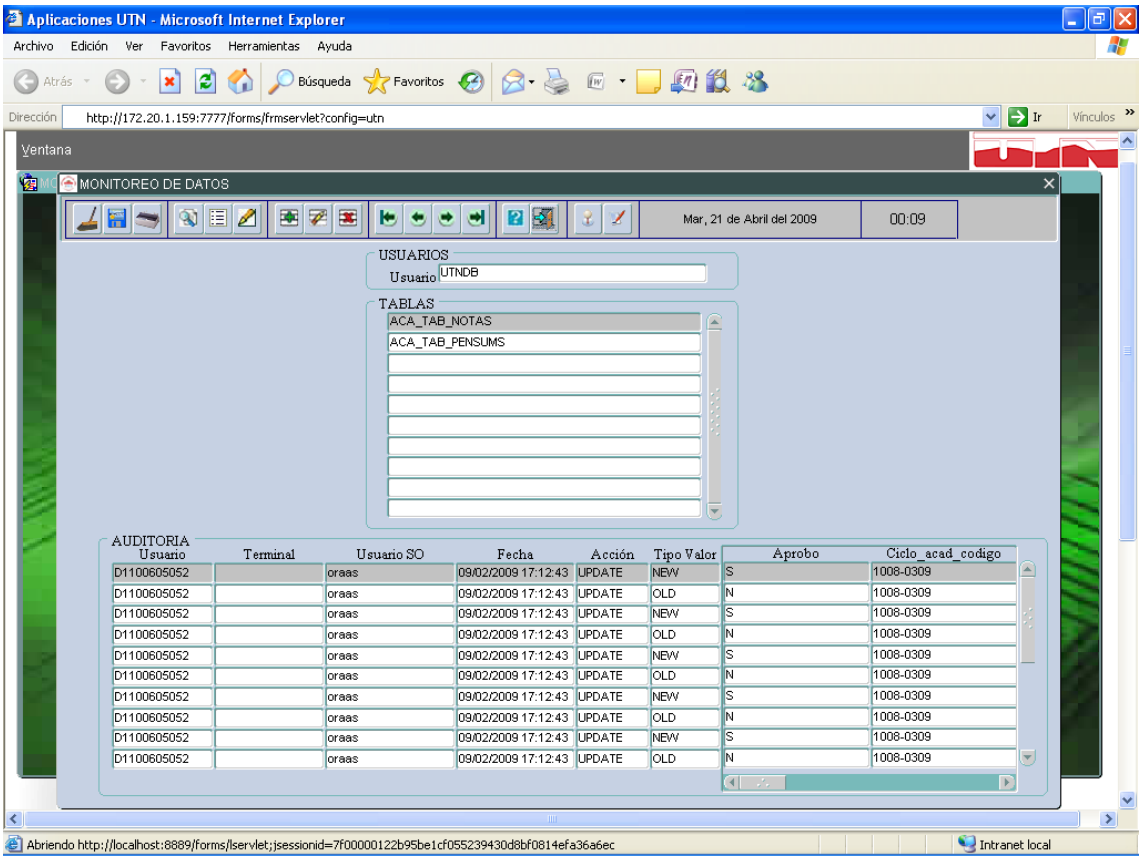


Ilustración 0-38 Monitoreo de Datos 1

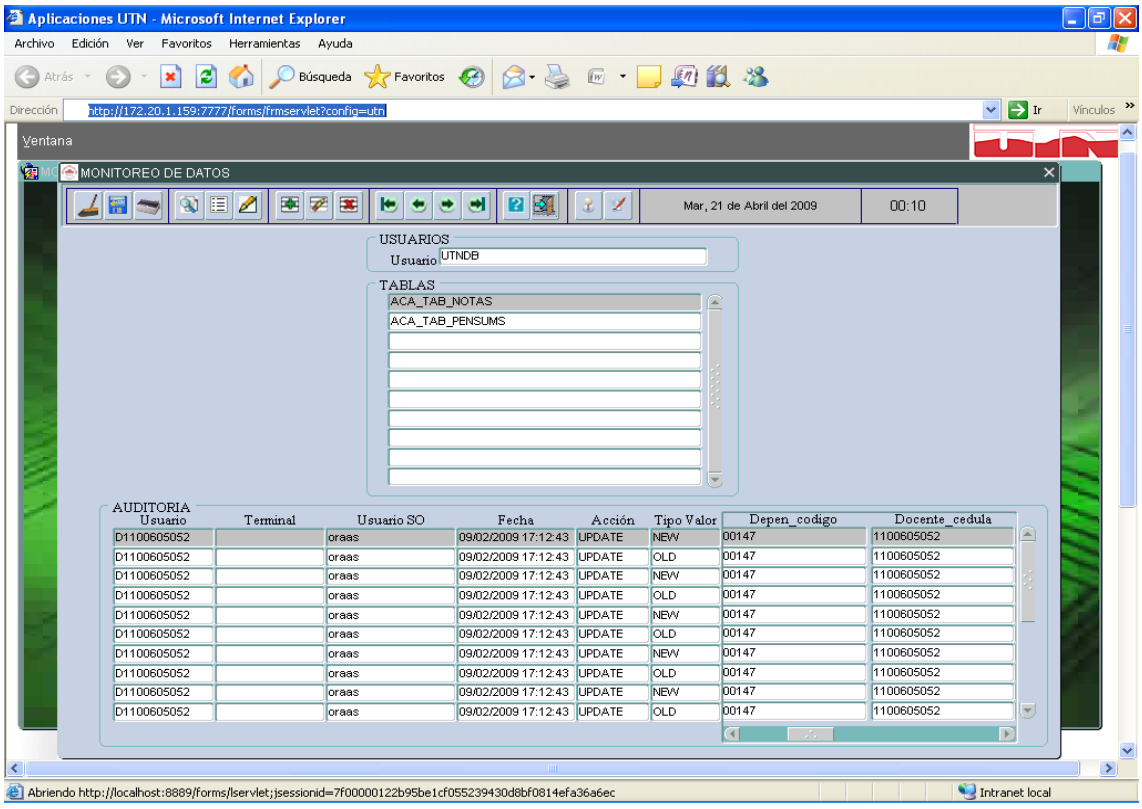


