

Trabajo Fin de Grado

Bases de Datos Relacionales

Diseño e Implementación de un Sistema de Bases de Datos para controlar un Inventario Informático



Estudiante: Alicia Orbis Espada

Consultor: Jordi Ferrer Duran

Enero - 2016



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/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	Diseño e Implementación de un Sistema de Bases de Datos para controlar un Inventario Informático
Nombre del autor:	Alicia Orbis Espada
Nombre del consultor:	Jordi Ferrer Duran
Fecha de entrega (mm/aaaa)	01/2016
Área del Trabajo Final	Bases de Datos Relacionales
Titulación	Grado de Ingeniería Informática

Resumen del Trabajo

El presente Trabajo de Fin de Grado (TFG), tiene como objetivo el diseño e implementación de una base de datos para controlar el inventario de los diferentes dispositivos informáticos de una empresa. Esto debe traducirse, por un lado, en el mejor control del material asignado a los distintos usuarios y, por otro, en el control financiero de las inversiones hechas en material informático por los diferentes departamentos de la empresa.

En términos generales, la base de datos, ha de permitir guardar todos los datos necesarios de los dispositivos informáticos, de los departamentos de la empresa, de los pedidos y líneas de pedido de compra, de los presupuestos, de los usuarios que utilizarán los dispositivos, de los movimientos de dispositivos y de los inventarios físicos que se realicen. Aparte de esto, el sistema también debe proporcionar diversa información estadística.

Para alcanzar este objetivo, se debe realizar un diseño de la base de datos (Diagrama E/R), implementar los script de creación de tablas, índices, disparadores (triggers) y procedimientos almacenados necesarios para realizar las tareas descritas por el enunciado, así como, implementar un mecanismo de iniciación de la base de datos y creación de un juego de pruebas exhaustivo que permita garantizar y demostrar el buen funcionamiento de las funciones implementadas.

Abstract (in English)

The final degree work (TFG), has the purpose of explaining the design and implementation of a database to control de inventory of the different computer devices of an enterprise. This provides, on one side, the correct control of the assigned material to the different users and, on the other, the financial control on the investments made in computer equipment by the different company's departments.

In general terms, the database has to allow saving all the necessary data of the computer devices, the departments of the enterprise, the orders and purchase order lines, the budgets, the users that will use the devices, the movement of the devices and the physical inventories that has been made. Apart from this, the system must provide varied statistic information.

To achieve this objective, is necessary to make a design of the database (E/R Diagram), to implement the scripts and table creation, indexes, triggers and the stored procedures needed to make the tasks descibed in the statement. Also, to implement a mechanism to iniciate the database and to create a thorough test kit that guarantees and proves the correct functioning of the implemented functions.

Palabras clave (entre 4 y 8):

Planificación, Análisis, Diseño, Procedimientos, Disparadores, Inventario

Índice

1.	Introducción	8
1.1	Justificación del TFG y contexto en el que se desarrolla.....	8
1.2	Objetivos del Trabajo	8
1.2.1	Objetivos Generales del TFG	8
1.2.2	Objetivos Específicos del TFG.....	9
1.3	Enfoque y Método seguido	9
1.4	Planificación del Trabajo	10
1.4.1	Calendario de Fechas	11
1.4.2	Identificación de Tareas	11
1.4.3	Planificación Temporal	13
1.4.4	Diagrama de Gantt	14
1.4.5	Seguimiento de la Planificación	14
1.5	Productos Obtenidos.....	16
1.6	Recursos Empleados.....	17
1.7	Análisis de Riesgos.....	18
1.8	Breve descripción del resto de capítulos	19
2.	Análisis del Sistema	19
2.1	Descripción Inicial.....	19
2.2	Análisis de Requisitos	20
2.2.1	Requisitos Funcionales	20
2.2.2	Requisitos no Funcionales.....	23
3.	Diseño del Sistema	24
3.1	Diseño Conceptual	24
3.1.1	Diagrama UML.....	25
3.1.2	Entidades.....	25
3.1.3	Atributos de las Entidades.....	27
3.1.4	Identificación de las Relaciones	29
3.2	Diseño Lógico	30
3.2.1	Diagrama Lógico-Relacional	33

3.3 Diseño Físico.....	34
3.3.1 Creación de la Base de Datos	34
3.3.2 Tablespaces	35
3.3.3 Usuarios.....	35
3.3.4 Tablas	36
3.3.5 Índices	41
3.3.6 Controles de Integridad.....	41
3.3.7 Repositorio Estadístico.....	42
3.3.8 Tabla de Logs.....	45
4. Implementación	46
4.1 Creación de Tablespaces	46
4.2 Creación de Usuarios.....	46
4.3 Creación de Secuencias	47
4.4 Creación de Tablas	48
4.5 Creación de Índices	48
4.6 Construcción de Procedimientos de Alta, Baja y Modificación	49
4.7 Construcción de Disparadores para el Módulo Estadístico	64
5. Pruebas.....	67
5.1 Carga Inicial de Datos	67
5.2 Secuencia de Pruebas.....	67
5.2.1 Pruebas de Alta, Baja y Modificación.....	67
5.2.2 Pruebas de funcionamiento del Módulo Estadístico	70
5.2.3 Pruebas de Consulta del Módulo Estadístico	70
5.2.4 Comprobación de Logs.....	72
5.3 Script de Pruebas.....	72
5.3.1 Código del Script	72
5.3.2 Resultados Esperados	73
5.3.3 Ejecución de Scripts.....	74
6. Valoración Económica del Proyecto.....	75
7. Conclusiones	75

8. Glosario de Términos	76
9. Bibliografía	77
10. Anexos	79

1. Introducción

1.1 Justificación del TFG y contexto en el que se desarrolla

El trabajo final de grado es un ejercicio de síntesis de los conocimientos adquiridos en varias de las asignaturas de la carrera y que requiere ponerlos en práctica conjuntamente en un proyecto concreto. Normalmente consiste en un trabajo de aplicación práctica vinculado totalmente al ejercicio de la informática.

En este caso concreto, en el área de Bases de Datos relacionales, el objetivo principal es consolidar los conocimientos que se han adquirido a través de asignaturas como Bases de Datos I, Bases de Datos II, Sistemas de Gestión de Bases de Datos e incluso de otras asignaturas como Ingeniería del Software.

El TFG consiste en un ejemplo práctico de la realización de un diseño de base de datos relacional, a partir de unos requisitos iniciales proporcionados por el cliente. El alumno tendrá que proponer un diseño que se ajuste a los requerimientos expuestos e implementar un sistema que encapsule las funciones de acceso a los datos.

De cara al alumno, este proyecto le permitirá poner en práctica los conocimientos adquiridos en asignaturas de Bases de Datos y ampliarlos en la medida de lo posible. Además, podrá adquirir habilidades de planificación, gestión, toma de decisiones y resolución de problemas que le serán de gran utilidad en su vida profesional.

1.2 Objetivos del Trabajo

1.2.1 Objetivos Generales del TFG

El objetivo general de este TFG es el de proporcionar a una empresa de software, el diseño e implementación de una base de datos para controlar el inventario de los diferentes dispositivos informáticos en una empresa. Esto se traducirá, por un lado, en el mejor control del material asignado a los distintos usuarios de la empresa y, por otro lado, en el control financiero de las inversiones hechas en material informático por los diferentes departamentos de la empresa.

El producto que debemos entregar deberá cumplir todos los requisitos especificados por el cliente. Nuestro trabajo consistirá en analizar la problemática planteada y definir una posible estructura de Base de Datos que de soporte a una futura aplicación de control de inventario.

El modelo ha de permitir guardar todos los datos necesarios de los dispositivos informáticos, de los departamentos de la empresa donde se instale la aplicación, de los pedidos y líneas de pedido de compra, de los presupuestos de compra, de los usuarios de la empresa que utilizarán los dispositivos, de los movimientos de dispositivos y de los inventarios físicos que se realicen. Aparte de esto, el sistema deberá proporcionar diversa información estadística.

Además, será necesaria la creación de un repositorio estadístico para dar una respuesta rápida y actualizada a una serie de consultas que propone el enunciado.

Para alcanzar el objetivo propuesto, cumpliendo las restricciones y tareas, se realizará lo siguiente:

- Diseño de la base de datos (Diagrama E/R). Implementación de los script de creación de las tablas, índices, disparadores (triggers) y procedimientos almacenados necesarios para poder realizar las tareas descritas en el enunciado.
- Implementación de un mecanismo de inicialización de la base de datos y creación de un juego de pruebas exhaustivo que permita garantizar y demostrar el buen funcionamiento de las funciones implementadas así como el control de errores y de situaciones de excepción.

A lo largo del ciclo de vida del Proyecto, se realizarán diferentes entregas:

- **PEC1:** Plan de trabajo. Debe recoger la planificación y estimación de las tareas necesarias para llevar a cabo los objetivos previstos.
- **PEC2:** Se corresponde con las fases de Análisis y Diseño.
- **PEC3:** Se entrega todo lo correspondiente a la fase de implementación (tablas, índices, secuencias, disparadores, procedimientos almacenados) así como el juego de pruebas.
- **Memoria:** es el documento que sintetizará el trabajo realizado y mostrará que se han alcanzado los objetivos propuestos. Incorporará toda la información relevante para comprender el problema planteado, la metodología utilizada para su resolución y detallará la solución elaborada.
- **Presentación:** Es un resumen claro y conciso del trabajo realizado y de los resultados obtenidos.

Para poder cumplir estos objetivos es necesario mantener una comunicación fluida con el cliente para poder aclarar todas las dudas que surjan, con el fin de que el producto resultante cubra todas las necesidades que se habían planteado.

1.2.2 Objetivos Específicos del TFG

A nivel específico, el objetivo de este Trabajo Final de Grado, es consolidar los conocimientos adquiridos a lo largo de los estudios de Ingeniería Informática con respecto a las asignaturas de Bases de Datos como son: Bases de Datos I (BDI), Bases de Datos II (BDII) y Sistemas de Gestión de Bases de Datos (SGBD).

1.3 Enfoque y Método seguido

Debido a la naturaleza del TFG la metodología seguida durante el ciclo de vida del proyecto es la que conocemos como “Ciclo de Desarrollo en Cascada”.

El primer paso para afrontar el proyecto es realizar un análisis inicial de los requerimientos recogidos en el enunciado y elaborar un plan realista y ajustado que permita alcanzar en plazo, alcance y calidad los objetivos establecidos.

Para conseguir una planificación lo más realista posible se ha estimado conveniente utilizar el Ciclo de Vida en Cascada, el cual es útil en aquellos proyectos cuya duración deba ser acotada y sea posible definir claramente unos requisitos iniciales que no vayan cambiando demasiado a lo largo de la duración del proyecto. De este modo, podemos planificar desde el inicio del proyecto cual será la duración de cada una de las fases, podemos planificar la asignación de recursos para ajustarse a esa planificación y se puede calcular el coste total del proyecto.

No se utilizará el ciclo de vida en cascada clásico, que no permite volver atrás una vez superada una etapa, sino una modificación sobre este modelo que consiste en la introducción de una revisión y vuelta atrás.

Las fases que se prevén en nuestro proyecto concreto son:

- Análisis de requisitos
- Diseño del sistema
- Implementación
- Pruebas

En los procesos de desarrollo de los productos previstos en el trabajo se utilizarán los métodos, técnicas y herramientas aprendidas en las diferentes asignaturas del grado y que resultan más adecuadas para la obtención de cada uno de los productos a desarrollar:

- La planificación se ha realizado de acuerdo a los métodos obtenidos en la asignatura Gestión de Proyectos.
- Se han aplicado los métodos y técnicas de diseño de bases de datos aprendidos en las asignaturas de Bases de Datos (diagramas de Entidad/Relación, diseño lógico, normalización de tablas, etc.) para el desarrollo de la base de datos y sus scripts correspondientes.
- Los métodos de desarrollo aprendidos en Bases de Datos I y II y Sistemas de Gestión de Bases de Datos se han utilizado para desarrollar los procedimientos almacenados y los disparadores necesarios para el buen funcionamiento de la base de datos.

1.4 Planificación del Trabajo

El proyecto se ha organizado atendiendo a las fechas propuestas por la UOC para la entrega de las distintas pruebas de evaluación continua. En principio se ha considerado una carga de trabajo de entre 10 y 15 horas semanales, pudiendo variar en función de una semana u otra.

Para realizar la planificación, en primer lugar, se han identificado los objetivos previstos, el alcance de los entregables y los hitos principales del proyecto. En segundo lugar se ha elegido el ciclo de vida en cascada y por último se ha realizado

una descomposición estructural de actividades adecuada, detallando el contenido y alcance de cada una de estas.

Una vez identificadas las tareas se ha realizado una planificación temporal, mediante la elaboración de una tabla de precedencias entre actividades y estimando la duración de estas, basándonos en los conocimientos adquiridos a lo largo de la carrera.

1.4.1 Calendario de Fechas

Las fechas claves de realización del TFG son las siguientes:

Fecha	Tarea	Descripción
16/09/2015	Inicio del semestre	Fecha de comienzo del Proyecto
05/10/2015	PEC1	Fecha prevista de entrega de la PEC1 (Plan de Trabajo)
09/11/2015	PEC2	Fecha prevista de entrega de la PEC2
10/12/2015	PEC3	Fecha prevista de entrega de la PEC3
11/01/2016	Entrega Final	Fecha prevista de entrega de la Memoria, la Presentación y el Producto.
25/01/2016	Inicio debate virtual	Fecha de inicio del debate virtual
27/01/2016	Fin debate virtual	Fecha de fin del debate virtual

1.4.2 Identificación de Tareas

Las tareas identificadas para las que se realiza la planificación temporal son las siguientes:

1. **Descripción preliminar del TFG:** Se realiza una comprensión de la problemática y de las funcionalidades deseadas, las cuales se reflejan en el documento del Plan de Trabajo.
2. **Definición de objetivos:** se fijan los objetivos a realizar en las siguientes entregas PEC1, PEC2, PEC3 y entrega final.
3. **Identificación de tareas:** se identifican las principales tareas a realizar a lo largo del ciclo de vida del proyecto.
4. **Planificación temporal de tareas:** una vez se han identificado las tareas se les asigna un tiempo estimado para cada una de ellas, teniendo en cuenta las fechas de entrega previstas y el peso que tenga cada una dentro del proyecto.
5. **Análisis de riesgos:** Se describirá un pequeño plan de contingencia para hacer frente a los posibles imprevistos.
6. **Elaboración de la PEC1:** elaboración del documento a entregar para la PEC1.
7. **Análisis de requisitos:** estudio de los diferentes requisitos que solicita el cliente. Habrá que reconocer y analizar con detalle las expectativas, las necesidades y los objetivos que tendrán los futuros usuarios de la base de datos.

8. **Diseño del sistema:** en base a los requisitos anteriores se realiza el diseño conceptual, lógico y físico de la Base de Datos. Para realizar el diseño conceptual se utilizarán los diagramas de clases del UML(Lenguaje Unificado de Modelado). Este diseño conceptual se convertirá en un diseño lógico, que consistirá en un modelo relacional donde se obtendrá un conjunto de relaciones con los atributos, las claves primarias y las claves foráneas correspondientes. Por último se transformará el diseño lógico en un modelo físico, que nos permitirá crear tablas, restricciones, índices, secuencias, disparadores, procedimientos, etc.
9. **Instalación y configuración del SGBD Oracle:** Se realiza la instalación de este software y se crea la base de datos con la que vamos a trabajar.
10. **Elaboración de la PEC2:** documento a entregar en la PEC2 que recoge la información referente a las fases de Análisis y Diseño.
11. **Incorporación de la PEC2 al proyecto:** Con el fin de ir avanzando en la elaboración de la memoria, se incorporan a esta los aspectos trabajados en la PEC2.
12. **Implementación:** creación de todo el código necesario para la implementación de la base de datos: disparadores, procedimientos almacenados, etc.
13. **Pruebas internas:** diseño, creación y ejecución del juego de pruebas que permitirá verificar el correcto funcionamiento del sistema.
14. **Depuración:** corrección de los diversos errores aparecidos en las pruebas del sistema. Una vez corregidos los errores se volverán a efectuar las pruebas hasta que el resultado cumpla con los requerimientos del sistema.
15. **Elaboración de la PEC3:** documento a entregar en la PEC3, que recogerá la información correspondiente a la fase de implementación y de pruebas internas (también incluirá las correcciones tras las pruebas).
16. **Incorporación de la PEC3 al proyecto:** Con el fin de ir avanzando en la elaboración de la memoria, se incorporan a esta los aspectos trabajados en la PEC3.
17. **Fase de documentación:** se elabora la Memoria Final del proyecto, así como la Presentación.

En este punto finalizaría la entrega del proyecto.

A la fase de pruebas y correcciones se les asignará un tiempo estimado que se considere suficiente para cumplir los propósitos, teniendo en cuenta que esta estimación es muy difícil porque dependerá de los errores que haya que corregir y del tiempo de respuesta del cliente.

1.4.3 Planificación Temporal

Para cada una de las PECs se han establecido una serie de tareas y para cada una de ellas se ha planificado el número de horas necesarias para llevarlas a cabo. En el cuadro siguiente se muestra esta planificación, incluyendo el número de horas estimadas.

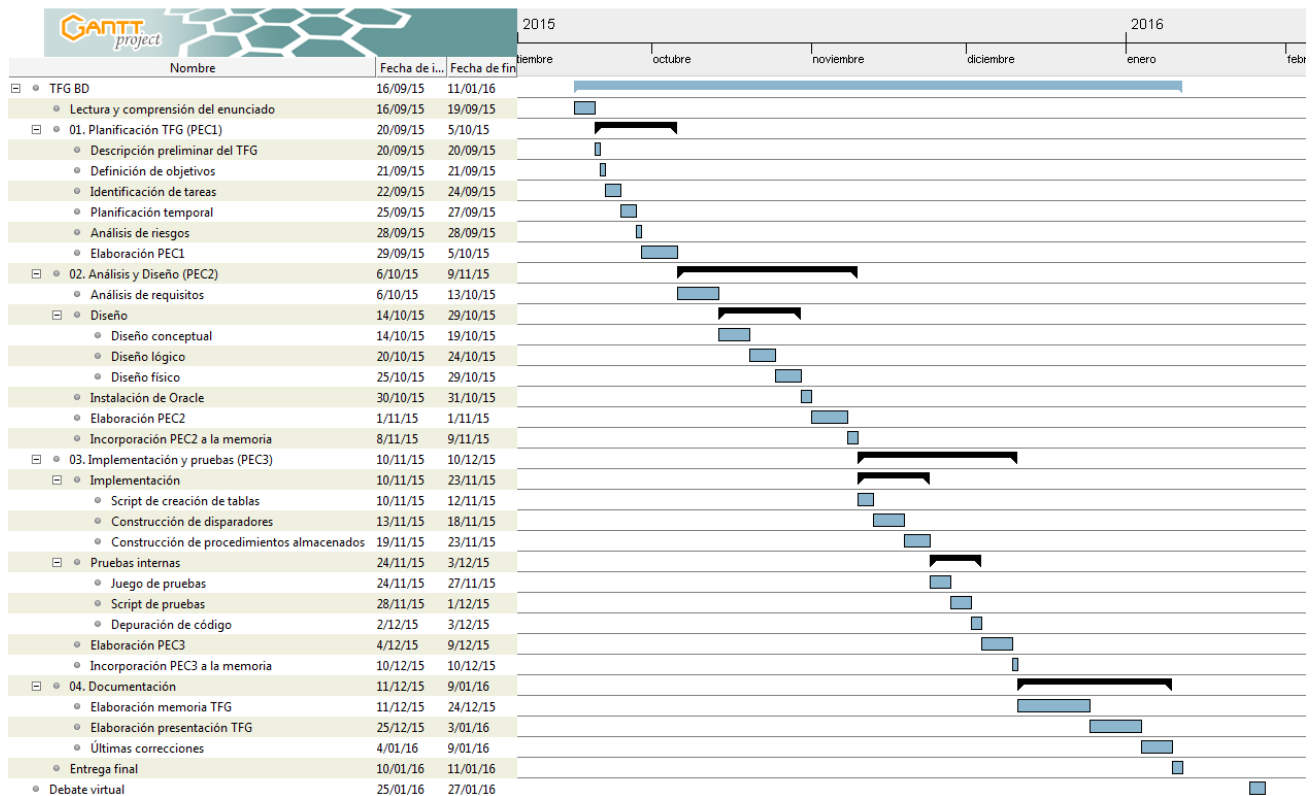
Nombre de Tarea	Tiempo estimado (horas)	Comienzo	Fin
TFG BD		16/09/2015	11/01/2016
Lectura y comprensión del enunciado	8	16/09/2015	19/09/2015
01- Planificación TFC (PEC1)		20/09/2015	05/10/2015
Descripción preliminar del TFG	2	20/09/2015	20/09/2015
Definición de objetivos	2	21/09/2015	21/09/2015
Identificación de tareas	5	22/09/2015	24/09/2015
Planificación temporal	6	25/09/2015	27/09/2015
Análisis de riesgos	2	28/09/2015	28/09/2015
Elaboración PEC1	5	29/09/2015	05/10/2015
02- Análisis y Diseño (PEC2)		06/10/2015	09/11/2015
Análisis de requisitos	12	06/10/2015	13/10/2015
Diseño		14/10/2015	29/10/2015
Diseño conceptual	11	14/10/2015	19/10/2015
Diseño lógico	10	20/10/2015	24/10/2015
Diseño físico	10	25/10/2015	29/10/2015
Instalación de Oracle	3	30/10/2015	31/10/2015
Elaboración PEC2	5	01/11/2015	07/11/2015
Incorporación PEC2 a la memoria	3	08/11/2015	09/11/2015
03- Implementación y pruebas (PEC3)		10/11/2015	10/12/2015
Implementación		10/11/2015	23/11/2015
Script de creación de tablas	6	10/11/2015	12/11/2015
Construcción de disparadores	14	13/11/2015	18/11/2015
Construcción de procedimientos almacenados	12	19/11/2015	23/11/2015
Pruebas internas		24/11/2015	03/12/2015
Juego de pruebas	10	24/11/2015	27/11/2015
Script de pruebas	10	28/11/2015	01/12/2015
Depuración de código	5	02/12/2015	03/12/2015
Elaboración PEC3	14	04/12/2015	09/12/2015
Incorporación PEC3 a la memoria	2	10/12/2015	10/12/2015
04- Documentación		11/12/2015	09/01/2016
Elaboración memoria TFG	35	11/12/2015	24/12/2015
Elaboración presentación TFG	20	25/12/2015	03/01/2016
Últimas correcciones	10	04/01/2016	09/01/2016
Entrega Final	5	10/01/2016	11/01/2016
Debate Virtual	5	25/01/2016	27/01/2016

Observaciones:

- La planificación es orientativa. A medida que vaya avanzando el proyecto habrá que hacer los reajustes necesarios, teniendo en cuenta las horas reales dedicadas.

- Ante la posibilidad de que haya retrasos en algunos puntos de la planificación, se ha contado con 7 días para la elaboración de las PECs. Esto permitirá que, aunque haya alguna tarea que sufra algún retraso, se pueda recuperar ese tiempo y se puedan efectuar las entregas dentro del plazo establecido.

