

Diseño e implementación de la base de datos para una aplicación de control y gestión de riesgos empresariales.

Gonzalo Pose Somoza

Grado de Ingeniería Informática
Bases de datos

Jordi Ferrer Duran

Josep Cobarsí Morales

Junio 2023



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

B) GNU Free Documentation License (GNU FDL)

Copyright © 2023 Gonzalo Pose Somoza.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

C) Copyright

© (el autor/a)

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Diseño e implementación de la base de datos para una aplicación de control y gestión de riesgos empresariales</i>
Nombre del autor:	<i>Gonzalo Pose Somoza</i>
Nombre del consultor/a:	<i>Jordi Ferrer Duran</i>
Nombre del PRA:	<i>Josep Cobarsí Morales</i>
Fecha de entrega (mm/aaaa):	<i>06/2023</i>
Titulación:	<i>Grado de Ingeniería Informática</i>
Área del Trabajo Final:	<i>Bases de datos</i>
Idioma del trabajo:	<i>Español</i>
Palabras clave	Oracle, Riesgos, Warehouse
Resumen del Trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.</i>	
<p>El presente Trabajo de Fin de Grado (TFG), tiene como finalidad el diseño e implementación de una base de datos para la gestión de riesgos empresariales para una empresa en el ámbito financiero, aunque se tendrá en cuenta de que dicha base de datos pueda ser escalable a otro tipo de empresa e independientemente del volumen de datos.</p> <p>La base de datos deberá permitir almacenar los datos necesarios para la identificación, ejecución y resolución de riesgos empresariales, así mismo, se dispondrá de un repositorio estadístico para una consulta rápida y eficaz de los registros.</p> <p>Durante la realización del TFG se ha usado la metodología que se ha explicado en la asignatura "Diseño de bases de datos" que consiste en el análisis de requisitos, la realización del diseño conceptual, lógico y físico de la base de datos, y por último la implementación, optimización y corrección de errores.</p> <p>El producto final se ha probado mediante la carga de datos aleatorios junto con la ejecución de consultas para comprobar la exactitud de los datos del repositorio estadístico.</p> <p>Entre los archivos de entrega del trabajo de fin de grado se encuentran la memoria, los anexos, scripts de implementación de base de datos y script de comprobación de datos.</p>	

Abstract (in English, 250 words or less):

The purpose of this Final Degree Project (TFG) is the design and implementation of a database for the management of business risks for a company in the financial field, although it will be taken into account that this database can be scalable to other types of company and regardless of the volume of data.

This database must be able to store the necessary data for the identification, execution and resolution of business risks, as well as a statistical repository for quick and efficient consultation of the records.

During the TFG, the methodology explained in the subject "Database design" has been used, which consists of the analysis of requirements, the conceptual, logical and physical design of the database, and finally the implementation, optimisation and correction of errors.

The final product has been tested by loading random data together with the execution of queries to check the accuracy of the data in the statistical repository.

The deliverable files of the final thesis include the report, annexes, database implementation scripts and data checking scripts.

Índice

1. Introducción.....	1
1.1. Contexto y justificación del Trabajo	1
1.2. Objetivos del Trabajo.....	1
1.3. Enfoque y método seguido.....	2
1.4. Planificación del Trabajo	2
1.5. Análisis de riesgos y planes de contingencia	3
2. Análisis y preparación de entorno	4
3. Análisis y diseño.....	5
3.1. Descripción inicial.....	5
3.2. Identificación de requisitos	5
4. Casos de uso	7
5. Diseño conceptual.....	8
5.1. Diseño conceptual de la aplicación	9
5.2. Diseño conceptual del repositorio estadístico	16
6. Diseño lógico.....	20
6.1. Diseño lógico de la aplicación	21
6.2. Diseño lógico para el repositorio estadístico	23
7. Diseño físico.....	24
7.1. Base de datos	24
7.2. Tablespace	24
7.3. Tablas.....	25
7.4. Índices	34
7.5. Secuencias.....	34
7.6. Triggers	35
8. Diseño de procedimientos almacenados.....	41
9. Implementación	54
9.1. Creación de usuarios.....	54
9.2. Creación de tablespaces	54
9.3. Creación de tablas.....	55
9.4. Adición de claves foráneas y de campos no nulos.....	55
9.5. Creación de índices de claves foráneas.....	55
9.6. Creación de secuencias	56
9.7. Creación de triggers	56
9.8. Creación de procedimientos almacenados.....	57
9.9. Borrado de base de datos.	57
10. Pruebas	58
10.1. Creación de plan de pruebas.	58
10.2. Ejecución pruebas.	58
11. Retrospectivas de entregas.....	68
11.1. PEC 1	68
11.2. PEC 2	68
11.3. PEC 3.....	69
11.4. Entrega final	69
12. Conclusiones.....	70
13. Glosario	72

14. Bibliografía	73
15. Anexos	74

Lista de figuras

Ilustración 1. Caso de uso 1. Persona de control interno.	7
Ilustración 2. Caso de uso 2. Persona de auditoría interna.	7
Ilustración 3. Caso de uso 3. Persona de auditoría externa.	8
Ilustración 4. Caso de uso 4. Persona usuaria.	8
Ilustración 5. Caso de uso 5. Persona de rol administrador.	8
Ilustración 6. Diagrama entidad relación aplicación.	9
Ilustración 7. Diagrama entidad relación del repositorio estadístico.	17
Ilustración 8. Usuarios.	54
Ilustración 9. Tablespaces de la aplicación.	54
Ilustración 10. Tablas de la aplicación.	55
Ilustración 11. Algunos de los índices de la aplicación.	56
Ilustración 12. Secuencias.	56
Ilustración 13. Triggers.	57
Ilustración 14. Algunos de los paquetes y procedimientos.	57
Ilustración 15. Script de ejemplo para pruebas de la tabla de CATEGORIA.	59
Ilustración 16. Resultado de la tabla de CATEGORIA después de la prueba.	59
Ilustración 17. Resultado de la tabla de LOGGER después de la prueba.	59

1. Introducción

1.1. Contexto y justificación del Trabajo

Actualmente, la empresa lleva un control y análisis de vulnerabilidades, las cuales son definidas como riesgos.

El equipo de control interno de la empresa establece una serie de controles que deben ser ejecutados para los departamentos correspondientes. Los resultados de las evaluaciones de dichos controles pueden dar lugar a distintos tipos de riesgos, los cuales pueden tener diferentes categorías y criticidad de acuerdo con el control ejecutado.

La empresa realiza el control y la gestión de riesgos de forma manual, lo cual no es óptimo, ya que es difícil llevar una trazabilidad, así como también llevar un registro de todos los datos que se evalúan y de los resultados, además, la inversión de tiempo y trabajo en las campañas de evaluación de controles es muy elevada.

Por dichos motivos, la directiva de la empresa se plantea informatizar el proceso, bien realizando un desarrollo propio de un sistema de gestión de riesgos, o la compra de un software de soluciones ERM (Enterprise Risk Management).

1.2. Objetivos del Trabajo

Con la implementación de un sistema propio de gestión de riesgos, se quiere tener informatizado y centralizado todo el trabajo de las evaluaciones para la identificación de vulnerabilidades, junto con los controles, personas, departamentos implicados, resultados y las acciones mitigadoras de aquellos riesgos encontrados.

Con el hecho de tener todo informatizado en un sistema propio, se logra una mayor escalabilidad y personalización a la hora de realizar las definiciones de los datos.

Por lo tanto, lo que se quiere realizar es una base de datos con una aplicación, en la que se pueda gestionar los diferentes datos para la evaluación e identificación de riesgos. Permitiendo así, tener configuradas las entidades involucradas en dicho proceso, como departamentos, personas, categorías, criticidad, así como también los controles, el proceso de ejecución de dichos controles y los resultados junto con los riesgos y planes de mitigación asociados, además de guardar un histórico y tener trazabilidad de todo lo ejecutado a través de logs.

1.3. Enfoque y método seguido

Ante el desarrollo de cualquier proyecto, es necesario saber que metodología usar para lograr los objetivos. Existen diferentes tipos de métodos de desarrollo de software¹, sin embargo, los más comunes y utilizados son los siguientes:

- Ciclo de vida clásico o en cascada.
Ideal para proyectos en los que se tiene claro que se quiere hacer y cómo se hará, además de una duración. Una característica de este tipo de proyectos es que existe poca tolerancia al cambio o que dichos cambios sean sencillos de aplicar.
- Ciclo de vida iterativo e incremental.
Ideal para proyectos en los que se tiene claro el objetivo, pero no se conoce del todo la forma de llegar a él. Es un método en el que se permiten cambiar ideas iniciales a medidas de que el proyecto va avanzando y se van encontrando nuevas o mejores ideas, identificadas en los ciclos de retroalimentación.
- Desarrollo learn y ágil.
Ideal para proyectos de gran envergadura, en las que no se conoce muy bien el problema ni tampoco la solución.

Dada la naturaleza del TFG, y leyendo el enunciado, se opta por usar el ciclo de vida en cascada, ya que, con dicha lectura se conoce de forma clara el problema y cuál es el objetivo deseado. Además, realizando una buena tarea en las fases de análisis de requisitos, diseño e implementación se minimiza ampliamente el riesgo de encontrar un error en las fases finales del proyecto.

Por otro lado, este ciclo de vida es ideal, ya que la creación de una base de datos, de manera correcta, está formada por una serie de pasos de forma secuencial, en la que es necesario haber finalizado un punto antes de comenzar con el siguiente.

Las fases identificadas en el proyecto, y que por lo tanto tienen la siguiente forma de cascada son las siguientes:

1. Análisis de requisitos.
2. Diseño de casos de uso, conceptual y lógico de la base de datos.
3. Implementación de la base de datos, que consta del diseño físico y del repositorio estadístico.
4. Pruebas.

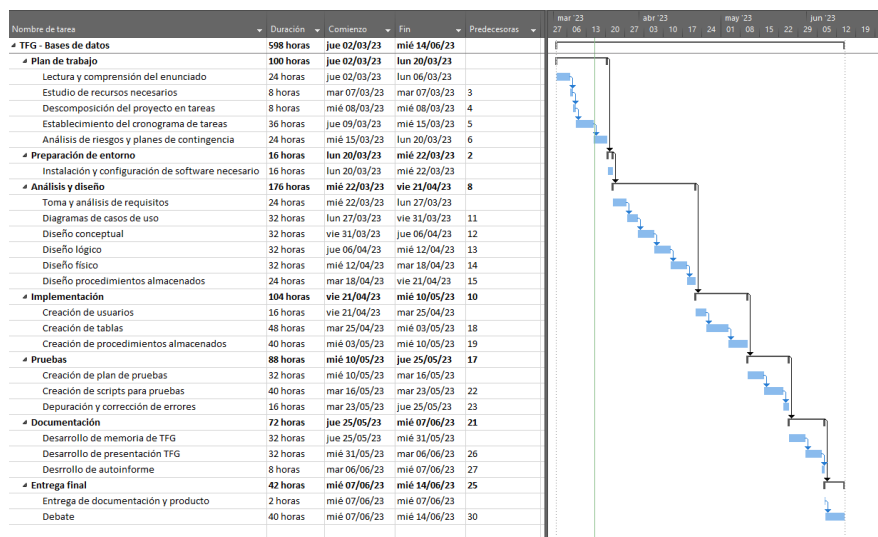
1.4. Planificación del Trabajo

¹ Material de Ingeniería del Software. Jordi Pradel Miquel, Jose Raya Martos UOC 3º Edición (febrero 2016).

Las tareas principales tareas que se han identificado son las siguientes:

- Lectura y comprensión del enunciado: Para obtener el contexto de la situación actual y lo que se quiere lograr.
- Toma y análisis de requisitos: A partir de la lectura y análisis del enunciado, tener claros e identificados todos los requisitos para tener el producto deseado.
- Estudio de recursos necesarios: Identificación de los distintos tipos de software que podrían ser necesarios para la implementación del producto.
- Preparación del entorno de trabajo: Instalación y configuración del software identificado y deseado para la elaboración del producto.
- Descomposición del proyecto en tareas: En las consistiría en crear las tareas para la creación de la base de datos, tablas, usuarios, procedimientos y todo lo necesario para el desarrollo del producto.
- Establecimiento del cronograma de tareas: Creación del diagrama de Gantt con las tareas y el tiempo estimado para cada una de ellas.
- Análisis de riesgos y planes de contingencia: En las que se identifican los posibles riesgos que puedan ocasionar alteraciones en las fechas de entrega, junto con dichos riesgos también se define el plan de contingencia para cada uno de ellos.

A continuación, se representa la descomposición de tareas en el diagrama de Gantt.



1.5. Análisis de riesgos y planes de contingencia

Se debe tener en cuenta que, en todo proyecto, siempre pueden surgir imprevistos que afecten a los plazos de entrega y/o realización del proyecto. Por ese motivo es importante realizar un análisis de los posibles riesgos que se identifiquen, valorando la probabilidad y el impacto de éstos².

² Material de Gestión de Proyectos. José Ramón Rodríguez, Pere Mariné Jové, UOC 3º Edición (septiembre 2018)

Una vez identificados los riesgos, es importante que, junto a cada uno, se identifiquen las medidas o planes de contingencia para intentar lograr que el impacto del riesgo producido sea nulo o por lo menos ampliamente minimizado en la medida de lo posible.

Los riesgos más significativos que se han identificado están resumidos en la siguiente tabla:

Código	Nombre	Descripción	Probabilidad	Impacto
01	Perdida del entorno de trabajo	Problemas en el hardware o software que se usa para el desarrollo	Baja	Alto
02	Perdida de datos	Perdida de datos del proyecto de forma repentina o accidental	Baja	Alto
03	Perdida de conectividad	Problemas con la conexión a internet, campus o distintas fuentes de información	Baja	Medio
04	Excesiva carga laboral	Aumento de la carga de trabajo externa a la universidad	Media	Medio
05	Enfermedad	Enfermedad mía, de mi pareja o de algún integrante de la familia	Media	Medio
06	Casos extremos	Casos repentinos y/o no predecibles	Baja	Alto

Los planes de contingencia para los códigos de riesgos identificados se definen a continuación:

Código	Acción	Tipo	Riesgo después de aplicación acción
01R1	Tener otro ordenador con el entorno replicado	Correctora	Bajo
02R1	El almacenamiento de datos está en una carpeta sincronizada constantemente con Google Drive	Mitigadora	Bajo
03R1	Usar la conectividad de algún familiar o vecino.	Mitigadora	Bajo
03R2	Usar la conectividad del teléfono móvil	Mitigadora	Bajo
04R1	Corrección de fechas en la planificación y aumentar las horas de dedicación	Correctora	Medio
05R1	Después de la enfermedad, aumentar las horas de dedicación en las tareas	Correctora	Medio
06R1	Buscar una solución negociando entre cliente y equipo	Correctora	Alto

2. Análisis y preparación de entorno

A continuación, se muestran los recursos utilizados para la elaboración de la base de datos, así como también la documentación relacionada con el TFG.

Hardware

Ordenador: Portátil Inspiron 16.

Procesador: Intel 11^o Generación. I7 2.3GHz.

RAM: 16GB.

Disco duro: SSD 1 TB.

Sistema operativo: Windows 10 PRO.

Software

Para la elaboración de la documentación se realiza con el Microsoft Word de la suit de Microsoft Office 635.

Los diagramas de Gantt fueron realizados con Microsoft Project Professional 2019.

Para el desarrollo de casos de uso y de diagramas UML, se han analizado diferentes programas, como GitMind, Visual Paradigm, MagicDraw y Draw.io. Tras una revisión y comparación entre ellos, se ha decidido usar Draw.io ya que es un programa conocido, con el que se ha trabajado anteriormente, gratuito y sin necesidad de registro online.

Como gestor de bases de datos se usa el recomendado por la asignatura, Oracle, específicamente usaremos “Oracle Database Express Edition 21c”, y el entorno de desarrollo será el “SQL Developer” en su versión 22.2.1. La instalación del gestor de bases de datos y del entorno de desarrollo se encuentra explicada en el Anexo I – Entorno desarrollo Oracle³.

3. Análisis y diseño

3.1. Descripción inicial

El objetivo del trabajo es la realización de una base de datos para dar respuesta a una necesidad de una empresa del sector financiero, la cual quiere mejorar la gestión de riesgos para dar una mejor reacción ante cualquier riesgo que se pueda detectar.

Dicha base de datos deberá ser capaz de permitir y almacenar todos los datos necesarios para la gestión de riesgos, desde el propio riesgo en sí, como las categorías, importancia, criticidad (entre otras características que conforma el riesgo), hasta los controles, campañas, departamentos y equipos que participan en la identificación y mitigación de riesgos.

Dicha base de datos, además de almacenar todos los datos mencionados anteriormente, debe permitir la generación de consultas, y consultas estadísticas, para el cual se crearán unas tablas y procedimientos almacenados propios de este apartado para que dichas consultas tengan una respuesta rápida y en tiempo constante igual a 1.

3.2. Identificación de requisitos

Tras analizar el enunciado, se han identificado una serie de requisitos que estaban perfectamente definidos, sin embargo, se hicieron consultas al tutor para aclarar ciertos puntos en los que no se estaba totalmente claro.

³ <https://www.youtube.com/watch?v=MZ4rJNYLTIY> visitado el 20 de marzo de 2023.

De acuerdo con el enunciado, los siguientes puntos son los requisitos en los que el sistema debe:

- Poder gestionar categorías (alta, baja y modificación) y en las que inicialmente deben existir: ciberseguridad, seguridad física de instalaciones, aplicaciones informáticas y finanzas.
- Tener identificadas la importancia para cada categoría siendo 1 (muy importante) a 3 (poco importante), también puede considerarse como datos maestros por si se quieren añadir o cambiar los valores.
- Tener identificada la criticidad de los riesgos, siendo 1 (riesgo crítico) a 3 (riesgo bajo), también puede considerarse como datos maestros por si se quieren añadir o cambiar los valores.
- Calcular el impacto del riesgo, el cual se obtiene a partir del producto de la criticidad del riesgo por la importancia de la categoría.
- Gestionar orígenes de detección, en los que se contemplan auditoría interna, externa o autoevaluación, también puede considerarse como datos maestros por si se quieren añadir nuevos orígenes en un futuro.
- Poder almacenar los riesgos detectados con sus datos (categoría, descripción, origen, criticidad, fecha de creación, persona responsable y todos los campos necesarios).
- Gestionar (alta, baja y modificación) de registros de control independientemente del equipo de origen que realizara la auditoria.
- Gestionar (alta, baja y modificación) departamentos.
- Gestionar (alta, baja y modificación) campañas, junto con sus datos correspondientes, fechas, personas implicadas, resultados, equipo de origen, entre otros campos.
- Gestionar (alta, baja y modificación) campañas, junto con sus datos correspondientes y los controles asociados.
- Almacenar planes de acción que relacionara el riesgo con las acciones para la mitigación o eliminación.
- Actualizar estado de los riesgos de acuerdo con las resoluciones de las acciones.
- Registrar las acciones a través de logs.
- Se creará un repositorio estadístico que tendrá que responder a las cuestiones siguientes:
- Actualizar el impacto del riesgo cuando la criticidad o la importancia cambien.
- Gestión de personas (alta, baja y modificación)
- Gestión de empresas para auditorías externas.
- Además, incluir todos los puntos solicitados en el enunciado:
 - En el momento de ejecutar la consulta, porcentaje de riesgos no corregidos o mitigados con un impacto menor a 4.
 - Dado un año en concreto, número total de riesgos con un impacto de 1.
 - En el último año finalizado, número de riesgos que actualmente aún están en estado abierto.
 - Con todos los datos disponibles en la BD, departamento con un número mayor de riesgos detectados por auditorías externas.

- En el año actual, número total de acciones en estado de en curso.
- Dado un año en concreto, diferencia entre el número de riesgos de impacto 1 detectados por auditoría interna y externa.
- En el momento de ejecutar la consulta, persona de la empresa que tiene más acciones abiertas (en estado de en curso o de definida) asignadas.
- Número medio de muestreos realizados por año sin considerar el año actual.
- Teniendo en cuenta el año en curso y el anterior, número de riesgos de ciberseguridad detectados por autoevaluación de los departamentos.
- En el último año, coste medio de todas las auditorías externas realizadas.
- Teniendo en cuenta todos los datos disponibles, año con un número mayor de acciones que al final del año en cuestión estaban en estado de implementada con el riesgo mitigado.
- Teniendo en cuenta sólo el último año finalizado, Top3 de acciones definidas teniendo en cuenta el tiempo que estuvieron abiertas (desde su creación hasta que pasen a uno de los estados de implementada). Se debe indicar el nombre de las 3 acciones que estuvieron más tiempo abiertas.
- En el momento de ejecutar la consulta, número de riesgos de categoría 1 (en cualquier estado).
- Dado un año en concreto, porcentaje de acciones descartadas.

4. Casos de uso

Los actores identificados se resumen en personas que pueden formar parte del equipo de autoevaluación, de control interno, de equipo externo, además se definen personas de tipo administrador y usuarios.

Para las personas de control interno o autoevaluación definimos el siguiente caso de uso:

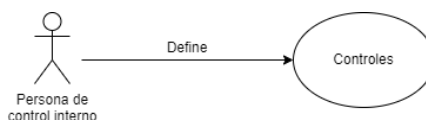


Ilustración 1. Caso de uso 1. Persona de control interno.

Para las personas de auditoría interna definimos el siguiente caso de uso:

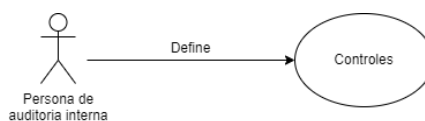


Ilustración 2. Caso de uso 2. Persona de auditoría interna.

En cuanto a las personas pertenecientes al equipo de auditoría externa:

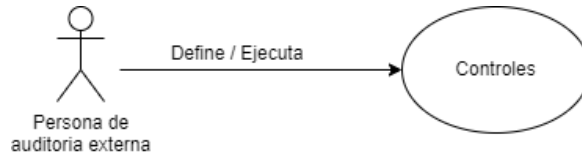


Ilustración 3. Caso de uso 3. Persona de auditoría externa.

Para las personas usuarias:

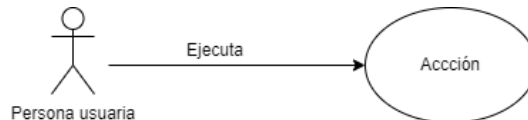


Ilustración 4. Caso de uso 4. Persona usuaria.

Por otro lado, tenemos usuarios administradores, cuyo caso de uso se corresponde con el siguiente:

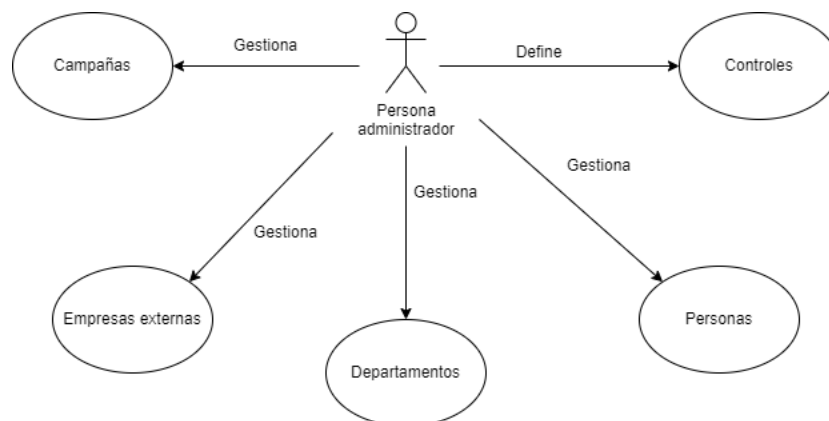


Ilustración 5. Caso de uso 5. Persona de rol administrador.

5. Diseño conceptual

Para el análisis y generación del diseño conceptual tanto de la aplicación como del repositorio estadístico, se ha utilizado la estrategia de diseño descendente⁴, en la que se empieza el proceso con un esquema de entidades de alto nivel y se van aplicando sucesivamente refinamientos sobre dichas entidades hasta tener el diseño completo.

Además, se realiza el modelo entidad-relación o modelo ER, que es un modelo conceptual de alto nivel independientemente de la tecnología a usar, es

⁴ Material de Bases De Datos. Jordi Casas Roma, Josep Cuartero Olivera. UOC 5º Edición (febrero 2020)

un modelo simple y de fácil uso, ya que solo se muestran las entidades, atributos y las relaciones entre dichas entidades.