Ilustración 0-39 Monitoreo de Datos 2

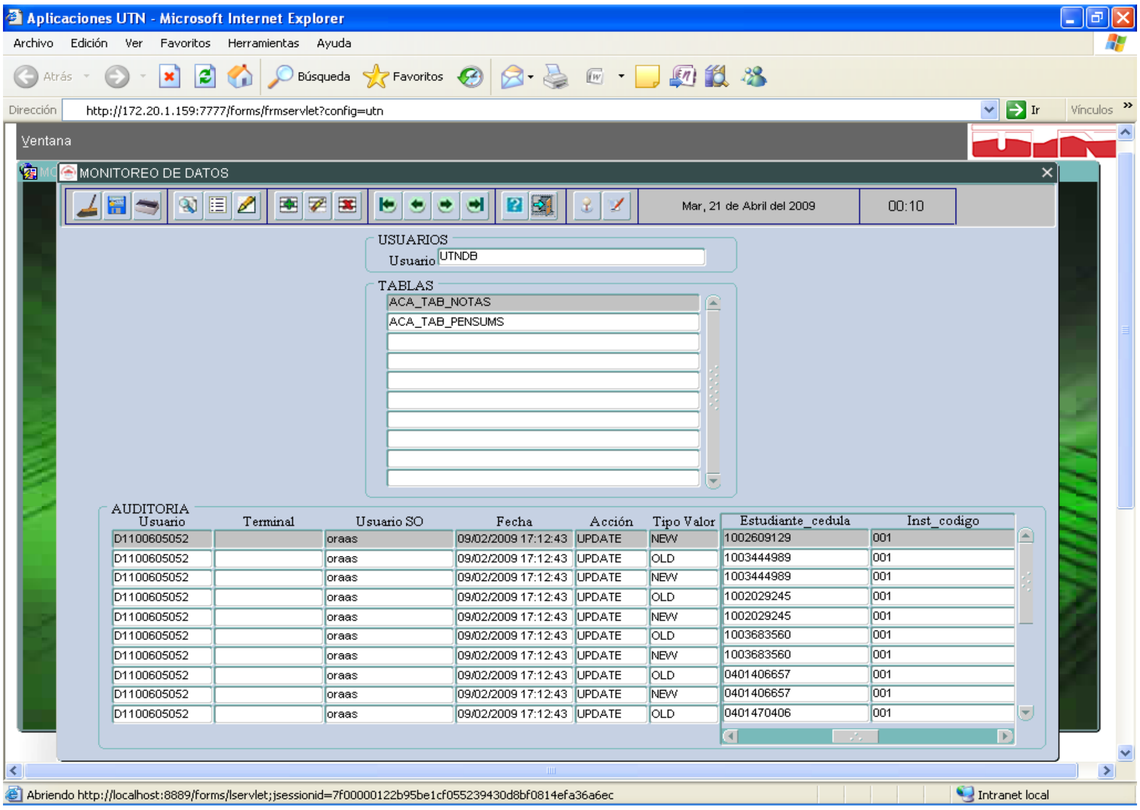


Ilustración 0-40 Monitoreo de Datos 3

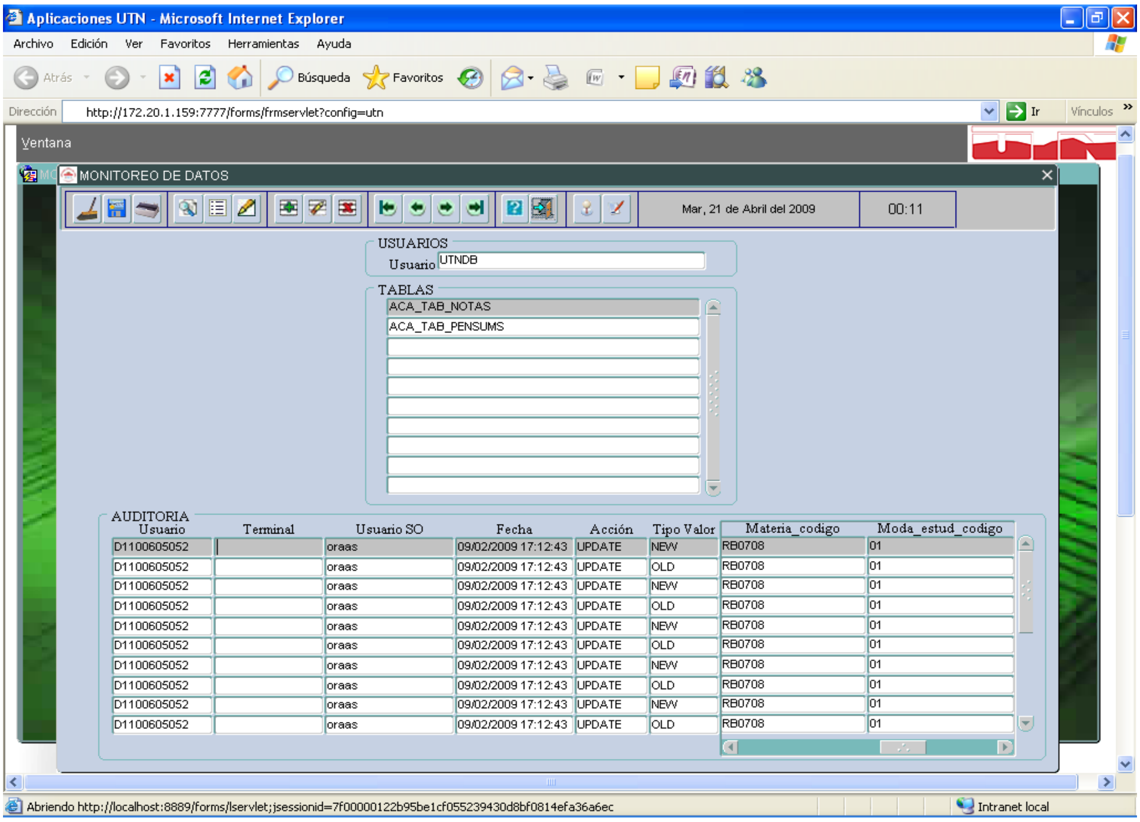