1.4.4 Diagrama de Gantt



1.4.5 Seguimiento de la Planificación

En el caso de la planificación de la **PEC 1**, el seguimiento ha sido el correcto, ya que ha permitido dar por finalizado el hito el día marcado y no se han producido grandes desviaciones en las horas dedicadas a cada una de las tareas.

En la fase de Análisis y Diseño (PEC2), la planificación ha sido la adecuada ya que la consecución de los objetivos finales se ha alcanzado sin dificultad, permitiendo realizar la entrega en la fecha prevista. En principio, las horas previstas para la tarea del diseño lógico fueron algo inferiores de las inicialmente previstas, pero, después de algunas recomendaciones del consultor en referencia a las tablas del repositorio estadístico, se tuvo que modificar tanto el diseño físico como el diseño lógico, por lo que las horas que se habían realizado de menos, tuvieron que utilizarse para las modificaciones pertinentes.

También se ahorró algún tiempo en la instalación de Oracle, ya que se utilizó una versión ya instalada en el ordenador. Sin embargo hubo que hacer una limpieza de los datos anteriores con lo que al final el tiempo empleado en dejar el programa operativo, fue el que se había previsto en un principio.

En la fase de Implementación y Pruebas (PEC3), la planificación ha sido bastante ajustada a la planificación inicial, pero con algunas desviaciones. En principio, las horas previstas para la creación de tablas, construcción de disparadores y construcción de procedimientos almacenados han sido las inicialmente previstas. En el caso de las pruebas internas, ha habido una mayor desviación, sobre todo en lo que se refiere a la depuración del código, para lo que se han tenido que utilizar más horas de las inicialmente previstas.

La primera dificultad encontrada fue que, al no ser posible crear algunas tablas con campos calculados (esto solo es posible desde la versión 11g de Oracle), se ha tenido que adecuar el código de los disparadores para poder calcular el porcentaje en las tablas que así lo requieren.

La segunda dificultad, y la que ha causado un mayor retraso, ha sido el problema de las “tablas mutantes”. Este error se produce porque se hacen select contra tablas sobre las que se está haciendo insert, update o delete. Como, por los requerimientos del sistema era necesario hacer estas select, se han intentado algunas soluciones sin mucho éxito, como crear procedimientos y llamarlos desde los disparadores (esto producía el mismo error). También se intentó utilizar disparadores BEFORE en lugar de AFTER, pero esto solo funcionaba en el caso de los insert, y había que hacer algunas correcciones porque los datos que había en la tabla eran anteriores a la actualización.

En vista de la problemática, al final se ha optado por crear dos tablas auxiliares, una idéntica a la tabla movimientos (movimientos_aux) y otra idéntica a la tabla proveedores (proveedores_aux).

Las dos tablas auxiliares se actualizan al mismo tiempo que sus homólogas, por lo que siempre contienen los mismos datos. Esto ha permitido hacer las select en los disparadores utilizando las tablas auxiliares en lugar de las originales, lo que ha solucionado el problema de “tablas mutantes”.

También se ha detectado al realizar las pruebas de consultas, que con los datos de algunas tablas no podíamos responder, en tiempo constante, a la consulta planteada. Esto ocurre con las tablas Usuarios_Averias e Inventario_Departamento. Para dar solución a este problema, se han creado dos nuevas tablas, Usuarios_Top10_Averias (que contendrá el ranking de los 10 usuarios con mayor número de averías) y Depto_Mas_TiposDisp (que contendrá el departamento con más tipos distintos de dispositivos) y se ha tenido que ajustar el código de los disparadores para que vayan actualizando estas tablas.

Al realizar las pruebas, también se ha visto que la precisión de algunos campos numéricos no era la adecuada, por lo que se ha procedido a cambiarla. Es el caso de los importes del pedido, del presupuesto y los precios de compra de los dispositivos. También se ha cambiado el tipo de datos, para las tablas del repositorio estadístico que tenían que trabajar con años. En un principio se consideró el campo año como numérico, pero con la realización de las pruebas se ha visto que era más conveniente que fueran tipo varchar2(4).

Debido a estos imprevistos, se ha solicitado al consultor la posibilidad de hacer la entrega un día más tarde, y este ha dado su autorización.

1.5 Productos Obtenidos

Los entregables que compondrán este Trabajo Final de Grado son los siguientes:

- **Plan de Trabajo:** consiste en la planificación y estimación de las tareas necesarias para llevar a cabo los objetivos propuestos.
- **Producto:** corresponde a los desarrollos de la base de datos.
- **Memoria:** es una síntesis de todo el trabajo realizado y muestra que se han alcanzado los objetivos propuestos.
- **Presentación:** es un resumen claro y conciso, por medio de diapositivas, del trabajo realizado y de los resultados que se han obtenido.

Las entregas que se irán haciendo a lo largo del curso, que corresponderán a la evaluación continua y que se contemplan en la planificación son las siguientes:

Planificación TFG (PEC1)

En esta primera fase, se determinarán las tareas necesarias para llevar a cabo el Proyecto y se hará una estimación del esfuerzo y los recursos necesarios para llevarlas a cabo.

Análisis y Diseño de la BD (PEC2)

- Se evaluará, en primer lugar, si los requerimientos que nos proporciona el enunciado están lo suficientemente claros. Si no es así, debemos ponernos en contacto con el cliente para clarificar todo aquello que sea necesario.
- Se elaborará el diseño conceptual de la BD mediante diagramas E/R y su consecuente transformación al diseño lógico
- Se instalará y configurará el Sistema de Gestión de Bases de Datos Oracle en los ordenadores que se hayan asignado al proyecto para que estén disponibles antes de empezar con el desarrollo.
- Se elaborará el diseño físico de la BD de forma que optimice el rendimiento y asegure la integridad de los datos.

Implementación de la BD (PEC3)

- Se construirán todos los objetos de la BD, incluyendo las tablas, secuencias, disparadores, procedimientos almacenados y las funciones que sean necesarias.
- Se construirá un repositorio estadístico para dar una respuesta rápida y actualizada a un conjunto de preguntas predefinidas. Para ello se crearán un conjunto de tablas que darán una respuesta inmediata a cada una de las consultas que aparecen en el enunciado y que constituyen los requisitos de este repositorio.
- Se construirá un juego de datos para poder efectuar las pruebas que sean necesarias del código desarrollado.

Producto final + Memoria + Presentación

- El producto final contendrá todos los scripts necesarios para el funcionamiento del proyecto (creación de tablas, creación de procedimientos, pruebas, etc.)
- Memoria: debe sintetizar el trabajo realizado y mostrar que se han alcanzado los objetivos propuestos. Además, debe contener información relevante que ayude a entender el problema planteado por el TFG y la metodología que se ha empleado para su resolución.
- Presentación Virtual: será un documento claro y conciso que, por medio de diapositivas, mostrará una síntesis del trabajo realizado.

1.6 Recursos Empleados

La asignación de recursos de un proyecto consiste en determinar las personas y recursos necesarios para llevar a cabo el proyecto.

Recursos humanos

En nuestro caso concreto, existirán dos participantes en el proyecto y cada uno de ellos tendrá un rol determinado:

Jordi Ferrer (consultor): empresa cliente

Alicia Orbis (alumna): empresa de desarrollo

Hardware

Para este proyecto se utilizará el siguiente hardware:

Ordenador portátil Sony vaio Intel P8400 de doble núcleo (Dual Core) de 2.26 GHz, 4 Gb de memoria RAM y 500 Gb de HDD (Disco Duro) con sistema operativo Windows Vista Home Premium Service Pack 1.

Software

Para la realización del trabajo se utilizará diferente software. Por un lado, se utilizarán los programas necesarios para el desarrollo de la aplicación, pero además se tendrán que utilizar otros programas para realizar los distintos diagramas y para elaborar la documentación. En el siguiente cuadro se resumen los programas que van a utilizarse.

Software	Función
Oracle Express 10 g 2.0	Gestor de base de datos
Oracle SQL Developer 1.0.0.15	Desarrollo de software
MagicDraw Personal Edition 18.2	Diseño diagramas de clases
Ganttproject 2.7	Planificación del proyecto
Paquete Microsoft Office 2007	Elaboración de la documentación
Camtasia Studio 8.5.2	Elaboración de la presentación

Recursos

Durante las distintas fases de proyecto será necesario acceder a diferentes fuentes de información que servirán de apoyo en su desarrollo:

- Materiales didácticos de las asignaturas Bases de Datos I, Bases de Datos II y Sistemas de Gestión de Bases de Datos.
- Materiales didácticos de Ingeniería del Software.
- Materiales didácticos de Gestión de Proyectos.
- Material de Presentación de documentos y elaboración de presentaciones.
- Material de Redacción de textos científico-técnicos.
- Manuales y guías de PL/SQL y base de datos Oracle.

1.7 Análisis de Riesgos

En todos los proyectos, hay un paso muy importante que consiste en identificar los riesgos que pueden afectar al mismo. Será necesario documentar sus características y crear un “Registro de riesgos” que se irá completando a medida que se avanza en el proyecto.

Es muy difícil detectar cuales son los factores que pueden amenazar la capacidad de entrega de los objetivos que hemos propuesto, porque existe la incertidumbre de no saber qué puede pasar, pero, aunque hay aspectos que no se pueden predecir, debemos hacer todos los esfuerzos posibles para controlarlos.

Los principales problemas que pueden surgir durante la realización del proyecto y las medidas a tomar para intentar evitarlos o resolverlos se sintetizan en el cuadro siguiente:

Descripción	Impacto	Medidas
Averías: Hardware o software	Pérdida del entorno de trabajo	Ordenador de sobremesa preparado con el mismo entorno que el ordenador habitual.
	Pérdida de datos	Se realizan varias copias de seguridad. Una en el disco duro, otra en un disco duro externo y otra en una unidad USB.
Interrupción del servicio de internet.	Pérdida de comunicación con la UOC y pérdida de acceso a otras fuentes de información.	Posibilidad de utilizar el servicio de internet de algún familiar o amigo.
Viajes imprevistos o enfermedad	Disminución de las horas dedicadas en la planificación.	En las fechas de realización de cada una de las PECs se deja algún día más por si surgiera algún imprevisto. Si la situación fuera más grave se intentaría pactar con el cliente un cambio en la planificación.
Casos extremos	Disminución importante de	En estos casos, difíciles de prever,

	<p>horas dedicadas a la planificación, lo que conlleva tener que hacer cambios significativos en ella.</p>	<p>habría que buscar una solución conjunta entre cliente y desarrollador.</p>
--	--	---

1.8 Breve descripción del resto de capítulos

Los siguientes capítulos del TFG serán:

- **Análisis del sistema:** lo primero que tenemos que hacer para construir un sistema es averiguar qué es exactamente lo que tiene que hacer el sistema. La etapa de análisis corresponde al proceso mediante el cual se intenta descubrir qué es lo que realmente se necesita y se llega a una comprensión adecuada de los requerimientos del sistema.
- **Diseño del sistema:** a partir de los requisitos diseñaremos la base de datos necesaria para albergar toda la información y para cumplir todas las funcionalidades que nos piden. El diseño se hará en diferentes etapas: diseño conceptual, diseño lógico y diseño físico.
- **Implementación:** para la fase de implementación hemos de seleccionar las herramientas adecuadas, un entorno de desarrollo que facilite nuestro trabajo y un lenguaje de programación apropiado para el tipo de sistema que vayamos a construir. La elección de estas herramientas dependerá en gran parte de las decisiones de diseño que hayamos tomado hasta el momento y del entorno en el que nuestro sistema deberá funcionar.
- **Plan de pruebas:** Permite verificar que el sistema cumple las necesidades establecidas por el usuario, con las debidas garantías de calidad.
- **Conclusiones:** conclusiones del trabajo fin de grado.

2. Análisis del Sistema

2.1 Descripción Inicial

El trabajo propuesto consiste en desarrollar un sistema de Bases de Datos para dar respuesta a una necesidad planteada por una empresa de software, que quiere implementar una nueva aplicación para controlar el inventario de los diferentes dispositivos informáticos. Esta nueva aplicación se debe adaptar al procedimiento de gestión de activos informáticos (Asset Management) que la propia empresa ha desarrollado.

Los objetivos principales de la gestión de activos informáticos son, por un lado un mejor control del material asignado a los diferentes usuarios de la empresa y, por otro lado, el control financiero de las inversiones hechas en material informático por los diferentes departamentos de la empresa.

A nivel general, el sistema ha de permitir guardar todos los datos necesarios de los dispositivos informáticos, de los departamentos de la empresa donde se instale la aplicación, de los pedidos y líneas de pedido de compra, de los presupuestos de compra, de los usuarios de la empresa que utilizan los dispositivos, de los movimientos de los dispositivos y de los inventarios físicos que se realicen.

El sistema a diseñar debe permitir almacenar toda la información comentada anteriormente y permitir generar las consultas más habituales que se realizan. Adicionalmente a este funcionamiento, la base de datos deberá encargarse de pre calcular y almacenar diversa información estadística. Este repositorio estadístico dará respuesta a las consultas y ofrecerá los datos en tiempo constante 1.

2.2 Análisis de Requisitos

2.2.1 Requisitos Funcionales

Los requerimientos proporcionados por el enunciado se han analizado para la realización de las tareas de diseño y para asegurar el cumplimiento de todos sus puntos. En aquellos puntos en los que las reglas no quedaban claramente definidas o surgía una duda sobre su interpretación, se hicieron una serie de consultas al consultor, que actúa como nuestro cliente.

Los requisitos establecidos por el enunciado para la base de datos son los siguientes:

Se pide al estudiante que implemente un sistema de Base de Datos para dar respuesta a una necesidad planteada por una empresa de software que quiere implementar una nueva aplicación para controlar el inventario de los diferentes dispositivos informáticos.

- El sistema ha de permitir almacenar los datos financieros correspondientes al pedido y a la naturaleza de la compra, diferenciando si se trata de un gasto más o si la compra va ligada a una orden de inversión.
- Cuando el material solicitado llega al centro de destino, los encargados de realizar las recepciones deben ser los técnicos informáticos, o una persona nombrada como responsable.
- El sistema ha de prever el registro y control de quien realiza las recepciones y llevar un registro detallado de las acciones realizadas sobre todo el material informático.
- Una vez que se ha recibido el material informático pasa a estar disponible para ser asignado a los usuarios del centro. Solo las personas del departamento de informática pueden realizar cualquier movimiento sobre los diferentes dispositivos. La aplicación debe permitir llevar un registro exhaustivo de todos esos movimientos.
- La aplicación debe permitir gestionar la auditoría financiera que anualmente, y por normativa legal, han de realizar las empresas para validar que es correcta la inversión en material informático.

- Como la auditoría financiera valida que se haga regularmente un inventario físico de los dispositivos, para asegurar que realmente existen, el sistema debe permitir guardar la fecha de la última validación física de cada dispositivo.
- La aplicación ha de permitir guardar todos los datos necesarios de los dispositivos informáticos, de los departamentos de la empresa donde se instale la aplicación, de los pedidos y líneas de pedido de compra, de los presupuestos de compra, de los usuarios que utilizarán los dispositivos, de los movimientos de los dispositivos y de los inventarios físicos que se realicen.
- La aplicación deberá permitir la ejecución de consultas e informes de control que se consideren necesarios para la correcta gestión del inventario informático.
- Toda la gestión y acceso a la información se hará mediante procedimientos de base de datos, siendo esta la única manera de acceder.
- A nivel de procedimientos será necesario implementar y describir con detalle los procedimientos de ABM (Alta + Baja + Modificación) de todas las entidades que se consideren relevantes.
- Se creará un repositorio estadístico que tendrá que responder a las cuestiones siguientes:
 - ✓ Dado un año concreto, número total de dispositivos comprados.
 - ✓ Porcentaje de ordenadores portátiles que, en el momento de hacer la consulta, no han tenido ninguna avería desde su asignación a un usuario cualquiera.
 - ✓ Dado un año concreto, importe total de los pedidos realizados en ese año, ligados a una orden de inversión.
 - ✓ Proveedor que, en el momento de hacer la consulta, tiene un periodo de entrega más corto contando desde el envío del pedido hasta la recepción del material.
 - ✓ Dado un determinado departamento, número de dispositivos en uso que tiene el departamento en el momento de la consulta para cada tipo de dispositivo activo. Se considera activo un tipo de dispositivo que, como mínimo, lo utiliza un usuario de la empresa.
 - ✓ Para cada departamento de la empresa, valor económico del inventario activo en el momento de hacer la consulta. Para este cálculo, y para simplificar, se considerará el valor de compra de cada dispositivo.
 - ✓ Número de dispositivos que en el momento de hacer la consulta, tienen una fecha de inventario físico mayor de un año.
 - ✓ Departamento que, en un año concreto, efectúa más movimientos de tipo reasignación de dispositivo.
 - ✓ Porcentaje de usuarios de la empresa que, en el momento de hacer la consulta, no tienen ningún dispositivo asignado.

- ✓ Para cada tipo de dispositivo, número de unidades que, en el momento de hacer la consulta, están en el almacén pendientes de ser asignadas.
- ✓ Departamento de la empresa que, en el momento de hacer la consulta, tiene más tipos diferentes de dispositivos informáticos. Es necesario saber el departamento y cuantos dispositivos diferentes tiene en ese momento.
- ✓ Número de usuarios que, en el momento de hacer la consulta, tienen asignados más de 5 dispositivos informáticos.
- ✓ Número de averías registradas en un año concreto en toda la empresa.
- ✓ Top10 de usuarios por número de averías registradas sobre sus dispositivos.

Como consecuencia de la revisión de los requerimientos del sistema, se identificaron una serie de puntos abiertos que fueron enviados al consultor para su aclaración, tal como hubiéramos hecho con un cliente. En la tabla siguiente se resume la descripción de estos puntos y su resolución:

Descripción	Resolución
¿Hay que considerar que con cada petición de un departamento se crea un nuevo pedido?	Considera que se crea un pedido por cada petición.
Los presupuestos, ¿Están ligados a un pedido concreto o se pide un presupuesto de varios pedidos a la vez?	Cada pedido tiene un presupuesto.
¿Sería interesante crear una tabla que recogiera los datos de las recepciones de los pedidos?	Si, es una buena aportación.
¿Existe la posibilidad de que un dispositivo (tipo impresora o fotocopiadora) se asigne a varios usuarios?	Considera que un dispositivo lo tiene asignado un solo usuario.
¿En qué consiste la actualización de un dispositivo?	Se considera actualización cuando se hace una revisión y se actualiza la fecha de inventario.
¿La reasignación de un dispositivo consiste en el cambio de usuario?	Si, es el cambio de usuario.
En el caso de dar de baja un dispositivo, ¿se trata de una baja lógica, que en ningún caso supone el borrado del dispositivo de la tabla?	Exacto.
En la tabla líneas de pedido, ¿el campo código de material se refiere a tipo de dispositivo?	Si, es así.
¿Es necesario realizar la gestión de la auditoría financiera?	El control de las auditorías lo hará la aplicación que use esa BD. Debes limitarte a crear los campos necesarios.
¿Habría que calcular el tiempo que tarda un proveedor en servir un pedido o hay que incluir el dato en la tabla proveedores?	Hay que incluir en la tabla el tiempo que tarda en media.
En la tabla de presupuestos, ¿se guardarán solo los presupuestos que se acepten o para cada pedido podrá haber varios presupuestos?	Puede haber más de un presupuesto, pero sólo uno aceptado.
En las consultas del repositorio estadístico, si no existe ningún elemento que cumpla los requisitos, ¿Habría que mostrarlos con 0?	Si
En el top 10 de usuarios por número de averías registradas sobre sus dispositivos. Si hubiera empates en el número de averías, ¿habría que sacar más de 10 registros?	Si
En cuanto a las sedes que pueda tener una empresa. ¿Debería crear una tabla que las recoja?	Yo creo que sí, es la mejor opción

2.2.2 Requisitos no Funcionales

En este apartado se desarrollarán todos los requisitos que no describen información a guardar, ni funciones a realizar, pero especifican criterios que pueden usarse para juzgar la operación de un sistema.

De la lectura del enunciado se desprenden los siguientes requisitos no funcionales:

- **Sistema de gestión de base de datos:** será Oracle.
- **Sistema operativo:** no se desarrollará en ningún sistema operativo concreto, ya que Oracle puede ser instalado en cualquier sistema operativo y el enunciado no impone nada al respecto.
- **Política de backups:** No se contempla la realización de tareas periódicas de backup de los datos almacenados en la Base de Datos. El proyecto consiste en el desarrollo de un sistema para el almacenamiento de dicha información. Este apartado debería tratarse una vez realizada la implementación.
- **Los procedimientos** dispondrán como mínimo de un parámetro de salida llamado RSP, de tipo String, que indicará si la ejecución ha finalizado con éxito (“OK”) o si ha fracasado (“ERROR+ TIPO DE ERROR”). Además, dispondrán de tratamiento de excepciones.
- **Carga del sistema:** Se prevé una carga del sistema que puede ser bastante alta. Como la aplicación debe servir para cualquier empresa, sea del tamaño que sea, es muy importante que la gestión de los datos almacenados se haga siguiendo las técnicas que se aplican a grandes volúmenes de datos (Data Warehouse).
- **Rendimiento:**
 1. Concurrencia: El sistema de información que se plantea, podrá responder de manera simultánea a las peticiones realizadas por las distintas aplicaciones de alto nivel que estén haciendo uso de él.
 2. Tiempo de respuesta: Para conseguir un buen tiempo de respuesta en los accesos a la BD, se aplicará el sistema de optimización sobre las consultas. También se añadirán los índices necesarios. Las respuestas del módulo estadístico tienen que ser inmediatas y tienen que estar siempre actualizadas con la última información de la BD.
 3. Disponibilidad y fiabilidad: El sistema debe estar operativo las 24 horas del día, con los mínimos riesgos de caída o fallo (estos requisitos no se considerarán en el desarrollo del proyecto).
 4. Seguridad: Se debe mantener la confidencialidad de los datos estableciendo un control de acceso a los mismos. Para ello se deben establecer mecanismos de acceso a los objetos de la base de datos aplicando permisos mediante los diferentes perfiles de usuarios (este requisito no se considerará en el desarrollo del proyecto).