5.1. Diseño conceptual de la aplicación

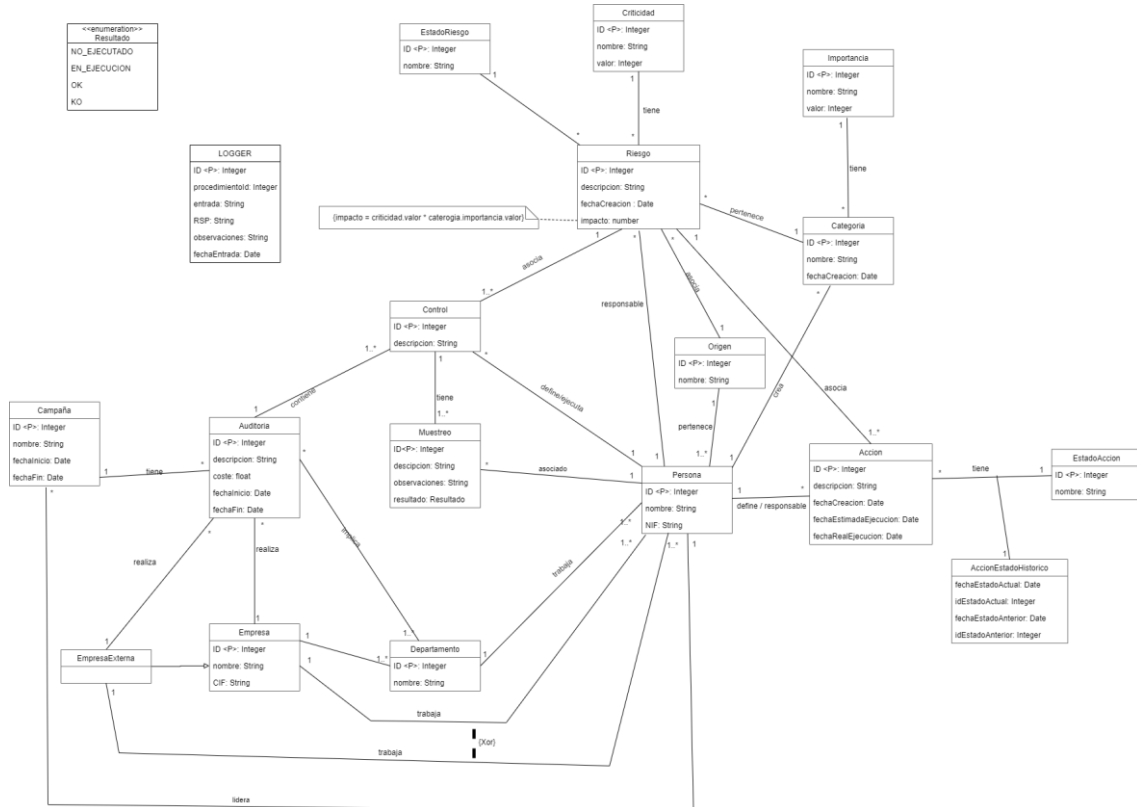


Ilustración 6. Diagrama entidad relación aplicación.

Se procede a explicar el diseño conceptual por entidades.

Entidad: Campaña		
Descripción: Entidad que representa los datos para las campañas en las que se realizan las auditorías.		
Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Texto que representa el nombre de la campaña.
fechalnicio	Date	Fecha de inicio.
fechaFin	Date	Fecha de fin.
Relaciones		
Entidad	Cardinalidad	Descripción
Auditoria	1_*	Indica que una campaña puede tener 0 o muchas auditorías.
Persona	*_1	Representa que una campaña está liderada por una única persona.

--	--	--

Entidad: Auditoría		
Descripción: Entidad que representa los datos para las auditorías.		
Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
descripcion	String	Breve descripción de la auditoría.
coste	Float	Indica el coste de la auditoría.
fechaInicio	Date	Indica la fecha de inicio de la auditoría.
fechaFin	Date	Indica la fecha de fin de la auditoría
Relaciones		
Entidad	Cardinalidad	Descripción
Campaña	*_1	Indica que una auditoría debe pertenecer a una única campaña.
Empresa	*_1	Representa que una auditoría está realizada por una única empresa.
Control	1_1..*	Representa que una auditoría debe tener como mínimo un control.
Departamento	*_1..*	Indica que una auditoría debe por lo menos incluir a un departamento.
EmpresaExterna	*_1	Representa que una auditoría está realizada por una única empresa externa.

Entidad: Empresa		
Descripción: Entidad que representa los datos para las empresas.		
Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
CIF	String	Código que identifica de forma fiscal a la empresa.
nombre	String	Nombre de la empresa.
Relaciones		
Entidad	Cardinalidad	Descripción
Auditoria	1_*	Indica que la empresa puede o no realizar auditorías.
Persona	1_1..*	Representa que en la empresa trabaja al menos una persona.
Departamento	1_1..*	Indica que una empresa debe tener al menos un departamento.

Entidad: Empresa > EmpresaExterna		
Descripción: Entidad que representa los datos para las empresas externas. Es una entidad que hereda de Empresa.		

Relaciones		
Entidad	Cardinalidad	Descripción
Auditoria	1_*	Indica que la empresa puede o no realizar auditoría.
Persona	1_1..*	Representa que en la empresa trabaja al menos una persona.

Entidad: Departamento

Descripción: Entidad que representa los departamentos de una empresa, en las que se encuentran implicadas las personas y las auditorías.

Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Texto que representa el nombre del departamento.

Relaciones

Entidad	Cardinalidad	Descripción
Auditoria	1..*_*	Indica que un departamento puede o no estar implicado en alguna auditoría.
Persona	1_1..*	Representa que un departamento debe tener por lo menos una persona.
Empresa	1..*_1	Indica que un departamento debe pertenecer a una única empresa.

Entidad: Muestreo

Descripción: Entidad que representa los datos de las muestras asociadas a los controles y a las personas.

Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
descripcion	String	Texto que representa la descripción del muestreo.
observaciones	String	Breve descripción de datos adicionales al muestreo.
resultado	Resultado	Valor del enumerado de resultado.

Relaciones

Entidad	Cardinalidad	Descripción
Control	1..*_1	Indica que un muestreo debe pertenecer a un único control.
Persona	*_1	Indica que un muestreo está asociado a una única persona.

Entidad: Control

Descripción: Entidad que representa los datos de los controles que están asociados a las auditorías, y que además se asocian a los riesgos y personas que los definen o ejecutan.

Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.

descripcion	String	Texto que representa el control a realizar.
Relaciones		
Entidad	Cardinalidad	Descripción
Auditoria	1..*_1	Indica que el control está asociado a una única auditoría.
Muestreo	1_1..*	Representa que el control debe tener como mínimo un muestreo.
Riesgo	1..*_1	Indica que el control tiene asociado un único riesgo.
Persona	*_1	Representa que un control está definido o ejecutado por una única persona.

Entidad: Persona		
Descripción: Entidad que representa los datos de una persona, la cual puede estar relacionada con distintos tipos de entidades, como departamentos, muestras, riesgos o equipos de origen por mencionar algunas de ellas. Junto con la entidad riesgos, ésta es una de las entidades de más peso de la aplicación.		
Atributo	Tipo	Descripción
Id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Texto que representa el nombre de la persona.
NIF	String	Código que representa de forma fiscal a la persona.
Relaciones		
Entidad	Cardinalidad	Descripción
Campaña	1_*	Indica que la persona puede o no liderar una campaña.
Empresa	1..*_1	Representa que una persona solo puede trabajar en una empresa. (Relación XOR con EmpresaExterna)
EmpresaExterna	1..*_1	Representa que una persona solo puede trabajar en una empresaExterna. (Relación XOR con Empresa)
Muestreo	1_*	Indica que una persona puede o no tener asociados muestreos.
Control	1_*	Indica que una persona puede o no definir o ejecutar controles.
Riesgo	1_*	Representa que una persona puede o no ser responsable de riesgos.
Origen	1..*_1	Indica que una persona debe pertenecer a un único equipo de origen.
Accion	1_*	Representa que una persona puede o no definir acciones, o ser responsable de ellas.
Categoria	1_*	Indica que una persona puede crear o no categorías.
Departamento	1..*_1	Indica que una persona pertenece a un único departamento.

Entidad: Acción		
Descripción: Entidad que representa los datos de las acciones definidas que están asociadas a los riesgos.		
Atributo	Tipo	Descripción

id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
descripcion	String	Texto que describe la acción a realizar.
fechaCreacion	Date	Fecha de creación de la acción
fechaEstimadaEjecucion	Date	Fecha en la que se estima que la acción deberá ser ejecutada.
fechaRealEjecucion	Date	Fecha real en la que ejecuta la acción.

Relaciones

Entidad	Cardinalidad	Descripción
EstadoAccion	*_1	Indica que una acción debe tener un estado (Dependencia con AccionEstadoHistorico).
AccionEstadoHistorico	*_1	Representa que una acción solo tiene un histórico en una fecha determinada.
Riesgo	1..*_1	Representa que una acción siempre pertenece a un único riesgo.
Persona	*_1	Representa que una acción está definida por una única persona y que una acción solo tiene un responsable.

Entidad: EstadoAccion

Descripción: Representa los datos de los estados de las acciones.

Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Nombre del estado.

Relaciones

Entidad	Cardinalidad	Descripción
Acción	1_*	Indica que un estado puede o no estar en las acciones.
AccionEstadoHistorico	1_1	Indica que un estado tiene un histórico en una fecha determinada para una determinada acción.

Entidad: AccionEstadoHistorico

Descripción: Representa los datos históricos de los estados de las acciones.

Atributo	Tipo	Descripción
fechaEstadoActual	Date	Fecha de inserción del estado
fechaEstadoAnterior	Date	Fecha del estado anterior
idEstadoAnterior	Integer	Id del estado anterior

Relaciones

Entidad	Cardinalidad	Descripción
Acción	1_*	Indica que el estado de una acción, para una fecha determinada

EstadoAccion	1_1	Indica que el estado de una acción para una fecha determinada.

Entidad: Origen		
Descripción: Representa el equipo origen de la identificación de riesgos (interno, externo o autoevaluación) al que pertenecen las personas.		
Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Texto que representa el nombre del equipo de identificación.
Relaciones		
Entidad	Cardinalidad	Descripción
Persona	1_1..*	Indica que al menos debe tener una persona.
Riesgo	1_*	Representa que el origen puede o no tener riesgos asociados.

Entidad: Riesgo		
Descripción: Entidad que representa los datos de los riesgos. En la que se definen su categoría, criticidad, evaluación de impacto y el equipo de identificación del riesgo, así como también los controles y acciones pertenecientes a cada uno de los riesgos.		
Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
descripcion	String	Texto que representa la descripción del riesgo.
fechaCreacion	Date	Fecha en la que se crea el riesgo.
impacto	Number	Número que identifica el impacto del riesgo que es el valor calculado de la criticidad por la importancia.
Relaciones		
Entidad	Cardinalidad	Descripción
Control	1_1..*	Representa que un riesgo debe tener asociado al menos un control.
Persona	*_1	Indica que cada riesgo tiene un único responsable.
Origen	*_1	Representa que cada riesgo tiene un único equipo de origen de identificación.
Accion	1_1..*	Indica que por cada riesgo al menos debe existir una acción.
Categoria	*_1	Representa que cada riesgo tiene una única categoría.
Criticidad	*_1	Indica que todo riesgo debe tener una criticidad.
EstadoRiesgo	*_1	Indica que todo riesgo debe tener un único estado.

Entidad: Categoría		
Descripción: Entidad que representa los datos de las categorías.		

Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Texto que identifica el nombre de la categoría.
fechaCreacion	Date	Fecha de creación de la categoría.

Relaciones

Entidad	Cardinalidad	Descripción
Riesgo	1_*	Indica que la categoría puede tener o no riesgos.
Persona	*_1	Representa que categoría fue creada por una única persona.
Importancia	*_1	Indica que categoría tiene una sola importancia.

Entidad: Importancia		
Descripción: Entidad que representa los datos de la importancia de la categoría.		
Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Texto que representa el nombre de la importancia.
valor	Integer	Valor que identifica la importancia. En un inicio se toma 1 como muy importante y 3 como poco importante.

Relaciones

Entidad	Cardinalidad	Descripción
Categoría	1_*	Indica que pueden existir o no categorías con esa importancia.

Entidad: Criticidad		
Descripción: Entidad que representa el nivel de criticidad de los riesgos.		
Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Texto que representa el nombre de la criticidad.
valor	Integer	Valor que identifica la criticidad de los riesgos. En un inicio se toma 1 como riesgo crítico y 3 como riesgo bajo.

Relaciones

Entidad	Cardinalidad	Descripción
Riesgo	1_*	Representa que pueden existir riesgos que tengan o no una criticidad específica.

Entidad: EstadoRiesgo		
Descripción: Entidad que representa los posibles estados para los riesgos.		

Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
nombre	String	Texto que representa el nombre de la criticidad.

Relaciones

Entidad	Cardinalidad	Descripción
Riesgo	1_*	Representa que pueden existir riesgos que tengan o no un estado específico.

Entidad: Logger		
Descripción: Representa los datos de log de las transacciones.		
Atributo	Tipo	Descripción
id	Integer	Clave primaria de la entidad. Número que identifica inequívocamente el registro en la entidad.
procedimientold	Integer	Identificador del procedimiento que escribe en el log.
entrada	String	Parámetros de entrada.
RSP	String	Valor de retorno.
observaciones	String	Texto que representa observaciones de la entrada.
fechaEntrada	Date	Fecha de inserción del registro en la tabla.

5.2. Diseño conceptual del repositorio estadístico

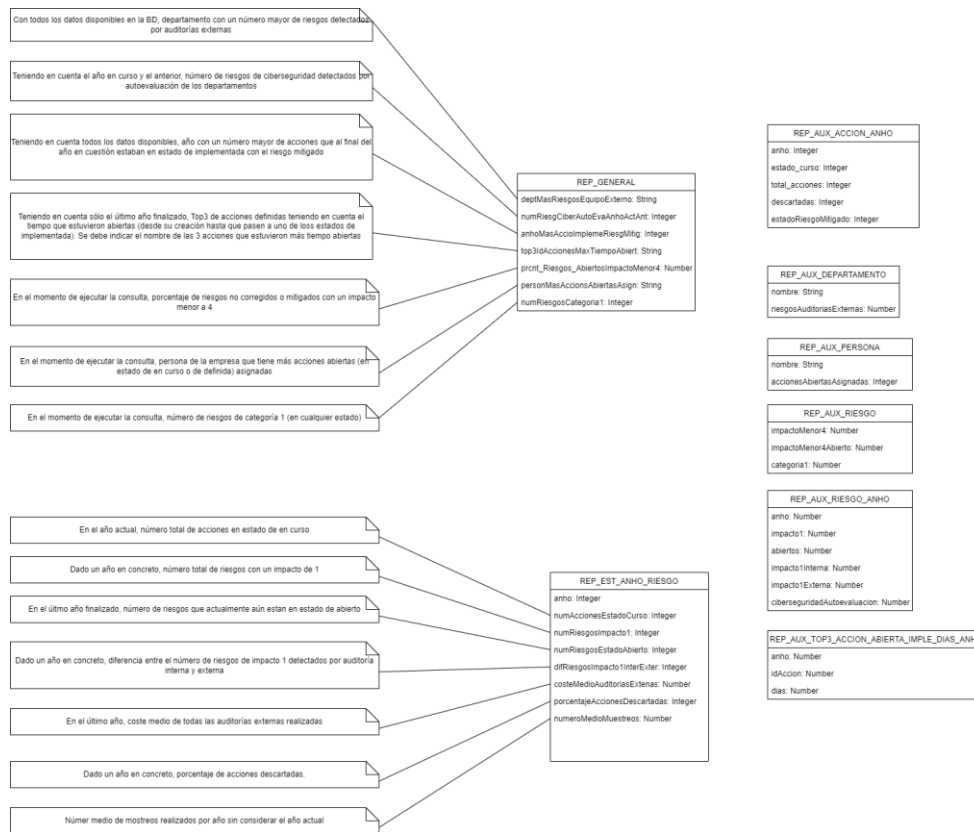


Ilustración 7. Diagrama entidad relación del repositorio estadístico.

El repositorio estadístico se crea para que los usuarios puedan obtener datos en tiempo constante igual a 1, es decir, de forma inmediata, sin tener que hacer cálculos con los datos de todas las tablas.

Para ello se crean unas tablas que contendrán los atributos para cada uno de los puntos solicitados que serán poblados a partir de tablas auxiliares. Los datos de dichas tablas se irán insertando y actualizando a través de unos procedimientos almacenados que se ejecutarán cuando se realicen llamadas de inserción, modificación o eliminación sobre las tablas de la aplicación.

Una de las tablas contendrá los datos de forma general, otra contendrá datos agrupados por año.

Entidad: REP_GENERAL		
Descripción: Entidad del repositorio estadístico que contendrá los datos de forma general.		
Atributo	Tipo	Descripción de requerimiento
depMasRiesgosEquipoExterno	String	Con todos los datos disponibles en la BD, departamento con un número mayor de riesgos detectados por auditorías externas.
numRiesgCiberAutoEvaAnhoActAnt	Integer	Teniendo en cuenta el año en curso y el anterior, número de riesgos de ciberseguridad detectados por autoevaluación de los departamentos.

anhoMasAcciImplmeRiesgMitig	Integer	Teniendo en cuenta todos los datos disponibles, año con un número mayor de acciones que al final del año en cuestión estaban en estado de implementada con el riesgo mitigado.
top3IdAccionesMaxTiempoAbiert	Integer	Teniendo en cuenta sólo el último año finalizado, Top3 de acciones definidas teniendo en cuenta el tiempo que estuvieron abiertas (desde su creación hasta que pasen a uno de los estados de implementada). Se debe indicar el nombre de las 3 acciones que estuvieron más tiempo abiertas
prcnt_Riesgos_AbiertosImpactoMenor4	Integer	En el momento de ejecutar la consulta, porcentaje de riesgos no corregidos o mitigados con un impacto menor a 4.
personMasAccionsAbiertasAsign	String	En el momento de ejecutar la consulta, persona de la empresa que tiene más acciones abiertas (en estado de en curso o de definida) asignadas.
numRiesgosCategoria1	Integer	En el momento de ejecutar la consulta, número de riesgos de categoría 1 (en cualquier estado).

Entidad: REP_EST_AÑO_RIESGO

Descripción: Entidad que representa los datos de forma agrupadas por años.

Atributo	Tipo	Descripción de requerimiento
anho	Integer	Año del cual se representan los datos.
numAccionesEstadoCurso	Integer	En el año actual, número total de acciones en estado de en curso.
numRiesgosImpacto1	Integer	Dado un año en concreto, número total de riesgos con un impacto de 1
numRiesgosEstadoAbierto	Integer	En el último año finalizado, número de riesgos que actualmente aún están en estado de abierto.
difRiesgosImpacto1InterExter	Integer	Dado un año en concreto, diferencia entre el número de riesgos de impacto 1 detectados por auditoría interna y externa.
costeMedioAuditoriasExtenas	Integer	En el último año, coste medio de todas las auditorías externas realizadas.
porcentajeAccionesDescartadas	Integer	Dado un año en concreto, porcentaje de acciones descartadas.
numeroMedioMuestreos	Number	Número medio de muestreos realizados por año sin considerar el año actual.

Entidad: REP_AUX_ACCION_ANHO

Descripción: Contiene datos de acciones agrupadas por año.

Atributo	Tipo	Descripción de requerimiento
anho	Number	Año de acción.
estadoCurso	Number	Número de acciones de estado En curso.
totalAcciones	Number	Número total de acciones
descartadas	Number	Número total de acciones descartadas
estadoRiesgoMitigado	Number	Número total de acciones en estado Implementada con riesgo mitigado

Entidad: REP_AUX_DEPARTAMENTO

Descripción: Contiene datos auxiliares por departamento

Atributo	Tipo	Descripción de requerimiento
nombre	Number	Nombre de departamento.
riesgosAuditoriasExternas	Number	Número de riesgos detectados por auditorías externas.

Entidad: REP_AUX_PERSONA

Descripción: Contiene datos auxiliares por persona

Atributo	Tipo	Descripción de requerimiento
nombre	Number	Nombre de la persona
accionesAbiertasAsignadas	Number	Número de acciones en estado abiertas o en curso que la persona tiene asignadas.

Entidad: REP_AUX_RIESGO

Descripción: Contiene datos auxiliares por riesgo

Atributo	Tipo	Descripción de requerimiento
impactoMenor4	Number	Número de riesgos con impacto menor que 4.
impactoMenor4Abierto	Number	Número de riesgos abiertos con impacto menor que 4.
categoria1	Number	Número de riesgos de categoría 1.

Entidad: REP_AUX_RIESGO_ANHO

Descripción: Contiene datos auxiliares por riesgo y año

Atributo	Tipo	Descripción de requerimiento
anho	Number	Año de riesgo.
impacto1	Number	Número de riesgos impacto igual a 1.
abiertos	Number	Número de riesgos abiertos.
impacto1Interna	Number	Número de riesgos impacto igual a 1 de auditorías internas.
impacto1Externa	Number	Número de riesgos impacto igual a 1 de

		auditorías externas.
ciberseguridadAutoevaluacion	Number	Número de riesgos de ciberseguridad detectados por autoevaluación.

Entidad: REP_AUX_TOP3_ACCION_ABIERTA_IMPLD_DIAS_ANHO		
Descripción: Contiene el top 3 por año, de acciones con el número de días que estuvieron abiertas.		
Atributo	Tipo	Descripción de requerimiento
anho	Number	Año de acción.
idAccion	Number	Identificador de la acción.
dias	Number	Número de días entre el cambio.

6. Diseño lógico

El diseño conceptual realizado representa una parte de lo que sería la base de datos, un esquema simple y de fácil lectura, pero un paso crucial para implementar una base de datos correctamente funcional.

Una vez realizado dicho diseño conceptual, es el momento de transformar dicho diseño en un diseño lógico, es decir, lo que se representaría en la base de datos real. Gracias a este diseño se pueden identificar errores que se hayan obtenido a la hora de realizar el diagrama conceptual. Por otra parte, también es de utilidad para definir el cómo implementar en una base de datos relacional las relaciones entre las distintas entidades, así como también la identificación de claves primarias, alternativas, foráneas y la identificación de campos no nulos.

Para la realización del diseño lógico que se explicara a continuación, se define la entidad, los supuestos que se realizan, y a continuación el diseño lógico, es decir, la tabla correspondiente a dicha entidad, junto con los atributos, así como también la identificación de claves primarias, alternativas, foráneas y los campos no nulos.

Los nombres de las tablas y atributos resultantes serán definidos de manera que sean fácil de identificar y de leer. Oracle permite que los nombres de las tablas y campos tengan un tamaño máximo de 128 caracteres. Sin embargo, se intentará en lo medida de lo posible tener nombres concisos y de un tamaño adecuado, es decir, por un lado, que no sean muy cortos y sea difícil conocer a que entidad o atributo se esta haciendo referencia, y por otro, que tampoco sean excesivamente largos, ya que a nivel de usuario y de desarrollo no es cómodo.

La notación utilizada será la siguiente:

- Se denotan las relaciones a partir del nombre, seguido de los atributos separados por comas entre paréntesis.

- Las claves primarias son los atributos que están subrayados por una línea continua.
- Las claves alternativas o únicas son aquellos atributos que están subrayados por una línea discontinua.
- Los atributos que están en negrita indican que son atributos que no pueden ser valores nulos.
- Las claves foráneas se identifican como texto, seguido de la entidad se especifica el campo indicando que es una clave foránea y la tabla correspondiente.

Se toma en cuenta también, que la gran mayoría de las tablas, tendrán un campo ID que se corresponderá con la clave primaria (y única) de los registros.

Ejemplo:

Entidad: Empleado

Supuestos: No puede existir empleados con el mismo nif.

Empleado (id, nif, **nombre**, **apellido**, **empresald**)

{empresald} es clave foránea de Empresa

6.1. Diseño lógico de la aplicación

Por lo que realizando la transformación a partir del modelo conceptual obtenemos el siguiente modelo lógico:

- Entidad: Campaña

Supuestos: No pueden existir campañas con el mismo nombre en las mismas fechas.

Campanha (id, nombre, fecha_Inicio, fecha_Fin, **idLider**)

{idLider} es clave foránea de Persona.

- Entidad: Auditoria

Auditoria (id, **descripción**, **coste**, **idCampanha**, **idEmpresa**, **idEmpresaExterna**)

{idCampanha} es clave foránea de Campanha.

{idEmpresa} es clave foránea de Empresa.

{idEmpresaExterna} es clave foránea de EmpresaExterna.

- Entidad: Empresa

Supuestos: No pueden existir empresas con el mismo CIF.

Empresa (id, CIF, **nombre**)

- Entidad: Empresa externa

Supuestos: No pueden existir empresas con el mismo CIF.

EmpresaExterna (id, CIF, **nombre**)

- Entidad: Departamento

Supuestos: No pueden existir departamentos con el mismo nombre en la misma empresa.