Ilustración 0-41 Monitoreo de Datos 4

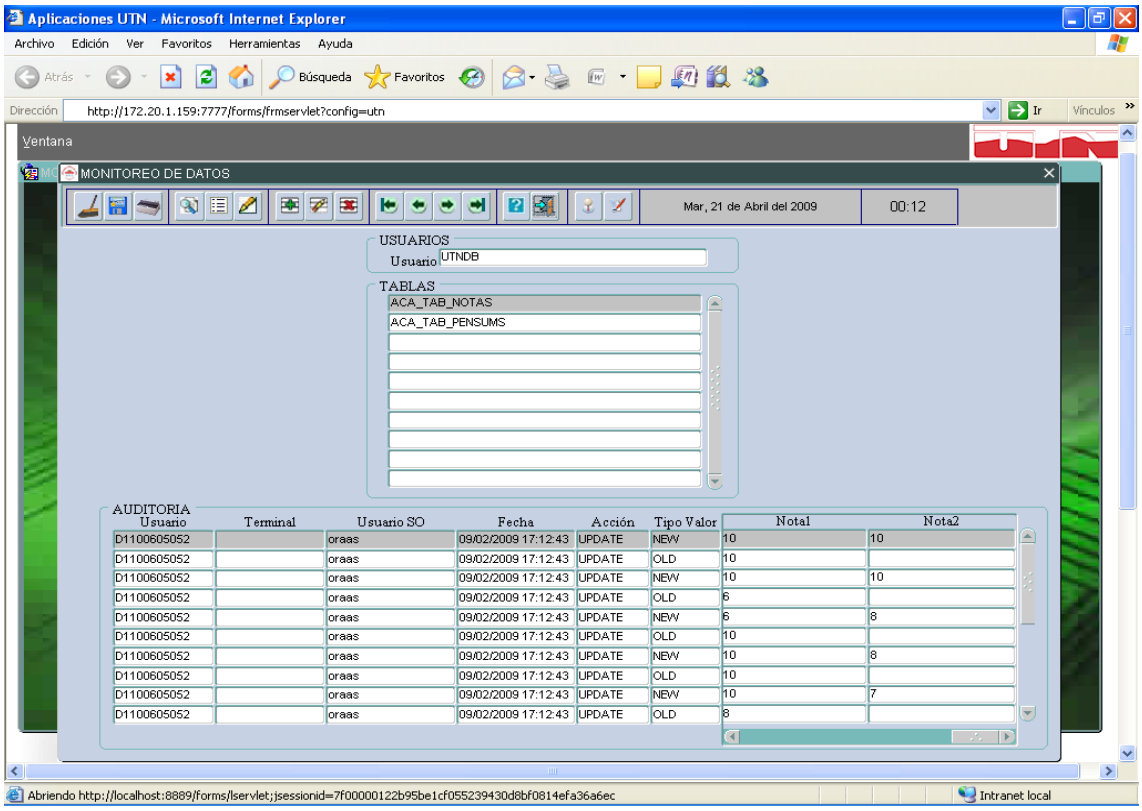


Ilustración 0-42 Monitoreo de Datos 5

En otro ejemplo se ha cambiado la tabla escogida y por lo tanto se cambian las columnas y datos mostrados, ver figura 43 y 44.