3. Diseño del Sistema

El diseño del sistema se abordará en tres etapas:

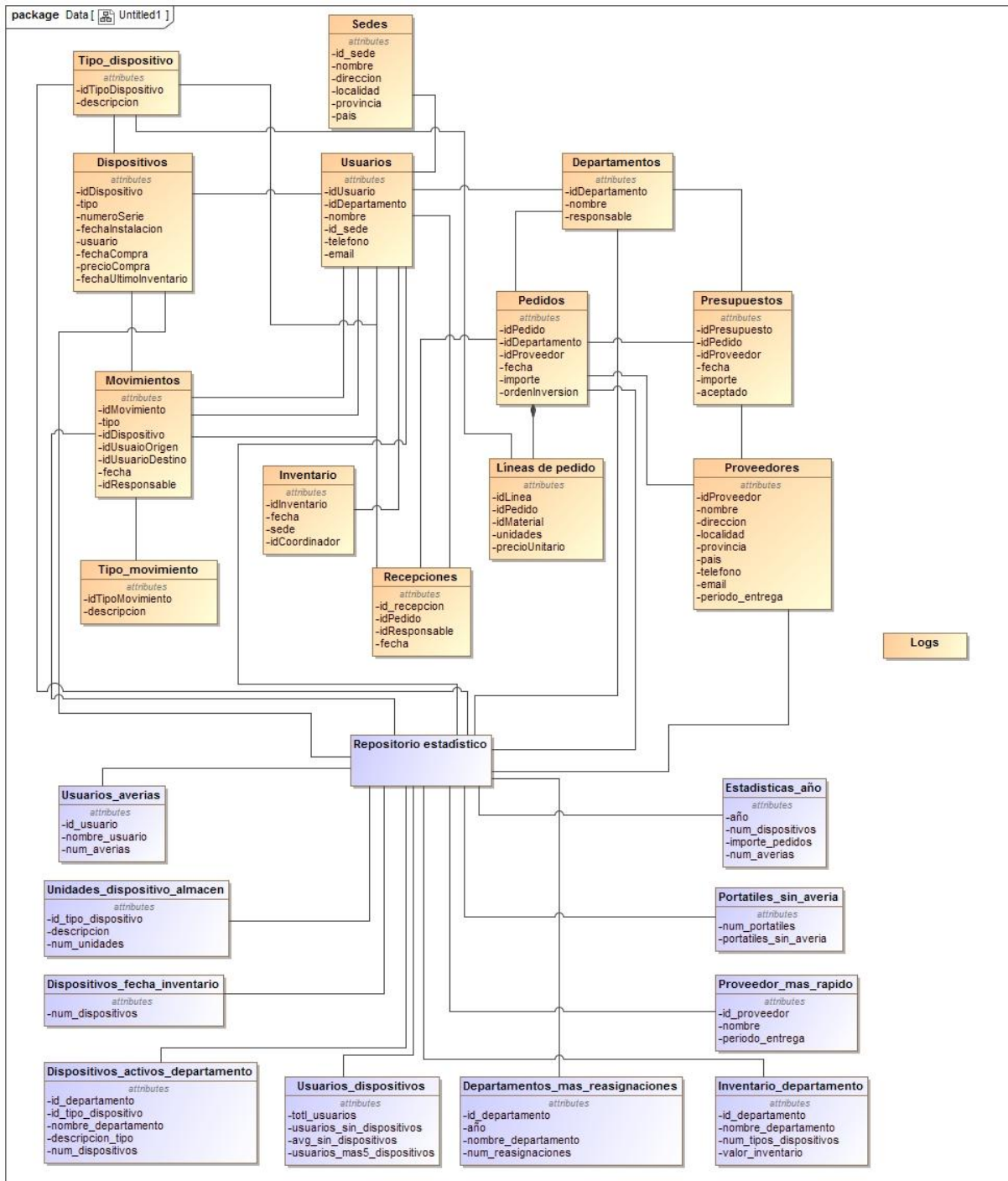
- **Diseño conceptual**: Se buscará un diseño independiente de la tecnología que se vaya a utilizar. Como resultado de esta etapa se obtendrá un modelo UML.
- **Diseño lógico**: Se transformará el modelo conceptual en el modelo de datos elegido (modelo relacional).
- **Diseño físico**: Es una fase del proceso de diseño de bases de datos que adapta el esquema lógico obtenido en la fase anterior al SGBD concreto que utilizará el sistema de información.

3.1 Diseño Conceptual

Con la ayuda del análisis de requisitos obtenido anteriormente, se puede elaborar el esquema conceptual en el que se describe el contenido de la información de la base de datos.

Para el diseño conceptual se utilizará el modelo Entidad/Relación debido a su simplicidad y a que facilita el diseño de la base de datos. Los principales elementos que incluye este modelo son las entidades, los atributos y las relaciones entre entidades.

3.1.1 Diagrama UML



3.1.2 Entidades

Una entidad es un objeto del mundo real, que tiene identidad propia y que es distinguible del resto de los objetos. Pueden ser objetos con existencia física u objetos con existencia conceptual.

Entidad	Descripción
DISPOSITIVOS	Tendrá los datos de todos los dispositivos dados de alta en el sistema. Es una de las entidades más importantes. En ella se relacionan los dispositivos con los usuarios a los que se han asignado. También se reflejan, entre otros datos, su fecha de instalación, su fecha de compra, su precio y la fecha del último inventario.
DEPARTAMENTOS	Tendrá datos de todos los departamentos de la empresa.
USUARIOS	Tendrá datos de todos los usuarios de la empresa. Se indicará el departamento al que pertenecen y la sede en la que se encuentran. En grandes empresas puede ocurrir que los trabajadores se encuentren en distintas sedes.
PEDIDOS	Tendrá los datos de los pedidos, indicando la fecha, el departamento que lo ha solicitado, el proveedor, el importe total del pedido y si va ligado a una orden de inversión o no.
LINEAS_PEDIDO	Tendrá los distintos tipos de dispositivos que van en el pedido, indicando para cada uno de ellos el número de unidades y su precio unitario.
PRESUPUESTOS	Tendrá datos de los presupuestos de cada proveedor para cada pedido. Se indicará el proveedor, el pedido, la fecha, el importe total y si ha sido aceptado o no.
RECEPCIONES	Se detallarán los datos de recepción de los pedidos, indicando el responsable y la fecha de entrega por parte del proveedor.
MOVIMIENTOS	En esta entidad se detallan los distintos movimientos (asignación, alta, baja, reasignación, averías) sobre los distintos dispositivos. Se indica el tipo de movimiento, el usuario origen y destino, la fecha y el responsable.
INVENTARIOS	Tendrá los datos del inventario, indicando la fecha, la sede donde se lleva a cabo y el identificativo del coordinador.

Además de estas entidades, que serán las principales, deben existir otras entidades auxiliares que vincularemos a las demás:

Entidad	Descripción
TIPO_MOVIMIENTO	En esta entidad se recogerán los distintos tipos de movimientos sobre los dispositivos (alta, baja, asignación, reparación de averías, etc).
TIPO_DISPOSITIVO	Se recogerán los distintos tipos de dispositivos que existen en la empresa (pc, impresoras, portátiles, etc).
PROVEEDORES	Tendrá los datos de los proveedores que trabajan con la empresa.
SEDES	Recogerá los datos de las distintas sedes que tenga la empresa.
LOGS	Almacenará los resultados de las distintas ejecuciones de los procedimientos almacenados.

También se considerarán entidades auxiliares a las que constituyen el **repositorio estadístico**:

Entidad	Descripción
ESTADÍSTICAS_AÑO	Reflejará las distintas estadísticas que se produzcan en cada año, en concreto, el número de dispositivos comprados, el importe de los pedidos o el número de averías que se producen.
PORTATILES_SIN_AVERIA	Almacenará el número de portátiles que tiene la empresa en cada momento y el número de portátiles que no han sufrido ninguna avería.
PROVEEDOR_MAS_RAPIDO	Almacenará los datos del proveedor que tiene un periodo de entrega más corto contando desde el envío del pedido hasta la recepción del material.
DISPOSITIVOS_ACTIVOS_DEPARTAMENTO	Almacenará para cada departamento y cada tipo de dispositivo, el número de dispositivos.
DISPOSITIVOS_FECHA_INVENTARIO	Almacenará el número de dispositivos que tienen una fecha de

	inventario mayor de un año.
DEPARTAMENTO_MAS_REASIGNACIONES	Para cada departamento y cada año se almacenarán el número de movimientos de reasignación que se han hecho.
USUARIOS_DISPOSITIVOS	Almacenará el total de usuarios, el número de usuarios con más de 5 dispositivos y el número de usuarios sin ningún dispositivo asignado. .
UNIDADES_DISPOSITIVO_ALMACEN	Se almacenarán los distintos tipos de dispositivos que hay en la empresa y para cada uno de ellos se indicará el número de unidades que se encuentran en el almacén.
USUARIOS_AVERIAS	Almacenará los datos de los usuarios que tienen mayor número de averías en sus dispositivos.

3.1.3 Atributos de las Entidades

Un atributo es una propiedad que tienen todas las entidades de un mismo tipo y que permite representar sus características.

En algunas de las entidades se han relacionado los atributos *fecha_alta* y *fecha_baja* para dar la posibilidad de que se puedan hacer consultas a nivel histórico. El atributo *fecha_baja* se rellenará cuando se dé una baja en esa entidad.

En algunas entidades del repositorio estadístico, se han utilizado atributos derivados, es decir, que se calculan a partir de otros atributos y que por lo tanto llevarán asociado un procedimiento de cálculo.

A continuación se detallan los atributos que se han definido para las distintas entidades. Se han subrayado los que componen las claves primarias.

TIPO_MOVIMIENTO

id_tipo_movimiento, descripción, fecha_alta, fecha_baja

TIPO_DISPOSITIVO

id_tipo_dispositivo, descripción, fecha_alta, fecha_baja

PROVEEDORES

id_proveedor, nombre, dirección, localidad, provincia, país, teléfono, email, periodo_entrega, fecha_alta, fecha_baja

SEDES

id_sede, nombre_sede, dirección_sede, localidad_sede, provincia_sede, país_sede, fecha_alta, fecha_baja

DEPARTAMENTOS

id_departamento, nombre_departamento, id_responsable, fecha_alta, fecha_baja

DISPOSITIVOS

id_dispositivo, tipo, numero_serie, fecha_instalacion, id_usuario, fecha_compra, precio_compra, fecha_ultimo_inventario, fecha_alta, fecha_baja

USUARIOS

id_usuario, id_departamento, nombre, teléfono, email, id_sede, fecha_alta, fecha_baja

PEDIDOS

id_pedido, id_proveedor, id_departamento, fecha_pedido, importe_pedido, orden_inversion

LINEAS_PEDIDO

id_pedido, id_linea, id_material, unidades, precio_unitario

PRESUPUESTOS

id_presupuesto, id_proveedor, id_pedido, fecha_presupuesto, importe, aceptado

RECEPCIONES

id_recepcion, id_pedido, id_responsable, fecha_recepcion, observaciones

MOVIMIENTOS

id_movimiento, id_tipo_movimiento, id_dispositivo, id_usuario_origen, id_usuario_destino, fecha_movimiento, id_responsable

INVENTARIOS

id_inventario, fecha_inventario, id_sede, id_coordinador

Entidades del Repositorio Estadístico

ESTADISTICAS_AÑO

año, num_dispositivos, importe_pedidos, num_averias

PORTATILES_SIN_AVERIA

num_portatiles, portátiles_sin_averia, avg_sin_averia

PROVEEDOR_MAS_RAPIDO

id_proveedor, nombre, periodo_entrega

DISPOSITIVOS_ACTIVOS_DEPARTAMENTO

id_departamento, id_tipo_dispositivo, nombre_departamento, descripción_tipo, num_dispositivos

INVENTARIO_DEPARTAMENTO

id_departamento, nombre_departamento, num_tipos_dispositivos, valor_inventario

DISPOSITIVOS_FECHA_INVENTARIO

num_dispositivos

DEPARTAMENTO_MAS_REASIGNACIONES

id_departamento, año, nombre_departamento, num_reasignaciones

USUARIOS_DISPOSITIVOS

total_usuarios, usuarios_sin_dispositivos, usuarios_mas5_dispositivos, avg_sin_dispositivos

UNIDADES_DISPOSITIVO_EN_ALMACEN

id_tipo_dispositivo, descripción, num_unidades

USUARIOS_AVERIAS

id_usuario, nombre_usuario, num_averias

LOGS

Id_log, fecha, usuario, procedimiento, paramsentrada, paramssalida

3.1.4 Identificación de las Relaciones

Entidad	Relación	Descripción	Entidad	Cardinalidad
USUARIOS	Tienen asignados	Un usuario puede tener varios dispositivos asignados. Un dispositivo solo se asigna a un usuario.	DISPOSITIVOS	1 : N
USUARIOS	Son responsables	Un usuario puede ser responsable de realizar varias recepciones de pedidos.	RECEPCIONES	1 : N
USUARIOS	Realizan	Un usuario podrá realizar varios movimientos con los dispositivos. Cada movimiento solo lo realizará un usuario.	MOVIMIENTOS	1 : N
USUARIOS	Coordinan	Los usuarios serán responsables de realizar el inventario informático. Un usuario podrá ser el coordinador de varios inventarios. Un inventario será coordinado por un usuario.	INVENTARIO	1 : N
TIPO_DISPOSITIVO	Pertenecen	Cada tipo de dispositivo tendrá varios dispositivos que pertenezcan a ese tipo. Un dispositivo solo podrá pertenecer a un tipo de dispositivo.	DISPOSITIVOS	1 : N
DISPOSITIVOS	Se realizan	Sobre cada dispositivo se realizarán distintos movimientos (alta, baja, asignación, etc.) Un movimiento solo se refiere a un dispositivo.	MOVIMIENTOS	1 : N
DEPARTAMENTOS	Están destinados	En un departamento pueden estar destinados varios usuarios. Un usuario estará destinado en un departamento.	USUARIOS	1 : N
DEPARTAMENTOS	Gestionan	El departamento financiero es el que se encarga de gestionar los diferentes presupuestos.	PRESUPUESTOS	1 : N
DEPARTAMENTOS	Realizan	Un departamento podrá	PEDIDOS	1 : N

		realizar varias peticiones de material. Un pedido será realizado por un único departamento.		
PROVEEDORES	Sirven	Un proveedor puede servir varios pedidos. Un pedido lo sirve un único proveedor.	PEDIDOS	1 : N
PROVEEDORES	Proporcionan	Un proveedor puede proporcionar varios presupuestos para distintos pedidos. Cada presupuesto será proporcionado por un único proveedor.	PRESUPUESTOS	1 : N
PEDIDOS	Se componen	Un pedido puede estar compuesto por varias líneas de pedido.	LINEAS_PEDIDO	1 : N
TIPO_DISPOSITIVO	Está incluido	Un tipo de dispositivo puede estar incluido en varias líneas de pedido. Una línea de pedido solo tendrá un tipo de dispositivo.	LINEAS_PEDIDO	1 : N
PEDIDOS	Se reciben	Un pedido se recoge en una recepción.	RECEPCIONES	1 : 1
PEDIDOS	Tienen	Un pedido puede tener varios presupuestos, aunque solo uno de ellos debe ser aceptado. Cada presupuesto corresponde a un único pedido.	PRESUPUESTOS	1 : N
SEDES	Trabajan	En cada sede pueden trabajar varios usuarios. Un usuario trabajará en una sede.	USUARIOS	1 : N
TIPO_MOVIMIENTO	Pertenece	Cada tipo de movimiento podrá pertenecer a varios movimientos. Cada movimiento solo pertenecerá a un tipo de movimiento.	MOVIMIENTOS	1 : N

3.2 Diseño Lógico

Una vez terminado el diseño conceptual, es decir, definido el esquema entidad-relación, el siguiente paso es realizar el diseño lógico de la BD.

Los elementos básicos del modelo ER son las entidades y las interrelaciones:

- Las entidades, cuando se traducen al modelo relacional, originan relaciones.
- Las interrelaciones, en cambio, cuando se transforman, pueden dar lugar a claves foráneas de alguna relación ya obtenida o pueden dar lugar a una nueva relación

En las relaciones 1: 1 sólo será necesario añadir a cualquiera de las dos entidades una clave foránea que referencie a la otra relación.

En las relaciones 1: N es necesario añadir en la relación correspondiente a la entidad del lado N, una clave foránea que referencie la otra relación.

Si una de las entidades es opcional en la interrelación, y la transformación ha consistido en poner una clave foránea en la relación que corresponde a la otra entidad, entonces esta clave foránea puede tomar valores nulos.

A continuación se detallan los atributos de las relaciones en las que se traducen las entidades e interrelaciones identificadas en el diseño conceptual y las referencias entre relaciones por medio de claves foráneas. Los atributos que forman parte de la clave primaria aparecen subrayados.

TIPO_MOVIMIENTO

id_tipo_movimiento, descripción, fecha_alta, fecha_baja

TIPO_DISPOSITIVO

id_tipo_dispositivo, descripción, fecha_alta, fecha_baja

PROVEEDORES

id_proveedor, nombre, dirección, localidad, provincia, país, teléfono, email, periodo_entrega, fecha_alta, fecha_baja

SEDES

id_sede, nombre_sede, dirección_sede, localidad_sede, provincia_sede, país_sede, fecha_alta, fecha_baja

DEPARTAMENTOS

id_departamento, nombre_departamento, id_responsable, fecha_alta, fecha_baja

donde [id_responsable] es clave foránea y referencia USUARIOS

DISPOSITIVOS

id_dispositivo, tipo, numero_serie, fecha_instalacion, id_usuario, fecha_compra, precio_compra, fecha_ultimo_inventario, fecha_alta, fecha_baja

donde [tipo] es clave foránea y referencia TIPO_DISPOSITIVO y

[id_usuario] es clave foránea y referencia USUARIOS

USUARIOS

id_usuario, id_departamento, nombre, teléfono, email, id_sede, fecha_alta, fecha_baja

donde [id_departamento] es clave foránea y referencia DEPARTAMENTOS y

[id_sede] es clave foránea y referencia SEDES

PEDIDOS

id_pedido, id_proveedor, id_departamento, fecha_pedido, importe_pedido, orden_inversion

donde [id_proveedor] es clave foránea y referencia PROVEEDORES y

[id_departamento] es clave foránea y referencia DEPARTAMENTOS

LINEAS_PEDIDO

id_pedido, id_linea, id_material, unidades, precio_unitario

donde [id_pedido] es clave foránea y referencia PEDIDOS y

[id_material] es clave foránea y referencia TIPO_DISPOSITIVO

PRESUPUESTOS

id_presupuesto, id_proveedor, id_pedido, fecha_presupuesto, importe, aceptado

donde [id_proveedor] es clave foránea y referencia PROVEEDORES y

[id_pedido] es clave foránea y referencia PEDIDOS

RECEPCIONES

id_recepcion, id_pedido, id_responsable, fecha_recepcion, observaciones

donde [id_pedido] es clave foránea y referencia PEDIDOS y

[id_responsable] es clave foránea y referencia USUARIOS

MOVIMIENTOS

id_movimiento, id_tipo_movimiento, id_dispositivo, id_usuario_origen, id_usuario_destino,
fecha_movimiento, id_responsable

donde [id_tipo_movimiento] es clave foránea y referencia TIPO_MOVIMIENTO,

[id_dispositivo] es clave foránea y referencia DISPOSITIVOS,

[id_usuario_origen] es clave foránea y referencia USUARIOS,

[id_usuario_destino] es clave foránea y referencia USUARIOS y

[id_responsable] es clave foránea y referencia USUARIOS

INVENTARIOS

id_inventario, fecha_inventario, id_sede, id_coordinador

donde [id_coordinador] es clave foránea y referencia USUARIOS Y

[id_sede] es clave foránea y referencia SEDES

Repositorio Estadístico

ESTADISTICAS_AÑO

año, num_dispositivos, importe_pedidos, num_averias

PORTATILES_SIN_AVERIA

num_portatiles, portatiles_sin_averia, avg_sin_averia

PROVEEDOR_MAS_RAPIDO

id_proveedor, nombre, periodo_entrega

donde [id_proveedor] es clave foránea y referencia PROVEEDORES

DISPOSITIVOS_ACTIVOS_DEPARTAMENTO

Id_departamento, id_tipo_dispositivo, nombre_departamento, descripción_tipo,
num_dispositivos

donde [id_departamento] es clave foránea y referencia DEPARTAMENTOS y
[id_tipo_dispositivo] es clave foránea y referencia TIPO_DISPOSITIVO

INVENTARIO_DEPARTAMENTO

Id_departamento, nombre_departamento, num_tipos_dispositivo, valor_inventario

donde [id_departamento] es clave foránea y referencia DEPARTAMENTOS

DISPOSITIVOS_FECHA_INVENTARIO

num_dispositivos

DEPARTAMENTO_MAS_REASIGNACIONES

Id_departamento, nombre_departamento, año, num_reasignaciones

donde [id_departamento] es clave foránea y referencia DEPARTAMENTOS

USUARIOS_DISPOSITIVOS

total_usuarios, usuarios_sin_dispositivos, avg_sin_dispositivos, usuarios_mas5_dispositivos

UNIDADES_DISPOSITIVO_ALMACEN

id_tipo_dispositivo, descripción, num_unidades

donde [id_tipo_dispositivo] es clave foránea y referencia TIPO_DISPOSITIVO

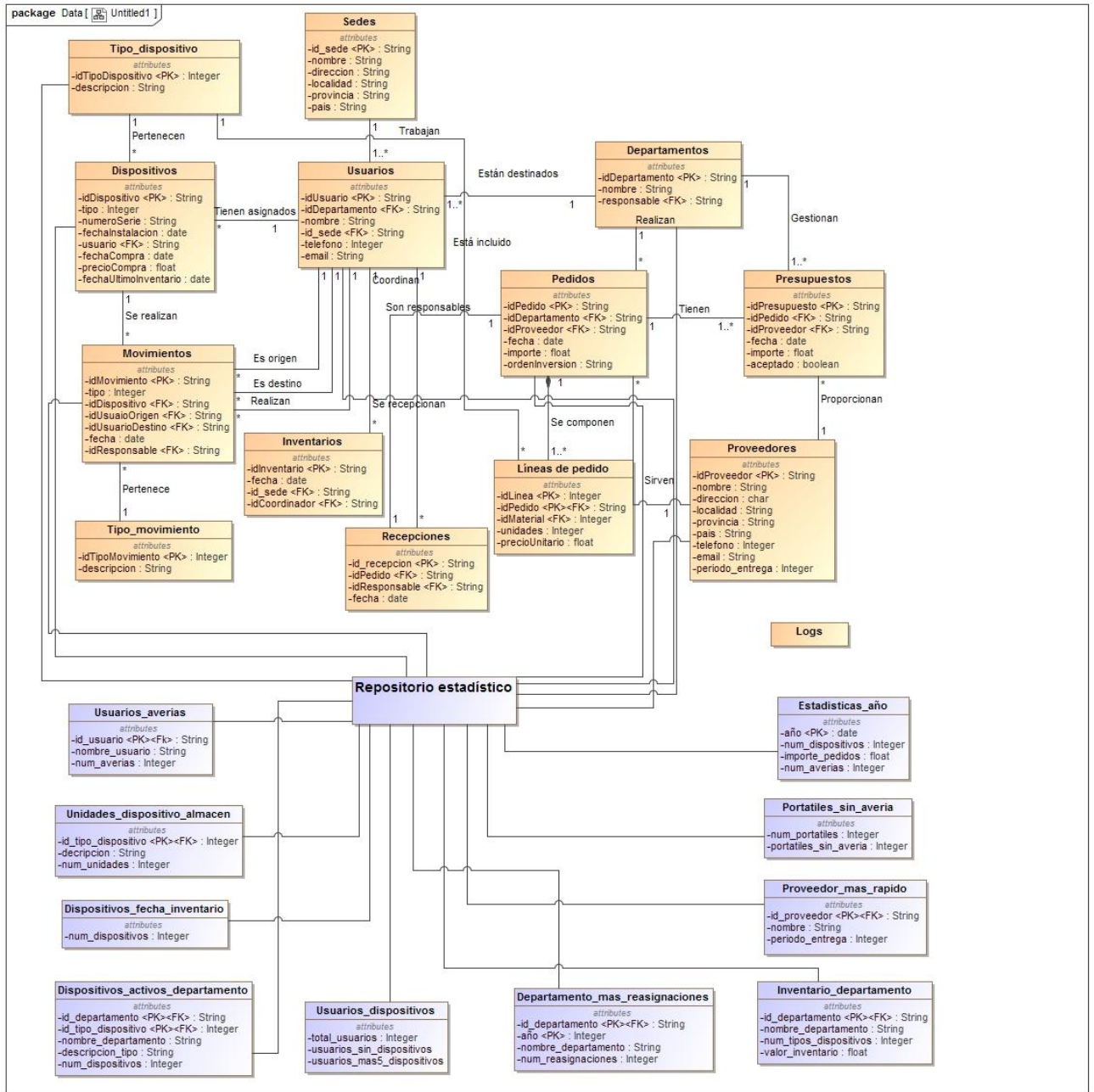
USUARIOS_AVERIAS

id_usuario, nombre_usuario, num_averias

donde [id_usuario] es clave foránea y referencia USUARIOS

3.2.1 Diagrama Lógico-Relacional

Al igual que ocurría con el diagrama UML, las tablas del repositorio estadístico se han dibujado de color azul para una mejor visualización. A diferencia del diagrama UML, en este diagrama aparecen las claves primarias y foráneas de cada entidad, además de las relaciones entre ellas.