Departamento (id, nombre, **idEmpresa**)

{idEmpresa} es clave foránea de Empresa

- Entidad: Auditoria Departamento
Relación entre la entidad auditoria y departamento.
AuditoriaDepartamento (**idAuditoria, idDepartamento**)
{idAuditoria} es clave foránea de Auditoria
{idDepartamento} es clave foránea de Departamento
- Entidad: Control
Control (**id, descripcion, idAuditoria, idCreador, idEjecutor, idRiesgo**)
{idAuditoria} es clave foránea de Auditoria
{idCreador} es clave foránea de Persona
{idEjecutor} es clave foránea de Persona
{idRiesgo} es clave foránea de Riesgo
- Entidad:Muestreo
Muestreo (**id, descripcion, observaciones, resultado, idControl, idResponsable**)
{idControl} es clave foránea de Control
{idResponsable} es clave foránea de Persona
- Entidad:Persona
Supuestos: No pueden existir dos personas con el mismo NIF.
Una persona o trabaja en una empresa o trabaja en una empresa externa, solo alguno de los dos.
Persona (**id, NIF, nombre, idEmpresa, idEmpresaExterna, idDepartamento, idOrigen**)
{idEmpresa} es clave foránea de Empresa
{idEmpresaExterna} es clave foránea de EmpresaExterna
{idDepartamento} es clave foránea de Departamento
{idOrigen} es clave foránea de Origen
- Entidad: Acción
Acción (**id, descripcion, fechaCreacion, fecha_Estimada_Ejecucion, fecha_Real_Ejecucion, idCreador, idEstado, idRiesgo, idResponsable**)
{idCreador} es clave foránea de Persona
{idResponsable} es clave foránea de Persona
{idEstado} es clave foránea de EstadoAccion
{idRiesgo} es clave foránea de Riesgo
- Entidad: Estado acción
EstadoAccion(**id, nombre**)
- Entidad: Acción estado Histórico
- AccionEstadoHistorico(**idAccion, idEstadoActual, fechaEstadoActual, idEstadoAnterior, fechaEstadoAnterior**)
{idAccion} es clave foránea de Accion
{idEstadoActual} es clave foránea de EstadoAccion

{idEstadoAnterior} es clave foránea de EstadoAccion

- Entidad: Estado Riesgo
EstadoRiesgo(id, nombre)
- Entidad: Equipo origen
Origen (id, nombre)
- Entidad: Categoría
Supuestos: No pueden existir categorías con el mismo nombre.
Categoría (id, nombre, **fecha_Creacion**, **idCreador**, **idImportancia**)
{idCreador} es clave foránea de Persona
{idImportancia} es clave foránea de Importancia
- Entidad: Importancia
Supuestos: No pueden existir importancias con el mismo valor.
Importancia (id, valor, nombre)
- Entidad: Criticidad
Supuestos: No pueden existir criticidades con el mismo valor.
Criticidad (id, valor, nombre)
- Entidad: Riesgo
Riesgo (id, **descripcion**, **fecha_Creacion**, **impacto**, **idCategoria**, **idCriticidad**, **idOrigen**, **idResponsable**, **idEstadoRiesgo**)
{idCategoria} es clave foránea de Categoría
{idCriticidad} es clave foránea de Criticidad
{idOrigen} es clave foránea de Origen
{idResponsable} es clave foránea de Persona
{idEstadoRiesgo} es clave foránea de EstadoRiesgo
- Entidad: Logger
Logger (id, **procedimientold**, **entrada**, RSP, observaciones, **fecha_Entrada**)

6.2. Diseño lógico para el repositorio estadístico

- Entidad: REP_GENERAL
REP_GENERAL (deptMasRiesgosEquipoExterno, numRiesgCiberAutoevaActAnt, anhoMasAccionImplemRiesgMitig, top3IdAccionesMaxTiempoAbiert, prcnt_Riesgos_AbiertosImpactoMenor4, personMasAccionsAbiertasAssign, numRiesgosCategoria1)
- Entidad: REP_EST_AÑO_RIESGO
REP_EST_ANHO_RIESGO (anho, numAccionesEstadoCurso, numRiesgosImpacto1, numRiesgosEstadoAbierto, difRiesgosImpacto1InterExter, costeMedioAuditoriasExternas, porcentajeAccionesDescartadas, numeroMedioMuestreos)

- Entidad: REP_AUX_ACCION_ANHO
REP_AUX_ACCION_ANHO (anho, estadoCurso, totalAcciones, descartadas, estadoRiesgoMitigado)
- Entidad: REP_AUX_DEPARTAMENTO
REP_AUX_DEPARTAMENTO (nombre, riesgosAuditoriasExternas)
- Entidad: REP_AUX_PERSONA
REP_AUX_PERSONA (nombre, accionesAbiertasAsignadas)
- Entidad: REP_AUX_RIESGO
REP_AUX_RIESGO (impactoMenor4, impactoMenor4Abierto, categoria1)
- Entidad: REP_AUX_RIESGO_ANHO
REP_AUX_RIESGO_ANHO (anho, impacto1, abiertos, impacto1Interna, impacto1Externa, ciberseguridadAutoevaluacion)
- Entidad: REP_AUX_TOP3_ACCION_ABIERTA_IMPLI_DIAS_ANHO
REP_AUX_TOP3_ACCION_ABIERTA_IMPLI_DIAS_ANHO (anho, idAccion, dias)

7. Diseño físico

Una vez realizado el diseño conceptual y transformado al diseño lógico, es el momento de realizar el diseño físico⁵, el cual es el último paso antes de realizar la creación e implementación de la base de datos.

En el diseño físico tomaremos como base lo que se debe crear, es decir, la base de datos, definición de tablespaces, tablas, índices, secuencias, triggers los procedimientos almacenados.

7.1. Base de datos

La base de datos será creada en Oracle, y a efectos del trabajo de fin de grado, se considera que la empresa ya tiene en un servidor el software para la gestión de bases de datos (SGBD) correspondiente, por lo que, en el momento de implantación, solo será necesario realizar la ejecución de los scripts necesarios para la creación de todos los objetos de la aplicación.

7.2. Tablespaces

⁵ Material de Bases De Datos - Diseño físico de bases de datos. Blai Cabré i Segarra, Jordi Casas Roma, Dolors Costal Costa, Pere Juanola Juanola, Ivo Plana Vallvé, Àngels Rius Gavidia, Ramon Segret i Sala. UOC 7ª Edición (septiembre 2020).

Un tablespace⁶ es la unidad lógica de almacenamiento dentro de la base de datos. Cada tablespace tiene asignado un archivo datafile, el cual es el fichero físico que contiene los espacios para las tablas.

En Oracle existen como mínimo 4 tablespaces que son generados automáticamente en la creación de la base de datos que son los siguientes:

- SYSTEM: Es el encargado de almacenar el diccionario de datos, es decir, todas aquellas tablas que contienen información propia de la base de datos.
- TEMP: En este solo se almacenan objetos temporales.
- ROLLBACK: Éste tablespace es utilizado para la gestión de transacciones incompletas.
- DATA: En el que se almacenan los datos de la aplicación.

Asimismo, se pueden crear tantos tablespaces como sean necesarios, bien a través del propio software SGBD, como también mediante consultas SQL.

Para la aplicación, se decide que será necesario la creación de tres tablespaces. Uno en el que se almacenaran todos los datos de la aplicación, otro para la gestión de índices y el tercero que es el correspondiente al repositorio estadístico.

7.3. Tablas

Las tablas son las estructuras finales donde se almacenan los registros y sobre las cuales se realizarán las consultas. El diseño de las tablas se puede realizar de forma sencilla partiendo de un buen diseño lógico, ya que cada relación se corresponde con una tabla. En este momento además de la tabla, se especifica el tipo de dato de cada atributo, así como también las características de claves primarias, foráneas, valores por defecto o si permite que el valor pueda ser nulo.

Como se desconoce la amplitud, escalabilidad y cantidad de registros que pueden existir en la base de datos, se decide que para los atributos de las tablas y aquellos campos que son identificadores (ID), se usará el tipo INTEGER⁷ cuyo valor máximo es 2147483647.

A continuación, se detallan las tablas resultantes para la base de datos que nos concierne y en la que identificamos los siguientes datos:

Nombre de la tabla, campos y tipos de la tabla, si aceptan o no valores nulos y una descripción del campo. Adicionalmente, se muestran también las restricciones con el nombre, el tipo, así como también la tabla de referencia y los campos afectados.

Las restricciones tienen la siguiente notación:

- Primary key (PK): PK_TABLA_CAMPO.

⁶ <https://www.mywebstudies.com/post/gesti%C3%B3n-de-tablespaces-en-oracle> visitado el 22 de abril de 2023.

⁷ <https://docs.aws.amazon.com/dms/latest/oracle-to-aurora-mysql-migration-playbook/chap-oracle-aurora-mysql.tables.common.html> visitado el 22 de abril de 2023.

- Unique key (UQ): UQ_TABLA_CAMPOS: Identifica que la combinación de campos debe ser única (UQ).
- Foreign key (FK): FK_TABLAORIGEN_TABLADESTINO_CAMPO.
- CHK_TABLA_CAMPO (CHK): Identifica que el campo solo admite ciertos valores.

CAMPANHA				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador de la campaña.	
NOMBRE	VARCHAR2 (250)	Not null	Nombre de la campaña.	
FECHA_INICIO	DATE	Not null	Fecha de inicio de la campaña.	
FECHA_FIN	DATE	Not null	Fecha de fin de la campaña.	
IDLIDER	INTEGER	Not null	Identificador de la persona responsable.	
Restricción				
Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_CAMPANHA_ID	PK	ID		
UQ_CAMPANHA_NOMBRE_FECHAS	UQ	NOMBRE, FECHA_INICIO, FECHA_FIN		
FK_CAMPANHA_PERSONA_ID	FK	IDLIDER	PERSONA	ID

AUDITORIA				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador de la auditoría.	
DESCRIPCION	VARCHAR2 (250)	Not null	Descripción para cada auditoría.	
COSTE	NUMBER (10,2)		Coste de la auditoría.	
IDCAMPANHA	INTEGER	Not null	Identificador de la campaña a la que pertenece la auditoría.	
IDEMPRESA	INTEGER	Not null	Identificador de la empresa a la que pertenece la auditoría.	
IDEMPRESAEXTERNA	INTEGER	Not null	Identificador de la empresa externa que realiza la auditoría.	
Restricción				
Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_AUDITORIA_ID	PK	ID		
FK_AUDITORIA_CAMPANHA_ID	FK	IDCAMPANHA	CAMPANHA	ID
FK_AUDITORIA_EMPRESA_ID	FK	IDEMPRESA	EMPRESA	ID
FK_AUDITORIA_EMPRESAEXTERNA_ID	FK	IDEMPRESAEXTERNA	EMPRESA_EXTERNA	ID

EMPRESA				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador de la empresa.	
CIF	VARCHAR2 (50)	Not null	Identificador fiscal de la empresa.	
NOMBRE	VARCHAR2(100)	Not null	Nombre de la empresa.	
Restricción				
Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_EMPRESA_ID	PK	ID		

UQ_EMPRESA_CIF	UQ	CIF		
----------------	----	-----	--	--

EMPRESA_EXTERNA					
Campo	Tipo	Permite nulo	Descripción		
ID	INTEGER	Not null	Identificador de la empresa externa.		
CIF	VARCHAR2 (50)	Not null	Identificador fiscal de la empresa externa.		
NOMBRE	VARCHAR2(100)	Not null	Nombre de la empresa externa.		
Restricción		Tipo	Campo local	Referencia	
				Tabla	Campo
PK_EMPRESA_EXTERNA_ID		PK	ID		
UQ_EMPRESA_EXTERNA_CIF		UQ	CIF		

DEPARTAMENTO					
Campo	Tipo	Permite nulo	Descripción		
ID	INTEGER	Not null	Identificador del departamento.		
NOMBRE	VARCHAR2(50)	Not null	Nombre del departamento.		
IDEMPRESA	INTEGER	Not null	Identificador de la empresa al que pertenece.		
Restricción		Tipo	Campo local	Referencia	
				Tabla	Campo
PK_DEPARTAMENTO_ID		PK	ID		
UQ_DEPARTAMENTO_NOMBRE_IDEMPRESA		UQ	NOMBRE, IDEMPRESA		
FK_DEPARTAMENTO_EMPRESA_ID		FK	IDEMPRESA	EMPRESA	ID

AUDITORIA_DEPARTAMENTO					
Campo	Tipo	Permite nulo	Descripción		
IDAUDITORIA	INTEGER	Not null	Identificador de la auditoría.		
IDDEPARTAMENTO	INTEGER	Not null	Identificador del departamento.		
Restricción		Tipo	Campo local	Referencia	
				Tabla	Campo
FK_AD_AUDITORIA_ID		FK	IDAUDITORIA	AUDITORIA	ID
FK_AD_DEPARTAMENTO_ID		FK	IDDEPARTAMENTO	DEPARTAMENTO	ID

CONTROL				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador del control.	
DESCRIPCION	VARCHAR2 (250)	Not null	Descripción del control a ejecutar.	
IDAUDITORIA	INTEGER	Not null	Identificador de la auditoría a la que pertenece el control.	
IDCREADOR	INTEGER	Not null	Identificador de la persona que crea el control.	
IDEJECUTOR	INTEGER	Not null	Identificador de la persona que ejecuta el control.	
IDRIESGO	INTEGER		Identificador del riesgo asociado al control.	

Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_CONTROL_ID	PK	ID		
FK_CONTROL_AUDITORIA_ID	FK	IDAUDITORIA	AUDITORIA	ID
FK_CONTROL_CREADOR_ID	FK	IDCREADOR	PERSONA	ID
FK_CONTROL_EJECUTOR_ID	FK	IDEJECUTOR	PERSONA	ID
FK_CONTROL_RIESGO_ID	FK	IDRIESGO	RIESGO	ID

MUESTREO			
Campo	Tipo	Permite nulo	Descripción
ID	INTEGER	Not null	Identificador de la muestra.
DESCRIPCION	VARCHAR2 (100)	Not null	Descripción de la muestra a realizar.
RESULTADO	VARCHAR2 (50)	Not null	Resultado de la muestra. Solo se aceptan valores del enumerado Resultado (NO_EJECUTADO, EN_EJECUCION, OK, KO).
OBSERVACIONES	VARCHAR (1000)		Texto de observaciones.
IDCONTROL	INTEGER	Not null	Identificador del control relacionado a la muestra.
IDRESPONSABLE	INTEGER	Not null	Identificador de la persona responsable a la muestra.

Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_MUESTREO_ID	PK	ID		
FK_MUESTREO_CONTROL_ID	FK	IDCONTROL	CONTROL	ID
FK_MUESTRO_RESPONSABLE_ID	FK	IDRESPONSABLE	PERSONA	ID
CHK_MUESTREO_RESULTADO	CHK	RESULTADO		

PERSONA			
Campo	Tipo	Permite nulo	Descripción
ID	INTEGER	Not null	Identificador de la persona.
NIF	VARCHAR2 (20)	Not null	Identificador fiscal de la persona.
NOMBRE	VARCHAR2 (200)	Not null	Nombre de la persona.
IDEMPRESA	INTEGER		Identificador de la empresa donde trabaja la persona (relación XOR con EmpresaExterna).
IDEMPRESAEXTERNA	INTEGER		Identificador de la empresa externa donde trabaja la persona (relación XOR con Empresa).
IDDEPARTAMENTO	INTEGER	Not null	Identificador del departamento donde trabaja la persona.
IDORIGEN	INTEGER	Not null	Identificador del equipo de origen de la persona.

Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_PERSONA_ID	PK	ID		
FK_PERSONA_EMPRESA_ID	FK	IDEMPRESA	EMPRESA	ID
FK_PERSONA_DEPARTAMENTO_ID	FK	IDDEPARTAMENTO	DEPARTAMENTO	ID
FK_PERSONA_ORIGEN_ID	FK	IDORIGEN	ORIGEN	ID

--

ACCION			
Campo	Tipo	Permite nulo	Descripción
ID	INTEGER	Not null	Identificador de la acción.
DESCRIPCION	VARCHAR2 (200)	Not null	Descripción de la acción.
FECHA_CREACION	DATE	Not null	Fecha de creación.
FECHA_ESTIMADA_EJECUCION	DATE		Fecha estimada de la ejecución.
FECHA_REAL_EJECUCION	DATE		Fecha real de ejecución.
IDCREADOR	INTEGER	Not null	Identificador de la persona creadora de la acción.
IDRESPONSABLE	INTEGER	Not null	Identificador de la persona responsable de la acción.
IDESTADO	INTEGER	Not null	Identificador del estado de la acción.
IDRIESGO	INTEGER	Not null	Identificador del riesgo asociado.

Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_ACCION_ID	PK	ID		
FK_ACCION_CREADOR_ID	FK	IDCREADOR	PERSONA	ID
FK_ACCION_ESTADO_ID	FK	IDESTADO	ESTADO_ACCION	ID
FK_ACCION_RIESGO_ID	FK	IDRIESGO	RIESGO	ID

ESTADO_ACCION			
Campo	Tipo	Permite nulo	Descripción
ID	INTEGER	Not null	Identificador del estado de la acción.
NOMBRE	VARCHAR2 (50)	Not null	Nombre del estado.

Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_ESTADO_ACCION_ID	PK	ID		
UQ_ESTADO_ACCION_NOMBRE	UQ	NOMBRE		

ACCION_ESTADO_HISTORICO			
Campo	Tipo	Permite nulo	Descripción
IDACCION	INTEGER	Not null	Identificador de la acción.
IDESTADOACTUAL	INTEGER	Not null	Identificador del estado actual de la acción.
FECHAESTADO	DATE	Not null	Fecha de inserción del registro.
IDESTADOANTERIOR	INTEGER		Identificador del estado anterior de la acción.
FECHAESTADOANTERIOR	DATE		Fecha del estado anterior.

Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_ACCION_ESTADO_HISTORICO	PK	IDACCION, IDESTADOACCION, FECHA		
FK_AEH_ACCION_ID	FK	IDACCION	ACCION	ID
FK_AEH_ESTADO_ACCION_ID	FK	IDESTADOACTUAL	ESTADO	ID

FK_AEH_ESTADO_ANT_ACCION_ID	FK	IDESTADOANTERIOR	ESTADO	ID
-----------------------------	----	------------------	--------	----

EQUIPO_ORIGEN				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador del equipo de origen.	
NOMBRE	VARCHAR2 (50)	Not null	Nombre del equipo de origen.	
Restricción		Tipo	Campo local	Referencia
				Tabla
				Campo
PK_EQUIPO_ORIGEN_ID		PK	ID	
UQ_EQUIPO_ORIGEN_NOMBRE		UQ	NOMBRE	

CATEGORIA				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador de la categoría.	
NOMBRE	VARCHAR2 (100)	Not null	Nombre de la categoría.	
FECHA_CREACION	DATE	Not null	Fecha de creación de la categoría.	
IDCREADOR	INTEGER	Not null	Identificador de la persona que crea la categoría.	
IDIMPORTANCIA	INTEGER	Not null	Identificador de la importancia de la categoría.	
Restricción		Tipo	Campo local	Referencia
				Tabla
				Campo
PK_CATEGORIA_ID		PK	ID	
FK_CATEGORIA_CREADOR_ID		FK	IDCREADOR	PERSONA ID
FK_CATEGORIA_IMPORTANCIA_ID		FK	IDIMPORTANCIA	IMPORTANCIA ID
UQ_CATEGORIA_NOMBRE		UQ	NOMBRE	

IMPORTANCIA				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador de la importancia.	
NOMBRE	VARCHAR2 (20)	Not null	Nombre de la importancia.	
VALOR	TINYINT	Not null	Valor de la importancia.	
Restricción		Tipo	Campo local	Referencia
				Tabla
				Campo
PK_IMPORTANCIA_ID		PK	ID	
UQ_IMPORTANCIA_NOMBRE_VALOR		UQ	NOMBRE, VALOR	

CRITICIDAD				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador de la criticidad.	
NOMBRE	VARCHAR2 (20)	Not null	Nombre de la criticidad.	
VALOR	TINYINT	Not null	Valor de la criticidad.	
Restricción		Tipo	Campo local	Referencia

			Tabla	Campo
PK_CRITICIDAD_ID	PK	ID		
UQ_CRITICIDAD_NOMBRE_VALOR	UQ	NOMBRE, VALOR		

ESTADO_RIESGO					
Campo	Tipo	Permite nulo	Descripción		
ID	INTEGER	Not null	Identificador del estado.		
NOMBRE	VARCHAR2 (20)	Not null	Nombre del estado.		
Restricción		Tipo	Campo local	Referencia	
PK_ESTADO_ID		PK	ID		
UQ_ESTADO_NOMBRE_VALOR		UQ	NOMBRE		

RIESGO					
Campo	Tipo	Permite nulo	Descripción		
ID	INTEGER	Not null	Identificador del riesgo.		
DESCRIPCION	VARCHAR2 (500)	Not null	Descripción del riesgo.		
FECHA_CREACION	DATE	Not null	Fecha de creación.		
IMPACTO	INTEGER	Not null	Impacto del riesgo (producto de criticidad por importancia).		
IDCATEGORIA	INTEGER	Not null	Identificador de la categoría correspondiente al riesgo.		
IDCRITICIDAD	INTEGER	Not null	Identificador de la criticidad correspondiente al riesgo.		
IDORIGEN	INTEGER	Not null	Identificador del equipo de origen que determino el riesgo.		
IDRESPONSABLE	INTEGER	Not null	Identificador de la persona responsable del riesgo.		
IDESTADORIESGO	INTEGER	Not null	Identificador del estado del riesgo.		
Restricción		Tipo	Campo local	Referencia	
PK_RIESGO_ID		PK	ID		
FK_RIESGO_CATEGORIA_ID		FK	IDCATEGORIA	CATEGORIA	ID
FK_RIESGO_CRITICIDAD_ID		FK	IDCRITICIDAD	CRITICIDAD	ID
FK_RIESGO_ORIGEN_ID		FK	IDORIGEN	ORIGEN	ID
FK_RIESGO_RESPONSABLE_ID		FK	IDRESPONSABLE	PERSONA	ID
FK_RIESGO_ESTADO_RIESGO_ID		FK	IDESTADORIESGO	ESTADORIESGO	ID

LOGGER				
Campo	Tipo	Permite nulo	Descripción	
ID	INTEGER	Not null	Identificador del registro.	
NOMBRE_PROCEDIMIENTO	VARCHAR2 (128)	Not null	Nombre del procedimiento que se registra.	
ENTRADA	VARCHAR2 (1000)	Not null	Parámetros de entrada con los que se ejecuta el procedimiento.	
RSP	VARCHAR2 (50)	Not null	Valor de retorno del procedimiento.	
OBSERVACIONES	VARCHAR2 (250)		Observaciones adicionales.	
FECHA_ENTRADA	DATE	Not null	Fecha en la que se ejecutó el procedimiento.	

Restricción	Tipo	Campo local	Referencia	
			Tabla	Campo
PK_LOGGER_ID	PK	ID		

REP_GENERAL			
Campo	Tipo	Permite nulo	Descripción
DEPTMASRIESGOSEQUIPOEXTERNO	VARCHAR2 (50)		Departamento con mayor número de riesgos detectados por auditorías externas.
NUMRIESGCIBERAUTOEVAECTANT	INTEGER		Número de riesgos de ciberseguridad detectados por autoevaluación.
ANHOMASACCIONIMPLEMRIESGMITIG	INTEGER		Año con mayor número de acciones que al final de año estaban en estado implementada con el riesgo mitigado.
TOP3NOMBREACCIONESMAXTIEMPOABIERT	VARCHAR2 (1000)		Top 3 de acciones que estuvieron más tiempo abiertas.
PORCENTAJERIESGOSABIERTOSIMPACTOMENOR4	NUMBER (3,2)		En el momento de ejecutar la consulta, porcentaje de riesgos no corregidos o mitigados con un impacto menor a 4
PERSONAMASACCIONESABIERTASASIGNADAS	VARCHAR2 (200)		En el momento de ejecutar la consulta, persona de la empresa que tiene más acciones abiertas (en estado de en curso o de definida) asignadas
NUMERORIESGOSCATEGORIA1	INTEGER		En el momento de ejecutar la consulta, número de riesgos de categoría 1 (en cualquier estado)

REP_EST_ANHO_RIESGO			
Campo	Tipo	Permite nulo	Descripción
ANHO	INTEGER	Not null	Año al que pertenecen los datos.
NUMACCIONESESTADOCURSO	INTEGER		Número de acciones en estado en curso.
NUMRIESGOSIMPACTO1	INTEGER		Número de riesgos con impacto 1.
NUMRIESGOSESTADOABIERTO	INTEGER		Número de riesgos en estado abierto.
DIFRIESGOSIMPACTO1INTEREXTER	INTEGER		Diferencia de riesgos de impacto 1 detectados entre el equipo interno y equipo externo.
COSTEMEDIOAUDITORIASEXTERNAS	NUMBER (10, 2)		Coste medio de las auditorías externas.
PORCENTAJEACCIONESDESCARTADAS	NUMBER (3, 2)		Porcentaje de acciones en estado descartada.
NUMEROMEDIOMUESTREOS	NUMBER (10,2)		Número medio de muestreos.