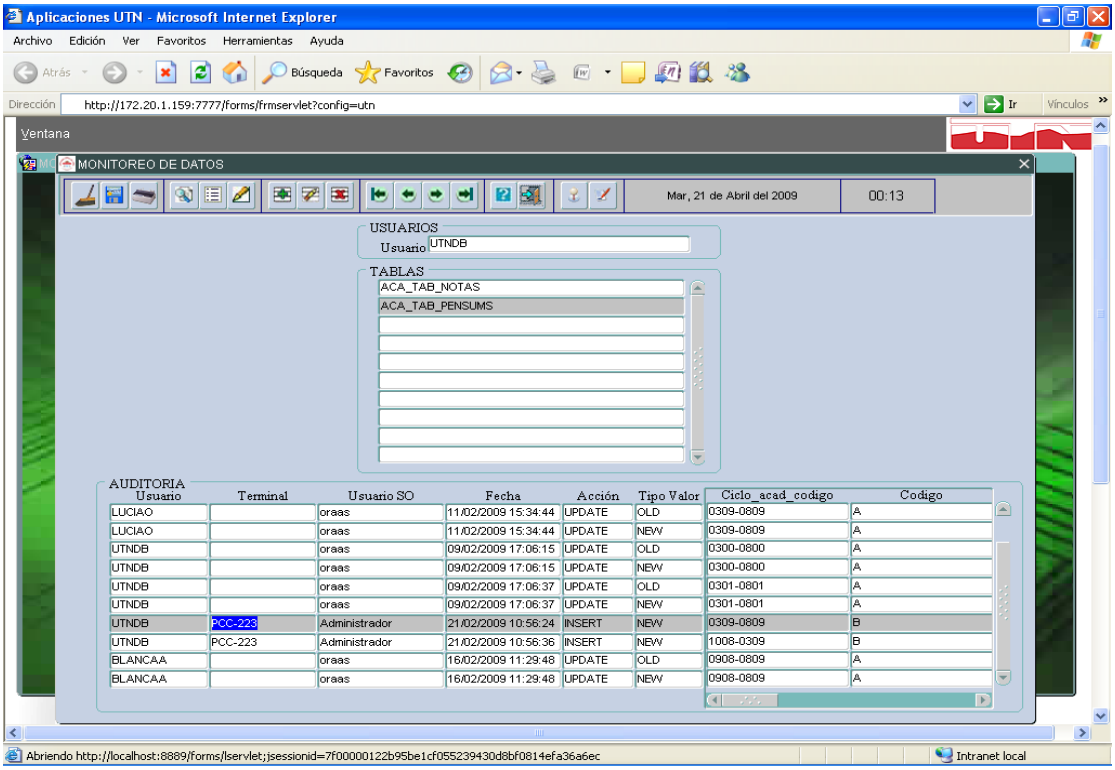


Ilustración 0-43 Monitoreo de Datos 6

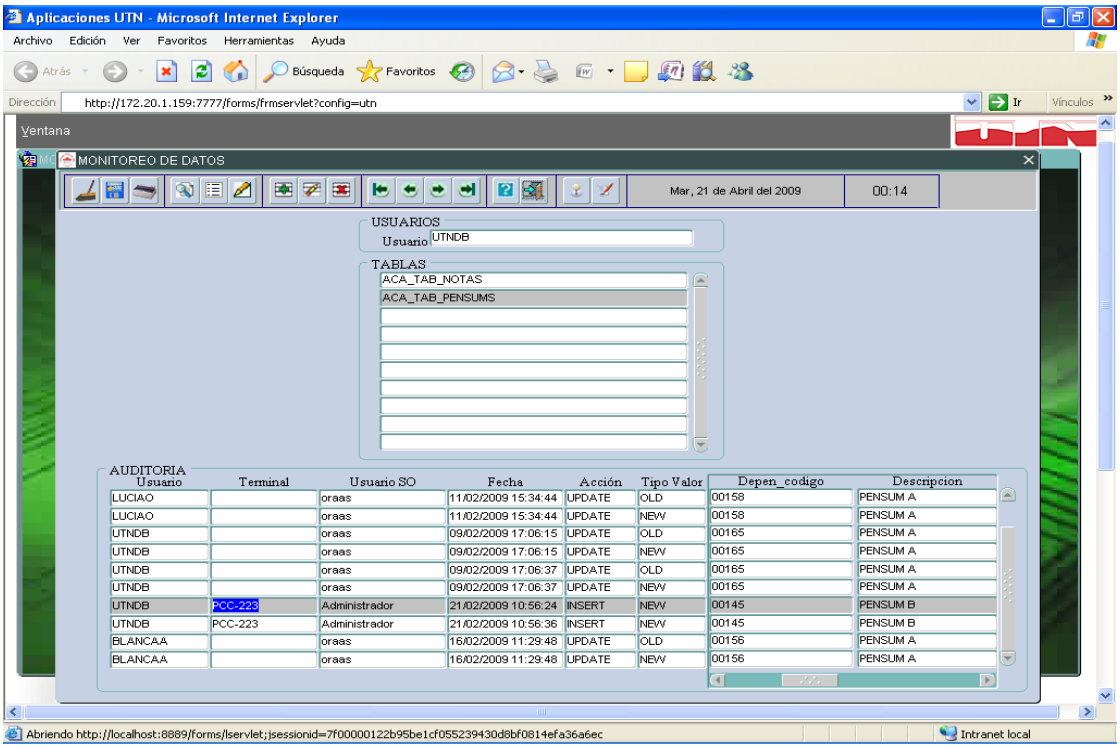


Ilustración 0-44 Monitoreo de Datos 7

E.2.3 REPORTES

El menú de reportes de auditoría comprende la auditoría de Sesiones, Acciones, Objetos y Datos. Ver figura 45.