3.3 Diseño Físico

En esta fase se trata de plasmar el diseño lógico en tablas de la base de datos. Para ello se deben seguir los siguientes pasos:

3.3.1 Creación de la Base de Datos

En primer lugar se ha procedido a la instalación de Oracle Database 10 g Express Edition. Al realizar la instalación, existe la posibilidad de crear directamente la base de datos o hacerlo después.

Para la creación de la base de datos se debe hacer lo siguiente:

- Definir y configurar las variables de entorno (ORACLE_SID, ORACLE_HOME).
- Crear un archivo init.ora. que contenga una lista de parámetros que Oracle lee cuando una base de datos es arrancada. Los parámetros identifican a la base de datos, ficheros de control (control files), tamaño de la SGA, recursos de la Base de Datos, ubicación de los ficheros de diagnóstico y estadísticas, etc. El init.ora es imprescindible para que nuestra Base Datos pueda arrancar, cada Base de Datos tiene su propio init.ora que tendrá como formato de nombre init más el nombre de la Base de datos más .ora y por defecto Oracle lo buscará en el arranque en el directorio ORACLE_HOME/database.
- Iniciar la instancia de Oracle.

A los efectos del TFG, se considera que las empresas en las que habrá que implantar el modelo de datos dispondrán de un servidor que ya tiene instalado el software del SGBD Oracle y que ya está creada la base de datos en la que hay que implementar el modelo de datos, por lo que solo habrá que crear aquellos objetos (tablespaces, usuario, tablas, etc) que sean necesarios para el modelo a implantar.

3.3.2 Tablespaces

Un tablespace es una unidad lógica de almacenamiento dentro de una base de datos Oracle. Cada tablespace se compone de al menos un Data File y un Data File solo puede corresponder a un tablespace.

Cada vez que se crea una tabla o un índice debe hacerse en un tablespace

En Oracle, existen varios tipos de tablespaces:

SYSTEM: se crea automáticamente al hacer la instalación de Oracle o al crear la base de datos. Es el encargado de almacenar el diccionario de datos (tablas con información sobre la propia BD)

TEMP: solamente puede almacenar objetos temporales

ROLLBACK o De tipo deshacer cambios: se utiliza para poder deshacer las operaciones incompletas.

DATA: almacena datos de la aplicación

Para realizar la implementación en el SGBD será necesario crear dos tablespaces donde se almacenará, por un lado, un espacio de trabajo para los índices y otro espacio de trabajo para los datos (tablas, secuencias, triggers, funciones ...)

Se generarán 2 “tablespaces”, uno para los datos y otro para los índices (en la fase de implementación se verán las sentencias de creación).

3.3.3 Usuarios

Según la descripción del trabajo a realizar, habrá determinados usuarios que serán los únicos que tengan acceso a determinadas operaciones. Por ejemplo, los usuarios del departamento financiero serán los encargados de gestionar los presupuestos y de crear los pedidos de compra necesarios y los técnicos informáticos serán los

encargados de realizar los movimientos sobre los diferentes dispositivos y las recepciones.

Como la gestión y acceso a la información debe hacerse por medio de procedimientos, y no es necesario implementar una interface de usuario, se dejará que sea la aplicación que gestione la base de datos, la que se ocupe de administrar esos permisos. Dependiendo de la singularidad de cada empresa, se podrán crear roles de usuario que otorgarán a cada uno de ellos los permisos necesarios sobre determinados objetos.

Por nuestra parte, se creará un único usuario que tendrá acceso a las diferentes tablas y funcionalidades de la aplicación. Este usuario será propietario de todos los objetos de la base de datos (tablas, índices, procedimientos, etc.) y tendrá los privilegios suficientes para crear todos los objetos necesarios y acceder a los tablespaces.

3.3.4 Tablas

A partir de lo que se ha obtenido en el diseño lógico, se podrán crear las tablas con sus correspondientes claves primarias y foráneas.

El tipo y tamaño de los campos se ha escogido en función de cada caso particular. Se han marcado como NOT NULL los campos que es imprescindible rellenar siempre para que el sistema funcione correctamente.

A continuación se detallan las tablas que se crearán en la base de datos.

TIPO_MOVIMIENTO			
Campos	Tipo	Nulo	Descripción
ID_TIPO_MOVIMIENTO	NUMBER	NOT NULL	Identificador del tipo de movimiento
DESCRIPCION	VARCHAR2(40)	NOT NULL	Descripción del tipo de movimiento
FECHA_ALTA	DATE	NOT NULL	Fecha de alta en el sistema
FECHA_BAJA	DATE		Fecha de baja en el sistema
Restricciones	Tipo	Tabla Referenciada	Campos
PK_TIPO_MOVIMIENTO	Primary Key		ID_TIPO_MOVIMIENTO

TIPO_DISPOSITIVO			
Campos	Tipo	Nulo	Descripción
ID_TIPO_DISPOSITIVO	NUMBER	NOT NULL	Identificador del tipo de dispositivo
DESCRIPCION	VARCHAR2(30)	NOT NULL	Descripción del tipo de dispositivo
FECHA_ALTA	DATE	NOT NULL	Fecha de alta en el sistema
FECHA_BAJA	DATE		Fecha de baja en el sistema
Restricciones	Tipo	Tabla Referenciada	Campos
PK_TIPO_DISPOSITIVO	Primary Key		ID_TIPO_DISPOSITIVO

PROVEEDORES			
Campos	Tipo	Nulo	Descripción
ID_PROVEEDOR	VARCHAR2(6)	NOT NULL	Identificador del proveedor
NOMBRE	VARCHAR2(60)	NOT NULL	Nombre del proveedor
DIRECCION	VARCHAR2(40)		Dirección del proveedor
LOCALIDAD	VARCHAR2(30)		Localidad del proveedor
PROVINCIA	VARCHAR2(30)		Provincia del proveedor
PAIS	VARCHAR2(30)		País del proveedor
TELEFONO	NUMBER(9)		Teléfono del proveedor
EMAIL	VARCHAR2(30)		E-mail del proveedor
PERIODO_ENTREGA	NUMBER	NOT NULL	Periodo de entrega (en días)
FECHA_ALTA	DATE	NOT NULL	Fecha de alta en el sistema
FECHA_BAJA	DATE		Fecha de baja en el sistema
Restricciones	Tipo	Tabla Referenciada	Campos
PK_PROVEEDORES	Primary Key		ID_PROVEEDOR

SEDES			
Campos	Tipo	Nulo	Descripción
ID_SEDE	VARCHAR2(3)	NOT NULL	Identificador de la sede
NOMBRE_SEDE	VARCHAR2(60)	NOT NULL	Nombre de la sede
DIRECCION_SEDE	VARCHAR2(40)		Dirección donde se encuentra la sede
LOCALIDAD_SEDE	VARCHAR2(30)		Localidad donde se encuentra la sede
PROVINCIA_SEDE	VARCHAR2(30)		Provincia donde se encuentra la sede
PAIS_SEDE	VARCHAR2(30)		País donde se encuentra la sede
FECHA_ALTA	DATE	NOT NULL	Fecha de alta en el sistema
FECHA_BAJA	DATE		Fecha de baja en el sistema
Restricciones	Tipo	Tabla Referenciada	Campos
PK_SEDE	Primary Key		ID_SEDE

DEPARTAMENTOS			
Campos	Tipo	Nulo	Descripción
ID_DEPARTAMENTO	VARCHAR2(3)	NOT NULL	Identificador del departamento
NOMBRE_DEPARTAMENTO	VARCHAR2(60)	NOT NULL	Nombre del departamento
ID_RESPONSABLE	VARCHAR2(6)		Identificador del responsable del departamento
FECHA_ALTA	DATE	NOT NULL	Fecha de alta en el sistema
FECHA_BAJA	DATE		Fecha de baja en el sistema
Restricciones	Tipo	Tabla Referenciada	Campos
PK_DEPARTAMENTOS	Primary Key		ID_DEPARTAMENTO
FK_DEPARTAMENTO_RESPONSABLE	Foreign Key	USUARIOS	ID_USUARIO

Aclaración: El campo id_responsable puede ser nulo, porque en determinadas circunstancias puede haber algún departamento que no tenga ningún responsable asignado.

DISPOSITIVOS			
Campos	Tipo	Nulo	Descripción
ID_DISPOSITIVO	VARCHAR2(6)	NOT NULL	Identificador del dispositivo
TIPO	NUMBER	NOT NULL	Identificador del tipo de dispositivo
NUMERO_SERIE	VARCHAR2(15)	NOT NULL	Número de serie del dispositivo
FECHA_INSTALACION	DATE		Fecha de instalación
ID_USUARIO	VARCHAR2(6)		Identificador del usuario asignado
FECHA_COMPRA	DATE	NOT NULL	Fecha de la compra del dispositivo
PRECIO_COMPRA	NUMBER(8,2)	NOT NULL	Precio de compra del dispositivo
FECHA_ULTIMO_INVENTARIO	DATE		Fecha del último inventario
FECHA_ALTA	DATE	NOT NULL	Fecha de alta en el sistema
FECHA_BAJA	DATE		Fecha de baja en el sistema
Restricciones	Tipo	Tabla Referenciada	Campos
PK_DISPOSITIVOS	Primary Key		ID_DISPOSITIVO
FK_DISPOSITIVO_TIPO	Foreign Key	TIPO_DISPOSITIVO	ID_TIPO_DISPOSITIVO
FK_DISPOSITIVO_USUARIO	Foreign Key	USUARIOS	ID_USUARIO

Aclaración: Los dispositivos se darán de alta en esta tabla una vez que se den de alta en la tabla de movimientos y es posible que no tengan ningún usuario asignado. Por ello se permite que los campos fecha_instalación e id_usuario sean nulos.

USUARIOS			
Campos	Tipo	Nulo	Descripción
ID_USUARIO	VARCHAR2(6)	NOT NULL	Identificador del usuario
ID_DEPARTAMENTO	VARCHAR2(3)	NOT NULL	Identificador del departamento del usuario
NOMBRE	VARCHAR2(60)	NOT NULL	Nombre del usuario
TELEFONO	NUMBER(9)		Teléfono del usuario
EMAIL	VARCHAR2(30)		Correo electrónico del usuario
ID_SEDE	VARCHAR2(3)	NOT NULL	Sede donde trabaja el usuario
FECHA_ALTA	DATE	NOT NULL	Fecha de alta en el sistema
FECHA_BAJA	DATE		Fecha de baja en el sistema
Restricciones	Tipo	Tabla Referenciada	Campos
PK_USUARIOS	Primary Key		ID_USUARIO
FK_USUARIO_DEPARTAMENTO	Foreign Key	DEPARTAMENTOS	ID_DEPARTAMENTO
FK_USUARIOS_SEDE	Foreign Key	SEDES	ID_SEDE

PEDIDOS			
Campos	Tipo	Nulo	Descripción
ID_PEDIDO	VARCHAR2(6)	NOT NULL	Identificador del pedido
ID_PROVEEDOR	VARCHAR2(6)	NOT NULL	Identificador del proveedor
ID_DEPARTAMENTO	VARCHAR2(3)	NOT NULL	Identificador del departamento que solicita el material
FECHA_PEDIDO	DATE	NOT NULL	Fecha del pedido
IMPORTE_PEDIDO	NUMBER(10,2)	NOT NULL	Importe del pedido
ORDEN_INVERSION	VARCHAR2(20)		Orden de inversión

Restricciones	Tipo	Tabla Referenciada	Campos
PK_PEDIDOS	Primary Key		ID_PEDIDO
FK_PEDIDO_PROVEEDOR	Foreing Key	PROVEEDORES	ID_PROVEEDOR
FK_PEDIDO_DEPARTAMENTO	Foreing Key	DEPARTAMENTOS	ID_DEPARTAMENTO

Aclaración: Cuando un pedido vaya ligado a una orden de inversión, el campo orden_inversión se rellenará con la orden. En caso contrario, este campo será nulo.

LINEAS_PEDIDO			
Campos	Tipo	Nulo	Descripción
ID_LINEA	NUMBER	NOT NULL	Identificador de línea de pedido
ID_PEDIDO	VARCHAR2(6)	NOT NULL	Identificador de pedido
ID_MATERIAL	NUMBER	NOT NULL	Identificador del material del pedido
UNIDADES	NUMBER	NOT NULL	Unidades de cada dispositivo
PRECIO_UNITARIO	NUMBER(8,2)	NOT NULL	Precio unitario de cada dispositivo
Restricciones	Tipo	Tabla Referenciada	Campos
PK_LINEASPEDIDO	Primary Key		ID_LINEA, ID_PEDIDO
FK_LINEASPEDIDO_PEDIDO	Foreing Key	PEDIDOS	ID_PEDIDO
FK_LINEASPEDIDO_MATERIAL	Foreing Key	TIPO_DISPOSITIVO	ID_TIPO_DISPOSITVO

PRESUPUESTOS			
Campos	Tipo	Nulo	Descripción
ID_PRESUPUESTO	VARCHAR2(6)	NOT NULL	Identificador del presupuesto
ID_PROVEEDOR	VARCHAR2(6)	NOT NULL	Identificador del proveedor
ID_PEDIDO	VARCHAR2(6)	NOT NULL	Identificador del pedido
FECHA_PRESUPUESTO	DATE	NOT NULL	Fecha del presupuesto
IMPORTE	NUMBER(10,2)	NOT NULL	Importe del presupuesto
ACEPTADO	CHAR		Indica si el presupuesto ha sido aceptado
Restricciones	Tipo	Tabla Referenciada	Campos
PK_PRESUPUESTOS	Primary Key		ID_PRESUPUESTO
FK_PRESUPUESTO_PROVEEDOR	Foreing Key	PROVEEDORES	ID_PROVEEDOR
FK_PRESUPUESTO_PEDIDO	Foreing Key	PEDIDOS	ID_PEDIDO
CK_PRESUPUESTO_ACEPT	Check		ACEPTADO IN ('N','S')

Aclaración: El campo aceptado se rellenará con 'N' si el presupuesto no es aceptado y con 'S' cuando el presupuesto sea aceptado. No puede haber varios presupuestos para un mismo pedido que estén aceptados. Esta situación tendrá que controlarse mediante un procedimiento.

RECEPCIONES			
Campos	Tipo	Nulo	Descripción

ID_RECEPCION	VARCHAR2(6)	NOT NULL	Identificador de la recepción de material
ID_PEDIDO	VARCHAR2(6)	NOT NULL	Identificador de pedido
ID_RESPONSABLE	VARCHAR2(6)	NOT NULL	Identificador de la persona que recoge el material
FECHA_RECEPCION	DATE	NOT NULL	Fecha de recepción del material
OBSERVACIONES	VARCHAR(100)		Incidencias que puedan producirse en la recepción del material
Restricciones	Tipo	Tabla Referenciada	Campos
PK_RECEPCIONES	Primary Key		ID_RECEPCION
FK_RECEPCION_PEDIDO	Foreing Key	PEDIDOS	ID_PEDIDO
FK_RECEPCION_RESPONSABLE	Foreing Key	USUARIOS	ID_USUARIO

MOVIMIENTOS			
Campos	Tipo	Nulo	Descripción
ID_MOVIMIENTO	VARCHAR2(6)	NOT NULL	Identificador del movimiento
ID_TIPO_MOVIMIENTO	NUMBER	NOT NULL	Identificador del tipo de movimiento
ID_DISPOSITIVO	VARCHAR2(6)	NOT NULL	Identificador de dispositivo
ID_USUARIO_ORIGEN	VARCHAR2(6)		Identificador del usuario origen
ID_USUARIO_DESTINO	VARCHAR2(6)		Identificador del usuario destino
FECHA_MOVIMIENTO	DATE	NOT NULL	Fecha en que se produce el movimiento
ID_RESPONSABLE	VARCHAR2(6)	NOT NULL	Identificador del usuario responsable
Restricciones	Tipo	Tabla Referenciada	Campos
PK_MOVIMIENTOS	Primary Key		ID_MOVIMIENTO
FK_MOVIMIENTO_TIPO	Foreing Key	TIPO_MOVIMIENTO	ID_TIPO_MOVIMIENTO
FK_MOVIMIENTO_DISPOSITIVO	Foreing Key	DISPOSITIVOS	ID_DISPOSITIVO
FK_MOVIMIENTO_USUARIOORIGEN	Foreing Key	USUARIOS	ID_USUARIO
FK_MOVIMIENTO_USUARIODESTINO	Foreing Key	USUARIOS	ID_USUARIO
FK_MOVIMIENTO_USUARIORESP	Foreing Key	USUARIOS	ID_USUARIO

INVENTARIOS			
Campos	Tipo	Nulo	Descripción
ID_INVENTARIO	VARCHAR2(5)	NOT NULL	Identificador de inventario
FECHA_INVENTARIO	DATE	NOT NULL	Fecha de inventario
ID_SEDE	VARCHAR2(3)	NOT NULL	Sede donde se realiza el inventario
ID_COORDINADOR	VARCHAR2(6)	NOT NULL	Coordinador encargado del inventario
Restricciones	Tipo	Tabla Referenciada	Campos
PK_INVENTARIOS	Primary Key		ID_INVENTARIO
FK_INVENTARIO_SEDE	Foreing Key	SEDES	ID_SEDE
FK_INVENTARIO_USUARIO	Foreing key	USUARIOS	ID_USUARIO

3.3.5 Índices

Un índice de una base de datos es una estructura de datos que mejora la velocidad en las operaciones, permitiendo un rápido acceso a los registros de una tabla. Como aumentan drásticamente la velocidad de acceso, se suelen usar sobre aquellos campos sobre los que se van a realizar búsquedas frecuentes.

Cuando se definen las claves primarias de las tablas, Oracle crea un índice único y no nulo (UNIQUE, NOT NULL) asociado a esa clave primaria. Aparte de estos, se crearán los siguientes:

- Se creará un índice por cada clave foránea para evitar problemas de rendimiento, concurrencias y bloqueos. Interesará tener índices de todos los campos que hacemos servir para relacionar tablas entre sí, de manera que las consultas relacionadas sean lo más rápidas posible.
- Se creará un índice para la fecha de compra del dispositivo, la fecha del movimiento de la tabla movimientos y para la fecha de los pedidos. Estos índices serán de utilidad para rellenar la tabla de Estadísticas_año del repositorio estadístico.

En principio, no se considera crear más índices, aunque se tendrá que valorar una vez que se comience con la fase de implementación. Aumentar el número de índices crearía una mayor sobrecarga, ya que estos se crean de nuevo con cada operación de insert, update o delete, sin mejorar sensiblemente el rendimiento.

3.3.6 Controles de Integridad

Un SGBD de un modelo determinado debe velar por el cumplimiento de las reglas de integridad preestablecidas por su modelo. En el caso del modelo relacional se establece lo siguiente:

- Toda clave primaria que se elija para una relación no debe tener valores repetidos y no puede aceptar valores nulos.
- Cuando se define un atributo como NOT NULL se obliga a que tenga un valor, es decir, debe contener algún dato.
- La regla de integridad referencial está relacionada con el concepto de clave foránea. Determina que todos los valores que toma una clave foránea deben ser valores nulos o valores que existen en la clave primaria que referencia.

Con estos mecanismos se asegura parte de la integridad de los datos, pero existen otras restricciones que no quedan cubiertas por estos mecanismos, y deben verificarse con comandos CHECK al definir las tablas, y o bien, por comprobaciones que deberán realizarse en el aplicativo.

En nuestro proyecto se debe revisar lo siguiente:

- Los responsables de las recepciones de los pedidos deben pertenecer al departamento informático. Se comprobará mediante algún procedimiento que el usuario responsable pertenece a este departamento.
- Solo las personas del departamento de informática podrán realizar cualquier movimiento. Se comprobará mediante algún procedimiento que los usuarios que realizan cada movimiento pertenezcan a este departamento.

- También se comprobará que en la tabla presupuestos no existan dos presupuestos de distinto proveedor, para el mismo pedido, que hayan sido aceptados.

3.3.7 Repositorio Estadístico

El repositorio estadístico tiene una serie de tablas, que según los requisitos del sistema, deben responder en un tiempo constante 1 a las consultas que se realicen, es decir, haciendo una SELECT sobre un registro de una tabla, que no sea una vista calculada o materializada, ni utilizando funciones de agregados (sum, avg, etc, con group by).

A continuación se detallan las tablas que formarán parte de este repositorio estadístico:

ESTADÍSTICAS_AÑO			
Campos	Tipo	Nulo	Descripción
AÑO	VARCHAR2(4)	NOT NULL	Año de la estadística
NUM_DISPOSITIVOS	NUMBER	NOT NULL	Número de dispositivos comprados en ese año
IMPORTE_PEDIDOS	NUMBER(10,2)	NOT NULL	Importe de los pedidos de ese año ligados a una orden de inversión
NUM_AVERIAS	NUMBER	NOT NULL	Número de averías en ese año en toda la empresa
Restricciones	Tipo	Tabla Referenciada	Campos
PK_ESTADISTICAS_AÑO	Primary Key		AÑO

PORTATILES_SIN_AVERIA			
Campos	Tipo	Nulo	Descripción
NUM_PORTATILES	NUMBER	NOT NULL	Número total de portátiles en la empresa
PORTATILES_SIN_AVERIA	NUMBER	NOT NULL	Número de portátiles sin ninguna avería
AVG_SIN_AVERIA	NUMBER	NOT NULL	Porcentaje de portátiles sin avería
Restricciones	Tipo	Tabla Referenciada	Campos

Aclaración: El campo AVG_SIN_AVERIA es un campo virtual¹ cuyo valor se derivará de una operación entre los campos num_portátiles y portátiles_sin_avería.