Restricción	Tipo	Campo local
PK_REP_EST_ANHO_RIESGO_ANHO	PK	ANHO

--	--	--

REP_AUX_ACCION_ANHO			
Campo	Tipo	Permite nulo	Descripción
ANHO	Number		Año de acción.
ESTADO_CURSO	Number		Número de acciones de estado En curso.
TOTAL_ACCIONES	Number		Número total de acciones.
DESCARTADAS	Number		Número total de acciones descartadas.
ESTADO_RIESGO_MITIGADO	Number		Número total de acciones en estado Implementada con riesgo mitigado.

REP_AUX_DEPARTAMENTO			
Campo	Tipo	Permite nulo	Descripción
NOMBRE	VARCHAR2(500)		Nombre de departamento.
RIESGOS_AUDITORIAS_EXTERNAS	NUMBER		Número de riesgos detectados por auditorías externas.

REP_AUX_PERSONA			
Campo	Tipo	Permite nulo	Descripción
NOMBRE	VARCHAR2(200)		Nombre de la persona.
ACCIONES_ABIERTAS_ASIGNADAS	NUMBER		Número de acciones en estado abiertas o en curso que la persona tiene asignadas.

REP_AUX_RIESGO			
Campo	Tipo	Permite nulo	Descripción
IMPACTO_MENOR_4	NUMBER		Número de riesgos con impacto menor que 4.
IMPACTO_MENOR_4_ABIERTO	NUMBER		Número de riesgos abiertos con impacto menor que 4.
CATEGORIA_1	NUMBER		Número de riesgos de categoría 1.

REP_AUX_RIESGO_ANHO			
Campo	Tipo	Permite nulo	Descripción
ANHO	NUMBER		Año de riesgo.
IMPACTO_1	NUMBER		Número de riesgos impacto igual a 1.
ABIERTOS	NUMBER		Número de riesgos abiertos.
IMPACTO_1_INTERNA	NUMBER		Número de riesgos impacto igual a 1 de auditorías internas.
IMPACTO_1_EXTERNA	NUMBER		Número de riesgos impacto igual a 1 de auditorías externas.
CIBERSEGURIDAD_AUTOEVALUACION	NUMBER		Número de riesgos de ciberseguridad detectados por autoevaluación.

REP_AUX_TOP3_ACCION_ABIERTA_IMPLD_DIAS_ANHO			
Campo	Tipo	Permite nulo	Descripción
ANHO	NUMBER		Año de acción
IDACCION	NUMBER		Identificador de la acción
DIAS	NUMBER		Número de días entre el cambio.

7.4. Índices

Los índices son estructuras de datos que mejoran la velocidad de las operaciones, de esta forma se logra realizar un acceso más rápido a los registros de una tabla. Se suelen aplicar a aquellos campos sobre los que realizan búsquedas frecuentemente.

En Oracle, cuando se crean las claves primarias en las tablas se crea automáticamente un índice único y no nulo.

Para la aplicación, además de los índices creados por defecto, se prevé crear índices por cada clave foránea de las tablas, donde su nomenclatura será IDX_TABLA_CAMPO.

7.5. Secuencias

Una secuencia es un objeto de base de datos que permite la generación automática de valores. Comúnmente son utilizados para las claves primarias de las tablas, de esa forma se garantiza de forma automática que cada registro tenga un identificador secuencial y único.

Las secuencias serán creadas usando el prefijo SEQ seguido por el nombre de la tabla y el campo sobre el que se aplicara la secuencia. Los valores se incrementarán de 1 en 1 hasta que el tamaño máximo del campo lo permita.

Las secuencias identificadas hasta el momento son las siguientes:

SECUENCIAS		
Nombre	Tabla	Campo
SEQ_CAMPANHA_ID	CAMPANHA	ID
SEQ_AUDITORIA_ID	AUDITORIA	ID
SEQ_EMPRESA_ID	EMPRESA	ID
SEQ_EMPRESA_EXTERNA_ID	EMPRESA_EXTERNA	ID
SEQ_DEPARTAMENTO_ID	DEPARTAMENTO	ID
SEQ_MUESTREO_ID	MUESTREO	ID
SEQ_CONTROL_ID	CONTROL	ID
SEQ_PERSONA_ID	PERSONA	ID
SEQ_ACCION_ID	ACCION	ID
SEQ_ESTADO_ACCION_ID	ESTADO_ACCION	ID
SEQ_ESTADO_RIESGO_ID	ESTADO_RIESGO	ID
SEQ_ORIGEN_ID	ORIGEN	ID
SEQ_RIESGO_ID	RIESGO	ID
SEQ_CATEGORIA_ID	CATEGORIA	ID

SEQ_IMPORTANCIA_ID	IMPORTANCIA	ID
SEQ_CRITICIDAD_ID	CRITICIDAD	ID
SEQ_LOGGER_ID	LOGGER	ID

7.6. Triggers

Por el momento se identifica que es necesario crear los triggers que se ejecuten antes de cada inserción, para calcular el valor de la primary key sobre las tablas correspondientes, realizando un nextval de la secuencia que afecta a la tabla donde se está realizando el insert. Aprovecharemos dichos triggers para realizar el poblado de datos en los repositorios estadísticos.

Los nombres de los triggers se corresponderán con la nomenclatura T_<<MOMENTO>>_<<EVENTO1>>_<<EVENTO2>>_<<TABLA>>_<<AUX>> donde pueden tener los siguientes valores:

MOMENTO: Indica cuando se va a ejecutar el trigger, antes o después del evento, los valores serán “ANT” para antes y “DES” para después.

EVENTOS: Indica que eventos realizara el trigger, puede ser una inserción (INS), eliminación (ELI) y actualización (ACT).

TABLA: Indica la tabla sobre la cual se ejecuta el trigger.

AUX: Un identificador opcional por si existen triggers idénticos.

De tal modo que, si queremos crear un trigger que se ejecute antes de una inserción en la tabla persona, el trigger será llamado T_ANT_INS_PERSONA.

Dicho esto, se identifican que es necesario crear tantos triggers en tablas como secuencias se describieron en el apartado anterior. Cada trigger sobre su correspondiente tabla, se ejecutarán antes de realizar una inserción y la función del trigger es devolver el valor de la secuencia que será el valor que se asignará a la primary key.

Para las relaciones de secuencias con las tablas del apartado anterior se crearán los triggers de la siguiente manera:

T_ANT_INS_<<TABLA>>	
Tabla	<<TABLA>>
Momento	Antes de.
Evento	Inserción
Descripción	Ejecutará la sentencia nextval para la secuencia correspondiente a dicha tabla.

T_ANT_INS_ACT_PERSONA	
Tabla	PERSONA
Momento	Antes de.
Evento	Inserción y actualización
Descripción	Se asume que una persona no puede pertenecer a dos empresas, por lo que el trigger se encargara de comprobar que en el registro a actualizar o insertar no vengán cubiertos simultáneamente los campos idEmpresa e idEmpresaExterna. Si vienen los dos valores, se devuelve un error.

T_ANT_INS_ACT_RIESGO	
Tabla	RIESGO

Momento	Antes de.
Evento	Inserción y actualización
Descripción	<ul style="list-style-type: none"> • Se recupera el valor de la importancia de la categoría del riesgo y se guarda en la variable importanciaValor. • Se recupera el valor de la criticidad y se guarda en la variable criticidadValor. • Se calcula riesgoValor * criticidadValor y se asigna al impacto que se va guardar.

T_DES_INS_ACT_RIESGO	
Tabla	RIESGO
Momento	Después de.
Evento	Inserción y actualización.
Paso inicial	<ul style="list-style-type: none"> • Obtenemos los valores IMPACTO_MENOR_4, IMPACTO_MENOR_4_ABIERTO, CATEGORIA_1 actuales de REP_AUX_RIESGO. • Obtenemos los valores IMPACTO_1, ABIERTOS, IMPACTO_1_INTERNA, IMPACTO_1_EXTERNA, CIBERSEGURIDAD_AUTOEVALUACION actuales de REP_AUX_RIESGO_ANHO para el año del riesgo en cuestión.
Requerimiento	En el momento de ejecutar la consulta, porcentaje de riesgos no corregidos o mitigados con un impacto menor a 4.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Si el valor de impacto es menor 4, se suma uno a IMPACTO_MENOR_4. ○ Si el valor de impacto es menor 4 y el estado es abierto, se suma uno a IMPACTO_MENOR_4_ABIERTO. • Si se está actualizando: <ul style="list-style-type: none"> ○ Si el nuevo valor de impacto es diferente al anterior: <ul style="list-style-type: none"> ▪ Si el nuevo valor de impacto es menor que 4: <ul style="list-style-type: none"> • Se suma uno a IMPACTO_MENOR_4. ▪ En caso contrario: <ul style="list-style-type: none"> • Se resta uno a IMPACTO_MENOR_4. ○ Si cambia el estado del riesgo: <ul style="list-style-type: none"> ▪ Si el nuevo estado es abierto se suma uno a IMPACTO_MENOR_4_ABIERTO. ▪ En caso contrario se resta uno a IMPACTO_MENOR_4_ABIERTO.
Requerimiento	Dado un año en concreto, número total de riesgos con un impacto de 1.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Si el valor de impacto es 1 entonces se suma uno a IMPACTO_1. • Si se está actualizando: <ul style="list-style-type: none"> ○ Si el nuevo valor de impacto es diferente al anterior: <ul style="list-style-type: none"> ▪ Si el nuevo valor de impacto es uno se suma uno a IMPACTO_1. ▪ En caso contrario se resta uno a IMPACTO_1.
Requerimiento	En el último año finalizado, número de riesgos que actualmente aún están en estado de abierto.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Si el estado es abierto entonces se suma uno a ABIERTOS. • Si se está actualizando: <ul style="list-style-type: none"> ○ Si el nuevo estado es diferente al anterior: <ul style="list-style-type: none"> ▪ Si el nuevo estado es abierto entonces se suma uno a ABIERTOS. ▪ En caso contrario se resta uno a ABIERTOS.
Requerimiento	Dado un año en concreto, diferencia entre el número de riesgos de impacto 1 detectados por auditoría interna y externa.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Si el valor de impacto es 1: <ul style="list-style-type: none"> ▪ Si el nuevo valor de origen se corresponde con el identificador de origen interno entonces se suma uno a IMPACTO_1_INTERNA. ▪ Si el nuevo valor de origen se corresponde con el identificador de origen externo entonces se suma uno a IMPACTO_1_EXTERNA. • Si se está actualizando: <ul style="list-style-type: none"> ○ Si ha cambiado el equipo de origen o el impacto:

	<ul style="list-style-type: none"> ▪ Si el nuevo valor de impacto es 1: <ul style="list-style-type: none"> • Si el nuevo valor de origen se corresponde con el identificador del equipo interno se suma uno a IMPACTO_1_INTERNA. • Si el nuevo valor de origen se corresponde con el identificador del equipo externo se suma uno a IMPACTO_1_EXTERNO. ▪ Si el nuevo valor de impacto es diferente de uno: <ul style="list-style-type: none"> • Si el nuevo valor de origen se corresponde con el identificador del equipo interno se resta uno a IMPACTO_1_INTERNA. • Si el nuevo valor de origen se corresponde con el identificador del equipo externo se resta uno a IMPACTO_1_EXTERNO.
Requerimiento	En el momento de ejecutar la consulta, número de riesgos de categoría 1 (en cualquier estado)
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ◦ Si el id de categoría es 1, se suma uno a CATEGORIA_1. • Si se está actualizando: <ul style="list-style-type: none"> ◦ Si la categoría ha cambiado: <ul style="list-style-type: none"> ▪ Si el nuevo id de categoría es 1, se suma uno a CATEGORIA_1. ▪ En caso contrario, se resta uno a CATEGORIA_1.
Requerimiento	Teniendo en cuenta el año en curso y el anterior, número de riesgos de ciberseguridad detectados por autoevaluación de los departamentos.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ◦ Si la categoría se corresponde con el identificador de la categoría de ciberseguridad, y el origen con autoevaluación, entonces se suma uno a CIBERSEGURIDAD_AUTOEVALUACION. • Si se está actualizando: <ul style="list-style-type: none"> ◦ Si ha cambiado la categoría o el equipo de origen: <ul style="list-style-type: none"> ▪ Si la nueva categoría es la categoría de ciberseguridad entonces se suma uno a CIBERSEGURIDAD_AUTOEVALUACION. ▪ En caso contrario se resta uno a CIBERSEGURIDAD_AUTOEVALUACION.
Requerimiento	Con todos los datos disponibles en la BD, departamento con un número mayor de riesgos detectados por auditorías externas.
	<ul style="list-style-type: none"> • Se inserta en REP_AUX_DEPARTAMENTO los departamentos de la tabla DEPARTAMENTO. • Si se está insertando: <ul style="list-style-type: none"> ◦ Si el origen se corresponde con el equipo externo: <ul style="list-style-type: none"> ▪ Obtenemos el nombre del departamento del responsable asignado al riesgo. ▪ Obtenemos de REP_AUX_DEPARTAMENTO el valor actual para el departamento asignado. ▪ Actualizamos REP_AUX_DEPARTAMENTO para dicho departamento sumando uno al valor RIESGOS_AUDITORIAS_EXTERNAS. • Si se está actualizando: • Si el nuevo equipo es equipo externo: <ul style="list-style-type: none"> ◦ Obtenemos el nombre del departamento del nuevo responsable asignado al riesgo. ◦ Obtenemos de REP_AUX_DEPARTAMENTO el valor actual para el departamento asignado. ◦ Actualizamos REP_AUX_DEPARTAMENTO para dicho departamento sumando uno al valor RIESGOS_AUDITORIAS_EXTERNAS. ◦ Si el nuevo responsable es diferente al antiguo responsable: <ul style="list-style-type: none"> ▪ Obtenemos el nombre del departamento del viejo responsable asignado al riesgo. ▪ Obtenemos de REP_AUX_DEPARTAMENTO el valor actual para dicho departamento. ▪ Actualizamos REP_AUX_DEPARTAMENTO para dicho departamento restando uno al valor RIESGOS_AUDITORIAS_EXTERNAS. • Si el nuevo equipo no es equipo externo: <ul style="list-style-type: none"> ◦ Obtenemos el nombre del departamento del viejo responsable asignado al riesgo. ◦ Obtenemos de REP_AUX_DEPARTAMENTO el valor actual para dicho

	<ul style="list-style-type: none"> departamento. ○ Actualizamos REP_AUX_DEPARTAMENTO para dicho departamento restando uno al valor RIESGOS_AUDITORIAS_EXTERNAS.
Paso final	<ul style="list-style-type: none"> • Se inserta o actualiza en REP_AUX_RIESGO con los valores previos calculados. • Se inserta o actualiza en REP_AUX_RIESGO_ANHO con los valores previamente calculados para el año en cuestión.

T_DES_INS_ACT_ACCION	
Tabla	ACCION
Momento	Después de.
Evento	Inserción y actualización.
Paso 1	Se recuperan los valores ESTADO_CURSO, TOTAL_ACCIONES, DESCARTADAS, ESTADO_RIESGO_MITIGADO actuales de la tabla REP_AUX_ACCION_ANHO.
Requerimiento	En el año actual, número total de acciones en estado de en curso.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Si el id de estado es el estado en curso, entonces se suma uno a ESTADO_CURSO. • Si se está actualizando y el nuevo estado es diferente al antiguo: <ul style="list-style-type: none"> ○ Si el nuevo estado es en curso, entonces se suma uno a ESTADO_CURSO. ○ En caso contrario, se resta uno a ESTADO_CURSO.
Requerimiento	Dado un año en concreto, porcentaje de acciones descartadas.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Se suma uno a TOTAL_ACCIONES. ○ Si el estado de la acción es descartado, entonces se suma uno a DESCARTADAS. • Si se está actualizando: <ul style="list-style-type: none"> ○ Si el nuevo estado es diferente al estado anterior: <ul style="list-style-type: none"> ▪ Si el nuevo estado es descartado entonces se suma uno a DESCARTADAS. ▪ En caso contrario, si el viejo estado es diferente a descartada entonces se resta uno a DESCARTADAS.
Requerimiento	Teniendo en cuenta todos los datos disponibles, año con un número mayor de acciones que al final del año en cuestión estaban en estado de implementada con el riesgo mitigado.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Si el nuevo estado se corresponde con "Implementada con el riesgo mitigado" entonces se suma uno a ESTADO_RIESGO_MITIGADO. • Si se está actualizando: <ul style="list-style-type: none"> ○ Si el nuevo estado es diferente al anterior estado: <ul style="list-style-type: none"> ▪ Si el nuevo estado se corresponde con "Implementada con el riesgo mitigado", entonces se suma uno a ESTADO_RIESGO_MITIGADO. ▪ En caso contrario, si el viejo estado se corresponde con "Implementada con el riesgo mitigado", entonces se resta uno a ESTADO_RIESGO_MITIGADO.
Requerimiento	En el momento de ejecutar la consulta, persona de la empresa que tiene más acciones abiertas (en estado de en curso o de definida) asignadas.
	<p>Definiciones: Estado abierto: estado en curso o definida. Estado no abierto: los siguientes estados Implementada con el riesgo mitigado, Implementada con el riesgo corregido o Descartada.</p> <ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Si el estado es abierto: <ul style="list-style-type: none"> ▪ Se inserta o actualiza en REP_AUX_PERSONA el campo ACCIONES_ABIERTAS_ASIGNADAS con el valor actual mas 1 para el nombre de la persona responsable a la acción. • Si se está actualizando: <ul style="list-style-type: none"> ○ Si cambia el responsable, pero no los estados, y el estado de la acción es abierta: <ul style="list-style-type: none"> ▪ En REP_AUX_PERSONA se resta uno a ACCIONES_ABIERTAS_ASIGNADAS para el viejo responsable. ▪ En REP_AUX_PERSONA se suma uno a

	<p>ACCIONES_ABIERTAS_ASIGNADAS para el nuevo responsable.</p> <ul style="list-style-type: none"> ○ Si no cambia el responsable, pero si el estado: <ul style="list-style-type: none"> ▪ Obtenemos el nombre del responsable. ▪ Si el nuevo estado es abierto y el viejo estado no es abierto: <ul style="list-style-type: none"> • En REP_AUX_PERSONA se suma uno a ACCIONES_ABIERTAS_ASIGNADAS para el responsable. ▪ En caso contrario, si el nuevo estado es no abierto y el viejo estado es abierto: <ul style="list-style-type: none"> • En REP_AUX_PERSONA se resta uno a ACCIONES_ABIERTAS_ASIGNADAS para el responsable. ○ Si cambia el estado y el responsable también: <ul style="list-style-type: none"> ▪ Si el nuevo estado es abierto, y el viejo es no abierto: <ul style="list-style-type: none"> • Obtenemos el nombre del nuevo responsable y en REP_AUX_PERSONA sumamos uno a ACCIONES_ABIERTAS_ASIGNADAS. • Obtenemos el nombre del viejo responsable y en REP_AUX_PERSONA restamos uno a ACCIONES_ABIERTAS_ASIGNADAS. ▪ Si el viejo estado es abierto y el nuevo estado es no abierto: <ul style="list-style-type: none"> • Obtenemos el nombre del viejo responsable y en REP_AUX_PERSONA restamos uno a ACCIONES_ABIERTAS_ASIGNADAS.
Requerimiento	Mantener histórico de estados de las acciones.
	<ul style="list-style-type: none"> • Si se está insertando: <ul style="list-style-type: none"> ○ Insertamos en ACCION_ESTADO_HISTORICO el idacción, el estado y la fecha de estado. • Si se está actualizando: <ul style="list-style-type: none"> ○ Si el estado cambia: <ul style="list-style-type: none"> ▪ Obtenemos para el id de acción en cuestión, el id de estado actual y la fecha de estado de la tabla ACCION_ESTADO_HISTORICO. ▪ Insertamos un nuevo registro en ACCION_ESTADO_HISTORICO para el id de acción en cuestión con la fecha actual, el nuevo estado, y estableciendo el estado anterior y la fecha de estado anterior.
Paso final	Insertamos o actualizamos en REP_AUX_ACCION_ANHO, para el año de creación de la acción, los valores calculados previamente.

T_DES_INS_ACT_ACCION_ESTADO_HISTORICO	
Tabla	ACCION_ESTADO_HISTORICO
Momento	Después de.
Evento	Inserción.
Paso inicial	Para la fecha del cambio de estado de la acción, recuperamos la cantidad de registros de REP_AUX_ACCION_ANHO.
Requerimiento	Almacenar el top 3 para cada año, de acciones definidas teniendo en cuenta que cuanto se tardó de pasar a un estado de implementación.
	<ul style="list-style-type: none"> • Si el estado anterior es definido y el nuevo estado es alguno de las implementadas: <ul style="list-style-type: none"> ○ Se calcula el número de días, restando la fecha de estado menos la fecha de estado anterior. ○ Si para el año en cuestión hay menos de 3 registros en la tabla REP_AUX_TOP3_ACCION_ABIERTA_IMPLD_DIAS_ANHO: <ul style="list-style-type: none"> ▪ Se inserta en REP_AUX_TOP3_ACCION_ABIERTA_IMPLD_DIAS_ANHO, para el año en cuestión, el id de acción y el número de días de transición de estado. ○ En caso de que haya 3 registros: <ul style="list-style-type: none"> ▪ Obtenemos de REP_AUX_TOP3_ACCION_ABIERTA_IMPLD_DIAS_ANHO para el año de la fecha de cambio, el idacción y el número de días menor. ▪ Si dicho numero es menor que el nuevo cálculo: <ul style="list-style-type: none"> • Sobrescribimos el valor mínimo, para establecer el nuevo id de acción y el nuevo cálculo, o lo que es lo mismo, eliminamos el mínimo e insertamos el nuevo id acción con el número de días de transición de estado.

T_DES_INS_ACT_AUDITORIA	
Tabla	AUDITORÍA
Momento	Después de.
Evento	Inserción y actualización.
Requerimiento	<ul style="list-style-type: none"> En el último año, coste medio de todas las auditorías externas realizadas.
Descripción	<p>Se asume que las auditorías internas tienen un valor 0 de coste.</p> <ul style="list-style-type: none"> Se consulta el número total de las auditorias que tienen coste diferente de cero del año de la auditoria insertada. Se suma el coste de las auditorias que tienen coste <> de 0 del año de la auditoria insertada. Se realiza el cálculo (sumaCosteAuditoriasExternas / numTotalAuditoriasExternas) para cada año y se guarda en la variable COSTEMEDIOAUDITORIASEXTERNAS en la tabla REP_EST_ANHO_RIESGO.

T_DES_INS_ACT_REP_AUX_RIESGO	
Tabla	REP_AUX_RIESGO
Momento	Después de.
Evento	Inserción y actualización.
Requerimientos afectados	<ul style="list-style-type: none"> En el momento de ejecutar la consulta, porcentaje de riesgos no corregidos o mitigados con un impacto menor a 4. En el momento de ejecutar la consulta, número de riesgos de categoría 1 (en cualquier estado). Teniendo en cuenta el año en curso y el anterior, número de riesgos de ciberseguridad detectados por autoevaluación de los departamentos.
Descripción	<ul style="list-style-type: none"> Obtenemos el año en curso. Obtenemos IMPACTO_MENOR_4, IMPACTO_MENOR_4_ABIERTO, CATEGORIA_1 de la tabla auxiliar REP_AUX_RIESGO. Si el IMPACTO_MENOR_4 es distinto de cero, se calcula el porcentaje de riesgos abiertos de impacto menor 4: $((IMPACTO_MENOR_4_ABIERTO * 100) / IMPACTO_MENOR_4)$. Obtenemos el valor CIBERSEGURIDAD_AUTOEVALUACION de REP_AUX_RIESGO_ANHO para el año en curso. Actualizamos en REP_GENERAL los registros PORCENTAJERIESGOSABIERTOSIMPACTOMENOR4, NUMERORIESGOSCATEGORIA1 y NUMRIESGCIBERAUTOEVAECTANT con los valores calculados previamente.

T_DES_INS_ACT_REP_AUX_RIESGO_ANHO	
Tabla	REP_AUX_RIESGO_ANHO
Momento	Después de.
Evento	Inserción y actualización.
Requerimientos afectados	<ul style="list-style-type: none"> Dado un año en concreto, número total de riesgos con un impacto de 1. En el último año finalizado, número de riesgos que actualmente aún están en estado de abierto. Dado un año en concreto, diferencia entre el número de riesgos de impacto 1 detectados por auditoría interna y externa.
Descripción	<ul style="list-style-type: none"> Obtenemos todos los datos de REP_AUX_RIESGO_ANHO. Para cara registro, calculamos la diferencia de riesgos de impacto 1 detectados por auditorías internas y externas (IMPACTO_1_INTERNA - IMPACTO_1_EXTERNA). Insertamos o actualizamos para el año de cada registro, los valores ANHO, NUMRIESGOSIMPACTO1, NUMRIESGOS ESTADO ABIERTO, DIFRIESGOSIMPACTO1INTEREXTER de la tabla REP_EST_ANHO_RIESGO.