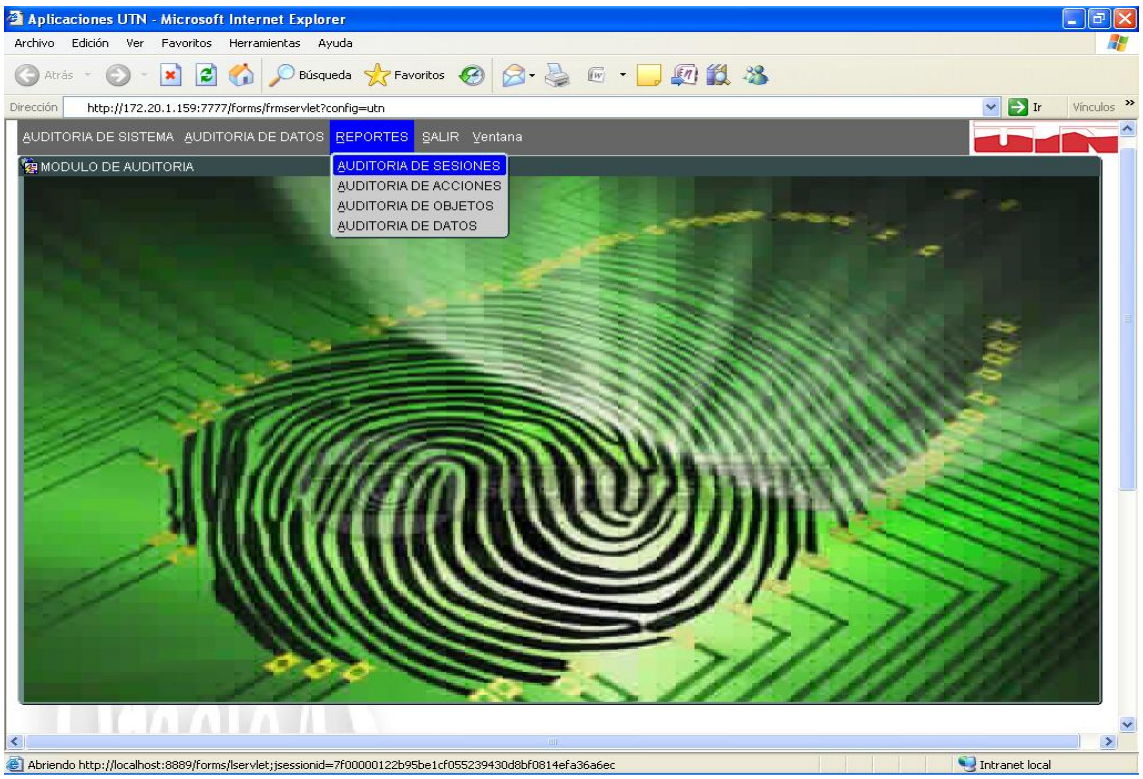


Ilustración 0-45 Menú de Reportes

E.2.4 SALIR DE LA APLICACIÓN

Para salir de la aplicación solamente tenemos que hacer click en la opción salir que se encuentra junto a la opción de reportes.