PROVEEDOR_MAS_RAPIDO			
Campos	Tipo	Nulo	Descripción
ID_PROVEEDOR	VARCHAR2(6)	NOT NULL	Identificador del proveedor

¹ Campo virtual define un campo que contiene datos derivados (calculados) de otros campos de la base de datos. Es una característica nueva de la versión 11 g de oracle.

NOMBRE	VARCHAR2(60)	NOT NULL	Nombre del proveedor
PERIODO_ENTREGA	NUMBER	NOT NULL	Periodo de entrega (en días)
Restricciones	Tipo	Tabla Referenciada	Campos
PK_PROVEEDOR_MAS_RAPIDO	Primary Key		ID_PROVEEDOR
FK_PROVEEDOR_MAS_RAPIDO_PROV	Foreing Key	PROVEEDORES	ID_PROVEEDOR

DISP_ACTIVOS_DEPARTAMENTO			
Campos	Tipo	Nulo	Descripción
ID_DEPARTAMENTO	VARCHAR2(3)	NOT NULL	Identificador del departamento
ID_TIPO_DISPOSITIVO	NUMBER	NOT NULL	Identificador del tipo de dispositivo
NOMBRE_DEPARTAMENTO	VARCHAR2(60)	NOT NULL	Nombre del departamento
DESCRIPCION_TIPO	VARCHAR2(30)	NOT NULL	Nombre del tipo de dispositivo
NUM_DISPOSITIVOS	NUMBER		Número de dispositivos de ese tipo para ese departamento
Restricciones	Tipo	Tabla Referenciada	Campos
PK_DISP_ACT_DEPARTAMENTO	Primary Key		ID_DEPARTAMENTO, ID_TIPO_DISPOSITIVO
FK_DISP_ACT_DEP_DEPARTAMENTO	Foreing Key	DEPARTAMENTOS	ID_DEPARTAMENTO
FK_DISP_ACT_DEP_TIPO_DISPOSITIVO	Foreing Key	TIPO_DISPOSITIVO	ID_TIPO_DISPOSITIVO

INVENTARIO_DEPARTAMENTO			
Campos	Tipo	Nulo	Descripción
ID_DEPARTAMENTO	VARCHAR2(3)	NOT NULL	Identificador del departamento
NOMBRE_DEPARTAMENTO	VARCHAR2(60)	NOT NULL	Nombre del departamento
NUM_TIPOS_DISPOSITIVOS	NUMBER	NOT NULL	Número de tipos diferentes de dispositivos que tiene el departamento
VALOR_INVENTARIO	NUMBER	NOT NULL	Valor del inventario activo en ese departamento
Restricciones	Tipo	Tabla Referenciada	Campos
PK_INVENTARIO_DEPARTAMENTO	Primary Key		ID_DEPARTAMENTO
FK_INVENTARIO_DEPT_DEPARTAMENTO	Foreing Key	DEPARTAMENTOS	ID_DEPARTAMENTO

DISPOSITIVOS_FECHA_INVENTARIO			
Campos	Tipo	Nulo	Descripción
NUM_DISPOSITIVOS	NUMBER	NOT NULL	Número de dispositivos con fecha de inventario mayor de un año
Restricciones	Tipo	Tabla Referenciada	Campos

DEPTO_MAS_REASIGNACIONES			
Campos	Tipo	Nulo	Descripción
ID_DEPARTAMENTO	VARCHAR2(3)	NOT NULL	Identificador del departamento
AÑO	VARCHAR2(4)	NOT NULL	Año en el que se producen las reasignaciones
NOMBRE_DEPARTAMENTO	VARCHAR2(60)	NOT NULL	Nombre del departamento
NUM_REASIGNACIONES	NUMBER	NOT NULL	Número total de reasignaciones en ese departamento y en ese año
Restricciones	Tipo	Tabla Referenciada	Campos
PK_DEP_MAS_REASIGNACIONES	Primary Key		ID_DEPARTAMENTO, AÑO
FK_DEP_MAS_REASIG_DEPART	Foreing Key	DEPARTAMENTOS	ID_DEPARTAMENTO

USUARIOS_DISPOSITIVOS			
Campos	Tipo	Nulo	Descripción
TOTAL_USUARIOS	NUMBER	NOT NULL	Número total de usuarios de la empresa
USUARIOS_SIN_DISPOSITIVOS	NUMBER	NOT NULL	Número de usuarios sin ningún dispositivo asignado
AVG_SIN_DISPOSITIVOS	NUMBER		Porcentaje de usuarios sin ningún dispositivo asignado
USUARIOS_MAS5_DISPOSITIVOS	NUMBER	NOT NULL	Número de usuarios que tienen más de 5 dispositivos asignados
Restricciones	Tipo	Tabla Referenciada	Campos

Aclaración: El campo AVG_SIN_DISPOSITIVOS es un campo virtual cuyo valor se derivará de una operación entre los campos total_usuarios y usuarios_sin_dispositivos.

UNIDADES_DISPOSITIVO_ALMACÉN			
Campos	Tipo	Nulo	Descripción
ID_TIPO_DISPOSITIVO	NUMBER	NOT NULL	Identificador del tipo de dispositivo
DESCRIPCION	VARCHAR2(30)	NOT NULL	Nombre del tipo de dispositivo
NUM_UNIDADES	NUMBER	NOT NULL	Número de unidades del tipo de dispositivo que se encuentran en el almacén.
Restricciones	Tipo	Tabla Referenciada	Campos
PK_UN_DISP_ALMACEN	Primary Key		ID_TIPO_DISPOSITIVO
FK_UN_DISP_ALMACEN_TIPO_DISP	Foreing Key	TIPO_DISPOSITIVO	ID_TIPO_DISPOSITIVO

USUARIOS_AVERIAS			
Campos	Tipo	Nulo	Descripción
ID_USUARIO	VARCHAR2(6)	NOT NULL	Identificador del usuario
NOMBRE_USUARIO	VARCHAR2(60)	NOT NULL	Nombre del usuario
NUM_AVERIAS	NUMBER	NOT NULL	Número de averías registradas en los dispositivos del usuario

Restricciones	Tipo	Tabla Referenciada	Campos
PK_USUARIOS_AVERIAS	Primary Key		ID_USUARIO
FK_USUARIOS_AVERIAS_USUARIO	Foreing Key	USUARIOS	ID_USUARIO

USUARIOS_TOP10_AVERIAS			
Campos	Tipo	Nulo	Descripción
ID_USUARIO	VARCHAR2(6)	NOT NULL	Identificador del usuario
NOMBRE_USUARIO	VARCHAR2(60)	NOT NULL	Nombre del usuario
NUM_AVERIAS	NUMBER	NOT NULL	Número de averías registradas en los dispositivos del usuario
RANKING	NUMBER		Puesto que ocupa el usuario en el ranking de averías
Restricciones	Tipo	Tabla Referenciada	Campos
PK_USUARIOS_TOP10_AVERIAS	Primary Key		ID_USUARIO
FK_USU_TOP10AVERIAS_USUARIO	Foreing Key	USUARIOS	ID_USUARIO

DEPTO_MAS_TIPOSDISP			
Campos	Tipo	Nulo	Descripción
ID_DEPARTAMENTO	VARCHAR2(3)	NOT NULL	Identificador del departamento
NOMBRE_DEPARTAMENTO	VARCHAR2(60)	NOT NULL	Nombre del departamento
NUM_TIPOS_DISPOSITIVOS	NUMBER	NOT NULL	Número de tipos diferentes de dispositivos que tiene el departamento
Restricciones	Tipo	Tabla Referenciada	Campos
PK_DEPTO_TIPOSDISP_DEPARTAMENTO	Primary Key		ID_DEPARTAMENTO
FK_DEPTO_TIPOSDISP_DEPT_DEPARTAMENTO	Foreing Key	DEPARTAMENTOS	ID_DEPARTAMENTO

3.3.8 Tabla de Logs

Las llamadas a los procedimientos del sistema se almacenarán en una tabla de logs, que tendrá información sobre los parámetros de entrada y salida de los mismos.

La tabla de logs tendrá los siguientes campos:

LOGS (id_log, fecha, procedimiento, paramsentrada, paramssalida)

Donde el campo “procedimiento” corresponderá al nombre del procedimiento almacenado, el campo “paramsentrada” tendrá los parámetros de entrada, separados por comas y “paramssalida” tendrá los parámetros de salida igualmente separados por comas. Cada entrada tendrá además una fecha en la cual se inserta dicha entrada.

Los procedimientos grabarán en la tabla LOGS los movimientos realizados, de forma que servirá de diario de la base de datos. Con esta tabla se podrá detectar cualquier problema y auditar los movimientos realizados.

A continuación se detalla la tabla:

LOGS			
Campos	Tipo	Nulo	Descripción
ID_LOG	NUMBER	NOT NULL	Identificación del Log
FECHA	DATE	NOT NULL	Fecha del Log
PROCEDIMIENTO	VARCHAR2(200)	NOT NULL	Procedimiento ejecutado
PARAMSENTRADA	VARCHAR2(2000)	NOT NULL	Parámetros de entrada separados por comas
PARAMSSALIDA	VARCHAR2(2000)	NOT NULL	Parámetros de salida separados por comas
Restricciones	Tipo	Tabla Referenciada	Campos
PK_LOGS	Primary Key		ID_LOG

4. Implementación

4.1 Creación de Tablespaces

Se crean 2 “tablespaces”, uno guardará todos los datos de las tablas y el otro guardará los índices (Se encuentran en el fichero *tablespaces.sql*)

```
CREATE TABLESPACE TFG_DAT DATAFILE
'C:\ORACLE\PRODUCT\10.2.0\ORADATA\ORA10G2WIN\TFG_DAT_01.DBF'
SIZE 10M AUTOEXTEND OFF
LOGGING
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 8K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;
```

```
CREATE TABLESPACE TFG_IND DATAFILE
'C:\ORACLE\PRODUCT\10.2.0\ORADATA\ORA10G2WIN\TFG_IND_01.DBF'
SIZE 10M AUTOEXTEND OFF
LOGGING
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 8K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;
```

4.2 Creación de Usuarios

Se ha creado un único usuario, aunque dependiendo de las características de la empresa donde se vaya a implementar la aplicación, podrán crearse más. Este usuario será propietario de todos los objetos de la base de datos (tablas, índices, procedimientos, disparadores, etc.) y tendrá los privilegios suficientes para crear todos los objetos necesarios y acceder a los tablespaces. El fichero con la creación del usuario se llama *usuario.sql*

```
CREATE USER TFG
```

```
IDENTIFIED BY 'TFG333'  
DEFAULT TABLESPACE TFG_DAT  
TEMPORARY TABLESPACE TEMP  
PROFILE DEFAULT  
ACCOUNT UNLOCK;  
  
GRANT RESOURCE TO TFG;  
GRANT CONNECT TO TFG;  
GRANT CREATE_VIEW TO TFG;  
  
ALTER USER TFG DEFAULT ROLE ALL;  
ALTER USER TFG QUOTA UNLIMITED ON TFG_DAT;  
ALTER USER TFG QUOTA UNLIMITED ON TFG_IND;
```

4.3 Creación de Secuencias

Las secuencias se emplean para generar valores enteros secuenciales únicos y asignárselos a campos numéricos; se utilizan generalmente para las claves primarias de las tablas garantizando que sus valores no se repitan.

Se utilizarán secuencias, para los identificadores de todas las tablas. En algunos casos serán valores numéricos y en otros se hará una conversión, mediante los disparadores asociados a las sentencias.

Todas las secuencias comenzarán por 1 y se incrementarán de 1 en 1. El valor máximo dependerá de la longitud del identificador de que se trate. Se utilizará en todas la cláusula NOCACHE², que garantiza que no se perderán números. El fichero con las secuencias y los disparadores necesarios para éstas es *secuencias.sql*

A continuación, se muestran un par de ejemplos:

```
CREATE SEQUENCE SEC_TIPO_MOV  
START WITH 1 INCREMENT BY 1  
NOCACHE;  
  
CREATE OR REPLACE  
TRIGGER INSERT_ID_TIPO_MOV BEFORE INSERT ON TIPO_MOVIMIENTO  
FOR EACH ROW  
BEGIN  
SELECT SEC_TIPO_MOV.NEXTVAL INTO :NEW.ID_TIPO_MOVIMIENTO FROM DUAL;  
END INSERT_ID_TIPO_MOV;  
  
CREATE SEQUENCE SEC_USUARIOS  
START WITH 1 INCREMENT BY 1  
MAXVALUE 9999  
NOCACHE;  
  
CREATE OR REPLACE  
TRIGGER INSERT_ID_USUARIO BEFORE INSERT ON USUARIOS  
FOR EACH ROW  
BEGIN  
SELECT 'US' || LPAD(SEC_USUARIOS.NEXTVAL,4,'0') INTO :NEW.ID_USUARIO FROM DUAL;  
END INSERT_ID_USUARIO;
```

² La cláusula CACHE O NOCACHE indica si Oracle asignará previamente números de secuencia en memoria (Oracle asigna previamente 20 valores de forma predeterminada)

4.4 Creación de Tablas

El fichero que contiene la creación de todas las tablas, incluidas las tablas del repositorio estadístico y la tabla de log, se llama *tablas.sql*

Se han tenido que crear dos tablas auxiliares, una para proveedores y otra para movimientos porque era necesario hacer select de estas tablas dentro de los disparadores delete y update que afectaban a estas mismas tablas, lo que producía problemas de “tablas mutantes”³. Para solucionar este problema, se utilizan las tablas auxiliares para hacer las select, ya que contienen los mismos datos que las tablas originales.

También se han añadido dos nuevas tablas, con respecto al diseño original, Usuarios_Top10_Averias, que se irá actualizando con los datos del ranking de usuarios por averias y Depto_Mas_TipoDisp, que contendrá en todo momento el Departamento que tiene más tipos diferentes de dispositivos. Para actualizar la tabla Usuarios_Top10_Averias se usará la función analítica dense_rank⁴ sobre la tabla Usuarios_Averias, consiguiendo la posición global que ocupa cada usuario dentro de la tabla. Esta tabla podrá contener más de 10 usuarios si hay empates en el ranking.

Ejemplo de creación de tabla:

```
DROP TABLE SEDES CASCADE CONSTRAINTS;

CREATE TABLE SEDES
(
    ID_SEDE                VARCHAR2(3)    NOT NULL,
    NOMBRE_SEDE            VARCHAR2(60)   NOT NULL,
    DIRECCION_SEDE         VARCHAR2(40),
    LOCALIDAD_SEDE         VARCHAR2(30),
    PROVINCIA_SEDE         VARCHAR2(30),
    PAIS_SEDE              VARCHAR2(30),
    FECHA_ALTA             DATE           NOT NULL,
    FECHA_BAJA             DATE
)
TABLESPACE TFG_DAT;

ALTER TABLE SEDES
ADD CONSTRAINT PK_SEDES PRIMARY KEY (ID_SEDE)
USING INDEX
TABLESPACE TFG_IND;
```

4.5 Creación de Índices

Cuando se definen las claves primarias de las tablas, Oracle crea un índice único y no nulo (UNIQUE, NOT NULL) asociado a esa clave primaria. Aparte de estos, se han creado los índices correspondientes a las claves foráneas para cada una de las tablas y un índice para algunos de los campos que contienen fechas, de las que será necesario efectuar búsquedas (fecha de compra y fecha del último inventario de la tabla dispositivos, fecha del movimiento, de la tabla movimientos, fecha de pedido, de la tabla pedidos. El fichero que contiene la creación de índices se llama *indices.sql*

³ Tabla mutante es una tabla que está siendo modificada por una sentencia SQL (insert, update, delete)

⁴ Dense_rank calcula el ranking de una fila en un conjunto ordenado de filas devolviendo dicho ranking como un número. Las filas con igual valor en el criterio del ranking tienen el mismo ranking.


```
CREATE INDEX I_USUARIO_DEPARTAMENTO ON USUARIOS (ID_DEPARTAMENTO)

TABLESPACE TFG_IND;

CREATE INDEX I_DISPOSITIVO_FECHA_COMPRA ON DISPOSITIVOS (FECHA_COMPRA)

TABLESPACE TFG_IND;
```

4.6 Construcción de Procedimientos de Alta, Baja y Modificación

Para realizar toda la gestión y acceso a la información se han utilizado Procedimientos Almacenados, ya que según el enunciado del Proyecto, esta era la única forma de hacerlo. A continuación se detallan los procedimientos de Alta, Baja y Modificación de cada una de las tablas principales. Las tablas auxiliares se rellenan a través de disparadores sobre éstas.

PR_ALTA_TIPO_MOVIMIENTO
Da de alta un tipo de movimiento. En caso de que ya exista se mostraría "Tipo de movimiento existente". En caso de que no exista se verificarán todos los datos que no admitan nulos.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_descripción: descripción del tipo de movimiento
Parámetros de salida
RSP
Pre-condición
No existe el tipo de movimiento a dar de alta
Post-condición
Se da de alta el tipo de movimiento
Retorna
<ul style="list-style-type: none"> • "OK" si la operación se ha realizado con éxito • "ERROR: el campo no puede ser nulo" • "ERROR: el tipo de movimiento ya existe" • "ERROR: tipo de error" en caso de que el error no haya sido tratado

PR_BAJA_TIPO_MOVIMIENTO
Da de baja un tipo de movimiento. Tiene que cumplirse que el tipo de movimiento exista y que no esté dado de baja.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_tipo_movimiento: identificador del tipo de movimiento
Parámetros de salida
RSP
Pre-condición
Existe el tipo de movimiento a dar de baja
Post-condición
Se da de baja el tipo de movimiento
Retorna
<ul style="list-style-type: none"> • "OK" si la operación se ha realizado con éxito • "ERROR: el campo no puede ser nulo" • "ERROR: el tipo de movimiento no existe en la base de datos" • "ERROR: el tipo de movimiento ya había sido dado de baja" • "ERROR: tipo de error" en caso de que el error no haya sido tratado

PR_MODIF_TIPO_MOVIMIENTO
Realiza una modificación en un tipo de movimiento en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el tipo de movimiento exista y que todos los campos obligatorios tengan algún valor.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_tipo_movimiento: identificador del tipo de movimiento
Opcionales:

<ul style="list-style-type: none"> • p_descripción: descripción del tipo de movimiento
Parámetros de salida
RSP
Pre-condición
Existe el tipo de movimiento a modificar
Post-condición
Se modifica el tipo de movimiento
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador del tipo de movimiento a modificar no existe” • “ERROR: el tipo de movimiento no puede ser nulo” • “ERROR: el tipo de movimiento que se va a modificar está de baja” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_TIPO_DISPOSITIVO
Da de alta un tipo de dispositivo. En caso de que ya exista se mostraría “Tipo de dispositivo existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_descripción: descripción del tipo de dispositivo
Parámetros de salida
RSP
Pre-condición
No existe el tipo de dispositivo a dar de alta
Post-condición
Se da de alta el tipo de dispositivo
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el tipo de dispositivo ya existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_TIPO_DISPOSITIVO
Da de baja un tipo de dispositivo. Tiene que cumplirse que el tipo de dispositivo exista y que no esté dado de baja.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_tipo_dispositivo: identificador del tipo de dispositivo
Parámetros de salida
RSP
Pre-condición
Existe el tipo de dispositivo a dar de baja
Post-condición
Se da de baja el tipo de dispositivo
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el tipo de dispositivo no existe en la base de datos” • “ERROR: el tipo de dispositivo ya había sido dado de baja” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_TIPO_DISPOSITIVO
Realiza una modificación en un tipo de dispositivo en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el tipo de dispositivo exista y que todos los campos obligatorios tengan algún valor.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_tipo_dispositivo: identificador del tipo de dispositivo
Opcionales: <ul style="list-style-type: none"> • p_descripción: descripción del tipo de dispositivo
Parámetros de salida
RSP

Pre-condición
Existe el tipo de dispositivo a modificar
Post-condición
Se modifica el tipo de dispositivo
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador del tipo de dispositivo a modificar no existe” • “ERROR: el identificador del tipo de dispositivo no puede ser nulo” • “ERROR: el tipo de dispositivo que se va a modificar está de baja” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_PROVEEDOR
Da de alta un proveedor. En caso de que ya exista se mostraría “Proveedor existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.
Parámetros de entrada
Obligatorios :
<ul style="list-style-type: none"> • p_nombre: nombre del proveedor • p_periodo_entrega: periodo medio de entrega del proveedor
Opcionales:
<ul style="list-style-type: none"> • p_dirección: dirección del proveedor • p_localidad: localidad del proveedor • p_provincia: provincia del proveedor • p_país: país del proveedor • p_teléfono: teléfono del proveedor • p_email: correo electrónico del proveedor
Parámetros de salida
RSP
Pre-condición
No existe el proveedor a dar de alta
Post-condición
Se da de alta el proveedor
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el proveedor ya existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_PROVEEDOR
Da de baja un proveedor. Tiene que cumplirse que el proveedor exista y que no esté dado de baja.
Parámetros de entrada
Obligatorios :
<ul style="list-style-type: none"> • p_id_proveedor: identificador del proveedor
Parámetros de salida
RSP
Pre-condición
Existe el proveedor a dar de baja
Post-condición
Se da de baja el proveedor
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el proveedor no existe en la base de datos” • “ERROR: el proveedor ya había sido dado de baja” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_PROVEEDOR
Realiza una modificación en un proveedor en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el proveedor exista y que todos los campos obligatorios tengan algún valor.
Parámetros de entrada
Obligatorios :

Opcionales:	<ul style="list-style-type: none"> • p_id_proveedor: identificador del proveedor • p_nombre: nombre del proveedor • p_dirección: dirección del proveedor • p_localidad: localidad del proveedor • p_provincia: provincia del proveedor • p_país: país del proveedor • p_teléfono: teléfono del proveedor • p_email: correo electrónico del proveedor • p_periodo_entrega: periodo medio de entrega en días
Parámetros de salida	
RSP	
Pre-condición	
Existe el proveedor a modificar	
Post-condición	
Se modifica el proveedor	
Retorna	
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador del proveedor a modificar no existe” • “ERROR: el identificador del proveedor no puede ser nulo” • “ERROR: el proveedor que se va a modificar está de baja” • “ERROR: tipo de error” en caso de que el error no haya sido tratado 	

PR_ALTA_SEDE	
Da de alta una sede. En caso de que ya exista se mostraría “Sede existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.	
Parámetros de entrada	
Obligatorios :	<ul style="list-style-type: none"> • p_nombre_sede: nombre de la sede
Opcionales:	<ul style="list-style-type: none"> • p_dirección_sede: dirección de la sede • p_localidad_sede : localidad de la sede • p_provincia_sede : provincia de la sede • p_país_sede: país de la sede
Parámetros de salida	
RSP	
Pre-condición	
No existe la sede a dar de alta	
Post-condición	
Se da de alta la sede	
Retorna	
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: la sede ya existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado 	

PR_BAJA_SEDE	
Da de baja una sede. Tiene que cumplirse que la sede exista y que no esté dada de baja.	
Parámetros de entrada	
Obligatorios :	<ul style="list-style-type: none"> • p_id_sede: identificador de la sede
Parámetros de salida	
RSP	
Pre-condición	
Existe la sede a dar de baja	
Post-condición	
Se da de baja la sede	
Retorna	
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: la sede no existe en la base de datos” • “ERROR: la sede ya había sido dada de baja” 	

- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_SEDE

Realiza una modificación en una sede en cualquiera de sus parámetros opcionales. Previamente debe comprobar que la sede exista y que todos los campos obligatorios tengan algún valor.