T_DES_INS_ACT_REP_AUX_DEPARTAMENTO	
Tabla	REP_AUX_DEPARTAMENTO
Momento	Después de.
Evento	Inserción y actualización.
Requerimientos afectados	<ul style="list-style-type: none"> Con todos los datos disponibles en la BD, departamento con un número mayor de riesgos detectados por auditorías externas.

Descripción	Obtenemos de REP_AUX_DEPARTAMENTO el departamento que tenga mas riesgos detectados por auditorías externas. Actualizamos el campo DEPTMASRIESGOSEQUIPOEXTERNO de REP_GENERAL con el valor obtenido previamente.
-------------	--

T_DES_INS_ACT_REP_AUX_ACCION_ANHO	
Tabla	REP_AUX_ACCION_ANHO
Momento	Después de.
Evento	Inserción y actualización.
Requerimientos afectados	<ul style="list-style-type: none"> En el año actual, número total de acciones en estado de en curso. Dado un año en concreto, porcentaje de acciones descartadas.
Descripción	<ul style="list-style-type: none"> Obtenemos todos los registros de la tabla auxiliar: REP_AUX_ACCION_ANHO. Se crea un contador para obtener el numero máximo de riesgos en estado mitigados por año. Para registro: <ul style="list-style-type: none"> Si el año del registro es diferente al actual: <ul style="list-style-type: none"> Si el valor de ESTADO_RIESGO_MITIGADO es mayor que el contador, establecemos el contador a este valor, siendo el nuevo máximo. Almacenamos en una variable el año que identifica el año en el que existen mas riesgos en estado mitigado. Si para dicho año existen acciones, se calcula el porcentaje de acciones descartadas. Se inserta o actualiza en REP_EST_ANHO_RIESGO los campos ANHO, NUMACCIONESESTADOCURSO, PORCENTAJEACCIONESDESCARTADAS con los valores calculados previamente.

T_DES_INS_ACT_REP_AUX_PERSONA	
Tabla	REP_AUX_PERSONA
Momento	Después de.
Evento	Inserción y actualización.
Requerimiento	<ul style="list-style-type: none"> En el momento de ejecutar la consulta, persona de la empresa que tiene más acciones abiertas (en estado de en curso o de definida) asignadas.
Descripción	Obtenemos de REP_AUX_PERSONA el nombre de la persona para el valor máximo en el campo ACCIONES_ABIERTAS_ASIGNADAS. Establecemos dicho valor en el campo PERSONAMASACCIONESABIERTASASIGNADAS de REP_GENERAL.

T_DES_INS_ACT_REP_AUX_TOP3_ACCION_ABIERTA_IMPLD_DIAS_ANHO	
Tabla	REP_AUX_TOP3_ACCION_ABIERTA_IMPLD_DIAS_ANHO
Momento	Después de.
Evento	Inserción y actualización.
Requerimiento	<ul style="list-style-type: none"> Teniendo en cuenta sólo el último año finalizado, Top3 de acciones definidas teniendo en cuenta el tiempo que estuvieron abiertas (desde su creación hasta que pasen a uno de los estados de implementada). Se debe indicar el nombre de las 3 acciones que estuvieron más tiempo abiertas.
Descripción	Obtenemos las acciones de REP_AUX_TOP3_ACCION_ABIERTA_IMPLD_DIAS_ANHO, y las insertamos separados por comas en el campo TOP3NOMBREACCIONESMAXTIEMPOABIERT de REP_GENERAL.

8. Diseño de procedimientos almacenados

Los procedimientos almacenados estarán agrupados en paquetes, un paquete por cada tabla correspondiente.

Los procedimientos que se detallan son de alta y modificación, ya que en ningún momento durante los requisitos se ha mencionado la eliminación de registros, por lo que suponemos que en caso de ser necesaria la eliminación de algún registro, la empresa deberá ponerse en contacto con el administrador de base datos para que se hagan las eliminaciones de forma manual.

Durante los procedimientos será necesario realizar comprobaciones comunes, que pueden dar lugar a excepciones, por lo que definimos la siguiente serie de excepciones que son comunes a todos los procedimientos:

- EX_PARAMETROS_OBLIGATORIOS: Utilizada cuando algún parámetro obligatorio tiene valor nulo.
- EX_PK_NO_ENCONTRADA: Utilizada cuando no existe un registro con el identificador pasado como parámetro.
- EX_FK_NO_ENCONTRADA: Utilizada cuando no existe un registro con el identificador pasado como parámetro y debe ser buscado en una tabla de referencia.

A continuación, se detallan los procedimientos almacenados identificados.

PKG_UTILIDADES		
Comunes		
Nombre	Tipo	Descripción
EX_PARAMETROS_OBLIGATORIOS	EXCEPTION	Tipo de excepción que se devuelve cuando existen parámetros obligatorios.
EX_PK_NO_ENCONTRADA	EXCEPTION	Tipo de excepción que se devuelve cuando no existe una clave primaria.
EX_FK_NO_ENCONTRADA	EXCEPTION	Tipo de excepción que se devuelve cuando no existe una clave foránea.
EX_EXCEPCION_NO_CONTROLADA	EXCEPTION	Tipo de excepción general.
TEXTO_EX_PARAMETROS_OBLIGATORIOS	VARCHAR2	Texto para la excepción generada para campos nulos que son obligatorios.
TEXTO_EX_PK_NO_ENCONTRADA	VARCHAR2	Texto de la excepción cuando no existe una clave primaria.
TEXTO_EX_FK_NO_ENCONTRADA	VARCHAR2	Texto de la excepción generada cuando no existe una clave foránea.
TEXTO_EX_EXCEPCION_NO_CONTROLADA	VARCHAR2	Texto para una excepción general.
ARRAY_STRING		Tipo de array asociativo (clave – valor) usado para la comprobación de campos obligatorios de tipo texto.
ARRAY_NUMBER		Tipo de array asociativo (clave – valor) usado para la comprobación de campos obligatorios de tipo número.
ARRAY_DATE		Tipo de array asociativo (clave – valor) usado para la comprobación de campos obligatorios de tipo fecha.
addLog		
Descripción	Inserta un log.	
Entrada	<ul style="list-style-type: none"> • p_nombreProcedimiento • p_entrada • p_RSP • p_observaciones 	
Funcionamiento	1. inserta en la tabla logger los parámetros pasados. La fecha de entrada se obtiene a partir del sysdate.	
compruebaCamposObligatorios		
Descripción	Comprueba que los campos pasados como parámetros sean distintos de null.	
Entrada	<ul style="list-style-type: none"> • obligatoriosString • obligatoriosString • obligatoriosString • vParamAnalizado 	
Funcionamiento	1. inserta en la tabla logger los parámetros pasados.	

	La fecha de entrada se obtiene a partir del sysdate.
Salida	vParamAnalizado con el valor del parámetro que es nulo, indica que hay error.
compruebaExistencialIDEnTabla	
Descripción	Comprueba que un id exista en una tabla.
Entrada	<ul style="list-style-type: none"> • p_id • p_nombre • p_pk • p_observaciones
Funcionamiento	<ul style="list-style-type: none"> • Se comprueba que el id no sea nulo. • Se construye una sentencia select para obtener el número de registros de la tabla especificada en el parámetro p_nombre con la condición where = p_id. • Si el resultado de la condición anterior es cero, quiere decir que no existe el id en dicha tabla, por lo que si el valor de p_pk es true, entonces se devuelve la excepción EX_PK_NO_ENCONTRADA, en caso contrario, es decir que el parámetro p_pk es false, se devuelve la excepción EX_PK_NO_ENCONTRADA.

PKG_CAMPANHA	
altaCampanha	
Descripción	Inserta una campaña en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_nombre • p_fechaInicio • p_fechaFin • p_idResponsable
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de responsable exista en la tabla PERSONA. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP
actualizaCampanha	
Descripción	Actualiza una campaña en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idCampanha • p_nombre • p_fechaInicio • p_fechaFin • p_idResponsable
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idCampanha no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS 3. Se comprueba que el id de campaña pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Se comprueba que idResponsable exista en la tabla PERSONA. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP
PKG_AUDITORIA	

altaAuditoria	
Descripción	Inserta una auditoría en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_descripcion • p_coste • p_idCampanha • p_idEmpresa • p_idEmpresaExterna • p_fechaInicio
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos, salvo el coste cuyo campo permite nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que los parámetros p_idCampanha, p_idEmpresa, p_idEmpresaExterna existan en las tablas CAMPANHA, EMPRESA, EMPRESA_EXTERNA respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaAuditoria	
Descripción	Actualiza una auditoría en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idAuditoria • p_descripcion • p_coste • p_idCampanha • p_idEmpresa • p_idEmpresaExterna
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idAuditoria no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS 3. Se comprueba que el id de auditoría pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Si se pasa el parámetro idCampanha, p_idEmpresa y p_idEmpresaExterna, se comprueba que dicho id exista en las tablas CAMPANHA, EMPRESA y EMPRESA_EXTERNA respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_EMPRESA	
altaEmpresa	
Descripción	Inserta una empresa en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_cif • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la inserción del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la

	<p>excepción generada.</p> <p>5. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado)..</p>
Salida	RSP

actualizaEmpresa	
Descripción	Actualiza una empresa en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idEmpresa • p_cif • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idEmpresa no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de empresa pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Si no hay excepción generada, se crea realiza la actualización del registro. 5. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_EMPRESA_EXTERNA

altaEmpresaExterna	
Descripción	Inserta una empresa externa en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_cif • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que dicho CIF no exista en la tabla EMPRESA. <ol style="list-style-type: none"> a. Si existe, se establece el RSP con la excepción EX_CIF_EMPRESA. 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaEmpresaExterna	
Descripción	Actualiza una empresa externa en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idEmpresaExterna • p_cif • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idEmpresaExterna no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de empresa externa pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Se comprueba que dicho CIF no exista en la tabla EMPRESA. <ol style="list-style-type: none"> a. Si existe, se establece el RSP con la excepción EX_CIF_EMPRESA. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_DEPARTAMENTO

altaDepartamento	
Descripción	Inserta un departamento en la base de datos.

Entrada	<ul style="list-style-type: none"> • p_nombre • p_idEmpresa
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el parámetro p_idEmpresa exista en la tabla EMPRESA. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaDepartamento	
Descripción	Actualiza un departamento en la base de datos.
Entrada	<ul style="list-style-type: none"> • P_idDepartamento • p_nombre • p_idEmpresa
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idDepartamento no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de departamento pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Si se pasa el parámetro p_idEmpresa se comprueba que dicho id exista en la tabla EMPRESA. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_AUDITORIA_DEPARTAMENTO

asignaAuditoriaDepartamento	
Descripción	Inserta un registro en auditoria_departamento en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idAuditoria • p_idDepartamento
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que los parámetros p_idAuditoria y p_idDepartamento existan en las tablas AUDITORIA y DEPARTAMENTO respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_CONTROL

altaControl	
Descripción	Inserta un control en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_descripcion • p_idAuditoria • p_idCreador

	<ul style="list-style-type: none"> • p_idEjecutor • p_idRiesgo
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos (salvo p_idRiesgo). <ol style="list-style-type: none"> a. Si algún valor es nulo (salvo p_idRiesgo), se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que los parámetros p_idAuditoria, p_idCreador y p_idEjecutor existan en las tablas AUDITORIA, PERSONA y PERSONA respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaControl	
Descripción	Actualiza un control en la base de datos.
Entrada	<ul style="list-style-type: none"> • P_idControl • p_descripcion • p_idAuditoria • p_idCreador • p_idEjecutor • p_idRiesgo
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idControl no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS 3. Se comprueba que el id de control pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Se comprueba que los parámetros p_idAuditoria, p_idCreador, p_idEjecutor, p_idRiesgo existan en las tablas AUDITORIA, PERSONA, PERSONA y RIESGO respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_MUESTREO	
altaMuestreo	
Descripción	Inserta una muestra en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_descripcion • p_resultado • p_observaciones • p_idControl • p_idResponsable
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos (salvo p_observaciones). <ol style="list-style-type: none"> a. Si algún valor es nulo (salvo p_ p_observaciones), se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que los parámetros p_idControl y p_idResponsable existan en las tablas CONTROL y PERSONA respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaMuestreo	
Descripción	Actualiza una muestra en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idMuestra • p_descripcion • p_resultado • p_observaciones • p_idControl • p_idResponsable
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idMuestreo no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de muestreo pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Se comprueba que los parámetros p_idControl y p_idResponsable existan en las tablas CONTROL y PERSONA respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_PERSONA	
altaPersona	
Descripción	Inserta una persona en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_nif • p_nombre • p_idEmpresa • p_idEmpresaExterna • p_idDepartamento • p_idOrigen
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Si viene el parámetro p_idEmpresa, se comprueba que los parámetros p_idEmpresa, p_idDepartamento y p_idOrigen existan en las tablas EMPRESA, DEPARTAMENTO y ORIGEN respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 3. Si viene el parámetro p_idEmpresaExterna, se comprueba que el parámetro exista en la tabla EMPRESA_EXTERNA. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 4. Se comprueba que los parámetros p_nif y p_nombre no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 5. Si no hay excepción generada, se crea realiza la inserción del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaPersona	
Descripción	Actualiza una persona en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idPersona • p_nif • p_nombre • p_idEmpresa • p_idEmpresaExterna • p_idDepartamento • p_idOrigen
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idPersona no sean nulo.

	<ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de muestreo pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Si viene el parámetro p_idEmpresa, se comprueba que los parámetros p_idEmpresa, p_idDepartamento y p_idOrigen existan en las tablas EMPRESA, DEPARTAMENTO y ORIGEN respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Si viene el parámetro p_idEmpresaExterna, se comprueba que el parámetro exista en la tabla EMPRESA_EXTERNA. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 6. Se comprueba que los parámetros p_nif y p_nombre no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 7. Si no hay excepción generada, se crea realiza la actualización del registro. 8. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 9. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_ACCION

altaAccion	
Descripción	Inserta una acción en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_descripcion • p_fechaEstimadaEjecucion • p_fechaRealEjecucion • p_idCreador • p_idResponsable • p_idEstado • p_idRiesgo
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que los parámetros p_idCreador, p_idResponsable, p_idEstado y p_idRiesgo existan en las tablas PERSONA, PERSONA, ESTADO_ACCION y RIESGO respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Se realiza la inserción en el histórico (ACCIONESTADOHISTORICO). 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaAccion	
Descripción	Actualiza una acción en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idAccion • p_descripcion • p_fechaEstimadaEjecucion • p_fechaRealEjecucion • p_idCreador • p_idResponsable • p_idEstado • p_idRiesgo
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idAccion no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de acción pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA.

	<ol style="list-style-type: none"> 4. Se comprueba que los parámetros p_idCreador, p_idResponsable, p_idEstado y p_idRiesgo existan en las tablas PERSONA, PERSONA, ESTADO_ACCION y RIESGO respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Se realiza la inserción en el histórico (ACCIONESTADOHISTORICO). 7. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 8. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_ESTADO_ACCION

altaEstadoAccion	
Descripción	Inserta un estado de acción en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la inserción del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 5. Actualización del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaEstadoAccion	
Descripción	Actualiza un estado de acción en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idEstadoAccion • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idEstadoAccion no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de estado de acción pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Si no hay excepción generada, se crea realiza la actualización del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_EQUIPO_ORIGEN

altaEquipoOrigen	
Descripción	Inserta un equipo origen en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la inserción del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 5. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaEquipoOrigen	
Descripción	Actualiza un equipo origen en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idEquipoOrigen • p_nombre

Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idEquipoOrigen no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que el id de estado de acción pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Solo se actualizarán aquellos valores que no sean nulos. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_CATEGORIA	
altaCategoria	
Descripción	Inserta una categoría en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_nombre • p_idCreador • p_idImportancia
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que los parámetros p_idcreador y p_idImportancia existan en las tablas PERSONA e IMPORTANCIA respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA 4. Si no hay excepción generada, se crea realiza la inserción del registro. 5. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 6. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaCategoria	
Descripción	Actualiza una categoría en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idCategoria • p_nombre • p_idCreador • p_idImportancia
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idCategoria no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS 3. Se comprueba que el id de categoría pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Se comprueba que los parámetros idCreador y p_idImportancia existan en las tablas PERSONA e IMPORTANCIA respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Si no hay excepción generada, se crea realiza la actualización del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_IMPORTANCIA	
altaImportancia	
Descripción	Inserta una importancia en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_nombre • p_valor
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de

	<p>entrada y la fecha.</p> <ol style="list-style-type: none"> 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la inserción del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 5. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaImportancia	
Descripción	Actualiza una importancia en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idImportancia • p_nombre • p_valor
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idImportancia no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la actualización del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 5. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_CRITICIDAD	
altaCriticidad	
Descripción	Inserta una criticidad en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_nombre • p_valor
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la inserción del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 5. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaCriticidad	
Descripción	Actualiza una criticidad en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idCriticidad • p_nombre • p_valor
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idCriticidad no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la actualización del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 5. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_ESTADO_RIESGO	
altaEstadoRiesgo	
Descripción	Inserta un estado de riesgo en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de

	<p>entrada y la fecha.</p> <ol style="list-style-type: none"> 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la inserción del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 5. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaEstadoRiesgo	
Descripción	Actualiza un estado de riesgo en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idEstadoRiesgo • p_nombre
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idEstadoRiesgo no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Si no hay excepción generada, se crea realiza la actualización del registro. 4. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 5. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

PKG_RIESGO

altaRiesgo	
Descripción	Inserta un riesgo en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_descripcion • p_idCategoria • p_idCriticidad • p_idOrigen • p_idResponsable • p_idEstadoRiesgo
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que los valores no sean nulos. <ol style="list-style-type: none"> a. Si algún valor es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS. 3. Se comprueba que los parámetros p_idCategoria, p_idCriticidad, p_Origen, p_idResponsable, p_idEstadoOrigen existan en las tablas CATEGORIA, CRITICIDAD, ORIGEN, RESPONSABLE y ESTADO_RIESGO respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 4. Se realiza el calculo del impacto a partir de la categoría y de la criticidad. 5. Si no hay excepción generada, se crea realiza la inserción del registro. 6. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 7. Inserción del registro de log, guardando el RSP y las observaciones (se guarda en las observaciones el id del registro generado).
Salida	RSP

actualizaRiesgo	
Descripción	Actualiza un riesgo en la base de datos.
Entrada	<ul style="list-style-type: none"> • p_idRiesgo • p_descripcion • p_idCategoria • p_idCriticidad • p_idOrigen • p_idResponsable • p_idEstadoRiesgo
Funcionamiento	<ol style="list-style-type: none"> 1. Inserción del registro en el log indicando, el procedimiento, los parámetros de entrada y la fecha. 2. Se comprueba que el p_idRiesgo no sean nulo. <ol style="list-style-type: none"> a. Si es nulo, se establece el RSP con la excepción EX_PARAMETROS_OBLIGATORIOS

	<ol style="list-style-type: none"> 3. Se comprueba que el id de riesgo pasado como parámetro exista. <ol style="list-style-type: none"> a. Si no existe, se establece el RSP con la excepción EX_PK_NO_ENCONTRADA. 4. Se comprueba que los parámetros p_idCategoria, p_idCriticidad, p_Origen, p_idResponsable, p_idEstadoOrigen existan en las tablas CATEGORIA, CRITICIDAD, ORIGEN, RESPONSABLE y ESTADO_RIESGO respectivamente. <ol style="list-style-type: none"> a. Si alguno no existe, se establece el RSP con la excepción EX_FK_NO_ENCONTRADA. 5. Se realiza el cálculo del impacto a partir de la categoría y de la criticidad. 6. Si no hay excepción generada, se crea realiza la actualización del registro. 7. Si ha ocurrido un error en la transacción, se establece el campo RSP con la excepción generada. 8. Inserción del registro de log, guardando el RSP y las observaciones.
Salida	RSP

9. Implementación

Como se especificó anteriormente, se presupone que la empresa ya tiene un servidor SGBD con la base de datos creada, por lo que la para la implementación solo será necesario la ejecución de scripts para su contenido.

9.1. Creación de usuarios

Por tener una organización, se procede a crear un usuario administrador para la gestión de nuestra base de datos, para ello, es necesario crearlo desde el usuario SYSTEM.

El usuario que hemos creado se llama TFG_GPOSESMZ.

El archivo script ejecutado para la creación del usuario de la aplicación se llama: "01_usuarios.sql".

The screenshot shows a SQL query result with the following data:

USUARIO	USER_ID	PASSWORD	ACCOUNT_STATUS	LOCK_DATE	EXPIRY_DATE	DEFAULT_TABLESPACE	TEMPORARY_TABLESPACE	LOCAL_TEMP_TABLESPACE	CREATED	PROFILE	INITIAL_RSRC_CONSUMER_GROUP	EXTERNAL_NAME	PASSWORD_VERIFY
TFG_GPOSESMZ	109 (null)	OPEN	(null)	15/10/23	10920	TFG	TFG		11/05/23	DEFAULT	DEFAULT_CONSUMER_GROUP	(null)	116-14C

Ilustración 8. Usuarios.

9.2. Creación de tablespaces

Se crean 3 tablespaces con 50 megas de almacenamiento, dicho espacio se puede modificar posteriormente en caso de ser necesario.

El script ejecutado para la creación de los tablespaces de la aplicación se llama: "02_tablespaces.sql".

The screenshot shows a SQL query result with the following data:

FILE_NAME	FILE_ID	TABLESPACE_NAME	BYTES	BLOCKS	STATUS	RELATIVE_FNO	AUTOEXTENSIBLE	MAXBYTES	MAXBLOCKS	INCREMENT_BY	USER_BYTE
C:\APP\GOSI2ALO\PRODUCT_21C\ORACLE\DATABASE\TSTFGDATOS.DBF	18	TFS_TFS_DATOS	52428800	6400	AVAILABLE	18	NO	0	0	0	5139021
C:\APP\GOSI2ALO\PRODUCT_21C\ORACLE\DATABASE\TSTFSPUNDECS.DBF	19	TFS_TFS_PUNDECS	52428800	6400	AVAILABLE	19	NO	0	0	0	5139022
C:\APP\GOSI2ALO\PRODUCT_21C\ORACLE\DATABASE\TSTFSPREPOSICIONESESTADISTICO.DBF	20	TFS_TFS_PRE_ESTADISTICO	52428800	6400	AVAILABLE	20	NO	0	0	0	5139022

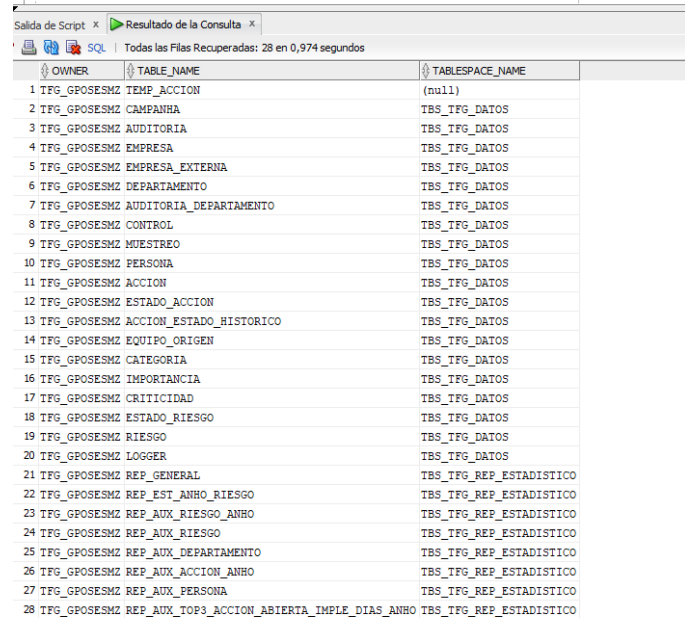
Ilustración 9. Tablespaces de la aplicación.

9.3. Creación de tablas

Se crean todas las tablas necesarias, a partir de la sentencia de `all_tables` podemos obtener todos los datos.

El script ejecutado para la creación de las tablas de la aplicación se llama: "03_tables.sql".

```
SELECT A.OWNER, A.TABLE_NAME, A.TABLESPACE_NAME FROM ALL_TABLES A WHERE OWNER = 'TFG_GPOSESMZ';
```



Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 28 en 0,974 segundos

OWNER	TABLE_NAME	TABLESPACE_NAME
TFG_GPOSESMZ	TEMP_ACCION	(null)
TFG_GPOSESMZ	CAMPANHA	TBS_TFG_DATOS
TFG_GPOSESMZ	AUDITORIA	TBS_TFG_DATOS
TFG_GPOSESMZ	EMPRESA	TBS_TFG_DATOS
TFG_GPOSESMZ	EMPRESA_EXTERNA	TBS_TFG_DATOS
TFG_GPOSESMZ	DEPARTAMENTO	TBS_TFG_DATOS
TFG_GPOSESMZ	AUDITORIA_DEPARTAMENTO	TBS_TFG_DATOS
TFG_GPOSESMZ	CONTROL	TBS_TFG_DATOS
TFG_GPOSESMZ	MUESTREO	TBS_TFG_DATOS
TFG_GPOSESMZ	PERSONA	TBS_TFG_DATOS
TFG_GPOSESMZ	ACCION	TBS_TFG_DATOS
TFG_GPOSESMZ	ESTADO_ACCION	TBS_TFG_DATOS
TFG_GPOSESMZ	ACCION_ESTADO_HISTORICO	TBS_TFG_DATOS
TFG_GPOSESMZ	EQUIPO_ORIGEN	TBS_TFG_DATOS
TFG_GPOSESMZ	CATEGORIA	TBS_TFG_DATOS
TFG_GPOSESMZ	IMPORTANCIA	TBS_TFG_DATOS
TFG_GPOSESMZ	CRITICIDAD	TBS_TFG_DATOS
TFG_GPOSESMZ	ESTADO_RIESGO	TBS_TFG_DATOS
TFG_GPOSESMZ	RIESGO	TBS_TFG_DATOS
TFG_GPOSESMZ	LOGGER	TBS_TFG_DATOS
TFG_GPOSESMZ	REP_GENERAL	TBS_TFG_REF_ESTADISTICO
TFG_GPOSESMZ	REP_EST_ANHO_RIESGO	TBS_TFG_REF_ESTADISTICO
TFG_GPOSESMZ	REP_AUX_RIESGO_ANHO	TBS_TFG_REF_ESTADISTICO
TFG_GPOSESMZ	REP_AUX_RIESGO	TBS_TFG_REF_ESTADISTICO
TFG_GPOSESMZ	REP_AUX_DEPARTAMENTO	TBS_TFG_REF_ESTADISTICO
TFG_GPOSESMZ	REP_AUX_ACCION_ANHO	TBS_TFG_REF_ESTADISTICO
TFG_GPOSESMZ	REP_AUX_PERSONA	TBS_TFG_REF_ESTADISTICO
TFG_GPOSESMZ	REP_AUX_TOP3_ACCION_ABIERTA_IMPLE_DIAS_ANHO	TBS_TFG_REF_ESTADISTICO

Ilustración 10. Tablas de la aplicación.

9.4. Adición de claves foráneas y de campos no nulos.

Una vez se crean las tablas, se modifican para añadir los campos no nulos y las claves foráneas a otras entidades.

El script ejecutado para la modificación de las tablas y así añadir los campos no nulos y claves foráneas se llama: "04_foreingKeys.sql".

9.5. Creación de índices de claves foráneas

A continuación, se crean los índices sobre las claves foráneas.

El script ejecutado para la creación de los índices sobre las claves foráneas se llama: "05_indicesForeingKeys.sql".


```
SELECT I.OWNER, I.INDEX_NAME, I.TABLE_OWNER, I.TABLE_NAME FROM ALL_INDEXES I WHERE OWNER = 'TFG_GPOSESMZ';
```

OWNER	INDEX_NAME	TABLE_OWNER	TABLE_NAME
TFG_GPOSESMZ	CAMPANHA_FK	TFG_GPOSESMZ	CAMPANHA
TFG_GPOSESMZ	UQ_CAMPANHA_NOMBRE_FECHAS	TFG_GPOSESMZ	CAMPANHA
TFG_GPOSESMZ	AUDITORIA_FK	TFG_GPOSESMZ	AUDITORIA
TFG_GPOSESMZ	EMPRESA_FK	TFG_GPOSESMZ	EMPRESA
TFG_GPOSESMZ	UQ_EMPRESA_CIF	TFG_GPOSESMZ	EMPRESA
TFG_GPOSESMZ	EMPRESA_EXTERNA_FK	TFG_GPOSESMZ	EMPRESA_EXTERNA
TFG_GPOSESMZ	EMPRESA_EXTERNA_UK1	TFG_GPOSESMZ	EMPRESA_EXTERNA
TFG_GPOSESMZ	DEPARTAMENTO_FK	TFG_GPOSESMZ	DEPARTAMENTO
TFG_GPOSESMZ	UQ_DEPARTAMENTO_NOMBRE_IDEMPRESA	TFG_GPOSESMZ	DEPARTAMENTO
TFG_GPOSESMZ	CONTROL_FK	TFG_GPOSESMZ	CONTROL
TFG_GPOSESMZ	MUESTREO_FK	TFG_GPOSESMZ	MUESTREO
TFG_GPOSESMZ	PERSONA_FK	TFG_GPOSESMZ	PERSONA
TFG_GPOSESMZ	ACCION_FK	TFG_GPOSESMZ	ACCION
TFG_GPOSESMZ	ESTADO_ACCION_FK	TFG_GPOSESMZ	ESTADO_ACCION
TFG_GPOSESMZ	UQ_ESTADO_ACCION_NOMBRE	TFG_GPOSESMZ	ESTADO_ACCION
TFG_GPOSESMZ	ACCION_ESTADO_HISTORICO_FK	TFG_GPOSESMZ	ACCION_ESTADO_HISTORICO
TFG_GPOSESMZ	EQUIPO_ORIGEN_FK	TFG_GPOSESMZ	EQUIPO_ORIGEN
TFG_GPOSESMZ	UQ_EQUIPO_ORIGEN_NOMBRE	TFG_GPOSESMZ	EQUIPO_ORIGEN
TFG_GPOSESMZ	CATEGORIA_FK	TFG_GPOSESMZ	CATEGORIA
TFG_GPOSESMZ	UQ_CATEGORIA_NOMBRE	TFG_GPOSESMZ	CATEGORIA
TFG_GPOSESMZ	IMPORTANCIA_FK	TFG_GPOSESMZ	IMPORTANCIA
TFG_GPOSESMZ	UQ_IMPORTANCIA_NOMBRE_VALOR	TFG_GPOSESMZ	IMPORTANCIA
TFG_GPOSESMZ	CRITICIDAD_FK	TFG_GPOSESMZ	CRITICIDAD
TFG_GPOSESMZ	UQ_CRITICIDAD_NOMBRE_VALOR	TFG_GPOSESMZ	CRITICIDAD
TFG_GPOSESMZ	ESTADO_RIESGO_FK	TFG_GPOSESMZ	ESTADO_RIESGO
TFG_GPOSESMZ	UQ_ESTADO_NOMBRE_VALOR	TFG_GPOSESMZ	ESTADO_RIESGO
TFG_GPOSESMZ	RIESGO_FK	TFG_GPOSESMZ	RIESGO
TFG_GPOSESMZ	LOGGER_FK	TFG_GPOSESMZ	LOGGER
TFG_GPOSESMZ	REP_EST_ANHO_RIESGO_FK	TFG_GPOSESMZ	REP_EST_ANHO_RIESGO
TFG_GPOSESMZ	IDX_CAMPANHA_IDLIDER	TFG_GPOSESMZ	CAMPANHA
TFG_GPOSESMZ	IDX_AUDITORIA_IDCAMPANHA	TFG_GPOSESMZ	AUDITORIA
TFG_GPOSESMZ	IDX_AUDITORIA_IDEMPRESA	TFG_GPOSESMZ	AUDITORIA
TFG_GPOSESMZ	IDX_AUDITORIA_IDEMPRESAEXTERNA	TFG_GPOSESMZ	AUDITORIA
TFG_GPOSESMZ	IDX_DEPARTAMENTO_IDEMPRESA	TFG_GPOSESMZ	DEPARTAMENTO

Ilustración 11. Algunos de los índices de la aplicación.

9.6. Creación de secuencias

Se crean las secuencias necesarias, para ello se ejecuta el archivo “06_secuencias.sql”.

```
select s.sequence_owner, s.sequence_name, s.min_value, s.max_value, s.increment_by from all_sequences s
where sequence_owner = 'TFG_GPOSESMZ';
```

SEQUENCE_OWNER	SEQUENCE_NAME	MIN_VALUE	MAX_VALUE	INCREMENT_BY
TFG_GPOSESMZ	SEQ_ACCION_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_AUDITORIA_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_CAMPANHA_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_CATEGORIA_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_CONTROL_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_CRITICIDAD_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_DEPARTAMENTO_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_EMPRESA_EXTERNA_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_EMPRESA_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_ESTADO_ACCION_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_ESTADO_RIESGO_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_IMPORTANCIA_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_LOGGER_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_MUESTREO_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_ORIGEN_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_PERSONA_ID	1	99999999999999999999999999999999	1
TFG_GPOSESMZ	SEQ_RIESGO_ID	1	99999999999999999999999999999999	1

Ilustración 12. Secuencias.

9.7. Creación de triggers

Una vez que tenemos las tablas, claves foráneas, índices y secuencias, procedemos a crear los triggers a través del archivo “07_triggers.sql”.

```
SELECT T.OWNER, T.TRIGGER_NAME, T.TRIGGER_TYPE, T.TRIGGERING_EVENT, T.TABLE_NAME FROM ALL_TRIGGERS T WHERE OWNER = 'TFG_GPOSESMZ';
```

OWNER	TRIGGER_NAME	TRIGGER_TYPE	TRIGGERING_EVENT	TABLE_NAME
TFG_GPOSESMZ	T_ANT_INS_MUESTREO	BEFORE EACH ROW INSERT	INSERTED	MUESTREO
TFG_GPOSESMZ	T_ANT_INS_CONTROL	BEFORE EACH ROW INSERT	CONTROL	CONTROL
TFG_GPOSESMZ	T_DES_INS_ACT_AUDITORIA	AFTER STATEMENT INSERT OR UPDATE	INSERT OR UPDATE	AUDITORIA
TFG_GPOSESMZ	T_ANT_INS_AUDITORIA	BEFORE EACH ROW INSERT	INSERTED	AUDITORIA
TFG_GPOSESMZ	T_ANT_INS_CAMPANHA	BEFORE EACH ROW INSERT	INSERTED	CAMPANHA
TFG_GPOSESMZ	T_ANT_INS_EMPRESA_EXTERNA	BEFORE EACH ROW INSERT	INSERTED	EMPRESA_EXTERNA
TFG_GPOSESMZ	T_ANT_INS_EMPRESA	BEFORE EACH ROW INSERT	INSERTED	EMPRESA
TFG_GPOSESMZ	T_ANT_INS_ESTADO_ACCION	BEFORE EACH ROW INSERT	INSERTED	ESTADO_ACCION
TFG_GPOSESMZ	T_ANT_INS_ESTADO_RIESGO	BEFORE EACH ROW INSERT	INSERTED	ESTADO_RIESGO
TFG_GPOSESMZ	T_DES_INS_ACT_REF_AUX_RIESGO	AFTER STATEMENT INSERT OR UPDATE	INSERT OR UPDATE	REF_AUX_RIESGO
TFG_GPOSESMZ	T_DES_INS_ACT_REF_AUX_RIESGO_ANNO	AFTER STATEMENT INSERT OR UPDATE	INSERT OR UPDATE	REF_AUX_RIESGO_ANNO
TFG_GPOSESMZ	T_DES_INS_ACT_REF_AUX_DEPARTAMENTO	AFTER STATEMENT INSERT OR UPDATE	INSERT OR UPDATE	REF_AUX_DEPARTAMENTO
TFG_GPOSESMZ	T_DES_INS_ACT_REF_AUX_ACCION_ANNO	AFTER STATEMENT INSERT OR UPDATE	INSERT OR UPDATE	REF_AUX_ACCION_ANNO
TFG_GPOSESMZ	T_DES_INS_ACT_REF_AUX_PERSONA	AFTER STATEMENT INSERT OR UPDATE	INSERT OR UPDATE	REF_AUX_PERSONA
TFG_GPOSESMZ	T_DES_INS_ACT_REF_AUX_TOP3_ACCION_ABIERTA_IMPLE_DIAS_ANNO	AFTER STATEMENT INSERT OR UPDATE	INSERT OR UPDATE	REF_AUX_TOP3_ACCION_ABIERTA_IMPLE_DIAS_ANNO
TFG_GPOSESMZ	T_ANT_INS_LOGGER	BEFORE EACH ROW INSERT	INSERTED	LOGGER
TFG_GPOSESMZ	T_ANT_INS_CATEGORIA	BEFORE EACH ROW INSERT	INSERTED	CATEGORIA
TFG_GPOSESMZ	T_ANT_INS_PERSONA	BEFORE EACH ROW INSERT OR UPDATE	INSERT OR UPDATE	PERSONA
TFG_GPOSESMZ	T_ANT_INS_PERSONA	BEFORE EACH ROW INSERT	INSERTED	PERSONA
TFG_GPOSESMZ	T_ANT_INS_IMPORTANCIA	BEFORE EACH ROW INSERT	INSERTED	IMPORTANCIA
TFG_GPOSESMZ	T_ANT_INS_CRITICIDAD	BEFORE EACH ROW INSERT	INSERTED	CRITICIDAD
TFG_GPOSESMZ	T_ANT_INS_EQUIPO_ORIGEN	BEFORE EACH ROW INSERT	INSERTED	EQUIPO_ORIGEN
TFG_GPOSESMZ	T_ANT_INS_DEPARTAMENTO	BEFORE EACH ROW INSERT	INSERTED	DEPARTAMENTO
TFG_GPOSESMZ	T_DES_INS_ACT_ACCION_ESTADO_HISTORICO	AFTER EACH ROW INSERT	INSERTED	ACCION_ESTADO_HISTORICO
TFG_GPOSESMZ	T_DES_INS_ACT_ACCION	AFTER EACH ROW INSERT OR UPDATE	INSERT OR UPDATE	ACCION
TFG_GPOSESMZ	T_ANT_INS_ACCION	BEFORE EACH ROW INSERT	INSERTED	ACCION
TFG_GPOSESMZ	T_DES_INS_ACT_RIESGO	AFTER EACH ROW INSERT OR UPDATE	INSERT OR UPDATE	RIESGO
TFG_GPOSESMZ	T_ANT_INS_RIESGO	BEFORE EACH ROW INSERT	INSERTED	RIESGO
TFG_GPOSESMZ	T_ANT_INS_ACT_RIESGO	BEFORE EACH ROW INSERT OR UPDATE	INSERT OR UPDATE	RIESGO

Ilustración 13. Triggers.

9.8. Creación de procedimientos almacenados

Por último, creamos los paquetes y los procedimientos almacenados con el archivo: “08_procedimientos.sql”.

```
select p.owner, p.object_name, p.procedure_name from ALL PROCEDURES p
where p.owner = 'TFG_GPOSESMZ' and p.object_type = 'PACKAGE' AND PROCEDURE_NAME IS NOT NULL;
```

OWNER	OBJECT_NAME	PROCEDURE_NAME
TFG_GPOSESMZ	PKG_ACCION	ACTUALIZAACCION
TFG_GPOSESMZ	PKG_ACCION	ALTAACCION
TFG_GPOSESMZ	PKG_AUDITORIA	ACTUALIZAAUDITORIA
TFG_GPOSESMZ	PKG_AUDITORIA	ALTAAUDITORIA
TFG_GPOSESMZ	PKG_AUDITORIA_DEPARTAMENTO	ASIGNAAUDITORIADEPARTAMENTO
TFG_GPOSESMZ	PKG_CAMPANHA	ACTUALIZACAMPANHA
TFG_GPOSESMZ	PKG_CAMPANHA	ALTACAMPANHA
TFG_GPOSESMZ	PKG_CATEGORIA	ACTUALIZACATEGORIA
TFG_GPOSESMZ	PKG_CATEGORIA	ALTACATEGORIA
TFG_GPOSESMZ	PKG_CONTROL	ACTUALIZACONTROL
TFG_GPOSESMZ	PKG_CONTROL	ALTACONTROL
TFG_GPOSESMZ	PKG_CRITICIDAD	ACTUALIZACRITICIDAD
TFG_GPOSESMZ	PKG_CRITICIDAD	ALTACRITICIDAD
TFG_GPOSESMZ	PKG_DEPARTAMENTO	ACTUALIZADEPARTAMENTO
TFG_GPOSESMZ	PKG_DEPARTAMENTO	ALTADEPARTAMENTO
TFG_GPOSESMZ	PKG_EMPRESA	ACTUALIZAEMPRESA
TFG_GPOSESMZ	PKG_EMPRESA	ALTAEMPRESA
TFG_GPOSESMZ	PKG_EMPRESA_EXTERNA	ACTUALIZAEMPRESAEXTERNA
TFG_GPOSESMZ	PKG_EMPRESA_EXTERNA	ALTAEMPRESAEXTERNA
TFG_GPOSESMZ	PKG_EQUIPO_ORIGEN	ACTUALIZAEQUIPOORIGEN
TFG_GPOSESMZ	PKG_EQUIPO_ORIGEN	ALTAEQUIPOORIGEN
TFG_GPOSESMZ	PKG_ESTADO_ACCION	ACTUALIZAESTADOACCION
TFG_GPOSESMZ	PKG_ESTADO_ACCION	ALTAESTADOACCION
TFG_GPOSESMZ	PKG_ESTADO_RIESGO	ACTUALIZAESTADORIESGO
TFG_GPOSESMZ	PKG_ESTADO_RIESGO	ALTAESTADORIESGO
TFG_GPOSESMZ	PKG_IMPORTANCIA	ACTUALIZAIMPORTANCIA
TFG_GPOSESMZ	PKG_IMPORTANCIA	ALTAIMPORTANCIA
TFG_GPOSESMZ	PKG_MUESTREO	ACTUALIZAMUESTREO
TFG_GPOSESMZ	PKG_MUESTREO	ALTAMUESTREO
TFG_GPOSESMZ	PKG_PERSONA	ACTUALIZAPERSONA
TFG_GPOSESMZ	PKG_PERSONA	ALTAPERSONA
TFG_GPOSESMZ	PKG_RIESGO	ACTUALIZARIESGO

Ilustración 14. Algunos de los paquetes y procedimientos.

9.9. Borrado de base de datos.

Existe también un archivo “borradoBD.sql” que contiene las sentencias para eliminar todos los registros creados anteriormente.

10. Pruebas

Antes de iniciar las pruebas, cabe destacar que se ha realizado un script para inserción de datos de pruebas y de diferentes datos como las categorías, importancias, criticidades, equipos de origen entre otras tablas de datos maestros. En dicho script, también se realiza la inserción de empresa, empresa externa, departamentos y personas (con nombres y NIFs ficticios⁸ salvo el mío) para comenzar con las pruebas de acuerdo con las auditorias, campañas, acciones y riesgos, así como también se realizan las inserciones de todos los datos que puedan estar relacionados, y poder probar el repositorio estadístico.

El script que realiza dicho poblado de datos se llama “09_inicializacionDatos.sql”.

10.1. Creación de plan de pruebas.

Se ha definido un plan de pruebas para comprobar los procesos de inserción y de actualización de los procedimientos almacenados que se encuentra detallado en el anexo: Anexo_II-Pruebas_Altas_Actualizaciones.xlsx.

10.2. Ejecución pruebas.

Por un lado, las pruebas de los procedimientos almacenados se han realizado modificando, por lo general, los nombres y descripciones de los registros generados, es decir, la prueba consta de la realización de una inserción, posteriormente la recuperación de dicho registro, y para terminar la actualización de éste.

Por ejemplo, para las pruebas de la tabla de categorías se realiza lo siguiente:

1. Inserción de una categoría con el nombre: TFG_PR_Categoria_Alta.
2. Recuperación de dicho registro.
3. Actualización del nombre de dicho registro a: TFG_PR_Categoria_Actualizada.

Al finalizar la prueba deberá existir un registro en la tabla categoría con el nombre TFG_PR_Categoria_Actualizada. Y en la tabla LOGGER, deberá existir dos registros, uno correspondiente a la inserción y otro a la actualización.

⁸ Para la generación de nombres y de NIFs ficticios se han usado las siguientes webs respectivamente: <https://generadordni.es> y <https://generadordenombres.online/>.

```

ALTER SESSION SET CURRENT_SCHEMA = TFG_GPOSESMZ;
set SERVEROUTPUT on;

DECLARE
    rsp LOGGERRSP%TYPE;
    vidAux NUMBER;
    vidCategoriaAux NUMBER;
BEGIN
    -- Recuperación de datos maestros
    BEGIN
        SELECT ID INTO vidPersonaAux FROM PERSONA WHERE NIP = '45952991A';
        SELECT ID INTO vidImportanciaAux FROM IMPORTANCIA WHERE NOMBRE = 'Muy Importante';
    EXCEPTION
        WHEN OTHERS THEN
            DBMS_OUTPUT.PUT_LINE('Error en la recuperacion de datos maestros. Código error: ' || SQLCODE || ' Mensaje de error: ' || SQLERRM);
    END;

    --Pruebas
    ----- Categoría
    BEGIN
        PKG_CATEGORIA.altaCategoria('TFG_PR_Categoria_Alta', vidPersonaAux, vidImportanciaAux, rsp);
        SELECT ID INTO vidAux FROM CATEGORIA WHERE NOMBRE = 'TFG_PR_Categoria_Alta';
        PKG_CATEGORIA.actualizaCategoria(vidAux, 'TFG_PR_Categoria_Actualizada', vidPersonaAux, vidImportanciaAux, rsp);
    EXCEPTION
        WHEN OTHERS THEN
            DBMS_OUTPUT.PUT_LINE('Error en las pruebas de CATEGORIA. Código error: ' || SQLCODE || ' Mensaje de error: ' || SQLERRM);
    END;
END;

```

Ilustración 15. Script de ejemplo para pruebas de la tabla de CATEGORIA.

ALTER SESSION SET CURRENT_SCHEMA = TFG_GPOSESMZ;

SELECT * FROM CATEGORIA;

Salida de Script x | Resultado de la Consulta x | Resultado de la Consulta 1 x

Todas las Filas Recuperadas: 5 en 0,004 segundos

ID	NOMBRE	FECHA_CREACION	IDCREADOR	IDIMPORTANCIA
1	1 Ciberseguridad	28/05/23	1	1
2	2 Aplicaciones informáticas	28/05/23	1	1
3	3 Finanzas	28/05/23	1	2
4	4 Seguridad fisica de las instalaciones	28/05/23	1	3
5	5 TFG_PR_Categoria_Actualizada	28/05/23	1	1

Ilustración 16. Resultado de la tabla de CATEGORIA después de la prueba.

Hoja de Trabajo | Generador de Consultas

ALTER SESSION SET CURRENT_SCHEMA = TFG_GPOSESMZ;

SELECT * FROM LOGGERRSP
WHERE ENTRADA LIKE 'TFG_PR_Cate%';

Salida de Script x | Resultado de la Consulta x | Resultado de la Consulta 1 x

Todas las Filas Recuperadas: 2 en 0,003 segundos

ID	NOMBRE_PROCEDIMIENTO	ENTRADA	RSP	OBSERVACIONES	FECHA_ENTRADA
1	TFG_CATEGORIA.altaCategoria	p_nombre = TFG_PR_Categoria_Alta, p_idCreador = 1, p_idImportancia = 1	OK	Categoría creada con el id: 5	28/05/23
2	TFG_CATEGORIA.actualizaCategoria	p_idCategoria = 5, p_nombre = TFG_PR_Categoria_Actualizada, p_idCreador = 1, p_idImportancia = 1	OK	Categoría actualizada con el id: 5	28/05/23

Ilustración 17. Resultado de la tabla de LOGGERRSP después de la prueba.

Las pruebas correspondientes a los procedimientos almacenados se ejecutan a través del archivo: “10_pruebas_procedimientos_almacenados.sql”.