Parámetros de entrada

Obligatorios :

- p_id_sede: identificador de la sede

Opcionales:

- p_nombre_sede: nombre de la sede
- p_dirección_sede: dirección de la sede
- p_localidad_sede: localidad de la sede
- p_provincia_sede: provincia de la sede
- p_país_sede: país de la sede

Parámetros de salida

RSP

Pre-condición

Existe la sede a modificar

Post-condición

Se modifica la sede

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el identificador de la sede a modificar no existe”
- “ERROR: el identificador de la sede no puede ser nulo”
- “ERROR: la sede que se va a modificar está de baja”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_DEPARTAMENTO

Da de alta un departamento. En caso de que ya exista se mostraría “Departamento existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.

Parámetros de entrada

Obligatorios :

- p_nombre_departamento: nombre del departamento

Opcionales:

- p_id_responsable: identificador del responsable del departamento

Parámetros de salida

RSP

Pre-condición

No existe el departamento a dar de alta. Si id_responsable es un parámetro de entrada debe ser nulo o existir en la base de datos

Post-condición

Se da de alta el departamento

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el campo no puede ser nulo”
- “ERROR: el departamento ya existe”
- “ERROR: el identificador del usuario responsable no existe”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_DEPARTAMENTO

Da de baja un departamento. Tiene que cumplirse que el departamento exista y que no esté dado de baja.

Parámetros de entrada

Obligatorios :

- p_id_departamento: identificador del departamento

Parámetros de salida

RSP

Pre-condición

Existe el departamento a dar de baja

Post-condición

Se da de baja el departamento

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el campo no puede ser nulo”
- “ERROR: el departamento no existe en la base de datos”
- “ERROR: el departamento ya había sido dado de baja”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_DEPARTAMENTO

Realiza una modificación en un departamento en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el departamento exista y que todos los campos obligatorios tengan algún valor.

Parámetros de entrada

Obligatorios :

- p_id_departamento: identificador del departamento

Opcionales:

- p_nombre_departamento: nombre del departamento
- p_id_responsable: identificador del responsable del departamento

Parámetros de salida

RSP

Pre-condición

Existe el departamento a modificar. Si se va a modificar el responsable, se verifica que el usuario exista en la BD.

Post-condición

Se modifica el departamento

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el identificador del departamento a modificar no existe”
- “ERROR: el identificador del departamento no puede ser nulo”
- “ERROR: el departamento que se va a modificar está de baja”
- “ERROR: el identificador del usuario responsable no existe”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_DISPOSITIVO

Da de alta un dispositivo. En caso de que ya exista se mostraría “Dispositivo existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.

Parámetros de entrada

Obligatorios :

- p_tipo: tipo de dispositivo
- p_numero_serie: número de serie del dispositivo
- p_fecha_compra: fecha de compra del dispositivo
- p_precio_compra: precio de compra del dispositivo

Opcionales:

- p_fecha_instalacion: fecha de instalación del dispositivo
- p_id_usuario: identificador del usuario al que se asigna el dispositivo
- p_fecha_ultimo_inventario: fecha del último inventario del dispositivo

Parámetros de salida

RSP

Pre-condición

No existe el dispositivo a dar de alta. Tiene que existir el tipo de dispositivo en la base de datos

Post-condición

Se da de alta el dispositivo

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el campo no puede ser nulo”
- “ERROR: el dispositivo ya existe”
- “ERROR: el tipo de dispositivo no existe”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_DISPOSITIVO

Da de baja un dispositivo. Tiene que cumplirse que el dispositivo exista y que no esté dado de baja.

Parámetros de entrada

Obligatorios :

<ul style="list-style-type: none"> • p_id_dispositivo: identificador del dispositivo
Parámetros de salida
RSP
Pre-condición
Existe el dispositivo a dar de baja
Post-condición
Se da de baja el dispositivo
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el dispositivo no existe en la base de datos” • “ERROR: el dispositivo ya había sido dado de baja” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_DISPOSITIVO	
Realiza una modificación en un dispositivo en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el dispositivo exista y que todos los campos obligatorios tengan algún valor.	
Parámetros de entrada	
Obligatorios :	<ul style="list-style-type: none"> • p_id_dispositivo: identificador del dispositivo
Opcionales:	<ul style="list-style-type: none"> • p_tipo: tipo del dispositivo • p_numero_serie: número de serie del dispositivo • p_fecha_instalacion: fecha de instalación del dispositivo • p_id_usuario: identificador del usuario al que se ha asignado el dispositivo • p_fecha_compra: fecha de compra del dispositivo • p_precio_compra: precio de compra del dispositivo • p_fecha_ultimo_inventario: fecha del último inventario del dispositivo
Parámetros de salida	
RSP	
Pre-condición	
Existe el dispositivo a modificar. Si se va a modificar el tipo de dispositivo, se comprueba que el nuevo tipo exista. Si se va a modificar el usuario al que se ha asignado, se verifica que el usuario exista en la BD.	
Post-condición	
Se modifica el dispositivo	
Retorna	
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador del dispositivo a modificar no existe” • “ERROR: el identificador del dispositivo no puede ser nulo” • “ERROR: el dispositivo que se va a modificar está de baja” • “ERROR: el identificador del tipo de dispositivo no existe” • “ERROR: el identificador del usuario al que se ha asignado no existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado 	

PR_ALTA_USUARIOS	
Da de alta un usuario. En caso de que ya exista se mostraría “Usuario existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.	
Parámetros de entrada	
Obligatorios :	<ul style="list-style-type: none"> • p_id_departamento: identificador del departamento al que pertenece el usuario • p_nombre: nombre del usuario • p_id_sede: identificador de la sede donde trabaja el usuario
Opcionales:	<ul style="list-style-type: none"> • p_teléfono: teléfono del usuario • p_email: correo electrónico del usuario
Parámetros de salida	
RSP	
Pre-condición	
No existe el usuario a dar de alta. Tienen que existir el departamento y la sede en la base de datos	
Post-condición	
Se da de alta el usuario	
Retorna	

<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el usuario ya existe” • “ERROR: el departamento asignado al usuario no existe” • “ERROR: la sede donde trabaja el usuario no existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado
--

PR_BAJA_USUARIO
Da de baja un usuario. Tiene que cumplirse que el usuario exista y que no esté dado de baja.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_usuario: identificador del usuario
Parámetros de salida
RSP
Pre-condición
Existe el usuario a dar de baja
Post-condición
Se da de baja el usuario
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el usuario no existe en la base de datos” • “ERROR: el usuario ya había sido dado de baja” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_USUARIO
Realiza una modificación en un usuario en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el usuario exista y que todos los campos obligatorios tengan algún valor.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_usuario: identificador del usuario
Opcionales: <ul style="list-style-type: none"> • p_id_departamento: identificador del departamento al que pertenece el usuario • p_nombre: nombre del usuario • p_telemfono: teléfono del usuario • p_email: correo electrónico del usuario • p_id_sede: identificador de la sede en la que trabaja el usuario
Parámetros de salida
RSP
Pre-condición
Existe el usuario a modificar. Si se va a modificar el departamento, se verifica que el nuevo departamento exista en la BD. Si se va a modificar la sede, se comprueba que la nueva sede exista en la base de datos.
Post-condición
Se modifica el usuario
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador del usuario a modificar no existe” • “ERROR: el identificador del usuario no puede ser nulo” • “ERROR: el usuario que se va a modificar está de baja” • “ERROR: el identificador del departamento no existe” • “ERROR: el identificador de la sede no existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_PEDIDOS
Da de alta un pedido. En caso de que ya exista se mostraría “Pedido existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_proveedor: identificador del proveedor del pedido

<ul style="list-style-type: none"> • p_id_departamento: identificador del departamento al que pertenece el pedido • p_fecha_pedido: fecha del pedido • p_importe_pedido: importe del pedido <p>Opcionales:</p> <ul style="list-style-type: none"> • p_orden_inversion: orden de inversión a la que va ligada el pedido
Parámetros de salida
RSP
Pre-condición
No existe el usuario a dar de alta. Tienen que existir el proveedor y el departamento
Post-condición
Se da de alta el pedido
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el pedido ya existe” • “ERROR: el proveedor que sirve el pedido no existe” • “ERROR: el departamento que ha solicitado el pedido no existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_PEDIDO
Da de baja un pedido. Tiene que cumplirse que el pedido exista.
Parámetros de entrada
Obligatorios :
<ul style="list-style-type: none"> • p_id_pedido: identificador del pedido
Parámetros de salida
RSP
Pre-condición
Existe el pedido a dar de baja
Post-condición
Se da de baja el pedido
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el pedido no existe en la base de datos” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_PEDIDO
Realiza una modificación en un pedido en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el pedido exista y que todos los campos obligatorios tengan algún valor.
Parámetros de entrada
Obligatorios :
<ul style="list-style-type: none"> • p_id_pedido: identificador del pedido
Opcionales:
<ul style="list-style-type: none"> • p_id_proveedor: identificador del proveedor que servirá el pedido • p_id_departamento: identificador del departamento al que pertenece el pedido • p_fecha_pedido: fecha del pedido • p_importe_pedido: importe del pedido • p_orden_inversion: orden de inversión a la que va ligada el pedido
Parámetros de salida
RSP
Pre-condición
Existe el pedido a modificar. Si se va a modificar el proveedor, se verifica que el nuevo proveedor exista en la BD. Si se va a modificar el departamento al que pertenece el pedido, se verifica que el nuevo departamento exista en la BD.
Post-condición
Se modifica el pedido
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador del pedido a modificar no existe” • “ERROR: el identificador del pedido a modificar no puede ser nulo” • “ERROR: el identificador del proveedor del pedido no existe” • “ERROR: el identificador del departamento al que pertenece el pedido no existe”

- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_LINEA_PEDIDO

Da de alta una línea de pedido. En caso de que ya exista se mostraría “Línea del pedido existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.

Parámetros de entrada

Obligatorios :

- p_id_linea: identificador de la línea del pedido
- p_id_pedido: identificador del pedido
- p_id_material: identificador del tipo de dispositivo
- p_unidades: unidades de cada tipo de dispositivo
- p_precio_unitario: precio unitario de cada tipo de dispositivo

Opcionales:

Parámetros de salida

RSP

Pre-condición

No existe la línea de pedido a dar de alta. Tienen que existir el pedido y el tipo de dispositivo

Post-condición

Se da de alta la línea del pedido

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el campo no puede ser nulo”
- “ERROR: la línea del pedido ya existe”
- “ERROR: el pedido al que corresponde la línea no existe”
- “ERROR: el tipo de dispositivo solicitado en el pedido no existe”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_LINEA_PEDIDO

Da de baja una línea de pedido. Tiene que cumplirse que la línea exista.

Parámetros de entrada

Obligatorios :

- p_id_linea: identificador de la línea del pedido
- p_id_pedido: identificador del pedido

Parámetros de salida

RSP

Pre-condición

Existe la línea de pedido a dar de baja y existe el pedido

Post-condición

Se da de baja la línea del pedido

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el campo no puede ser nulo”
- “ERROR: la línea del pedido no existe en la base de datos”
- “ERROR: el pedido no existe en la base de datos”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_LINEA_PEDIDO

Realiza una modificación en una línea de un pedido en cualquiera de sus parámetros opcionales. Previamente debe comprobar que la línea y el pedido existan y que todos los campos obligatorios tengan algún valor.

Parámetros de entrada

Obligatorios :

- p_id_linea: identificador de la línea
- p_id_pedido: identificador del pedido al que pertenece la línea que se quiere modificar

Opcionales:

- p_id_material: identificador del tipo de dispositivo
- p_unidades: unidades que se piden del tipo de dispositivo
- p_precio_unitario: precio unitario de cada tipo de dispositivo

Parámetros de salida

RSP

Pre-condición

Existe la línea y el pedido a modificar. Si se va a modificar el material, se verifica que el nuevo tipo de dispositivo exista en la BD.
Post-condición
Se modifica la línea del pedido
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador de la línea no puede ser nulo” • “ERROR: el identificador del pedido no puede ser nulo” • “ERROR: el identificador de la línea no existe para ese pedido” • “ERROR: el identificador del material no existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_PRESUPUESTO
Da de alta un presupuesto. En caso de que ya exista se mostraría “Presupuesto existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.
Parámetros de entrada
Obligatorios :
<ul style="list-style-type: none"> • p_id_proveedor: identificador del proveedor que da el presupuesto • p_id_pedido: identificador del pedido sobre el que se hace el presupuesto • p_fecha_presupuesto: fecha en la que se da el presupuesto • p_importe: importe total del presupuesto
Opcionales:
<ul style="list-style-type: none"> • p_aceptado: indica si el presupuesto ha sido aceptado
Parámetros de salida
RSP
Pre-condición
No existe el presupuesto a dar de alta. Tienen que existir el proveedor y el pedido. No pueden existir dos presupuestos para el mismo proveedor y el mismo pedido. Si aceptado es TRUE, no puede existir en la tabla otro presupuesto aceptado para ese mismo pedido.
Post-condición
Se da de alta el presupuesto
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el presupuesto ya existe” • “ERROR: el proveedor que da el presupuesto no existe” • “ERROR: el pedido asociado al presupuesto no existe” • “ERROR: ya existe un presupuesto para ese proveedor y ese pedido” • “ERROR: ya existe otro presupuesto para ese mismo pedido que ha sido aceptado” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_PRESUPUESTO
Da de baja un presupuesto. Tiene que cumplirse que el presupuesto exista.
Parámetros de entrada
Obligatorios :
<ul style="list-style-type: none"> • p_id_presupuesto: identificador del presupuesto
Parámetros de salida
RSP
Pre-condición
Existe el presupuesto a dar de baja
Post-condición
Se da de baja el presupuesto
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el presupuesto no existe en la base de datos” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_PRESUPUESTO
Realiza una modificación en un presupuesto en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el presupuesto exista y que todos los campos obligatorios tengan algún valor.
Parámetros de entrada

Obligatorios :	<ul style="list-style-type: none"> • p_id_presupuesto: identificador del presupuesto
Opcionales:	<ul style="list-style-type: none"> • p_id_proveedor: identificador del proveedor que dará el presupuesto • p_id_pedido: identificador del pedido para el que se ha dado el presupuesto • p_fecha_presupuesto: fecha del presupuesto • p_importe: importe total del presupuesto • p_aceptado: indica si el presupuesto ha sido aceptado
Parámetros de salida	
RSP	
Pre-condición	
Existe el presupuesto a modificar. Si se va a modificar el proveedor, se verifica que el nuevo proveedor exista en la BD. Si se va a modificar el pedido, se verifica que el nuevo pedido exista en la BD. Si se va a modificar el campo aceptado poniéndolo a TRUE, se comprobará que no exista otro presupuesto aceptado para ese mismo pedido.	
Post-condición	
Se modifica el presupuesto	
Retorna	
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador del presupuesto a modificar no existe” • “ERROR: el identificador del presupuesto no puede ser nulo” • “ERROR: el identificador del proveedor que da el presupuesto no existe” • “ERROR: el identificador del pedido para el que se da el presupuesto no existe” • “ERROR: ya existe otro presupuesto para ese mismo pedido que ha sido aceptado” • “ERROR: tipo de error” en caso de que el error no haya sido tratado 	

PR_ALTA_RECEPCIÓN	
Da de alta una recepción de material. En caso de que ya exista se mostraría “Recepción existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.	
Parámetros de entrada	
Obligatorios :	<ul style="list-style-type: none"> • p_id_pedido: identificador del pedido del que se hace la recepción • p_id_responsable: identificador del usuario responsable de la recepción del pedido • p_fecha_recepcion: fecha en la que se recibe el material del pedido.
Opcionales:	<ul style="list-style-type: none"> • p_observaciones: recoge las incidencias que se hayan producido en la recepción del material
Parámetros de salida	
RSP	
Pre-condición	
No existe la recepción a dar de alta. Tienen que existir el pedido y el usuario responsable. El usuario responsable debe pertenecer al departamento de Informática	
Post-condición	
Se da de alta la recepción	
Retorna	
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: la recepción ya existe” • “ERROR: el pedido que se ha recepcionado no existe” • “ERROR: el usuario responsable de la recepción no existe” • “ERROR: el usuario responsable no pertenece al departamento de Informática” • “ERROR: tipo de error” en caso de que el error no haya sido tratado 	

PR_BAJA_RECEPCIÓN	
Da de baja una recepción. Tiene que cumplirse que la recepción exista.	
Parámetros de entrada	
Obligatorios :	<ul style="list-style-type: none"> • p_id_recepcion: identificador de la recepción
Parámetros de salida	
RSP	
Pre-condición	
Existe la recepción a dar de baja	
Post-condición	
Se da de baja la recepción	

Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: la recepción no existe en la base de datos” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_RECEPCIÓN
Realiza una modificación en una recepción en cualquiera de sus parámetros opcionales. Previamente debe comprobar que la recepción exista y que todos los campos obligatorios tengan algún valor.
Parámetros de entrada
Obligatorios :
<ul style="list-style-type: none"> • p_id_recepcion: identificador de la recepción de un pedido
Opcionales:
<ul style="list-style-type: none"> • p_id_pedido: identificador del pedido que se recepciona • p_id_responsable: identificador del usuario que recoge el pedido • p_fecha_recepción: fecha en la que se recepciona el pedido • p_observaciones: refleja las incidencias que puedan producirse en la recepción del material
Parámetros de salida
RSP
Pre-condición
Existe la recepción a modificar. Si se va a modificar el pedido, se verifica que el nuevo pedido exista en la BD. Si se va a modificar el responsable, se verifica que el nuevo responsable exista en la BD y que pertenezca al departamento de informática.
Post-condición
Se modifica la recepción
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador de la recepción a modificar no existe” • “ERROR: el identificador de la recepción no puede ser nulo” • “ERROR: el identificador del pedido que se recepciona no existe” • “ERROR: el identificador del usuario responsable de recoger el pedido no existe” • “ERROR: el usuario responsable no pertenece al departamento de informática” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_MOVIMIENTOS
Da de alta un movimiento de un dispositivo. En caso de que ya exista se mostraría “Movimiento existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.
Parámetros de entrada
Obligatorios :
<ul style="list-style-type: none"> • p_id_tipo_movimientor: identificador del tipo de movimiento • p_id_dispositivo: identificador del dispositivo sobre el que se realiza el movimiento • p_fecha_movimiento: fecha en la que se realiza el movimiento • p_id_responsable: identificador del usuario responsable del movimiento
Opcionales:
<ul style="list-style-type: none"> • p_id_usuario_origen: identificador del usuario que tiene el dispositivo • p_id_usuario_destino: identificador del nuevo usuario al que se ha asignado el dispositivo
Parámetros de salida
RSP
Pre-condición
No existe el movimiento a dar de alta. Tienen que existir el tipo de movimiento, el dispositivo y el usuario responsable de llevar a cabo el movimiento. El usuario responsable debe pertenecer al departamento de Informática.
Post-condición
Se da de alta el movimiento
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el movimiento ya existe” • “ERROR: el tipo de movimiento no existe” • “ERROR: el dispositivo sobre el que se realiza el movimiento no existe” • “ERROR: el usuario que tiene asignado el dispositivo no existe” • “ERROR: el usuario al que se le asigna el dispositivo no existe” • “ERROR: el usuario responsable no existe” • “ERROR: el usuario responsable no pertenece al departamento de Informática”

- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_MOVIMIENTO

Da de baja un movimiento. Tiene que cumplirse que el movimiento exista.

Parámetros de entrada

Obligatorios :

- p_id_movimiento: identificador del movimiento

Parámetros de salida

RSP

Pre-condición

Existe el movimiento a dar de baja

Post-condición

Se da de baja el movimiento

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el campo no puede ser nulo”
- “ERROR: el movimiento no existe en la base de datos”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_MOVIMIENTO

Realiza una modificación en un movimiento de un dispositivo en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el movimiento exista y que todos los campos obligatorios tengan algún valor.

Parámetros de entrada

Obligatorios :

- p_id_movimiento: identificador del movimiento

Opcionales:

- p_id_tipo_movimiento: identificador del tipo de movimiento
- p_id_dispositivo: identificador del dispositivo sobre el que se ha realizado el movimiento
- p_id_usuario_origen: identificador del usuario que tiene el dispositivo
- p_id_usuario_destino: identificador del nuevo usuario al que se ha asignado el dispositivo
- p_fecha_movimiento: fecha del movimiento
- p_id_responsable: identificador del usuario responsable del movimiento

Parámetros de salida

RSP

Pre-condición

Existe el movimiento a modificar. Si se va a modificar el tipo de movimiento, se verifica que el nuevo tipo de movimiento exista en la BD. Si se va a modificar el dispositivo, se verifica que el nuevo dispositivo exista en la BD. Si se van a modificar los campos usuario_origen, usuario_destino o id_responsable, se comprobará que el usuario exista en la BD y en el caso del responsable que pertenezca al departamento de informática.

Post-condición

Se modifica el movimiento

Retorna

- “OK” si la operación se ha realizado con éxito
- “ERROR: el identificador del movimiento a modificar no existe”
- “ERROR: el identificador del movimiento no puede ser nulo”
- “ERROR: el identificador del tipo de movimiento no existe”
- “ERROR: el identificador del dispositivo no existe”
- “ERROR: el usuario que tiene asignado el dispositivo no existe”
- “ERROR: el usuario al que se asigna el dispositivo no existe”
- “ERROR: el usuario responsable del movimiento no existe”
- “ERROR: el usuario responsable no pertenece al departamento de informática”
- “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_ALTA_INVENTARIO

Da de alta un inventario de material. En caso de que ya exista se mostraría “Inventario existente”. En caso de que no exista se verificarán todos los datos que no admitan nulos.