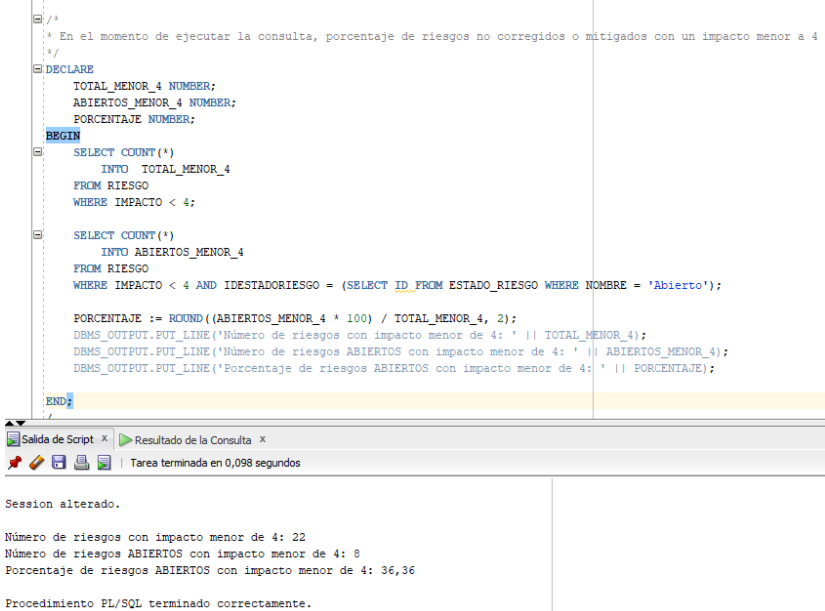
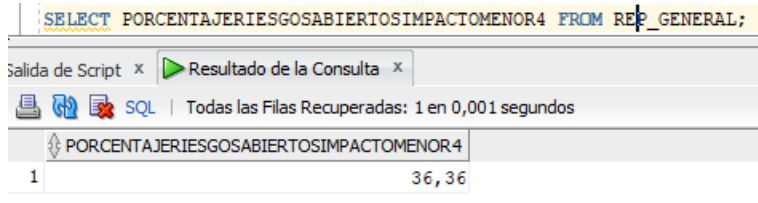
Por otra parte, las pruebas del repositorio estadístico se han realizado de forma secuencial a los requerimientos especificados, para ello se han insertado datos manualmente para los años 2021, 2022 y 2023 y al finalizar se comprueban los datos que existen en las tablas con las consultas y/o operaciones realizadas de forma manual.

El script para la inserción de datos de pruebas es el correspondiente al archivo: “11_datosPruebasRepositorioEstadistico.sql”.

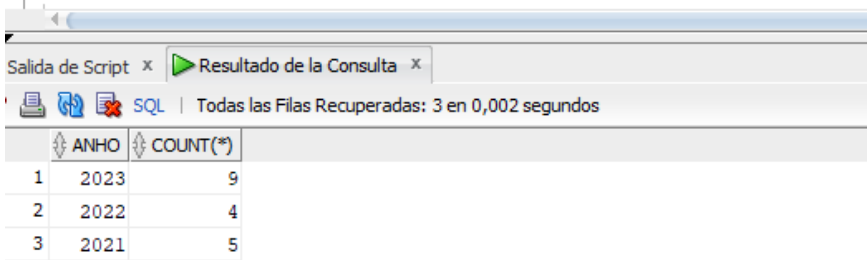
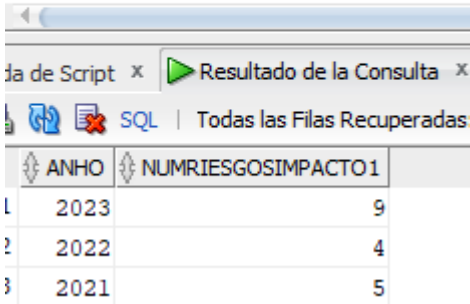
También se proporciona el script que tiene las queries para la comprobación de los datos cuyo nombre es “consultas_comprobaciones_RE.sql”.

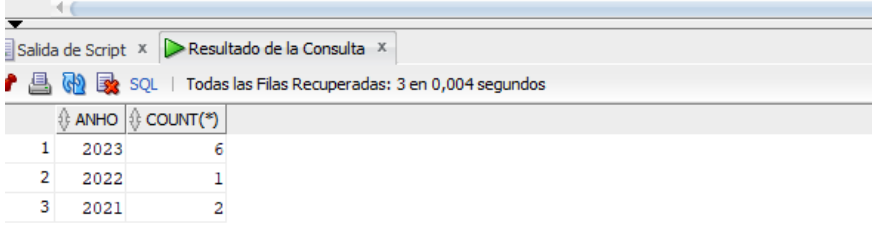
Dicho script contiene por cada requisito dos consultas, una primera que se realiza la obtención de los datos a partir de las tablas de la aplicación, sin tener en cuenta el repositorio estadístico, en esta consulta se obtienen los datos con agrupaciones, operaciones y conteos, es decir, estas consultas no son de tiempo constante igual a 1. A continuación de esta consulta, se realiza una nueva consulta directamente sobre la tabla del repositorio estadístico, sin agrupaciones ni uniones con otras tablas, esta consulta sí es de tiempo constante igual a 1. El resultado de ambas consultas debe ser el mismo.

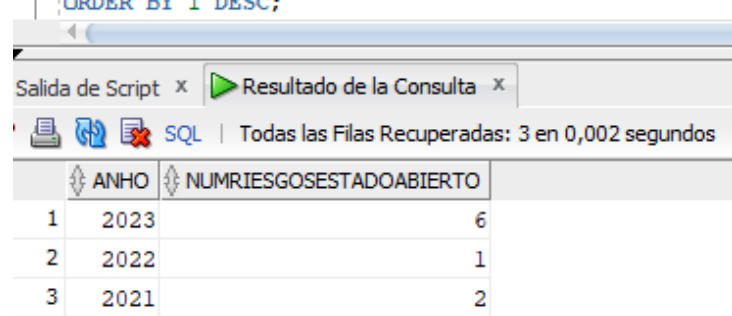
Se procede a mostrar dicha comprobación de los datos después de haber realizado las pruebas.

Requerimiento	En el momento de ejecutar la consulta, porcentaje de riesgos no corregidos o mitigados con un impacto menor a 4.		
Query manual	 <pre> /* * En el momento de ejecutar la consulta, porcentaje de riesgos no corregidos o mitigados con un impacto menor a 4 */ DECLARE TOTAL_MENOR_4 NUMBER; ABIERTOS_MENOR_4 NUMBER; PORCENTAJE NUMBER; BEGIN SELECT COUNT(*) INTO TOTAL_MENOR_4 FROM RIESGO WHERE IMPACTO < 4; SELECT COUNT(*) INTO ABIERTOS_MENOR_4 FROM RIESGO WHERE IMPACTO < 4 AND IDESTADORIESGO = (SELECT ID FROM ESTADO_RIESGO WHERE NOMBRE = 'Abierto'); PORCENTAJE := ROUND((ABIERTOS_MENOR_4 * 100) / TOTAL_MENOR_4, 2); DBMS_OUTPUT.PUT_LINE('Número de riesgos con impacto menor de 4: ' TOTAL_MENOR_4); DBMS_OUTPUT.PUT_LINE('Número de riesgos ABIERTOS con impacto menor de 4: ' ABIERTOS_MENOR_4); DBMS_OUTPUT.PUT_LINE('Porcentaje de riesgos ABIERTOS con impacto menor de 4: ' PORCENTAJE); END; </pre> <p>Salida de Script x Resultado de la Consulta x</p> <p>Tarea terminada en 0,098 segundos</p> <p>Session alterado.</p> <p>Número de riesgos con impacto menor de 4: 22 Número de riesgos ABIERTOS con impacto menor de 4: 8 Porcentaje de riesgos ABIERTOS con impacto menor de 4: 36,36</p> <p>Procedimiento PL/SQL terminado correctamente.</p>		
Query repositorio estadístico	 <pre> SELECT PORCENTAJERIESGOSABIERTOSIMPACTOMENOR4 FROM REP_GENERAL; </pre> <p>Salida de Script x Resultado de la Consulta x</p> <p>SQL Todas las Filas Recuperadas: 1 en 0,001 segundos</p> <table border="1"> <thead> <tr> <th>PORCENTAJERIESGOSABIERTOSIMPACTOMENOR4</th> </tr> </thead> <tbody> <tr> <td>36,36</td> </tr> </tbody> </table>	PORCENTAJERIESGOSABIERTOSIMPACTOMENOR4	36,36
PORCENTAJERIESGOSABIERTOSIMPACTOMENOR4			
36,36			
Resultado	OK		

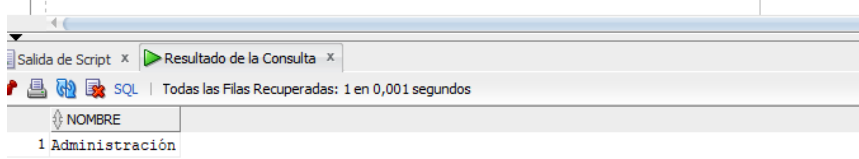
Requerimiento	Dado un año en concreto, número total de riesgos con un impacto de 1.
Query manual	

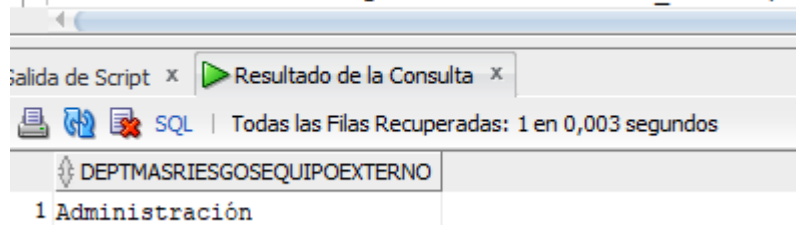
	<pre> /* * Dado un año en concreto, número total de riesgos con un impacto de 1 */ SELECT extract(year from fecha_creacion) anho, COUNT(*) FROM RIESGO WHERE IMPACTO = 1 GROUP BY extract(year from fecha_creacion) ORDER BY 1 DESC; </pre> 
Query Repositorio estadístico	<pre> select ANHO, NUMRIESGOSIMPACTO1 FROM REP_EST_ANHO_RIESGO ORDER BY 1 DESC; </pre> 
Resultado	OK

Requerimiento	En el último año finalizado, número de riesgos que actualmente aún están en estado de abierto.
Query manual	<pre> SELECT extract(year from fecha_creacion) ANHO, COUNT(*) FROM RIESGO WHERE IDESTADORIESGO = (SELECT ID FROM ESTADO_RIESGO WHERE NOMBRE = 'Abierto') GROUP BY extract(year from fecha_creacion) ORDER BY 1 DESC; </pre> 
Query repositorio estadístico	

	<pre>SELECT ANHO, NUMRIESGOESTADOABIERTO FROM REP_EST_ANHO_RIESGO ORDER BY 1 DESC;</pre> 
Resultado	OK

Requerimiento	Con todos los datos disponibles en la BD, departamento con un número mayor de riesgos detectados por auditorías externas.
---------------	---

Query manual	<pre>SELECT NOMBRE FROM (SELECT D.NOMBRE, COUNT(*) TOTAL FROM RIESGO R INNER JOIN PERSONA P ON P.ID = R.IDRESPONSABLE INNER JOIN DEPARTAMENTO D ON D.ID = P.IDDEPARTAMENTO WHERE R.IDORIGEN = (SELECT ID FROM EQUIPO_ORIGEN WHERE NOMBRE = 'Auditoria externa') GROUP BY D.NOMBRE ORDER BY total desc) WHERE ROWNUM = 1;</pre> 
--------------	--

Query repositorio estadístico	<pre>SELECT DEPTMASRIESGOSEQUIPOEXTERNO FROM REP_GENERAL;</pre> 
Resultado	OK

Requerimiento	En el año actual, número total de acciones en estado de en curso.
Query manual	

```

/*
 * En el año actual, número total de acciones en estado de en curso
 */
SELECT extract( year from fecha_creacion) anho, count(*) FROM ACCION
WHERE IDESTADO = (SELECT ID FROM ESTADO_ACCION WHERE NOMBRE = 'En curso')
AND extract( year from fecha_creacion) = EXTRACT (YEAR FROM SYSDATE)
group by extract( year from fecha_creacion)
order by 1 desc;

```

ANHO	COUNT(*)	
1	2023	5

Query repositorio estadístico

```

SELECT anho, NUMACCIONESESTADOCURSO FROM REP_EST_ANHO_RIESGO
WHERE anho = EXTRACT (YEAR FROM SYSDATE);

```

ANHO	NUMACCIONESESTADOCURSO	
1	2023	5

Resultado OK

Requerimiento Dado un año en concreto, diferencia entre el número de riesgos de impacto 1 detectados por auditoría interna y externa.

Query manual

```

/*
 * Dado un año en concreto, diferencia entre el número de riesgos de impacto 1 detectados por auditoria interna y externa
 */
DECLARE
INTERNOS_IMPACTO_1 NUMBER;
EXTERNOS_IMPACTO_1 NUMBER;
CURSOR c_ANHOS IS
SELECT distinct extract( year from fecha_creacion) anho FROM RIESGO ORDER BY 1 DESC;
BEGIN
FOR fila in c_ANHOS
LOOP
SELECT COUNT(*)
INTO INTERNOS_IMPACTO_1
FROM RIESGO
WHERE IMPACTO = 1 AND IDORIGEN = (SELECT ID FROM EQUIPO_ORIGEN WHERE NOMBRE = 'Auditoria interna')
AND extract( year from fecha_creacion) = fila.anho;

SELECT COUNT(*)
INTO EXTERNOS_IMPACTO_1
FROM RIESGO
WHERE IMPACTO = 1 AND IDORIGEN = (SELECT ID FROM EQUIPO_ORIGEN WHERE NOMBRE = 'Auditoria externa')
AND extract( year from fecha_creacion) = fila.anho;

DBMS_OUTPUT.PUT_LINE('La diferencia entre el número de riesgos de impacto 1 detectados por auditoria interna y ex
END LOOP;
END;

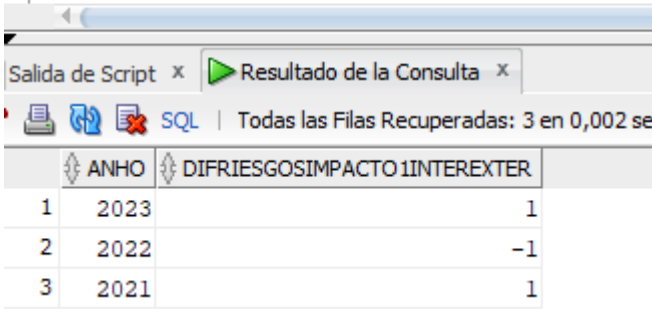
```

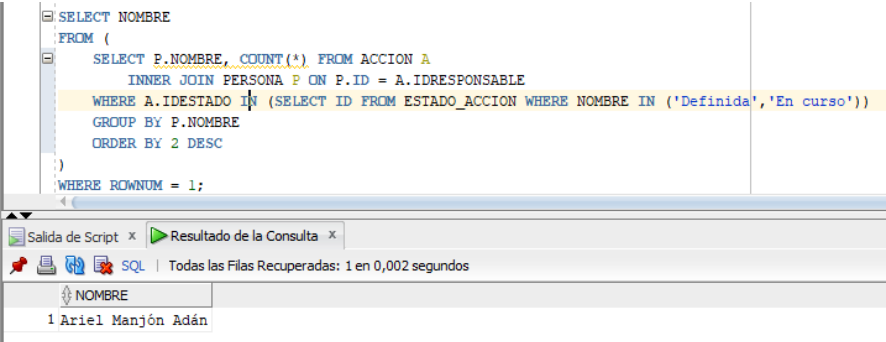
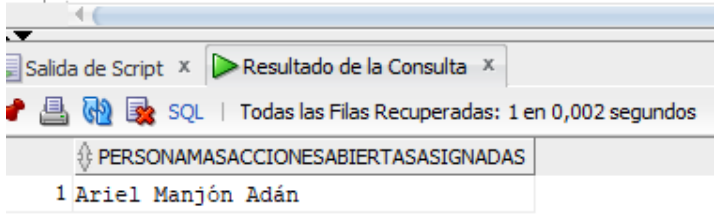
Session alterado.

La diferencia entre el número de riesgos de impacto 1 detectados por auditoria interna y externa para el año 2023 es de: 1
La diferencia entre el número de riesgos de impacto 1 detectados por auditoria interna y externa para el año 2022 es de: -1
La diferencia entre el número de riesgos de impacto 1 detectados por auditoria interna y externa para el año 2021 es de: 1

Procedimiento PL/SQL terminado correctamente.

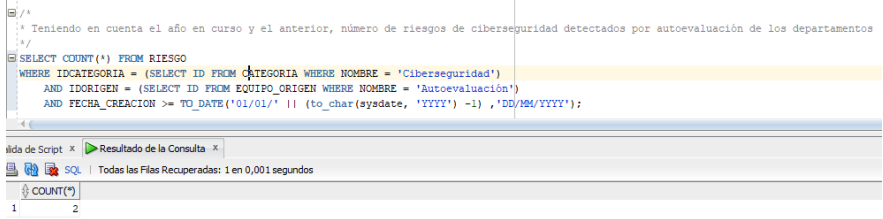
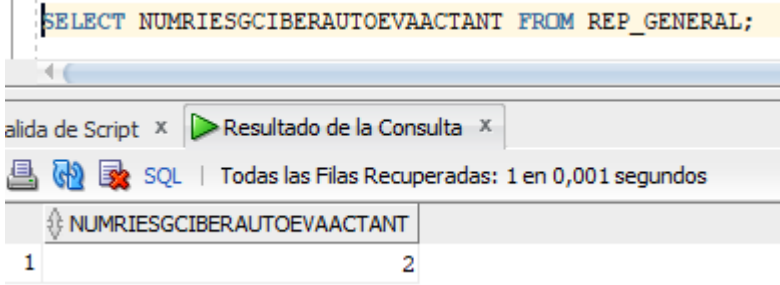
Query repositorio

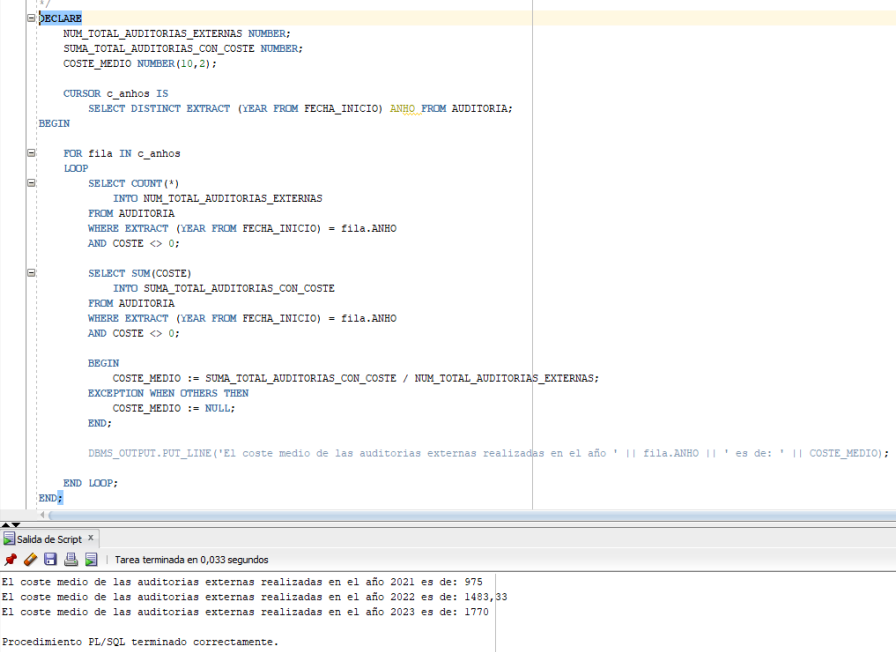
estadístico	<pre>SELECT ANHO, DIFRIESGOSIMPACTO1INTEREXTER FROM REP_EST_ANHO_RIESGO ORDER BY 1 DESC;</pre> 
Resultado	OK

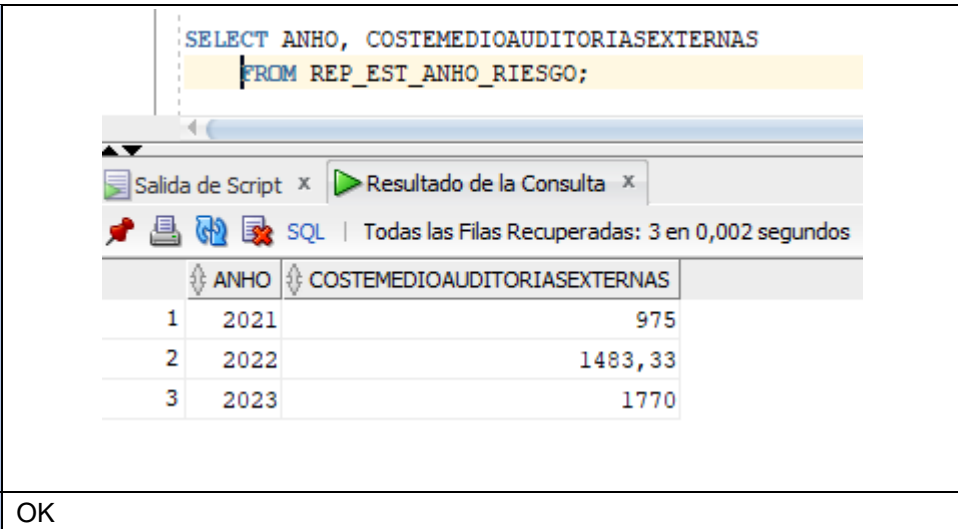
Requerimiento	En el momento de ejecutar la consulta, persona de la empresa que tiene más acciones abiertas (en estado de en curso o de definida) asignadas.
Query manual	<pre>SELECT NOMBRE FROM (SELECT P.NOMBRE, COUNT(*) FROM ACCION A INNER JOIN PERSONA P ON P.ID = A.IDRESPONSABLE WHERE A.IDESTADO IN (SELECT ID FROM ESTADO_ACCION WHERE NOMBRE IN ('Definida','En curso')) GROUP BY P.NOMBRE ORDER BY 2 DESC) WHERE ROWNUM = 1;</pre> 
Query repositorio estadístico	<pre>SELECT PERSONAMASACCIONESABIERTASASIGNADAS FROM REP_GENERAL;</pre> 
Resultado	OK

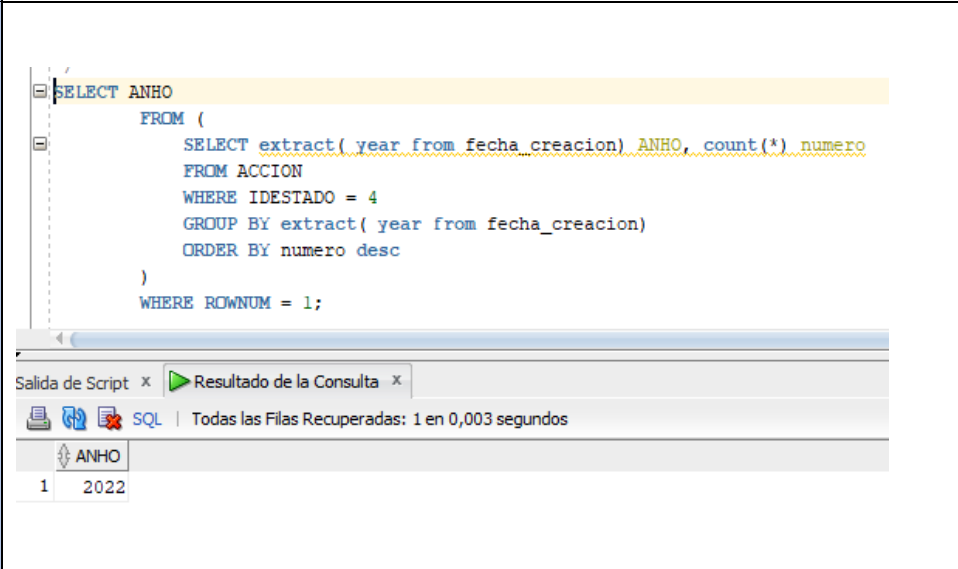
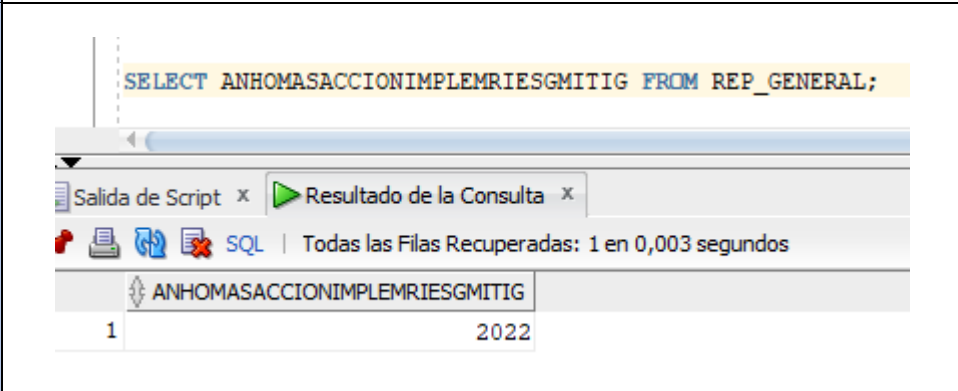
Requerimiento	Número medio de mostreos realizados por año sin considerar el año actual
Query manual	NO IMPLEMENTADO.
Query repositorio estadístico	NO IMPLEMENTADO.
Resultado	KO.

Requerimiento	Teniendo en cuenta el año en curso y el anterior, número de riesgos de
---------------	--

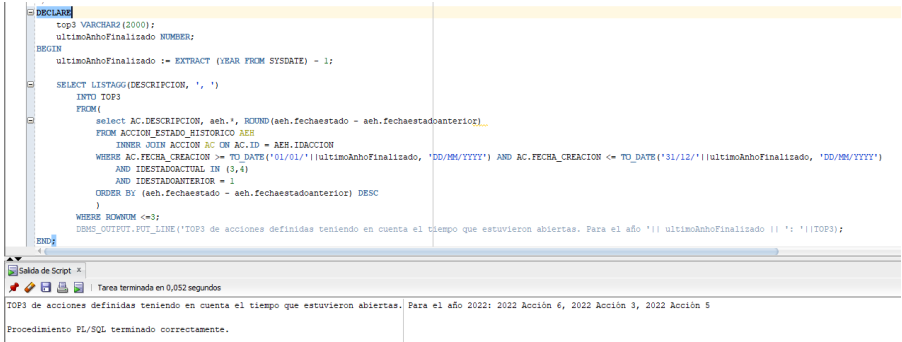
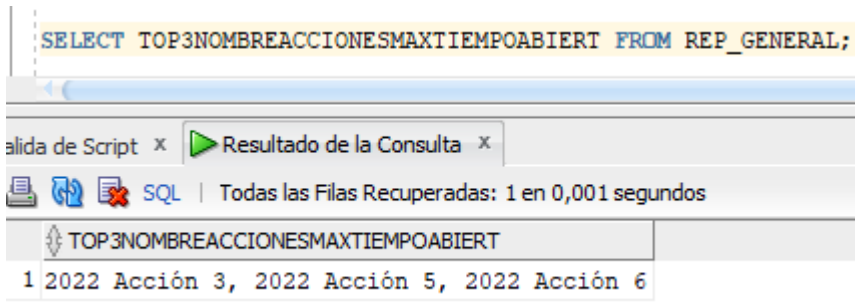
	ciberseguridad detectados por autoevaluación de los departamentos.			
Query manual	 <pre> -- Teniendo en cuenta el año en curso y el anterior, número de riesgos de ciberseguridad detectados por autoevaluación de los departamentos -- SELECT COUNT(*) FROM RIESGO WHERE IDCATEGORIA = (SELECT ID FROM CATEGORIA WHERE NOMBRE = 'Ciberseguridad') AND IDORIGEN = (SELECT ID FROM EQUIPO_ORIGEN WHERE NOMBRE = 'Autoevaluación') AND FECHA_CREACION >= TO_DATE('01/01/' (to_char(sysdate, 'YYYY') -1), 'DD/MM/YYYY'); </pre> <p>Resultado de la Consulta</p> <table border="1"> <thead> <tr> <th>COUNT(*)</th> </tr> </thead> <tbody> <tr> <td>2</td> </tr> </tbody> </table>	COUNT(*)	2	
COUNT(*)				
2				
Query repositorio estadístico	 <pre> SELECT NUMRIESGCIBERAUTOEVA FROM REP_GENERAL; </pre> <p>Resultado de la Consulta</p> <table border="1"> <thead> <tr> <th>NUMRIESGCIBERAUTOEVA</th> </tr> </thead> <tbody> <tr> <td>1</td> </tr> <tr> <td>2</td> </tr> </tbody> </table>	NUMRIESGCIBERAUTOEVA	1	2
NUMRIESGCIBERAUTOEVA				
1				
2				
Resultado	OK			

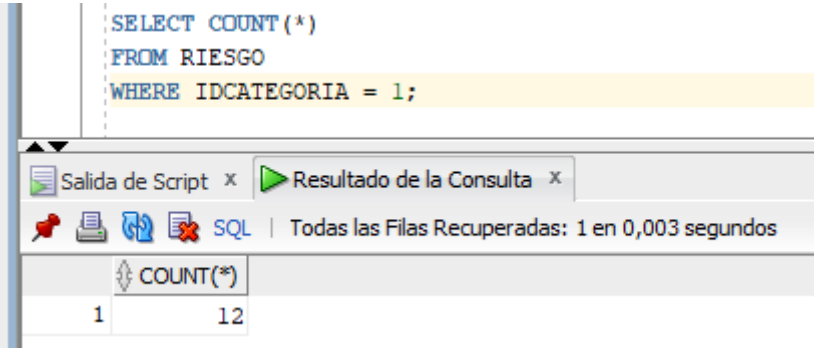
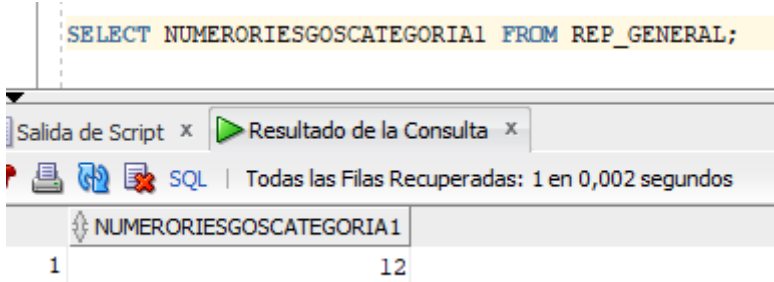
Requerimiento	En el último año, coste medio de todas las auditorías externas realizadas.
Query manual	 <pre> DECLARE NUM_TOTAL_AUDITORIAS_EXTERNAS NUMBER; SUMA_TOTAL_AUDITORIAS_CON_COSTE NUMBER; COSTE_MEDIO NUMBER(10,2); CURSOR c_anhos IS SELECT DISTINCT EXTRACT (YEAR FROM FECHA_INICIO) ANHO FROM AUDITORIA; BEGIN FOR fila IN c_anhos LOOP SELECT COUNT(*) INTO NUM_TOTAL_AUDITORIAS_EXTERNAS FROM AUDITORIA WHERE EXTRACT (YEAR FROM FECHA_INICIO) = fila.ANHO AND COSTE <> 0; SELECT SUM(COSTE) INTO SUMA_TOTAL_AUDITORIAS_CON_COSTE FROM AUDITORIA WHERE EXTRACT (YEAR FROM FECHA_INICIO) = fila.ANHO AND COSTE <> 0; BEGIN COSTE_MEDIO := SUMA_TOTAL_AUDITORIAS_CON_COSTE / NUM_TOTAL_AUDITORIAS_EXTERNAS; EXCEPTION WHEN OTHERS THEN COSTE_MEDIO := NULL; END; DEMS_OUTPUT.PUT_LINE('El coste medio de las auditorias externas realizadas en el año ' fila.ANHO ' es de: ' COSTE_MEDIO); END LOOP; END; </pre> <p>Salida de Script</p> <p>Tarea terminada en 0,033 segundos</p> <pre> El coste medio de las auditorias externas realizadas en el año 2021 es de: 975 El coste medio de las auditorias externas realizadas en el año 2022 es de: 1483,33 El coste medio de las auditorias externas realizadas en el año 2023 es de: 1770 Procedimiento PL/SQL terminado correctamente. </pre>
Query repositorio estadístico	


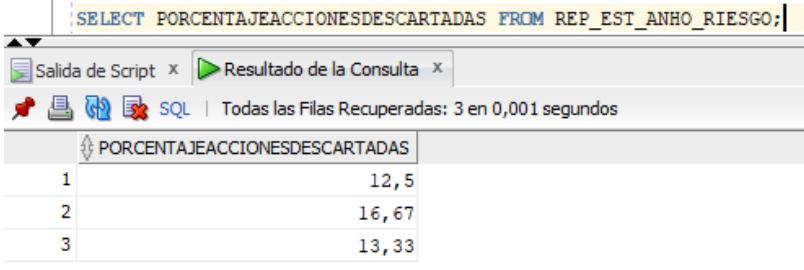
	 <pre>SELECT ANHO, COSTEMEDIOAUDITORIASEXTERNAS FROM REP_EST_ANHO_RIESGO;</pre> <table border="1"> <thead> <tr> <th></th> <th>ANHO</th> <th>COSTEMEDIOAUDITORIASEXTERNAS</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>2021</td> <td>975</td> </tr> <tr> <td>2</td> <td>2022</td> <td>1483,33</td> </tr> <tr> <td>3</td> <td>2023</td> <td>1770</td> </tr> </tbody> </table>		ANHO	COSTEMEDIOAUDITORIASEXTERNAS	1	2021	975	2	2022	1483,33	3	2023	1770
	ANHO	COSTEMEDIOAUDITORIASEXTERNAS											
1	2021	975											
2	2022	1483,33											
3	2023	1770											
Resultado	OK												

Requerimiento	Teniendo en cuenta todos los datos disponibles, año con un número mayor de acciones que al final del año en cuestión estaban en estado de implementada con el riesgo mitigado.				
Query manual	 <pre>SELECT ANHO FROM (SELECT extract(year from fecha_creacion) ANHO, count(*) numero FROM ACCION WHERE IDESTADO = 4 GROUP BY extract(year from fecha_creacion) ORDER BY numero desc) WHERE ROWNUM = 1;</pre> <table border="1"> <thead> <tr> <th></th> <th>ANHO</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>2022</td> </tr> </tbody> </table>		ANHO	1	2022
	ANHO				
1	2022				
Query repositorio estadístico	 <pre>SELECT ANHOMASACCIONIMPLEMRIESGMITIG FROM REP_GENERAL;</pre> <table border="1"> <thead> <tr> <th></th> <th>ANHOMASACCIONIMPLEMRIESGMITIG</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>2022</td> </tr> </tbody> </table>		ANHOMASACCIONIMPLEMRIESGMITIG	1	2022
	ANHOMASACCIONIMPLEMRIESGMITIG				
1	2022				
Resultado	OK				

Requerimiento	Teniendo en cuenta sólo el último año finalizado, Top3 de acciones definidas teniendo en cuenta el tiempo que estuvieron abiertas (desde su creación hasta que pasen a uno de los estados de implementada). Se debe indicar el nombre de las 3 acciones que estuvieron más tiempo abiertas.
---------------	---

<p>Query manual</p>	 <pre> DECLARAR top3 VARCHAR2(2000); ultimoAñoFinalizado NUMBER; BEGIN ultimoAñoFinalizado := EXTRACT (YEAR FROM SYSDATE) - 1; SELECT LISTAGG(DESCRIPCION, ', ') INTO TOP3 FROM(select AC.DESCRIPCION, aeh.*, ROUND(aeh.fechaestado - aeh.fechaestadoanterior) FROM ACCION_ESTADO_HISTORICO AEH INNER JOIN ACCION AC ON AC.ID = AEH.IDACCION WHERE AC.FECHA_CREACION >= TO_DATE('01/01/' ultimoAñoFinalizado, 'DD/MM/YYYY') AND AC.FECHA_CREACION <= TO_DATE('31/12/' ultimoAñoFinalizado, 'DD/MM/YYYY') AND IDESTADOACTUAL IN (3,4) AND IDESTADOANTERIOR = 1 ORDER BY (aeh.fechaestado - aeh.fechaestadoanterior) DESC) WHERE ROWNUM <=3; DBMS_OUTPUT.PUT_LINE('TOP3 de acciones definidas teniendo en cuenta el tiempo que estuvieron abiertas. Para el año ' ultimoAñoFinalizado ': ' TOP3); END; </pre> <p>Tarea terminada en 0,052 segundos TOP3 de acciones definidas teniendo en cuenta el tiempo que estuvieron abiertas. Para el año 2022: 2022 Acción 6, 2022 Acción 3, 2022 Acción 5 Procedimiento PL/SQL terminado correctamente.</p>
<p>Query repositorio estadístico</p>	 <pre> SELECT TOP3NOMBREACCIONESMAXTIEMPOABIERT FROM REP_GENERAL; </pre> <p>Resultado de la Consulta x SQL Todas las Filas Recuperadas: 1 en 0,001 segundos TOP3NOMBREACCIONESMAXTIEMPOABIERT 1 2022 Acción 3, 2022 Acción 5, 2022 Acción 6</p>
<p>Observaciones</p>	<p>Los datos son idénticos solo cambia el orden.</p>
<p>Resultado</p>	<p>OK</p>

<p>Requerimiento</p>	<p>En el momento de ejecutar la consulta, número de riesgos de categoría 1 (en cualquier estado).</p>
<p>Query manual</p>	 <pre> SELECT COUNT (*) FROM RIESGO WHERE IDCATEGORIA = 1; </pre> <p>Resultado de la Consulta x SQL Todas las Filas Recuperadas: 1 en 0,003 segundos COUNT(*) 1 12</p>
<p>Query repositorio estadístico</p>	 <pre> SELECT NUMERORIESGOSCATEGORIA1 FROM REP_GENERAL; </pre> <p>Resultado de la Consulta x SQL Todas las Filas Recuperadas: 1 en 0,002 segundos NUMERORIESGOSCATEGORIA1 1 12</p>
<p>Resultado</p>	<p>OK</p>

Requerimiento	Dado un año en concreto, porcentaje de acciones descartadas.				
Query manual	 <pre> DECLAR NNM_TOTAL_ACCIONES NUMBER; NNM_TOTAL_ACCIONES_DESCARTADAS NUMBER; CANTON_C_AÑOS IS SELECT distinct extract(year from fecha_creacion) anno FROM ACCION ORDER BY 1 DESC; BEGIN FOR fila IN c_AÑOS LOOP SELECT COUNT(*) INTO NNM_TOTAL_ACCIONES FROM ACCION WHERE FECHA_CREACION >= TO_DATE('01/01/' fila.anno,'DD/MM/YYYY') AND FECHA_CREACION < TO_DATE('01/01/' (fila.anno+1),'DD/MM/YYYY'); SELECT COUNT(*) INTO NNM_TOTAL_ACCIONES_DESCARTADAS FROM ACCION WHERE IDESTADO = 0 AND FECHA_CREACION >= TO_DATE('01/01/' fila.anno,'DD/MM/YYYY') AND FECHA_CREACION < TO_DATE('01/01/' (fila.anno+1),'DD/MM/YYYY'); DBMS_OUTPUT.PUT_LINE('El porcentaje de acciones descartadas para el año ' fila.anno ' es ' ROUND((NNM_TOTAL_ACCIONES_DESCARTADAS * 100) / NNM_TOTAL_ACCIONES, 2) '%'); END LOOP; END; </pre>				
Query repositorio estadístico	 <pre> SELECT PORCENTAJEACCIONESDESCARTADAS FROM REP_EST_ANHO_RIESGO; </pre> <table border="1" data-bbox="494 862 909 996"> <thead> <tr> <th>PORCENTAJEACCIONESDESCARTADAS</th> </tr> </thead> <tbody> <tr> <td>1</td> </tr> <tr> <td>2</td> </tr> <tr> <td>3</td> </tr> </tbody> </table>	PORCENTAJEACCIONESDESCARTADAS	1	2	3
PORCENTAJEACCIONESDESCARTADAS					
1					
2					
3					
Resultado	OK				

11. Retrospectivas de entregas.

11.1. PEC 1

La entrega de la primera prueba de evaluación continua ha sido correcta y ha ido según lo previsto. Gracias a la lectura y comprensión del enunciado, junto con las dudas realizadas al tutor, ha sido fácil realizar las tareas de identificación de los objetivos del trabajo, el tipo de enfoque a seguir y realizar una planificación del trabajo con el diagrama de Gantt, la cual se realizó teniendo en cuenta posibles riesgos que pudieran causar retraso de las entregas.

11.2. PEC 2

Para la segunda entrega, después de un análisis profundo del enunciado se encontraron dudas que fueron expuestas al tutor; Gracias a la pronta respuesta, se pudo avanzar con el diseño conceptual, lógico y el inicio del diseño físico.

En este punto se realiza el primer anexo, que es el correspondiente a la preparación e instalación del entorno.

En este punto, en el momento de la entrega se iba con retraso, a fecha 21 de abril se debía estar con inicio de la implementación de la base de datos,

sin embargo, a ese día por casos repentinos no predecibles, aún se estaba trabajando en el diseño de los procedimientos almacenados.

11.3. PEC 3

En la tercera entrega, durante la creación de los triggers y procedimientos almacenados, se encuentran detalles que hacen que sea necesario una serie de modificaciones en los diseños, sobre todo en el del repositorio estadístico. Al mismo tiempo que se realizan los desarrollos se van realizando pruebas, que ayudaran a futuro en el diseño y creación de los casos de pruebas para testear la base de datos.

Durante el desarrollo se observa que en los procedimientos almacenados existe código que se va a repetir en muchos de ellos, por lo que se crea un nuevo paquete de utilidades, donde se desarrollan procedimientos comunes, como el añadido a la tabla de logs, la comprobación de campos obligatorios, o si un registro, ya sea clave primaria o foránea, existe o no en cierta tabla.

En esta entrega también se modifica el documento que servirá de memoria, para incluir el feedback de las evaluaciones proporcionado por el tutor, donde indica que sería una buena idea incluir un apartado para la retrospectiva de las entregas, así como también la de incluir la bibliografía a pie de página, en donde se haga uso.

11.4. Entrega final

Para la entrega final, se han terminado de realizar el desarrollo de todos los procedimientos almacenados, así como también el plan y ejecución de pruebas de dichos procedimientos, el juego de datos de prueba para realizar las consultas contra el repositorio estadístico y cotejarlo con consultas realizadas de forma manual a partir de la unión de tablas.

Tras realizar una entrega, el tutor ha identificado que no estaba correcto el funcionamiento de las consultas en el repositorio estadístico, ya que se no se estaban realizando en tiempo constante igual a 1, el motivo es que, durante el desarrollo, no se tomo en cuenta de que no se podían realizar agrupaciones en los cálculos previos, por lo que se tuvo que cambiar para crear nuevas tablas auxiliares para el repositorio estadístico y cambiar en los triggers para hacer uso de dichas tablas.

El archivo de consultas de las comprobaciones del repositorio estadístico, también fue modificado para incluir comentarios, ya que por cada requisito se hacen dos consultas, uno a nivel de datos agrupados (sin tiempo constante = 1) y otro a nivel de tablas de repositorio estadístico, pero no estaba claro el motivo de ambas consultas, por lo que se modificó para incluir los comentarios oportunos y así tener identificado el motivo de ambas consultas.

12. Conclusiones

Un trabajo de esta índole se puede dividir en 4 etapas fundamentales:

- Análisis de datos y planificación.
- Diseño.
- Implementación.
- Pruebas.

La etapa de análisis de datos y planificación fue sencilla, ya que, a partir del enunciado, junto con las dudas presentadas al tutor, se logró identificar fácilmente los requisitos que debía cumplir la base de datos, así como también fue sencillo realizar la planificación, aunque posteriormente ésta se haya visto afectada con retrasos.

Durante la etapa de diseño, no se obtuvo gran dificultad, una vez más, gracias a la ayuda del tutor que respondió clara, concisa y rápidamente a todas las dudas presentadas. El desarrollo de los diseños se realizó de manera satisfactoria, a pesar de que, por motivos personales, se realizaron tareas que iban por detrás de la planificación original y que se siguieron arrastrando hasta el final del proyecto.

En la etapa de implementación, no hubo problemas, gracias a los conocimientos adquiridos durante el grado, junto con la experiencia profesional en PL-SQL, el desarrollo de los scripts de la base de datos se realizaron de manera efectiva.

Por último, en la etapa de pruebas, se ha encontrado una pequeña dificultad a la hora de saber cómo presentarlas, así como también de preparar y lograr un conjunto de datos que pudiese ser usado para probar todas las funcionalidades de la aplicación. Sin embargo, se logró generar un pequeño conjunto de datos para los años 2021, 2022 y 2023 con la finalidad de poder probar todos los puntos del repositorio estadístico.

Como punto de mejora se detectó que, en las tablas maestras, sería ideal incluir un nuevo campo de texto único como clave primaria. El motivo es que en los triggers, procedimientos y juegos de pruebas, se hacen consultas a dichas tablas maestras buscando el registro por el campo nombre o descripción, por ejemplo: "SELECT * FROM CATEGORIA WHERE NOMBRE="Ciberseguridad";".

Si en un futuro se quiere cambiar el nombre de ese campo, además de cambiar el registro, se debe cambiar en todos los sitios para realizar la adaptación, lo cual no es óptimo.

Lo ideal sería, por ejemplo, a la tabla de categoría, añadir un campo CLAVE único que no cambie en el tiempo y al registro de ciberseguridad establecerle el valor "CIBER". Posteriormente sería el momento de cambiar las queries para que sean del estilo: SELECT * FROM CATEGORIA WHERE CLAVE = "CIBER";".

De esta forma, si en un futuro se quiere cambiar el nombre de la categoría, solo sería necesario realizar dicha actualización y los procedimientos no se verían afectados.

Durante las pruebas también se encontró que el proceso correspondiente para calcular el número medio de muestreos no está implementado, por lo que sería ideal incluirlo en una futura versión.

Para finalizar, expreso que a nivel personal, a pesar de ir con retraso en la planificación, me he sentido cómodo ejerciendo diferentes roles, desde gestor, analista, desarrollador hasta el de tester.

Me he dado cuenta de primera mano de varios puntos:

- Una buena y correcta planificación al inicio de un proyecto, junto con su correspondiente análisis de riesgos y planes de contingencia es de vital importancia para el correcto desarrollo y entrega de un producto.
- Tener una metodología y análisis detallado, conciso y sin dudas hace más sencillo el tiempo de diseño e implementación de un producto.
- Dedicar tiempo extra para tener un correcto y buen diseño conceptual, hace que los puntos de diseño lógico y físico sean menos complejos de realizar, y con una tolerancia a cambios más efectiva y aplicable.
- Los diseños de toda operativa de base de datos como los procedimientos, realizados con seudocódigo, hace que en el momento de implementación nos centremos solo en el desarrollo.

En general, considero que el producto entregado cumple satisfactoriamente con la mayoría de los requisitos solicitados, con un código bien estructurado, de fácil comprensión y en correcto funcionamiento.

13. Glosario

- BBDD: Bases de datos.
- ER: Siglas de modelo Entidad-Relación.
- ERM: Enterprise Risk Management: Marco para administrar los riesgos en el ámbito organizacional.
- Exception: Excepción. Retorno que aparece cuando en alguna parte del proceso ha ocurrido algún error.
- Foreign Key: Clave foránea.
- Gantt: Herramienta gráfica cuyo objetivo es exponer el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado.
- Hardware: Elementos físicos que constituyen un sistema informático.
- ORACLE: Compañía empresarial especializada en soluciones en la nube y en gestión de bases de datos.
- PK: Iniciales de Primary Key que se corresponde con Clave principal.
- Primary Key: Clave Principal.
- SGBD: Sistema Gestor de Base de Datos.
- Software: Programas dentro de un sistema informático.
- Tablespace: Espacio de tablas. Ubicación de almacenamiento donde pueden ser guardados los datos correspondientes a los objetos de una base de datos.
- TFG: Iniciales de "Trabajo de Fin de Grado".
- Trigger: Bloque de código que se ejecuta automáticamente cuando ocurre algún evento.

14. Bibliografía

- Material de Ingeniería del Software
Jordi Pradel Miquel, Jose Raya Martos
UOC 3º Edición (febrero 2016)
- Material de Gestión de Proyectos
José Ramón Rodríguez, Pere Mariné Jové
UOC 3º Edición (septiembre 2018).
- <https://www.youtube.com/watch?v=MZ4rJNYLTIY> visitado el 20 de marzo de 2023.
- <https://dba.stackexchange.com/questions/405/what-is-the-difference-between-sys-and-system-accounts-in-oracle-databases> visitado el 20 de marzo de 2023.
- Material de Bases De Datos
Jordi Casas Roma, Josep Cuartero Olivera
UOC 5º Edición (febrero 2020).
- <https://www.mywebstudies.com/post/gesti%C3%B3n-de-tablespaces-en-oracle> visitado el 22 de abril de 2023.
- Material de Bases De Datos - Diseño físico de bases de datos
Blai Cabré i Segarra, Jordi Casas Roma, Dolors Costal Costa, Pere Juanola Juanola, Ivo Plana Vallvé, Àngels Rius Gavidia, Ramon Segret i Sala
UOC 7º Edición (septiembre 2020).
- <https://docs.aws.amazon.com/dms/latest/oracle-to-aurora-mysql-migration-playbook/chap-oracle-aurora-mysql.tables.common.html> visitado el 22 de abril de 2023.
- <https://generadordenombres.online/> visitado el 24 de mayo de 2023.
- <https://generadordni.es/> visitado el 24 de mayo de 2023.

15. Anexos

1. Anexo I – Entorno desarrollo Oracle.
Una guía en la que se realiza la instalación de Oracle en su versión “Oracle Database 21c Express Edition” junto con el entorno de desarrollo SQL Developer en su versión 22.2.1.
2. Anexo_II-Pruebas_Altas_Actualizaciones.xlsx.
Plan de pruebas para las altas y actualizaciones de los procedimientos almacenados.