Parámetros de entrada

Obligatorios :

- p_id_sede: identificador de la sede donde se realiza el inventario
- p_id_coordinador: identificador del usuario responsable de hacer el inventario
- p_fecha_inventario: fecha en la que se realiza el inventario

Opcionales:
Parámetros de salida
RSP
Pre-condición
No existe el inventario a dar de alta. Tienen que existir la sede donde se realiza el inventario y el usuario responsable.
Post-condición
Se da de alta el inventario
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el inventario ya existe” • “ERROR: la sede donde se ha realizado el inventario no existe” • “ERROR: el usuario responsable de realizar el inventario no existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_BAJA_INVENTARIO
Da de baja un inventario. Tiene que cumplirse que el inventario exista.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_inventario: identificador del inventario
Parámetros de salida
RSP
Pre-condición
Existe el inventario a dar de baja
Post-condición
Se da de baja el inventario
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el campo no puede ser nulo” • “ERROR: el inventario no existe en la base de datos” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

PR_MODIF_INVENTARIO
Realiza una modificación en un inventario en cualquiera de sus parámetros opcionales. Previamente debe comprobar que el inventario exista y que todos los campos obligatorios tengan algún valor.
Parámetros de entrada
Obligatorios : <ul style="list-style-type: none"> • p_id_inventario: identificador del inventario
Opcionales: <ul style="list-style-type: none"> • p_id_sede: identificador de la sede donde se realiza el inventario • p_id_coordinador: identificador del usuario responsable del inventario • p_fecha_inventario: fecha en que se realiza el inventario
Parámetros de salida
RSP
Pre-condición
Existe el inventario a modificar. Si se va a modificar la sede, se verifica que la nueva sede exista en la BD. Si se va a modificar el coordinador, se verifica que el nuevo coordinador exista en la BD.
Post-condición
Se modifica el inventario
Retorna
<ul style="list-style-type: none"> • “OK” si la operación se ha realizado con éxito • “ERROR: el identificador del inventario a modificar no existe” • “ERROR: el identificador del inventario no puede ser nulo” • “ERROR: el identificador de la sede no existe” • “ERROR: el identificador del usuario responsable no existe” • “ERROR: tipo de error” en caso de que el error no haya sido tratado

El fichero donde se encuentran los procedimientos se llama *procedimientos.sql*

4.7 Construcción de Disparadores para el Módulo Estadístico

Para rellenar las tablas del módulo estadístico se han diseñado disparadores, que se activarán con las inserciones, borrados y actualizaciones de algunas tablas. En el siguiente cuadro se muestran los disparadores, indicando en qué momento y sobre qué tabla se activan, así como las acciones que producen sobre la base de datos.

Disparador	Cuando se activa	Acción
TRG_DISPOSITIVO_INSERT	Después de cada inserción en la tabla Dispositivos	Incrementa en uno el número de dispositivos comprados en el año de compra del dispositivo en la tabla Estadísticas_año. Si no hay datos de ese año en la tabla Estadística_año, inserta el año, el número de dispositivos a 1 y el importe de los pedidos y el número de averías a 0. Si el dispositivo es un portátil, incrementa en 1 el número de portátiles y los portátiles sin avería de la tabla Portátiles_sin_avería. Si no hay datos en la tabla y el dispositivo es un portátil, se inserta un registro en la tabla con el año y número de portátiles y portátiles sin avería igual a 1.
TRG_DEPARTAMENTO_INSERT	Después de cada inserción en la tabla Departamentos	Se inserta en la tabla Dispositivos_Activos_Departamento, el departamento con cada uno de los tipos de dispositivos existentes en la tabla Tipo_Dispositivo, con número de dispositivos igual a 0. Se inserta en la tabla Inventario_Departamento el identificador y el nombre del departamento con número de tipos de dispositivos y valor del inventario igual a 0.
TRG_DEPARTAMENTO_DELETE	Después de cada borrado en la tabla Departamentos	Se eliminan de la tabla Dispositivos_Activos_Departamento todas las instancias en las que el departamento sea el borrado. Se elimina de la tabla Inventario_Departamento el registro correspondiente al departamento borrado.
TRG_PROVEEDOR_INSERT	Después de cada inserción en la tabla Proveedores	Mira la tabla Proveedor_Mas_Rapido para ver si tiene algún registro. Si lo tiene, compara el periodo de entrega del proveedor que hay en la tabla con el del proveedor que se está dando de alta. Si es menor el periodo del nuevo proveedor, se borran los datos de la tabla Proveedor más rápido y se insertan los datos del nuevo proveedor. Si los periodos de entrega son iguales, se insertan los datos del nuevo proveedor sin borrar el anterior. Si la tabla Proveedor_Mas_Rapido no tiene ningún registro, se insertan los datos del nuevo proveedor.
TRG_PROVEEDOR_DELETE	Después de cada borrado en la tabla Proveedores	Mira si el proveedor borrado se encuentra en la tabla Proveedor_Mas_Rapido. Si es así lo borra y busca quien es ahora el proveedor más rápido para insertarlo en la tabla Proveedor_Mas_Rapido
TRG_PROV_UPD_PERIODO	Después de cada actualización del periodo de entrega en la tabla Proveedores	Mira si el proveedor que se ha modificado está en la tabla Proveedor_Mas_Rapido. Si es así, borra la tabla e inserta los datos del proveedor más rápido. Si no está en la tabla se comprueba si el nuevo periodo de entrega es menor que el que hay en la tabla Proveedor_Mas_Rapido. Si es así, borra los datos de la tabla e inserta los datos del proveedor que se ha modificado. Si el nuevo periodo de entrega es igual que el de la tabla, inserta los datos del proveedor modificado.
TRG_TIPO_DISPOSITIVO_INSERT	Después de cada inserción en la tabla Tipo_Dispositivo	Se insertan en la tabla Dispositivos_Activos_Departamento, todos los departamentos de la empresa para el nuevo tipo de dispositivo creado. Se inserta en la tabla Unidades_Dispositivo_Almacen, el tipo de dispositivo, su descripción y el número de unidades que se encuentran en el almacén igual a 0.
TRG_TIPO_DISPOSITIVO_DELETE	Después de cada borrado en la tabla Tipo_Dispositivo	Se eliminan de la tabla Dispositivos_Activos_Departamento todas las instancias en las que el tipo de dispositivo sea el borrado. Se borra de la tabla Unidades_Dispositivo_Almacen la instancia del tipo de dispositivo borrado
TRG_USUARIO_INSERT	Después de cada inserción en la tabla Usuarios	Incrementa en 1 el total de usuarios y usuarios sin dispositivos de la tabla Usuarios_Dispositivos. Inserta en la tabla Usuarios_Averias el identificador del usuario y el nombre con número de averías igual a 0.
TRG_USUARIO_DELETE	Después de cada borrado en la tabla Usuarios	Borra de la tabla Usuarios_Averias el registro correspondiente al usuario borrado. Actualiza la tabla Usuarios_Dispositivos, decrementando en 1 el campo total usuarios. Si el usuario no tiene asignado todavía ningún dispositivo, decrementa el campo usuarios sin dispositivos de la tabla Usuarios_Dispositivos. Si el usuario se encuentra en la tabla Usuarios_Top10_Averias, se borra la tabla y se calcula de nuevo el ranking de los 10 usuarios que tienen más averías.
TRG_PEDIDO_INSERT	Después de cada inserción en la tabla Pedidos	Siempre que el pedido esté ligado a una orden de inversión, si el año del pedido no existe en la tabla Estadísticas_año, inserta el año y pone el número de dispositivos a 0, el importe del pedido con el importe del pedido y el número de averías a 0. Si el año ya existe, suma el importe del pedido al campo importe_pedido. Si es un pedido que no está ligado a una orden de inversión, inserta el año con el número de dispositivos, el importe del pedido y el número de averías a 0.

TRG_PEDIDO_DELETE	Después de cada borrado en la tabla pedidos	Siempre que el pedido esté ligado a una orden de inversión, resta el importe del pedido al importe de los pedidos para ese año de la tabla Estadísticas_Año.
TRG_PEDIDO_UPD_IMPORTE	Después de actualizar el importe del pedido de la tabla pedidos	Si el pedido corresponde a una orden de inversión y se modifica el importe, actualiza la tabla Estadísticas_Año, sumando al importe de los pedidos, la diferencia entre el importe del pedido actual y el anterior.
TRG_PEDIDO_UPD_ORDEN	Después de actualizar la orden de inversión de la tabla pedidos	Si la orden de inversión es un valor no nulo, actualiza la tabla Estadísticas_Año sumando al importe de los pedidos, el importe del pedido que ha sido modificado
TRG_MOVIMIENTO_INSERT	Después de cada inserción en la tabla Movimientos	<p>Si el movimiento corresponde a un alta: Se incrementa en 1 el campo num_dispositivos de la tabla Unidades_Dispositivo_Almacen para el tipo de dispositivo que se ha dado de alta. Se actualiza la fecha_ultimo_inventario del dispositivo, con la fecha del movimiento de alta.</p> <p>Si el movimiento corresponde a una baja: Se actualiza el campo fecha_baja de la tabla dispositivos para el dispositivo que se da de baja, con la fecha del movimiento. Si el dispositivo no tiene ningún usuario asignado, decrementa en 1 el número de unidades para ese tipo de dispositivo en la tabla Unidades_Dispositivo_Almacen. Si el dispositivo es un portátil, se decrementa en 1 el campo número de portátiles de la tabla Portátiles_Sin_Averia. Si el portátil no ha tenido ninguna avería, se decrementa en 1 el campo portátiles sin avería de la misma tabla.</p> <p>Si el movimiento corresponde a una actualización: Se decrementa en 1 el número de dispositivos que hay en la empresa con una fecha de inventario mayor de un año, actualizando el campo num_dispositivos de la tabla Dispositivos_Fecha_Inventario. Se actualiza el campo fecha_ultimo_inventario de la tabla Dispositivos con la fecha del movimiento.</p> <p>Si el movimiento corresponde a una avería: Incrementa en uno el número de averías de la tabla Estadísticas_Año. Incrementa en 1 el número de averías del usuario que tiene asignado el dispositivo en la tabla Usuarios_Averias. Si la avería corresponde a un dispositivo portátil y es su primera avería, se decrementa en 1 el campo portátiles sin avería de la tabla Portátiles_sin_avería. Finalmente se calcula el ranking de los usuarios que tienen más averías y los inserta en la tabla Usuarios_Top10_Averias.</p> <p>Si el movimiento corresponde a una asignación: Se rellena el campo id_usuario de la tabla dispositivos con los datos del usuario_destino de la asignación y la fecha de instalación con la fecha del movimiento. Se decrementa el número de unidades en el almacén para ese tipo de dispositivo en la tabla Unidades_Dispositivo_Almacen. Si el usuario no tenía asignado ningún otro dispositivo, se decrementa el número de usuarios sin dispositivos en la tabla Usuarios_Dispositivos. Se mira el tipo al que pertenece el dispositivo y el departamento al que ha sido asignado y se actualiza la tabla Dispositivos_Activos_departamento, incrementando en 1 el número de dispositivos para ese departamento y ese tipo de dispositivo. Se cuentan los tipos diferentes de dispositivos que hay en el departamento y se modifica el campo num_tipos_dispositivos de la tabla Inventario_Departamento. Se calcula el valor del inventario que tiene el departamento, sumando el precio de compra de todos los dispositivos activos que hay en el departamento y actualizando el campo valor_inventario de la tabla Inventario_Departamento. Se cuentan los usuarios que tienen asignados más de 5 dispositivos y se actualiza el campo usuarios_mas5_dispositivos de la tabla Usuarios_Dispositivos.</p> <p>Si el movimiento corresponde a una reasignación: Se modifica en la tabla Dispositivos, el id_usuario, poniendo id_usuario_destino. Si el usuario origen no tiene asignado ningún otro dispositivo, se incrementa el número de usuarios sin dispositivos de la tabla Usuarios_Dispositivos. Si el usuario destino no tenía asignado ningún otro dispositivo, se decrementa el número de usuarios sin dispositivos de la tabla Usuarios_Dispositivos. Se mira si el departamento es distinto para el usuario_origen y para el usuario_destino. Si es así, se actualiza la tabla Disp_Activos_departamento, incrementando en 1 el número de dispositivos para el departamento del usuario destino y ese tipo de dispositivo. Se cuentan los tipos diferentes de dispositivos que hay en el departamento destino y se modifica el campo num_tipos_dispositivos de la tabla Inventario_Departamento. Se calcula el valor del inventario que tiene el departamento destino, sumando el precio de compra de todos los dispositivos activos que hay en el departamento y actualizando el campo valor_inventario de la tabla Inventario_Departamento.</p>

		<p>Se decrementa en 1 el número de dispositivos para el departamento del usuario anterior en la tabla Dispositivos_Activos_Departamento.</p> <p>Se recalcula el número de tipos de dispositivo y el valor del inventario para el departamento del usuario anterior y se modifica el campo num_tipos_dispositivos de la tabla Inventario_Departamento.</p> <p>Se mira si hay datos en la tabla Depto_Mas_Reasignaciones, Se recoge el número de reasignaciones que hay en la tabla Depto_Mas_Reasignaciones.</p> <p>Se cuentan el número de reasignaciones del departamento del usuario destino. Si este número es mayor que el anterior, se borran los datos de la tabla Depto_Mas_Reasignaciones para ese año y se insertan en la tabla los datos del departamento. Si el número de reasignaciones es igual se insertan los datos del departamento. Si no hay ningún departamento en la tabla Depto_Mas_Reasignaciones se insertan los datos del departamento del usuario destino.</p> <p>Se cuentan los usuarios que tienen asignados más de 5 dispositivos y se actualiza el campo usuarios_mas5_dispositivos de la tabla Usuarios_Dispositivos.</p>
TRG_MOVIMIENTO_DELETE	Después de cada borrado en la tabla Movimientos	<p>Si el movimiento corresponde a un alta:</p> <p>Se decrementa en 1 el campo num_dispositivos de la tabla Unidades_Dispositivo_Almacen para el tipo de dispositivo que se ha dado de baja. Se actualiza la fecha del último inventario del dispositivo, poniéndole la fecha de alta del dispositivo.</p> <p>Si el movimiento corresponde a una baja: Se actualiza el campo fecha_baja de la tabla dispositivos para el dispositivo que se da de baja, poniéndolo a null</p> <p>Si el dispositivo es un portátil, se incrementa en 1 el campo numero de portátiles de la tabla Portatiles_Sin_Averia. Si el portátil no había tenido ninguna avería, se incrementa en 1 el campo portátiles sin avería de la misma tabla.</p> <p>Si el movimiento corresponde a una actualización:</p> <p>Se incrementa en 1 el número de dispositivos que hay en la empresa con una fecha de inventario mayor de un año, actualizando el campo num_dispositivos de la tabla Dispositivos_Fecha_Inventario.</p> <p>Se actualiza el campo fecha_ultimo_inventario de la tabla Dispositivos con la fecha de alta del dispositivo.</p> <p>Si el movimiento corresponde a una avería:</p> <p>Decrementa en uno el número de averías de ese año en la tabla Estadisticas_Año.</p> <p>Decrementa en 1 el número de averías del usuario que tiene asignado el dispositivo en la tabla Usuarios_Averias.</p> <p>Si la avería corresponde a un dispositivo portátil, se incrementa en 1 el campo portátiles sin avería de la tabla Portatiles_sin_averia. Finalmente se calcula el ranking de los usuarios que tienen más averías y los inserta en la tabla Usuarios_Top10_Averias.</p> <p>Si el movimiento corresponde a una asignación:</p> <p>Se modifican los campos, id_usuario y fecha de instalación de la tabla dispositivos, poniéndolos a null.</p> <p>Se mira si el usuario tenía asignado algún otro dispositivo. Si no es así, se incrementa en 1 el número de usuarios sin dispositivo de la tabla Usuarios_Dispositivos.</p> <p>Se incrementa en 1 el número de unidades en el almacén para ese tipo de dispositivo de la tabla Unidades_Dispositivo_Almacen.</p> <p>Se mira el tipo al que pertenecía el dispositivo y el departamento al que había sido asignado y se actualiza la tabla Dispositivos_Activos_departamento, decrementando en 1 el número de dispositivos para ese departamento y ese tipo de dispositivo.</p> <p>Se cuentan los tipos diferentes de dispositivos que hay en el departamento y se modifica el campo num_tipos_dispositivos de la tabla Inventario_Departamento.</p> <p>Se calcula el valor del inventario que tiene el departamento, sumando el precio de compra de todos los dispositivos que hay en el departamento y actualizando el campo valor_inventario de la tabla Inventario_Departamento.</p> <p>Se cuentan los usuarios que tienen asignados más de 5 dispositivos y se actualiza el campo usuarios_mas5_dispositivos de la tabla Usuarios_Dispositivos.</p> <p>Si el movimiento corresponde a una reasignación:</p> <p>En este caso no se contempla, ya que borrar una reasignación produciría el mismo efecto que dar de alta una nueva reasignación en la que el usuario origen sea el usuario destino y viceversa.</p>
TRG_MOVIMIENTO_UPDATE	Después de cada modificación en la tabla Movimientos	<p>Si el movimiento corresponde a una avería y se modifica el usuario_destino:</p> <p>Incrementa en 1 el número de averías del nuevo usuario que tiene asignado el dispositivo en la tabla Usuarios_Averias y</p> <p>Decrementa en 1 el número de averías del usuario anterior.</p> <p>Si el movimiento corresponde a una asignación y se modifica el usuario_destino:</p>

		<p>Se rellena el campo <code>id_usuario</code> de la tabla <code>dispositivos</code> con los datos del nuevo <code>usuario_destino</code> de la asignación.</p> <p>Se mira si son diferentes los departamentos del usuario anterior y del actual. Si es así, se actualiza la tabla <code>Dispositivos_Activos_departamento</code>, incrementando en 1 el número de dispositivos para el departamento del nuevo usuario y ese tipo de dispositivo.</p> <p>Se cuentan los tipos diferentes de dispositivos que hay en el departamento y se modifica el campo <code>num_tipos_dispositivos</code> de la tabla <code>Inventario_Departamento</code>.</p> <p>Se calcula el valor del inventario que tiene el departamento, sumando el precio de compra de todos los dispositivos que hay en el departamento y actualizando el campo <code>valor_inventario</code> de la tabla <code>Inventario_Departamento</code>.</p> <p>Se mira el tipo al que pertenece el dispositivo y el departamento al que había sido asignado anteriormente y se actualiza la tabla <code>Dispositivos_Activos_departamento</code>, decrementando en 1 el número de dispositivos para ese departamento y ese tipo de dispositivo.</p> <p>Se cuentan los tipos diferentes de dispositivos que hay en el departamento y se modifica el campo <code>num_tipos_dispositivos</code> de la tabla <code>Inventario_Departamento</code>.</p> <p>Se calcula el valor del inventario que tiene el departamento, sumando el precio de compra de todos los dispositivos que hay en el departamento y actualizando el campo <code>valor_inventario</code> de la tabla <code>Inventario_Departamento</code>.</p> <p>Se cuentan los usuarios que tienen asignados más de 5 dispositivos y se actualiza el campo <code>usuarios_mas5_dispositivos</code> de la tabla <code>Usuarios_Dispositivos</code>.</p>
TRG_INVENTARIO_INSERT	Después de cada inserción en la tabla <code>Inventarios</code>	Se calcula el número de dispositivos que hay en la empresa con una fecha de inventario mayor de un año y se actualiza el campo <code>num_dispositivos</code> de la tabla <code>Dispositivos_Fecha_Inventario</code> .
TRG_INVENTARIO_DELETE	Después de cada borrado en la tabla <code>inventarios</code>	Cuando se da de baja un inventario, se calculan los dispositivos que hay en la empresa con una fecha de inventario mayor de un año y se actualiza el campo número de dispositivos de la tabla <code>Dispositivos_Fecha_Inventario</code> .

5. Pruebas

5.1 Carga Inicial de Datos

Para comprobar el correcto funcionamiento del sistema y su ajuste a los requerimientos, se ha desarrollado un script que contiene una serie de sentencias para la carga de datos de las tablas.

Antes de pasar a realizar las pruebas, se realiza una carga de datos inicial. Para la realización de esta carga, se ha desarrollado un script “`carga_inicial.sql`”.

La carga de datos se hará a través de los procedimientos almacenados de Alta que se han diseñado expresamente para ello, por lo que, a la vez que se van rellorando las tablas, se podrá probar su funcionamiento. Las tablas que se rellorarán serán las siguientes: `Tipo_Movimiento`, `Tipo_Dispositivo`, `Proveedores`, `Sedes`, `Departamentos`, `Usuarios`, `Pedidos`, `Lineas_Pedido`, `Dispositivos`, `Presupuestos`, `Recepciones`, `Movimientos` e `Inventarios`.

Al finalizar la carga, se podrá comprobar, como la mayoría de las tablas del repositorio estadístico tienen datos.

5.2 Secuencia de Pruebas

5.2.1 Pruebas de Alta, Baja y Modificación

Una vez que se ha terminado la carga de datos iniciales, se procederá a insertar, borrar y modificar datos con la finalidad de ver si se detectan las restricciones

especificadas. Hay que reseñar que los datos se utilizarán únicamente para realizar las pruebas, pero no serán datos reales, sino que se adaptarán a los términos necesarios para poder comprobar las actualizaciones de las tablas del repositorio estadístico. Tampoco se intentarán probar todas las posibles restricciones, sino las que se consideren más relevantes para el funcionamiento del sistema. El fichero con las pruebas se llama *pruebas.sql*

La secuencia de pruebas planificada es la siguiente:

TIPO_MOVIMIENTO

- Alta de un tipo de movimiento que ya existe
- Alta de un tipo de movimiento nuevo
- Modificación de un tipo de movimiento
- Baja de un tipo de movimiento
- Baja de un tipo de movimiento que no existe
- Baja de un tipo de movimiento que ya había sido dado de baja

TIPO_DISPOSITIVO

- Alta de un tipo de dispositivo que ya existe
- Alta de un tipo de dispositivo nuevo
- Modificación de un tipo de dispositivo
- Baja de un tipo de dispositivo
- Baja de un tipo de dispositivo que no existe
- Baja de un tipo de dispositivo que ya había sido dado de baja

PROVEEDORES

- Alta de un proveedor que ya existe
- Alta de un proveedor nuevo
- Modificación de un proveedor
- Baja de un proveedor
- Baja de un proveedor que no existe
- Baja de un proveedor que ya había sido dado de baja

SEDES

- Alta de una sede que ya existe
- Alta de una nueva sede
- Modificación de una sede
- Baja de una sede
- Baja de una sede que no existe
- Baja de una sede que ya había sido dada de baja

DEPARTAMENTOS

- Alta de un departamento que ya existe
- Alta de un departamento nuevo
- Modificación de un departamento
- Baja de un departamento
- Baja de un departamento que no existe
- Baja de un departamento que ya había sido dado de baja

USUARIOS

- Alta de un nuevo usuario
- Alta de un usuario, asignándole un departamento que no existe
- Alta de un usuario, asignándole una sede que no existe
- Modificación de un usuario asignándole un departamento que no existe
- Modificación de un usuario
- Baja de un usuario que no existe
- Baja de un usuario

PEDIDOS

- Alta de un pedido asignándole un proveedor que no existe
- Alta de un pedido asignándole un departamento que no existe
- Alta de un pedido

- Modificación de un pedido, asignándole un proveedor que no existe
- Modificación de un pedido, asignándole un departamento que no existe
- Modificación de un pedido
- Baja de un pedido que no existe
- Baja de un pedido

LINEAS_PEDIDO

- Alta de una línea de un pedido inexistente
- Alta de una línea con un tipo de dispositivo que no existe
- Alta de una línea
- Modificación de una línea asignándole un tipo de dispositivo que no existe
- Modificación de una línea
- Baja de una línea de pedido que no existe para ese pedido
- Baja de una línea de pedido

DISPOSITIVOS

- Alta de un dispositivo asignándole un tipo de dispositivo que no existe
- Alta de un dispositivo asignándole un usuario que no existe
- Alta de un dispositivo
- Modificación de un dispositivo asignándole un usuario que no existe
- Modificación de un dispositivo asignándole un tipo de dispositivo que no existe
- Baja de un dispositivo que no existe
- Baja de un dispositivo

PRESUPUESTOS

- Alta de un presupuesto de un proveedor que no existe
- Alta de un presupuesto de un pedido que no existe
- Alta de un presupuesto
- Modificación de un presupuesto asignándole un proveedor que no existe
- Modificación del campo “aceptado” para un pedido que ya tiene un presupuesto aceptado
- Baja de un presupuesto que no existe
- Baja de un presupuesto

RECEPCIONES

- Alta de una recepción de un pedido que no existe
- Alta de una recepción de un responsable que no existe
- Alta de una recepción
- Modificación de una recepción asignándole un pedido que no existe
- Modificación de una recepción asignándole un responsable que no existe
- Modificación de una recepción asignándole un responsable que no pertenece al departamento de informática.
- Baja de una recepción que no existe
- Baja de una recepción

MOVIMIENTOS

- Alta de un movimiento asignándole un tipo de movimiento que no existe
- Alta de un movimiento correspondiente a un dispositivo que no existe
- Alta de un movimiento asignándole un usuario_destino que no existe
- Modificación de un movimiento asignándole un tipo de movimiento que no existe
- Modificación de un movimiento asignándole un usuario_destino que no existe
- Modificación de un movimiento asignándole un responsable que no pertenece al departamento de informática.
- Baja de un movimiento que no existe
- Baja de un movimiento

INVENTARIOS

- Alta de un inventario asignándole una sede que no existe
- Alta de un inventario asignándole un coordinador que no existe
- Alta de un inventario
- Modificación de un inventario asignándole una sede que no existe
- Modificación de un inventario asignándole un coordinador que no existe
- Baja de un inventario que no existe

- Baja de un inventario

5.2.2 Pruebas de funcionamiento del Módulo Estadístico

Para poder probar algunas de las funcionalidades del sistema, relacionadas con el módulo estadístico, se procede a introducir en el sistema distintos datos, que nos permitirán comprobar que las tablas tienen los datos correctos. Se necesita introducir más usuarios, más dispositivos, más movimientos de reparación de averías y más movimientos de reasignación de dispositivos para ver si se cumplen todos los requerimientos del sistema. Estas pruebas se incluyen en el mismo fichero que se utiliza para probar los procedimientos de alta, baja y modificación de los procedimientos *pruebas.sql*

5.2.3 Pruebas de Consulta del Módulo Estadístico

Una vez se han rellenado todas las tablas, se puede proceder a realizar las pruebas de consulta sobre el repositorio estadístico. Para ello, se han definido las siguientes consultas:

Consulta 1: Dado un año concreto, número total de dispositivos comprados.

```
*****
SELECT año , num_dispositivos FROM estadisticas_año
WHERE año = :año;
*****
```

Consulta 2: Porcentaje de ordenadores portátiles que no han tenido ninguna avería.

```
*****
SELECT num_portatiles , portatiles_sin_averia , por_sin_averia
FROM portatiles_sin_averia;
*****
```

Consulta 3: Importe total de los pedidos realizados en un año ligados a una orden de inversión.

```
*****
SELECT año , importe_pedidos
FROM estadisticas_año
WHERE año = :año;
*****
```

Consulta 4: Proveedor que tiene un periodo de entrega más corto.

```
*****
SELECT id_proveedor , nombre , periodo_entrega
FROM proveedor_mas_rapido;
*****
```

Consulta 5: Número de dispositivos en uso que tiene un departamento dado, para cada tipo de dispositivo activo.

```
*****
SELECT id_departamento , nombre_departamento , id_tipo_dispositivo , descripcion_tipo ,
num_dispositivos
FROM disp_activos_departamento
WHERE id_departamento = :departamento;
*****
```

Consulta 6: Valor económico del inventario activo para cada departamento.

```
SELECT id_departamento , nombre_departamento , valor_inventario
FROM inventario_departamento;
```

Consulta 7: Número de dispositivos que tienen fecha de inventario físico mayor de un año.

```
SELECT num_dispositivos
FROM DISPOSITIVOS_FECHA_INVENTARIO;
```

Consulta 8: Departamento, que en un año, efectúa más movimientos de tipo reasignación.

```
SELECT id_departamento , nombre_departamento , año , num_reasignaciones
FROM depto_mas_reasignaciones
WHERE año = :año;
```

Consulta 9: Porcentaje de usuarios sin ningún dispositivo asignado.

```
SELECT total_usuarios , usuarios_sin_dispositivos , por_sin_dispositivos
FROM usuarios_dispositivos;
```

Consulta 10: Para cada tipo de dispositivo, número de unidades que hay en el almacén.

```
SELECT id_tipo_dispositivo , descripcion , num_unidades
FROM unidades_dispositivo_almacen;
```

Consulta 11: Departamento de la empresa que tiene más tipos diferentes de dispositivos.

```
SELECT id_departamento , nombre_departamento , num_tipos_dispositivos
FROM depto_mas_tiposdisp;
```

Consulta 12: Número de usuarios que tienen asignados más de 5 dispositivos.

```
SELECT usuarios_mas5_dispositivos
FROM usuarios_dispositivos;
```

Consulta 13: Número de averías registradas en un año concreto.

```
SELECT año , num_averias
FROM estadisticas_año
WHERE año = :año;
```

Consulta 14: Top 10 de usuarios por número de averías registradas sobre sus dispositivos.

```
SELECT id_usuario , nombre_usuario , num_averias  
FROM usuarios_top10_averias;
```

Los campos que aparecen con dos puntos delante, son las variables que debe introducir el usuario, dependiendo del año o departamento del que quiera conocer los datos.

5.2.4 Comprobación de Logs

Aparte de hacer estas comprobaciones, también podemos hacer una select sobre la tabla de log para ver si todos los datos se han introducido correctamente o ha habido algún error.

```
SELECT * FROM LOGS  
  
ORDER BY ID_LOG
```

5.3 Script de Pruebas

5.3.1 Código del Script

El código del script para las pruebas se encuentra en el fichero “pruebas.sql”. Una vez que se ejecute la carga inicial de datos, se ejecutará este fichero, que además de las pruebas para los procedimientos de alta, baja y modificación, tiene las pruebas sobre las consultas del módulo estadístico. En aquellas consultas que había que elegir un año o un departamento se ha elegido uno al azar. La salida de este fichero estará en un fichero “pruebas.log”, que nos permitirá comprobar si los resultados que se han obtenido han sido los esperados. Para ver con mayor claridad los resultados de las pruebas, se han colocado delante de cada salida de la tabla logs, las pruebas que dan lugar a esa salida.

A la hora de hacer el script de pruebas se ha tenido en cuenta el orden lógico que deben tener las inserciones en las tablas. Este orden estará en consonancia con el procedimiento de gestión de activos que exige la aplicación:

En primer lugar se deben de dar de alta los tipos de movimientos, los tipos de dispositivos, las sedes, los proveedores, los departamentos y los usuarios. El siguiente paso, a nivel de funcionamiento, sería hacer un pedido. Este pedido tendría uno o varios presupuestos (solo uno de ellos aceptado) y constaría de una o varias líneas de pedido, con sus tipos de dispositivo correspondiente. Después de hacer los pedidos, hay que recepcionarlos y una vez que se han recepcionado, se comienzan a dar de alta en la tabla movimientos y en la tabla dispositivos, cada uno de los dispositivos. En este momento, se asignarán al almacén (para lo que se ha creado un usuario ficticio ‘US0001’). Cuando llegue el momento de asignarlos a un usuario, habrá que hacer un movimiento de asignación sobre ellos, siendo el usuario_origen el almacén y el usuario_destino el usuario correspondiente. Si queremos reasignarlo a otro usuario,

debemos hacer un movimiento de reasignación, entonces, el usuario_origen será el usuario que tenía el dispositivo y el usuario_destino el nuevo usuario.

Cuando se da de alta un dispositivo mediante un movimiento de alta, se actualiza su fecha de último inventario, con la fecha del movimiento de alta. Cuando se lleva a cabo un inventario, se hace un movimiento de actualización sobre los productos y se cambia su fecha de último inventario.

Si se quiere dar de baja un dispositivo, habrá que hacerlo mediante un movimiento de baja del dispositivo, ya que serán los disparadores sobre la tabla movimientos los que se encargarán de actualizar la tabla dispositivos.

5.3.2 Resultados Esperados

Una vez que se han ejecutado los scripts de prueba, se tendrá que comprobar, sobre todo en las tablas del repositorio estadístico, que los datos que contienen sean los esperados.

En la tabla de Logs, habrá una entrada por cada llamada a un procedimiento almacenado, con sus correspondientes parámetros de entrada y de salida. Todas las llamadas tendrán como parámetro de salida un “OK”, que indicará que las operaciones se han llevado a cabo correctamente, o “ERROR” + una explicación del error producido, en aquellos casos en los que se ha forzado la salida de un error en las pruebas.

En las tablas del repositorio estadístico tendremos que comprobar lo siguiente:

- Estadisticas_Año, debe tener los datos del número de dispositivos comprados, el importe de los pedidos ligados a una orden de inversión y el número de averías para cada uno de los años.
- Portatiles_Sin_Averia, tendrá los datos del número de portátiles total, el número de portátiles sin avería y el porcentaje que representan estos últimos.
- Proveedor_Mas_Rapido, tendrá los datos del proveedor de la empresa que tiene un periodo de entrega más corto.
- Disp_Activos_Departamento, tendrá, para cada departamento, el número de dispositivos en uso que tienen ese departamento para cada tipo de dispositivo activo.
- Inventario_Departamento, tendrá, para cada departamento, el número de tipos diferentes de dispositivos que tiene y el valor del inventario.
- Depto_Mas_TiposDisp, tendrá, el departamento que tiene mayor número de tipos diferentes de dispositivos.
- Dispositivos_Fecha_Inventario, tendrá el número de dispositivos que tienen una fecha de inventario mayor de un año.
- Depto_Mas_Reasignaciones, tendrá, para cada año, los datos del departamento que ha tenido más reasignaciones. Si hay varios departamentos con las mismas reasignaciones deben estar todos.

- **Usuarios_Dispositivos** tendrá el total de usuarios de la empresa, el número de usuarios que todavía no tienen ningún dispositivo asignado y el porcentaje que representan estos últimos. Además tendrá el número de usuarios que tienen asignados más de 5 dispositivos.
- **Unidades_Dispositivo_Almacen**, tendrá, para cada tipo de dispositivo, el número de unidades que hay en el almacén, pendientes de ser asignadas a algún usuario.
- **Usuarios_Averias**, tendrá, para cada usuario, el número de averías que han tenido en los distintos dispositivos que tengan asignados. Si no han tenido ninguna aparecerán con 0.
- **Usuarios_Top10_Averias**, tendrá el ranking de usuarios por el número de averías. Esta tabla podrá contener más de diez usuarios, si tienen el mismo número de averías.

5.3.3 Ejecución de Scripts

Los distintos scripts, que acompañan al proyecto tendrán que ejecutarse en un orden predefinido para que den los resultados esperados:

- En primer lugar habrá que modificar el script “tablespaces.sql”, para adecuar el path a los ficheros con los que se vaya a trabajar.
- Después hay que conectarse a la base de datos con un usuario que tenga privilegios para crear Tablespaces (SYS o SYSTEM) y ejecutar el script con la creación de los tablespaces.
- Crear el usuario TFG, mediante el script “usuario.sql” y conectarse a la base de datos con este usuario.
- Crear las tablas de la base de datos ejecutando el script “tablas.sql”
- Crear los índices de la base de datos ejecutando el script “indices.sql”
- Crear las secuencias, ejecutando el script “secuencias.sql”. Con la ejecución de este script se crearán también los disparadores necesarios para la generación de las secuencias.
- Crear los procedimientos de Alta, Baja y Modificación, ejecutando el script “procedimientos.sql”
- Crear los disparadores necesarios para la actualización del módulo estadístico, ejecutando el script “disparadores.sql”
- Realizar la carga inicial de datos, ejecutando el script “carga_inicial.sql”
- Realizar las pruebas de alta, baja, modificación y las consultas, ejecutando el script “pruebas.sql”
- Realizar las consultas pertinentes sobre la base de datos

La ejecución de estos scripts genera un fichero log, que podrá consultarse para ver los resultados.

6. Valoración Económica del Proyecto

Para hacer la valoración económica del Proyecto, se deben tener en cuenta los distintos roles que tienen que intervenir en su elaboración.

En primer lugar, debe existir un Jefe de Proyecto que será el responsable del proyecto en el día a día. Organizará el trabajo dentro del equipo y en sus relaciones con el cliente. Es la persona que dirige el proyecto desde la planificación hasta el cierre y tiene la responsabilidad del cumplimiento de objetivos en tiempo y coste y de los recursos asignados para ello.

Además del Jefe de Proyecto serán necesarios, como mínimo, un Analista y un Programador. El analista deberá analizar los requisitos del sistema y crear el diseño de la base de datos. El programador se encargará de la instalación del software, la programación de procedimientos, la programación de disparadores y la ejecución de las pruebas.

A continuación, se muestra una tabla con las distintas tareas del proyecto, los roles que las realizan y las horas que deben dedicar a cada una de ellas. El precio por hora para cada uno de los roles es aproximado e incluye el IVA.

Nombre de Tarea	Rol	Horas	Precio/hora	Total
Planificación TFG	Jefe de Proyecto	22	60.00 €	1.320,00 €
Análisis y Diseño				
Análisis de requisitos	Analista	12	50.00 €	600,00 €
Diseño	Analista	31	50.00 €	1.550,00 €
Diseño conceptual		11		
Diseño lógico		10		
Diseño físico		10		
Instalación y configuración del SGBD Oracle	Programador	3	35.00 €	105,00 €
Implementación y pruebas				
Implementación	Programador	32	35.00 €	1.120,00 €
Script de creación de tablas		6		
Construcción de disparadores		14		
Construcción de procedimientos almacenados		12		
Pruebas internas	Programador	35	35.00 €	1.225,00 €
Juego de pruebas		10		
Script de pruebas		10		
Depuración de código		5		
Últimas correcciones		10		
Documentación	Jefe de Proyecto	55	60.00 €	3.300,00 €
Elaboración memoria TFG		35		
Elaboración presentación TFG		20		
			TOTAL	9.220,00 €

Se ha dado por hecho, ya que fue el cliente quién pidió que se desarrollara el proyecto con el SGBD Oracle, que este ya dispone de las licencias necesarias para su uso, por lo que no se ha tenido en cuenta, a la hora de calcular los costes, ningún coste adicional por este concepto.

7. Conclusiones

El proyecto fin de grado es la culminación de varios años de trabajo y esfuerzo y supone un gran reto para cualquier estudiante. En mi caso, el reto fue mayor si cabe,

porque mi trabajo no está relacionado con la Informática. Elegí el área de Bases de Datos Relacionales porque en algún momento de mi vida profesional tuve contacto con ellas y con el SGBD Oracle, y, la verdad, me pareció muy interesante. La primera lección aprendida es, que con esfuerzo y constancia es posible alcanzar los retos.

Una conclusión que he sacado de este trabajo es la importancia que tiene el análisis de requisitos y, en general, las reuniones con el cliente. Después de la lectura del enunciado del trabajo, se suele hacer una interpretación personal de lo que allí se refleja, pero esta interpretación no siempre es acertada. Después de realizar varias preguntas al consultor (en este caso, el cliente) he conseguido clarificar cada uno de los puntos necesarios para que la aplicación realice exactamente lo que el cliente quiere que haga.

En cuanto a la planificación, he aprendido que una buena planificación es muy importante, pero no está exenta de tener que incluir algunas modificaciones, aunque sean pequeñas. En este trabajo, los objetivos planteados inicialmente se han cumplido, pero ha habido algunas desviaciones en cuanto a la duración de las tareas que estaban programadas. Pese a esto, los hitos han podido cumplirse y las entregas se han realizado en plazo.

La metodología prevista en un principio (ciclo de vida en cascada), ha sido la adecuada porque me ha permitido volver atrás con el fin de corregir las deficiencias detectadas durante las distintas etapas, o para completar o aumentar las funcionalidades del sistema. Este ha sido el caso de algunas tablas, que he tenido que introducir a medida que iba avanzando el proyecto porque eran necesarias para conseguir los objetivos y que no estaban en el diseño original o porque podían evitar errores en la aplicación.

Por último, como línea de trabajo futuro, intentaría sacarme una espinita, con la versión de Oracle utilizada. Si bien en un principio, pensé que sería más cómodo utilizar la versión 10 g que ya tenía instalada en el ordenador, al comenzar con la implementación me di cuenta que había sido un error, porque he tenido que hacer manualmente muchas cosas que con la versión 11 g hubiera evitado (por ejemplo, la posibilidad de crear una tabla con un campo calculado a partir de otros campos de la misma tabla). También me hubiera evitado el problema de las “tablas mutantes”, utilizando disparadores compuestos que están disponibles desde la versión 11g.

En la misma línea de trabajo, incluiría los Almacenes de Datos. En ninguna de las asignaturas estudiadas en el grado he visto nada sobre este tema y me ha parecido que es algo muy útil. Mi intención es adentrarme un poco más en el Data Warehouse y en general, en todas las tecnologías que están surgiendo en torno a las bases de datos relacionales, para poder aplicarlas en un futuro.

8. Glosario de Términos

Archivo: Grupo de registros relacionados

Atributos: Unidad básica e indivisible de información acerca de una entidad o una relación.

BD: Abreviatura de Base de Datos.

Base de datos: Colección de registros o archivos relacionados de manera lógica

Base de datos relacional: Una colección de relaciones normalizadas en la que cada relación tiene un nombre distintivo.

Campo: Unidad básica de una base de datos

Claves: Campo o grupo de campos en un registro, que se usa para identificar a este.

Clave Foránea (del inglés Foreign Key): campo o grupo de campos que hacen referencia a la clave primaria de otra tabla o entidad.

Clave Primaria (del inglés Primary Key): campo o grupo de campos que identifican exclusivamente un registro.

Diagrama Entidad_Relación: Modelo de datos que usa símbolos gráficos para mostrar la organización y las relaciones entre los datos.

Diseño conceptual: Primera fase en el desarrollo de la base de datos. Traduce el universo del discurso o historia a un esquema de entidades, atributos e interrelaciones.

Diseño físico: Implementación del diseño lógico en un Sistema de Gestión de Bases de Datos específico.

Diseño lógico: Segunda fase en el desarrollo de la base de datos. Traduce el diseño conceptual a un modelo lógico estándar: las entidades se convierten en relaciones y las interrelaciones, dependiendo de su cardinalidad, en relaciones o propagación de claves.

Disparador (del inglés Trigger): conjunto de instrucciones de la base de datos que tiene esta estructura: llamada, actuación, restricción y acción a ejecutar.

Entidad: Objeto del mundo real que podemos distinguir del resto de los objetos y del cual nos interesan algunas propiedades.

Interrelación: Asociación entre entidades.

Integridad de datos: Es la corrección y exactitud de la información contenida.

Integridad referencial: Es un sistema de reglas que utilizan la mayoría de las bases de datos relacionales para asegurarse que los registros de tablas relacionadas son válidos.

Oracle: Sistema de Gestión de Bases de Datos utilizado en el desarrollo de este proyecto.

Permiso: Privilegio que se otorga a un usuario (o conjunto de usuarios) para realizar una operación determinada sobre un objeto de la Base de Datos.

PL/SQL: Lenguaje de programación procedural para el Sistema de Gestión de Bases de Datos ORACLE. Está almacenado y compilado en la base de datos y corre dentro de ORACLE.

Procedimiento (del inglés procedure): Conjunto de instrucciones que interactúan con la base de datos. Pueden incluir parámetros de entrada y salida.

Rol: Agrupación de privilegios sobre alguno de los componentes de una Base de Datos.

SGBD (Sistema de Gestión de Bases de Datos): Software que gestiona y controla bases de datos. Sus principales funciones son las de facilitar la utilización simultánea a muchos usuarios de tipos diferentes, independizar al usuario del mundo físico y mantener la integridad de los datos.

SQL: Lenguaje estándar de consulta de bases de datos.

UML: Lenguaje unificado de modelación. Es el lenguaje de modelado de sistemas de software más utilizado en la actualidad.

9. Bibliografía

- Material de Bases de Datos I
Jordi Conesa Caralt, Angels Rius Gavidia, M. Elena Rodriguez González
UOC 1ª Edición (febrero 2011)
- Material de Bases de Datos II
Angels Rius Gavidia (coordinadora)
UOC 4ª Edición (septiembre 2015)
- Material de Ingeniería del Software I
Jordi Pradel Mikel, José Raya Martos
UOC 2ª Edición (septiembre 2014)
- Material de Gestión de Proyectos
Jose Ramón Rodriguez (coordinador) , Pere Mariné Jové
UOC 2ª Edición (septiembre 2014)
- Oracle Database Online Documentation 10 g Release 2 (10.2). [En línea]
<http://www.oracle.com/pls/xe102/homepage>
- Guia de Usuario de SQL Developer [En línea]
https://docs.oracle.com/cd/E12151_01/doc.150/e12152.pdf
- Top-N Queries [En línea]
<https://oracle-base.com/articles/misc/top-n-queries>
- Virtual Columns in Oracle Database 11 g Release 1. [En línea]
<https://oracle-base.com/articles/11g/virtual-columns-11gr1>
- Oracle. Tablas mutantes. [En línea]
<http://www.exa.unicen.edu.ar/catedras/dbases1/apuntes/Tablas.Mutantes.pdf>
- Manual de Magic Draw. [En línea]
<http://www.magicdraw.com/files/manuals/MagicDraw%20UserManual.pdf>
- Manual de Gantt Project. [En línea]
<http://es.slideshare.net/reamari/manual-ganttproject>
- Manual de Camtasia Studio [En línea]
<https://aulademedios-vesp.wikispaces.com/file/view/manual-de-camtasia.pdf/402733288/manual-de-camtasia.pdf>

10. Anexos

Como anexo a la memoria, se presenta la siguiente documentación:

- **Manual de Instalación de Oracle Express:** Es una guía de ayuda para realizar la instalación de Oracle. Hay que reseñar, que la instalación en esta guía se lleva a cabo a través de unos cd's de la UOC, pero nos puede servir igualmente si entramos en la página de descargas de oracle, ya que es gratuito, aunque tiene algunas restricciones a nivel de licencia de uso.
- **Guía de Instalación de SQLDeveloper:** Es una guía para la instalación de SQLDeveloper, que es un IDE para trabajar con SQL en el motor de base de datos de Oracle. Esta herramienta es totalmente dependiente de la base de datos, por lo que antes de ejecutarlo (ya que no requiere instalación, solo se descomprime, se ejecuta y se utiliza) hay que asegurarse de tener instalada la base de datos Oracle. Al igual que Oracle Express, puede descargarse desde la página de Oracle.
- **Manual de usuario de SQLDeveloper**
- **Presentación en Power Point con formato ppt**

Esta documentación se encuentra en el fichero `aorbis_anexos.zip`